



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

«Nutriscope με Fitbit: Εφαρμογή σε Android»

Διπλωματική εργασία
της
Αγγελικής Γρηγοροπούλου

Βόλος, Ιανουάριος 2016

[Αυτή η σελίδα είναι σκόπιμα λευκή]



UNIVERSITY OF THESSALY

DEPARTMENT OF ELECTRICAL AND COMPUTER
ENGINEERING

“Nutriscope with Fitbit: Android app”

Diploma Thesis
Aggeliki Grigoropoulou

Volos, January 2016

Επιβλέπων:

Αλκιβιάδης Ακρίτας
Καθηγητής Τ.Η.Μ.Μ.Υ.

Εγκρίθηκε από την διμελή εξεταστική επιτροπή τον Ιανουάριο 2016.

(Υπογραφή)

(Υπογραφή)

.....
Αλκιβιάδης Ακρίτας
Καθηγητής Τ.Η.Μ.Μ.Υ.

.....
Σταμούλης Γεώργιος
Καθηγητής Τ.Η.Μ.Μ.Υ.

(Υπογραφή)

.....
**ΑΓΓΕΛΙΚΗ
ΓΡΗΓΟΡΟΠΟΥΛΟΥ**

Διπλωματούχος Μηχανικός Ηλεκτρονικών Υπολογιστών, Τηλεπικοινωνιών και Δικτύων του Πανεπιστημίου Θεσσαλίας

Copyright © Aggeliki Grigoropoulou, 2016

Με επιφύλαξη παντός δικαιώματος, all rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ' ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Ευχαριστίες

Με την ολοκλήρωση της διπλωματικής εργασίας, θα ήθελα να ευχαριστήσω πρωτίστως τους γονείς μου που στάθηκαν δίπλα μου και με στήριξαν με κάθε δυνατό μέσο στην περάτωση των σπουδών μου. Χωρίς τη συμβολή τους τίποτα από όσα έχω καταφέρει σήμερα δεν θα ήταν πραγματικότητα. Επίσης ένα μεγάλο ευχαριστώ στην αδερφή μου, Παρασκευή Γρηγοροπούλου για την συμπαράστασή της σε όλες τις δύσκολες στιγμές αυτής της πορείας.

Επίσης θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου και επιβλέπων αυτής της διπλωματικής εργασίας, κύριο Αλκιβιάδη Ακρίτα για την εμπιστοσύνη που έδειξε στο πρόσωπό μου με την επιλογή του θέματος και την βοήθεια που μου παρείχε.

Τέλος, ευχαριστώ από καρδιάς όλους τους φίλους μου και ιδιαιτέρως τον Αναστάσιο Βελλή και Γιώργο Κεχαγιά για τις συμβουλές τους και την στήριξή τους στα εύκολα και στα δύσκολα.

Περίληψη

Στόχο της διπλωματικής εργασίας αποτέλεσε ο σχεδιασμός και η υλοποίηση μιας εφαρμογής σε λειτουργικό σύστημα Android που απευθύνεται σε πελάτες διατροφολόγων και σκοπό έχει την καταγραφή των σχετικών βιομετρικών τους στοιχείων και την καλύτερη παρακολούθηση και έλεγχο της προσπάθειάς τους.

Τα δεδομένα που συλλέγονται και αναπαρίστανται στη εφαρμογή προέρχονται από δύο πηγές. Αφενός συγκεντρώνονται από την επικοινωνία της εφαρμογής με τη συσκευή Fitbit flex και αφορούν την καύση θερμίδων κατά τη διάρκεια της ημέρας. Αφετέρου στην εφαρμογή εισάγεται το εβδομαδιαίο πρόγραμμα διατροφής που παρέχει ο διατροφολόγος στον πελάτη του.

Ο χρήστης της εφαρμογής είναι σε θέση να παρακολουθεί το ποσοστό καύσης θερμίδων συγκριτικά με τον ημερήσιο στόχο του και να επιλέγει τα γεύματα που κατανάλωσε από το μενού. Ο διατροφολόγος λαμβάνει στο τέλος κάθε μέρας ένα ηλεκτρονικό μήνυμα με τα δεδομένα του πελάτη του.

Abstract

The aim of this thesis is the design and implementation of an application on Android operating system that is directed to customers of nutritionists to identify the relevant biometric data and assist monitoring and control of their effort.

The data collected and represented in application come from two sources. First collected by the connection of application to the Fitbit flex device concerning burning calories during the day. Secondly into the application is introduced weekly meal plan provided by the nutritionist to his customer.

The user is able to monitor the calorie burn rate compared to the daily target and select meals consumed from the menu. The nutritionist receives at the end of each day an email with customer data.

Πίνακας περιεχομένων

Πίνακας περιεχομένων.....	9
Πίνακας Εικόνων	11
ΚΕΦΑΛΑΙΟ 1	12
1.1 Το κίνημα Quantified Self	12
1.2 Η ανάγκη της υλοποίησης.....	13
1.2.1. Πλεονεκτήματα για τον πελάτη διατροφολόγου.....	13
1.2.2 Πλεονεκτήματα για τον διατροφολόγο	13
1.3 Οργάνωση και δομή της εργασίας	14
ΚΕΦΑΛΑΙΟ 2	15
2.1 Τι είναι το Android?.....	15
2.2 Εργαλεία ανάπτυξης εφαρμογών σε Android.....	15
2.3 Ο AVD Manager του Android Studio	17
ΚΕΦΑΛΑΙΟ 3	19
3.1 Το υλικό της συσκευής Fitbit flex.	19
3.2. Ο αλγόριθμος του Fitbit.....	21
3.3 Διασύνδεση Fitbit με εφαρμογές, προϊόντα και υπηρεσίες	21
3.3.1 Επισκόπηση συστήματος	22
3.3.2 Το Fitbit API	22
3.3.2.1 Τι είναι ένα API?	22
3.3.2.2 Το Fitbit open Web API	22
3.3.2.3 Το πρωτόκολλο πιστοποίησης OAuth.....	23
ΚΕΦΑΛΑΙΟ 4	25
4.1 Αρχιτεκτονική της εφαρμογής Nutriscope	25
4.2 Βασικές έννοιες και ορισμοί.....	26
4.2.1 Βασικές έννοιες εφαρμογής Nutriscope.....	26
4.2.2 Έννοιες και κλάσεις της βιβλιοθήκης Scribe 1.3.5	28
4.3 Η διεπαφή χρήστη της εφαρμογής Nutriscope και περιγραφή της υλοποίησης....	29
4.4 Τα αρχεία της Εφαρμογής Nutriscope	38
4.4.1 Αρχεία .java.....	38
4.4.2 Αρχεία .xml	38
4.4.3 Αρχεία .json.....	39

4.4.4 Αρχείο .txt	40
4.4.5 Αρχείο scribe 1.3.5 (.jar)	40
4.4.6 Logo εφαρμογής.....	40
ΚΕΦΑΛΑΙΟ 5	41
5.1 Σύνοψη.....	41
5.2 Μελλοντικές επεκτάσεις	41
ΚΕΦΑΛΑΙΟ 6	43
Βιβλιογραφία	43

Πίνακας Εικόνων

Εικόνα 1. Το Quantified Self σε μια εικόνα.....	12
Εικόνα 2. AVD Manager	18
Εικόνα 3. Nexus 5 Android Studio emulator	18
Εικόνα 4. Η συσκευή Fitbit flex σε διάφορα χρώματα	19
Εικόνα 5. Επιταχυνσιόμετρο τριών αξόνων.....	20
Εικόνα 6. Ο κινητήρας δόνησης.....	20
Εικόνα 7. Καλώδιο φόρτισης, Προσαρμογέας ασύρματου συγχρονισμού, Αισθητήρας	21
Εικόνα 8. Τα μέρη του συστήματος Fitbit	22
Εικόνα 9. OAuth 1.0a authentication flow.....	24
Εικόνα 10. Διάγραμμα ροής Nutriscope	25
Εικόνα 11. Fitbit Login	29
Εικόνα 12. Έγκριση πρόσβασης Nutriscope στις πληροφορίες-χρήστη.....	30
Εικόνα 13. Απόδοση PIN από το Fitbit.	31
Εικόνα 14. Κομμάτι κώδικα από την κλάση Authentication	32
Εικόνα 15. Μεταφορά στην εφαρμογή Nutriscope.....	32
Εικόνα 16. Dashboard - εμφάνιση ProgressDialog.....	33
Εικόνα 17. Κομμάτι κώδικα για το παράθυρο διαλόγου.	33
Εικόνα 18. Θερμίδες και πρόγραμμα διατροφής στο Nutriscope.....	34
Εικόνα 19. Toast	35
Εικόνα 20. Κομμάτι κώδικα για το toast και τα checkboxes.....	35
Εικόνα 21. Σελίδα αποτελεσμάτων και toast από την επιλογή στοιχείων της προηγούμενης σελίδας (FoodMenu).....	36
Εικόνα 22. Μέθοδος getPrefs().	36
Εικόνα 23. Άνοιγμα υπηρεσίας ηλεκτρονικής αλληλογραφίας (Αριστερά),	37
Εικόνα 24. Κώδικας για κλήση Gmail από το Nutriscope.....	37
Εικόνα 25. Logo εφαρμογής Nutriscope.....	40

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

1.1 Το κίνημα Quantified Self

Το quantified self είναι ένα κίνημα που εστιάζει στην χρήση της τεχνολογίας για την απόκτηση δεδομένων σχετικά με τις καθημερινές πτυχές της ζωής ενός ατόμου, όσον αφορά τις εισόδους που δέχεται το ανθρώπινο σώμα (π.χ. φαγητό ή νερό που καταναλώνει), την κατάσταση στην οποία βρίσκεται (π.χ. επίπεδα οξυγόνου στο αίμα, καρδιακοί παλμοί) και την απόδοσή του (π.χ. ποιότητα ύπνου σε ποσοστό, την φυσική δραστηριότητα). Οι μετρήσεις των παραπάνω δεδομένων γίνονται με τη χρήση φορητών αισθητήρων π.χ. μετρητές παλμών, επιταχυνσιόμετρα. Η δράση αυτή αποσκοπεί στην καλύτερη αυτό-παρακολούθηση του ατόμου και στην ποσοτικοποίηση βιομετρικών στοιχείων με στόχο την βελτίωση της καθημερινής ζωής. Τα εν λόγω δεδομένα μορφοποιούνται και αναπαρίστανται μέσω εφαρμογών που είτε έχουν ως βάση το διαδίκτυο, είτε εγκαθίστανται σε ηλεκτρονικό υπολογιστή, smart phones και tablets.

Ωστόσο, οι συγκεκριμένες πληροφορίες μπορούν να αξιοποιηθούν πλήρως και με σωστό τρόπο κυρίως από ειδικούς που κατέχουν την απαραίτητη γνώση και μπορούν να καθοδηγήσουν τους χρήστες.

Στη διπλωματική εργασία θα περιγράψουμε πως τα δεδομένα φυσικής δραστηριότητας από το βραχιόλι Fitbit flex μπορούν να χρησιμοποιηθούν από διατροφολόγους και τους πελάτες τους μέσω της εφαρμογής Nutriscope για την αύξηση της απόδοσης στη δουλειά των μεν και στον καλύτερο έλεγχο των δε.



Εικόνα 1. Το Quantified Self σε μια εικόνα.

1.2 Η ανάγκη της υλοποίησης

1.2.1. Πλεονεκτήματα για τον πελάτη διατροφολόγου

Το σημαντικότερο πλεονέκτημα για τους πελάτες διατροφολόγων είναι ότι με τη χρήση των δεδομένων στην εφαρμογή προκύπτει ένα πρόγραμμα διατροφής προσαρμοσμένο στα μέτρα του. Το νέο πρόγραμμα διατροφής έχει δημιουργηθεί με βάση τις καθημερινές συνήθειες του χρήστη, δηλαδή τα γεύματα που καταναλώνει και την φυσική του δραστηριότητα ως σύνολο, όχι αποσπασματικά ως άθληση.

Επιπρόσθετα, ο χρήστης απολαμβάνει το προνόμιο της φορητότητας καθώς μπορεί να έχει μαζί του ανά πάσα στιγμή το πρόγραμμα διατροφής του και να επιλέγει τα γεύματα που καταναλώνει ακόμα και όταν δεν βρίσκεται στον προσωπικό του χώρο χωρίς τον κίνδυνο να παραλείψει κάποιο. Ταυτόχρονα μπορεί να προγραμματίσει τα επόμενα γεύματά του ελέγχοντας απλά την συσκευή του, π.χ. όταν βρίσκεται σε διακοπές ή αγοράζει τα απαραίτητα ψώνια για το σπίτι.

Ακόμη, κατόπιν συνεντεύξεων με ψυχολόγους σχετικά με το ζήτημα της διατροφής εξάγεται το συμπέρασμα ότι οι πελάτες διατροφολόγων τηρούν το πρόγραμμά τους επειδή νιώθουν τον έλεγχο που τους ασκεί το επικείμενο ραντεβού με τον διατροφολόγο τους. Με βάση αυτό θεωρούμε ότι η καταγραφή της φυσικής δραστηριότητας εισάγει ένα επιπλέον στοιχείο ελέγχου που παρακινεί τους πελάτες να μην παρεκκλίνουν.

Τέλος, παρατηρώντας το ποσοστό της προόδου του σε καύση θερμίδων με βάση τον ημερήσιο στόχο ο πελάτης μπορεί να ελέγχει την φυσική του δραστηριότητα καλύτερα και να αποκτά το κίνητρο για μεγαλύτερη προσπάθεια όταν αυτό απαιτείται.

1.2.2 Πλεονεκτήματα για τον διατροφολόγο

Ο διατροφολόγος που θα προτείνει στους πελάτες του να χρησιμοποιήσουν την εφαρμογή θα απολαύσει αρκετά πλεονεκτήματα. Κατ' αρχάς, μπορεί να βελτιώσει την απόδοση της δουλειάς του ως εξής

- Πρωτίστως, ελέγχει αποτελεσματικά την φυσική δραστηριότητα των πελατών του αφού δεν βασίζεται πλέον στα λεγόμενά τους, αλλά διαθέτει κοινό μέτρο αξιολόγησης και μέτρησης της άσκησης.
- Δευτερευόντως, παρακολουθεί με μεγαλύτερη λεπτομέρεια την τήρηση του προγράμματος διατροφής από τον εκάστοτε πελάτη εφόσον εκείνοι επιλέγουν τα γεύματα που καταναλώνουν ή μη.

1.3 Οργάνωση και δομή της εργασίας

Η διπλωματική εργασία είναι δομημένη με τον εξής τρόπο:

Στο **κεφάλαιο 2**, περιγράφεται το λειτουργικό σύστημα Android και τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής Nutriscope.

Στο **κεφάλαιο 3**, παρουσιάζεται λειτουργία και η συμβολή της συσκευής Fitbit flex στην εφαρμογή Nutriscope.

Στο **κεφάλαιο 4**, παρουσιάζεται η υλοποίηση της εφαρμογής. Αρχικά, περιγράφεται η αρχιτεκτονική της εφαρμογής Nutriscope. Έπειτα, αναλύονται οι βασικές έννοιες και οι όροι που χρησιμοποιούνται στην εργασία και τέλος αναπτύσσονται με λεπτομέρεια τα επιμέρους αρχεία της εφαρμογής.

Στο **κεφάλαιο 5**, αναφέρεται η σύνοψη της διπλωματικής εργασίας και οι πιθανές επεκτάσεις της εφαρμογής στο μέλλον.

Στο **κεφάλαιο 6**, καταγράφεται η βιβλιογραφία και οι πηγές που χρησιμοποιήθηκαν στην εργασία.

ΚΕΦΑΛΑΙΟ 2

Το λειτουργικό σύστημα Android και εργαλεία ανάπτυξης εφαρμογών σε αυτό.

2.1 Τι είναι το Android?

Το Android είναι ένα ευέλικτο λειτουργικό σύστημα που βασίζεται στον πυρήνα Linux και έχει σχεδιαστεί κυρίως για φορητές συσκευές που διαθέτουν οθόνης αφής, όπως τα smartphones και τα tablets. Αρχικά, το Android αναπτύχθηκε από την Android Inc., το 2005 αγοράστηκε από την Google και το 2007 παρουσιάστηκε για πρώτη φορά, μαζί με τον συνιδρυτή Open Handset Alliance - μια κοινοπραξία εταιρειών υλικού, λογισμικού και τηλεπικοινωνιών επικεντρωμένη στην προώθηση ανοιχτών προτύπων για φορητές συσκευές. Η διεπαφή χρήστη για android λειτουργεί με χειρονομίες αφής όπως το σύρσιμο και το άγγιγμα της οθόνης και με ένα εικονικό πληκτρολόγιο. Η κύρια πλατφόρμα υλικού για το Android είναι η αρχιτεκτονική της ARM (ARMv7 και ARMv8-A) ενώ οι x86 και MIPS αρχιτεκτονικές επίσης υποστηρίζονται επίσημα από τις τελευταίες εκδόσεις Android. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού. Τη σημερινή εποχή, το Android είναι το πιο ευρέως διαδεδομένο λειτουργικό σύστημα. Ωστόσο, η δημοτικότητά του σε συνδυασμό με το γεγονός ότι είναι λογισμικού ανοιχτού κώδικα αποτέλεσε και πόλο έλξης κακόβουλων χρηστών που διενεργούν αυθαίρετες ενέργειες μέσω απομακρυσμένης εκτέλεσης κώδικα και κλιμακώνουν τα προκαθορισμένα (από τους κατόχους των συσκευών) δικαιώματα χρήσης των εφαρμογών. Το παραπάνω αποτελεί και το μεγαλύτερο μειονέκτημα του λειτουργικού συστήματος Android.

2.2 Εργαλεία ανάπτυξης εφαρμογών σε Android

Ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) είναι μια λύση όλα-σε-ένα που επιτρέπει σε ένα προγραμματιστή εφαρμογών να εκτελεί τον κύκλο ανάπτυξης λογισμικού επανειλημμένα και γρήγορα. Αυτός ο κύκλος περιλαμβάνει την συγγραφή, τον προγραμματισμό (coding), το compile, την επαλήθευση (testing), την αποσφαλμάτωση και το πακετάρισμα της εφαρμογής λογισμικού. Για την ανάπτυξη εφαρμογών σε Android η Google υποστήριζε το Eclipse με την προσθήκη ADT (Android Development Tool), ωστόσο το Δεκέμβριο του 2014 κυκλοφόρησε η πρώτη σταθερή έκδοση του Android Studio (v1.0.). Το Android Studio έχει σχεδιαστεί αποκλειστικά για την ανάπτυξη εφαρμογών για Android και είναι βασισμένο στο λογισμικό IntelliJ IDEA της εταιρείας JetBrains, το οποίο είναι ένα περιβάλλον ανάπτυξης εφαρμογών σε java.

Οι κύριες δυνατότητες που παρέχει το Android Studio είναι οι ακόλουθες:

- Gradle-based build: Υποστήριξη αυτόματου μηχανισμού χτισίματος της εφαρμογής.
- Γρήγορες διορθώσεις κατά την υλοποίηση του προγράμματος ειδικές για Android
- Ειδικά lint εργαλεία που ελέγχουν τον κώδικα για σφάλματα, αποδοτικότητα, ασυμβατότητα εκδόσεων λογισμικού κλπ.
- Ενσωμάτωση Pro Guard σχετικά την ασφάλεια και την υπογραφή της εφαρμογής.
- Πλούσιος συντάκτης που παρέχει δυνατότητες, όπως απευθείας σύρσιμο και ενσωμάτωση στην οθόνη εξαρτημάτων διεπαφής χρήστη (UI), έλεγχος του σχεδιασμού σε πολλαπλές διαμορφώσεις οθόνης κ.ά.
- Υποστήριξη για δημιουργία εφαρμογών Android Wear.
- Υποστήριξη για την πλατφόρμα Cloud της Google, επιτρέποντας την ενσωμάτωση με το Google Cloud Messaging και App Engine.
- Υποστήριξη διευρυμένης φόρμας για τις υπηρεσίες της Google
- Υποστήριξη διάφορων ειδών συσκευών

Εναλλακτικά, υπάρχουν αρκετά προγραμματιστικά περιβάλλοντα που επιτελούν τον ίδιο σκοπό με το Android Studio, ενδεικτικά αναφέρουμε τα παρακάτω τα οποία διατίθενται δωρεάν. Ωστόσο, κάποια από αυτά απαιτούν την εγκατάσταση του Android Software Development Kit (SDK) που ανήκει στα εργαλεία της Google.

NAME	LANGUAGE	C-P	URL
AIDE (Android IDE)	HTML5/C/C++	Yes	http://www.android-ide.com/
Application Craft	HTML5	Yes	http://www.applicationcraft.com/
Basic4Android	BASIC	No	http://www.basic4ppc.com/
Cordova	HTML5	Yes	https://cordova.apache.org/
Corona	Lua	Yes	http://coronalabs.com/
Intel XDK	HTML5	Yes	https://software.intel.com/en-us/html5/tools
IntelliJIDEA	Java	No	https://www.jetbrains.com/idea/features/android.html

Kivy	Python	Yes	http://kivy.org/#home
MIT App Inventor	Blocks	Yes	http://appinventor.mit.edu/explore/
Monkey X	BASIC	Yes	http://www.monkeycoder.co.nz/
MonoGame	C#	Yes	http://www.monogame.net/
MoSync	HTML5/C/C++	Yes	http://www.mosync.com/

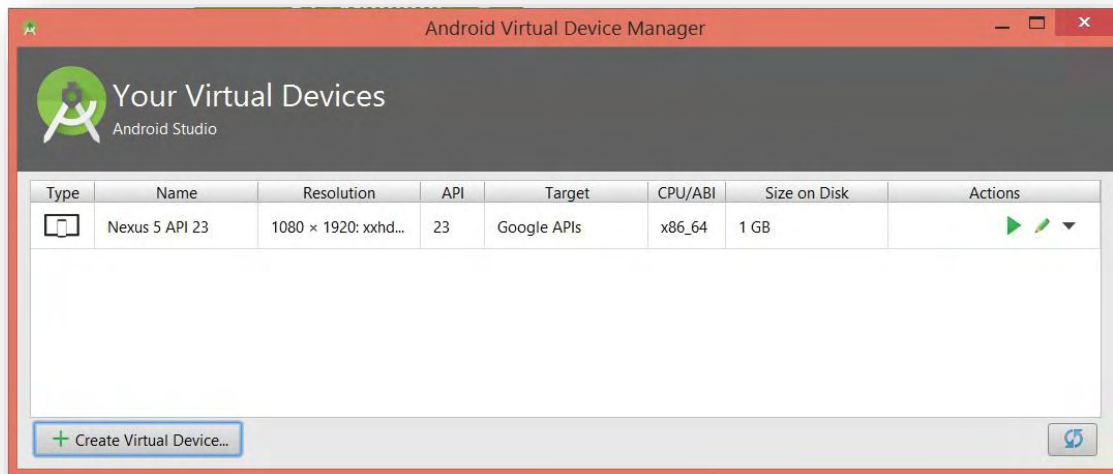
Η υλοποίηση της εφαρμογής Nutriscope έγινε εξολοκλήρου στο Android Studio και οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν ήταν η Java και η XML.

**C-P : cross-platform, μπορεί να χρησιμοποιηθεί σε διαφορετικούς τύπους υπολογιστών ή με διαφορετικά πακέτα λογισμικού.*

2.3 Ο AVD Manager του Android Studio

Ο διαχειριστής AVD (Android Virtual Device) είναι ένα εργαλείο για την δημιουργία και την διαχείριση των εικονικών συσκευών Android με τη χρήση του εξομοιωτή Android. Ο εξομοιωτής Android περιλαμβάνεται μέσα στο SDK και αποτελεί μια εικονική φορητή συσκευή που τρέχει στον υπολογιστή και επιτρέπει την ανάπτυξη και την δοκιμή εφαρμογών χωρίς την παρουσία φυσικής συσκευής. Ο AVD Manager υποστηρίζει τις εξής αρχιτεκτονικές: x86_32, x86_64, MIPS και ARM. Αυτό επιτρέπει στον προγραμματιστή να δοκιμάζει την εφαρμογή του σχεδόν σε όλες τις διαμορφώσεις συσκευών. Ειδικότερα, διαθέτει 21 εικονικές συσκευές για smartphones, 3 συσκευές android wear, 4 συσκευές android tv και 6 συσκευές tablet, όμως ο προγραμματιστής έχει τη δυνατότητα να ενσωματώσει νέα εικονικά προφίλ υλικού. Επιπρόσθετα, επειδή αποτελεί εργαλείο της Google διαθέτει πάντα την νεότερη έκδοση android.

Γενικά, οι εικονικές συσκευές είναι αρκετά πιο αργές από μια φυσική συσκευή. Γι' αυτό η google συνεργάστηκε με την Intel για να διορθώσει το πρόβλημα και να προσφέρει καλύτερη εμπειρία χρήσης στους προγραμματιστές. Συνεπώς, με την επιλογή αρχιτεκτονικής x86_64 υπάρχει η δυνατότητα να τρέξει η εικονική μηχανή android σε « Γρήγορη εικονική λειτουργία». Κατά τη διάρκεια αυτής της λειτουργίας χρησιμοποιείται ο διαχειριστής επιταχυνόμενης εκτέλεσης υλικού της Intel (HAXM), ο οποίος επιταχύνει την απόδοση της εικονικής μηχανής. Το μόνο μειονέκτημα ωστόσο είναι ότι η παραπάνω δυνατότητα προσφέρεται μόνο για την αρχιτεκτονική συνόλου εντολών x86_64.



Εικόνα 2. AVD Manager



Εικόνα 3. Nexus 5 Android Studio emulator

ΚΕΦΑΛΑΙΟ 3

Η συσκευή Fitbit flex

3.1 Το υλικό της συσκευής Fitbit flex.

Το υλικό της συσκευής Fitbit flex αποτελείται από 4 μέρη, το περικάρπιο που διαθέτει τον υποδοχέα του αισθητήρα, τον αισθητήρα, το καλώδιο φόρτισης και τον προσαρμογέα ασύρματου συγχρονισμού.



Εικόνα 4. Η συσκευή Fitbit flex σε διάφορα χρώματα.

Το περικάρπιο διατίθεται σε δύο μεγέθη, μικρό και μεγάλο. Το μικρό έχει περίμετρο 140-177mm (5.4-6.9 inches) και το μεγάλο 161-210mm (6.3-8.3 inches). Το πλάτος είναι 15.14mm (0.6 inches) και το ύψος είναι 10,16mm (0,4 inches).

Η συσκευή διαθέτει μεταξύ άλλων ένα επιταχυνσιόμετρο τριών αξόνων, ως αισθητήρα κίνησης, ένα κινητήρα δόνησης και μνήμη.

- Ένα *επιταχυνσιόμετρο* είναι μια συσκευή που μετατρέπει την κίνηση (επιτάχυνση) του σώματος σε ψηφιακές μετρήσεις (δεδομένα) όταν συνδέεται με το σώμα. Εννοιολογικά, ένα επιταχυνσιόμετρο συμπεριφέρεται ως αποσβεννυμένη μάζα σε ένα ελατήριο. Όταν το επιταχυνσιόμετρο βιώνει μια επιτάχυνση, η μάζα μετατοπίζεται στο σημείο που το ελατήριο είναι σε θέση να επιταχύνει τη μάζα με τον ίδιο ρυθμό όπως το περίβλημα. Η μετατόπιση στη συνέχεια υπολογίζεται για να δώσει την επιτάχυνση. Με την ανάλυση των δεδομένων επιτάχυνσης, οι αισθητήρες παρέχουν λεπτομερείς πληροφορίες σχετικά με τη συχνότητα, τη διάρκεια, την ένταση και τη διαμόρφωση των κινήσεων για να καθορίσει τον αριθμό των βημάτων, την απόσταση που διανύθηκε, τις θερμίδες που καίγονται, και την ποιότητα του ύπνου.



Εικόνα 5. Επιταχυνσιόμετρο τριών αξόνων.

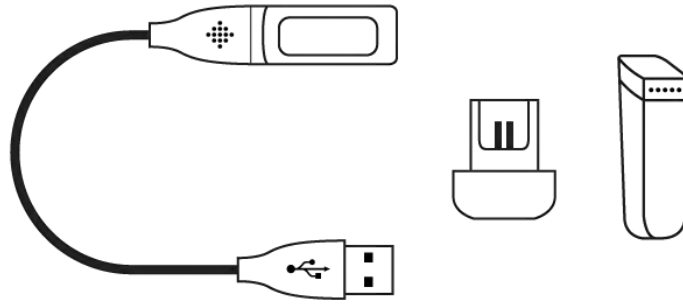
- Ο κινητήρας δόνησης αφενός λειτουργεί ως ξυπνητήρι κατόπιν ρύθμισης και αφετέρου με δύο συνεχόμενα χτυπήματα ο αισθητήρας μπαίνει σε κατάσταση «ύπνου» και αντίστροφα, οπότε με τη δόνηση σηματοδοτείται η αλλαγή από την μια κατάσταση στην άλλη.



Εικόνα 6. Ο κινητήρας δόνησης.

- Η μνήμη της συσκευής καταγράφει λεπτομερή δεδομένα 7 ημερών, λεπτό το λεπτό και ημερήσια δεδομένα για 30 ημέρες.

Το καλώδιο φόρτισης συνδέεται μέσω USB θύρας με τον υπολογιστή και ο προσαρμογέας ασύρματου συγχρονισμού χρησιμοποιείται για τον συγχρονισμό των δεδομένων της συσκευής με τον εξυπηρετητή Fitbit.



Εικόνα 7. Καλώδιο φόρτισης, Προσαρμογέας ασύρματου συγχρονισμού, Αισθητήρας

3.2. Ο αλγόριθμος του Fitbit

Οι αισθητήρες Fitbit έχουν έναν καλά συντονισμένο αλγόριθμο για την καταμέτρηση βημάτων. Ο αλγόριθμος έχει σχεδιαστεί για να αναζητά μοτίβα κίνησης κυρίως των ανθρώπων που περπατούν. Μία προϋπόθεση για ένα μοτίβο κίνησης να αναγνωριστεί ως ένα βήμα είναι η κίνηση να είναι αρκετά μεγάλη. Ο αλγόριθμος υλοποιεί τα παραπάνω με τον καθορισμό ενός κατώτατου ορίου (κατωφλίου). Αν η κίνηση και τα δεδομένα της μέτρησης της επιτάχυνσης πλησιάζουν το κατώτατο όριο, τότε η κίνηση θα υπολογίζεται ως ένα βήμα. Αν δεν πλησιάσει την τιμή κατωφλίου, ο αλγόριθμος δεν θα μετρήσει την κίνηση ως ένα βήμα. Ωστόσο υπάρχουν άλλοι παράγοντες που μπορεί να δημιουργήσουν αρκετή επιτάχυνση για να ξεπεράσουν το όριο και ως εκ τούτου να προκαλέσουν κάποια υπερβολική καταμέτρηση των βημάτων, όπως η ιπασία σε ανώμαλο δρόμο. Επίσης, είναι πιθανό ο αλγόριθμος να μην καταμετρά το απαραίτητο ποσοστό επιτάχυνσης, για παράδειγμα περπάτημα σε μια πολύ μαλακή επιφάνεια, όπως ένα παχύ χαλί.

3.3 Διασύνδεση Fitbit με εφαρμογές, προϊόντα και υπηρεσίες

Οι συσκευές Fitbit είναι σχεδιασμένες έτσι ώστε να φοριούνται από τους ιδιοκτήτες τους όλη την ημέρα. Εξαιτίας αυτού, οι συσκευές Fitbit έχουν σχεδιαστεί για να αποθηκεύουν τα δεδομένα της δραστηριότητας σε τοπικό επίπεδο στη συσκευή. Ενίοτε, ο χρήστης πρέπει να συγχρονίσει τη συσκευή του με την υπηρεσία Fitbit. Μια εφαρμογή για Android, iOS και desktop πλατφόρμες επιτελεί το ρόλο του συγχρονισμού. Κατά το συγχρονισμό, η εφαρμογή προωθεί τα δεδομένα της δραστηριότητας του χρήστη στους Fitbit-servers μέσω του Διαδικτύου, όπου αποθηκεύονται μόνιμα. Τα δεδομένα δραστηριότητας δεν παραμένουν στο smartphone ή τον προσωπικό υπολογιστή του χρήστη αλλά παραλαμβάνονται από τον cloud server του Fitbit κατά τη διάρκεια κάθε συγχρονισμού. Ο συγχρονισμός μεταξύ των συσκευών Fitbit και smartphones γίνεται μέσω Bluetooth. Ειδικότερα, χρησιμοποιείται

ασύρματη τεχνολογία Bluetooth v4.0 (Bluetooth Smart), η οποία περιλαμβάνει κλασσικό Bluetooth, Bluetooth υψηλής ταχύτητας και BTLE (Bluetooth Low Energy) πρωτόκολλα. Ο συγχρονισμός μεταξύ smartphones και την υπηρεσία Fitbit επιτυγχάνεται σε ένα κρυπτογραφημένο session μέσω Διαδικτύου.

3.3.1 Επισκόπηση συστήματος

Στο παρακάτω σχήμα εμφανίζονται τα μέρη του συστήματος Fitbit και ο τρόπος επικοινωνίας μεταξύ τους.



Εικόνα 8. Τα μέρη του συστήματος Fitbit

3.3.2 To Fitbit API

3.3.2.1 Τι είναι ένα API?

Μια διεπαφή προγραμματισμού εφαρμογών (*API - application programming interface*) είναι ένα σύνολο από ρουτίνες, πρωτόκολλα, και εργαλεία για τη δημιουργία λογισμικού και εφαρμογών. Ένα API αποτελεί ένα στοιχείο λογισμικού από την άποψη των λειτουργιών του, τις εισόδους, τις εξόδους και τους βασικούς τύπους, καθορίζοντας τις λειτουργίες που είναι ανεξάρτητες από τις αντίστοιχες εφαρμογές. Ταυτόχρονα επιτρέπει ορισμοί και οι εφαρμογές να ποικίλουν χωρίς να διακυβεύεται η διεπαφή. Ένα καλό API κάνει πιο εύκολη την ανάπτυξη ενός προγράμματος, παρέχοντας όλα τα δομικά στοιχεία, τα οποία στη συνέχεια συνδυάζονται και τοποθετούνται μαζί από τον προγραμματιστή.

3.3.2.2 To Fitbit open Web API

Ένα *open API* (που συχνά αναφέρεται και ως Δημόσιο API) παρέχει στους προγραμματιστές πρόσβαση (μέσω προγραμματισμού) σε μια ιδιόκτητη εφαρμογή λογισμικού. Τα API είναι σύνολα των απαιτήσεων που υποδεικνύουν πως μια εφαρμογή μπορεί να επικοινωνεί και να αλληλοεπιδρά με μια άλλη. Επίσης, επιτρέπει

στους προγραμματιστές να έχουν πρόσβαση σε ορισμένες εσωτερικές λειτουργίες ενός προγράμματος.

Το Fitbit διαθέτει open Web API για την πρόσβαση σε δεδομένα από trackers Fitbit. Έτσι επιτρέπει στους ενδιαφερόμενους προγραμματιστές να αναπτύξουν τις δικές τους εφαρμογές για να έχουν πρόσβαση και να τροποποιήσουν τα δεδομένα του χρήστη Fitbit για λογαριασμό τους. Αρχικά, ο προγραμματιστής καταχωρεί μια εφαρμογή στην σελίδα του Fitbit για προγραμματιστές (dev.fitbit.com/apps) προκειμένου να αποκτήσει τα εξής διαπιστευτήρια: Client (Consumer) Secret και Client (Consumer) Key τα οποία είναι μοναδικά για κάθε εφαρμογή. Βασική προϋπόθεση για το παραπάνω είναι ο χρήστης να διαθέτει λογαριασμό Fitbit. Έπειτα, με αυτά τα διαπιστευτήρια ο προγραμματιστής κάνει κλήση στο API για να αποκτήσει πρόσβαση στα δεδομένα. Η διαδικασία αυτή πραγματοποιείται μέσω του πρωτοκόλλου πιστοποίησης OAuth.

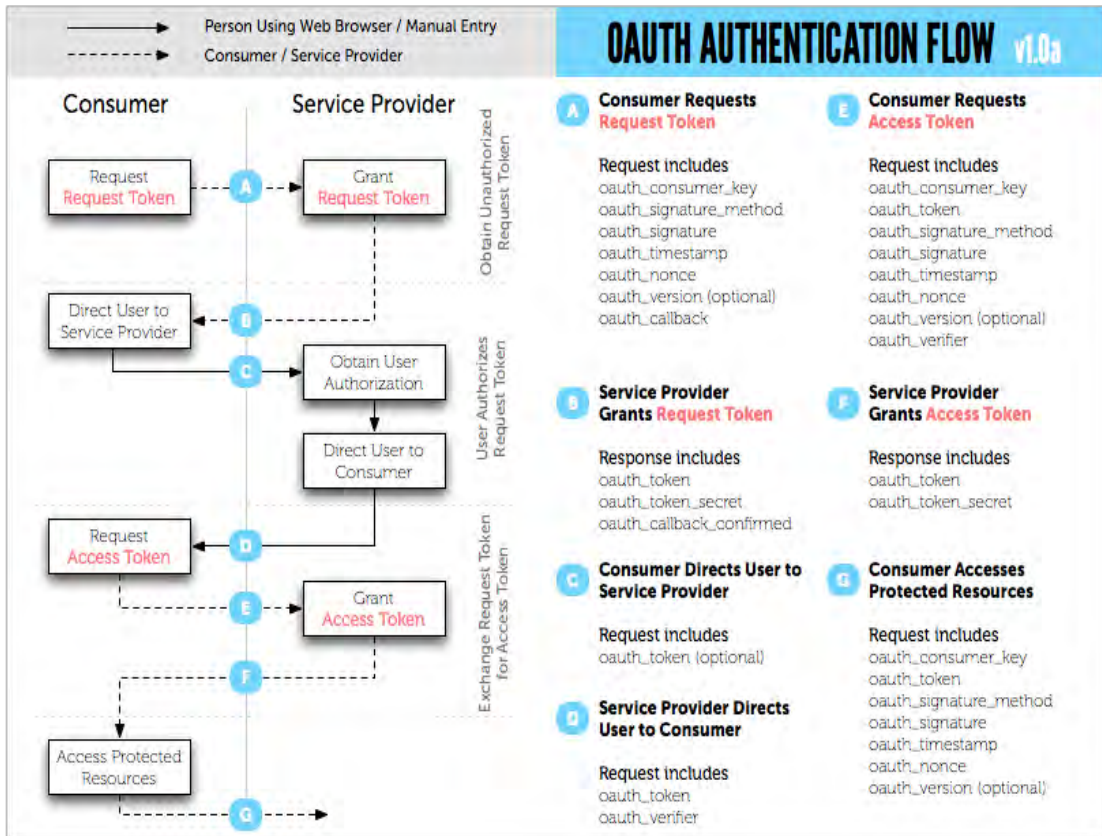
3.3.2.3 Το πρωτόκολλο πιστοποίησης OAuth.

Το πρωτόκολλο OAuth είναι ένα ανοιχτό πρότυπο για τη χορήγηση άδειας, που συνήθως χρησιμοποιείται ως ένας μέσο σύνδεσης σε δικτυακούς τόπους τρίτων που χρησιμοποιούν τους Microsoft, Google, Facebook, Twitter, Fitbit λογαριασμούς τους, χωρίς να εκθέτουν τον κωδικό πρόσβασής τους. Γενικά, το OAuth παρέχει στους πελάτες του μια ασφαλή εξουσιοδοτημένη πρόσβαση σε πόρους του διακομιστή για λογαριασμό του ιδιοκτήτη ενός πόρου. Εν προκειμένω ο προγραμματιστής ακολουθεί τα εξής 5 βήματα για να αποκτήσει πρόσβαση η εφαρμογή που υλοποιεί στα δεδομένα -χρήστη που είναι αποθηκευμένα στον Fitbit server:

1. Ο client ζητά και λαμβάνει προσωρινά πιστοποιητικά από Fitbit.
2. Ο client ανακατευθύνει το χρήστη Fitbit προκειμένου ο χρήστης να πιστοποιήσει την εφαρμογή.
3. Ο χρήστης εγκρίνει την εφαρμογή και το Fitbit ανακατευθύνει τον χρήστη στην εφαρμογή.
4. Ο client ζητά και λαμβάνει συμβολικά διαπιστευτήρια (token credentials) από το Fitbit χρησιμοποιώντας τον ελεγκτή (verifier) που έλαβε.
5. Χρησιμοποιώντας συμβολικά διαπιστευτήρια, ο client κάνει κλήσεις για να προσπελάσει πόρους του Fitbit για λογαριασμό του χρήστη.

Στην εφαρμογή Nutriscope για πλατφόρμα Android η πιστοποίηση των δεδομένων έγινε με βάση την έκδοση 1.0a του πρωτοκόλλου OAuth.

Η όλη διαδικασία πιστοποίησης των 5 βημάτων παρουσιάζεται με λεπτομέρεια στο παρακάτω σχήμα.



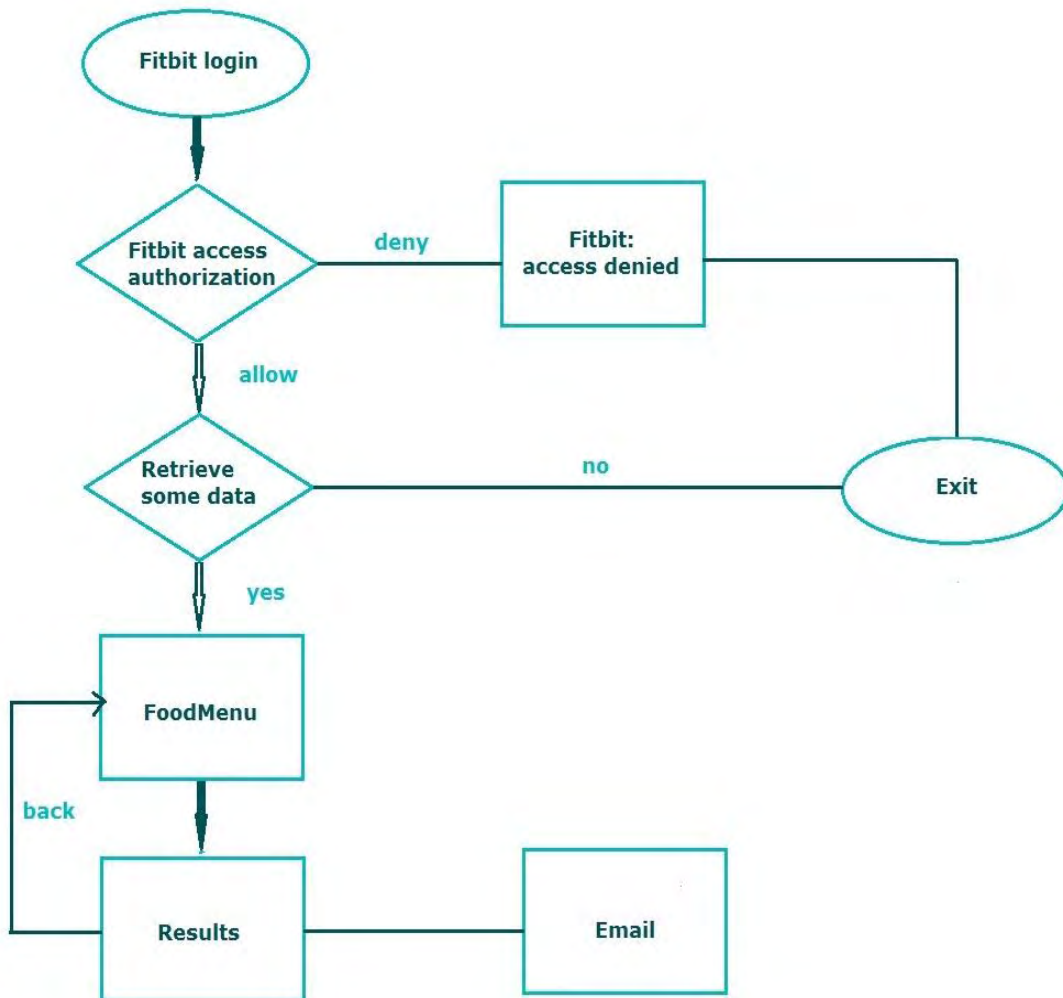
Εικόνα 9. OAuth 1.0a authentication flow

ΚΕΦΑΛΑΙΟ 4

Η εφαρμογή Nutriscope

4.1 Αρχιτεκτονική της εφαρμογής Nutriscope

Στο κεφάλαιο αυτό παρουσιάζεται η δομή της εφαρμογής Nutriscope και ο τρόπος με τον οποίο είναι οργανωμένες οι επιμέρους μονάδες. Στο ακόλουθο διάγραμμα ροής περιγράφεται η βασική δομή και λειτουργία της εφαρμογής.



Εικόνα 10. Διάγραμμα ροής Nutriscope

Στο παραπάνω διάγραμμα τα ελλειπτικά σχήματα αφορούν την έναρξη και τη λήξη του προγράμματος, οι ρόμβοι περιγράφουν υποθέσεις ή αποφάσεις, δηλαδή καταστάσεις στις οποίες η απάντηση είναι ένα ναι ή ένα όχι, και τα παραλληλόγραμμα παρουσιάζουν στάδια επεξεργασίας ή εισόδου/εξόδου δεδομένων.

4.2 Βασικές έννοιες και ορισμοί

Στην παρούσα ενότητα θα παρουσιάσουμε τις βασικές έννοιες και όρους που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής Nutriscope.

4.2.1 Βασικές έννοιες εφαρμογής Nutriscope

Activity: Η κλάση Activity είναι ένα στοιχείο της εφαρμογής το οποίο παρέχει στο χρήστη μια οθόνη αλληλεπίδρασης. Μια τυπική εφαρμογή συνήθως περιέχει αρκετές δραστηριότητες (activities) που αποσκοπούν στο να επιτύχουν κάτι, για παράδειγμα να στείλουν ένα email ή να εκτελέσουν μια κλήση τηλεφωνική.

View: Η κλάση View αποτελεί το βασικό δομικό στοιχείο μιας διεπαφής χρήστη. Ένα View καταλαμβάνει μια ορθογώνια περιοχή στην οθόνη και είναι υπεύθυνη για τον σχεδιασμό και την διαχείριση γεγονότων. Η υποκλάση ViewGroup είναι η βασική κλάση για τα layouts.

- **WebView:** Ένα View για την προβολή ιστοσελίδων.
- **TextView:** Εμφανίζει κείμενο στο χρήστη και προαιρετικά του επιτρέπει να το διαμορφώνει, μέσω της λειτουργίας EditText.
- **ListView:** Εμφανίζει τα στοιχεία σε μια κάθετη λίστα με μια μπάρα κύλισης.

Layouts: Ένα layout ορίζει την οπτική δομή μια διεπαφής χρήστη (UI), όπως ένα UI για μια δραστηριότητα (activity) ή το widget μιας εφαρμογής. Η δήλωση ενός layout γίνεται με 2 τρόπους, είτε με δήλωση στοιχείων διεπαφής χρήστη σε XML, είτε με δήλωσή τους μέσα στην εφαρμογή προγραμματιστικά.

- **Linear Layout:** ευθυγραμμίζει όλα τα στοιχεία-παιδιά σε μία μόνο κατεύθυνση, κάθετα ή οριζόντια.
- **Relative Layout:** εμφανίζει τα στοιχεία-παιδιά σε σχετικές θέσεις αναφορικά με τα στοιχεία-γονείς (ευθυγραμμισμένα ως προς τα κάτω, αριστερά ή στο κέντρο) ή τα στοιχεία-αδέλφια (όπως στα αριστερά του ή στα δεξιά του).

Intents: Μία παθητική δομή δεδομένων που κατέχει μια αφηρημένη περιγραφή της δράσης που πρέπει να εκτελεστεί. Σημαντικότερη χρήση του είναι κατά την έναρξη των δραστηριοτήτων, όπου μπορεί να θεωρηθεί ως ο συνδετικός κρίκος μεταξύ των δραστηριοτήτων.

Thread: Ένα νήμα είναι μια ταυτόχρονη μονάδα εκτέλεσης. Έχει τη δική του στοίβα κλήσεων για τις μεθόδους που καλούνται, τα ορίσματα και τις τοπικές μεταβλητές.

- **Runnable:** Αντιπροσωπεύει μια εντολή που μπορεί να εκτελεστεί. Συχνά χρησιμοποιείται για να εκτελέσει κώδικα σε ένα διαφορετικό νήμα.

ProgressBar: Οπτική ένδειξη της προόδου σε κάποια εργασία. Εμφανίζει μια μπάρα στο χρήστη που αντιπροσωπεύει πόσο η εργασία έχει προχωρήσει.

ProgressDialog: Ένα παράθυρο διαλόγου εμφανίζει μια ένδειξη προόδου και ένα προαιρετικό μήνυμα κειμένου ή ένα view. Μόνο ένα μήνυμα κειμένου ή ένα view μπορεί να χρησιμοποιηθεί ταυτόχρονα. Ο διάλογος ακυρώνεται μόλις πατηθεί κάποιο πλήκτρο.

Button: Αντιπροσωπεύει ένα widget πατήματος ενός κουμπιού.

FloatingActionButton: “Επιπλέοντα” κουμπιά ενεργειών χρησιμοποιούνται για έναν ειδικό τύπο της προώθησης δράσεων. Διακρίνονται από ένα εικονίδιο κύκλο, αιωρούνται πάνω από τη διεπαφή χρήστη (UI) και έχουν ειδικές συμπεριφορές που σχετίζονται με την κίνηση αλλαγής από μια εικόνα σε μια άλλη με χρήση τεχνικών animation και την μεταφορά σταθερού σημείου.

Toast: παρέχει απλή ανάδραση σχετικά με μια πράξη σε ένα μικρό αναδυόμενο παράθυρο. Γεμίζει μόνο το ποσό του χώρου που απαιτείται για το μήνυμα και η τρέχουσα δραστηριότητα παραμένει ορατή και διαδραστική.

External Storage: πρόσβαση σε εξωτερική αποθήκευση, δηλαδή αποθήκευση σε SD κάρτα ή στη μνήμη της συσκευής. Τα αρχεία μπορούν να τροποποιηθούν από τον χρήστη και να μεταφερθούν σε υπολογιστή.

Internal Storage: πρόσβαση σε εσωτερική αποθήκευση, δηλαδή στην μνήμη της συσκευής στον χώρο που είναι δεσμευμένος για την εφαρμογή. Τα αρχεία είναι ιδιωτικά και δεν μπορούν να τροποποιηθούν από τον χρήστη.

- **SharedPreferences:** παρέχει ένα γενικό πλαίσιο που επιτρέπει την αποθήκευση και την ανάκτηση ζευγών κλειδιών-τιμών των βασικών τύπων δεδομένων, όπως booleans, floats, ints, long, και strings. Αυτά τα δεδομένα διατηρούνται μεταξύ των συνεδριών του χρήστη ακόμα και αν η εφαρμογή τερματιστεί.

ArrayList: υλοποίηση μιας λίστας που υποστηρίζεται από έναν πίνακα.

Calendar: είναι μια αφηρημένη κλάση βάσης για τη μετατροπή μεταξύ ενός αντικειμένου Ημερομηνία (Date) και μια σειρά ακέραιων πεδίων, όπως το έτος, ο μήνας, η ημέρα, η ώρα, και ούτω καθεξής. Υποκλάσεις του Calendar ερμηνεύουν μια ημερομηνία, σύμφωνα με τους κανόνες ενός συγκεκριμένου ημερολογιακού συστήματος.

Locale: αντιπροσωπεύει έναν συνδυασμό γλώσσας / χώρας / μεταβλητής, χρησιμοποιούνται για να μεταβάλλουν την παρουσίαση των πληροφοριών, όπως αριθμούς ή ημερομηνίες για να ταιριάζουν με τις συμβάσεις στην περιοχή που περιγράφουν.

Log: API για την αποστολή αρχείου καταγραφής στην έξοδο. Σε γενικές γραμμές, χρησιμοποιούνται οι μέθοδοι Log.v(), Log.d(), Log.i(), Log.w() και Log.e(), τα οποία αντιπροσωπεύουν αντιστοίχως τα εξής: VERBOSE, DEBUG, INFO, WARN, ERROR.

Exception: η υπερκλάση όλων των κλάσεων που αντιπροσωπεύουν ανακτήσιμες εξαιρέσεις. Όταν οι εξαιρέσεις προκαλούνται μπορούν να συλλεχθούν από κώδικα της εφαρμογής. Στην εφαρμογή Nutriscope εξαιρέσεις προκαλούνται από τον κώδικα των νημάτων, των αρχείων και προβλημάτων που σχετίζονται με το JSON API.

JSONObject: αντιπροσωπεύει έναν τύπο αντικειμένου JSON. Ένα αντικείμενο αποτελείται από ζεύγη ονόματος-τιμής όπου τα ονόματα είναι strings (αλφαριθμητικά), και οι τιμές είναι οποιοδήποτε άλλο είδος JsonElement.

JavaScriptInterface: annotation (σχολιασμός) που επιτρέπει την έκθεση μεθόδων σε JavaScript.

ArrayAdapter: Ένας συγκεκριμένος προσαρμογέας βάσης (BaseAdapter) που υποστηρίζεται από μια σειρά αυθαίρετων αντικειμένων.

4.2.2 Έννοιες και κλάσεις της βιβλιοθήκης Scribe 1.3.5

Με την προσθήκη της βιβλιοθήκης *scribe 1.3.5* στην εφαρμογή χρησιμοποιήθηκε ένα σύνολο στοιχείων για την περάτωση της πιστοποίησης των δεδομένων μέσω του πρωτοκόλλου OAuth 1.0a. Αναφέρουμε ενδεικτικά τα παρακάτω:

OAuthService: Το κύριο αντικείμενο Scribe. Μια υπηρεσία υπεύθυνη για την ανάκτηση του αιτήματος και των access tokens και για την υπογραφή των HTTP αιτημάτων (requests).

OAuthRequest: Η αναπαράσταση μιας αίτησης OAuth HTTP. Προσθέτει λειτουργικότητα που σχετίζεται με το OAuth στην κλάση Request.

Verifier: Αντιπροσωπεύει έναν κωδικό OAuth επιβεβαιωτή. Στην διεπαφή χρήστη στην εφαρμογή Nutriscope αναφέρεται ως PIN.

Verb: Μια απαρίθμηση που περιέχει τις πιο συνηθισμένες ενέργειες HTTP. Οι ενέργειες HTTP είναι οι εξής: GET, POST, PUT, DELETE, HEAD, OPTIONS, TRACE, PATCH.

DefaultApi10a: Προκαθορισμένη υλοποίηση του πρωτοκόλλου OAuth, έκδοση 1.0a. Αυτή η κλάση έχει ως στόχο να επεκταθεί με συγκεκριμένες υλοποιήσεις του API, παρέχοντας τα τελικά σημεία (endpoints) και το endpoint-http-verbs.

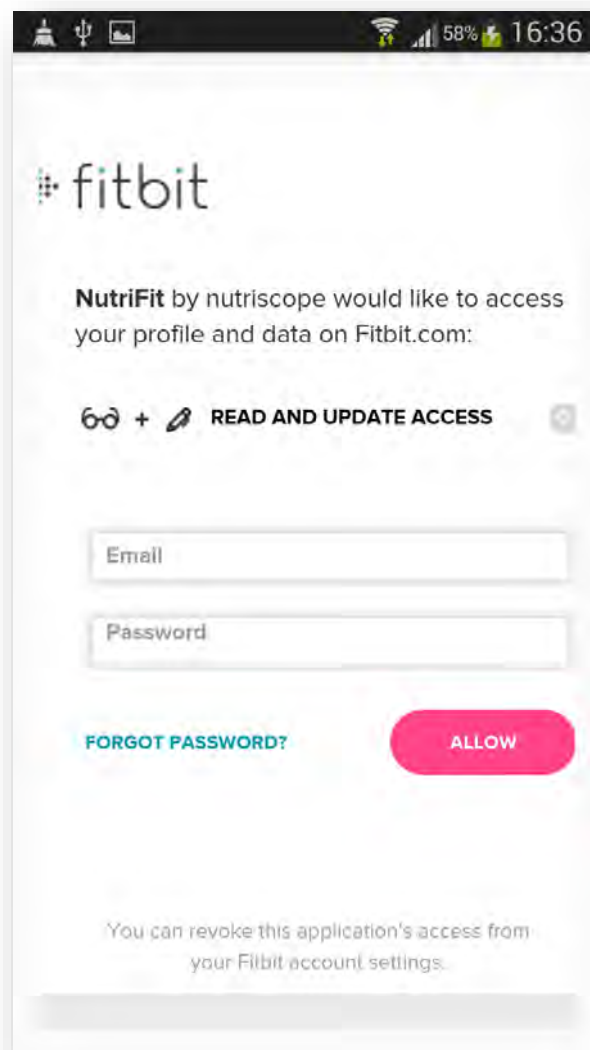
Scribe.model.Response: Αντιπροσωπεύει μια απόκριση HTTP.

scribe.model.Token: Αντιπροσωπεύει ένα token OAuth (είτε token αιτήματος είτε προσβάσεως) και το μυστικό του.

ServiceBuilder: Υλοποίηση του μοτίβου δόμησης, με μια διεπαφή που δημιουργεί ένα OAuthService αντικείμενο.

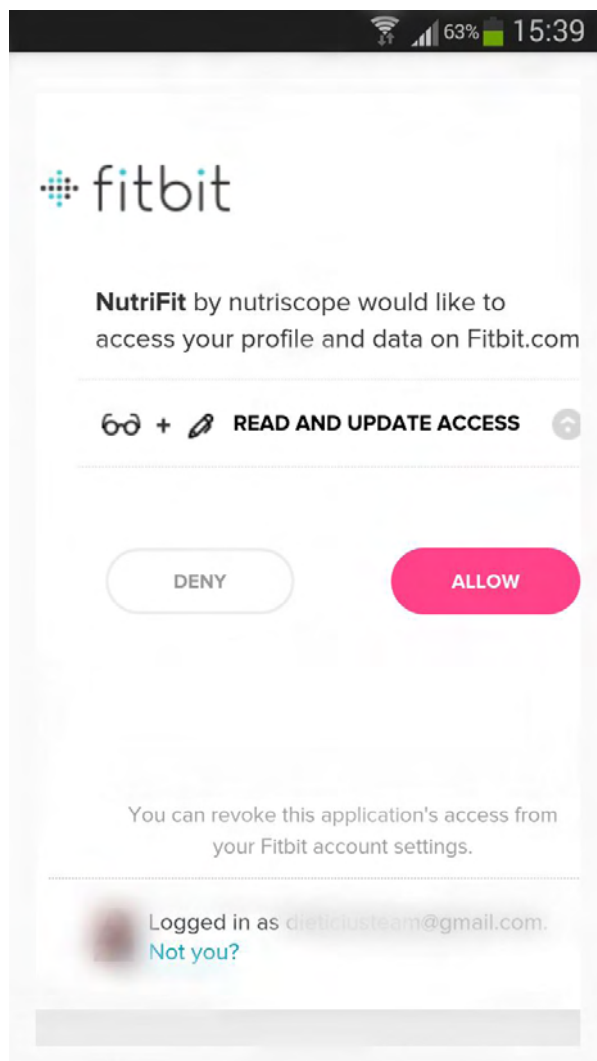
4.3 Η διεπαφή χρήστη της εφαρμογής Nutriscope και περιγραφή της υλοποίησης.

Ο χρήστης του Nutriscope την πρώτη φορά που κάνει είσοδο στην εφαρμογή μεταφέρεται στην ιστοσελίδα του Fitbit μέσω της λειτουργίας WebView που προβάλλει ιστοσελίδες για λογαριασμό της εφαρμογής. Σε αυτό το σημείο συμπληρώνει το όνομα-χρήστη και τον κωδικό πρόσβασης που έχει παραλάβει με την συσκευή Fitbit. Έτσι επιτρέπει την πρόσβαση του Nutriscope στις πληροφορίες της φυσικής του δραστηριότητας που έχουν συλλεχθεί από τον αισθητήρα.



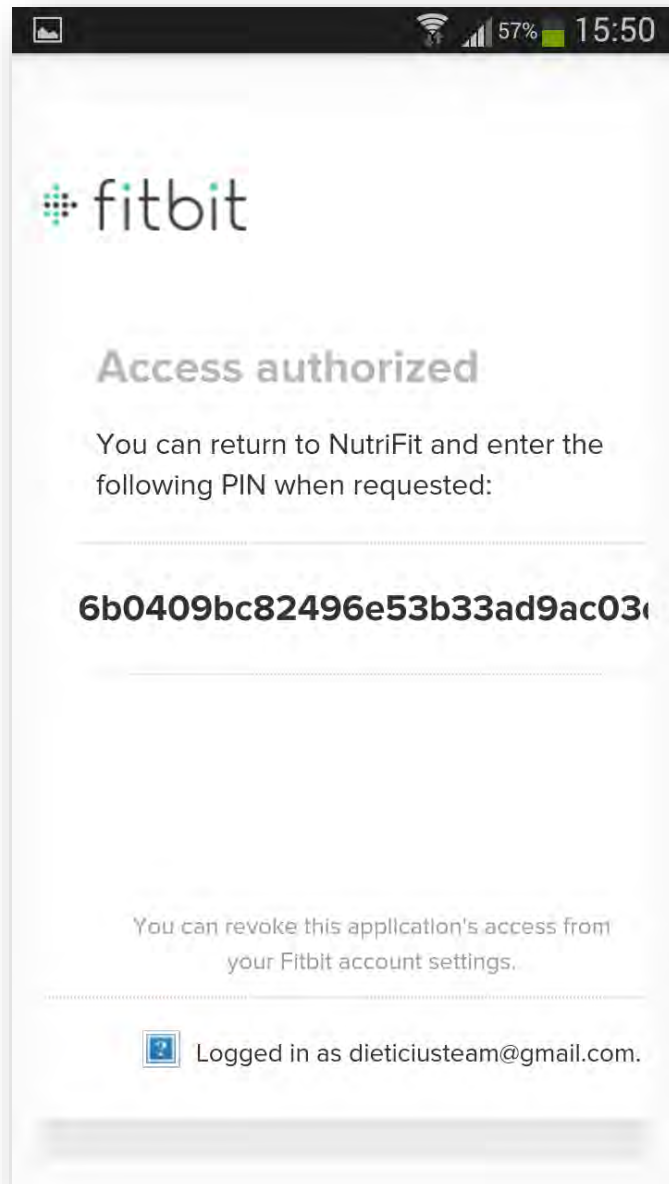
Εικόνα 11. Fitbit Login

Κάθε επόμενη φορά που ο χρήστης θα χρησιμοποιεί την εφαρμογή, αν έχει επιλέξει την πρώτη φορά απομνημόνευση των διαπιστευτηρίων του (όνομα-χρήστη και κωδικό πρόσβασης) θα μεταφέρεται στην εξής σελίδα:



Εικόνα 12. Έγκριση πρόσβασης Nutriscope στις πληροφορίες-χρήστη

Όταν εγκριθεί η πρόσβαση, δίνεται ένα PIN για το συγκεκριμένο session το οποίο χρησιμοποιείται στο επόμενο βήμα της πιστοποίησης των διαπιστευτηρίων του χρήστη. Ο χρήστης μεταφέρεται στην ακόλουθη σελίδα για λίγα δευτερόλεπτα και έπειτα πίσω στην εφαρμογή, όπου και ολοκληρώνεται η διαδικασία της πιστοποίησης.



Εικόνα 13. Απόδοση PIN από το Fitbit.

Στην παραπάνω εικόνα παρουσιάζεται το PIN που αποδίδεται στο session. Ο χρήστης παραμένει λίγα δευτερόλεπτα στην σελίδα αυτή και με τη χρήση της διεπαφής JavaScript αυτόματα μεταφέρεται στην εφαρμογή.

```

import android.support.v7.app.AppCompatActivity;
import android.webkit.JavascriptInterface;
import android.webkit.WebView;
import android.webkit.WebViewClient;

import org.scribe.builder.ServiceBuilder;

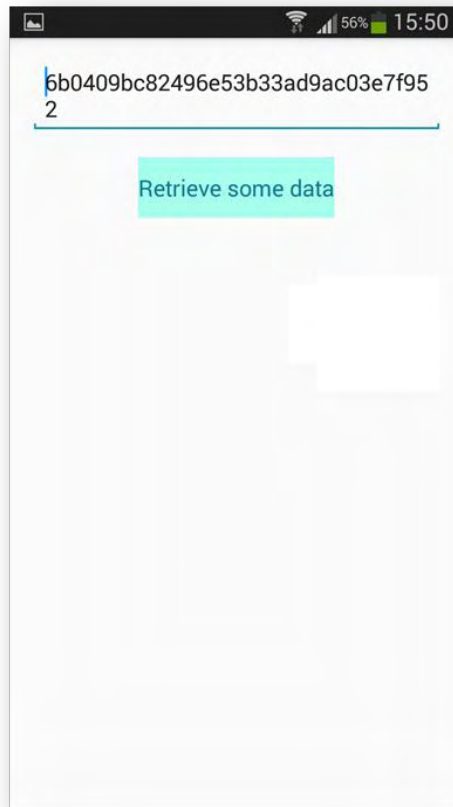
@SuppressLint("SetJavaScriptEnabled")
public class Authentication extends AppCompatActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_authentication);
        final WebView wvAuthorise = (WebView) findViewById(R.id.wvAuthorise);
        wvAuthorise.getSettings().setJavaScriptEnabled(true);
    }
}

```

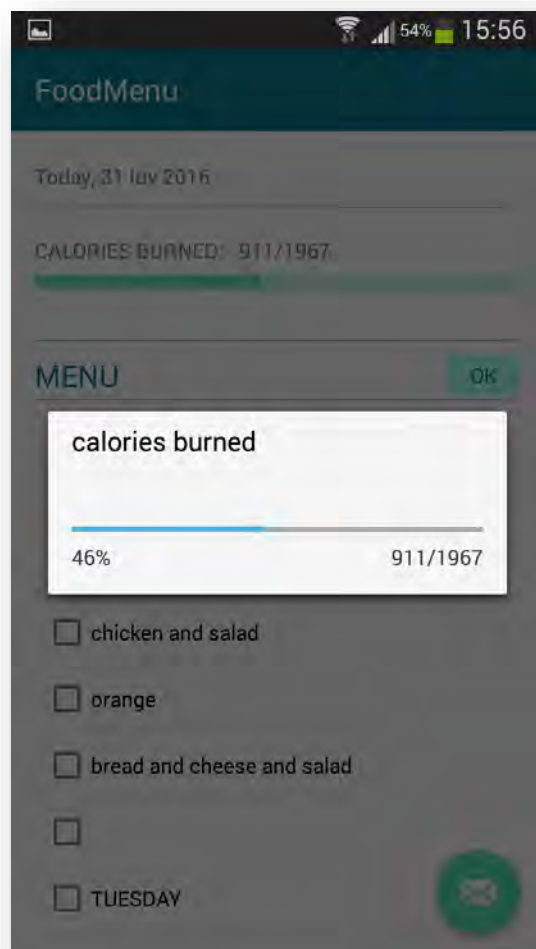
Εικόνα 14. Κομμάτι κώδικα από την κλάση Authentication



Εικόνα 15. Μεταφορά στην εφαρμογή Nutriscope.

Σε αυτό το σημείο, εμφανίζεται το PIN από το προηγούμενο βήμα το οποίο συμπληρώνεται αυτόματα. Ο χρήστης επιλέγει την ανάκτηση των πληροφοριών, ολοκληρώνεται το τελευταίο βήμα της πιστοποίησης του πρωτοκόλλου OAuth και

μεταφέρεται στο βασικό μέρος της εφαρμογής, Dashboard, όπου και επιτελείται η βασική λειτουργία της εφαρμογής.



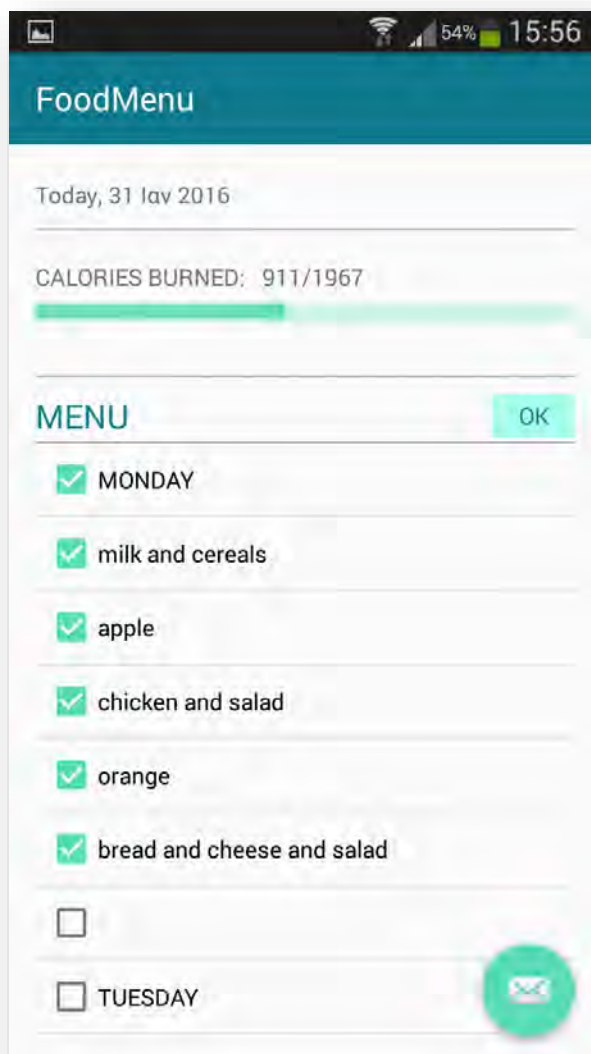
Εικόνα 16. Dashboard - εμφάνιση ProgressDialog.

Με την επιτυχή σύνδεση χρήστη και την επιβεβαίωση των στοιχείων του εμφανίζεται στην οθόνη του χρήστη ένα παράθυρο-διάλογος, το οποίο παρουσιάζει την πρόοδο του χρήστη σε καύση θερμίδων συγκριτικά με τον ημερήσιο στόχο που έχει οριστεί από τον διατροφολόγο του. Ο χρήστης μπορεί να πατήσει εκτός του πλαισίου του διαλόγου και να τον εξαφανίσει.

```
progress=new ProgressDialog(this);  
progress.setMessage("calories burned");  
progress.setProgressStyle(ProgressDialog.STYLE_HORIZONTAL);  
progress.setMax(goalCalories);  
progress.show();
```

Εικόνα 17. Κομμάτι κώδικα για το παράθυρο διαλόγου.

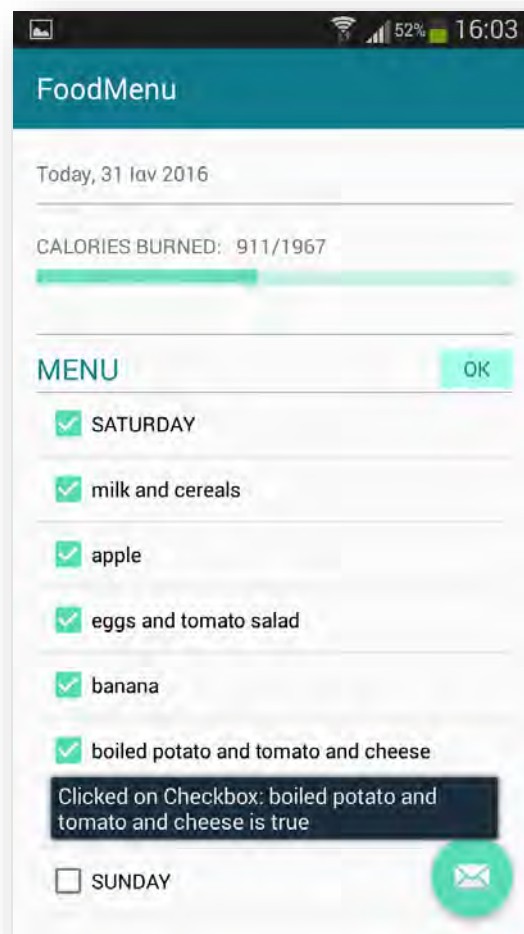
Στην παρακάτω εικόνα εμφανίζεται πάνω πάνω η τρέχουσα ημερομηνία, στη συνέχεια μια μπάρα προόδου με τις θερμίδες που έχει κάψει ο χρήστης και τέλος το εβδομαδιαίο πρόγραμμα διατροφής. Ο πελάτης του διατροφολόγου αποκτά το πρόγραμμα διατροφής στην εφαρμογή του ως εξής: ο διατροφολόγος συνδέει με καλώδιο USB τον υπολογιστή του με την συσκευή του χρήστη και στέλνει ένα αρχείο .txt με το πρόγραμμα διατροφής το οποίο έχει ήδη διαμορφώσει. Το πρόγραμμα αποθηκεύεται στη μνήμη της συσκευής, είτε στην SD κάρτα. Η εφαρμογή εντοπίζει το αρχείο μέσω του αντίστοιχου μονοπατιού που έχει οριστεί από τον προγραμματιστή και το εμφανίζει.



Εικόνα 18. Θερμίδες και πρόγραμμα διατροφής στο Nutriscopε.

Ο χρήστης επιλέγει την ημέρα και τα γεύματα που κατανάλωσε κατά τη διάρκεια της. Το μενού έχει μπάρα κύλισης, αφού στην ουσία πρόκειται για ένα ListView και έτσι ο χρήστης μπορεί να επιλέξει το μενού της εκάστοτε ημέρας μέσα στην εβδομάδα.

Μόλις ο χρήστης επιλέξει ένα αντικείμενο από τη λίστα του μενού εμφανίζεται το αντίστοιχο μήνυμα στην οθόνη του (Toast). Αντίστοιχο μήνυμα εμφανίζεται και όταν επιλέξει ένα αντικείμενο το οποίο ήταν ήδη επιλεγμένο.

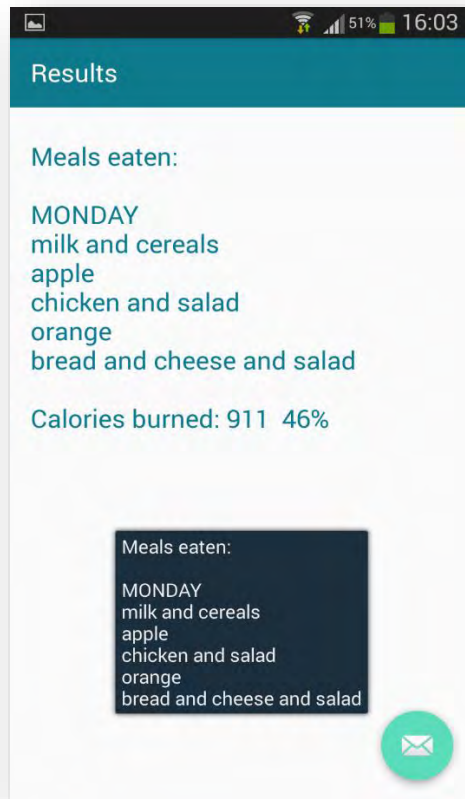


Εικόνα 19. Toast

```
holder.name.setOnClickListener( (v) -> {  
    CheckBox cb = (CheckBox) v ;  
    Meals meal = (Meals) cb.getTag() ;  
  
    Toast.makeText(getApplicationContext(),  
        "Clicked on Checkbox: " + cb.getText() +  
        " is " + cb.isChecked(),  
        Toast.LENGTH_LONG).show() ;  
    meal.setSelected(cb.isChecked()) ;  
});
```

Εικόνα 20. Κομμάτι κώδικα για το toast και τα checkboxes.

Πιέζοντας το OK ο χρήστης μεταφέρεται στη σελίδα Results της εφαρμογής όπου είναι συγκεντρωμένα τα αποτελέσματα της διατροφής και της φυσικής του δραστηριότητας. Σε αυτό το σημείο μπορεί να επιλέξει το floating button, δηλαδή το κυκλικό κουμπί με το σύμβολο της αλληλογραφίας και να στείλει τα αποτελέσματα στον διατροφολόγο του.

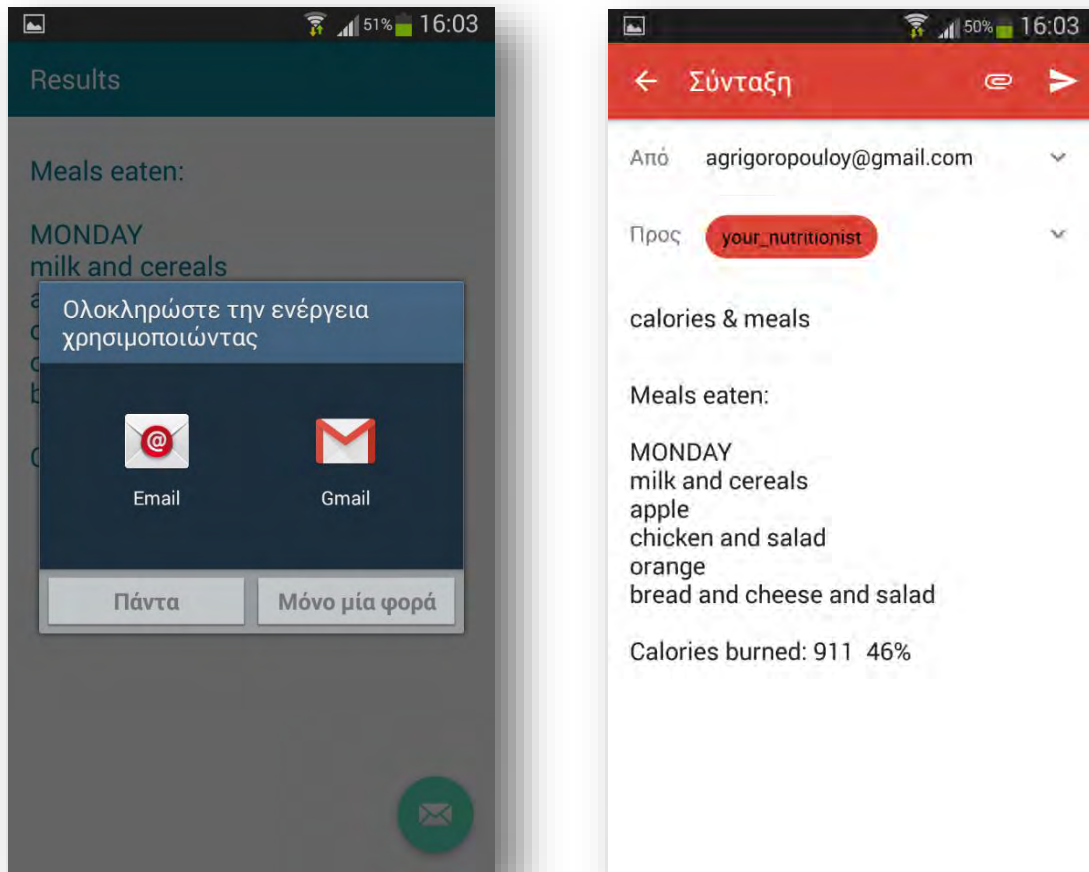


Εικόνα 21. Σελίδα αποτελεσμάτων και toast από την επιλογή στοιχείων της προηγούμενης σελίδας (FoodMenu).

Οι επιλογές του χρήστη αποθηκεύονται σε ένα αλφαριθμητικό με SharedPreferences στη σελίδα FoodMenu και ανακτώνται στην σελίδα Results με τη μέθοδο getPrefs().

```
public void getPrefs()
{
    SharedPreferences preferences = this.getSharedPreferences("general_settings", Context.MODE_PRIVATE);
    if(preferences!=null)
    {
        String todayMeals = preferences.getString(prefs,prefs);
        // System.out.println("meals eaten: " + todayMeals);
    }
}
```

Εικόνα 22. Μέθοδος getPrefs().



Εικόνα 23. Άνοιγμα υπηρεσίας ηλεκτρονικής αλληλογραφίας (Αριστερά), Υπηρεσία Gmail (Δεξιά).

Παραπάνω αποτυπώνεται η ενέργεια της αποστολής των αποτελεσμάτων με υπηρεσία Gmail. Το θέμα του μηνύματος, όπως και το περιεχόμενο συμπληρώνονται αυτόματα από την εφαρμογή. Ωστόσο, ο χρήστης είναι σε θέση να συμπληρώσει τυχόν παρατηρήσεις ή σχόλια στο μήνυμά του προς τον διατροφολόγο του.

```

TextView myTextView = (TextView) findViewById(R.id.sharedPrefs);
myTextView.setText(prefs);

FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener()
{
    @Override
    public void onClick(View view)
    {
        Intent intent = new Intent (Intent.ACTION_VIEW , Uri.parse("mailto:" + "your_nutritionist"));
        intent.putExtra(Intent.EXTRA_SUBJECT, "calories & meals");
        intent.putExtra(Intent.EXTRA_TEXT, prefs);
        startActivity(intent);
    }
});

```

Εικόνα 24. Κώδικας για κλήση Gmail από το Nutriscope.

4.4 Τα αρχεία της Εφαρμογής Nutriscope

Η εφαρμογή Nutriscope περιέχει 5 είδη αρχείων, αρχεία Java, αρχεία xml, αρχεία JSON, ένα αρχείο txt και ένα αρχείο βιβλιοθήκης (jar), το οποίο περιέχει αρχεία java.

4.4.1 Αρχεία .java

Πρόκειται για τις κλάσεις του προγράμματος που καθορίζουν τις βασικές ενέργειες που επιτελούνται στην εφαρμογή.

- **MainActivity**: (or launcher activity) η δραστηριότητα που λειτουργεί ως το κύριο σημείο εισόδου στο περιβάλλον εργασίας χρήστη (UI) της εφαρμογής. Όταν ο χρήστης επιλέγει το εικονίδιο της εφαρμογής από την αρχική οθόνη, το σύστημα καλεί τη μέθοδο onCreate () για τη δραστηριότητα στην εφαρμογή που έχει δηλωθεί ως Main. Η main στην εφαρμογή καλεί την κλάση Authentication η οποία υλοποιεί τα βήματα πιστοποίησης του OAuth. Επίσης μετά την πιστοποίηση επιστρέφεται στην main, ένα αρχείο Json με την απάντηση από τον server του fitbit. Η Main περιέχει επίσης την μέθοδο parseResponse (String resp), η οποία επεξεργάζεται το αρχείο Json, κρατάει σε έναν ακέραιο τα δεδομένα καύσης θερμίδων και καλεί την κλάση Dashboard (μέσω intent).
- **FitbitApi**: πρόκειται για μια κλάση που κληρονομεί την κλάση DefaultApi10a και υλοποιεί κάποιες μεθόδους της.
- **Authentication**: η κλάση που ακολουθεί τα βήματα πιστοποίησης του OAuth. Είναι η κλάση που περιέχει το κλειδί και το μυστικό που έλαβε ο προγραμματιστής στη σελίδα του Fitbit, όπου δηλώνονται οι εφαρμογές.
- **Dashboard**: η κλάση που περιέχει το παράθυρο διαλόγου, την ημερομηνία, την μπάρα προόδου και το μενού. Επίσης, εντοπίζει το μονοπάτι του αρχείου με το εβδομαδιαίο πρόγραμμα διατροφής, το επεξεργάζεται και το εμφανίζει σε λίστα. Τέλος, αποθηκεύει τα επιλογές του χρήστη και καλεί την κλάση Results.
- **Meals**: κλάση set-get που τα αντικείμενά της καλούνται από την Dashboard.
- **Results**: εμφανίζει τα δεδομένα του χρήστη και δίνει την επιλογή της ηλεκτρονικής αλληλογραφίας με το διατροφολόγο.

4.4.2 Αρχεία .xml

Σε γλώσσα XML γράφεται το manifest της εφαρμογής και τα αρχεία που περιγράφουν την διεπαφή χρήστη (UI).

- **Manifest**: Κάθε εφαρμογή πρέπει να έχει ένα αρχείο AndroidManifest.xml (με ακριβώς αυτό το όνομα) στο κατάλογο root του. Το manifest παρουσιάζει

ουσιώδεις πληροφορίες σχετικά με την εφαρμογή του συστήματος Android, πληροφορίες που το σύστημα πρέπει να έχει για να μπορέσει να τρέξει οποιοδήποτε κώδικα της εφαρμογής.

Τα αρχεία xml που περιγράφουν την διεπαφή χρήστη βρίσκονται στον φάκελο resources της εφαρμογής και αφορούν τα layouts, τα χρώματα, τον τρόπο παρουσίασης των μενού, τα αλφαριθμητικά που χρησιμοποιήθηκαν.

4.4.3 Αρχεία .json

Τα αρχεία JSON περιέχουν τις πληροφορίες δραστηριότητας χρήστη, όμως για να διαβαστούν χρειάζονται ειδική επεξεργασία. Παράδειγμα τμήματος JSON αρχείου της εφαρμογής,

```
"activities":[  
  
],  
  
"goals":{  
  
"activeMinutes":40,  
  
"caloriesOut":1967,  
  
"distance":8.05,  
  
"steps":15000  
  
},  
  
"summary":{  
  
"activeScore":-1,  
  
"activityCalories":0,  
  
"caloriesBMR":1104,  
  
"caloriesOut":1104,
```


4.4.4 Αρχείο .txt

Πρόκειται για ένα απλό αρχείο κειμένου που περιέχει το εβδομαδιαίο πρόγραμμα διατροφής.

4.4.5 Αρχείο scribe 1.3.5 (.jar)

Στον φάκελο libs προστέθηκε το αρχείο scribe 1.3.5 της Maven το οποίο περιέχει όλες τις κλάσεις που απαιτούνται από το πρωτόκολλο OAuth.

4.4.6 Logo εφαρμογής



Εικόνα 25. Logo εφαρμογής Nutriscope

ΚΕΦΑΛΑΙΟ 5

Επίλογος

5.1 Σύνοψη

Στόχος της παρούσας διπλωματικής εργασίας ήταν η μελέτη και η ανάπτυξη μιας εφαρμογής σε λειτουργικό σύστημα Android, η οποία λαμβάνει δεδομένα φυσικής δραστηριότητας από το περικάρπιο Fitbit. Η εφαρμογή μετά από κατάλληλη επεξεργασία παρουσιάζει τα δεδομένα μαζί με το εβδομαδιαίο πρόγραμμα διατροφής στους χρήστες-πελάτες διατροφολόγων. Το εβδομαδιαίο πρόγραμμα διατροφής εισάγεται στην εφαρμογή κατόπιν επεξεργασίας ενός αρχείου κειμένου που είναι αποθηκευμένο στη συσκευή του χρήστη. Η εφαρμογή βοήθησε να αποδειχθεί ότι η χρήση gadget, όπως το fitbit, σε συνδυασμό με τη χρήση εφαρμογών για smartphones ή tablets μπορεί να βοηθήσει στην επίτευξη στόχων μέσω της πληθώρας πληροφοριών που κατέχει πλέον για τον εαυτό του ο κάθε άνθρωπος.

Σκοπός της υλοποίησης ήταν οι πελάτες διατροφολόγων να μπορούν να παρακολουθούν ανά πάσα στιγμή την πρόοδό τους σε καύση θερμίδων μέσω της εφαρμογής και ταυτόχρονα να είναι σε θέση να επιλέγουν ποια γεύματα κατανάλωσαν από το εβδομαδιαίο πρόγραμμα διατροφής. Τέλος, έχουν τη δυνατότητα να στείλουν τα αποτελέσματα της προσπάθειάς τους στο διατροφολόγο που τους παρακολουθεί, ώστε να δημιουργήσει το επόμενο πρόγραμμα διατροφής με τη χρήση των συγκεκριμένων πληροφοριών. Συνοψίζοντας όλα τα παραπάνω, καταλήγουμε στο συμπέρασμα ότι εφαρμογές που παρέχουν πληροφορίες στο χρήστη για τον εαυτό του και προσθέτουν επιπλέον στοιχεία ελέγχου στην καθημερινότητά του, μπορούν να συντελέσουν στην επίτευξη στόχων, στην αύξηση της απόδοσης μιας εργασίας και στην βελτίωση της ποιότητας ζωής.

5.2 Μελλοντικές επεκτάσεις

Η εφαρμογή αφορά τον τομέα της υγιεινής διατροφής και άσκησης, οπότε στο πλαίσιο αυτό υπάρχουν αρκετές επεκτάσεις που θα μπορούσαν να αυξήσουν την λειτουργικότητά της και να επιτελέσουν πληρέστερα το σκοπό της. Αφενός, θα μπορούσαν να εισαχθούν στοιχεία gamification, δηλαδή οι χρήστες να έχουν την δυνατότητα σύνδεσης της εφαρμογής με ένα κοινωνικό δίκτυο, όπου και θα μπορούν να δημοσιεύουν την πρόοδό τους και να κερδίζουν κάποιου είδους επιβράβευση όταν επιτύχουν ή ξεπεράσουν τον στόχο τους. Έτσι θα μπορούν μέσω του κοινωνικού δικτύου να παρατηρούν και την πρόοδο των φίλων τους και να αποκτούν μεγαλύτερο κίνητρο για να προσπαθήσουν.

Επιπρόσθετα, μια μελλοντική επέκταση που ενδεχομένως να αποτελέσει εξαιρετική βοήθεια στην προσπάθεια των πελατών διατροφολόγων είναι η προσθήκη ειδικού προγράμματος άσκησης που θα δημιουργούν καθηγητές φυσικής αγωγής και θα είναι προσαρμοσμένο στις ανάγκες και στο σωματότυπο του κάθε πελάτη. Τέλος, η προσθήκη ενός forum εντός της εφαρμογής, όπου οι χρήστες θα ανταλλάσσουν απόψεις ή θα συζητούν θέματα διατροφής, φυσικής δραστηριότητας και υγιεινού τρόπου ζωής πιθανότατα να έκανε την εφαρμογή πιο αποτελεσματική και ευχάριστη όσον αφορά την εμπειρία χρήστη.

ΚΕΦΑΛΑΙΟ 6

Βιβλιογραφία και πηγές

Βιβλιογραφία

- [1] Quantified Self https://en.wikipedia.org/wiki/Quantified_Self
- [2] Android (Operating System) [https://en.wikipedia.org/wiki/Android_\(operating_system\)](https://en.wikipedia.org/wiki/Android_(operating_system))
- [3] Fitbit flex <https://www.fitbit.com/eu/flex>
- [4] Fitbit algorithm https://help.fitbit.com/articles/en_US/Help_article/How-does-my-tracker-count-steps
- [5] Bluetooth low energy https://en.wikipedia.org/wiki/Bluetooth_low_energy
- [6] Accelerometer <https://en.wikipedia.org/wiki/Accelerometer>
- [7] IFIXIT Fitbit Flex Teardown <http://www.slideshare.net/diakonia7/fitbit-flex-teardown>
- [8] Open API https://en.wikipedia.org/wiki/Open_API
- [9] Fitbit API <https://dev.fitbit.com/eu>
- [10] Oauth protocol <http://oauth.net/core/1.0a/>
- [11] List of Alternative Android App Development IDEs <http://tekeye.biz/2014/list-of-android-app-development-ides>
- [12] Scribe 1.3.5 <http://mvnrepository.com/artifact/org.scribe/scribe/1.3.5>
- [13] Json Formatter <https://jsonformatter.curiousconcept.com/>
- [14] <http://developer.android.com/>
- [15] Stack overflow <http://stackoverflow.com/questions/18372450/how-to-set-arrayadapter-to-listview-android>
- [16] Fitbit system overview <https://courses.csail.mit.edu/6.857/2014/files/17-cyrbritt-webbhorn-specter-dmiao-hacking-fitbit.pdf>