



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

*Σχεδίαση και Ανάπτυξη Διαδραστικού συστήματος
για ξενάγηση σε αρχαιολογικούς χώρους
με χρήση του πρωτοκόλλου Bluetooth 5.0*

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Μάριος - Κωνσταντίνος Γ. Μακεδών

Επιβλέπων : Πέτρος - Παύλος Σωτηριάδης
Αναπλ. Καθηγητής Ε.Μ.Π.

Εργαστήριο Ηλεκτρονικής
Αθήνα, Απρίλιος 2019



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ
ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

*Σχεδίαση και Ανάπτυξη Διαδραστικού συστήματος
για ξενάγηση σε αρχαιολογικούς χώρους
με χρήση του πρωτοκόλλου Bluetooth 5.0*

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Μάριος - Κωνσταντίνος Γ. Μακεδών

Επιβλέπων : Πέτρος - Παύλος Σωτηριάδης
Αναπλ. Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 16^η Απριλίου 2019

.....
Πάυλος-Πέτρος Σωτηριάδης
Αναπλ. Καθηγητής Ε.Μ.Π.

.....
Συμεών Παπαβασιλείου
Καθηγητής Ε.Μ.Π.

.....
Δημήτριος Σούντρης
Αναπλ. Καθηγητής Ε.Μ.Π.

Εργαστήριο Ηλεκτρονικής
Αθήνα, Απρίλιος 2019

.....

Μάριος - Κωνσταντίνος Γ. Μακεδών

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Μάριος - Κωνσταντίνος Γ. Μακεδών, 2019. 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.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Σκοπός της παρούσας διπλωματικής, εργασίας είναι η σχεδίαση και η ανάπτυξη διαδραστικού συστήματος, για ξενάγηση σε αρχαιολογικούς χώρους, με χρήση του πρωτοκόλλου Bluetooth 5.0. Για την επίτευξη του στόχου αυτού, έγινε, αρχικά, μελέτη του πρωτοκόλλου Bluetooth, καθώς και του Bluetooth Low Energy, η οποία αποτελεί αναβάθμιση του πρώτου. Η διπλωματική εργασία αποτελείται από δύο κύρια μέρη. Το πρώτο μέρος σχετίζεται με την κατασκευή και τον προγραμματισμό αναπτυξιακής πλακέτας, η οποία χρησιμοποιεί το πρωτόκολλο Bluetooth, ώστε αυτή να είναι ορατή από την εφαρμογή Android. Το δεύτερο μέρος αποτελείται από την ανάπτυξη εφαρμογής σε λειτουργικό σύστημα Android. Χρησιμοποιώντας την εφαρμογή αυτή, ο χρήστης θα μπορεί να πληροφορηθεί για τα πλησιέστερα, προς αυτόν, εκθέματα του αρχαιολογικού χώρου, καθώς και να βρει σχετικές πληροφορίες.

Λέξεις κλειδιά:

Bluetooth, Bluetooth Low Energy, Αναπτυξιακή πλακέτα, μικροελεγκτής DA14585, Dialog Semiconductor, εφαρμογή, Android, εγγύτητα, απόσταση, αρχαιολογικός χώρος

Abstract

The purpose of this diploma thesis is the interactive systems' design and development for guide tour archaeological sites by using the Bluetooth 5.0 protocol. To achieve this goal, initially, the protocol Bluetooth was studied, as well as the Bluetooth Low Energy, which is an upgrade of the first one. The diploma thesis consists of two main parts. The first part relates to the fabrication and programming of the development board, which uses the Bluetooth protocol, so as to be visible to the Android application. The second part refers to the application's development on Android operating system. By using this application, the user can be informed about the nearest exhibits of the archaeological site and find relevant information. .

Key words:

Bluetooth, Bluetooth Low Energy, Development board, microcontroller DA14585, Dialog Semiconductor, εφαρμογή, Android, proximity, distance, archaeological site

Ευχαριστίες

Η παρούσα διπλωματική εργασία αποτελεί το επιστέγασμα μιας πενταετούς προσπάθειας, που είχε ως απώτερο στόχο την απόκτηση θεμελιωδών γνώσεων, σχετικών με το αντικείμενο του Ηλεκτρολόγου Μηχανικού και Μηχανικού Υπολογιστών, η οποία εκπονήθηκε στο Εργαστήριο Ηλεκτρονικής του Εθνικού Μετσόβιου Πολυτεχνείου, υπό την επίβλεψη του Καθηγητή Πέτρου - Παύλου Σωτηριάδη.

Θα ήθελα σε αυτό το σημείο να εκφράσω τον σεβασμό και την εκτίμησή μου, στον επιβλέποντα Καθηγητή μου, κ. Πέτρο - Παύλο Σωτηριάδη, Διευθυντή του Εργαστηρίου Ηλεκτρονικής. Τον ευχαριστώ τόσο για την εμπιστοσύνη που μου έδειξε, αναθέτοντας μου ένα τόσο ενδιαφέρον θέμα διπλωματικής εργασίας, όσο και για το σύνολο των γνώσεων που μου μετέδωσε κατά τη φοίτησή μου στο Εθνικό Μετσόβιο Πολυτεχνείο.

Επιπλέον, θα ήθελα να ευχαριστήσω τον υποψήφιο διδάκτορα Κωνσταντίνο Παπαφώτη, τόσο για την πολύτιμη καθοδήγησή του, όσο και για τις γνώσεις που μου μετέφερε. Θα ήθελα, επίσης, να ευχαριστήσω τους διπλωματούχους Αθανάσιο Ανδριόπουλο και Χρήστο Κουτσοουραδή για τις εποικοδομητικές συμβουλές τους, οι οποίες συνετέλεσαν στην ολοκλήρωση αυτού του πονήματος.

Ακόμη, θα ήθελα να ευχαριστήσω ιδιαίτερος τους εκπροσώπους της εταιρείας Dialog Semiconductor, για την παροχή της αναπτυξιακής πλακέτας, καθώς και των λοιπών υλικών που απαιτήθηκαν για την κατασκευή της.

Θέλω να αφιερώσω την διπλωματική μου εργασία στην οικογένεια μου, χωρίς την ηθική και υλική υποστήριξη της οποίας, ελάχιστα θα είχαν επιτευχθεί.

Τέλος, ευχαριστώ τους εκλεκτούς φίλους μου, για τις ωραίες, αλλά και τις δύσκολες στιγμές που μοιραστήκαμε κατά τη διάρκεια των φοιτητικών μας χρόνων.

Περιεχόμενα

Ευχαριστίες	9
Περιεχόμενα	11
Κεφάλαιο 1 - Εισαγωγή	13
Κεφάλαιο 2 - Bluetooth Low Energy	15
Ενότητα 2.1 - Εισαγωγικά Στοιχεία για το Bluetooth	16
Ενότητα 2.2 - Περιγραφή της στοίβας πρωτοκόλλων Bluetooth	20
Υποενότητα 2.2.1 - Το Φυσικό Επίπεδο	22
Υποενότητα 2.2.2 - Το Επίπεδο Ζεύξης Δεδομένων	23
Υποενότητα 2.2.3 - Διεπαφή Host Controller	26
Υποενότητα 2.2.4 - Logical Link Control - Adaptation Protocol..	27
Υποενότητα 2.2.5 - Security Manager Protocol	28
Υποενότητα 2.2.6 - Generic Access Profile	29
Υποενότητα 2.2.7 - Attribute Protocol	29
Υποενότητα 2.2.8 - Generic Attribute Profile	30
Κεφάλαιο 3 - Development Board DA14585	31
Ενότητα 3.1 - Battery Charger BQ24090DSQR	32
Ενότητα 3.2 - Buck-Boost Converter TPS63001DRCR	33
Ενότητα 3.3 - Microcontroller DA14585	34
Κεφάλαιο 4 - Ανάπτυξη Εφαρμογής σε Android	43
Ενότητα 4.1 - Εισαγωγικά Στοιχεία για το λειτουργικό Android	44
Ενότητα 4.2 - Αρχεία XML και αρχεία Resources	46
Ενότητα 4.3 - Λειτουργία της Εφαρμογής	48
Υποενότητα 3.3.1 - Java Activities	49
Υποενότητα 3.3.2 - Δημιουργία Λίστας Αγαπημένων	54
Υποενότητα 3.3.3 - Εκτίμηση Απόστασης Κόμβων	54
Κεφάλαιο 5 - Επίλογος	60
Ενότητα 5.1 - Συμπεράσματα	61
Ενότητα 5.2 - Μελλοντικές Κατευθύνσεις	61
Παράρτημα Α	62
Βιβλιογραφία	63

Κεφάλαιο 1

Εισαγωγή

Σε μία εποχή, όπου η ραγδαία τεχνολογική ανάπτυξη αποτελεί μέρος της καθημερινότητας μας, είναι, ορισμένες φορές, αναγκαία η προσαρμογή των υπηρεσιών στις ανάγκες του πληθυσμού, αλλά και στις σύγχρονες τεχνολογικές εφευρέσεις. Η πλειονότητα του πληθυσμού χρησιμοποιεί, πλέον, το κινητό τηλέφωνο, το τάμπλετ, τον υπολογιστή, πολλές ώρες καθημερινά. Μάλιστα, ο πιο συνήθης λόγος χρήσης τους είναι η πλοήγηση στο Διαδίκτυο, και δη η επίσκεψη σε μέσα κοινωνικής δικτύωσης. Είναι εμφανές, πως πλέον, ήδη πολλές λειτουργίες της καθημερινότητας που απαιτούσαν δια ζώσης επικοινωνία, έχουν αντικατασταθεί από αυτοματοποιημένες εφαρμογές των κινητών τηλεφώνων και των υπολογιστών.

Σε μία χώρα, με Ιστορία, πολλών χιλιάδων χρόνων, πολύς κόσμος φαίνεται να λησμονεί τα ιστορικά κατορθώματα (επιστήμες, τέχνες, γράμματα κ.α.) των προγόνων του. Σκοπός της παρούσας διπλωματικής εργασίας, είναι η εύρεση τρόπων μαγνητισμού του ενδιαφέροντος των νέων, αλλά και μεγαλύτερων σε ηλικία, ανθρώπων σχετικά με την πολιτιστική κληρονομιά, την ιστορία και τις παραδόσεις.

Το συνολικό αποτέλεσμα αποτελεί ένα ολοκληρωμένο εγχείρημα για μια διαδραστική περιήγηση σε αρχαιολογικούς χώρους. Αποτελείται από δύο τμήματα. Το πρώτο τμήμα αποτελεί την σχεδίαση σε PCB αναπτυξιακής πλακέτας, η οποία, με χρήση φωτοβολταϊκού πάνελ, και λόγω της πολύ χαμηλής κατανάλωσης ενέργειάς της, θα είναι αυτόνομη, ως προς την παροχή ισχύος. Στην πλακέτα αυτή, έχει τοποθετηθεί, μεταξύ άλλων στοιχείων, ο μικροελεγκτής DA14585, ο οποίος υποστηρίζει το πρωτόκολλο Bluetooth Low Energy 5.0. Η συσκευή, που προκύπτει, αποτελεί έναν beacon (φάρος) για τις κινητές συσκευές των τουριστών. Το δεύτερο τμήμα, της διπλωματικής εργασίας, αποτελεί η ανάπτυξη εφαρμογής σε λειτουργικό σύστημα Android. Έχοντας εγκατεστημένη, την εφαρμογή αυτή, στο Smart Phone, ένας χρήστης θα μπορεί να σαρώνει την περιοχή κοντά του, εντοπίζοντας όλες τις συσκευές που είναι τοποθετημένες στα εκθέματα του αρχαιολογικού χώρου. Οι συσκευές αυτές, θα εμφανίζονται στην οθόνη του με σειρά απόστασης από αυτόν (η πλησιέστερη πάνω αριστερά κ.ο.κ.). Εν συνεχεία, αυτός θα έχει τη δυνατότητα να επιλέξει τη συσκευή - έκθεμα που επιθυμεί και να ενημερωθεί, για αυτό, με σχετικό οπτικοακουστικό υλικό. Παράλληλα, παρέχεται η επιλογή της λειτουργίας: *Προσθήκη στα Αγαπημένα*, με την οποία, ο χρήστης θα μπορεί να έχει πρόσβαση σε αυτές τις πληροφορίες και σε μελλοντικό χρόνο, αφού έχει απομακρυνθεί από τον αρχαιολογικό χώρο και οι συσκευές δεν είναι εντός της εμβέλειας του κινητού τηλεφώνου του.

Το συγγραφικό κομμάτι της διπλωματικής εργασίας αποτελείται από 5 Κεφάλαια:

- Κεφάλαιο 1 - Εισαγωγή: Στο κεφάλαιο αυτό γίνεται συνοπτική παρουσίαση της διπλωματικής εργασίας
- Κεφάλαιο 2 - Bluetooth Low Energy: Στο κεφάλαιο αυτό παρουσιάζονται στοιχεία σχετικά με το πρωτόκολλο Bluetooth Classic και Bluetooth Low Energy
- Κεφάλαιο 3 - Development Board DA14585: Στο κεφάλαιο αυτό γίνεται λεπτομερής περιγραφή της διαδικασίας σχεδίασης, κατασκευής και προγραμματισμού της αναπτυξιακής πλακέτας
- Κεφάλαιο 4 - Ανάπτυξης Εφαρμογής σε Android: Στο κεφάλαιο αυτό, πραγματοποιείται διεξοδική ανάλυση της διαδικασίας ανάπτυξης της εφαρμογής
- Κεφάλαιο 5 - Επίλογος: Στο κεφάλαιο αυτό περιγράφονται τα συμπεράσματα που προέκυψαν, καθώς και πιθανές μελλοντικές κατευθύνσεις

Κεφάλαιο 2

Bluetooth Low Energy

2.1 Εισαγωγικά Στοιχεία για το Bluetooth

Το Bluetooth αποτελεί μία ασύρματη τηλεπικοινωνιακή τεχνολογία που αποσκοπεί στη μεταφορά δεδομένων, κυρίως σε μικρές αποστάσεις. Έτσι, ένας χρήστης μπορεί να συνδέσει μια σειρά από σταθερές, αλλά και, κινητές συσκευές, ώστε να δημιουργήσει ένα Personal Area Network (PAN). Παραδείγματα τέτοιων συσκευών, αποτελούν οι συσκευές PDA (προσωπικός ψηφιακός οδηγός ή υπολογιστής παλάμης), τα κινητά τηλέφωνα, οι φορητοί και προσωπικοί υπολογιστές, οι εκτυπωτές, καθώς και οι ψηφιακές φωτογραφικές και κάμερες, χάρη σε μια ασφαλή, οικονομική και παγκοσμίως διαθέσιμη, χωρίς ειδική άδεια ραδιοσυχνότητας μικρής εμβέλειας. Ωστόσο, η σύνδεση με περιφερειακό ακουστικό αποτελεί το βασικότερο λόγο που καθιστά δημοφιλή τη συγκεκριμένη τεχνολογία. Παρόλο που η μεταφορά δεδομένων μεταξύ κινητών δεν είναι πλέον διαθέσιμη, με το Bluetooth Low Energy, μπορούν να χρησιμοποιηθούν περισσότερα προϊόντα για περισσότερες λειτουργίες.



Σχήμα 1: Λογότυπο Bluetooth

Ως τα τέλη της δεκαετίας του 1990, δεν υπήρχε κάποιο ευρέως αποδεκτό πρότυπο WPAN, ούτε ανάλογες εμπορικές εφαρμογές. Τότε, η Ericsson ανέπτυξε μια τεχνολογία, η οποία επέτρεπε τον σχηματισμό τοπικών δικτύων, μικρής εμβέλειας, με σκοπό την ασύρματη και ad hoc δικτύωση ετερογενών φορητών συσκευών. Το πρότυπο που προέκυψε υιοθετήθηκε στη συνέχεια από την IEEE ως το πρότυπο 802.15 για WPAN. Οι σχεδιαστές του κλήθηκαν να επιλέξουν το όνομα με το οποίο αυτή η τεχνολογία θα γινόταν αργότερα γνωστή σε όλο τον κόσμο. Οι Σουηδοί developers του 802.15 ήταν βέβαιοι ότι το νέο πρότυπο θα επικρατούσε και θα έφερνε ακόμη πιο κοντά τους ανθρώπους και τις συσκευές τους. Εμπνεύστηκαν από τη δράση του Δανού Βασιλιά Χάραλντ , ο οποίος έζησε στα τέλη του 10ου αιώνα μ.Χ. και κατέλαβε με τα στρατεύματά του πολλές χώρες, ενώ λέγεται ότι κατάφερε να ενώσει τη Δανία με τη Νορβηγία.

Το 1999, κυκλοφορεί η πρώτη έκδοση του Bluetooth 1.0 και ένα χρόνο αργότερα κινητά τηλέφωνα για ασύρματη μετάδοση πληροφορίας και Bluetooth ακουστικά, που χρησιμοποιούν τη συγκεκριμένη τεχνολογία. Το 2004, κυκλοφορεί η δεύτερη

έκδοση του Bluetooth 2.0, στην οποία έχει προστεθεί το Enhanced Data Rate, το οποίο εξασφαλίζει υψηλότερη ταχύτητα από την προηγούμενη έκδοση. Λίγα χρόνια αργότερα κυκλοφορεί η έκδοση 2.1, η οποία αυξάνει κυρίως το επίπεδο ασφαλείας. Κατά τη χρονική αυτή περίοδο, υπάρχουν ήδη 2 δισεκατομμύρια προϊόντα που υποστηρίζουν το συγκεκριμένο πρωτόκολλο. Το 2009, έχουμε την τρίτη έκδοση 3.0, η οποία χρησιμοποιώντας ένα διαφορετικό μέσο μεταφοράς στο χαμηλότερο επίπεδο, επιτυγχάνει μεγαλύτερες ταχύτητας μεταφοράς δεδομένων.

Η πρώτη εμφάνιση του Bluetooth Low Energy πραγματοποιήθηκε το 2010 με την τέταρτη έκδοση 4.0, η οποία προσέφερε χαμηλή κατανάλωση και χαμηλού κόστους ασύρματη τεχνολογία, με την εμπορική ονομασία Bluetooth Smart. Η υποστήριξη του Bluetooth 4.0 από το iPhone 4S και λίγα χρόνια αργότερα του Bluetooth 4.3 από το λειτουργικό σύστημα Android λειτούργησε επικουρικά στην ανάπτυξη και τη διάδοση του Bluetooth Low Energy. Με την κυκλοφορία των Windows 8, η Microsoft παρέχει, επίσης, υποστήριξη για τη νέα τεχνολογία, ενώ πολλοί σταθεροί και φορητοί υπολογιστές πωλούνται με ενσωματωμένο Bluetooth 4.0 hardware.

Τον Δεκέμβριο του 2013, κυκλοφόρησε η επέκταση 4.1, η οποία αποτελεί αναβάθμιση σε επίπεδο software και όχι σε επίπεδο hardware, έχει σκοπό την βελτιστοποίηση της εμπειρίας του χρήστη και περιλαμβάνει ακόμη υψηλότερες ταχύτητες μετάδοσης της πληροφορίας. Ένα χρόνο αργότερα, κυκλοφορεί η έκδοση 4.2, που εισαγάγει features για το Διαδίκτυο των πραγμάτων (Internet of Things - IoT). Οι κυριότεροι τομείς αναβάθμισης είναι η ασφαλής χαμηλούς κατανάλωσης σύνδεση και η δυνατότητα χρήσης του IPv6 από Bluetooth Smart συσκευές. Το τελευταίο αποτελεί λύση για IoT εφαρμογές που είχαν περιορισμούς στην κατανάλωση και το μέγεθος.

Τον Ιούνιο του 2016, παρουσιάστηκε το πρωτόκολλο Bluetooth 5.0, το οποίο προσανατολίζεται στις ανάγκες της τεχνολογίας του Διαδικτύου Πραγμάτων. Ήδη πολλές καινούριες συσκευές περιλαμβάνουν αυτήν την τεχνολογία, όπως το Samsung Galaxy S8 και τα iPhone 8, 8+ και X. Το Bluetooth 5 προσφέρει διπλάσια ταχύτητα (2Mbit/s burst) εις βάρος όμως της εμβέλειας, υπερτετραπλάσια εμβέλεια εις βάρος του ρυθμού μετάδοσης δεδομένων και οκταπλάσια αύξηση του ρυθμού μετάδοσης δεδομένων, η οποία επιτυγχάνεται με αύξηση του μήκους των πακέτων. Το τελευταίο θα μπορούσε να είναι ιδιαίτερος σημαντικό για τις Internet of Things συσκευές που χρησιμοποιούνται μέσα σε ένα σπίτι, όπου υπάρχει μεγάλος αριθμός κόμβων. Το Bluetooth 5 προσφέρει λειτουργικότητα για ασύρματες υπηρεσίες, με κύρια παραδείγματα τον προσανατολισμό και την πλοήγηση σε μικρό ή εσωτερικό χώρο.

Το πρότυπο Bluetooth Low Energy σχεδιάστηκε με γνώμονα τη μείωση της κατανάλωσης ενέργειας, αλλά και του κόστους κατασκευής. Σκοπός των σχεδιαστών ήταν η χρήση του σε οικονομικά προϊόντα μαζικής παραγωγής. Σε αντίθεση με άλλα πρωτόκολλα ασύρματης μετάδοσης πληροφορίας, στο BLE ορίστηκε ολόκληρη η στοίβα πρωτοκόλλων που πρέπει να υλοποιείται από τις συμβατές συσκευές. Με τον τρόπο αυτό μπορούν να βελτιστοποιηθούν όλα τα στρώματα του πρωτοκόλλου. Επιπλέον, το Bluetooth Low Energy είναι πολύ πιο απλό από το Bluetooth Classic. Αυτό συνεπάγεται εκτέλεση λιγότερων εντολών από το μικροεπεξεργαστή της

συσκευής, άρα και χαμηλότερο κόστος, δεδομένου ότι απαιτείται λιγότερη μνήμη. Η μεγαλύτερη κατανάλωση ενέργειας παρατηρείται όταν η συσκευή κάνει scanning, δηλαδή ελέγχει το φάσμα για απεσταλμένα πακέτα. Το πρόβλημα αυτό, επιλύθηκε με το να παραμένει κλειστό το ράδιο για όσο χρόνο δεν χρειάζεται να κάνει τον παραπάνω έλεγχο. Όλα τα παραπάνω συνέβαλαν στο να μπορεί μια συσκευή, έχοντας μόνο μια τυπική μικρή μπαταρία του εμπορίου να λειτουργεί για πολλούς μήνες ή και χρόνια.

Όπως και σε κάθε τεχνολογία, έτσι και στην περίπτωση του Bluetooth Low Energy, έγιναν ορισμένες υποχωρήσεις. Ο κυριότερος περιορισμός του είναι στο data throughput. Η μέγιστη ταχύτητα μεταφοράς της πληροφορίας είναι σχετικά μικρή, με αποτέλεσμα, το Bluetooth Low Energy να μην είναι χρηστικό για εφαρμογές που απαιτούν αποστολή μεγάλης ποσότητας δεδομένων σε σύντομο χρονικό διάστημα. Έναν ακόμη περιορισμό αποτελεί η απόσταση. Παρόλο που βελτιώθηκε σημαντικά σε σχέση με το Bluetooth Classic, το BLE μπορεί να λειτουργεί ικανοποιητικά σε εμβέλεια 10 μέτρων σε κλειστό χώρο. Βέβαια, δεν πρέπει να λησμονηθεί ότι ο πρωτεύων στόχος είναι η μείωση στην κατανάλωση ενέργειας.



Σχήμα 2: Συσκευές που χρησιμοποιούν την τεχνολογία Bluetooth (wearables)

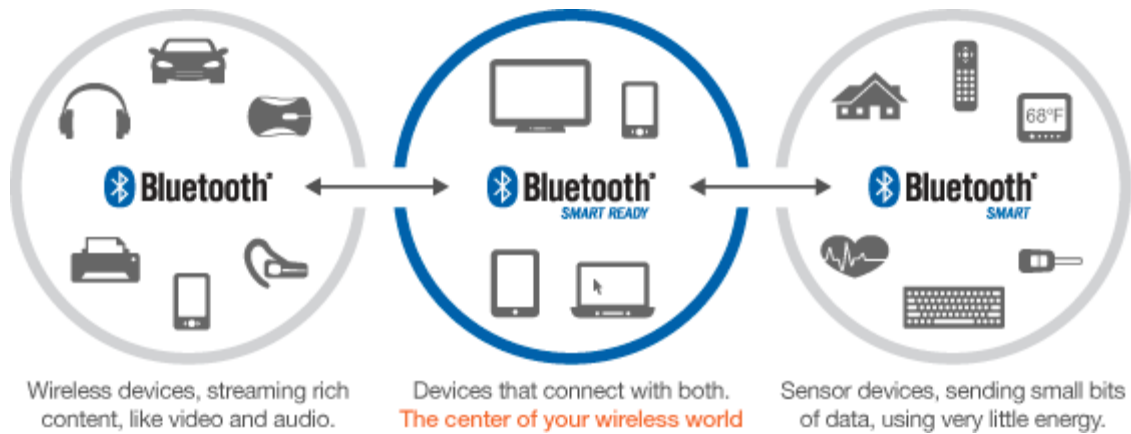
Οι προοπτικές για την εξέλιξη του Bluetooth είναι πολύ θετικές και οι ειδικοί εκτιμούν ότι το πλήθος των συσκευών που υποστηρίζουν το πρότυπο αναμένεται να αυξηθεί τις επόμενες δεκαετίες. Η κυκλοφορία του Bluetooth Smart συμπίπτει με την ραγδαία τεχνολογική πρόοδο και την ανάπτυξη των smartphones και tablets, ενώ χάρη σε αυτή την τεχνολογία σχεδιάζονται νέες συσκευές, όπως τα wearables (Σχήμα 2), συσκευές που μπορεί να "φοράει" ένας χρήστης. Ουσιαστικά, πρόκειται για μικρού μεγέθους συσκευές, οι οποίες, λόγω χαμηλής κατανάλωσης, μπορούν να λειτουργούν με μια μπαταρία για χρόνια. Οι συσκευές αυτές δεν παρουσιάζουν ιδιαίτερη χρησιμότητα, καθώς συνήθως δεν έχουν user interface. Ωστόσο, σε συνδυασμό με μια εφαρμογή ενός smartphone, οι παραπάνω συσκευές αποκτούν χρησιμότητα που δε θα ήταν αλλιώς εφικτή.

Οι περισσότερες συσκευές τύπου wearable σχετίζονται κυρίως με την μέτρηση της δραστηριότητας του χρήστη (καρδιακός ρυθμός, υπολογισμός απόστασης που

διένυσε κ.λ.π.), οι οποίες βρίσκουν χρήση τόσο στις αθλητικές του ενασχολήσεις, όσο και στον τομέα της υγείας (π.χ. θερμόμετρο, πιεσόμετρο). Η δυνατότητα αποστολής εντολών από και προς τις συσκευές ανοίγει νέους ορίζοντες σε τομείς όπως το έξυπνο σπίτι (smart home). Ο χρήστης θα μπορεί από κάποιο κεντρικό σημείο, το οποίο μπορεί να είναι ακόμα και το wearable, να ελέγχει διάφορες λειτουργίες του σπιτιού, όπως η ενεργοποίηση του θερμοσίφωνα, το κλείσιμο της τηλεόρασης, η αλλαγή στη θερμοκρασία του φούρνου ή του ψυγείου.

Μια άλλη περίπτωση χρήσης, που αναμένεται να εξελιχθεί σε ιδιαίτερα δημοφιλή στο μέλλον και η οποία παρέχει απεριόριστες δυνατότητες στους developers και τους χρήστες, είναι ο έλεγχος εγγύτητας (proximity). Για παράδειγμα, ένα Bluetooth Smart μπρελόκ μπορεί να χρησιμοποιηθεί, ώστε ο χρήστης να εντοπίζει τα κλειδιά του ή ακόμα και για να μην ξεχάσει το τηλέφωνό του, το οποίο θα τον ειδοποιήσει όταν απομακρυνθεί από αυτό. Μια άλλη περίπτωση, που αποτελεί σε ένα βαθμό και το αντικείμενο της παρούσας διπλωματικής εργασίας, είναι η χρησιμοποίηση μεγάλου πλήθους συσκευών Bluetooth Smart και η τοποθέτησή τους σε μικρό, σχετικά, χώρο. Οι συσκευές αυτές έχουν ως μόνη λειτουργία να μεταδίδουν, ανά τακτά χρονικά διαστήματα, μία μικρή ποσότητα δεδομένων, παρέχοντας τη δυνατότητα σε μια κατάλληλα σχεδιασμένη smartphone εφαρμογή να τις αναγνωρίσει. Με αυτό τον τρόπο, μπορούμε να έχουμε τεχνικές πλοήγησης και εντοπισμού θέσης σε εσωτερικό χώρο, καθώς και παρουσίαση διαφορετικού περιεχομένου στον χρήστη, ανάλογα με την απόσταση του από συγκεκριμένες συσκευές. Παραδείγματος χάριν, ένας χρήστης που εισέρχεται σε ένα μουσείο ή σε έναν αρχαιολογικό χώρο μπορεί, αφού κατεβάσει και εγκαταστήσει την εφαρμογή που παρέχεται στο App Store της πλατφόρμας του κινητού του, να περιπλανηθεί και να ξεναγηθεί με γραπτές πληροφορίες αλλά και με οπτικοακουστικό υλικό, που θα εμφανίζεται κάθε φορά που θα πλησιάζει ένα έκθεμα. Κοντά σε κάθε έκθεμα θα έχει τοποθετηθεί μια συσκευή, την οποία η εφαρμογή θα αναγνωρίζει. Επιπλέον, με την είσοδο του χρήστη στο μουσείο ή τον αρχαιολογικό χώρο, η εφαρμογή θα επιβεβαιώνει την κράτηση που είχε κάνει νωρίτερα, χωρίς αυτός να χρειάζεται να περιμένει στην ουρά των εισιτηρίων.

Στο BLE πρωτόκολλο, μια συσκευή, ανάλογα με το αν μπορεί να υποστηρίξει το Bluetooth Classic ή το BLE, μπορεί να ανήκει σε δύο κατηγορίες: τις dual-mode συσκευές που μπορούν να επικοινωνούν με οποιοδήποτε από τα δυο πρωτόκολλα, και τις single-mode που μπορούν να επικοινωνούν αποκλειστικά με BLE συσκευές. Στην πρώτη κατηγορία ανήκουν προϊόντα που μπορούν να λειτουργήσουν ως κεντρικοί σταθμοί σε ένα δίκτυο, όπως ο υπολογιστής, το κινητό τηλέφωνο και το tablet. Στη δεύτερη κατηγορία ανήκουν προϊόντα, που έχουν σχεδιασθεί για να καταναλώνουν χαμηλή ενέργεια και να έχουν ανάγκη για μικρή σε όγκο μεταφορά δεδομένων, όπως smartwatches, πιεσόμετρα, θερμόμετρα. Στην τρίτη κατηγορία, ανήκουν οι συσκευές που δεν μπορούν να επικοινωνήσουν με συσκευές BLE, και αυτό γιατί σχεδιάσθηκαν πριν κυκλοφορήσει η συγκεκριμένη τεχνολογία, όπως τα Bluetooth ακουστικά.



Σχήμα 3: Παραδείγματα Bluetooth συσκευών

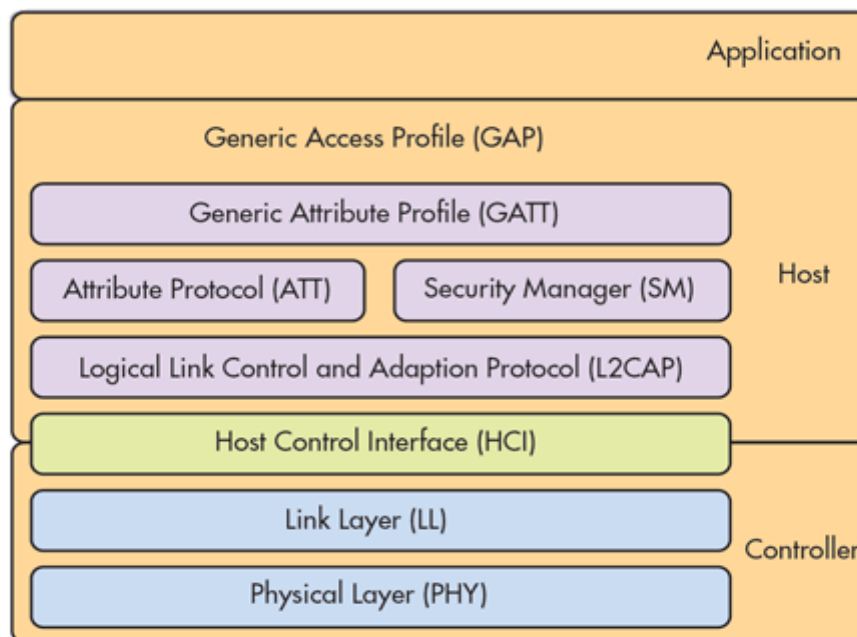
Οι κυριότερες διαφορές ανάμεσα στο Bluetooth Classic και στο Bluetooth Low Energy παρουσιάζονται επιγραμματικά στον Πίνακα 1:

Πίνακας 1: Διαφορές Bluetooth Classic - Bluetooth Low Energy

Χαρακτηριστικό	Bluetooth Classic	Bluetooth Low Energy
Nodes	7	Απεριόριστος αριθμός
Modulation Index	0.35	0.5 (ανοχή στο θόρυβο)
Max Output Power	20dBm	10 dBm
Range (typical)	10-30 μέτρα	50-100 μέτρα
Max Packet Size	2875 μ s (1021 bytes)	328 μ s (27 bytes)
Max Data Rate	2178.1 kbps	305 kbps
CRC Strength	16 bit	24 bit
Authentication	μία φορά	ανά πακέτο

2.2 Περιγραφή της στοίβας πρωτοκόλλων BLE

Στο Σχήμα 4, απεικονίζεται η αρχιτεκτονική και τα στοιχεία που αποτελούν τη στοίβα πρωτοκόλλων του Bluetooth Low Energy.



Σχήμα 4: Αρχιτεκτονική στοίβας πρωτοκόλλων Bluetooth Low Energy

Στο κατώτερο τμήμα της στοίβας πρωτοκόλλων βρίσκεται ο Controller. Το χαμηλότερο τμήμα του Controller είναι το φυσικό επίπεδο (physical layer), το οποίο είναι υπεύθυνο για την αποστολή και τη λήψη πακέτων. Πάνω από αυτό βρίσκεται το επίπεδο ζεύξης δεδομένων (link layer), στο οποίο γίνεται έλεγχος ορθής λειτουργίας του φυσικού επιπέδου.

Όπως φαίνεται στο Σχήμα 4, όλα τα υπόλοιπα στοιχεία του συστήματος αποτελούν τον Host. Το επίπεδο που συνδέει τον Host και τον Controller ονομάζεται Host Control Interface (HCI), το οποίο περιέχει ένα πλήθος εντολών και διαδικασιών, με σκοπό να εκτελούνται οι διάφορες λειτουργίες του πρωτοκόλλου. Πέρα από αυτό, υπάρχουν και ορισμένα transport layers, που χρησιμοποιούνται για τη σύνδεση των Host και Controller. Τέτοια layers είναι το UART και το USB. Χάρη στο HCI έχουν δημιουργηθεί αρχιτεκτονικές όπου ο Host βρίσκεται σε διαφορετικό chip set από τον Controller, οι οποίες βρίσκουν χρήση σε dual-mode (Bluetooth Smart Ready) συσκευές, αφού σε αυτές υπάρχει ήδη μια CPU μεγάλης υπολογιστικής ισχύς και ο host θα εξυπηρετείται από τον κεντρικό επεξεργαστή.

Πάνω από την διεπαφή Host και Controller, βρίσκεται το Logical Link Control and Adaptation Protocol (L2CAP). Το L2CAP παρέχει υπηρεσίες πολυπλεξίας στα υψηλότερα τμήματα της στοίβας πρωτοκόλλων BLE. Ο Security Manager διαχειρίζεται τις ενέργειες για την παραγωγή και την ανταλλαγή των κλειδιών μεταξύ των συσκευών. Είναι υπεύθυνος για την ασφάλεια της σύνδεσης, δηλαδή τη διασφάλιση απορρήτου, την πιστοποίηση της αυθεντικότητας, την κρυπτογράφηση, καθώς και για το pairing και το bonding. Το Generic Access Profile (GAP) ορίζει όλους τους τρόπους λειτουργίας και τις διαδικασίες, που χρησιμοποιούν οι συσκευές με σκοπό να εντοπίσουν η μια την άλλη, να επιτύχουν σύνδεση και να ανταλλάξουν πληροφορία.

Τα μοναδικά τμήματα της παραπάνω στοίβας που ένας developer έρχεται σε

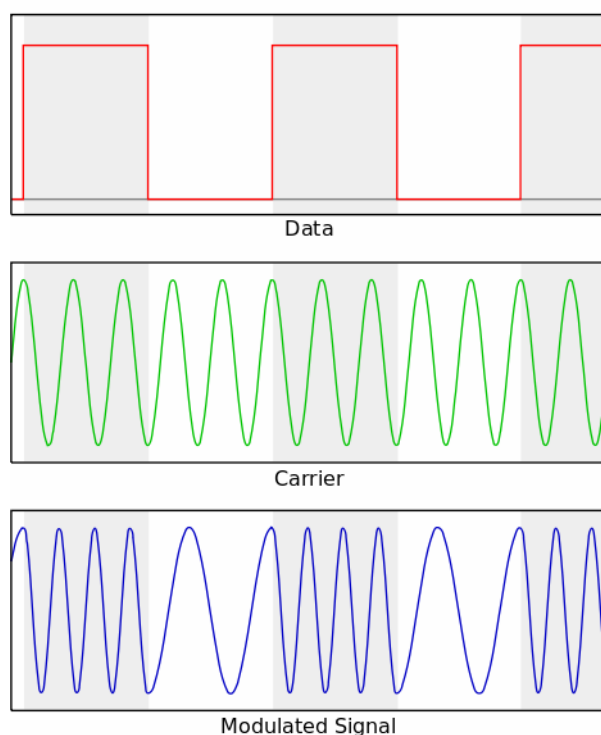
επαφή και μπορεί να αλληλεπιδράσει είναι το Attribute Protocol (ATT) και το Generic Attribute Profile (GATT). Σε αυτά ο developer μπορεί να σχεδιάσει και να υλοποιήσει όλες τις εφαρμογές, με τις οποίες έρχεται σε άμεση επαφή ο χρήστης.

Στη συνέχεια παρουσιάζονται αναλυτικά τα τμήματα της στοίβας πρωτοκόλλου του Bluetooth Low Energy.

2.2.1 Το Φυσικό Επίπεδο

Το Bluetooth Low Energy λειτουργεί, όπως και ο πρόγονός του, το Bluetooth Classic, σε ένα εύρος συχνοτήτων, το οποίο έχει ελεύθερη χρήση σε παγκόσμια κλίμακα, με αποτέλεσμα να μην χρειάζονται ειδικές άδειες αλλά και χρηματική επιβάρυνση για τη χρήση του. Πρόκειται για ένα εύρος συχνοτήτων που εκτείνεται μεταξύ των 2,4 GHz και 2,4835 GHz και ανήκει στις συχνότητες UHF (Ultra High Frequency).

Σε αντίθεση με το Bluetooth Classic, το πρωτόκολλο BLE χρησιμοποιεί λιγότερα κανάλια, για την ακρίβεια 40 έναντι των 79 που χρησιμοποιεί το πρώτο, έχοντας ως συνέπεια το εύρος κάθε καναλιού να ορίζεται στα 2 MHz. Με τον τρόπο αυτό, επιτυγχάνεται μείωση των παρεμβολών από τεχνολογίες και πρωτόκολλα που χρησιμοποιούν τις γειτονικές συχνότητες στο Bluetooth, αλλά και αντιστρόφως. Η προαναφερθείσα αύξηση του εύρους των καναλιών προσδίδει στην τεχνολογία ανοχή στο θόρυβο, πράγμα που σημαίνει ότι χρειάζεται μικρότερη απαιτούμενη εκπεμπόμενη ισχύ.



Σχήμα 5: Διαμόρφωση που χρησιμοποιεί το Bluetooth Low Energy

Η διαμόρφωση που χρησιμοποιεί το Bluetooth Low Energy βασίζεται στη διαμόρφωση μετατόπισης συχνότητας (FSK - Frequency-shift Keying), όπως φαίνεται στο σχήμα 5. Πρόκειται για έναν τύπο διαμόρφωσης σήματος, στον οποίο ψηφιακά δεδομένα παρουσιάζονται ως αλλαγές στη συχνότητα του φέροντος σήματος. Πιο συγκεκριμένα, το πρωτόκολλο BLE χρησιμοποιεί μια παραλλαγή του FSK, το Gaussian FSK (GFSK). Η παραλλαγή αυτή διαφοροποιείται με την πρωτότυπη εκδοχή ως προς το σήμα του φέροντος. Σε αυτήν, γίνεται χρήση ενός Gaussian φίλτρου, ώστε οι μεταβάσεις μεταξύ των διαφορετικών συχνοτήτων κωδικοποίησης να είναι πιο ομαλές.

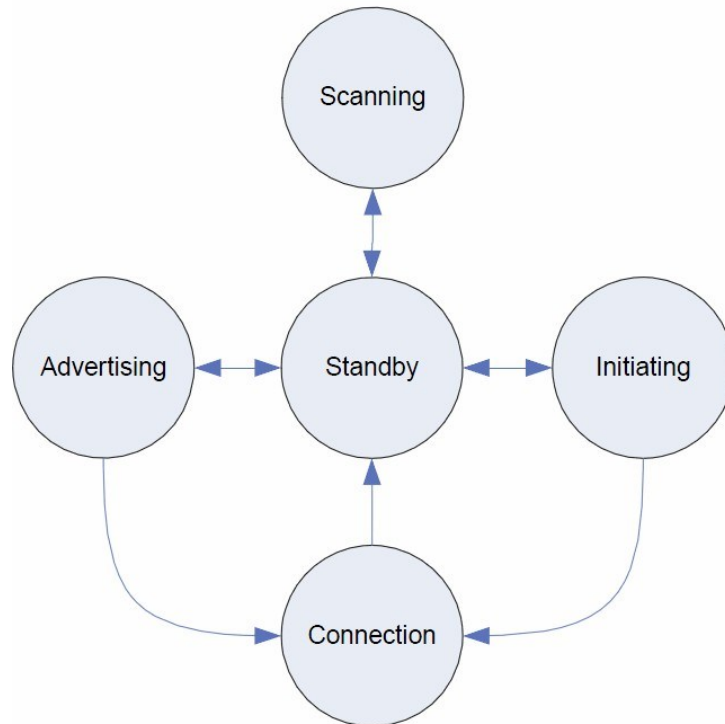
2.2.2 Το Επίπεδο Ζεύξης Δεδομένων

Το επίπεδο ζεύξης δεδομένων είναι μια μηχανή καταστάσεων, η οποία αποτελείται από τις εξής πέντε καταστάσεις: Standby, Advertising, Scanning (Active, Passive), Initiating, Connection (Master, Slave):

1. **Standby:** η κατάσταση στην οποία το επίπεδο ζεύξης δεδομένων (link layer) δεν εκπέμπει, ούτε λαμβάνει δεδομένα. Πρόκειται για την κεντρική κατάσταση της μηχανής καταστάσεων και συνδέεται με όλες τις υπόλοιπες.
2. **Advertising:** η κατάσταση στην οποία το επίπεδο ζεύξης δεδομένων εκπέμπει δεδομένα κάνοντας χρήση των advertising καναλιών και ελέγχει, ανά ορισμένη περίοδο χρόνου, αν λαμβάνονται απαντήσεις στα πακέτα που έστειλε προηγουμένως. Στην κατάσταση advertising μπορεί μια συσκευή να βρεθεί μόνο μετά την κατάσταση Standby, ενώ την πρώτη κατάσταση μπορεί να ακολουθήσει η κατάσταση Connection. Όταν μια συσκευή βρίσκεται σε αυτή την κατάσταση, αποκαλείται και advertiser.
3. **Scanning:** η κατάσταση στην οποία το επίπεδο ζεύξης δεδομένων ελέγχει τα advertising κανάλια για τυχόν πακέτα που εκπέμπονται από άλλες συσκευές. Η κατάσταση αυτή συνδέεται μόνο με την κατάσταση Standby. Οι συσκευές που βρίσκονται σε αυτό το στάδιο ονομάζονται scanners και συνήθως δεν έχουν αποσκοπούν στη σύνδεση με τις παραπάνω συσκευές.
4. **Initiating:** η κατάσταση στην οποία το επίπεδο ζεύξης δεδομένων ελέγχει τα advertising κανάλια για advertising πακέτα. Ωστόσο, στην κατάσταση αυτή, ο initiator (όπως αποκαλείται κάθε συσκευή που λειτουργεί στην παρούσα κατάσταση) περιμένει πακέτα από ορισμένη συσκευή, ώστε τη χρονική στιγμή που θα έρθουν, να απαντά και να εκκινεί σύνδεση με τη συσκευή που έστειλε αρχικά τα πακέτα. Η κατάσταση initiating είναι συνέχεια της κατάστασης Standby, ενώ από αυτήν μπορεί να ακολουθήσει η κατάσταση Connection.
5. **Connection:** η κατάσταση στην οποία το επίπεδο ζεύξης δεδομένων έρχεται μόνο ύστερα από advertising ή initiating. Ουσιαστικά, στην κατάσταση αυτή, γίνεται ο ορισμός των ρολών του αφέντη (master) και του σκλάβου (slave). Η συσκευή, η οποία ορίζεται ως αφέντης είναι αυτή που βρισκόταν σε κατάσταση Initiating,

ενώ η συσκευή που ορίζεται ως σκλάβος, είναι αυτή που βρισκόταν σε κατάσταση Advertising. Όπως είναι προφανές και από την ονομασία των συσκευών, η συσκευή που είναι υπεύθυνη για τον καθορισμό διάφορων παραμέτρων, σχετικών με την σύνδεση, είναι ο αφέντης.

Στο Σχήμα 6 παρουσιάζεται η μηχανή καταστάσεων του επιπέδου ζεύξης δεδομένων, με όλες τις καταστάσεις της και τις δυνατές μεταβολές μεταξύ των:



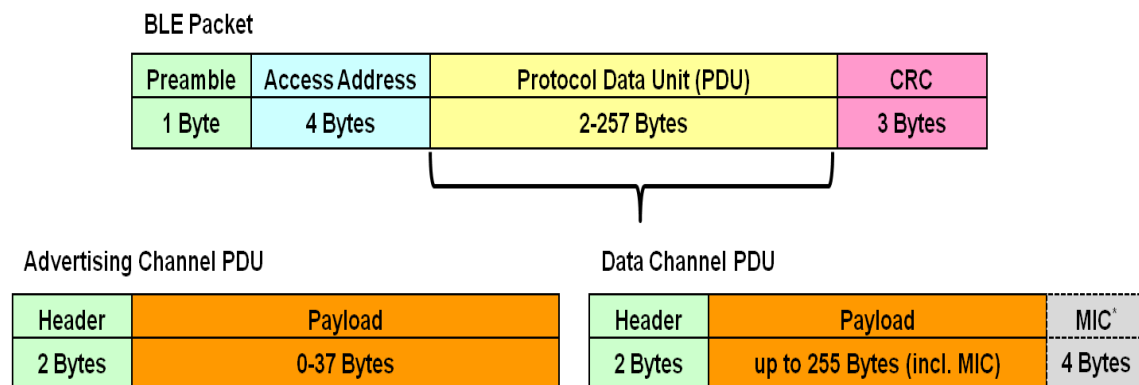
Σχήμα 6: Μηχανή καταστάσεων του επιπέδου ζεύξης δεδομένων

Παρόλο που η μηχανή καταστάσεων του επιπέδου ζεύξης δεδομένων μπορεί να βρίσκεται σε μία μόνο κατάσταση, γίνεται να εγκαταστήσουμε περισσότερες τέτοιες μηχανές, οι οποίες μπορούν να βρίσκονται η καθεμία σε διαφορετική κατάσταση. Στην πρώτη έκδοση της τεχνολογίας BLE, υπήρχαν κάποιοι περιορισμοί όσον αφορά στις καταστάσεις που μπορούν να βρίσκονται οι μηχανές μέσα σε μια συσκευή. Ωστόσο, στη δεύτερη έκδοση του πρωτοκόλλου Bluetooth Low Energy, όλοι οι περιορισμοί καταργήθηκαν. Μια συσκευή μπορεί να βρίσκεται ταυτόχρονα σε οποιαδήποτε κατάσταση. Αυτό σημαίνει ότι μια μηχανή - αφέντης σε σύνδεση μπορεί να είναι μηχανή - σκλάβος σε άλλη σύνδεση.

Οι συσκευές Bluetooth αναγνωρίζουν η μία την άλλη με τον ίδιο τρόπο που αναγνωρίζονται δυο συσκευές όταν βρίσκονται στο Ethernet δίκτυο. Έχουν μια εικονική διεύθυνση, η οποία δεν είναι τίποτα άλλο από μια αλληλουχία από 48 bits. Οι διευθύνσεις μπορούν να χωριστούν σε δυο κατηγορίες: τις δημόσιες (public) διευθύνσεις και τις τυχαίες (random) διευθύνσεις. Για να μπορεί να επικοινωνήσει μια συσκευή χρειάζεται τουλάχιστον μια εκ των δύο διευθύνσεων. Οι δημόσιες διευθύνσεις προγραμματίζονται και δηλώνονται κατά την παραγωγή και την κατασκευή των συσκευών. Αντιθέτως, οι τυχαίες διευθύνσεις καταχωρούνται συνήθως κατά την εκκίνηση της λειτουργίας τους.

Όσον αφορά στα κανάλια που παρέχει το Bluetooth Low Energy, προκύπτουν 40 με εύρος συχνοτήτων 2MHz και χωρίζονται σε δυο κατηγορίες: τα data κανάλια, που είναι τα 37 πρώτα, και τα advertising κανάλια, που είναι τα 3 τελευταία. Η θέση του κάθε καναλιού είναι συγκεκριμένη, καθώς με αυτόν τον τρόπο διασφαλίζεται η αποφυγή παρεμβολών με άλλα πρωτόκολλα μεταφοράς δεδομένων που λειτουργούν σε αυτό το φάσμα συχνοτήτων. Τα data κανάλια χρησιμοποιούνται αποκλειστικά για τη μεταφορά δεδομένων μεταξύ συσκευών που έχουν ήδη συνδεθεί. Για να μειωθούν οι παρεμβολές, το Bluetooth χρησιμοποιεί το Adaptive Frequency Hopping. Αυτό είναι μια τεχνική, στην οποία το κανάλι που χρησιμοποιείται για να μεταδοθεί η πληροφορία αλλάζει ανά μια προκαθορισμένη περίοδο χρόνου με σκοπό την αποφυγή παρεμβολών. Ο ρόλος των advertising καναλιών είναι η παροχή δυνατότητας στις συσκευές να αναζητούν, να ανακαλύπτουν και, εν τέλει, να συνδέονται με άλλες συσκευές στο χώρο. Τα κανάλια αυτά είναι διάσπαρτα μέσα στο φάσμα συχνοτήτων του Bluetooth, ώστε να μην υπάρχει επικάλυψη ή θόρυβος από άλλες τεχνολογίες.

Όπως στα κανάλια, έτσι και στα πακέτα που στέλνει μια συσκευή μέσω του πρωτοκόλλου Bluetooth Low Energy, υπάρχουν δύο κατηγορίες: τα advertising πακέτα και τα πακέτα δεδομένων. Προφανώς, τα πρώτα μεταδίδονται μέσω των advertising καναλιών και τα δεύτερα μέσω των data καναλιών. Τα πακέτα μεταδίδονται bit προς bit, με το λιγότερο σημαντικό bit (Least Significant Bit) να μεταδίδεται πρώτο. Συγκριτικά με άλλα πρωτόκολλα μεταφοράς δεδομένων, τα πακέτα στέλνονται μέσω του BLE είναι σχετικά μικρά. Ο λόγος που το μέγεθος των πακέτων είναι μικρό συνδέεται άμεσα με το γεγονός ότι η τεχνολογία αυτή σχεδιάστηκε έχοντας ως γνώμονα τη μεταφορά μικρού όγκου πληροφορίας σε σύντομο χρονικό διάστημα. Σίγουρα, καθοριστικό ρόλο διαδραμάτισε και το σύνολο των ηλεκτρικών στοιχείων, καθώς η χρήση της τεχνολογίας CMOS προϋποθέτει να μην αυξάνεται ιδιαίτερα η θερμοκρασία γιατί θα έχει σαν αποτέλεσμα τη μεταβολή της συχνότητας λειτουργίας τους. Για σταθερή θερμοκρασία θα χρειαζόταν προσθήκη αρκετών ακόμη στοιχείων, το οποίο σημαίνει αύξηση του κόστους κατασκευής. Έτσι, οι συσκευές που χρησιμοποιούν το Bluetooth Low Energy δεν πρέπει να έχουν απαιτήσεις που αφορούν σε μετάδοση μεγάλης ποσότητας δεδομένων. Η δομή ενός πακέτου Bluetooth Low Energy φαίνεται στο Σχήμα 7:



Σχήμα 7: Δομή ενός πακέτου Bluetooth Low Energy

Όπως και σε ένα πακέτο Ethernet, το προοίμιο (preamble ή header) είναι μια ακολουθία των 8 bits που βρίσκεται πάντα στην αρχή κάθε πακέτου. Το προοίμιο είναι απαραίτητο για να αναγνωρίσει τις συχνότητες μετάδοσης του πομπού, καθώς και για τον συγχρονισμό της μετάδοσης δεδομένων μεταξύ δύο ή περισσότερων συσκευών. Το access address είναι μια ακολουθία των 32 bits, το οποίο είναι υπεύθυνο για την αναγνώριση και το διαχωρισμό όλων των συνδέσεων μιας συσκευής. Πιο συγκεκριμένα, το access address είναι ικανό να αναγνωρίζει την αρχή, δηλαδή το προοίμιο ενός πακέτου. Κάτι τέτοιο είναι πολύ σημαντικό, αφού η συσκευή δεν μπορεί να γνωρίζει πότε θα ξεκινήσει η άλλη πλευρά την μετάδοση.

Όπως φαίνεται και στο Σχήμα 7, τα headers είναι διαφορετικά για τα δύο είδη πακέτων. Οι βασικότερες διαφορές τους σχετίζονται με την ύπαρξη 2 bits στα advertising πακέτα που ορίζουν τον τύπο της διεύθυνσης του peer που λειτουργεί ή ως πομπός ή ως δέκτης και την ύπαρξη 2 bits στα data πακέτα που χρησιμοποιούνται για να αντιλαμβάνεται η συσκευή ότι εκκινείται ή συνεχίζεται η διαδικασία μετάδοσης του μηνύματος. Στο σημείο αυτό, αξίζει να σημειωθεί ότι η επιβεβαίωση των πακέτων βασίζεται στη μέθοδο lazy acknowledgement, κατά την οποία δεν είναι απαραίτητη η επιβεβαίωση λήψης ενός πακέτου τη χρονική στιγμή που αυτή πραγματοποιείται. Η συσκευή έχει τη δυνατότητα να περιμένει να στείλει αυτή την επιβεβαίωση αργότερα, όταν το επόμενο πακέτο είναι μεγαλύτερο. Αυτό συνεπάγεται αύξηση της αποδοτικότητας και μείωση της κατανάλωσης ενέργειας.

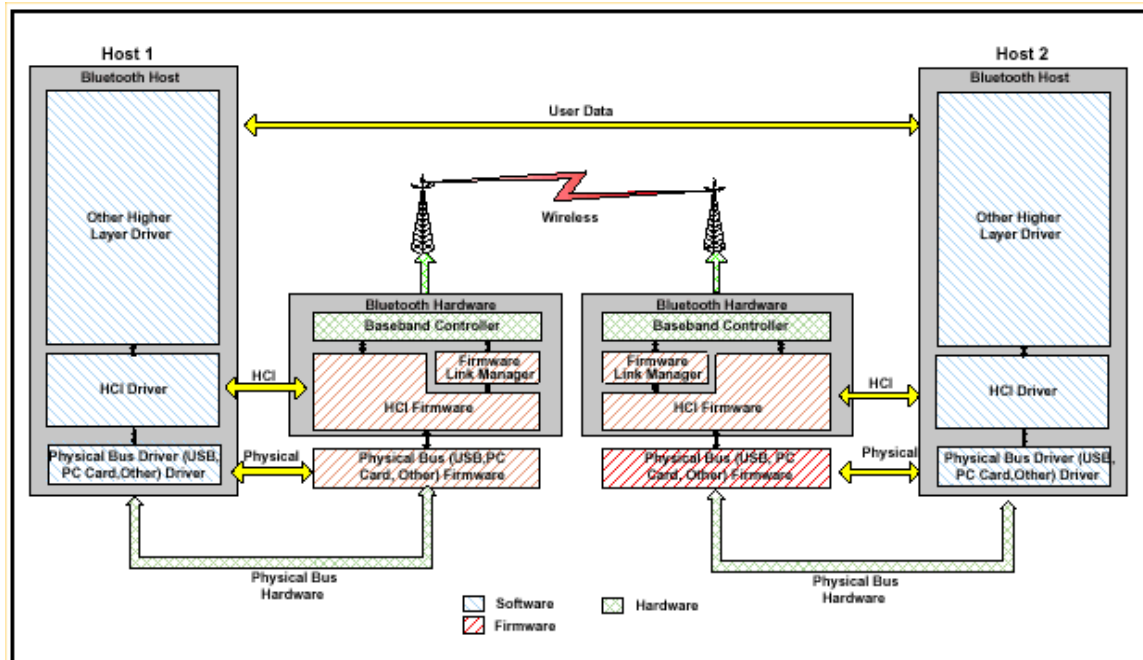
Τα δύο τμήματα της δομής ενός πακέτου, που περιγράφηκαν προηγουμένως, τοποθετούνται πάντα στην αρχή κάθε πακέτου. Στη συνέχεια ακολουθεί η πληροφορία που θέλει ο εκάστοτε χρήστης να μεταδώσει. Στο τέλος κάθε πακέτου βρίσκεται ο έλεγχος κυκλικού πλεονασμού (cyclic redundancy check - CRC) με συνολικό μήκος 24 bits. Το CRC είναι ένας μαθηματικής φύσεως έλεγχος δεδομένων που συμπεριλαμβάνεται στον υπολογισμό του μήκους ενός πακέτου, σε αντίθεση με το προοίμιο και το access address. Ο ρόλος του είναι ο εντοπισμός όλων των σφαλμάτων.

Όπως αναλύσαμε και παραπάνω, η συσκευή που βρίσκεται σε advertising state ονομάζεται advertiser. Η διαδικασία που ακολουθεί ο advertiser είναι περιοδική και ονομάζεται advertising event. Τόσο η επιλογή καναλιού μετάδοσης, όσο και η επιλογή του advertising interval, δηλαδή του χρόνου που μεσολαβεί μεταξύ των advertising γεγονότων καθορίζονται από ανώτερα επίπεδα, και πιο συγκεκριμένα από τη διεπαφή Host Controller. Πριν την έναρξη ενός advertising event, ο advertiser ελέγχει εάν έχουν ληφθεί αιτήματα για scan ή connection (SCAN_REQ, CONNECT_REQ). Σε περίπτωση που λάβει scanning request, αυτός απαντά (SCAN_RSP), παρέχοντας στον scanner τις απαιτούμενες πληροφορίες. Έτσι, το advertising event συνεχίζεται. Ωστόσο, εάν λάβει connection request, τερματίζει το advertising event και εκκινεί η διαδικασία σύνδεσης.

2.2.3 Διεπαφή Host Controller

Η διεπαφή Host Controller (HCI) επιτρέπει την επικοινωνία, με τη χρήση εντολών, μεταξύ του baseband controller - processor (το τμήμα της συσκευής που

διαχειρίζεται όλες τις λειτουργίες μετάδοσης) και του host. Ο τελευταίος μέσω αυτής της διεπαφής μπορεί να έχει πρόσβαση στην κατάσταση του hardware και να διαμορφώνει, όπως αυτός επιθυμεί, τους καταχωρητές.



Σχήμα 8: Διεπαφή Host Controller

Η τεχνολογία Bluetooth Low Energy παρέχει μια σειρά διαφορετικών transport layers, με βάση τα οποία πραγματοποιείται η επικοινωνία του HCI. Αυτά είναι:

1. **UART Transport Layer:** Ο σκοπός του HCI UART Transport Layer είναι να καταστήσει δυνατή την επικοινωνία μεταξύ δυο UARTs (κύκλωμα υπολογιστών το οποίο διαμεσολαβεί στην σειριακή επικοινωνία υπολογιστών) στο ίδιο PCB (πλακέτα τυπωμένου κυκλώματος). Παρ' όλα αυτά, το UART προϋποθέτει ότι δεν υπάρχουν λάθη στις γραμμές. Τα πακέτα διατρέχουν αυτό το layer, αλλά δεν γίνεται η αποκωδικοποίησή τους.
2. **RS232 Transport Layer:** Ο τρόπος λειτουργίας του και οι δυνατότητές τους προσεγγίζουν αρκετά αυτά του προηγούμενου transport layer, καθώς δεν γίνεται ούτε εδώ η αποκωδικοποίηση των πακέτων.
3. **USB Transport Layer:** Ο σκοπός του Ενιαίου Σειριακού Διαύλου (Universal Serial Bus - USB) είναι η επίτευξη επικοινωνίας μέσω μιας διεπαφής USB hardware σχεδιασμένης για την Bluetooth τεχνολογία. Το βασικό πλεονέκτημα, σε σχέση με τις άλλες δύο επιλογές, είναι η ταχύτητα μεταφοράς δεδομένων.

2.2.4 Logical Link Control - Adaptation Protocol (L2CAP)

Η διεπαφή Host Controller (HCI) επιτρέπει την επικοινωνία, με τη χρήση εντολών, μεταξύ του baseband controller. Αυτό επιτυγχάνεται με την πολυπλεξία των δεδομένων των πρωτοκόλλων των ανωτέρων επιπέδων και με τη διάσπαση και

ανασύνθεση πακέτων δεδομένων , τα οποία είναι μεγαλύτερα από το μέγιστο μήκος πακέτου του link layer (27 bytes). Η λειτουργία του συγκεκριμένου τμήματος βασίζεται στην χρήση πολλών διαφορετικών καναλιών. Στην τεχνολογία BLE, γίνεται χρήση των καναλιών: 4 για τα δεδομένα του Attribute protocol, 6 για το SMP και 5 για την αποστολή εντολών που ορίζονται στο L2CAP. Παράλληλα, είναι δυνατό να δημιουργούνται κανάλια δυναμικά, ώστε να μπορούν να χρησιμοποιηθούν για μεταφορά δεδομένων με μεγαλύτερο throughput. Η μορφή του πακέτου του L2CAP είναι η εξής:

Length	CID	Data
2 bytes	2 bytes	0 – 65535 bytes

Το πεδίο Length ορίζει το μήκος του πακέτου που ακολουθεί. Το CID είναι το αναγνωριστικό κάθε καναλιού και χρησιμοποιείται για να γίνεται εύκολα και γρήγορα η διάκριση των διαφορετικών καναλιών. Τέλος, το πεδίο Data αποτελεί το ωφέλιμο φορτίο που στέλνεται. Αφού το μέγιστο μήκος πακέτου είναι 27 bytes, το τμήμα data μπορεί να είναι έως 23 bytes.

2.2.5 Security Manager Protocol (SMP)

Το πρωτόκολλο διαχείρισης ασφάλειας (SMP) ορίζει όλες τις διαδικασίες, με τις οποίες οι συσκευές δημιουργούν κλειδιά κρυπτογράφησης, τα ανταλλάζουν μεταξύ τους και τα αποκωδικοποιούν. Το SMP δίνει στη χρήστη τη δυνατότητα να βεβαιωθεί για την αυθεντικότητα του αποστολέα και να αποκρύψει τη δημόσια διεύθυνση της συσκευής, με σκοπό την αποφυγή οποιασδήποτε αλληλεπίδρασης με κακόβουλους χρήστες. Κατά τη σύνδεση δυο συσκευών, γίνεται ανταλλαγή κλειδιών, ούτως ώστε οι συσκευές να επικοινωνούν σε ασφαλές και κρυπτογραφημένο κανάλι. Η ζεύξη αυτή ονομάζεται και authenticated link. Η κύρια χρήση της είναι η προστασία από επιθέσεις τύπου Man in the Middle. Ανάλογα με τον τρόπο δημιουργίας του, διακρίνονται σε τρία επίπεδα ασφαλείας:

- υποστηρίζει επικοινωνία χωρίς καθόλου ασφάλεια και εφαρμόζεται σε οποιαδήποτε Bluetooth σύνδεση
- υποστηρίζει κρυπτογράφηση AES-CMAC, χωρίς όμως προστασία από επιθέσεις τύπου MITM
- υποστηρίζει κρυπτογράφηση ECDHE και παρέχει προστασία σε επιθέσεις τύπου MITM. Είναι το ανώτερο επίπεδο ασφαλείας

Το pairing είναι το σημείο όπου οποιαδήποτε, με την ασφάλεια σχετική, διαδικασία λαμβάνει χώρα. Σκοπός του είναι να αντιληφθεί τι ζητά κάθε πλευρά και στη συνέχεια να τις ετοιμάσει για σύνδεση. Η διαδικασία του pairing χρειάζεται τρεις φάσεις για να ολοκληρωθεί: την γνωστοποίηση προς την άλλη πλευρά των

δυνατοτήτων της συσκευής, την παραγωγή κλειδιού και εν τέλει τη διαμοίρασή του.

2.2.6 Generic Access Profile (GAP)

Το Generic Access Profile ορίζει τρόπους λειτουργίας, ρόλους και διαδικασίες, όπου δυο συσκευές χρησιμοποιούν για να ανακαλύψουν άλλες και να συνδεθούν. Στο GAP ορίζονται τέσσερις βασικοί ρόλοι, με τους οποίους εμφανίζεται μια συσκευή:

1. **Broadcaster:** Συσκευή, η οποία στέλνει πακέτα advertising σε κάθε περίοδο χρόνου.
2. **Observer:** Συσκευή, η οποία ελέγχει αν τα advertising πακέτα, που δέχεται στα κανάλια της, απευθύνονται σε αυτήν.
3. **Peripheral:** Συσκευή, η οποία εκτελεί συνεχώς τη διαδικασία του advertising για να δηλώσει προς τις περιφερειακές συσκευές ότι είναι διαθέσιμη για σύνδεση.
4. **Central:** Συσκευή, η οποία ελέγχει τα advertising κανάλια, ώστε να όταν αναγνωρίσει αυτό που επιθυμεί να αρχίσει τη διαδικασία σύνδεσης.

Αντιστοίχως, έχουμε τέσσερα είδη τρόπων λειτουργίας και διαδικασιών, τα οποία σχετίζονται άμεσα με τον ρόλο που προσδίδει το GAP στη συσκευή. Επομένως για έναν broadcaster έχουμε broadcast mode και procedure, για έναν observer έχουμε discovery mode και procedure κ.ο.κ.

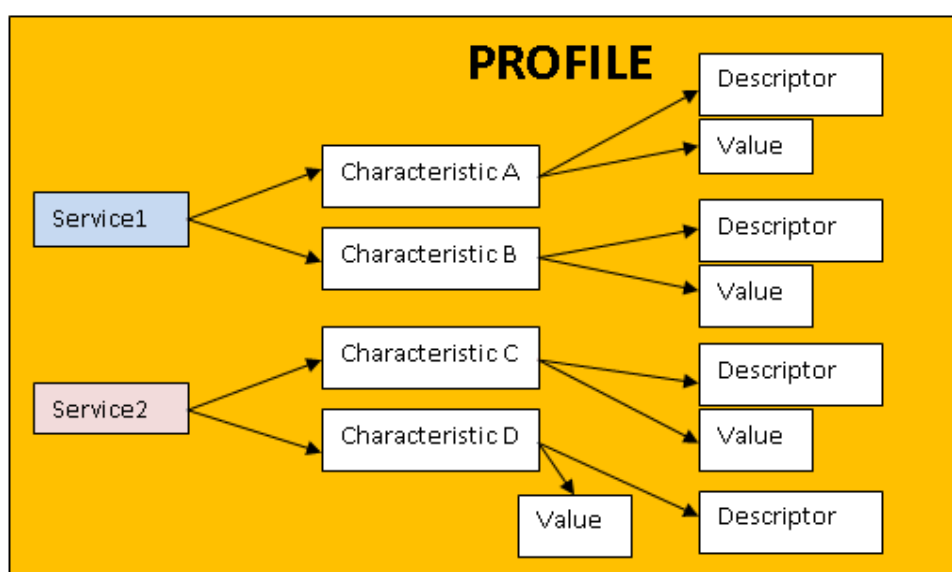
2.2.7 Attribute Protocol (ATT)

Το Attribute Protocol (ATT) ορίζει και διαχειρίζεται μια δομή, η οποία ονομάζεται attribute. Αυτό αποτελείται από έναν 16-bit handler, ένα UUID που καθορίζει το attribute type και ένα value συγκεκριμένου μήκους. Το τελευταίο είναι μια αλληλουχία από bytes, με μέγεθος το πολύ 512 bytes. Σε περίπτωση που ο όγκος της πληροφορίας ξεπερνά τα 512 bytes, τότε στέλνεται σε περισσότερα πακέτα. Η τιμή του handler είναι απλά ένας μοναδικός αριθμός (μεταξύ του 0 και του 65535) που είναι χαρακτηριστικός για το συγκεκριμένο attribute. Όσον αφορά στο attribute type, αυτό αντιστοιχεί σε ένα μοναδικό UUID (Universally Unique Identifier). Αυτό ορίζεται συνήθως από το αμέσως υψηλότερο επίπεδο: το GATT.

Εκτός από αυτά, υπάρχουν και ειδικές άδειες (permissions), που σχετίζονται με το attribute στο οποίο περιγράφονται. Είναι καταχωρημένες σε ένα ορισμένο σημείο μέσα στο attribute value, χωρίς όμως να ελέγχονται από το ATT. Τα permissions χωρίζονται σε τρεις βασικές κατηγορίες: permissions πρόσβασης (access), κρυπτογράφησης (encryption), πιστοποίησης αυθεντικότητας (authentication) και εξουσιοδότησης (authorization). Σαφώς, σε ένα attribute είναι δυνατό να αντιστοιχούν περισσότερες από μια άδειες.

2.2.8 Generic Attribute Profile (GATT)

Το Generic Attribute Profile (GATT) είναι το αμέσως ανώτερο επίπεδο από το ATT. Χρησιμοποιεί τα attributes που διαμορφώνονται στο ATT ομαδοποιώντας τα με σκοπό την οργάνωσή τους. Η βάση της πυραμίδας ιεραρχίας του GATT είναι το προφίλ (profile). Το προφίλ αποτελείται από μία ή περισσότερες υπηρεσίες (services). Κάθε υπηρεσία αποτελείται ένα πλήθος χαρακτηριστικών (characteristic). Κάθε χαρακτηριστικό περιέχει μία τιμή, καθώς και επιπλέον δεδομένα, τα οποία σχετίζονται με την τιμή και ονομάζονται descriptors. Οι τιμές των χαρακτηριστικών είναι αυτές που χρησιμοποιούν οι εφαρμογές BLE για να φέρουν εις πέρας όλες τις επιθυμητές λειτουργίες τους. Μια υπηρεσία μπορεί να χρησιμοποιεί μεθόδους κάποιας άλλης υπηρεσίας με την διαδικασία του include. Η ιεραρχία και η δομή ενός προφίλ απεικονίζεται στο Σχήμα 9:



Σχήμα 9: Ιεραρχία και η δομή ενός προφίλ

Επιπροσθέτως, το GATT παρέχει ένα σύνολο λειτουργιών, οι οποίες δίνουν τη δυνατότητα σε έναν client να μπορεί να ανακαλύψει τη δομή του προφίλ, να εξερευνήσει και να ανακαλύψει όλες τις πληροφορίες που προσφέρονται από τις τιμές των χαρακτηριστικών και πιθανώς να τις τροποποιήσει. Οι βασικές λειτουργίες του GATT είναι οι εξής:

1. Characteristic Discovery
2. Characteristic Descriptor Discovery
3. Characteristic Value Read
4. Characteristic Value Write
5. Characteristic Value Notification
6. Characteristic Value Indication
7. Characteristic Descriptor Value Read
8. Characteristic Descriptor Value Write

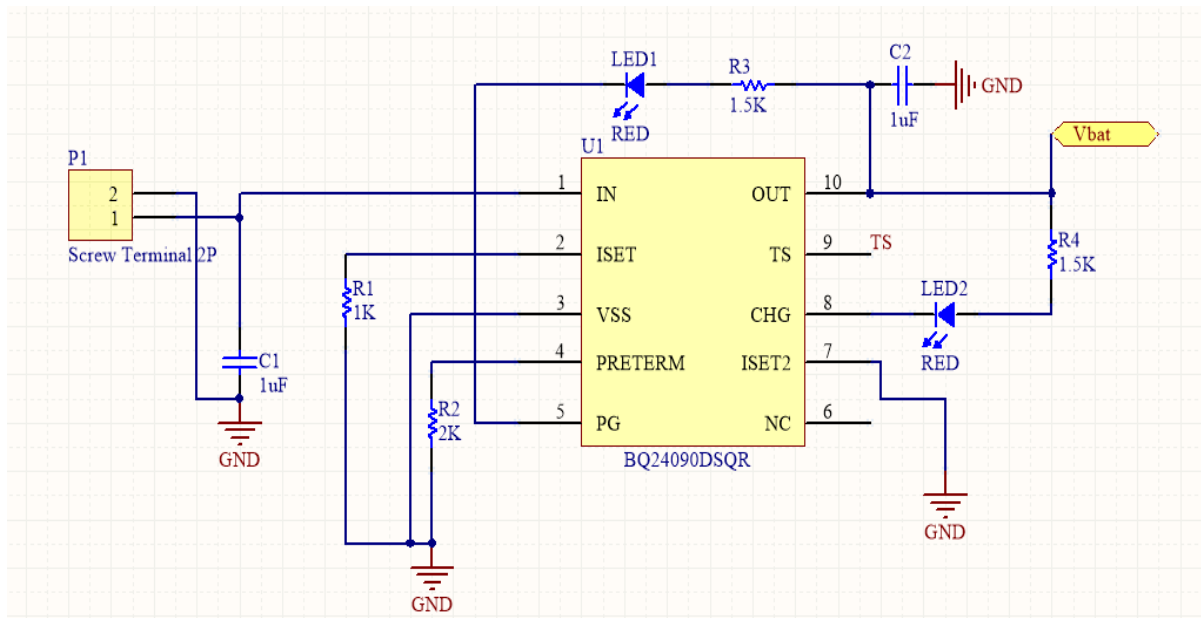
Κεφάλαιο 3

Development Board DA1458

Για την υποστήριξη του Bluetooth Low Energy πρωτοκόλλου, πραγματοποιήθηκε σχεδίαση και κατασκευή αναπτυξιακής πλακέτας. Το κατασκευαστικό μέρος αποτελείται από τρία μέρη: **α)** τον φορτιστή (battery charger), **β)** τον buck-boost converter και **γ)** τον μικροελεγκτή DA14585, τα οποία και θα αναλυθούν διεξοδικά στη συνέχεια

3.1 Battery Charger BQ24090DSQR

Ένα πλεονέκτημα της συγκεκριμένης συσκευής είναι η χαμηλή κατανάλωση ισχύος. Αυτό δίνει τη δυνατότητα να γίνεται χρήση μπαταρίας, η οποία μάλιστα θα τροφοδοτεί όλο το σύστημα για μεγάλο χρονικό διάστημα (πολλές εβδομάδες). Ωστόσο, είναι προφανές πως η μπαταρία θα χρειάζεται περιστασιακά να φορτίζεται. Η λειτουργία, λοιπόν, του battery charger είναι να φορτίζει την μπαταρία της συσκευής, όταν αυτή αποφορτίζεται. Αυτό επιτυγχάνεται μέσω φωτοβολταϊκού πάνελ με έξοδο V_{out} 5V.

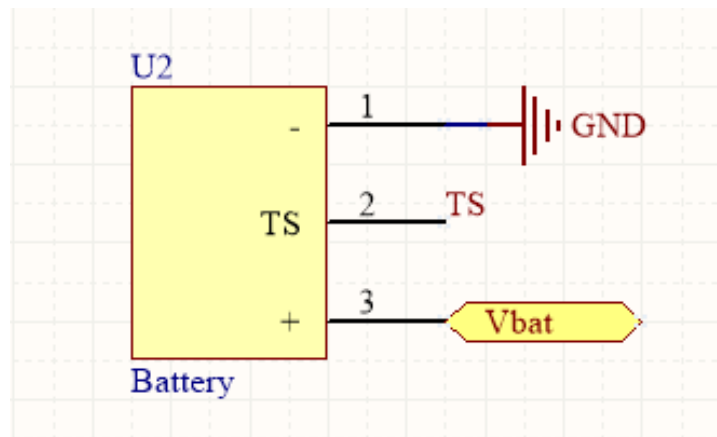


Σχήμα 10: Battery Charger

Όπως απεικονίζεται στο Σχήμα 10, ο πυκνωτής (bypass) C_1 τοποθετείται στην είσοδο του φορτιστή προκειμένου να συμβάλει στη μείωση του θορύβου. Η αντίσταση R_1 συνδέεται με την επαφή ISET, η οποία ρυθμίζει την ταχύτητα φόρτισης στην περίπτωση που ο φορτιστής τροφοδοτείται από adapter και όχι από USB, όπου θα χρειαζόταν τοποθέτηση αντίστασης σε τιμή κοντά σε αυτή της R_1 στην υποδοχή ISET2. Η αντίσταση R_2 , που συνδέεται με την επαφή PRETERM, επιλέγεται συνήθως σε τιμή διπλάσια της R_1 . Στην επαφή CHG έχει συνδεθεί μια φωτοδίοδος (LED), ώστε ο χρήστης να αντιλαμβάνεται πότε η μπαταρία φορτίζεται. Όταν η τάση εισόδου είναι μεγαλύτερη από την τάση εξόδου, πράγμα το οποίο δεν συνάδει με την ορθή λειτουργία του φορτιστή, ενεργοποιείται και η φωτοδίοδος που είναι συνδεδεμένη στην επαφή PG. Η επαφή Vss αναφέρεται στη γείωση, ενώ η επαφή TS σχετίζεται με τον έλεγχο της θερμοκρασίας. Επιπλέον, η επαφή NC χρησιμοποιείται για

εσωτερικές λειτουργίες του ολοκληρωμένου κυκλώματος. Τέλος, από την επαφή OUT ξεκινά η έξοδος του φορτιστή, η οποία συνεχίζει σαν είσοδος στο δεύτερο μέρος της πλακέτας. Η τάση εξόδου είναι επίσης 5V. Στο σημείο αυτό του κυκλώματος θα μπορούσαν, επίσης, να χρησιμοποιηθούν πυκνωτές (bypass) για να μειώνουν τον θόρυβο. Ωστόσο, τέτοιοι πυκνωτές υπάρχουν στην είσοδο του επόμενου μέρους, με αποτέλεσμα να μην χρειάζονται εδώ.

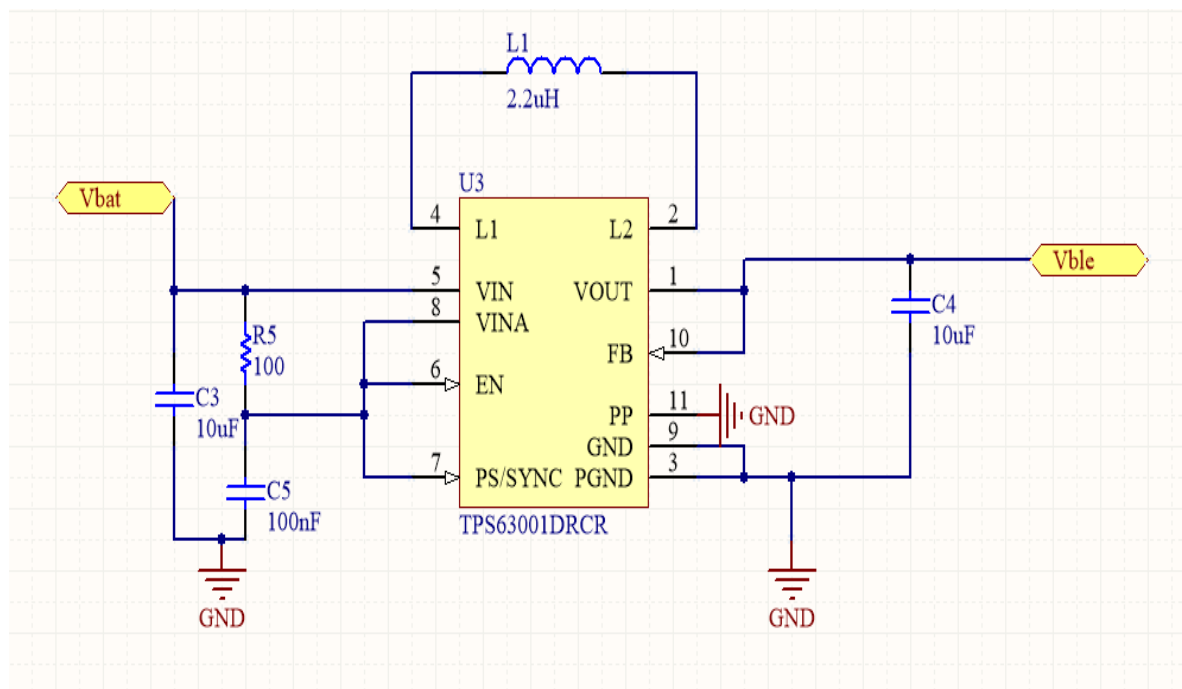
Η μπαταρία που χρησιμοποιήθηκε είναι μια τυπική μπαταρία ιόντων λιθίου 850mAh και 3.7V τροφοδοσίας. Το σχηματικό της μπαταρίας παρουσιάζεται στο Σχήμα 11:



Σχήμα 11: Battery

3.2 Buck-Boost Converter TPS63001DRCR

Ο μικροελεγκτής DA14585 της Dialog Semiconductor χρειάζεται, για την ορθή λειτουργία του, τάση εισόδου V_{ble} ίση με 3.3V. Επειδή όμως η έξοδος της μπαταρίας και του φορτιστή είναι 3.7 V, είναι απαραίτητη η χρήση μετατροπέα τάσης ώστε να επιτυγχάνεται η επιθυμητή τάση εισόδου στον μικροελεγκτή.



Σχήμα 12: Buck-Boost Converter

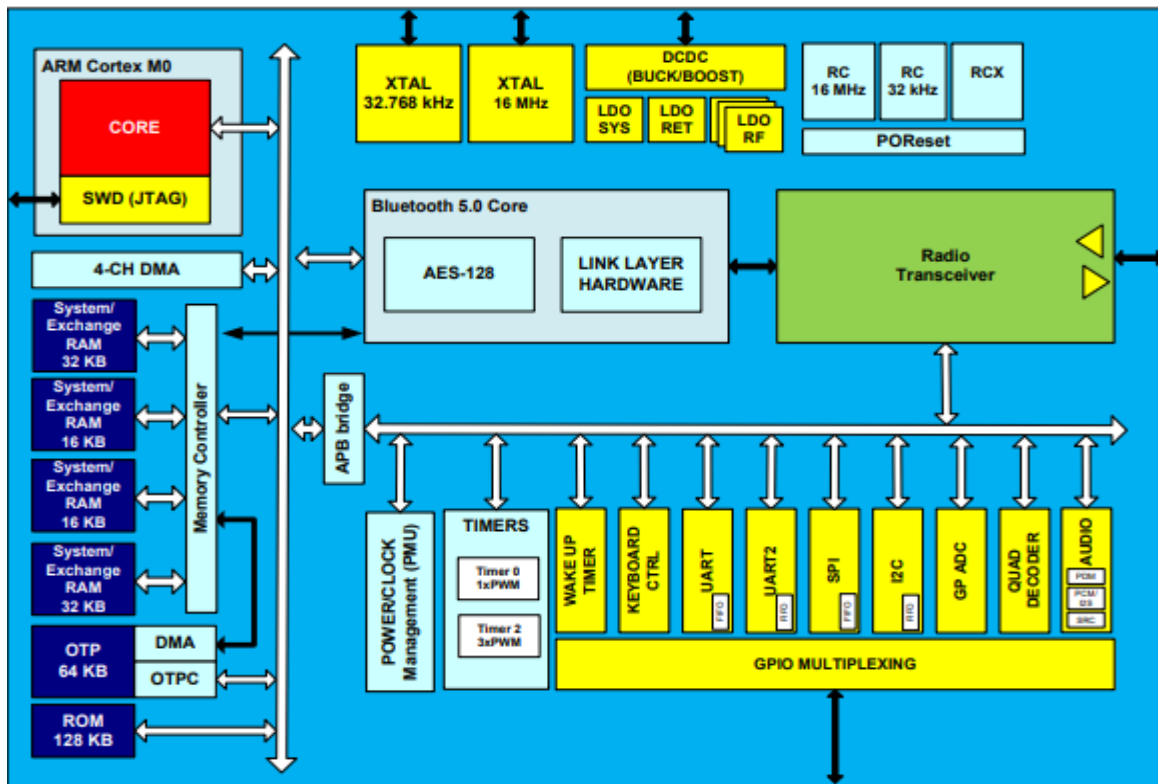
Στον συγκεκριμένο buck-boost converter περιέχεται ο controller, ο οποίος υπολογίζει συνεχώς το μέσο ρεύμα εισόδου, καθώς και το μέγιστο ρεύμα εισόδου. Αυτός ελέγχει και περιορίζει, όταν αυτό χρειάζεται, την καταναλισκόμενη ισχύ, έτσι ώστε να επιτυγχάνεται ασφαλής και σταθερή λειτουργία κάτω από οποιεσδήποτε συνθήκες. Στο παραπάνω σχηματικό, οι επαφές που σχετίζονται με την λειτουργία του controller, είναι η VINA, η οποία τον τροφοδοτεί με την τάση εισόδου, η EN, η οποία ενεργοποιεί και απενεργοποιεί την παροχή στην είσοδο, και η PS/SYNC, η οποία σχετίζεται με την ενεργοποίηση ή μη της κατάστασης εξοικονόμησης ενέργειας. Η επαφή VIN αποτελεί την είσοδο του buck-boost converter. Οι (bypass) πυκνωτές C_3 και C_5 χρησιμοποιούνται για τον περιορισμό του θορύβου, ενώ η R_5 ρυθμίζει την τάση VINA. Μεταξύ των επαφών L1 και L2 τοποθετήθηκε πηνίο, το οποίο είναι απαραίτητο για την buck-boost λειτουργία. Τέλος, Η επαφή FB συνδέεται με την επαφή VOUT, η οποία αποτελεί και την έξοδο του DC-DC converter. Σε αυτό το σημείο γίνεται χρήση ενός ακόμη (bypass) πυκνωτή, ο οποίος εκτός από τον περιορισμό του θορύβου, συμβάλλει επικουρικά και στην σταθεροποίηση της τάσης εξόδου.

3.3 Microcontroller DA14585

Για την υλοποίηση της σχεδίασης και της ανάπτυξης του διαδραστικού συστήματος για ξενάγηση σε αρχαιολογικούς χώρους με χρήση του πρωτοκόλλου Bluetooth 5.0 έγινε χρήση του ενσωματωμένου συστήματος DA14585. Το DA14585 είναι ένα Bluetooth Low Energy System on Chip (SoC), το οποίο αποτελεί προϊόν της Dialog Semiconductor. Το SoC αυτό περιλαμβάνει όλα τα απαραίτητα στοιχεία για την λειτουργία BLE εφαρμογών, δηλαδή αποτελείται από έναν πομποδέκτη που εκπέμπει και λαμβάνει σε συχνότητα 2.4GHz, έναν μικροελεγκτή ARM Cortex-M0 με 96 kB RAM και μία One-Time Programmable (OTP) μνήμη χωρητικότητας 64kB. Επιπλέον, περιλαμβάνει BLE controller και baseband, ενώ το firmware παρέχει πληθώρα υπηρεσιών, πάνω στις οποίες μπορούν να σχεδιασθούν και να πραγματοποιηθούν οι εφαρμογές.

Το DA14585 είναι ένα small form factor chipset, το οποίο μπορεί να αποτελεί τη βάση για τη δημιουργία ενός πλήρους Bluetooth Smart συστήματος. Είναι ιδανικό για χρήση σε μεγάλο αριθμό εφαρμογών BLE, λόγω της μικρής κατανάλωσης ισχύος, τόσο κατά την εκπομπή όσο και κατά την λήψη δεδομένων. Έτσι, ακόμη και με μία κοινή μπαταρία, μπορεί να λειτουργεί για μεγάλα χρονικά διαστήματα.

Στο Σχήμα 13 φαίνονται τα κύρια στοιχεία, από τα οποία αποτελείται το DA14585 Soc. Ο πυρήνας του Bluetooth 5.0 περιέχει εξειδικευμένο επεξεργαστή για το Link Layer, ενώ υπάρχει και hardware υλοποίηση του αλγορίθμου AES-128.



Σχήμα 13: Bluetooth SoC

Επιπροσθέτως, ο μικροελεγκτής DA14585 παρέχει στον χρήστη ένα σύνολο περιφερειακών με σκοπό την επικοινωνία με εξωτερικά στοιχεία. Για κάθε περιφερειακό, υπάρχει ένας τουλάχιστον καταχωρητής, ο οποίος είναι υπεύθυνος για τον έλεγχο της ορθής λειτουργίας του chip. Ένα πλεονέκτημα, το οποίο συμβάλλει στην εύκολη χρήση του, αποτελεί, ακόμη, η παροχή drivers για όλα σχεδόν τα περιφερειακά, είτε από το firmware, είτε από το Software Development Kit (SDK) του αναπτυξιακού. Για την σωστή και εύκολη χρήση των περιφερειακών, διατίθενται General-Purpose Input/Output (GPIO) pins, τα οποία μπορούν να ελέγχονται ανά πάσα στιγμή από τον προγραμματιστή και να λειτουργούν είτε σαν είσοδοι είτε σαν έξοδοι του chip. Τα pins αυτά, έχουν τη δυνατότητα να διατηρούν την τιμή τους, ακόμα και κατά τη διάρκεια ενός sleep mode. Τα υποστηριζόμενα περιφερειακά είναι τα εξής:

1. **UART (Universal Asynchronous receiver/transmitter):** Δύο σειριακές θύρες με ταχύτητα μετάδοσης πληροφορίας έως και 1Mbps
2. **SPI (Serial Peripheral Interface):** Μια σύγχρονη full duplex επικοινωνία για μικρές αποστάσεις
3. **I2C bus:** Ένας σειριακός δίαυλος που λειτουργεί είτε στα 100kHz είτε στα 400kHz
4. **3-axes capable Quadrature Decoder:** Αποκωδικοποιητής σημάτων ενός rotary encoder, όπως αυτοί που εμφανίζονται σε συσκευές τύπου HID, π.χ. ποντίκι υπολογιστή

Ο μικροελεγκτής DA14585 έχει σχεδιασθεί, ώστε να μπορεί να λειτουργεί με τέσσερις διαφορετικούς τρόπους (power modes). Ο πρώτος δεν είναι άλλος από τον ενεργό τρόπο λειτουργίας (active mode), κατά τη διάρκεια του οποίου το σύστημα είναι ενεργό, δηλαδή όλες οι λειτουργίες του είναι ενεργές και σε πλήρη ταχύτητα. Οι υπόλοιποι τρεις τρόποι λειτουργίας ανήκουν στην κατηγορία του sleep mode. Κάθε ένα από αυτά απενεργοποιεί έναν αριθμό στοιχείων, όταν αυτά δεν απαιτούνται από την εκάστοτε εφαρμογή, με σκοπό την

ακόμη μεγαλύτερη μείωση κατανάλωσης ισχύος. Τα τρία sleep modes που παρέχονται από τον σχεδιαστή είναι:

- 1. Sleep Mode:** Αποτελεί το συνηθέστερο sleep mode, στο οποίο εισέρχεται ο επεξεργαστής όταν όλες οι διαδικασίες του προγράμματός έχουν περατωθεί. Ο επεξεργαστής βρίσκεται σε idle κατάσταση, μέχρι να εμφανισθεί κάποια διακοπή (interrupt). Τα power domains του συστήματος (PD_SYS) είναι ενεργά, ενώ τα περιφερειακά (PD_PER) εξαρτώνται πλήρως από ό,τι έχει προβλεφθεί από τον προγραμματιστή στις τιμές των καταχωρητών του συστήματος.
- 2. Extended Sleep Mode:** Όλα τα power domains του συστήματος είναι απενεργοποιημένα εκτός από το Always On (PD_AON), το οποίο αποτελεί τη γραμμή τροφοδοσίας σε όλα τα στοιχεία που πρέπει να λειτουργούν συνεχώς. Ενεργή παραμένει επίσης η System RAM, μαζί με όλα εκείνα τα στοιχεία που μπορούν να επαναφέρουν το σύστημα σε active mode, όπως το Power Management Unit (PMU) και ο Quadrature Decoder. Είναι προφανές ότι η κατανάλωση ισχύος είναι αρκετά χαμηλή. Αν το σύστημα εξέλθει του extended sleep mode, η μεταφορά του προγράμματος δεν είναι απαραίτητη από την OTP στην RAM, αφού η τελευταία ήταν ενεργή.
- 3. Deep Sleep Mode:** Η βασική διαφορά με το προηγούμενο sleep mode είναι ότι σε αυτό η RAM του συστήματος απενεργοποιείται. Είναι σαφές ότι μια BLE σύνδεση δεν μπορεί να διατηρηθεί, ενώ σε περίπτωση που ο χρήστης επιθυμεί να επιστρέψει το σύστημα σε active mode, η μεταφορά του προγράμματος από την OTP μνήμη στη RAM είναι απαραίτητη.

Για την εκπόνηση της συγκεκριμένης διπλωματικής εργασίας ήταν απαραίτητη η συγγραφή προγραμματιστικού κώδικα στο αναπτυξιακό DA14585. Παρόλο που η android συσκευή χρειάζεται ελάχιστες πληροφορίες από το κάθε development board, που θα είναι εγκατεστημένο στο αντίστοιχο αρχαιολογικό ή μουσειακό έκθεμα, χρειάστηκε η προσαρμογή και κατάλληλες μετατροπές και προσθήκες στο παρεχόμενο SDK για το αναπτυξιακό από την εταιρία Dialog Semiconductor. Η συγγραφή έγινε σε γλώσσα προγραμματισμού C και πραγματοποιήθηκε στο προγραμματιστικό περιβάλλον MDK-ARM, το οποίο έχει αναπτυχθεί από την Keil. Στο περιβάλλον MDK-ARM περιλαμβάνεται το μVision, ένα ολοκληρωμένο περιβάλλον ανάπτυξης - IDE, το C/C++ Compilation Toolchain της ARM, καθώς ακόμη και ένα σύνολο εργαλείων που σχετίζονται με τον προγραμματισμό, την φόρτωση του κώδικα στο chip, τον έλεγχο και το debugging των εφαρμογών. Το μVision μπορεί να λειτουργήσει και ως user interface για τα υπόλοιπα εργαλεία, προσφέροντας επιπλέον λειτουργίας όπως visual debugging και simulation environment.

Ένα άλλο εργαλείο είναι το SmartSnippets, το οποίο παρέχει λειτουργίες για πιο εύκολο χειρισμό του development board και έλεγχο στις εφαρμογές που εκτελούνται σε αυτήν. Η σημαντικότερη λειτουργία του είναι η εγγραφή στην OTP μνήμη του αναπτυξιακού και ο υπολογισμός της καταναλισκόμενης ισχύος του συστήματος, όταν αυτό βρίσκεται είτε σε active mode είτε σε κάποιο από τα sleep modes.

Στο σημείο αυτό, καθίσταται απαραίτητη η αναφορά του γεγονότος ότι η αρχική ιδέα για την υλοποίηση του συγκεκριμένου εγχειρήματος ήταν ότι όλα τα δεδομένα που θα εμφανίζονται τελικώς στον τουρίστα θα μεταφέρονται από την αναπτυξιακή πλακέτα, μέσω του πρωτοκόλλου Bluetooth, στην android συσκευή. Ωστόσο, η μεταφορά των δεδομένων ήταν εξαιρετικά χρονοβόρα. Έγινε προσπάθεια υλοποίησης αυτής της μεθόδου με δύο διαφορετικούς τρόπους.

Ο πρώτος τρόπος δεν είναι άλλος από το pairing των συσκευών. Σε αυτόν, οι δυο συσκευές πραγματοποιούν σύνδεση, ανταλλάσσουν όλα τα απαραίτητα δεδομένα, και στη συνέχεια αποσυνδέονται. Παρόλο που ο ρυθμός μετάδοσης της πληροφορίας είναι αρκετά υψηλός συγκριτικά με τις απαιτήσεις της παρούσας εφαρμογής, η διαδικασία της σύνδεσης διαρκούσε αρκετά δευτερόλεπτα, καθιστώντας το αναπτυξιακό αόρατο σε όλους τους υπόλοιπους τουρίστες. Υπολογίστηκε πως ο μέσος χρόνος σύνδεσης, μεταφοράς της επιθυμητής πληροφορίας και αποσύνδεσης ήταν περίπου 4-5 δευτερόλεπτα. Είναι σαφές πως με τον τρόπο αυτό, σε μια ομάδα τουριστών, ακόμη και λίγων ατόμων, θα χρειαζόταν πάνω από ένα με δύο λεπτά για να λάβουν όλοι τις πληροφορίες για το έκθεμα. Αυτό θα ήταν μη πρακτικό και ουσιαστικά δυσάρεστο στον τουρίστα.

Ο δεύτερος τρόπος ήταν τα δεδομένα να γίνονται advertised συνεχώς, έτσι ώστε οι συσκευές android των τουριστών να μην χρειάζονται να συνδέονται. Δυστυχώς, ο τρόπος αυτός είχε όρια στον όγκο της πληροφορίας που γίνεται advertise. Πιο συγκεκριμένα, το μέγιστο μέγεθος πακέτου που μπορεί να σταλεί είναι 251 byte. Σε μια εφαρμογή στην οποία οι απαιτήσεις του παρεχόμενου οπτικοακουστικού υλικού είναι υψηλές, θα έπρεπε να αυξήσουμε την κατανάλωση ισχύος, λόγω μικρότερου packet interval.

Μην έχοντας άλλη επιλογή, καταλήξαμε στο τρόπο που χρησιμοποιήσαμε εν τέλει και στην παρούσα διπλωματική εργασία, κατά τον οποίο η συσκευή android κάνει scanning στον περιβάλλοντα χώρο, εντοπίζει τις οικείες αναπτυξιακές πλακέτες που εκπέμπουν, και ανάλογα με μια τιμή UUID, φορτώνει από μια βάση δεδομένων όλες τις απαραίτητες πληροφορίες. Η τιμή UUID, όπως και το όνομα του εκθέματος, ορίζονται από τον προγραμματιστή στον κώδικα που φορτώνεται στο embedded chip.

```

155 | while(1)
156 | {
157 |     do {
158 |         // schedule all pending events
159 |         schedule_while_ble_on();
160 |     }
161 |     while (app_asynch_proc() != GOTO_SLEEP); //grant control to the application, try to go to power down
162 |                                                //if the application returns GOTO_SLEEP
163 |
164 |     //wait for interrupt and go to sleep if this is allowed
165 |     if (((!BLE_APP_PRESENT) && (check_gtl_state())) || (BLE_APP_PRESENT))
166 |     {
167 |         //Disable the interrupts
168 |         GLOBAL_INT_STOP();
169 |
170 |         app_asynch_sleep_proc();
171 |
172 |         // get the allowed sleep mode
173 |         // time from rwip_power_down() to WFI() must be kept as short as possible!!
174 |         sleep_mode = rwip_power_down();
175 |
176 |         if ((sleep_mode == mode_ext_sleep) || (sleep_mode == mode_ext_sleep_otp_copy))
177 |         {
178 |             //power down the radio and whatever is allowed
179 |             arch_goto_sleep(sleep_mode);
180 |
181 |             // In extended sleep mode the watchdog timer is disabled
182 |             // (power domain PD_SYS is automatically OFF). Although, if the debugger
183 |             // is attached the watchdog timer remains enabled and must be explicitly
184 |             // disabled.
185 |             if ((GetWord16(SYS_STAT_REG) & DBG_IS_UP) == DBG_IS_UP)
186 |             {

```

Σχήμα 14: Συνάρτηση main_func

Έτσι λοιπόν, μια από τις λίγες συναρτήσεις που δέχθηκαν μετατροπές ήταν η main_func. Σε αυτήν, αρχικοποιούνται όλα τα στοιχεία του συστήματος, κάνοντας διαδοχικές κλήσεις συναρτήσεων για να αρχικοποιηθούν διάφορες διαδικασίες του συστήματος. Ύστερα από την αρχικοποίηση του συστήματος, το πρόγραμμα περνά στο main loop του προγράμματος και

μένει εκεί μέχρι να το σύστημα να επανεκκινηθεί. Στην παρούσα διπλωματική εργασία, αφαιρέσαμε όλα τα στοιχεία που δεν χρειάζονταν, έτσι ώστε να μειωθεί η κατανάλωση ισχύος, καθώς και ο όγκος του προγράμματος. Συνεπώς, η μορφή του main βρόχου είναι η παρακάτω:

```

187         wdg_freeze();    // Stop watchdog timer
188     }
189
190     //wait for an interrupt to resume operation
191     WFI();
192
193     //resume operation
194     arch_resume_from_sleep();
195 }
196 else if (sleep_mode == mode_idle)
197 {
198     if (((!BLE_APP_PRESENT) && check_gtl_state()) || (BLE_APP_PRESENT))
199     {
200         //wait for an interrupt to resume operation
201         WFI();
202     }
203 }
204 // restore interrupts
205 GLOBAL_INT_START();
206 }
207 wdg_reload(WATCHDOG_DEFAULT_PERIOD);
208

```

Σχήμα 15: Βρόχος αρχικοποίησης των sleep modes

Στον βρόχο του Σχήματος 15, αρχικοποιούνται οι συναρτήσεις κλήσης των sleep modes, ενεργοποιούνται και απενεργοποιούνται οι διακοπές, με τις οποίες γίνονται οι εκτελέσεις των διαφόρων sleep modes και ανάλογα με αυτά, πραγματοποιείται ή δεν πραγματοποιείται αποστολή και λήψη μηνυμάτων. Ο τρόπος λειτουργίας του συστήματος, ανάλογα με το αν γίνεται ή όχι αποστολή μηνυμάτων, μπορεί να διαχωριστεί σε σύγχρονο και ασύγχρονο.

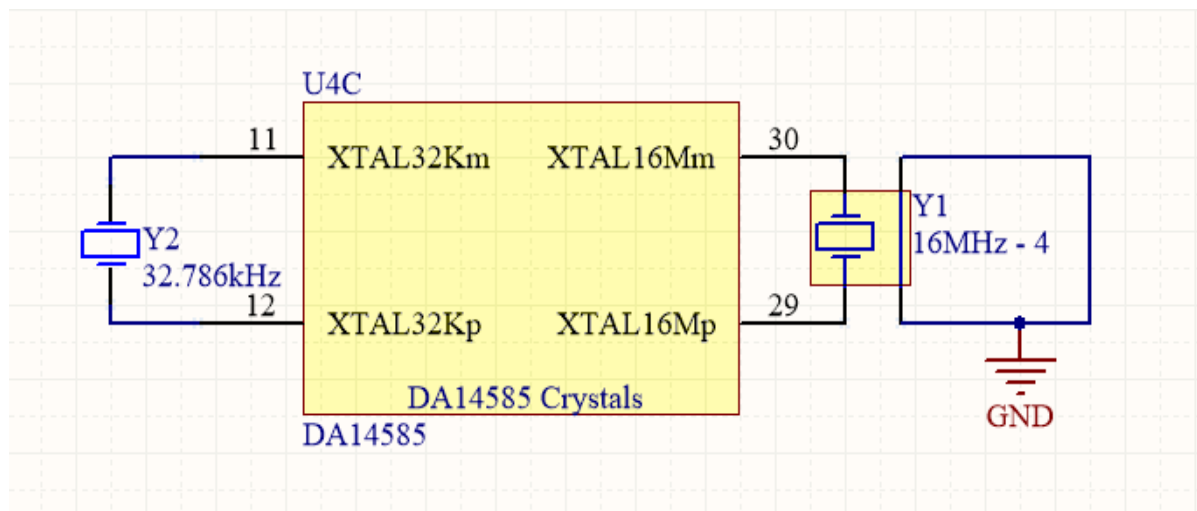
Το προφίλ ανήκει στα σημαντικότερα στοιχεία μια BLE εφαρμογής. Σε αυτό, πραγματοποιείται ορισμός των περιπτώσεων χρήσης, συμπεριφορές που αναμένονται από τις συνδεδεμένες συσκευές, τις υπηρεσίες που πρέπει να υποστηρίζονται και τα χαρακτηριστικά που αυτές που διαθέτουν. Μια εφαρμογή BLE μπορεί να αποτελείται από ένα ή περισσότερα προφίλ, τα οποία μπορεί να είναι ορισμένα είτε από το Bluetooth SIG, είτε customized, δηλαδή ορισμένα από τον developer της εκάστοτε εφαρμογής. Στην περίπτωσή μας, έγινε χρήση ενός έτοιμου προφίλ, ορισμένο από το Bluetooth SIG, το οποίο μας παρείχε το SDK της Dialog Semiconductor. Τα UUIDs, όμως, που χρησιμοποιήσαμε έπρεπε να αλλάξουν ώστε να είναι μοναδικά για την κάθε συσκευή μεν, ομαδοποιημένα όλα μαζί δε. Έτσι, ορίσαμε ένα 128-bit UUID, με βάση το οποίο ορίστηκαν και όλα τα υπόλοιπα, προσθέτοντας δηλαδή για το κάθε ένα, ένα 16-bit αναγνωριστικό. Οι τιμές που χρησιμοποιήθηκαν ήταν ουσιαστικά για λόγους ευκολίας διάκρισης μεταξύ τους. Το βασικό UUID είναι το εξής:

{2E2A93A6-BDD8-4152-AC0B-10992EC6FEED}

Το αναγνωριστικό κάθε καινούριου UUID, προστίθεται στο υπογραμμισμένο μέρος. Κάθε νέο παραγόμενο UUID, προσθέτει τον αριθμό 1 στην 16αδική τιμή του προηγούμενου UUID.

Όσον αφορά στο κατασκευαστικό κομμάτι, το αναπτυξιακό DA14585 μπορεί να λειτουργήσει είτε σε boost είτε σε buck mode. Στην παρούσα διπλωματική εργασία, χρησιμοποιήθηκε το buck mode, με τάση εισόδου περίπου 3V και μέση θερμοκρασία περιβάλλοντος χώρου 25 °C. Το σχηματικό, για λόγους ευκολίας, γίνεται να διαχωριστεί σε τρία διαφορετικά τμήματα, ανάλογα με τις διαδικασίες που εξυπηρετεί το καθένα. Αρχικά, το αναπτυξιακό πλακέτα DA14585 περιέχει έναν clock generator, ο οποίος είναι υπεύθυνος

για τον χρονισμό του συστήματος. Το πρώτο μέρος, λοιπόν, του σχηματικού του DA14585, αποτελεί τον clock generator, ο οποίος φαίνεται στο Σχήμα 16:



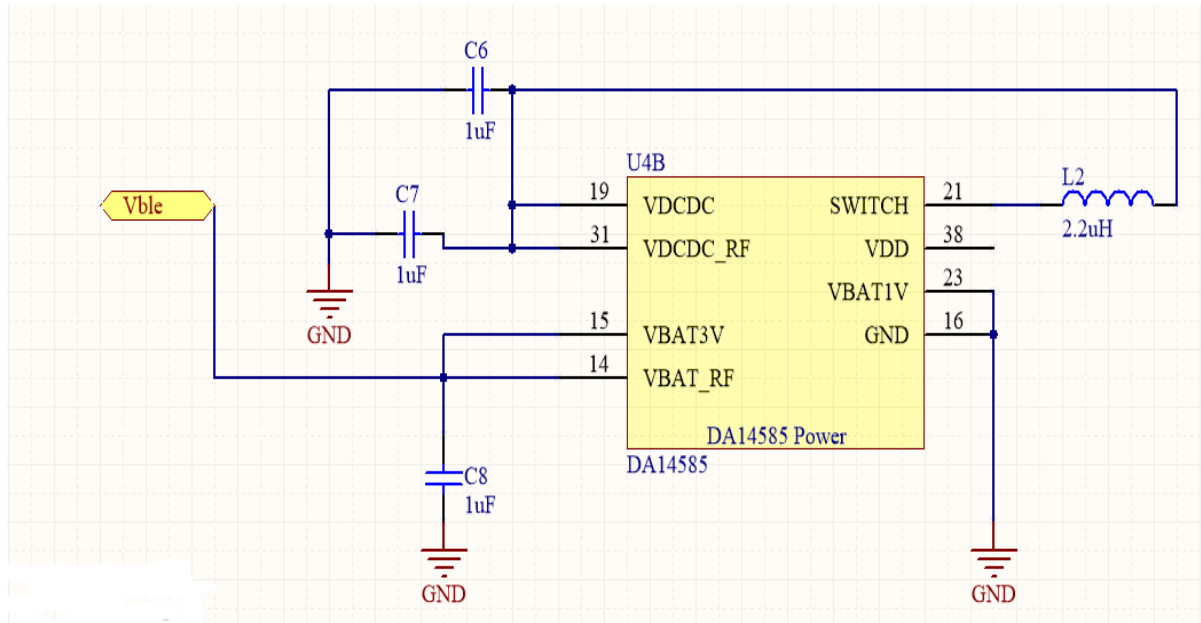
Σχήμα 16: DA14585 Clock Generator

Ο clock generator αποτελείται από 2 XTAL (κρυσταλλικός ταλαντωτής, που με τη βοήθεια ενός δονούμενου κρυσταλλικού πιεζοηλεκτρικού υλικού δημιουργεί ηλεκτρικά σήματα σε μια ακριβή συχνότητα):

1. **έναν 16 MHz:** Χρησιμοποιείται στην περίπτωση που το σύστημα δεν βρίσκεται σε κάποιο sleep mode.
2. **έναν 32 kHz:** Χρησιμοποιείται στην περίπτωση που το σύστημα βρίσκεται σε κάποιο από τα τρία sleep modes.

Το δεύτερο τμήμα αποτελείται από τα pin εισόδου και εξόδου που σχετίζονται με την τροφοδοσία. Πιο συγκεκριμένα, τα pins VDCDC_RF και VBAT_RF συνδέονται με τα V_{DCDC} και V_{BAT} αντίστοιχα, και σχετίζονται, όταν αυτό χρειάζεται, με την ενίσχυση χαμηλής ισχύος σήματος ραδιοσυχνοτήτων σε ένα υψηλής ισχύος σήμα. Το pin VDD χρησιμοποιείται μόνο για δοκιμές και debugging, ενώ σε μια τυπική περίπτωση λειτουργίας του DA14585, είναι ανοικτοκυκλωμένο. Το GND αντιπροσωπεύει τη γείωση, ενώ το VBAT1V στην περίπτωση του buck mode συνδέεται και αυτό στη γείωση. Αντιθέτως, αν είχε γίνει χρήση του boost mode, το VBAT1V θα αποτελούσε το pin που συνδέεται με την τάση εισόδου.

Το pin VBAT3V αποτελεί την είσοδο που τροφοδοτεί την συγκεκριμένη αναπτυξιακή πλακέτα. Σε αυτό είναι επίσης συνδεδεμένος ένας bypass πυκνωτής, ο οποίος αποσκοπεί κυρίως στην ελάττωση του θορύβου.

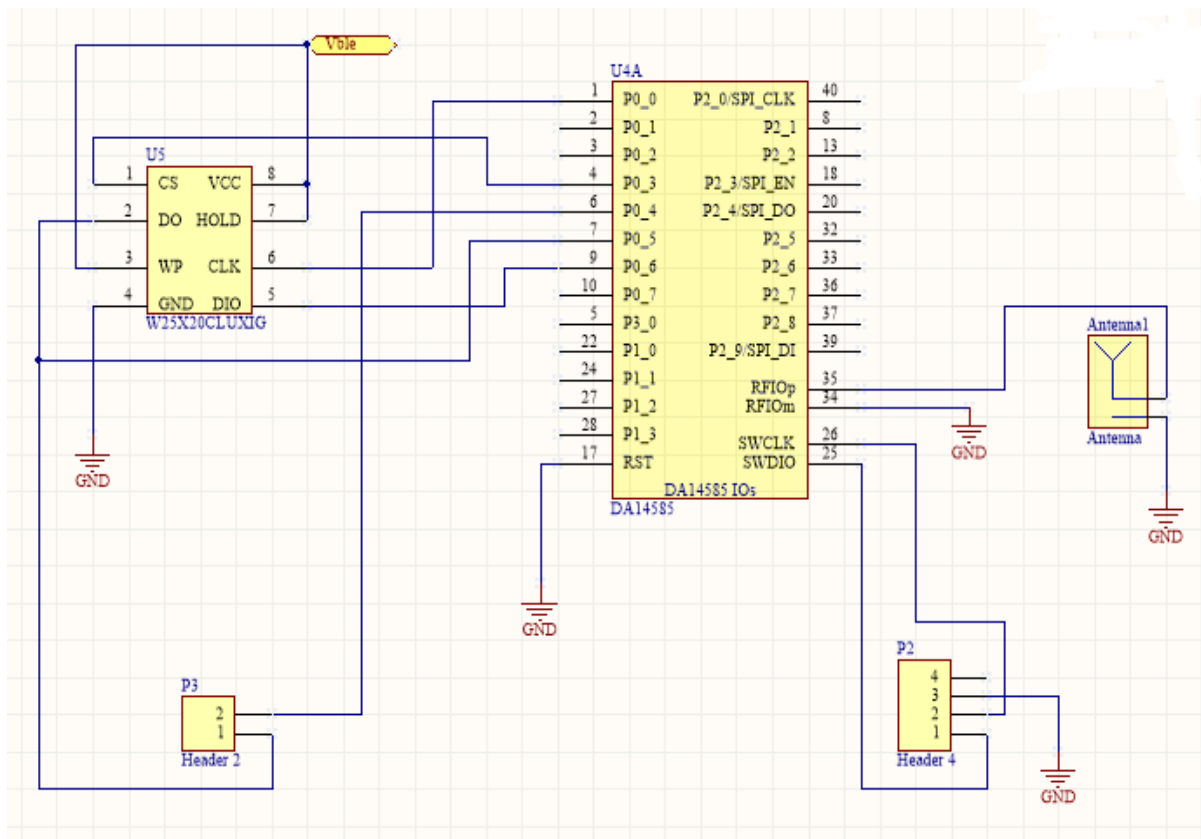


Σχήμα 17: DA14585 Power Supply

Όπως προαναφέρθηκε ο μικροελεγκτής περιλαμβάνει έναν εσωτερικό DC-DC converter, ο οποίος ρυθμίζεται είτε σε buck είτε σε boost mode, ανάλογα με τις ανάγκες του προγραμματιστή. Ο μικροελεγκτής ελέγχει ποιο από τα VBAT3V ή VBAT1V τροφοδοτείται και κάνει τις απαραίτητες ρυθμίσεις. Το pin VDCDC αποτελεί την έξοδο αυτού του DC-DC converter. Μάλιστα, συνδέεται και αυτό με έναν bypass πυκνωτή, έτσι ώστε να περιορίζεται ο θόρυβος του κυκλώματος. Όσον αφορά στη διεπαφή SWITCH, αυτή συνδέεται με ένα πηνίο, το οποίο είναι απαραίτητο για την λειτουργία του buck boost converter. Η λειτουργία του περιλαμβάνει την περιοδική παροχή ή μη ισχύος τάσης στο κύκλωμα. Ο ρόλος του πηνίου είναι η τροφοδοσία του κυκλώματος όταν η πηγή αποσυνδέεται με το φορτίο.

Το τρίτο, και τελευταίο τμήμα, αποτελείται από όλες τις διεπαφές εισόδου/εξόδου που ο ρόλος τους είναι η διασύνδεση του μικροελεγκτή με διάφορα περιφερειακά στοιχεία. Αυτά είναι η κεραία, μια εξωτερική flash memory και ένας adapter, που μέσω αυτού γίνεται ο προγραμματισμός της OTP μνήμης. Έτσι, στο σχηματικό που παρουσιάζεται στην επόμενη σελίδα, οι διεπαφές RFIOp και RFIOm συνδέονται με την κεραία που χρησιμοποιήσαμε και με την γείωση αντίστοιχα. Η διεπαφή RFIOp περιλαμβάνει και μια ηλεκτρική εμπέδηση περίπου ίση με 50 Ω. Οι διεπαφές SWCLK και SWDIO μπορούν να χρησιμοποιηθούν ως είσοδοι/εξόδοι γενικού σκοπού, αλλά στην συγκεκριμένη εφαρμογή λειτουργούν ως είσοδοι για τον προγραμματισμό, μέσω JTAG, του μικροελεγκτή DA14585.

Η flash memory που χρησιμοποιήθηκε ανήκει στην οικογένεια W25X σειριακών flash memories. Έχει χωρητικότητα 256 kB, τα οποία είναι αρκετά για τους σκοπούς αυτής της εφαρμογής. Έχει αρκετά μεγάλη ταχύτητα μετάδοσης πληροφορίας, περί τα 208 Mbps, ενώ η κατανάλωση ισχύος της είναι αρκετά χαμηλή. Στο σημείο αυτό, καθίσταται αναγκαίο να σημειωθεί ότι προσθέσαμε στο σχηματικό και έναν adapter για να είναι δυνατός ο προγραμματισμός μέσω UART. Το σχηματικό της flash memory καθώς και του τρίτου τμήματος του μικροελεγκτή εμφανίζονται στην επόμενη σελίδα:



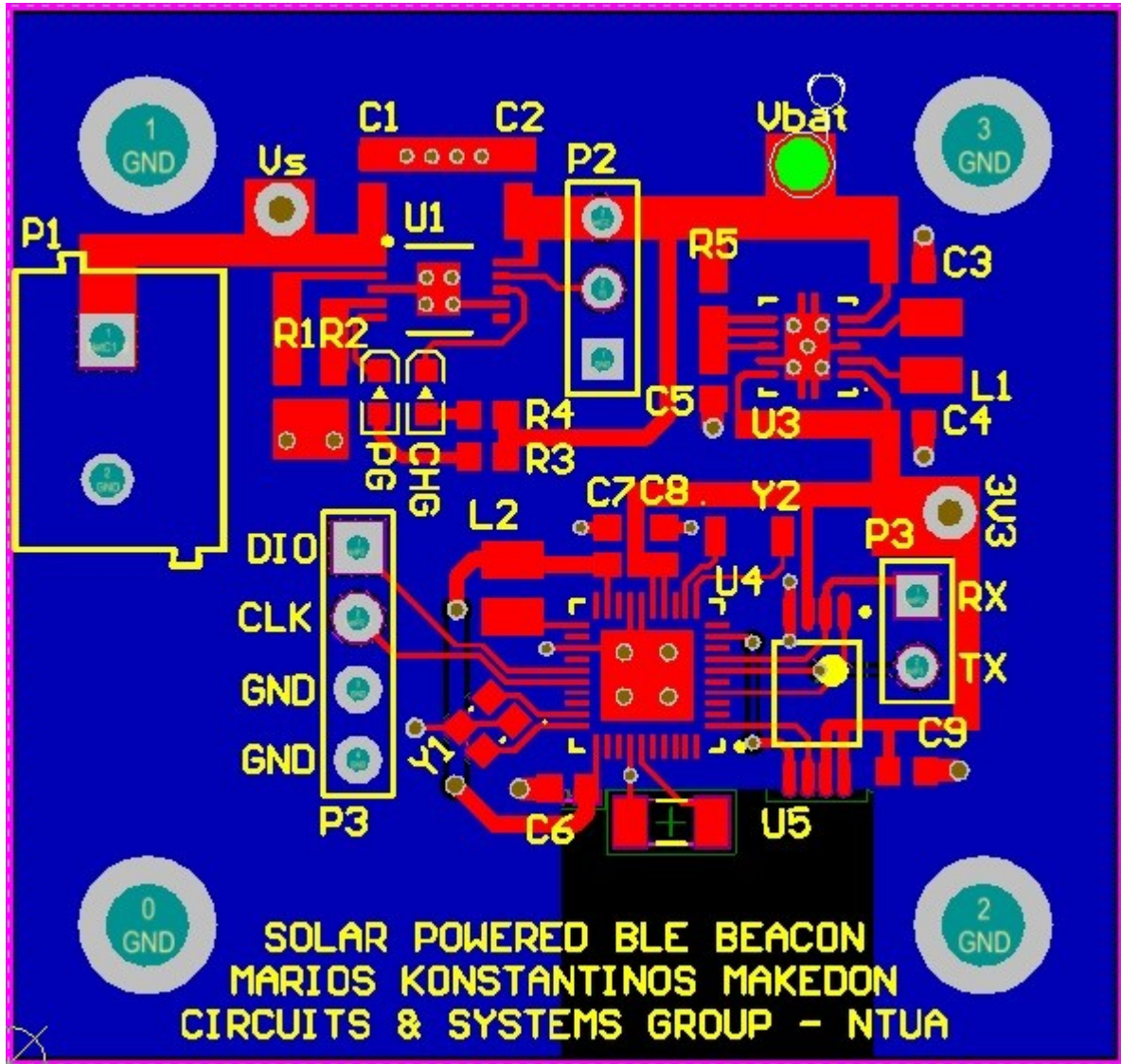
Σχήμα 18: DA14585 IOs

Αξίζει να σημειωθεί ότι η κεραία, που αρχικά, χρησιμοποιήθηκε ήταν τυπωμένη εσωτερικά στο κύκλωμα. Ωστόσο, ύστερα από μια σειρά προσπαθειών βελτιστοποίησης, έγινε αντιληπτό ότι η χρήση μια εξωτερικής, και ανεξάρτητης από το κύκλωμα, κεραίας θα την έκανε πιο αποδοτική. Αυτό οφείλεται στην πολυπλοκότητα στη σχεδίαση της τυπωμένης κεραίας, καθώς και στην αδυναμία, λόγω οικονομικών παραγόντων, υλοποίησης αρκετών δοκιμών. Παρατηρήθηκε πως σε μια απόσταση 10 μέτρων, με την τυπωμένη κεραία, το κινητό που χρησιμοποιήθηκε για τα πειράματα και για δοκιμή της εφαρμογής λάμβανε δύναμη ισχύος περίπου -70 dB. Αντιθέτως, με την νέα εξωτερική κεραία, στην ίδια απόσταση, η λαμβανόμενη ισχύς ήταν περίπου -50 dB. Όλες οι μετρήσεις έγιναν στον ίδιο χώρο, στον οποίο υπήρχαν ελάχιστα φυσικά εμπόδια, αποφεύγοντας έτσι φαινόμενα όπως η περίθλαση.

Η πλακέτα που σχεδιάστηκε και κατασκευάστηκε για την διπλωματική εργασία είναι μια πλακέτα τυπωμένου κυκλώματος, η οποία χρησιμοποιείται για την υποστήριξη και τη διασύνδεση ηλεκτρονικών στοιχείων μέσω αγωγίμων δρόμων, οι οποίοι τυπώνονται πάνω στην πλακέτα. Η συγκεκριμένη πλακέτα αποτελείται από δύο επίπεδα. Η πλειονότητα των συνδεσμολογιών γίνεται στο πάνω επίπεδο, ενώ σε ελάχιστες περιπτώσεις γίνεται και στο κάτω, αφού το τελευταίο λειτουργεί κυρίως ως γείωση. Ο σχεδιασμός της πλακέτας έγινε με το εξειδικευμένο λογισμικό Altium Designer 16.0.

Η συγκεκριμένη πλακέτα είναι αρκετά μικρή (4cm x 4 cm) και θα μπορούσε να σχεδιαστεί ακόμη μικρότερη. Για λόγους, όμως, οικονομίας διατηρήσαμε αυτές τις διαστάσεις, οι οποίες ταυτόχρονα προσδίδουν στην πλακέτα, την ιδιότητα να μπορεί να τοποθετηθεί εύκολα σε πολλά σημεία, ανεξάρτητα από το σχήμα και το μέγεθος που έχει το έκθεμα του αρχαιολογικού χώρου. Την πλακέτα που υλοποιήσαμε, καθώς και την μπαταρία

που την τροφοδοτεί, τα τοποθετήσαμε μέσα σε μια πλαστική θήκη για λόγους προστασίας και παρουσίας. Παρόλο που το κόστος του πρωτότυπου ανέρχεται στα 60 με 70 ευρώ περίπου, σε μια μαζική παραγωγή το κόστος θα μπορούσε να μειωθεί έως και τα 40 ευρώ. Παρακάτω απεικονίζεται το PCB της πλακέτας που σχεδιάσαμε, καθώς και το τελικό αποτέλεσμα, ύστερα δηλαδή από την προσθήκη του πλαστικού περιβλήματος.



Σχήμα 19: PCB Design

Το τελικό αποτέλεσμα, ύστερα δηλαδή από την σύνδεση με μπαταρία και την προσθήκη πλαστικού περιβλήματος παρουσιάζεται στο Παράρτημα Α.

Κεφάλαιο 4

Ανάπτυξη Εφαρμογής σε Android

4.1 Εισαγωγικά Στοιχεία για το λειτουργικό Android

Για την ανάπτυξη και την σχεδίαση του διαδραστικού συστήματος για περιήγηση σε αρχαιολογικούς χώρους και μουσεία, χρησιμοποιήθηκε το λειτουργικό σύστημα Android. Η συγκεκριμένη εφαρμογή μπορεί να εκτελεστεί τόσο σε smartphones όσο και σε tablets που υποστηρίζουν το συγκεκριμένο λειτουργικό σύστημα.

Το android είναι ένα λειτουργικό σύστημα, το οποίο βασίζεται στον πυρήνα του Linux, το οποίο απευθυνόταν αρχικά σε κινητές συσκευές. Υποστηρίζεται από την εταιρία Google και αποτελεί το πιο διαδεδομένο λειτουργικό σύστημα για smartphones και tablets. Το λειτουργικό αυτό βρίσκει εφαρμογές και σε άλλα είδη συσκευών, όπως τηλεοράσεις (smart TV), κονσόλες παιχνιδιών, ρολόγια (smart watches) και αυτοκίνητα.

Το σύστημα Android αναπτύσσεται ιδιωτικά από την εταιρία Google μέχρι οι τελευταίες αλλαγές, ενημερώσεις και δυνατότητες να είναι έτοιμες προς κυκλοφορία. Όταν συμβεί αυτό, ο κώδικας του λειτουργικού συστήματος γίνεται διαθέσιμος προς το ευρύ κοινό. Αντιθέτως, ο πυρήνας του Linux είναι διαθέσιμος προς τους developers εξ αρχής. Η πιο πρόσφατη έκδοση του λειτουργικού συστήματος Android είναι η έκδοση 9.0, η οποία έχει κωδική ονομασία Pie. Στον παρακάτω πίνακα φαίνονται οι σημαντικότερες εκδόσεις που έχουν κυκλοφορήσει έως σήμερα μαζί με τα API level τους. Το επίπεδο API υποδηλώνει το σύνολο των δυνατοτήτων που παρέχονται στους developers, που παρέχονται από την κάθε έκδοση.

Πίνακας 2: Εκδόσεις Android & API levels

Κωδικό Όνομα	Νούμερο Έκδοσης	Κυκλοφορία	Επίπεδο API
n/a	1.0	23 Σεπ 2008	1
n/a	1.1	09 Φεβ 2009	2
Cupcake	1.5	27 Απρ 2009	3
Donut	1.6	15 Σεπ 2009	4
Eclair	2.0 - 2.1	26 Οκτ 2009	5 - 7
Froyo	2.2 - 2.2.3	20 Μαΐ 2010	8
Gingerbread	2.3 - 2.3.7	06 Δεκ 2010	9 - 10
Honeycomb	3.0 - 3.2.6	22 Φεβ 2011	11 -13
Ice Cream Sandwich	4.0 - 4.0.4	18 Οκτ 2011	14 - 15
Jelly Bean	4.1 - 4.3.1.	09 Ιουλ 2012	16 - 18
KitKat	4.4 - 4.4.4	31 Οκτ 2013	19 - 20
Lollipop	5.0 - 5.1.1	12 Νοε 2014	21 - 22
Marshmallow	6.0 - 6.0.1	05 Οκτ 2015	23

Nougat	7.0 - 7.1.2	22 Αυγ 2016	24 - 25
Oreo	8.0 - 8.1	21 Αυγ 2017	26 - 27
Pie	9.0	06 Αυγ 2018	28

Στο σημείο αυτό, χρειάζεται να σημειωθεί πως η υποστήριξη του Bluetooth Low Energy ξεκίνησε με την έκδοση 4.3 και το επίπεδο API 18.

Οι εφαρμογές Android αποτελούνται από τέσσερις τύπους components:

1. Activities
2. Services
3. Content Providers
4. Broadcast Receivers

Το activity είναι μια από τις οθόνες της εφαρμογής που εμφανίζονται στον χρήστη. Συνήθως, παρουσιάζεται σε αυτόν ένα user interface (UI). Πρόκειται για το τμήμα της εφαρμογής που έρχεται σε άμεση επαφή ο χρήστης, πάνω στο οποίο μπορεί να εκτελεί διάφορες λειτουργίες και να βλέπει τα αποτελέσματά τους. Η δημιουργία ενός activity προκύπτει από τη δημιουργία ενός αρχείου τύπου XML, το οποίο μπορεί είτε να οριστεί γραφικά είτε να προγραμματιστεί. Κάθε ένας από τους δύο τρόπους έχει τα πλεονεκτήματα και μειονεκτήματα του και αυτός που χρησιμοποιείται εν τέλει από τον developer, εξαρτάται από την προσωπική του προτίμηση.

Ένα activity μπορεί να βρίσκεται σε μια εκ των καταστάσεων:

- **resumed** (ή **running**) , στην οποία το activity λειτουργεί στο προσκήνιο,
- **paused**, στην οποία κάποιο άλλο activity βρίσκεται στο προσκήνιο, αφήνοντας αυτό να τρέχει στο παρασκήνιο
- **stopped**, στην οποία το activity βρίσκεται στο παρασκήνιο και είναι αόρατο από τον χρήστη

Κατά τη διάρκεια λειτουργίας ενός activity, το σύστημα καλεί ένα σύνολο callback μεθόδων. Οι κυριότερες από αυτές είναι:

Πίνακας 3: Λίστα callback μεθόδων

onCreate()	Καλείται την πρώτη φορά που πρέπει να εκτελεστεί το activity
onRestart()	Καλείται όταν το activity έχει γίνει stopped
onStart()	Καλείται όταν το activity πρέπει να γίνει ορατό προς τον χρήστη
onResume()	Καλείται πριν ο χρήστης αρχίσει να αλληλεπιδρά με αυτό
OnPause()	Καλείται όταν κάποιο άλλο activity πρέπει να έρθει στο προσκήνιο

onStop()	Καλείται όταν το activity είναι αόρατο στο χρήστη
onDestroy()	Καλείται για να καταστραφεί το activity

Σε περίπτωση που πραγματοποιηθεί κλήση των onPause, onStop ή onDestroy, το activity θεωρείται πλέον killable, και μπορεί ανά πάσα στιγμή να τερματισθεί, ελεγχόμενα, από το σύστημα.

Το service είναι ένα στοιχείο της εφαρμογής, το οποίο χρησιμοποιείται για την εκτέλεση διαφόρων λειτουργιών, που απαιτούν μεγάλο χρονικό διάστημα εκτέλεσης. Το service δεν παρέχει στο χρήστη user interface UI. Ένα service μπορεί να εμφανιστεί σε δύο μορφές:

- **started:** ένα component μιας συσκευής μπορεί να ξεκινήσει το service, κάνοντας κλήση της μεθόδου startService(). Έκτοτε, το service μπορεί να εκτελείται για απεριόριστο χρονικό διάστημα, ακόμα και αν το αρχικό component καταστραφεί. Για να σταματήσει το service, χρειάζεται η παρέμβαση κάποιου άλλου service ή αυτό να σταματήσει τον εαυτό του.
- **bound:** ένα component μπορεί να συνδεθεί (bind) με ένα service, καλώντας τη μέθοδο bindService(). Ένα bound service παρέχει ένα interface, με το οποίο τα components που συνδέονται σε αυτό, μπορούν να αλληλεπιδράσουν μαζί του. Το bound service τερματίζεται όταν και το τελευταίο component αποσυνδέεται από αυτό.

Κατά τη διάρκεια εκτέλεσης ενός service, το σύστημα καλεί μια σειρά callback μεθόδων, οι οποίες είναι οι εξής:

onCreate()	Καλείται κατά τη δημιουργία του service
onDestroy()	Καλείται όταν το service πρόκειται να καταστραφεί από το σύστημα
onStartCommand()	Καλείται όταν ένα component εκκινεί το service
OnBind()	Καλείται όταν ένα component συνδέεται με το service
OnUnbind()	Καλείται όταν ένα συνδεδεμένο component αποσυνδέεται

Ένας content provider είναι ένα στοιχείο μιας εφαρμογής, ο οποίος αποσκοπεί στη διαχείριση της πρόσβασης σε ένα σύνολο δεδομένων. Αυτός ορίζει ένα authority και μερικά URIs (Uniform Resource Identifiers), έτσι ώστε τα components να αποκτούν πρόσβαση σε δεδομένα που δεσμεύονται από τον provider. Έτσι, μπορεί να γίνει προσθήκη, διαγραφή, μετακίνηση και ανάκτηση δεδομένων. Ένα παράδειγμα content provider αποτελεί ο provider επαφών, ο οποίος, με κατάλληλα δικαιώματα που του δίνει ο χρήστης, επιτρέπει σε διάφορες εφαρμογές να αποκτούν πρόσβαση στις επαφές της κινητής συσκευής.

Ένας broadcast receiver είναι ένα στοιχείο της εφαρμογής, ο οποίος αποσκοπεί στην αποδοχή system-wide broadcast μηνυμάτων, που στέλνονται είτε από το λειτουργικό σύστημα είτε από άλλες εφαρμογές. Τα μηνύματα αυτά συνδέονται με συγκεκριμένες ενέργειες. Παραδείγματος χάριν, η οθόνη έκλεισε, το κινητό μπήκε σε

λειτουργία εξοικονόμησης ενέργειας λόγω χαμηλής στάθμης στην ένδειξη της μπαταρίας. Οι broadcast receivers δεν έχουν κάποιο interface.

4.2 Αρχεία XML και αρχεία Resources

Τα αρχεία με τα οποία έρχεται άμεσα σε επαφή ένας android developer είναι κυρίως κλάσεις, στις οποίες γράφεται ο προγραμματιστικός κώδικας και θα δούμε αναλυτικότερα στη συνέχεια, και τα XML αρχεία. Το android API χρησιμοποιεί ποικιλοτρόπως τα αρχεία αυτά με σκοπό την εξυπηρέτηση διαφόρων αναγκών. Ανάλογα με το επιθυμητό αποτέλεσμα, τα XML αρχεία ορίζουν τη μορφή του user interface UI των activities, widgets, integers, menus, layouts, background, fonts, πολλά είδη από values και γενικότερα όλα τα είδη resources που θα χρησιμοποιηθούν στην εφαρμογή. Σε πολλές περιπτώσεις, τα αντικείμενα που ορίζονται στα XML αρχεία έχουν αντικείμενο ορισμένο σε java κλάση και ελέγχονται από τον developer στις κλάσεις αυτές.

Όπως προαναφέραμε, το UI ορίζεται στα αρχεία τύπου XML. Με τον τρόπο αυτό, ο κώδικας γίνεται πιο γενικός, δηλαδή περισσότερα activities μπορούν να τον χρησιμοποιήσουν, το καθένα με τον τρόπο που αυτό που επιθυμεί. Έτσι, διαφορετικά activities μπορούν να χρησιμοποιήσουν το ίδιο XML αρχείο, αλλάζοντας το χρώμα, το κείμενο, τον τρόπο παρουσίασης μόνο με μερικές γραμμές προγραμματιστικού κώδικα. Εκτός αυτού, μπορούν να οριστούν αρχεία για διαφορετικού μεγέθους συσκευές ή διαφορετικό προσανατολισμού οθόνης (landscape - portrait).

Ένα από τα πιο σημαντικά αρχεία XML σε μια Android εφαρμογή είναι το AndroidManifest.xml. Το αρχείο αυτό περιλαμβάνει την πλειονότητα των κρίσιμων πληροφοριών που έχει ανάγκη το σύστημα, ώστε να αυτό να εκτελέσει μια εφαρμογή. Σε αυτό περιλαμβάνεται το όνομα, η έκδοση της εφαρμογής. Περιέχονται, επίσης, πληροφορίες όπως το εικονίδιο και η ονομασία της συσκευής με τα οποία θα εμφανίζεται η συσκευή στη λίστα με τις εφαρμογές του χρήστη, η έκδοση του λειτουργικού android που μπορεί να υποστηρίξει την συγκεκριμένη εφαρμογή, καθώς και το activity, το οποίο είναι το πρώτο που εμφανίζεται όταν εκκινείται η εφαρμογή. Είναι πιθανό, βεβαίως, να ορίζονται και μερικά ακόμη activities, παρόλο που δεν είναι αυτά, με τα οποία εκκινείται η εφαρμογή. Ο λόγος που μπορεί να συμβεί κάτι τέτοιο, είναι η ιδιαιτερότητα που έχουν ορισμένα activities, καθώς αλληλεπιδρούν με άλλες εφαρμογές ή με πολλούς πόρους του συστήματος.

```

1 <application
    android:icon="@mipmap/valeria"
    android:label="MuseIt"
    android:theme="@android:style/Theme.Holo.Light.DarkActionBar">
2 <activity
    android:name=".DeviceScanActivity"
    android:label="@string/app_name">
3 <intent-filter>
4     <action android:name="android.intent.action.MAIN" />
5
6     <category android:name="android.intent.category.LAUNCHER" />
7 </intent-filter>
8 </activity>
9 <activity android:name=".DeviceControlActivity" />
10 <service
    android:name=".BluetoothLeService"
    android:enabled="true" />
11 <activity
    android:name=".StartingActivity"
    android:label="@string/title_activity_starting" />
12 <activity
    android:name=".SecondaryActivity"
    android:label="@string/title_activity_secondary" />
13 <activity android:name=".FavoriteActivity" />
14 <activity android:name=".SecondFavActivity" />
15 <activity android:name=".ChocoActivity"></activity>
16 </application>
17 </manifest>

```

Σχήμα 20: AndroidManifest.xml

Ένας λόγος που ο ορισμός του API level και των features είναι ιδιαίτερα σημαντικός στο AndroidManifest.xml είναι η παροχή της δυνατότητας σε εφαρμογές τύπου PlayStore να μην εμφανίζουν στη λίστα με τις προτεινόμενες εφαρμογές την συγκεκριμένη εφαρμογή, όταν αυτή δεν πληροί τις απαιτούμενες προϋποθέσεις. Επί προσθέτως, ορίζονται και παρουσιάζονται όλα τα permissions που πρέπει να απαιτήσει η εφαρμογή, έτσι ώστε να λειτουργήσει ορθά. Για παράδειγμα, η ανάγνωση επαφών, φωτογραφιών, η ενεργοποίηση του Wi-Fi, του Bluetooth, του GPS. Συνήθως, για την αποφυγή αιτημάτων άδειας χρήσης αυτών των χαρακτηριστικών, η εφαρμογή, κατά την εγκατάστασή της, ζητά από τον χρήστη να της παραχωρήσει πρόσβαση σε αυτούς τους πόρους κάθε φορά που αυτή εκκινείται. Προφανώς, ο χρήστης μπορεί μέσω των ρυθμίσεων της κινητής συσκευής να αναιρέσει αυτή την ενέργεια.


```

<?xml version="1.0" encoding="utf-8"?><!--...-->
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.android.bluetoothlegatt"
    android:versionCode="1"
    android:versionName="1.0">
    <!-- Min/target SDK versions (<uses-sdk>) managed by build.gradle -->
    <!--...-->
    <uses-feature
        android:name="android.hardware.bluetooth_le"
        android:required="true" />
    <!--...-->
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.BLUETOOTH" />
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />

```

Σχήμα 21: Permission Requests

Όπως φαίνεται στο άνωθεν τμήμα του προγραμματιστικού κώδικα, η εφαρμογή ζητά από τον εκάστοτε χρήστη να ενεργοποιήσει το Bluetooth, σε κάθε περίπτωση, όταν ανοίγει την εφαρμογή. Η ενεργοποίηση του Internet, είτε WiFi είτε σύνδεση με data, γίνεται μόνο όταν ο χρήστης επιλέξει το mode της εφαρμογής που χρειάζεται video streaming. Στο σημείο, είναι αναγκαίο να σημειωθεί πως σε κινητές συσκευές που έχουν εγκατεστημένη την έκδοση Android 6.0, ή κάποια νεότερη (ή αλλιώς το επίπεδο API να είναι 21 ή πιο σύγχρονο), χρειάζεται η ενεργοποίηση της χρήσης της υπηρεσίας εύρεσης τοποθεσίας GPS. Για το λόγο αυτό, όταν η εφαρμογή αντιληφθεί ότι η κινητή συσκευή, στην οποία αυτή έχει εγκατασταθεί, έχει λειτουργικό Android 6.0, ή κάποια άλλη πιο σύγχρονη έκδοση, η εφαρμογή θα ζητήσει και την ενεργοποίηση του GPS.

4.3 Λειτουργία της εφαρμογής

Το αντικείμενο της παρούσας εφαρμογής είναι η χρήση της σε αρχαιολογικό χώρο, στον οποίο έχει πραγματοποιηθεί εγκατάσταση των αναπτυξιακών συσκευών που σχεδιάστηκαν και αναλύθηκαν στο Κεφάλαιο 2. Ανοίγοντας την εφαρμογή, ο χρήστης θα έχει τη δυνατότητα, με τη βοήθεια του πρωτοκόλλου Bluetooth, να σαρώνει την περιοχή κοντά του, να εντοπίζει τα εκθέματα που βρίσκονται σε μικρή απόσταση από αυτόν και εν συνεχεία, με ένα απλό κλικ, να του διατίθεται οπτικοακουστικό υλικό, αλλά και γραπτό κείμενο, το οποίο θα τον ενημερώνει για το εκάστοτε έκθεμα. Σκοπός του MuseIt είναι να αποτελέσει βοήθημα στα χέρια του ξεναγού, κάνοντας πιο εύκολη και ταυτόχρονα ευέλικτη τη δουλειά του, καθώς και με μελλοντικές επεκτάσεις της, ακόμα και να τον αντικαταστήσει.

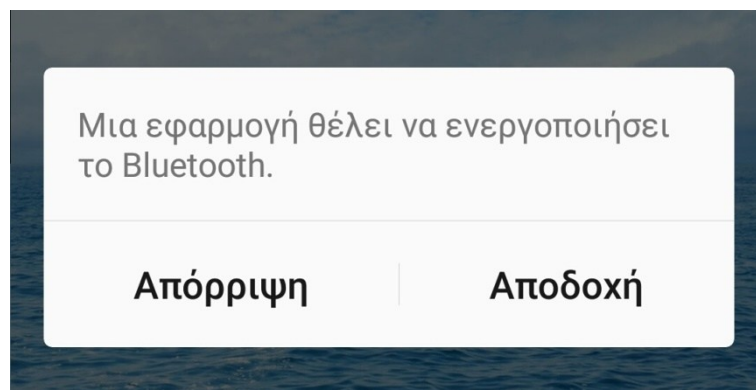
4.3.1 Java Activities

Το τμήμα του κώδικα που πραγματοποιείται η ερώτηση προς το χρήστη σχετικά με την ενεργοποίηση του Bluetooth είναι το αρχικό activity, δηλαδή αυτό που εμφανίζεται στην οθόνη της κινητής συσκευής, όταν αυτή εκκινείται. Για να γίνει αυτό, χρειάζεται η εφαρμογή να ελέγξει αν η συγκεκριμένη κινητή συσκευή υποστηρίζει το πρωτόκολλο Bluetooth. Σε περίπτωση που το υποστηρίζει, αρχικοποιεί και έναν Bluetooth adapter, με τη βοήθεια του οποίου πραγματοποιούνται διάφορες Bluetooth λειτουργίες, όπως scanning, σύνδεση κ.α.

```
if (!getPackageManager().hasSystemFeature(PackageManager.FEATURE_BLUETOOTH_LE)) {  
    Toast.makeText( context: this, "BLE is not supported", Toast.LENGTH_SHORT).show();  
    finish();  
}  
  
//...  
final BluetoothManager bluetoothManager =  
    (BluetoothManager) getSystemService(Context.BLUETOOTH_SERVICE);  
mBluetoothAdapter = bluetoothManager.getAdapter();  
  
// Checks if Bluetooth is supported on the device.  
if (mBluetoothAdapter == null) {  
    Toast.makeText( context: this, "Bluetooth not supported.", Toast.LENGTH_SHORT).show();  
    finish();  
    return;  
}
```

Σχήμα 22: Bluetooth Support Check

Όπως φαίνεται στο Σχήμα 23, αν ο χρήστης δεν έχει ενεργοποιήσει τη λειτουργία Bluetooth, η εφαρμογή θα του ζητήσει να την ενεργοποιήσει. Με τον τρόπο αυτό, η εφαρμογή εξασφαλίζει ότι η λειτουργία του Bluetooth scanning θα εκκινηθεί ομαλά.



Σχήμα 23: Bluetooth Support Demand

Στο activity αυτό, λαμβάνουν χώρα και πολλές ακόμα λειτουργίες μείζονος σημασίας για την λειτουργικότητα της εφαρμογής. Το πρώτο αντικείμενο που κατασκευάστηκε ήταν το πλαίσιο στο πάνω μέρος οθόνης. Σε αυτό, είναι δυνατή η πρόσθεση διαφόρων εικονικών στοιχείων, όπως το εφέ ένδειξης προόδου του scanning process και ο "υπερσύνδεσμος" για να παραπέμψει το χρήστη στις συσκευές

- εκθέματα, που έχει χαρακτηρίσει αγαπημένα και επιθυμεί να τα αποθηκεύσει στην Android συσκευή του για μελλοντική χρήση.

```
getMenuInflater().inflate(R.menu.main, menu);
menu.findItem(R.id.menu_refresh).setActionView(R.layout.actionbar_indeterminate_progress);
menu.findItem(R.id.menu_favorite).setVisible(true); //setActionView(R.layout.action_layout

case R.id.menu_favorite:
    Log.e( tag: "BLE_SCANNER", msg: "menu_favorite");
    final Intent intent = new Intent(getApplicationContext(), FavoriteActivity.class);
    startActivity(intent);
```

Σχήμα 24: Κατασκευή menu

Μία από τις πιο σημαντικές διαδικασίες που πραγματοποιούνται στην εφαρμογή είναι το scanning. Η συχνότητα ανανέωσης των κοντινών, στον χρήστη, συσκευών είναι την παρούσα στιγμή μία φορά ανά δύο δευτερόλεπτα. Ένα πλεονέκτημα της μικρής περιόδου είναι η ελαχιστοποίηση των λαθών που προκύπτουν. Η μέθοδος που χρησιμοποιούμε για να υπολογίσουμε προσεγγιστικά την εγγύτητα είναι ο υπολογισμός της ληφθείσας ισχύος RSSI. Όπως έχουμε ήδη αναφέρει, η συγκεκριμένη μέθοδος μπορεί, σε ορισμένες μορφολογίες του χώρου ή χρονικές στιγμές, να παρουσιάζει λάθη. Πιο αναλυτικά, φαινόμενα σκίασης και περίθλασης, όταν το σήμα εκτρέπεται από την ευθύγραμμη πορεία του, μπορούν να εμφανίζουν μια μακρινή συσκευή (15 m) πιο κοντά από μια πλησιέστερη συσκευή (5 m).

```
private Runnable refreshScan = () -> {
    //cRssi=0;
    Log.e( tag: "BLE_SCANNER", msg: "reSTART_SCAN");
    //...

    mLeDeviceListAdapter.refresh();
    mHandler.postDelayed(refreshScan, delayMillis: 2000);
    //...
};

private Runnable startScan = () -> {
    cRssi=0;
    Log.e( tag: "BLE_SCANNER", msg: "START_SCAN");
    mLeDeviceListAdapter.clear();
    mHandler.postDelayed(refreshScan, delayMillis: 2000);
    mBluetoothAdapter.startLeScan(scanFilter, mLeScanCallback);
    // mBluetoothAdapter.startLeScan(mLeScanCallback);
    mScanning=true;
};

private Runnable stopScan = () -> {
    Log.e( tag: "BLE_SCANNER", msg: "STOP_SCAN");
    mScanning=false;
    mBluetoothAdapter.stopLeScan(mLeScanCallback);
    mHandler.postDelayed(startScan, 500);
};
```

Σχήμα 25: Scanning Methods & Callbacks

Το δεύτερο μπορεί να αντιμετωπισθεί από αυτόν που θα εγκαταστήσει το συγκεκριμένο εγχείρημα στον αρχαιολογικό χώρο, διαλέγοντας προσεκτικά τα σημεία τοποθέτησης των αναπτυξιακών συσκευών. Ωστόσο, το πρώτο είναι αποτέλεσμα, συνήθως, των τουριστών που περνούν μπροστά από τα διάφορα εκθέματα. Αυτό θα μπορούσε να αντιμετωπισθεί με την προσθήκη ενός μετρητή βημάτων ή ενός αισθητήρα κίνησης, με τον οποίο η εφαρμογή θα αντιλαμβάνεται αν ο χρήστης περπάτησε ή συνέβη κάποιο σφάλμα μέτρησης του RSSI. Ένα μειονέκτημα της μικρής περιόδου στην ανανέωση των πλησιέστερων συσκευών είναι η περιστασιακή, όμως συχνή, εναλλαγή στην σειρά των πλησιέστερων συσκευών, εξαιτίας των λόγων που προαναφέρθηκαν. Βεβαίως, μια πολύ μεγάλη χρονική περίοδος, άνω των 5-6 δευτερόλεπτων, θα δημιουργούσε μια μικρή καθυστέρηση στην εναλλαγή της σειράς εγγύτητας των εκθεμάτων. Πιθανώς, η ιδανική χρονική περίοδος είναι περίπου 3-4 δευτερόλεπτα.

Στο Σχήμα 25, φαίνονται οι μέθοδοι `refreshScan()`, `startScan()` και `stopScan()`. Είναι προφανές, λόγω της ονομασίας τους, ότι αναφέρονται στις τρεις καταστάσεις που ενδέχεται να βρίσκεται το scanning. Την πρώτη φορά, ύστερα από την εκκίνηση ή από μια διακοπή του activity, εκτελείται η μέθοδος `startScan()`. Αυτή βεβαιώνεται ότι δεν έχουν ξεμείνει ενεργά διάφορα callbacks και στη συνέχεια, ξεκινά τη διαδικασία του scanning. Όταν, μετά από δύο δευτερόλεπτα, προκύψει το callback, θα συνεχίσει παρόμοια διαδικασία η μέθοδος `refreshScan()`. Η μέθοδος `stopScan()` θα κληθεί μόνο στην περίπτωση που το activity θα περάσει σε `onPause` κατάσταση. Αυτό μπορεί να συμβεί είτε στην αλλαγή ενός activity στο προσκήνιο είτε στην διακοπή λειτουργίας της εφαρμογής.

Σε κάθε περίοδο που πραγματοποιείται η διαδικασία του scanning, οι Bluetooth Low Energy συσκευές που θα βρεθούν να εκπέμπουν πληροφορία, αφού περάσουν από ένα UUID φίλτρο που κατασκευάσαμε προγραμματιστικά εμείς, θα πρέπει να αποθηκεύονται κάπου έτσι ώστε να είναι διαχειρίσιμες. Την εργασία αυτή αναλαμβάνει ο `LeDeviceListAdapter`, ο οποίος είναι ένας `BaseAdapter`, ο οποίος έχει σκοπό την αποθήκευση και την διαχείριση των συσκευών που βρέθηκαν.

```

public void addDevice(BluetoothDevice device, int rssi) {

    //if(rssi>-70) {
        tempScanResults.add(device.hashCode());
        if (mLeDevices.indexOfKey(device.hashCode()) < 0) {
            DeviceData entry = new DeviceData(rssi,
                device.getName(), device.getAddress());
            mLeDevices.put(device.hashCode(), entry);
            // mRssi[cRssi]=rssi;
            // cRssi++;
            mDevicesShown.add(entry);
        }
        else {
            DeviceData entry = mLeDevices.get(device.hashCode());
            entry.setRssi(rssi);
        }

        notifyDataSetChanged();
    }
}

```

Σχήμα 26: Adding a Device

Κάθε φορά που εντοπίζει μια συσκευή ο adapter, ενεργοποιείται η μέθοδος addDevice(), με την οποία θα γίνει η προσθήκη ή όχι της συσκευής στη λίστα των πλησιέστερων συσκευών. Η λίστα mDevicesShown είναι αυτή που θα περιέχει όλες τις συσκευές που θα εμφανιστούν στην οθόνη της κινητής συσκευής. Περιέχει ήδη τις συσκευές που είχαν εντοπιστεί σε προηγούμενο scanning.. Έτσι, αν εμφανιστεί κάποια καινούρια, θα την προσθέσει στη λίστα. Σε κάθε, όμως, περίπτωση θα σημειωθεί η νέα τιμή RSSI που έχει υπολογισθεί, έτσι ώστε να γίνει εκ νέου ταξινόμηση, με βάση τη λαμβάνουσα ισχύ, στις συσκευές.

Αφού, λοιπόν, έχει ολοκληρωθεί η διαδικασία του scanning και η επιλογή των συσκευών, που πρόκειται να προβληθούν στην οθόνη της κινητής συσκευής του χρήστη, πρέπει να οι συσκευές να παρουσιαστούν διατεταγμένες με σειρά απόστασης. Αυτό πραγματοποιείται με μία τυπική μέθοδο getView(), στην οποία εξετάζονται όλες οι πιθανές περιπτώσεις σχετικά με τον αριθμό των στοιχείων της λίστας. Σε αυτό το σημείο, χρειάζεται να τονιστεί ότι πραγματοποιήθηκε παρουσίαση της λειτουργικότητας και της εμπειρίας του χρήστη. Ύστερα από ορισμένες παρατηρήσεις, το συμπέρασμα που εξήχθη ήταν οι περισσότεροι χρήστες θεωρούσαν αυτονόητο ότι το πλησιέστερο έκθεμα προς αυτούς, ήταν αυτό που εμφανιζόταν στο πάνω και αριστερό μέρος της οθόνης τους.

```

public View getView(int i, View view, ViewGroup viewGroup) {
    DeviceData deviceData = getDevice(i);
    final int deviceRssi = deviceData.getRssi();
    ViewHolder viewHolder;
    // General ListView optimization code.
    if (view == null) {
        view = mInflater.inflate(R.layout.listitem_device, root: null);

        viewHolder = new ViewHolder();
        // viewHolder.deviceAddress = (TextView) view.findViewById(R.id.device_address);
        viewHolder.deviceImage = (ImageView) view.findViewById(R.id.device_image);
        viewHolder.deviceName = (TextView) view.findViewById(R.id.device_name);
        viewHolder.deviceDistance = (ImageView) view.findViewById(R.id.device_distance);
        view.setTag(viewHolder);
    } else {
        viewHolder = (ViewHolder) view.getTag();
    }

    final String deviceName =deviceData.getName();
    final int deviceImage =deviceData.getImage();
    final int deviceDistance =deviceData.getRssi();

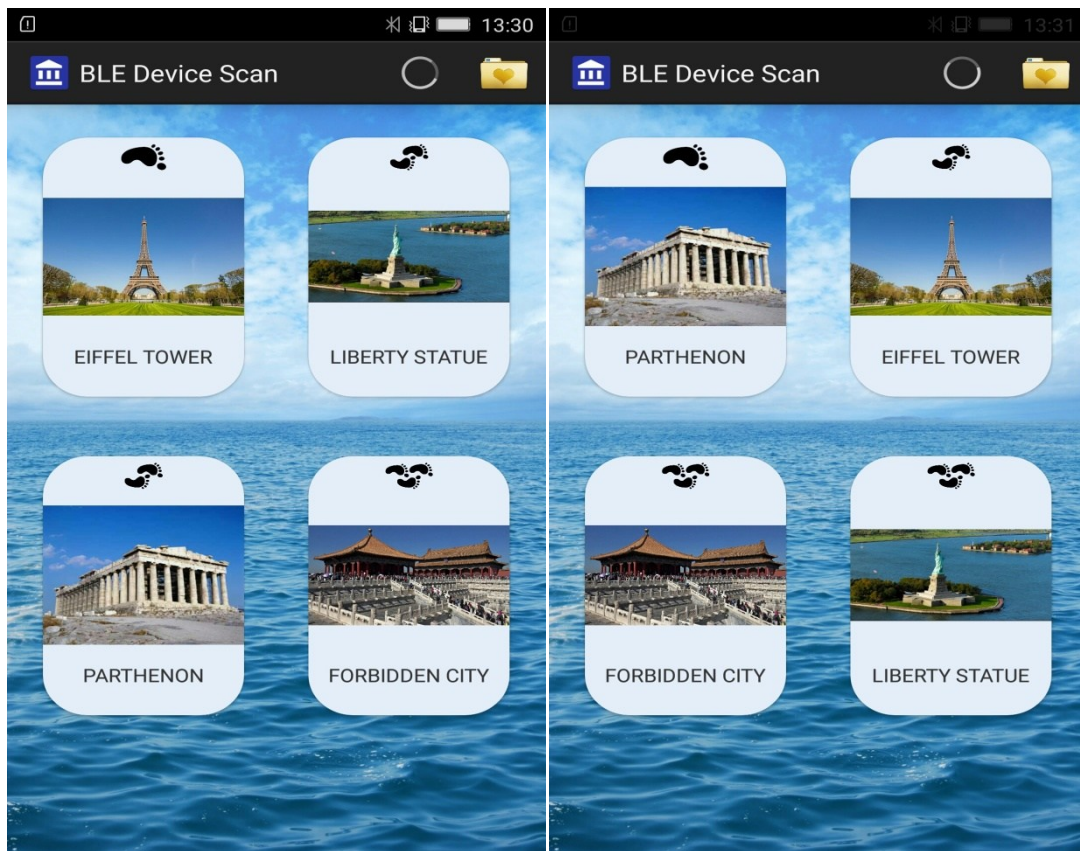
    if(deviceDistance > -60) viewHolder.deviceDistance.setImageDrawable(getDrawable(R.drawable.feet1));
    else if(deviceDistance > -75) viewHolder.deviceDistance.setImageDrawable(getDrawable(R.drawable.feet2));
    else viewHolder.deviceDistance.setImageDrawable(getDrawable(R.drawable.feet3));
    // Intent intent = new Intent(BluetoothDevice.ACTION_FOUND);
    final int deviceRssi = intent.getShortExtra(BluetoothDevice.EXTRA_RSSI,Short.MIN_VALUE);
    if (deviceName != null && deviceName.length() > 0)
        viewHolder.deviceName.setText(deviceName);
        //viewHolder.deviceRssi.setText(mLeScanCallback.onLeScan(device, int rssi, byte[] scanRecord));
    else
        viewHolder.deviceName.setText("Unknown device");
    // viewHolder.deviceAddress.setText(deviceData.getAddress());
    viewHolder.deviceImage.setImageDrawable(getDrawable(deviceImage));//////////////////////
}

```

Σχήμα 27: Presenting each tile

Για τον περιορισμό των όποιων αμφιβολιών, σχετικά με αυτό, προέκυπταν, προστέθηκε ένα εικονίδιο άνωθεν της εικόνας περιγραφής του εκθέματος. Το εικονίδιο αυτό, εμπνευσμένο από γνωστό παιχνίδι που απαιτεί την πλοήγηση του χρήστη σε δρόμους, βοηθά το χρήστη να αντιληφθεί ποιο έκθεμα είναι το πλησιέστερο, και έτι περισσότερο τη σειρά εγγύτητας. Το εικονίδιο ενδέχεται να είναι ένα, δύο ή τρία πέλματα μικρού παιδιού. Επομένως, στην περίπτωση που εμφανίζεται μόνο ένα, ο χρήστης αντιλαμβάνεται ότι βρίσκεται αρκετά κοντά στο συγκεκριμένο έκθεμα. Αντιθέτως, όταν εμφανίζονται τρία πέλματα, ο χρήστης αντιλαμβάνεται ότι απέχει αρκετά. Επιπροσθέτως, επειδή η παρουσίαση της διπλωματικής εργασίας πραγματοποιείται σε εσωτερικό χώρο, και δη αυτόν του εργαστηρίου, έγιναν κάποιες προσαρμογές, ούτως ώστε να γίνει μια προσομοίωση εσωτερικού χώρου, δηλαδή το ένα πέλμα θα εμφανίζεται σε μια απόσταση μικρότερη των 5 μέτρων, τα δύο πέλματα θα εμφανίζονται σε μια απόσταση περίπου 10 μέτρα, ενώ σε μεγαλύτερη απόσταση, μικρή όμως ταυτόχρονα για να εντοπίζεται με το Bluetooth πρωτόκολλο, εμφανίζονται 3 πέλματα. Στο τμήμα κώδικα που παρουσιάζεται που αφορά στη μέθοδο getView(), παρατηρείται ότι η επιλογή των εικονιδίων γίνεται με σύγκριση της ληφθείσας ισχύος RSSI με κάποιες σταθερές τιμές. Οι τιμές αυτές έχουν προκύψει από συλλογή, μεγάλης διάρκειας και όλων των διαφορετικών συσκευών που χρησιμοποιήθηκαν, μετρήσεων και, εν συνεχεία, κανονικοποίησή τους σε μία κοινή κλίμακα μέτρησης.

Παράδειγμα σε μια τυπική λειτουργία της εφαρμογής, καθώς ο χρήστης μετακινείται στον αρχαιολογικό χώρο, αποτελούν το Σχήμα 28 και το Σχήμα 29 που φαίνονται ακολούθως:



Σχήμα 28: App Example 1

Σχήμα 29: App Example 2

Οι δύο κύριες προκλήσεις που προέκυψαν κατά τον προγραμματισμό του συγκεκριμένου activity ήταν:

- η αποθήκευση των εκθεμάτων - συσκευών που επιθυμεί ο χρήστης να χρησιμοποιήσει σε μελλοντική χρονική στιγμή
- η ταξινόμηση κατά αύξουσα σειρά, με κριτήριο την απόσταση, όλων των εμφανιζομένων συσκευών

4.3.2 Δημιουργία Λίστας Αγαπημένων

Όσον αφορά στην πρώτη πρόκληση, έγινε χρήση δύο μεθόδων, ειδικά προγραμματισμένων για αυτόν τον σκοπό. Η μέθοδος `saveArrayList()` αποθηκεύει ένα array list ουσιαστικά σε ένα αρχείο κειμένου τύπου `.txt`. Με τον τρόπο αυτό, κάθε φορά που το activity εισέρχεται σε `onPause()` κατάσταση, η μέθοδος αυτή αναλαμβάνει την αποθήκευση των αγαπημένων, προς το χρήστη, αρχαιολογικών εκθεμάτων σε ένα αρχείο στη μνήμη του κινητού Android. Έτσι, είτε το πρόγραμμα κλείσει ξαφνικά είτε αλλάξει το activity που βρίσκεται στο προσκήνιο, η λίστα των αγαπημένων θα είναι αποθηκευμένη στην μνήμη του κινητού. Αντίστοιχα, η μέθοδος

getSavedArrayList() εντοπίζει το συγκεκριμένο αρχείο τύπου .txt στη μνήμη του κινητού, ανακτά το array list που είχε αποθηκευθεί σε παρελθοντική χρονική στιγμή και το μεταφέρει στη λίστα που χρειάζεται για να είναι και πάλι πλήρως λειτουργική η εφαρμογή. Η μέθοδος αυτή καλείται κάθε φορά που εκτελείται η μέθοδος onCreate(), για την πρώτη φορά που ανοίγει η συσκευή, ή onResume(), για όλες τις επόμενες φορές.

```
private ArrayList<String> getSavedArrayList(){
    ArrayList<String> savedArrayList = new ArrayList<>();
    try{
        FileInputStream inputStream = openFileInput( name: "marios.txt");
        ObjectInputStream in = new ObjectInputStream(inputStream);
        savedArrayList = (ArrayList<String>) in.readObject();
        in.close();
        inputStream.close();
    }catch(FileNotFoundException e){
        Log.d( tag: "TAG", e.getMessage());
        saveArrayList(savedArrayList);
    }
    catch (IOException e) {
        e.printStackTrace();
    }
    catch (ClassNotFoundException e) {
        e.printStackTrace();
    }
    return savedArrayList;
}
private void saveArrayList(ArrayList<String> arraylist){
    try{
        FileOutputStream fileOutputStream = openFileOutput( name: "marios.txt", Context.MODE_PRIVATE);
        ObjectOutputStream out = new ObjectOutputStream(fileOutputStream);
        out.writeObject(arraylist);
        Log.e( tag: "BLE_SCANNER", msg: "written");
        out.close();
        fileOutputStream.close();
    }catch (IOException e){
        Log.e( tag: "BLE_SCANNER", msg: "not written");
        e.printStackTrace();
    }
}
```

Σχήμα 30: Saving Favorite Device

4.3.4 Εκτίμηση απόστασης κόμβων

Η δεύτερη πρόκληση, η οποία σχετίζεται με την ταξινόμηση με κριτήριο την απόσταση, αντιμετωπίστηκε με τη βοήθεια της ληφθείσας ισχύος RSSI. Πιο συγκεκριμένα, έγινε χρήση της έτοιμης, από την βιβλιοθήκη του Android Studio, μεθόδου που ταξινομεί array lists. Η λίστα αυτή περιείχε όλες τις συσκευές και τις αντίστοιχες τιμές τους σε RSSI. Έτσι, με την μέθοδο Arrays.sort και στη συνέχεια με αντιστροφή της λίστας που δημιουργήθηκε προέκυψε η τελική λίστα που εμφανίζεται στην οθόνη. Χρειάζεται να σημειωθεί ότι το activity που εμφανίζεται στο προσκήνιο όταν ο χρήστης επιλέγει τα αγαπημένα του, λειτουργεί και παρουσιάζεται με παρόμοιο τρόπο με αυτόν του activity που περιγράψαμε προηγουμένως.


```

for (int i = 0; i < mLeDevices.size(); i++)
    mDevicesShown.add(mLeDevices.valueAt(i));

DeviceData[] dataArray = new DeviceData[mDevicesShown.size()];

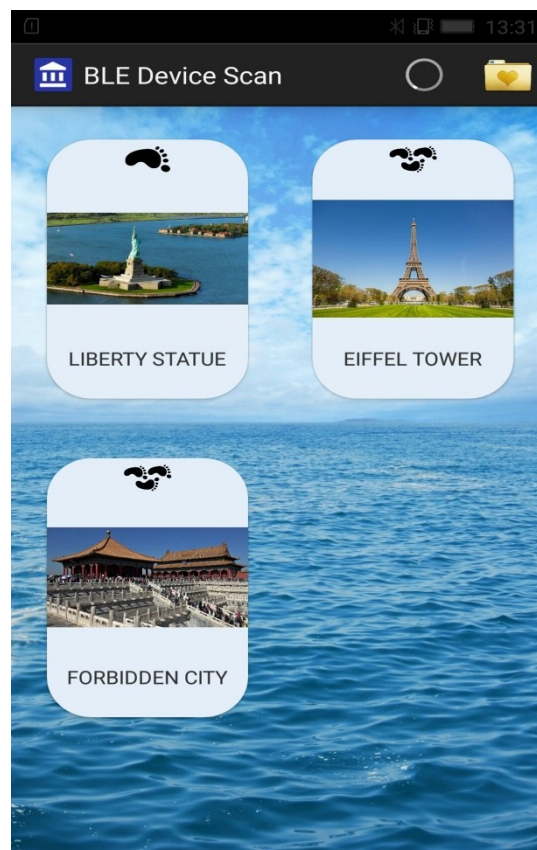
Arrays.sort(mDevicesShown.toArray(dataArray), new DeviceData.Comparator());

mDevicesShown = new ArrayList<>(Arrays.asList(dataArray));
Collections.reverse(mDevicesShown);

```

Σχήμα 31: Ταξινόμηση με σειρά απόστασης των εκθεμάτων

Στο Σχήμα 32, απεικονίζεται ένα τυπικό παράδειγμα από την κατάσταση λειτουργίας της εφαρμογής. Το έκθεμα που εμφανίζεται στο πάνω και αριστερό μέρος της οθόνης αντιστοιχεί στο πλησιέστερο έκθεμα. Το έκθεμα που βρίσκεται δεξιά του αντιστοιχεί στο δεύτερο πλησιέστερο έκθεμα. Στην επόμενη σειρά εκθεμάτων παρουσιάζονται το τρίτο και το τέταρτο έκθεμα με σειρά απόστασης. Επομένως, η απόσταση του εκθέματος από τον χρήστη σε σχέση με τα υπόλοιπα εκθέματα, όταν αυτό εμφανίζεται κάτω και δεξιά..



Σχήμα 32: App Example 3

Όπως προαναφέρθηκε και προηγουμένως, για να καταστεί περισσότερο προφανές σε έναν χρήστη, ο οποίος δεν είναι οικείος με την παρούσα εφαρμογή, έγινε προσθήκη ενός εικονιδίου άνωθεν της εικόνας κάθε εκθέματος. Το εικονίδιο αυτό ενδέχεται να απεικονίζει ένα, δύο ή τρία πέλματα υποδηλώνοντας έτσι την απόσταση (περισσότερα πέλματα αντιστοιχούν σε μεγαλύτερη απόσταση από τον

χρήστη). Στο Σχήμα 32, το έκθεμα "Liberty Statue" βρίσκεται σε απόσταση μικρότερη των 5 μέτρων, σε αντίθεση με τα άλλα δύο, τα οποία βρίσκονται σε απόσταση 10 μέτρων, ή ίσως λίγη περισσότερη.

Όταν ο χρήστης επιλέξει κάποιο από τα πλησιέστερα εκθέματα, που εμφανίζονται στο άνω μέρος της οθόνης του, η εφαρμογή αλλάζει το activity του προσκηνίου. Έτσι, εμφανίζονται ορισμένες εικόνες και ένας περιορισμένος όγκος κειμένου, προκειμένου να ενημερωθεί ο χρήστης σχετικά με το συγκεκριμένο έκθεμα. Βεβαίως, το μέγεθος των στοιχείων αυτών μπορεί να είναι έως και αρκετά MB, καθώς αυτά αποθηκεύονται τοπικά στην κινητή συσκευή και η ταχύτητα μετάδοσης τους είναι αρκετά υψηλή. Σε περίπτωση που είχε επιλεγεί η αποθήκευση όλων αυτών των στοιχείων στο Development Board, θα υπήρχε, ίσως με μεγάλα αρχεία, μικρή καθυστέρηση στην μετάδοση των δεδομένων.



Σχήμα 33: Πληροφορίες για το έκθεμα

Παρόλο, που στην Android εφαρμογή, εφαρμόζεται φίλτρο ως προς τις διευθύνσεις UUID, με σκοπό τον περιορισμό των συσκευών που εμφανίζονται στο κινητό του χρήστη και είναι μη σχετικές με το αντικείμενο (π.χ. τηλεοράσεις smart TV, custom beacons), υπάρχει μια πολύ μικρή πιθανότητα, να εμφανιστεί συσκευή με παρόμοιο UUID και να διαπεράσει το φίλτρο των UUID. Για το λόγο αυτό, σε πολλά σημεία του κώδικα της εφαρμογής, γίνεται έλεγχος και ως προς το όνομα των συσκευών, το οποίο έχει τεθεί προγραμματιστικά σε προηγούμενες ενέργειες. Ο τρόπος που γίνεται αυτό, εμφανίζεται στο Σχήμα 34, το οποίο είναι τμήμα

προγραμματιστικού κώδικα του activity που σχετίζεται με το activity προσκηνίου, όταν πατηθεί ένα από τα εκθέματα - συσκευές.

```
ImageView iv = (ImageView) findViewById(R.id.secondary_image);
TextView tv = (TextView) findViewById(R.id.secondary_text);
// Sets up UI references.
//((TextView) findViewById(R.id.device_address)).setText(mDeviceAddress);
if (mDeviceName.equals("PARTHENON")) {
    iv.setImageResource(R.drawable.parthenon1);
    tv.setText("    The Parthenon is a former temple, on the Athenian...");
}
else if (mDeviceName.equals("EIFFEL TOWER")) {
    iv.setImageResource(R.drawable.eiffel1);
    tv.setText("    The Eiffel Tower ...");
}
}
```

Σχήμα 34: Περαιτέρω έλεγχος συσκευών δικτύου

Πρωτεύων στόχος της παρούσας διπλωματικής εργασίας ήταν η εξυπηρέτηση αρχαιολογικών χώρων που βρίσκονται σε απομονωμένες τοποθεσίες και ίσως δεν παρέχεται πρόσβαση στο Διαδίκτυο. Σαφώς, μια εφαρμογή που έχει πρόσβαση σε WiFi, θα έχει πολύ περισσότερες λειτουργίες. Έτσι, για να είναι δελεαστική η χρήση της συγκεκριμένης εφαρμογής και σε αρχαιολογικό χώρο που υποστηρίζει σύνδεση στο Διαδίκτυο, προστέθηκε ένα feature σχετικό με τη σύνδεση σε WiFi ή Δεδομένα Κινητής Τηλεφωνίας. Όπως έχει ήδη αναφερθεί, όταν ο χρήστης επιλέγει μια συσκευή - έκθεμα, η εφαρμογή αλλάζει το activity του προσκηνίου. Στο activity αυτό, εμφανίζεται ένα menu στο πάνω τμήμα της οθόνης, στο οποίο ο χρήστης έχει τη δυνατότητα να επιλέξει, εφόσον αυτός επιθυμεί:

- την αποθήκευση της τρέχουσας συσκευής στο κινητό, έτσι ώστε να μπορεί να την επισκεφθεί και σε μελλοντική χρονική στιγμή
- την παραπομπή σε σχετικό οπτικοακουστικό υλικό, το οποίο βρίσκεται στην ιστοσελίδα YouTube.com

Όσον αφορά στην δεύτερη επιλογή, η εφαρμογή έχει λάβει ειδική άδεια χρήσης βίντεο από τον συγκεκριμένο ιστοχώρο. Τα βίντεο αυτά αναπαράγονται εντός της εφαρμογής, χωρίς να χρειάζεται να εκκινηθεί κάποιο άλλη εφαρμογή αναπαραγωγής βίντεο.

```
@Override
public void onInitializationSuccess(Provider provider, YouTubePlayer player, boolean wasRestored) {
    /** add listeners to YouTubePlayer instance **/
    player.setPlayerStateChangeListener(playerStateChangeListener);
    player.setPlaybackEventListener(playbackEventListener);

    /** Start buffering **/
    if (!wasRestored) {
        player.cueVideo(VIDEO_ID);
    }
}
}
```

Σχήμα 35: Αναπαραγωγή του αντίστοιχου βίντεο

Κεφάλαιο 5

Επίλογος

5.1 Συμπεράσματα

Ολοκληρώνοντας την σχεδίαση και ανάπτυξη διαδραστικού συστήματος για ξενάγηση σε αρχαιολογικούς χώρους με χρήση του πρωτοκόλλου Bluetooth Low Energy 6.0 προέκυψε δύο βασικά συμπεράσματα. Το πρώτο αφορά στη χρήση του δείκτη μέτρησης ληφθείσας ισχύος σήματος RSSI, το οποίο, παρόλο που είναι μια μέθοδος που χρησιμοποιείται σε παρόμοιες εφαρμογές και τεχνολογίες, παρουσιάζει ορισμένα μειονεκτήματα. Το πιο σημαντικό μειονέκτημα είναι οι επιπτώσεις του φαινομένου της σκίασης. Εξαιτίας αυτού, είναι πιθανό ένας κόμβος του συστήματος να εμφανίζεται πιο μακριά από ότι βρίσκεται πραγματικά. Αυτό συμβαίνει, καθώς τα φυσικά εμπόδια, μεταξύ των κόμβων και του κινητού τηλεφώνου, αποδυναμώνουν το σήμα. Ένα άλλο πρόβλημα που ενδέχεται να προκύψει είναι η προσωρινή εξαφάνιση ενός κόμβου, λόγω μεγάλου interval. Ο λόγος που ένας developer αυξάνει το interval είναι η ακόμα περισσότερη μείωση της κατανάλωσης ισχύος. Έτσι, εάν το κινητό σαρώσει την περιοχή, τη χρονική στιγμή που δεν εκπέμπεται σήμα, ο κόμβος θα εξαφανιστεί προσωρινά (περί το 1 δευτερόλεπτο).

Επιπλέον, ένα ακόμα συμπέρασμα αποτελεί η ανάγκη προσθήκης σε μία βάση δεδομένων χαρακτηριστικών για όλα τα πιθανά μοντέλα κινητών τηλεφώνων. Τέτοια χαρακτηριστικά αφορούν, κυρίως, στην αντιστοίχιση των διαφόρων τιμών RSSI με την απόσταση σε μέτρα (m). Με βάση όλα τα παραπάνω, παρατηρήθηκε πως η ακρίβεια εκτιμάται περίπου 4 - 5 μέτρα.

5.2 Μελλοντικές Κατευθύνσεις

Το σύστημα, σε αυτή του τη μορφή, έχει πολλές δυνατές επεκτάσεις, οι οποίες θα μπορούσαν να βελτιώσουν τόσο την εμπειρία του χρήστη, όσο και να αυξήσουν τη λειτουργικότητα της συσκευής. Έτσι, με μια αναβάθμιση του UI (user interface), θα μπορούσε το σύστημα να γίνει πιο ελκυστικό προς τον χρήστη. Παράλληλα, για αύξηση της ακρίβειας, θα μπορούσε να γίνει η χρήση της τεχνολογίας GPS, έτσι ώστε οι αποκλίσεις να είναι πιο σπάνιες και η λειτουργία πιο σταθερή. Τέλος, θα μπορούσε να αναπτυχθεί πλατφόρμα, με την οποία ο developer θα μπορούσε να αλληλεπιδράσει με την συσκευή, χωρίς να χρειάζεται η αποσυναρμολόγηση της, και να την ρυθμίσει όπως αυτός επιθυμεί.

Παράρτημα Α

Στο Παράρτημα αυτό, παρουσιάζονται φωτογραφίες από το τελικό προϊόν που κατασκευάστηκε:



Βιβλιογραφία

Bluetooth

1. Kevin Townsend, Carles Cufi, Akiba, Robert Davidson, “Getting Started with Bluetooth Low Energy”, O’Reilly, 2014
2. “Bluetooth Core Specification Version 4.0”, Bluetooth SIG, 2010
3. “Bluetooth Core Specification Version 4.1”, Bluetooth SIG, 2013
4. “Bluetooth Core Specification Version 5.0”, Bluetooth SIG, 2016
5. Bluetooth Technology Website: <http://www.bluetooth.com>
6. Bluetooth Technology Special Interest Group: <https://www.bluetooth.org>
7. Bluetooth History: <http://www.bluetooth.com/Pages/History-of-Bluetooth.aspx>
8. Bluetooth XML Schemas: <http://schemas.bluetooth.org/Pages/default.aspx>

Development Board DA14585

1. "DA14585 Product Brief", Dialog Semiconductor
2. "DA14585 Datasheet", Dialog Semiconductor
3. "CodeLess Tutorial (HTML)", Dialog Semiconductor
4. "Tutorial 1: Advertising contents (HTML)", Dialog Semiconductor
5. "Tutorial 2: Custom profile example_v1.2", Dialog Semiconductor
6. "Tutorial 3: Custom profile gatt cmd example_v1.2", Dialog Semiconductor
7. "Tutorial 4: Sleep modes current measurements_v1.0", Dialog Semiconductor
8. "Tutorial 5: BLE security example_v1.2", Dialog Semiconductor
9. "AN-B-055: DA14585 interfacing with external memory v1.1", Dialog Semiconductor
10. "AN-B-027: Designing Printed Antennas for Bluetooth Smart", Dialog Semiconductor
11. "UM-B-082: DA14585 SDK6 Porting Guide", Dialog Semiconductor
12. "UM-B-048: DA14585 Getting Started Guide with the Basic-Development Kit (HTML)", Dialog Semiconductor
13. "UM-B-080: DA14585 Software Developer's Guide", Dialog Semiconductor
14. Dialog Semiconductor: <http://www.dialog-semiconductor.com/>
15. Support Dialog Semiconductor: <http://support.dialog-semiconductor.com/>
16. Keil Embedded Development Tools: <http://www.keil.com/>
17. "BQ24090x Datasheet", Texas Instruments
18. "TPS6300x Datasheet", Texas Instruments

Android

1. Android Developers: <http://developer.android.com/index.html>
2. Android Development Tools: <http://developer.android.com/tools/index.html>
3. Android API Guide – Bluetooth:
<http://developer.android.com/guide/topics/connectivity/bluetooth.html>
4. Android API Guide – Bluetooth Low Energy:
<http://developer.android.com/guide/topics/connectivity/bluetooth-le.html>
6. Android Documentation – android.bluetooth package:
<https://developer.android.com/reference/android/bluetooth/package-summary.html>
7. Android API Guide – Activities:
<https://developer.android.com/guide/components/activities.html>
8. Android API Guide – Services:
<https://developer.android.com/guide/components/services.html>
9. Android API Guide – App Widgets:
<https://developer.android.com/guide/topics/appwidgets/index.html>
10. Stack Overflow: <http://stackoverflow.com>