

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ  
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ



**ΣΥΣΤΗΜΑ ΣΥΛΛΟΓΗΣ ΚΑΙ ΕΠΕΞΕΡΓΑΣΙΑΣ ΔΕΔΟΜΕΝΩΝ ΠΡΑΓΜΑΤΙΚΟΥ  
ΧΡΟΝΟΥ ΑΠΟ ΑΥΤΟΚΙΝΗΤΟ ΣΕ ΣΥΣΚΕΥΗ ANDROID**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**ΑΛΕΞΑΝΔΡΟΣ Α.  
ΠΑΛΟΓΛΟΥ**

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ:

ΚΑΘΗΓΗΤΗΣ Α. ΔΟΛΛΑΣ (ΕΠΙΒΛΕΠΩΝ)

ΚΑΘΗΓΗΤΗΣ Δ. ΠΝΕΥΜΑΤΙΚΑΤΟΣ

ΑΝΑΠΛ. ΚΑΘΗΓΗΤΗΣ Ι. ΠΑΠΑΕΥΣΤΑΘΙΟΥ

ΧΑΝΙΑ, ΜΑΡΤΙΟΣ 2015



# Πρόλογος

Η παρούσα διπλωματική εργασία υλοποιήθηκε, στα πλαίσια του Προγράμματος Προπτυχιακών Σπουδών του τμήματος Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών του Πολυτεχνείου Κρήτης, από τον Μάιο του 2014 έως και τον Ιανουάριο του 2015. Βασικό κριτήριο της επιλογής του συγκεκριμένου θέματος προς διερεύνηση ήταν η περεταίρω εμβάθυνση πάνω στον τομέα της δημιουργίας εφαρμογών για συσκευές με λειτουργικό σύστημα Android.

Στο σημείο αυτό, θα ήθελα να εκφράσω τις θερμές ευχαριστίες μου σε όλους όσους με βοήθησαν και συνέβαλλαν στην ολοκλήρωση της εργασίας αυτής. Πρώτον από όλους ευχαριστώ τον επιβλέποντα καθηγητή μου κ. Δόλλα για την επιλογή του θέματος, για την πολύ καλή και ευχάριστη συνεργασία που είχαμε και για τις γνώσεις που με τη βοήθειά του αποκόμισα όλο αυτό το διάστημα.

Συνεχίζοντας, θα ήθελα να πω ένα μεγάλο ευχαριστώ στην οικογένειά μου, για την πολύτιμη στήριξη της όλα αυτά τα χρόνια, συμβάλλοντας καθοριστικά στην εξέλιξή μου.

Τέλος, ευχαριστώ όλους τους φίλους και τα κοντινά μου πρόσωπα για την υπομονή, τη βοήθεια και τη στήριξη που μου πρόσφεραν κατά τη διάρκεια της εκπόνησης της παρούσας εργασίας, αλλά και καθ' όλη τη διάρκεια των σπουδών μου.

Χανιά, Μάρτιος 2015

Πιαλόγλου Αλέξανδρος



## Περίληψη

Στόχος της παρούσας διπλωματικής εργασίας είναι να εκμεταλλευτεί τις δυνατότητες που προσφέρουν οι έξυπνες κινητές συσκευές και συγκεκριμένα το λειτουργικό σύστημα Android, για την ανάπτυξη μιας εφαρμογής, η οποία δίνει τη δυνατότητα στο χρήστη της να λαμβάνει δεδομένα και πληροφορίες πραγματικού χρόνου.

Η εφαρμογή που υλοποιήθηκε έχει ως σκοπό την ενημέρωση σε πραγματικό χρόνο των χρηστών για πληροφορίες σχετικές με στοιχεία της μηχανής του αυτοκινήτου, της τοποθεσίας της συσκευής, αλλά και της διαδρομής που πραγματοποιούν. Οι πληροφορίες αυτές καλύπτουν ένα ευρύ φάσμα των αναγκών των οδηγών, καθώς πολλές από αυτές δεν είναι διαθέσιμες από το ίδιο το αυτοκίνητο. Η εφαρμογή διαθέτει ακόμα λειτουργίες που αποσκοπούν στην διάγνωση και πρόληψη τυχόν προβλημάτων που εμποδίζουν την ομαλή λειτουργία του αυτοκινήτου.

Πιο συγκεκριμένα, η εφαρμογή συλλέγει και εμφανίζει δεδομένα, τα οποία λαμβάνει από τον εγκέφαλο του αυτοκινήτου με τη βοήθεια του Bluetooth. Επιπλέον δίνει την επιλογή στο χρήστη, εφόσον το επιθυμεί, να καταγράψει τα συλλεγμένα δεδομένα σε αρχείο είτε για να τα μελετήσει εκ των υστέρων, είτε για να τα προσκομίσει σε κάποιο συνεργείο αυτοκινήτων, ώστε να μελετηθούν περαιτέρω από κάποιον ειδικό. Τέλος συλλέγει και επεξεργάζεται δεδομένα πραγματικού χρόνου, τα οποία και εμφανίζονται στον χρήστη, με βάση την τοποθεσία και ειδικότερα τις συντεταγμένες που λαμβάνονται από το GPS της συσκευής.



# Περιεχόμενα

Πρόλογος .....	iii
Περίληψη .....	v
<b>1. Εισαγωγή</b> .....	<b>1</b>
1.1. Γενικά για Ενσωματωμένα Συστήματα .....	1
1.2. Αντικείμενο και Συνεισφορά της Εργασίας .....	2
1.3. Διάρθρωση της Εργασίας .....	3
<b>2. Σχετική Τεχνολογία</b> .....	<b>5</b>
2.1. Σχετικές Εφαρμογές .....	5
2.2. Λογισμικό Android .....	6
2.2.1. Αρχιτεκτονική Λογισμικού Android .....	8
2.2.2. Δομή μίας Εφαρμογής Android .....	10
2.2.3. Κύκλος Ζωής μιας Δραστηριότητας .....	11
2.3. Bluetooth .....	13
2.4. GPS .....	14
2.4.1. Geolocation .....	15
2.5. On Board Diagnostics II (OBD-II) .....	15
2.5.1. Υποστηριζόμενα Πρωτόκολλα OBD-II .....	17
2.5.1.1. CAN Bus .....	18
2.5.2. OBD-II Modes .....	19
2.5.3. OBD-II PIDs .....	19
2.6. ELM327 .....	21
<b>3. Εργαλεία</b> .....	<b>23</b>
3.1. Εργαλεία Ανάπτυξης .....	23
3.1.1. Java Development Kit .....	23
3.1.2. Eclipse IDE για Java .....	24
3.1.3. Android SDK .....	25
3.1.4. ADT (Android Development Tools) Plug-in για το Eclipse .....	25
3.1.5. Adobe Photoshop CC 2014 .....	26

<b>4. Μοντελοποίηση και Αρχιτεκτονική Συστήματος</b>	<b>27</b>
4.1. Μοντελοποίηση Συστήματος .....	27
4.1.1. Επιλογή Bluetooth .....	28
4.1.2. Δεδομένα Πραγματικού Χρόνου .....	30
4.1.3. Χρονοπρογραμματισμός Εργασιών .....	33
4.2. Αρχιτεκτονική Συστήματος .....	35
4.3. Σχεδίαση εφαρμογής .....	37
4.3.1. Λειτουργικές Απαιτήσεις .....	37
4.3.2. Υποστήριξη Πολλαπλών Συσκευών .....	38
4.3.3. Σχεδίαση Διεπαφής Χρήστη .....	39
4.4. Ρυθμός Συλλογής Δεδομένων .....	40
4.4.1. Bluetooth .....	41
4.4.2. GPS .....	41
4.5. Βιβλιοθήκες .....	42
4.5.1. Bluetooth API .....	42
4.5.2. LocationManager .....	42
4.5.3. CustomGauge .....	43
4.5.4. SpeedometerView .....	43
<b>5. Σχεδίαση και Υλοποίηση Συστήματος</b>	<b>45</b>
5.1. Android Manifest.xml .....	45
5.2. Δραστηριότητες του Συστήματος .....	46
5.2.1. MainActivity .....	46
5.2.2. HelpActivity .....	47
5.2.3. BluetoothConnection .....	48
5.2.4. BluetoothChat .....	49
5.2.4.1. Καταγραφή Αποτελεσμάτων σε Αρχείο .....	58
5.2.4.2. Αποστολή Επιπλέον Κωδικού .....	59
5.2.4.3. Εκκαθάριση Κωδικών Σφαλμάτων .....	60
5.2.5. GPSTBasics .....	60
5.2.6. GPSInfo .....	63
5.3. Πιθανά Προβλήματα .....	64
<b>6. Επιβεβαίωση λειτουργίας</b>	<b>67</b>
6.1. Επιβεβαίωση Λειτουργίας Bluetooth .....	67
6.2. Επιβεβαίωση Εμφάνισης Στοιχείων Μηχανής .....	67
6.3. Επιβεβαίωση Λειτουργίας Καταγραφής Στοιχείων .....	68

6.4. Επιβεβαίωση Λειτουργίας GPS .....	68
6.5. Τελική Δοκιμή .....	69
<b>7. Συμπεράσματα - Μελλοντικές επεκτάσεις</b>	<b>71</b>
7.1. Συμπεράσματα .....	71
7.1.1. Απόκτηση Τεχνογνωσίας .....	72
7.1.2. Κατανομή Χρόνου .....	72
7.1.3. Υποστήριξη Διαφορετικών Πρωτοκόλλων και Κεφαλών .....	73
7.1.4. Ανάπτυξη νέων Εφαρμογών .....	73
7.2. Μελλοντικές Επεκτάσεις .....	74
<b>Παράρτημα OBD-II Modes and PIDs</b>	<b>77</b>
<b>Βιβλιογραφία</b>	<b>87</b>



# Κεφάλαιο 1

## Εισαγωγή

### 1.1 Γενικά για Ενσωματωμένα Συστήματα

Ας προσπαθήσουμε να κάνουμε μία όσο το δυνατόν πιο σύντομη εισαγωγή στον άκρως ενδιαφέροντα κόσμο των ενσωματωμένων συστημάτων, με μία γενική περιγραφή τους. Ο όρος ενσωματωμένο χρησιμοποιείται για να περιγράψει το υλικό και το λογισμικό ενός υπολογιστικού συστήματος ειδικού σκοπού, το οποίο έχει σχεδιαστεί για να επιτελεί μία συγκεκριμένη λειτουργία ή μία ορισμένη ομάδα λειτουργιών.

Τα ενσωματωμένα συστήματα συνήθως αποτελούν μέρος μεγαλύτερων συστημάτων και υπόκεινται συχνά σε χρονικούς περιορισμούς πραγματικού χρόνου, όσον αφορά το χρονικό διάστημα της επεξεργασίας των δεδομένων. Το κύριο χαρακτηριστικό τους είναι ότι χειρίζονται αποκλειστικά μια συγκεκριμένη εργασία, σε αντίθεση με τους υπολογιστές γενικού σκοπού, που ικανοποιούν μια μεγάλη ποικιλία λειτουργιών με την κατάλληλη χρήση του απαραίτητου λογισμικού. Η διάκριση βέβαια μεταξύ των δύο αυτών τύπων συστημάτων δεν είναι πάντα ξεκάθαρη, εφόσον πολλά ενσωματωμένα συστήματα παρέχουν τη δυνατότητα προγραμματισμού για πολλαπλές λειτουργίες, όπως η κατασκευή που μελετάμε στην παρούσα εργασία.



Εικόνα 1. Παραδείγματα ενσωματωμένων συστημάτων.<sup>[34]</sup>

Τα έξυπνα τηλέφωνα (smartphones) αποτελούν χαρακτηριστικό παράδειγμα ενσωματωμένων συστημάτων. Είναι συσκευές βασισμένες σε ένα λειτουργικό σύστημα και παρουσιάζουν περισσότερη προηγμένη υπολογιστική ικανότητα και συνδεσιμότητα σε σχέση με τα συμβατικά κινητά τηλέφωνα. Τα πρώτα έξυπνα τηλέφωνα συνδυάζαν τις λειτουργίες ενός προσωπικού ψηφιακού βοηθού (PDA) και ενός κινητού τηλεφώνου, ενώ στα μεταγενέστερα μοντέλα προστέθηκαν και άλλες λειτουργίες, όπως GPS, Bluetooth, μεταδότες USB, Wi-Fi, αισθητήρες επιτάχυνσης και γυροσκοπίου, ψηφιακές κάμερες και άλλα, με αποτέλεσμα σήμερα να διαμορφωθεί μια πολυχρηστική συσκευή με δυνατότητες αντίστοιχες ενός υπολογιστή. Πολλά σύγχρονα smartphones περιλαμβάνουν επίσης οθόνες αφής υψηλής ανάλυσης και web browsers που εμφανίζουν τυποποιημένες ιστοσελίδες, καθώς και βελτιστοποιημένες ιστοσελίδες για κινητά. Η πρόσβαση σε δεδομένα υψηλής ταχύτητας παρέχεται μέσω Wi-Fi και μέσω κινητών ευρυζωνικών υπηρεσιών. Τα τελευταία χρόνια, η ταχεία ανάπτυξη στην αγορά των εφαρμογών για κινητά και στο εμπόριο κινητών τηλεφώνων έχει γίνει οδηγός για την ευρεία υιοθέτηση των smartphones.<sup>[15]</sup>

## 1.2 Αντικείμενο και Συνεισφορά της Εργασίας

Στόχος της παρούσας Διπλωματικής Εργασίας είναι η υλοποίηση μιας εφαρμογής, η οποία δίνει τη δυνατότητα αλληλεπίδρασης του χρήστη της με το αυτοκίνητο. Πιο συγκεκριμένα, συλλέγει και επεξεργάζεται δεδομένα πραγματικού χρόνου και τα εμφανίζει σε συσκευές με λειτουργικό σύστημα Android. Παρότι υπάρχουν παρόμοια συστήματα ήδη, σκοπός μας είναι η απόκτηση της απαραίτητης τεχνογνωσίας για την ανάπτυξη μιας τέτοιας εφαρμογής.

Αρχικός στόχος της υλοποίησης μας ήταν να δημιουργηθεί μία εφαρμογή, η οποία θα δίνει τη δυνατότητα στον χρήστη να καταγράφει τα δεδομένα του εγκεφάλου της μηχανής την στιγμή εκείνη που στο αυτοκίνητο κάτι υπολειτουργεί. Αυτό μπορεί να βοηθήσει σε περιπτώσεις που κάτι συμβαίνει και διακόπτει την ομαλή λειτουργία του κινητήρα, αλλά πηγαίνοντας το σε κάποιο συνεργείο το πρόβλημα δεν παρουσιάζεται και έτσι δεν μπορεί να εντοπιστεί και να διορθωθεί.

Η εφαρμογή μας, ουσιαστικά ενσωματώνει δύο διαφορετικές εφαρμογές οι οποίες όμως μπορούν να λειτουργούν παράλληλα. Η πρώτη συλλέγει δεδομένα πραγματικού χρόνου με τη βοήθεια του Bluetooth από τον εγκέφαλο του αυτοκινήτου, τα οποία και εμφανίζει στην οθόνη. Ο χρήστης έχει τη δυνατότητα κάθε χρονική στιγμή να καταγράφει τα δεδομένα πραγματικού χρόνου, τα οποία συλλέγονται, σε ένα αρχείο. Σύμφωνα με το χρήστη, σε περίπτωση που κάτι δε λειτουργεί σωστά στο αυτοκίνητο, υπάρχει ειδική καταγραφή στο αρχείο επισημαίνοντας το γεγονός ότι εκείνη την χρονική στιγμή κάτι υπολειτουργεί. Τα αποτελέσματα μπορεί ο χρήστης είτε να τα μελετήσει εκ των υστέρων, είτε να τα προσκομίσει

σε κάποιο συνεργείο αυτοκινήτων για περαιτέρω μελέτη από κάποιον ειδικό. Η δεύτερη συλλέγει, εμφανίζει και επεξεργάζεται δεδομένα πραγματικού χρόνου με βάση την τοποθεσία και ειδικότερα τις συντεταγμένες που λαμβάνονται από το GPS της συσκευής μας. Εκτός από τις πληροφορίες για την τοποθεσία, προσφέρει την δυνατότητα στον χρήστη να υπολογίζει και να εμφανίζει πληροφορίες ανάλογες με αυτές ενός υπολογιστή ταξιδιού.

Η συνεισφορά της διπλωματικής συνοψίζεται ως εξής:

1. Εμφάνιση στοιχείων του αυτοκινήτου πραγματικού χρόνου μέσω Bluetooth.
2. Δυνατότητα καταγραφής των στοιχείων σε αρχείο στην μνήμη της συσκευής.
3. Επιλογή ειδικής καταγραφής σε περίπτωση βλάβης, σύμφωνα πάντα με το χρήστη.
4. Δυνατότητα μελέτης των αποτελεσμάτων από το χρήστη, αλλά και προσκόμιση του αρχείου σε κάποιο συνεργείο για μελέτη από κάποιον ειδικό.
5. Εκκαθάριση κωδικών σφαλμάτων που είναι καταχωρημένοι στην ηλεκτρονική μονάδα ελέγχου (ECU) του αυτοκινήτου.
6. Εμφάνιση στοιχείων πραγματικού χρόνου με τη βοήθεια του GPS της συσκευής.
7. Δυνατότητες έναρξης-παύσης-επανεκκίνησης υπολογισμού ταξιδιού και εμφάνιση των παραμέτρων.

### 1.3 Διάρθρωση της Εργασίας

Τελειώνοντας το πρώτο κεφάλαιο, στο οποίο αρχικά έγινε μία μικρή εισαγωγή για τα ενσωματωμένα συστήματα και στη συνέχεια παρουσιάστηκε ο σκοπός και η συνεισφορά της διπλωματικής εργασίας, ακολουθεί μια σύντομη περιγραφή για το τι θα περιέχουν τα επόμενα κεφάλαια. Τα κεφάλαια δύο και τρία ενσωματώθηκαν για να ενημερώσουν τον αναγνώστη για τις έννοιες και την απαραίτητη τεχνολογία που χρησιμοποιήθηκε και είναι περισσότερο θεωρητικά, ενώ στα κεφάλαια που ακολουθούν αναλύεται σε βάθος η σχεδίαση και η αρχιτεκτονική του συστήματος που υλοποιήθηκε και παρουσιάζονται αναλυτικά όλες οι λειτουργίες του.

Πιο συγκεκριμένα, στο δεύτερο κεφάλαιο αναφέρονται ορισμένες πληροφορίες, σχετικές με την έρευνα που κάναμε και τις τεχνολογίες που χρησιμοποιήσαμε στην παρούσα διπλωματική, όπως είναι το λογισμικό Android, το Bluetooth και το GPS. Ακόμα αναλύεται το σύστημα διαγνωστικού ελέγχου αυτοκινήτου, αλλά και ο μικροελεγκτής ELM327 που είναι απαραίτητος για το παραπάνω σύστημα.

Το τρίτο κεφάλαιο παρουσιάζει στα εργαλεία που μας βοήθησαν στην υλοποίηση του συστήματός μας, καθώς και στις βιβλιοθήκες που χρησιμοποιήθηκαν.

Στο τέταρτο κεφάλαιο περιγράφεται σε βάθος η μοντελοποίηση και η αρχιτεκτονική του συστήματός μας. Ακόμα αναλύεται ο τρόπος και ο ρυθμός συλλογής των δεδομένων μας, αλλά και οι κανόνες που ακολουθήθηκαν στην σχεδίαση του συστήματός μας.

Στο πέμπτο κεφάλαιο γίνεται η παρουσίαση της εφαρμογής μας, στην οποία παραθέτονται όλες τις οθόνες που μπορούν να εμφανιστούν στο χρήστη κατά την διάρκεια της εκτέλεσής της, ενώ ακόμα αναλύονται εκτενώς και όλες οι λειτουργίες που είναι διαθέσιμες.

Στο έκτο κεφάλαιο αναφέρονται τα βήματα που ακολουθήθηκαν για την επιβεβαίωση της ορθής λειτουργία της εφαρμογής. Αρχικά επιβεβαιώνεται ξεχωριστά κάθε ένα από τα κομμάτια της υλοποίησης, ενώ στην συνέχεια επιβεβαιώνονται οι λειτουργίες της ολοκληρωμένης εφαρμογής με όλα τα συστατικά μέρη της συνενωμένα.

Τέλος στο έβδομο κεφάλαιο παρατίθενται κάποια χρήσιμα συμπεράσματα τα οποία εξάγαμε κατά την διάρκεια εκπόνησης της παρούσας εργασίας και προτάσεις, για πιθανές μελλοντικές επεκτάσεις που μπορεί να γίνουν βασισμένες στην εργασία αυτή.

## Κεφάλαιο 2

### Σχετική Τεχνολογία

Στο κεφάλαιο αυτό θα παρουσιάσουμε όλες τις τεχνολογίες που είναι απαραίτητες για την καλύτερη κατανόηση και μελέτη του συστήματος μας. Επίσης θα αναφερθούμε σε σχετικές διαθέσιμες εφαρμογές που κυκλοφορούν.

#### 2.1 Σχετικές Εφαρμογές

Ψάχνοντας, μπορεί κανείς να βρει αρκετές σχετικές εφαρμογές, που αφορούν την συλλογή δεδομένων από τον εγκέφαλο του αυτοκινήτου, καθώς και την συλλογή δεδομένων της τοποθεσίας με την βοήθεια του GPS δέκτη της συσκευής. Όμως οι περισσότερες από αυτές είναι διαθέσιμες επί πληρωμή, καθώς επίσης ο κώδικας τους δεν είναι ελεύθερος στο διαδίκτυο για μελέτη.

Οι πιο γνωστές σχετικές εφαρμογές για την συλλογή δεδομένων από το αυτοκίνητο είναι οι εξής:

- Η Android-OBDR-Reader<sup>[47]</sup>, η οποία είναι η μόνη αντίστοιχη εφαρμογή ανοιχτού κώδικα. Η συγκεκριμένη εφαρμογή όμως χρησιμοποιεί μια βιβλιοθήκη, την OBD Java API, η οποία επιτρέπει στους προγραμματιστές να γράφουν εύκολα και γρήγορα εφαρμογές OBD, χωρίς να γράφουν κώδικα χαμηλού επιπέδου για την επικοινωνία.
- Η Torque Pro<sup>[48]</sup> είναι μια ολοκληρωμένη επαγγελματική εφαρμογή, η οποία όμως δεν είναι δωρεάν για τους χρήστες και ο κώδικας της δεν είναι ελεύθερος στο διαδίκτυο. Έχει αρκετές παρόμοιες λειτουργίες με την εφαρμογή μας, όμως δεν περιέχει λειτουργία για τον υπολογισμό στοιχείων της διαδρομής που πραγματοποιεί ο χρήστης.
- Η εφαρμογή OBD Link<sup>[49]</sup> είναι μια ακόμα εφαρμογή. Παρέχεται δωρεάν στους χρήστες και τους παρέχει δεδομένα πραγματικού χρόνου του οχήματος, καθώς και καταγραφή της διαδρομής που πραγματοποιούν με τη βοήθεια της βιβλιοθήκης Maps API.

- Η OBD Droidscan PRO<sup>[50]</sup> είναι μια ακόμα επί πληρωμή και κλειστού κώδικα εφαρμογή για την συλλογή δεδομένων πραγματικού χρόνου μόνο από το αυτοκίνητο με λειτουργία καταγραφής των δεδομένων.
- Η OBD Car Doctor<sup>[51]</sup> επίσης δεν παρέχεται δωρεάν στους χρήστες και παρέχει λειτουργίες συλλογής και καταγραφής στοιχείων οχήματος και διαδρομής με τη βοήθεια της βιβλιοθήκης Maps API.

Αντίθετα για το κομμάτι του GPS υπάρχουν πάρα πολλές παρόμοιες εφαρμογές, γεγονός που προκύπτει από τον πολύ χαμηλότερο βαθμό δυσκολίας για την ανάπτυξη μιας τέτοιας εφαρμογής.

Συνοπτικά οι περισσότερες εφαρμογές που συναντήσαμε δεν ενσωματώνουν ταυτόχρονα και τους δύο τρόπους συλλογής δεδομένων σε μια ενιαία εφαρμογή (GPS, Bluetooth). Σε μερικές από αυτές συναντήσαμε λειτουργίες και δυνατότητες καταγραφής των δεδομένων που συλλέγονται, ωστόσο σε καμία από αυτές τις εφαρμογές δεν παρατηρήθηκε κάποια λειτουργία, η οποία να μπορεί να πραγματοποιεί ειδική καταγραφή σε περίπτωση που ο χρήστης της αντιληφθεί πως κάτι στο όχημα υπολειτουργεί.

## 2.2 Λογισμικό Android

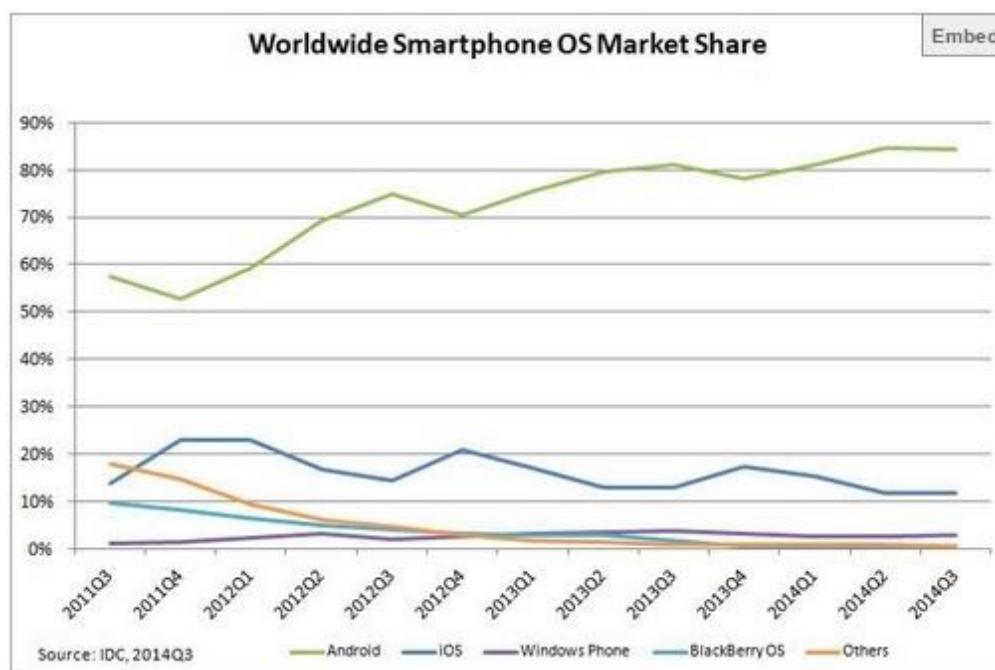
Ένα από τα πιο διαδεδομένα λειτουργικά συστήματα σήμερα είναι το Android. Το Android αποτελεί ένα λειτουργικό σύστημα που συναντάται κυρίως σε ενσωματωμένα συστήματα, όπως είναι τα κινητά τηλέφωνα, τα tablet αλλά και άλλες κινητές συσκευές.



Εικόνα 2. Το λογότυπο για το λειτουργικό σύστημα Android είναι ένα ρομπότ σε χρώμα πράσινου μήλου.<sup>[22]</sup>

Το Android αρχικά αναπτύχθηκε από τη Google και αργότερα από την Open Handset Alliance. Ο πυρήνας του Android είναι βασισμένος στον πολύ γνωστό και αρκετά δοκιμασμένο πυρήνα του λειτουργικού Linux. Κληρονομώντας όλα τα χαρακτηριστικά ασφαλείας του Linux και όλες τις διαχειριστικές τεχνικές μνήμης και επεξεργαστή που αυτό διαθέτει, το Android, καθίσταται ένα αρκετά αξιόπιστο λειτουργικό σύστημα. Επιπλέον λόγω του ότι το Android είναι λογισμικό ανοιχτού κώδικα, ο καθένας μπορεί να το προσαρμόσει στις δικές του ανάγκες και στο δικό του υλικό. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με τη χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας τη συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από τη Google.

Η πρώτη παρουσίαση της πλατφόρμας Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance, μιας σύμπραξης εταιρειών-κολοσσών στον τομέα της τεχνολογίας. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού.<sup>[16]</sup> Στις 30 Απριλίου 2009 κυκλοφόρησε η επίσημη ενημέρωση έκδοσης 1.5 για το Android, ενώ η τελευταία μέχρι στιγμής έκδοση είναι η 5.0 με κωδική ονομασία Lollipop.



Period	Android	iOS	Windows Phone	BlackBerry OS	Others
Q3 2014	84.4%	11.7%	2.9%	0.5%	0.6%
Q3 2013	81.2%	12.8%	3.6%	1.7%	0.6%
Q3 2012	74.9%	14.4%	2.0%	4.1%	4.5%
Q3 2011	57.4%	13.8%	1.2%	9.6%	18.0%

Εικόνα 3. Παγκόσμιο μερίδιο αγοράς λειτουργικού συστήματος για smartphones.<sup>[24]</sup>

Τέλος, σύμφωνα με τις τελευταίες μελέτες το μεγαλύτερο ποσοστό κινητών συσκευών αυτή τη στιγμή χρησιμοποιεί το λειτουργικό σύστημα Android. Το ποσοστό συσκευών που τρέχουν το λειτουργικό Android ξεπερνά το 80%, πολύ περισσότερα δηλαδή απ' ότι όλα τα άλλα λειτουργικά συστήματα για κινητές συσκευές μαζί, όπως iOS, Windows Phone, Blackberry OS κλπ.

Όλα τα προαναφερθέντα χαρακτηριστικά του Android, αλλά και η ευκολία χρήσης και ανάπτυξης του συγκεκριμένου λειτουργικού ήταν σημαντικοί παράγοντες που μας οδήγησαν στην επιλογή αυτού του λειτουργικού συστήματος.

### 2.2.1 Αρχιτεκτονική Λογισμικού Android

Το Android αποτελεί μια πλατφόρμα όπου βασίζεται στο Linux Kernel και το καινοτόμο γεγονός πως είναι ανοιχτό στο κόσμο, δηλαδή ο καθένας μπορεί να το τροποποιήσει και να φτιάξει μια δικιά του εκδοχή του λειτουργικού, το καθιστά ιδιαίτερα δημοφιλές.

Βασικά χαρακτηριστικά του Android λογισμικού είναι τα εξής:<sup>[16]</sup>

- Λειτουργίες Οθόνης
- Αποθήκευση Δεδομένων
- Αποστολή μηνυμάτων
- Περιήγηση στον Ιστό
- Συνδεσιμότητα (Bluetooth, 3G/4G,EDGE και Wi-Fi)
- Υποστήριξη Java
- Υποστήριξη Πολυμέσων
- Περιβάλλον Ανάπτυξης Λογισμικού
- Αγορά και Εγκατάσταση Εφαρμογών
- Δυνατότητα να συνεργαστεί με κάμερες στατικής ή κινούμενης εικόνας, οθόνες αφής, GPS, αισθητήρες επιτάχυνσης, μαγνητόμετρα, δισδιάστατους καθώς και τρισδιάστατους επιταχυντές γραφικών

Το λογισμικό αυτό βασίζεται στον πυρήνα του Linux για τις κύριες λειτουργίες όπως:

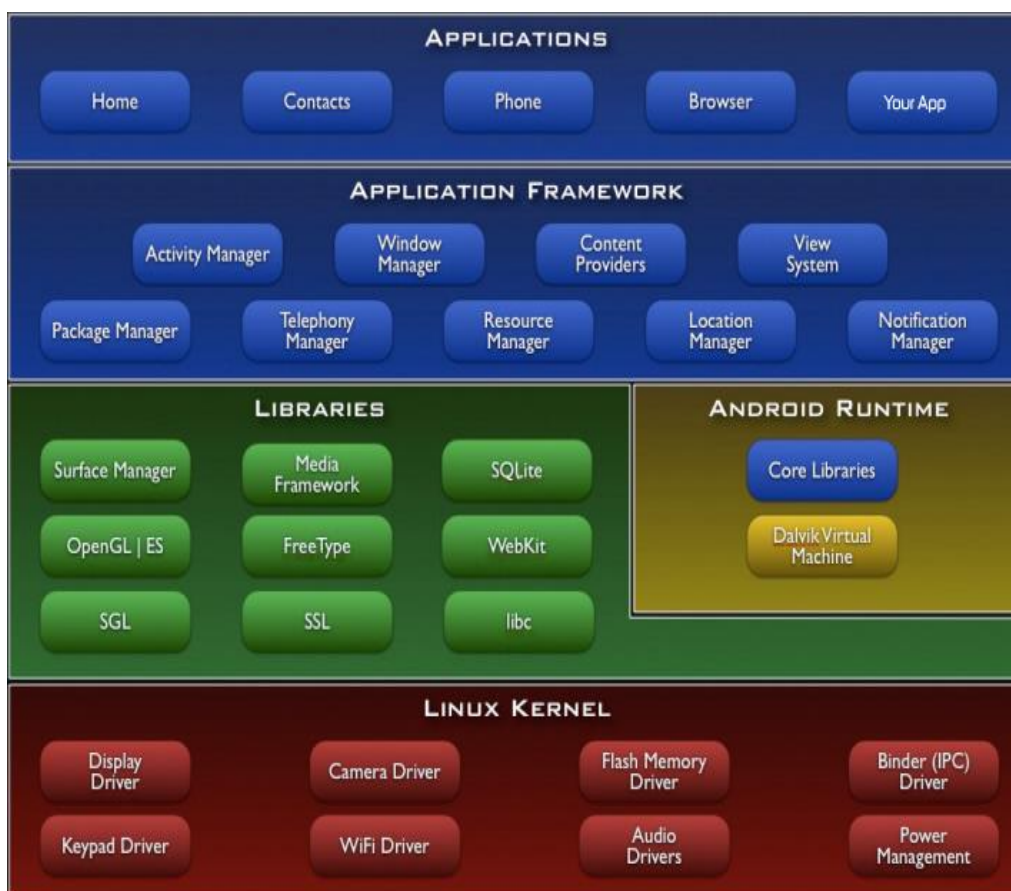
- Security
- Memory Management
- Process Management
- Network Stack
- Driver Model

Το λογισμικό Android όπως φαίνεται και στο σχήμα που ακολουθεί, μπορεί να χωριστεί σε πέντε επίπεδα: Το Linux Kernel, τις μητρικές βιβλιοθήκες (native libraries), το Android Runtime, το Application framework και το Applications.

Το Kernel που χρησιμοποιείται είναι της σειράς Linux 2.6, τροποποιημένο λόγω των ιδιαίτερων αναγκών στη διαχείριση ενέργειας, στη διαχείριση της μνήμης και του περιβάλλοντος λειτουργίας.

Επειδή το Android προορίζεται για συσκευές με μικρή κύρια μνήμη και χαμηλής ενέργειας επεξεργαστές, οι βιβλιοθήκες για εντατικές εργασίες του CPU και του GPU μεταφράζονται σε τροποποιημένες μητρικές βιβλιοθήκες της συσκευής. Βασικές βιβλιοθήκες όπως η libc και η lirm αναπτύχθηκαν ειδικά για χαμηλή κατανάλωση ενέργειας.

Το Android Runtime αποτελείται από την εικονική συσκευή Dalvik και βιβλιοθήκες πυρήνα Java. Η εικονική συσκευή Dalvik είναι μεταγλωττιστής για κώδικα byte ο οποίος έχει μετατραπεί από κώδικα byte Java σε κώδικα byte Dalvik. Τα framework είναι γραμμένα σε Java και παρέχουν αφαιρέσεις από μητρικές βιβλιοθήκες και δυνατότητες του Dalvik στις εφαρμογές. Οι εφαρμογές Android τρέχουν τη δική τους εικονική συσκευή Dalvik και μπορούν να περιέχουν διάφορα στοιχεία όπως είναι τα εξής: Activities, services, broadcast receivers. Τα στοιχεία αυτά αλληλεπιδρούν μεταξύ τους και μπορούν να αλληλεπιδράσουν και με άλλες εφαρμογές μέσω μίας προσθήκης (intent).



Εικόνα 4. Αρχιτεκτονική Λογισμικού Android. [35]

### 2.2.2 Δομή μίας Εφαρμογής Android

Σε αυτή την ενότητα θα αναλυθούν λεπτομερώς ο τρόπος και οι τεχνικές που χρησιμοποιήθηκαν για την υλοποίηση των κυριότερων λειτουργιών της Android εφαρμογής. Πρώτα να αναφέρουμε κάποια βασικά χαρακτηριστικά των Android εφαρμογών. Μια εφαρμογή για το λειτουργικό σύστημα Android αποτελείται κυρίως από πολλές δραστηριότητες (activities), ενώ μπορεί να διαθέτει επίσης services και broadcast receivers.

Τα στοιχεία αυτά δεν είναι απαραίτητα ως σύνολο για την ανάπτυξη μίας εφαρμογής, αλλά προκειμένου η εφαρμογή να περιέχει ένα γραφικό περιβάλλον χρήσης πρέπει να υπάρχει τουλάχιστον το στοιχείο Activity. Μια Activity είναι μια σελίδα στην εφαρμογή. Ουσιαστικά activity είναι η κάθε κλάση η οποία διαθέτει δική της γραφική διεπαφή χρήστη.

Τα στοιχεία Services και Broadcast Receivers επιτρέπουν στην εφαρμογή να εκτελεί διάφορες εργασίες στο παρασκήνιο της εφαρμογής. Ένα Service είναι το συστατικό της εφαρμογής το οποίο τρέχει συνεχώς, συνήθως για την εκτέλεση κάποιας χρονοβόρας διαδικασίας ή για την εκτέλεση κάποιας λειτουργίας που δεν απαιτείται αλληλεπίδραση με τον χρήστη. Τέλος ένα broadcast receiver είναι το συστατικό της εφαρμογής το οποίο «ακούει» και περιμένει για ένα συγκεκριμένο broadcast που προκαλεί κάποιο event. Για παράδειγμα ο broadcast receiver μπορεί να καλείται στο ξεκίνημα της συσκευής ή στο ξεκίνημα της εφαρμογής. Μόλις συμβεί το συγκεκριμένο event, το broadcast receiver θα εκτελέσει τον κώδικά του και θα τερματίσει.

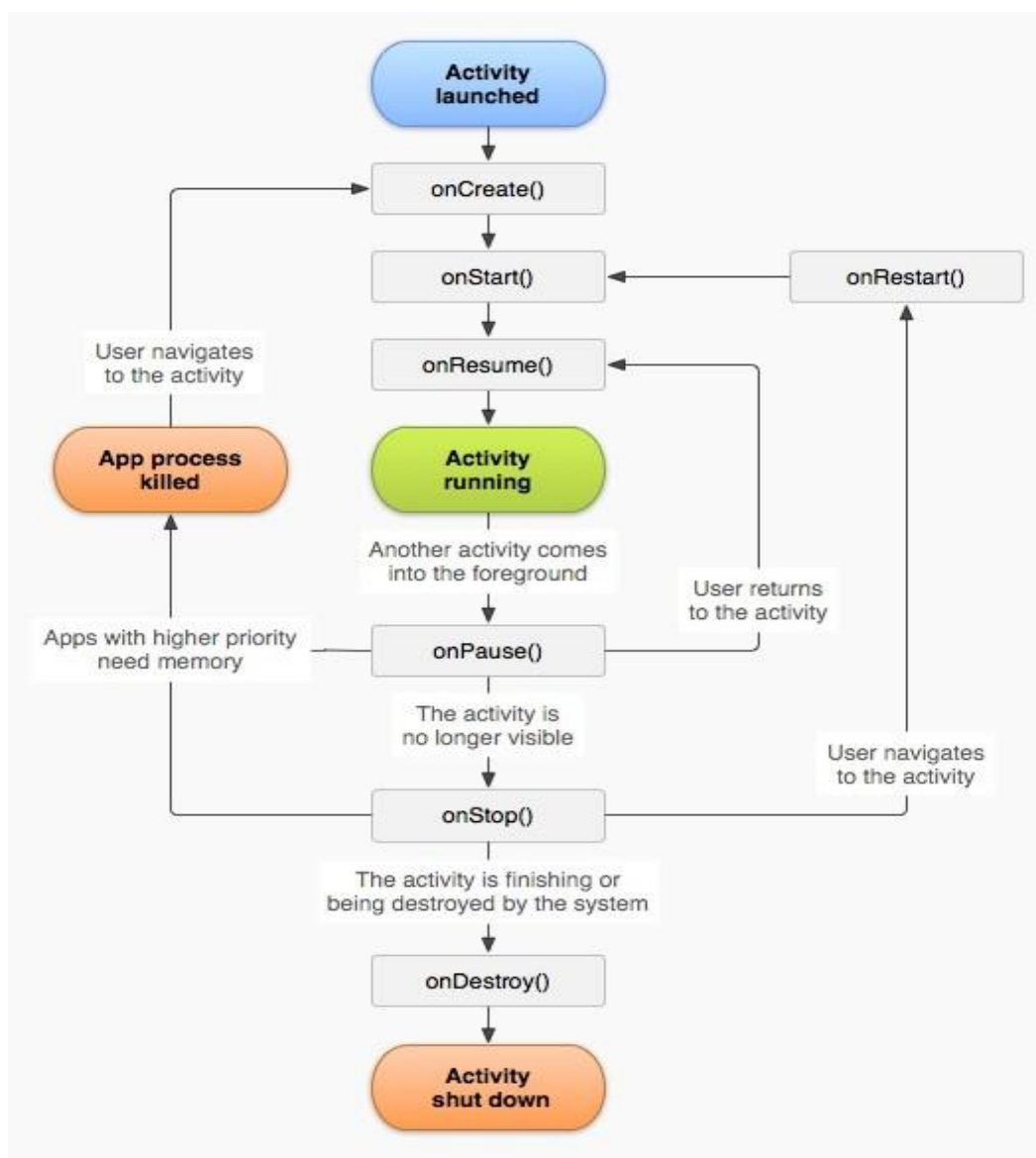
Επίσης μια εφαρμογή σε Android διαθέτει και άλλα σημαντικά στοιχεία όπως:

- Το Manifest αρχείο το οποίο περιέχει σημαντικές λεπτομέρειες για την εφαρμογή, όπως είναι οι εκδόσεις του Android που υποστηρίζει η εφαρμογή, οι απαιτούμενες άδειες που αποδέχεται ο χρήστης κατά την εγκατάστασή της, καθώς και όλες τις activities που υπάρχουν στην εφαρμογή.
- Το strings αρχείο το οποίο περιέχει το κείμενο που χρησιμοποιείται στην εφαρμογή.
- Το styles αρχείο που περιέχει τα θέματα και άλλα δεδομένα που έχουν να κάνουν με την εμφάνιση της εφαρμογής.
- Το colors αρχείο με όλους τους κωδικούς χρωμάτων που χρησιμοποιεί η εφαρμογή.
- Τον φάκελο drawable που περιέχει όλες τις εικόνες που χρησιμοποιεί η εφαρμογή.
- Τα xml αρχεία που περιέχουν όλες τις πιθανές οθόνες που εμφανίζονται κατά την εκτέλεση της εφαρμογής.

Τέλος ο κώδικας από κάθε στοιχείο μιας εφαρμογής μαζί με πρόσθετες πηγές όπως βιβλιοθήκες, εικόνες και άλλα απαραίτητα δεδομένα, συνιστούν ένα απλό αρχείο .apk, το οποίο αποτελεί τον τύπο αρχείου για μία εκτελέσιμη εφαρμογή Android.

### 2.2.3 Κύκλος Ζωής μιας Δραστηριότητας

Μια Activity (δραστηριότητα) είναι μία απλή οθόνη μιας εφαρμογής, όπως ένα παράθυρο περιαγωγής ή μία οθόνη ρυθμίσεων. Περιέχει τα οπτικά στοιχεία, είτε αυτά παρουσιάζουν κάποια δεδομένα, όπως μία εικόνα, είτε επιτρέπουν την αλληλεπίδραση με τον χρήστη, όπως ένα πλήκτρο. Κάθε εφαρμογή μπορεί να περιέχει πολλές activities. Η μεταφορά μεταξύ αυτών των δραστηριοτήτων γίνεται μέσω του intent. Όλες οι δραστηριότητες είναι υποκλάσεις της android.app. Αυτή η μέθοδος χρειάζεται προκειμένου να ελέγχεται η πολλαπλή λειτουργία εφαρμογών (multitasking) και βοηθάει ώστε να διασφαλίζεται η καλύτερη λειτουργία όταν η μνήμη είναι περιορισμένη.



Εικόνα 5. Κύκλος ζωής μιας δραστηριότητας.<sup>[22]</sup>

Τα στάδια τα οποία καλούνται κατά την διάρκεια εκτέλεσης μιας δραστηριότητας είναι: <sup>[32]</sup>

- `onCreate()`: Η μέθοδος αυτή καλείται προκειμένου να γίνουν οι απαραίτητες αρχικοποιήσεις και να κατασκευαστεί η διαδικασία. Η επόμενη μέθοδος είναι πάντα η `onStart()`.
- `onRestart()`: Αυτή η μέθοδος καλείται όταν η διαδικασία είναι σταματημένη και πρέπει να επανεκκινήσει.
- `onStart()`: Με αυτή τη μέθοδο το δέντρο διεργασιών γίνεται ορατό και η διαδικασία είναι ορατή στο χρήστη, αλλά δεν είναι ακόμα στο προσκήνιο.
- `onResume()`: Η διαδικασία είναι στο επίκεντρο και μπορεί να δεχτεί είσοδο από το χρήστη. Σε αυτό το σημείο η εφαρμογή είναι στο προσκήνιο.
- `onPause()`: Εάν η εφαρμογή πάψει να είναι στο προσκήνιο ή η συσκευή μπει σε κατάσταση αναμονής, τότε καλείται αυτή η μέθοδος και ο τύπος της διεργασίας γίνεται απλά ορατός. Μετά την εκτέλεση αυτής της μεθόδου, το σύστημα μπορεί ανά πάσα στιγμή να τερματίσει την εφαρμογή, επομένως πρέπει όλες οι διεργασίες να τερματιστούν και όλα τα δεδομένα που δεν είναι αποθηκευμένα να αποθηκευτούν. Η διαδικασία μπορεί είτε να επανεκκινήσει είτε να τερματιστεί.
- `onStop()`: Η διαδικασία δεν είναι πλέον ορατή, έχει περάσει στο παρασκήνιο και η εφαρμογή μπορεί να τερματιστεί ανά πάσα στιγμή από το σύστημα προκειμένου να ελευθερωθεί μνήμη. Η διαδικασία μπορεί είτε να καταστραφεί είτε να επανεκκινηθεί.
- `onDestroy()`: Αυτή η μέθοδος καλείται ακριβώς πριν το σύστημα τερματίσει την εφαρμογή ή όταν η εφαρμογή διαγράψει τη διαδικασία.

### 2.3 Bluetooth

Πρόκειται για μια ασύρματη τηλεπικοινωνιακή τεχνολογία μικρών αποστάσεων, η οποία μπορεί να μεταδώσει σήματα μέσω μικροκυμάτων σε ψηφιακές συσκευές. Επομένως το Bluetooth είναι ένα πρωτόκολλο, το οποίο παρέχει προτυποποιημένη ασύρματη επικοινωνία ανάμεσα σε PDA, κινητά τηλέφωνα, φορητούς υπολογιστές, προσωπικούς υπολογιστές, εκτυπωτές, καθώς και ψηφιακές φωτογραφικές μηχανές ή ψηφιακές κάμερες, μέσω μιας ασφαλούς, φθηνής και παγκοσμίως διαθέσιμης, χωρίς ειδική άδεια, ραδιοσυχνότητας μικρής εμβέλειας.



Εικόνα 6. Εικονίδιο Bluetooth.<sup>[36]</sup>

Οι προδιαγραφές του Bluetooth καθορίζουν την «ασύρματη» τεχνολογία χαμηλού κόστους και χαμηλής ισχύος, που εξαλείφει τα καλώδια μεταξύ των κινητών συσκευών και επιτρέπει τη διασύνδεσή τους. Το Bluetooth λειτουργεί στο «αδέσμευτο» φάσμα συχνοτήτων των 2,4 GHz, ώστε οι συσκευές που το ενσωματώνουν να μπορούν να λειτουργήσουν απροβλημάτιστα σε οποιοδήποτε σημείο του πλανήτη.

Για να περιοριστούν στο ελάχιστο οι παρεμβολές από παρεμφερείς συσκευές, το Bluetooth εκμεταλλεύεται την αμφίδρομη επικοινωνία και τη μέθοδο μετάδοσης με διασπορά φάσματος Frequency Hopping (έως και 1600 εναλλαγές συχνότητας ανά δευτερόλεπτο). Από φυσική άποψη επίσης το Bluetooth λειτουργεί περίπου στα 2.4 GHz, προδιαγράφει τρία επίπεδα ισχύος της εκπομπής από τα οποία εξαρτάται και η εμβέλεια επικοινωνίας.<sup>[17]</sup> Η μέση εμβέλεια για τις συσκευές Bluetooth είναι περίπου δέκα μέτρα, αλλά πιο ισχυρές συσκευές μπορούν να φτάσουν ακόμα και τα εκατό μέτρα.

Σε γενικές γραμμές μπορούμε να πούμε ότι το Bluetooth είναι μια τεχνολογία με μικρή εμβέλεια, ειδικά σε σύγκριση με ασύρματες τεχνολογίες της κατηγορίας IEEE 802.11. Ωστόσο, η μικρή εμβέλεια των δικτύων Bluetooth κάποιες φορές αποτελεί πλεονέκτημα όσον αφορά την ασφάλεια.

Το Bluetooth επιτρέπει τις απευθείας συνδέσεις από συσκευή προς συσκευή (point to point), καθώς και την ταυτόχρονη σύνδεση έως και 7 συσκευών με τη χρήση μιας μοναδικής συχνότητας. Τις προδιαγραφές της συγκεκριμένης τεχνολογίας ανέπτυξε και υποστηρίζει το Bluetooth Special Interest Group, ενώ η τελευταία «δημόσια» έκδοσή τους είναι η 4.2.

## 2.4 GPS (Global Positioning System)

Το GPS (Παγκόσμιο Σύστημα Θεσιθεσίας) είναι ένα παγκόσμιο σύστημα εντοπισμού θέσης, το οποίο βασίζεται σε ένα “πλέγμα” 24 έως 32 δορυφόρων που βρίσκονται σε τροχιά γύρω από την Γη, σε τέσσερις επίγειους σταθμούς που ελέγχουν και συντονίζουν τους δορυφόρους και στους δέκτες GPS των χρηστών.<sup>[19]</sup>



Εικόνα 7. Πλέγμα δορυφόρων.<sup>[37]</sup>

Πιο συγκεκριμένα, οι δέκτες GPS των χρηστών υπολογίζουν τη θέση του χρήστη από την ακρίβεια στο χρόνο των σημάτων που στέλνονται από τους δορυφόρους του συστήματος GPS. Κάθε τέτοιος δορυφόρος μεταδίδει συνεχώς σήματα - μηνύματα που περιλαμβάνουν:

- τη χρονική στιγμή που μεταδίδεται το σήμα
- ακριβείς πληροφορίες τροχιάς
- το γενικό σύστημα “υγείας” και σύστημα τροχιών των δορυφόρων στο σύνολό τους.

Ο δέκτης μετράει το χρόνο διέλευσης του κάθε μηνύματος και υπολογίζει την απόσταση από κάθε δορυφόρο. Γεωμετρική trilateration χρησιμοποιείται για να συνδυάσει αυτές τις αποστάσεις από τις θέσεις των δορυφόρων ώστε το σύστημα να καθορίσει την θέση του δέκτη. Η θέση αυτή εμφανίζεται στη συνέχεια σε μια οθόνη εμφανίζοντας το χάρτη της περιοχής ή το γεωγραφικό πλάτος και μήκος. Πληροφορίες υψομέτρου μπορούν να συμπεριληφθούν. Πολλές συσκευές GPS δείχνουν επίσης επιπλέον πληροφορίες, όπως η κατεύθυνση και η ταχύτητα, οι οποίες υπολογίζονται από τις αλλαγές θέσης του κινητού τερματικού που περιέχει το δέκτη.

Τρεις δορυφόροι μπορεί να είναι επαρκείς για να βρεθεί η θέση, επειδή ο διαθέσιμος χώρος έχει τρεις διαστάσεις. Ωστόσο, ακόμη και ένα πολύ μικρό λάθος στο ρολόι, επηρεαζόμενο από την πολύ μεγάλη ταχύτητα του φωτός, που είναι ίση με τη ταχύτητα με την οποία τα δορυφορικά σήματα-διαδίδονται, έχει ως αποτέλεσμα ένα μεγάλο σφάλμα θέσης. Ως εκ τούτου οι δέκτες χρησιμοποιούν τέσσερις ή περισσότερους δορυφόρους για την επίλυση της τοποθεσίας και του χρόνου στο δέκτη.

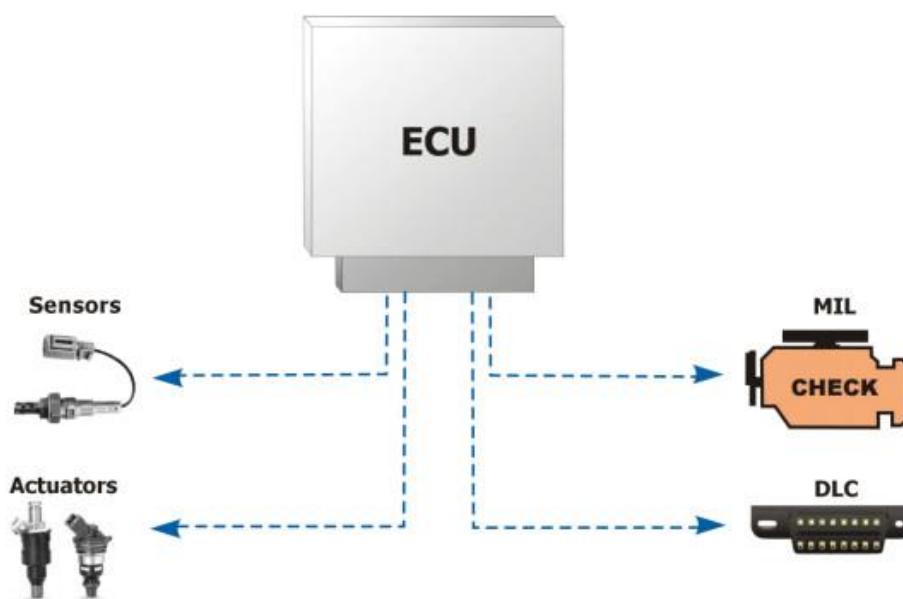
### 2.4.1 Geolocation

Η εφαρμογή που δημιουργήθηκε στηρίζεται στο geolocation, για να εμφανίζει στο χρήστη τις συντεταγμένες που λαμβάνει από το GPS. Ο όρος geolocation είναι ο προσδιορισμός της γεωγραφικής θέσης ενός αντικειμένου, όπως ένα ραντάρ ή ένα κινητό τηλέφωνο στον πραγματικό κόσμο. Το geolocation μπορεί πρακτικά να εκτιμήσει την τοποθεσία ή να αντιπροσωπεύει την ήδη εκτιμημένη θέση. Το geolocation είναι στενά συνδεδεμένο με τα συστήματα εντοπισμού θέσης.

## 2.5 On Board Diagnostics II (OBD-II)

Τα συστήματα διαγνωστικών αυτοκινήτων, ή όπως είναι ευρέως γνωστά On Board Diagnostic Systems (OBD), αντιπροσωπεύουν την ικανότητα του οχήματος να πληροφορεί το μηχανικό ή τον ιδιοκτήτη του, για όλες τις παραμέτρους του και ταυτόχρονα να τον ειδοποιεί για τυχόν βλάβες σε οποιοδήποτε από τα υποσυστήματά του.

Ένα βασικό σύστημα OBD αποτελείται από την Ηλεκτρονική Μονάδα Ελέγχου (ECU), η οποία χρησιμοποιεί δεδομένα που λαμβάνει από διάφορους αισθητήρες για τον έλεγχο όλων των λειτουργιών του ηλεκτρικού συστήματος αλλά και άλλων υποσυστημάτων του οχήματος. Η ενδεικτική λυχνία γνωστή ως MIL ή αλλιώς Check Engine Light, παρέχει μια έγκαιρη προειδοποίηση των δυσλειτουργιών στον ιδιοκτήτη του οχήματος. Ένα σύγχρονο όχημα μπορεί να υποστηρίξει εκατοντάδες παραμέτρους, οι οποίες μπορούν να προσεγγιστούν μέσω του DLC (Diagnostic Link Connector) χρησιμοποιώντας μια συσκευή που ονομάζεται εργαλείο σάρωσης.



Εικόνα 8. Παράδειγμα ECU.<sup>[41]</sup>

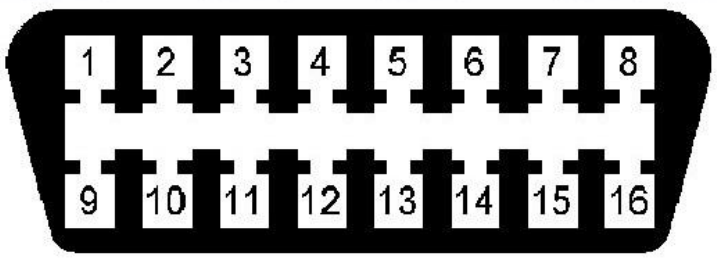
Το είδος και η ποσότητα των διαγνωστικών πληροφοριών που διατίθενται μέσω των OBD έχει διαφοροποιηθεί σε μεγάλο βαθμό από την εμφάνισή τους στις πρώτες τους εκδόσεις τη δεκαετία του 1980. Οι πρώτες εκδόσεις των ενσωματωμένων διαγνωστικών συστημάτων ενεργοποιούσαν κάποια ενδεικτική λυχνία βλάβης, αλλά δεν παρείχαν καμία πληροφορία ως προς τη φύση του προβλήματος. Έκτοτε ο αριθμός των δυνατοτήτων τους αυξάνεται συνεχώς, με αποκορύφωμα τα τηλεμετρικά συστήματα που χρησιμοποιούνται στο μηχανοκίνητο αθλητισμό. Με το πέρασμα των χρόνων και την ραγδαία εξέλιξη των οχημάτων, έγινε επιτακτική η ανάγκη για πιο εξεζητημένα συστήματα διαγνωστικών, κάτι που οδήγησε στην εμφάνιση διαφορετικών πρωτοκόλλων, τα οποία αξιοποιούσαν όλο και μεγαλύτερο μέρος των δυνατοτήτων των οχημάτων. Στις μέρες μας, είναι εφικτός τόσο ο έλεγχος όλων των μερών του κινητήρα, όσο και η διαρκής παρακολούθηση του αμαξώματος και των υπολοίπων συστημάτων που απαρτίζουν το όχημα.

Ακόμα, με τη στροφή των αυτοκινητοβιομηχανιών στις εναλλακτικές μορφές ενέργειας και τις διαρκείς δοκιμές για την εύρεση της βέλτιστης λύσης, προς την ίδια κατεύθυνση στρέφονται και τα OBD Systems, αναπτύσσοντας συνεχώς καινούριες δυνατότητες.

Τα μηχανοκίνητα οχήματα είναι η μεγαλύτερη πηγή τοξικών ατμοσφαιρικών ρύπων, όπως για παράδειγμα της αιθαλομίχλης. Τα σύγχρονα οχήματα μπορεί να είναι πιο "καθαρά", λόγω της τεχνολογικής προόδου και τις στρατηγικές ελέγχου εκπομπών καυσαερίων, αλλά οι εκπομπές ρύπων μειώνονται μόνον όταν λειτουργούν ομαλά όλα τα συστήματα του οχήματος. Όταν ένας κινητήρας δεν λειτουργεί σωστά, η απόδοση του μειώνεται, τα καύσιμα σπαταλώνονται και οι εκπομπές καυσαερίων από την εξάτμιση αυξάνονται δραματικά. Τίθεται, επίσης, και θέμα ασφάλειας του οχήματος, ειδικά σε υψηλές ταχύτητες. Το σύστημα OBD (On Board Diagnostics) μπορεί να ανιχνεύσει και να κατευθύνει τον εκπαιδευμένο τεχνικό, ακόμα και εάν το σφάλμα που υπάρχει δεν είναι η πηγή του προβλήματος. Πολλά από αυτά τα προβλήματα δεν είναι εύκολο να εντοπιστούν, επειδή προέρχονται από το ηλεκτρικό σύστημα ή είναι χημικής φύσης.<sup>[18]</sup>

Το OBD προσφέρει τη δυνατότητα εντοπισμού του προβλήματος, της διόρθωσής του και της κατάλληλης συντήρησης, πριν εκδηλωθούν σοβαρότερα και δαπανηρότερα προβλήματα.

Στην παρούσα διπλωματική εργασία θα ασχοληθούμε με την δεύτερη έκδοση των ενσωματωμένων διαγνωστικών συστημάτων (OBD-II), η οποία αποτελεί βελτίωση των εκδόσεων OBD-I και OBD-1.5.



PIN	DESCRIPTION	PIN	DESCRIPTION
1	Vendor Option	9	Vendor Option
2	J1850 Bus +	10	j1850 BUS
3	Vendor Option	11	Vendor Option
4	Chassis Ground	12	Vendor Option
5	Signal Ground	13	Vendor Option
6	CAN (J-2234) High	14	CAN (J-2234) Low
7	ISO 9141-2 K-Line	15	ISO 9141-2 Low
8	Vendor Option	16	Battery Power

Εικόνα 9. Εικονίδιο αντιστοιχίας pin θύρας OBD.<sup>[38]</sup>

### 2.5.1 Υποστηριζόμενα Πρωτόκολλα OBD-II

Πέντε είναι τα διαθέσιμα πρωτόκολλα στο OBD-II και, εκτός από ειδικές περιπτώσεις, κάθε όχημα υποστηρίζει ένα από αυτά. Ο συνηθέστερος τρόπος για την αναγνώριση του υποστηριζόμενου πρωτοκόλλου είναι ο εντοπισμός των pins που υπάρχουν στον υποδοχέα. Σύμφωνα με τον πιο πρόσφατο κανονισμό ο υποδοχέας πρέπει να βρίσκεται σε ακτίνα 60 εκατοστών από το τιμόνι του οχήματος, ενώ παλαιότερα βρισκόταν κάτω από το καπό του αυτοκινήτου.

Τα διαθέσιμα πρωτόκολλα είναι:

1. SAE J1850 PWM (pulse-width modulation)
2. SAE J1850 VPW (variable pulse width)
3. ISO 9141-2
4. ISO 14230 KWP2000 (Keyword Protocol 2000)
5. ISO 15765 CAN

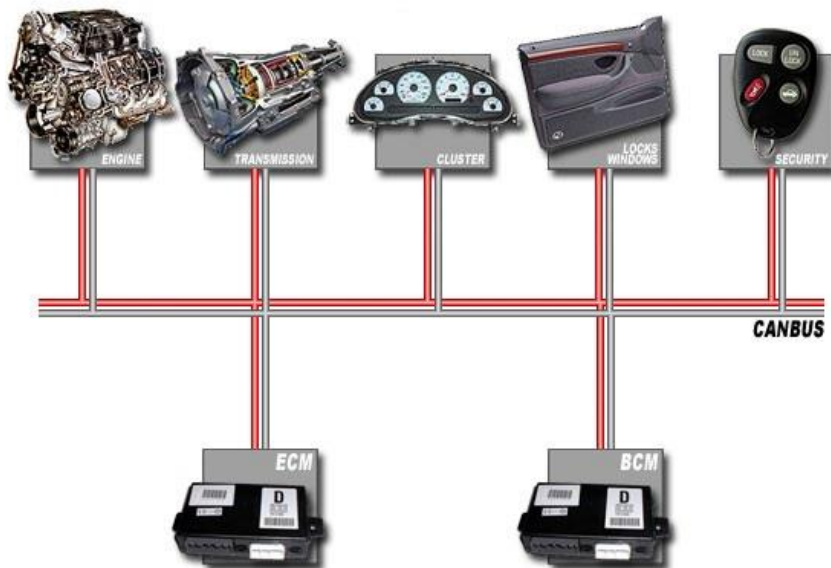
Η παρούσα διπλωματική επικεντρώνεται στο πρωτόκολλο CAN, για τους λόγους που εξηγούνται στην συνέχεια.

### 2.5.1.1 CAN Bus

Το CAN Bus είναι το πλέον σύγχρονο δίκτυο μεταφοράς δεδομένων. Είναι ειδικά σχεδιασμένο για την αυτοκινητοβιομηχανία. Τα πολλά όμως πλεονεκτήματά του έχουν οδηγήσει στην ανάπτυξη συστημάτων βασισμένων στο CAN και σε άλλους τομείς όπως η βιομηχανία, η αεροναυπηγική, η ναυπηγική και η βιοϊατρική.

Το CAN αναπτύχθηκε με σκοπό να αντικαταστήσει την πολύπλοκη καλωδίωση των οχημάτων με ένα σύστημα δύο καλωδίων (dual wire bus). Η Bosch GmbH, το 1983, ήταν η πρώτη εταιρεία που επιχείρησε να αναπτύξει ένα τέτοιο σύστημα, ενώ η επίσημη παγκόσμια αναγνώριση του CAN, σαν πρωτόκολλο επικοινωνίας, ήρθε το 1986. Από τότε, η δημοτικότητά του αυξήθηκε ραγδαία και το 1993 θεσπίστηκε ως το παγκόσμιο πρωτόκολλο επικοινωνίας στο χώρο της αυτοκινητοβιομηχανίας, γνωστό και ως ISO 11898.

Η βασική λειτουργία του διαύλου CAN είναι η διασύνδεση των διαφορετικών ηλεκτρονικών μονάδων του αυτοκινήτου. Οι πιο σημαντικές Ηλεκτρονικές Μονάδες Ελέγχου (ECU) είναι το Engine Control Module, το Transmission Control Module και το ABS Module αλλά ακολουθούνται από ένα πολύ μεγάλο αριθμό ECUs που αντιστοιχεί στο κάθε υποσύστημα του οχήματος. Τα σύγχρονα αυτοκίνητα έχουν περισσότερες από εβδομήντα ηλεκτρονικές μονάδες ελέγχου.



Εικόνα 10. Παράδειγμα διασύνδεσης ηλεκτρικών μονάδων οχήματος με CAN δίκτυο.<sup>[39]</sup>

Το CAN Bus έχει σημαντικά πλεονεκτήματα σε σχέση με τα υπόλοιπα πρότυπα, όπως είναι το χαμηλό κόστος δικτύου, η μείωση του βάρους του οχήματος, η αυτόνομη επικοινωνία, η προτεραιότητα μηνυμάτων, αλλά και ο εντοπισμός λαθών. Κύρια εξ' αυτών είναι η

μεγαλύτερη ταχύτητα ροής των δεδομένων, η αυτονομία των συνδεδεμένων υποσυστημάτων, το μειωμένο κόστος και η απλή αρχιτεκτονική του δικτύου. Επίσης από το 2008 έχει καθιερωθεί σαν πρωτόκολλο μεταφοράς δεδομένων στην αυτοκινητοβιομηχανία.

### 2.5.2 OBD-II Modes

Στο πρότυπο SAE J1979 καθορίστηκαν ακόμα δέκα λειτουργίες (Modes Of Operation). Κάθε Mode επιτελεί διαφορετική λειτουργία, συνεπώς έχει διαφορετικούς υποστηριζόμενους κωδικούς. Ο διαχωρισμός αυτός έγινε κυρίως για λόγους κατηγοριοποίησης των λειτουργιών ενός διαγνωστικού συστήματος αυτοκινήτου. Είναι σημαντικό να αναφερθεί ότι η κωδικοποίηση τόσο των PIDs όσο και των Modes γίνεται σύμφωνα με το δεκαεξαδικό σύστημα αρίθμησης.

Λόγω του ότι στην παρούσα εργασία χρησιμοποιήθηκαν τα Modes ένα και τέσσερα, αναλύονται μόνο αυτά.

- Η πρώτη λειτουργία (Mode 1) πρόκειται για τη βασική λειτουργία του διαγνωστικού εργαλείου. Μέσω του Mode 1, ο χρήστης έχει τη δυνατότητα να ζητήσει οποιαδήποτε πληροφορία για το όχημα σε πραγματικό χρόνο, κάτι που σημαίνει ότι η συγκεκριμένη λειτουργία μπορεί να χρησιμοποιηθεί και για τη γραφική απεικόνιση των μεταβαλλόμενων μεγεθών.
- Το Mode 4 επιτελεί στην ουσία τη λειτουργία της εκκαθάρισης των κωδικών σφαλμάτων που είναι καταχωρημένοι στην ηλεκτρονική μονάδα ελέγχου (ECU) του οχήματος, καθώς επίσης απενεργοποιεί την ένδειξη για τον έλεγχο της μηχανής (check engine light) ή αλλιώς MIL (malfunction indicator lamp).

Σημειώνεται ότι το κάθε Mode έχει ένα δεκαεξαδικό κωδικό που χρησιμοποιείται από το λογισμικό του διαγνωστικού εργαλείου για την αναγνώρισή του από το σύστημα.

### 2.5.3 OBD-II PIDs (On-board diagnostics Parameter IDs)

Σε προηγούμενα κεφάλαια αναλύθηκαν το υπόβαθρο και οι δυνατότητες του OBD-II και του πρωτοκόλλου CAN ξεχωριστά. Σε αυτήν την παράγραφο, θα αναπτυχθούν οι λειτουργίες και οι κωδικοί που χρησιμοποιούνται από το OBD-II και θα επεξηγηθεί η διαδικασία που ακολουθείται από το χειριστή ενός συστήματος διαγνωστικών αυτοκινήτου.

Η τυποποίηση των κωδικών που χρησιμοποιούνται σε συστήματα διαγνωστικών αυτοκινήτου έγινε το 1979, από την Ένωση Μηχανικών Αυτοκινητοβιομηχανίας (Society of Automotive Engineers – SAE) και είναι γνωστή ως πρότυπο SAE J1979. Σύμφωνα με τη συμφωνία αυτή, θεσπίζονται συγκεκριμένοι κωδικοί για κάθε λειτουργία του οχήματος και ονομάζονται Parameter IDs (PIDs). Το παράρτημα της εργασίας περιέχει μια λίστα από τυποποιημένους PIDs που όλες οι βιομηχανίες είναι ακόμα και σήμερα υποχρεωμένες να την εφαρμόζουν στα διαγνωστικά συστήματά τους.

Η διαδικασία που ακολουθείται για το χειρισμό του διαγνωστικού συστήματος αποτελείται από τα εξής βήματα:

- 1 Αρχικά ο χειριστής του διαγνωστικού εργαλείου, που στην περίπτωση μας είναι η εφαρμογή που δημιουργήσαμε, εισάγει τον PID που τον ενδιαφέρει.
- 2 Το διαγνωστικό εργαλείο στέλνει το μήνυμα στο δίκτυο του αυτοκινήτου, σύμφωνα με το κατάλληλο πρωτόκολλο.
- 3 Το σύστημα αναγνωρίζει τον PID και επιστρέφει το ζητούμενο αποτέλεσμα.
- 4 Το διαγνωστικό εργαλείο διαβάζει την απάντηση και τη δείχνει στο χειριστή.

Για παράδειγμα η διαδικασία που απαιτείται για την λήψη των ανά λεπτό στροφών του κινητήρα του οχήματος παρουσιάζεται στην συνέχεια.

Αρχικά η πρόσβαση στα δεδομένα πραγματικού χρόνου γίνεται μέσω της λειτουργίας 1 που αναφέραμε νωρίτερα. Κάθε στοιχείο έχει έναν μοναδικό κωδικό(PID). Ο κωδικός στην περίπτωση των στροφών είναι ο δεκαεξαδικός “0C”. Έτσι πρέπει να στείλουμε την γραμματοσειρά “01 0C” για να λάβουμε απάντηση για την συγκεκριμένη τιμή.

Η απάντηση που λαμβάνουμε περιέχει 2 byte τα οποία αφορούν το αίτημα που κάναμε και τον κωδικό του στοιχείου (41 0C) και ακόμα δύο που αφορούν την κωδικοποιημένη τιμή του στοιχείου.

Αν δηλαδή λάβουμε σαν απάντηση την γραμματοσειρά 41 0C 0F A0, μας επιστρέφεται σαν αποτέλεσμα της τιμής που ζητήσαμε το 0F A0. Εισάγοντας τα δύο byte στην αντίστοιχη εξίσωση,

$$(A * 256 + B)/4,$$

όπου A η τιμή του 0F στο δεκαδικό σύστημα αρίθμησης και B η αντίστοιχη τιμή του A0, η τελική τιμή των ανά λεπτό στροφών του κινητήρα είναι 1000.

## 2.6 ELM327

Ο ELM327 Programmed Microcontroller είναι ένα μικρό σε μέγεθος και πολύ χαμηλού κόστους εργαλείο. Ο μικροελεγκτής ELM327 λειτουργεί ως γέφυρα μεταξύ του εγκεφάλου (ECU) του αυτοκινήτου σας και του υπολογιστή σας (Laptop, Pc κλπ.) ή της φορητής συσκευής (Android - iOS). Ο ELM327 είναι ικανός να επικοινωνεί με το OBDII σύστημα με το όχημα και στη συνέχεια να εκτελεί μετεγκατάσταση των δεδομένων μέσω USB καλωδίου, Wi-Fi ή Bluetooth, ανάλογα με τη συγκεκριμένη εφαρμογή. Υποστηρίζει έναν αριθμό διαφορετικών πρωτοκόλλων, καθώς και τη δυνατότητα να επικοινωνεί με οποιοδήποτε όχημα με OBDII υποδοχή.



Εικόνα 11. Φωτογραφία ενός μικροελεγκτή ELM327.<sup>[40]</sup>

Οι συσκευές ELM327 μπορεί να συνδεθούν με υπολογιστές, Smartphones, tablets, και άλλες συσκευές μέσω ενός αριθμού διαφορετικών μέσων, όπως τα παρακάτω:

- USB Καλώδιο
- Wi-Fi
- Bluetooth
- Συνδυασμό USB Καλώδιο και Wi-Fi

Στην παρούσα διπλωματική χρησιμοποιήσαμε το Bluetooth ως μέσο σύνδεσης με την κινητή συσκευή μας με λογισμικό Android. Ο ELM327 μπορεί να παρέχει πρόσβαση στους κωδικούς βλάβης και επίσης να εμφανίζει τα PIDs. Δεδομένου ότι η επικοινωνία είναι αμφίδρομη, η συσκευή ELM327 μπορεί επίσης να επιτρέψει τον καθαρισμό των κωδικών βλάβης, αφού προηγουμένως έχει επιδιορθωθεί το υφιστάμενο πρόβλημα. Οι ακριβείς ενέργειες που μπορούν να εκτελεστούν, εξαρτώνται από την συγκεκριμένη συσκευή ELM327, το λογισμικό που χρησιμοποιείται και την εταιρεία και το μοντέλο του οχήματος. Επίσης δίνεται η δυνατότητα προβολής - ανάγνωσης στην οθόνη μιας συσκευής μεγάλου αριθμού δεδομένων που σχετίζονται με την λειτουργία του οχήματος και ιδίως του κινητήρα.

#### Πλεονεκτήματα ELM327

- Εύκολο στη χρήση και σύνδεση με συσκευές Android.
- Διαβάζει και καθαρίζει τους κωδικούς προβλήματος βλάβης.
- Παρέχει πρόσβαση στα δεδομένα αισθητήρων σε πραγματικό χρόνο.
- Πολύ χαμηλή τιμή αγοράς, συχνά κάτω από 15 ευρώ.

#### Μειονεκτήματα ELM327

- Περιορισμένη συμβατότητα σε ορισμένα οχήματα.
- Πιο δύσκολη η σύνδεση για χρήση με συσκευές iOS.
- Αν παραμείνει μόνιμα συνδεδεμένο στο αυτοκίνητο αυξάνει την κατανάλωση ενέργειας.

## Κεφάλαιο 3

### Εργαλεία

#### 3.1 Εργαλεία Ανάπτυξης

Όπως έχουμε δει μέχρι τώρα, οι εφαρμογές για το λειτουργικό σύστημα Android γράφονται στην γλώσσα προγραμματισμού Java. Απαραίτητο είναι λοιπόν για όποιον επιθυμεί να αναπτύξει εφαρμογές για Android να διαθέτει λειτουργικό σύστημα Windows, Linux ή Mac OS. Επιπρόσθετα, χρειάζονται τα παρακάτω εργαλεία ώστε να πραγματοποιηθεί η διαδικασία ανάπτυξης:

1. Java Development Kit
2. Eclipse IDE για Java
3. Android SDK
4. ADT (Android Development Tools) Plug-in για το Eclipse
5. Adobe Photoshop CC 2014

Τα τέσσερα πρώτα απαραίτητα εργαλεία είναι διαθέσιμα δωρεάν και η εγκατάστασή τους είναι αρκετά εύκολη, αλλά η χρονική διάρκεια της εγκατάστασης εξαρτάται σημαντικά από την ταχύτητα της σύνδεσης στο διαδίκτυο.

##### 3.1.1 Java Development Kit

Για την συγγραφή οποιασδήποτε εφαρμογής Java είναι απαραίτητη η ύπαρξη του Java Development Kit (JDK). Το JDK περιλαμβάνει ένα σύνολο εργαλείων ανάπτυξης προγραμμάτων για την γλώσσα προγραμματισμού Java όπως είναι ο `java` (φορτωτής Java εφαρμογών), ο `javac` (ο μεταφραστής του Java κώδικα σε Java bytecodes), ο `javah` (δημιουργός C stubs για συγγραφή native μεθόδων), ο `jar` (ο γνωστός Java archiver) και άλλα. Το JDK είναι το υποσύνολο του Java SDK που είναι απαραίτητο για τον προγραμματισμό και την εκτέλεση εφαρμογών σε Java.

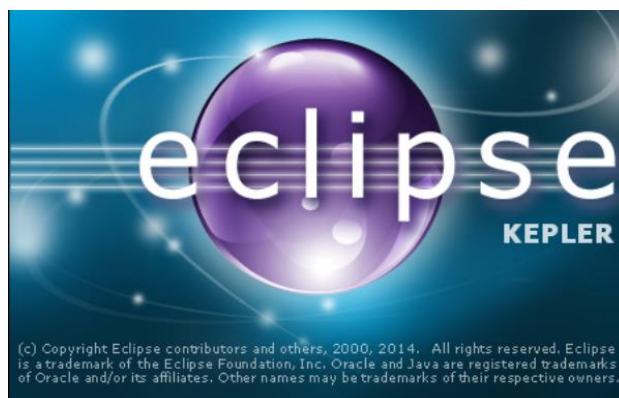


Εικόνα 12. Εικονίδιο Oracle Java Development Kit 8.<sup>[42]</sup>

Το Java SDK περιλαμβάνει επιπλέον στοιχεία όπως application servers, επιπλέον debuggers και documentation. Για την ανάπτυξη εφαρμογών για το λειτουργικό σύστημα Android, το JDK είναι αρκετό. Η έκδοση που χρησιμοποιήσαμε είναι η 8 (update 5) και το αρχείο εγκατάστασης είναι διαθέσιμο από τον ισότοπο της Oracle.

### 3.1.2 Eclipse IDE για Java

Το Eclipse IDE (Integrated Development Environment) είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών. Κατά κύριο λόγο είναι ανεπτυγμένο με χρήση της γλώσσας προγραμματισμού JAVA. Μπορεί να χρησιμοποιηθεί για την ανάπτυξη εφαρμογών σε JAVA, ενώ με την βοήθεια διαφόρων προσθέτων που παρέχει, είναι δυνατή η ανάπτυξη εφαρμογών και σε άλλες γλώσσες.



Εικόνα 13. Eclipse Kepler.<sup>[43]</sup>

Το Eclipse παρέχει ένα περιβάλλον για την οργάνωση του κώδικα ενός ολοκληρωμένου project. Λειτουργίες όπως, διατήρηση εκδόσεων, διαμοιρασμός αλλαγών στον κώδικα μεταξύ των διαφόρων μελών της ομάδας ανάπτυξης, εικονικοί εξυπηρετητές για έλεγχο ορθότητας λειτουργίας web εφαρμογών και άλλες, παρέχονται εντός του ίδιου του περιβάλλοντος.

Επιπλέον, με την υποστήριξη των plug-in το Eclipse μπορεί να επεκτείνει τις λειτουργίες που προσφέρει ακόμα παραπέρα. Χαρακτηριστικό παράδειγμα είναι το ADT Plug-in για την ανάπτυξη εφαρμογών Android, που θα δούμε παρακάτω. Η εγκατάσταση του Eclipse είναι πολύ απλή υπόθεση και μπορούμε να το λάβουμε από τον ιστότοπο [www.eclipse.org](http://www.eclipse.org). Η έκδοσή που χρησιμοποιήσαμε είναι η Kepler SR2.

### 3.1.3 Android SDK

Το Android SDK αποτελεί απαραίτητο εργαλείο για την ανάπτυξη εφαρμογών για Android. Το Android SDK περιλαμβάνει τις απαραίτητες βιβλιοθήκες για την ανάπτυξη εφαρμογών Android, παραδείγματα εφαρμογών με τον πηγαίο κώδικά τους, τον εξομοιωτή συσκευών καθώς και άλλες βιβλιοθήκες που έχουν να κάνουν με την τιμολόγηση των εφαρμογών και την πώλησή τους στο Google play. Το SDK μπορεί να το προμηθευτεί κανείς από τον ιστότοπο της Google developer.google.com.

### 3.1.4 ADT (Android Development Tools) Plug-in για το Eclipse

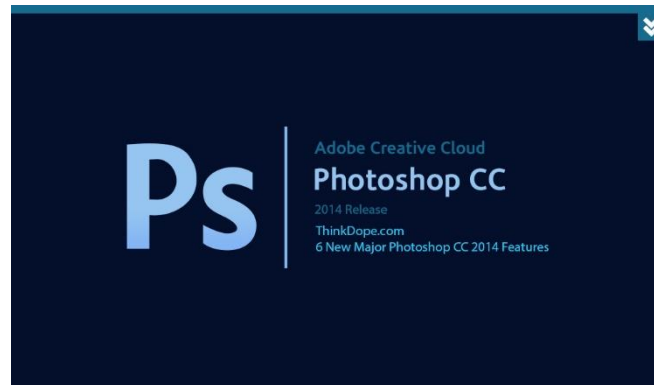
Χάρη στα Plug-in το Eclipse μπορεί να επεκτείνει τις λειτουργικότητες που προσφέρει. Εκμεταλλευόμενη το γεγονός αυτό η Google, ανέπτυξε το ADT (Android Development Tools) Plugin για το Eclipse. Με το plug-in αυτό το Eclipse προσφέρει την δυνατότητα σχεδίασης διεπαφών χρήστη για τις εφαρμογές Android με γραφικό τρόπο. Επεκτείνει τις επιλογές των μενού εισαγωγής του Eclipse έτσι ώστε, για παράδειγμα, να μπορεί κανείς να δημιουργήσει ένα καινούριο project Android με την σωστή δομή και αρχεία αυτοματοποιημένα εντός του περιβάλλοντος. Επιπλέον, παρέχεται ειδικός editor για τα διάφορα XML αρχεία, που περιέχονται σε ένα project Android καθώς και εργαλεία για την παραγωγή των τελικών πακέτων της εφαρμογής.



Εικόνα 14. Εικονίδιο για το Android Development Tools.<sup>[44]</sup>

### 3.1.5 Adobe Photoshop CC 2014

Το Adobe Photoshop, ή απλά Photoshop, είναι ένα πρόγραμμα επεξεργασίας γραφικών που αναπτύχθηκε και κυκλοφόρησε από την Adobe Systems. Αυτή τη στιγμή αποτελεί ηγέτη της αγοράς των προγραμμάτων επεξεργασίας εικόνων. Στην παρούσα διπλωματική ήταν απαραίτητο και χρησιμοποιήθηκε για την δημιουργία του εικονιδίου της εφαρμογής καθώς και για τα κυρίως μενού της.



Εικόνα 15. Εικονίδιο Adobe Photoshop.<sup>[45]</sup>

## Κεφάλαιο 4

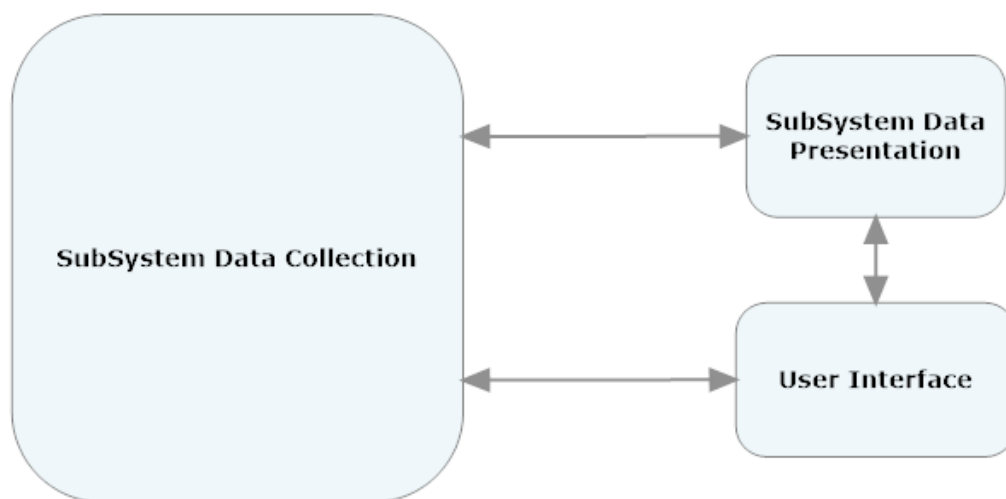
### Μοντελοποίηση και Αρχιτεκτονική Συστήματος

Σε αυτό το κεφάλαιο αναλύονται σε βάθος η μοντελοποίηση, η αρχιτεκτονική, καθώς επίσης και οι κανόνες που ακολουθήθηκαν ώστε να σχεδιαστεί και να υλοποιηθεί το σύστημά μας. Όλα τα διαγράμματα τα οποία παραθέτονται από το κεφάλαιο και μετά αποτελούν κομμάτι της εργασίας και μόνο όσα έχουν αναφορά στην πηγή τα βρήκαμε έτοιμα.

#### 4.1 Μοντελοποίηση Συστήματος

Η σχεδίαση μας αποτελείται από τρία επιμέρους υποσυστήματα, όπως φαίνεται στην Εικόνα 16. Αυτά είναι:

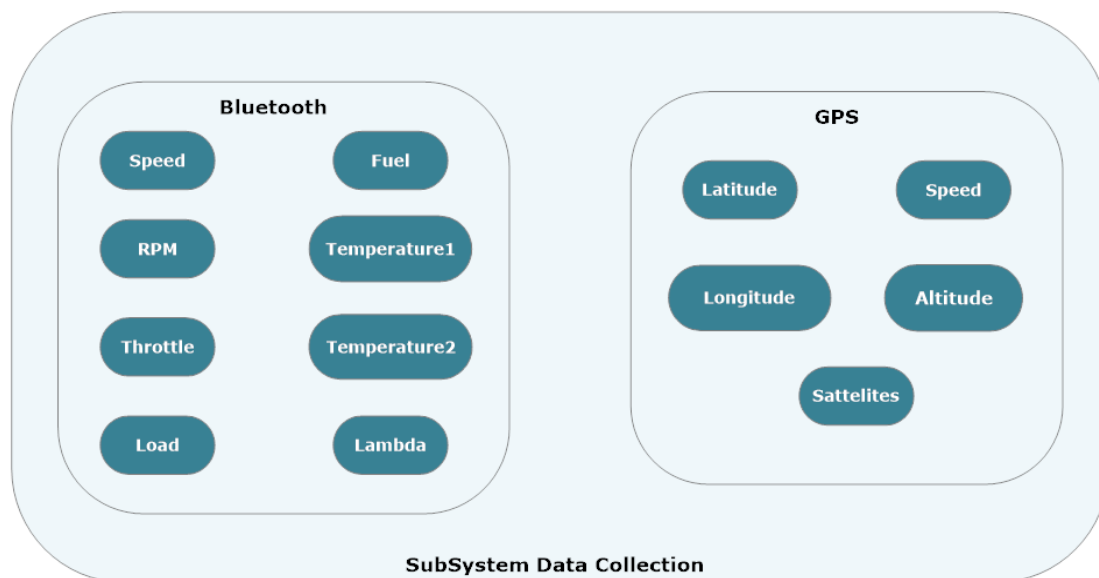
- Η διεπαφή χρήστη
- Υποσύστημα παρουσίασης δεδομένων
- Υποσύστημα συλλογής δεδομένων



Εικόνα 16. Μοντελοποίηση Συστήματος δείχνοντας τα επιμέρους Υποσυστήματα.

Καθένα από τα υποσυστήματα που παρουσιάζονται εξυπηρετούν συγκεκριμένους σκοπούς στο σύστημα μας. Στο τμήμα της διεπαφής χρήστη όπως είναι ευνόητο παρουσιάζονται τα διαθέσιμα δεδομένα στους χρήστες της εφαρμογής μέσω του γραφικού περιβάλλοντος (graphical user interface). Στο υποσύστημα της παρουσίασης δεδομένων, αφού

τα απαραίτητα στοιχεία έχουν συλλεχθεί, επεξεργάζονται κατάλληλα και στη συνέχεια είναι διαθέσιμα για την παρουσίασή τους στον χρήστη. Το υποσύστημα συλλογής δεδομένων είναι υπεύθυνο για τη συλλογή των απαραίτητων για την εφαρμογή δεδομένων. Τα στοιχεία, τα οποία συλλέγει αναλυτικά για το κάθε κομμάτι της υλοποίησης ( Bluetooth, GPS) φαίνονται στην εικόνα 17.



Εικόνα 17. Υποσύστημα Συλλογής Δεδομένων.

Οι ρυθμοί συλλογής δεδομένων διαφοροποιούνται ανάλογα με το αυτοκίνητο, τη συσκευή Android αλλά και τον μικροελεγκτή που χρησιμοποιούνται. Για ένα στοιχείο του οποίου οι τιμές αλλάζουν συνεχώς, ένας ιδανικός ρυθμός δειγματοληψίας θα ήταν 5 έως 10 δείγματα το δευτερόλεπτο. Αντίθετα για κάποιο στοιχείο, το οποίο δε μεταβάλλεται αρκετά συχνά, δε χρειάζεται να υπάρχει τόσο μεγάλος ρυθμός δειγματοληψίας.

#### 4.1.1 Επιλογή Bluetooth

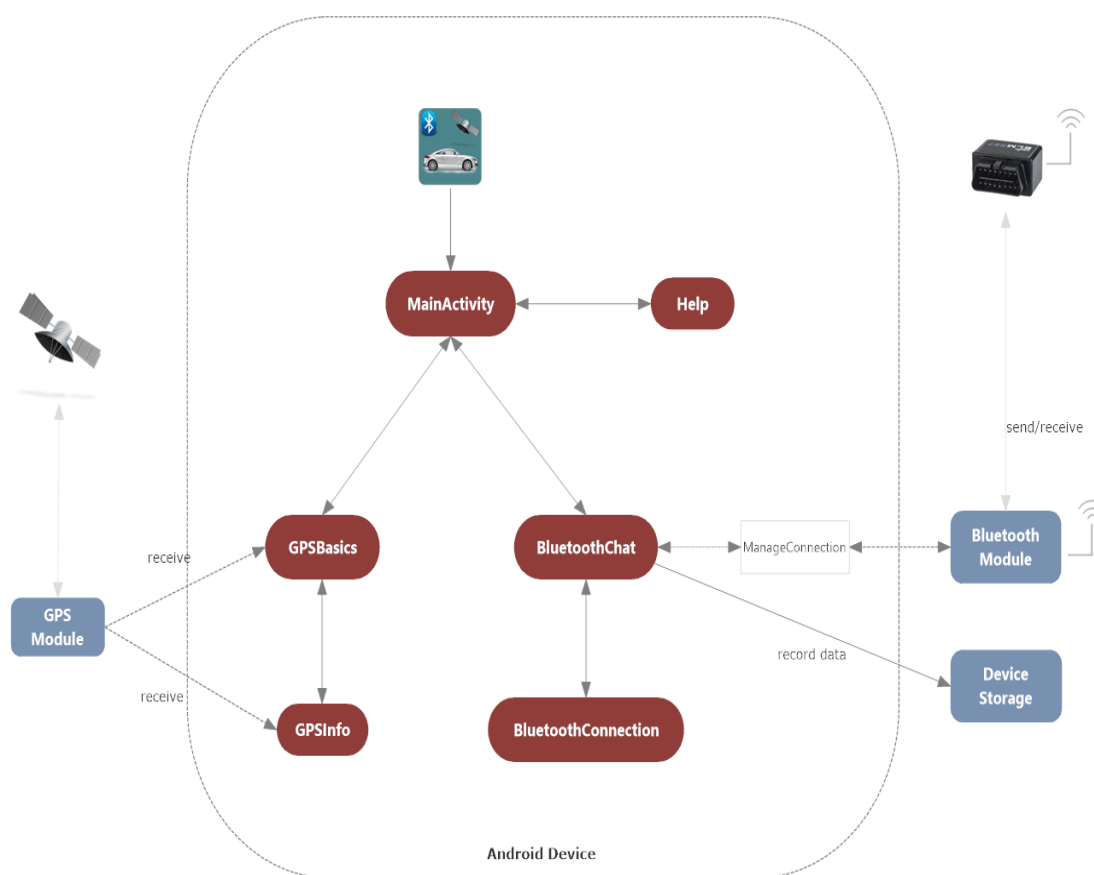
Από τις διαθέσιμες τεχνολογίες επιλέχθηκε η τεχνολογία του Bluetooth για την επικοινωνία και την ανταλλαγή δεδομένων μεταξύ της συσκευής Android και του αυτοκινήτου. Η επιλογή μας αυτή αποφασίστηκε ύστερα από αρκετή σκέψη και μελέτη.

Καταρχάς η τεχνολογία που χρησιμοποιήθηκε, βασίζεται στην ασύρματη επικοινωνία μεταφοράς δεδομένων, πράγμα που μας απαλλάσσει από την χρήση καλωδίων. Κάτι τέτοιο κατά τη διάρκεια χρήσης της εφαρμογής, ειδικά σε περιπτώσεις οδήγησης, θα ήταν αρκετά άβολο έως και επικίνδυνο για τον χρήστη της. Η χρήση καλωδίων καθιστά πιο δύσκολη την χρήση της συσκευής από τον χρήστη, καθώς ακόμα δεν του επιτρέπει την παράλληλη φόρτισή της, αφού η αντίστοιχη θύρα είναι δεσμευμένη. Αξίζει να αναφέρουμε πως κάτι τέτοιο θα ήταν

ιδιαίτερα επιζήμιο, διότι η εφαρμογή μας είναι αρκετά απαιτητική σε θέματα αυτονομίας και συνεπώς η παράλληλη φόρτισή της ορισμένες φορές είναι αναγκαία για την χρήση της. Ακόμα το Bluetooth προτιμήθηκε σε σχέση με το Wi-Fi, διότι είναι λιγότερο απαιτητικό σε θέματα κατανάλωσης πόρων της συσκευής και επίσης είναι πιο δύσκολη η παραβίαση της επικοινωνίας μεταξύ των συνδεδεμένων συσκευών.

Τέλος, το μόνο αρνητικό του Bluetooth σε σχέση με το καλώδιο usb και το Wi-Fi είναι ότι δεν προσφέρει τις ίδιες ταχύτητες στη μεταφορά δεδομένων, πράγμα το οποίο δε λειτουργεί αρνητικά στην περίπτωση μας, καθώς ο όγκος των δεδομένων που μεταφέρονται είναι πολύ μικρός.

Στην εικόνα 18 βλέπουμε μια απεικόνιση του συστήματος μας και πως αυτό συνδέεται με την τεχνολογία που χρησιμοποιήθηκε.



Εικόνα 18. Απεικόνιση Συστήματος και Τεχνολογίας.

### 4.1.2 Δεδομένα Πραγματικού Χρόνου

Για το κάθε τμήμα του υποσυστήματος συλλογής δεδομένων λαμβάνουμε κάποια δεδομένα, τα οποία αφού τα επεξεργαστούμε τα εμφανίζουμε στην οθόνη του χρήστη.

Αναλυτικά για το κομμάτι που αφορά το Bluetooth τα δεδομένα αυτά είναι:

- Η ταχύτητα με την οποία κινείται το αυτοκίνητο (Vehicle speed). Για τον υπολογισμό της ταχύτητας, αρχικά στέλνεται ο κωδικός 01 0D και μας επιστρέφεται ως απάντηση για την τιμή της αυτή ένα byte. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και την εμφανίζουμε απευθείας στην οθόνη χωρίς να την εισάγουμε σε κάποια εξίσωση στην περίπτωση της ταχύτητας.
- Οι ανά λεπτό στροφές του κινητήρα (Engine RPM). Για τον υπολογισμό των στροφών του κινητήρα, στέλνεται ο κωδικός 01 0C και μας επιστρέφονται ως απάντηση για την τιμή της αυτή δύο byte, αφού η αναπαράσταση της τιμής των στροφών του κινητήρα ανά λεπτό δεν χωράει να αναπαρασταθεί με λιγότερα . Μετατρέπουμε τις δύο τιμές που λάβαμε στο δεκαδικό σύστημα αρίθμησης και τις εισάγουμε στην παρακάτω εξίσωση

$$((A * 256) + B)/4,$$

όπου A η πρώτη κατά σειρά τιμή που μας επιστρέφεται και B η δεύτερη, και εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

- Το επί τοις εκατό ποσοστό της θέσης πεταλούδας γκαζιού (Throttle position). Για τον υπολογισμό της θέσης πεταλούδας γκαζιού, στέλνεται ο κωδικός 01 11 και μας επιστρέφεται ως απάντηση για την τιμή αυτή ένα byte. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και βάζοντάς την στην παρακάτω εξίσωση

$$(A * 100)/255,$$

όπου A η τιμή που μας επιστρέφεται, εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

- Το επί τοις εκατό ποσοστό φορτίο του κινητήρα (Calculated engine load value). Για τον υπολογισμό του φορτίου του κινητήρα, στέλνεται ο κωδικός 01 04 και μας επιστρέφεται ως απάντηση για την τιμή του ένα byte. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και βάζοντας της στην παρακάτω εξίσωση

$$(A * 100)/255,$$

όπου A η τιμή που μας επιστρέφεται, εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

- Η θερμοκρασία αντιψυκτικού της μηχανής (Engine coolant temperature). Για τον υπολογισμό της θερμοκρασίας αντιψυκτικού της μηχανής, στέλνεται ο κωδικός 01 05 και μας επιστρέφεται ως απάντηση για την τιμή της ένα byte. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και βάζοντας της στην παρακάτω εξίσωση

$$A - 40,$$

όπου A η τιμή που μας επιστρέφεται, εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

- Η θερμοκρασία του αέρα εισαγωγής στον κινητήρα (Intake air temperature). Για τον υπολογισμό της θερμοκρασίας του αέρα εισαγωγής στον κινητήρα, στέλνεται ο κωδικός 01 0F και μας επιστρέφεται ως απάντηση για την τιμή της ένα byte. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και βάζοντας την στην παρακάτω εξίσωση

$$A - 40,$$

όπου A η τιμή που μας επιστρέφεται, εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

- Η στιγμιαία κατανάλωση καυσίμου (Fuel Consumption). Ο υπολογισμός της στιγμιαίας κατανάλωσης καυσίμου είναι κάπως πιο περίπλοκος. Αρχικά στέλνεται ο κωδικός 01 10, ο οποίος αφορά την τιμή του αισθητήρα μέτρησης της ροής μάζας αέρα στην εισαγωγή ενός κινητήρα εσωτερικής καύσης (MAF air flow rate) και μας επιστρέφονται ως απάντηση δύο byte αυτήν την φορά σαν αποτέλεσμα. Μετατρέπουμε τις δύο τιμές που λάβαμε στο δεκαδικό σύστημα αρίθμησης και τις εισάγουμε στην παρακάτω εξίσωση

$$((A * 256) + B) / 100$$

όπου A η πρώτη κατά σειρά τιμή που μας επιστρέφεται και B η δεύτερη. Το αποτέλεσμα που λαμβάνουμε (MAF) είναι σε grams/sec. Τώρα παίρνοντας αυτό το αποτέλεσμα και βάζοντας το σε μια δεύτερη εξίσωση,

$$\frac{VSS * 710.7}{(MAF * 100)},$$

όπου VSS η στιγμιαία ταχύτητα, μπορούμε να υπολογίσουμε τα απαιτούμενο MPG (Miles per gallon). Τέλος για την μετατροπή της στιγμιαίας κατανάλωσης σε λίτρα ανά εκατό χιλιόμετρα διαιρούμε με την παρακάτω εξίσωση και εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

$$235.214/MPG$$

- Η τιμή του αισθητήρα λ, ο οποίος φροντίζει ώστε τα ποσοστά των ρύπων στα καυσαέρια να παραμένουν κάτω από τα επιτρεπτά όρια τιμών. Για τον υπολογισμό της τιμής του αισθητήρα λ, στέλνεται ο κωδικός 01 34 και μας επιστρέφονται ως απάντηση τέσσερα byte αυτήν την φορά σαν αποτέλεσμα, όμως τα δύο πρώτα είναι αυτά που χρειαζόμαστε. Μετατρέπουμε τις δύο τιμές που λάβαμε στο δεκαδικό σύστημα αρίθμησης και τις εισάγουμε στην παρακάτω εξίσωση

$$((A * 256) + B) / 32768,$$

όπου A η πρώτη κατά σειρά τιμή που μας επιστρέφεται και B η δεύτερη, και εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

Σε ότι αφορά τώρα τα δεδομένα που λαμβάνονται με τη βοήθεια του δέκτη GPS της συσκευής αυτά είναι:

- Η ταχύτητα. Μας παρέχεται κάθε χρονική στιγμή από το GPS και την εμφανίζουμε σε χιλιόμετρα ανά δευτερόλεπτο.
- Η στιγμιαία επιτάχυνση. Υπολογίζεται από την διαφορά της ταχύτητας δύο διαφορετικών χρονικών στιγμών και για την εμφάνισή της, σε μέτρα ανά δευτερόλεπτο εις στο τετράγωνο, διαιρούμε την προηγούμενη διαφορά με το 3,6.
- Το γεωγραφικό μήκος και πλάτος
- Το υψόμετρο
- Οι συνδεδεμένοι δορυφόροι

Με την βοήθεια των παραπάνω στοιχείων υπολογίζουμε και εμφανίζουμε και κάποια επιπλέον στοιχεία που αφορούν την διαδρομή που διανύει ο χρήστης, όπως είναι:

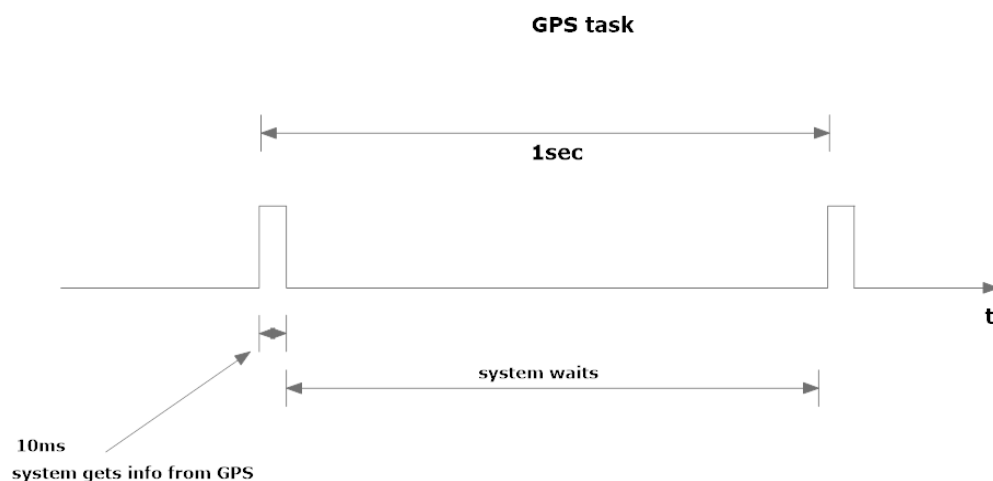
- Η μέση ταχύτητα. Υπολογίζεται παίρνοντας την συνολική απόσταση που έχουμε διανύσει σε μέτρα και διαιρώντας την ως προς τον συνολικό χρόνο που χρειαστήκαμε σε δευτερόλεπτα, αφού πρώτα πολλαπλασιάσουμε με 3,6 ώστε να την εμφανίζουμε σε χιλιόμετρα ανά δευτερόλεπτο.
- Η μέγιστη ταχύτητα. Κάθε χρονική στιγμή συγκρίνουμε την στιγμιαία ταχύτητα με την μέγιστη ταχύτητα (την οποία έχουμε αρχικοποιήσει στην τιμή μηδέν) και αν είναι μεγαλύτερη τότε ορίζουμε ως μέγιστη ταχύτητα την υπάρχουσα τιμή της στιγμιαίας ταχύτητας. Την μέγιστη ταχύτητα την εμφανίζουμε και αυτήν σε χιλιόμετρα ανά δευτερόλεπτο.

- Η συνολική απόσταση. Χρησιμοποιώντας μια συνάρτηση που υπολογίζει τη διαφορά σε μέτρα, δύο διαδοχικών τοποθεσιών που έχουμε λάβει από το GPS, παίρνουμε το αποτέλεσμα που θέλουμε. Κάθε δευτερόλεπτο που πραγματοποιείται η ανανέωση της τοποθεσίας μας, προσθέτουμε τη νέα διαφορά που υπολογίζεται στην συνολική απόσταση που έχουμε διανύσει και την εμφανίζουμε σε χιλιόμετρα με την ακρίβεια δύο δεκαδικών στοιχείων, δηλαδή δέκα μέτρων.

### 4.1.3 Χρονοπρογραμματισμός Διεργασιών

Ο τρόπος που πραγματοποιείται η δειγματοληψία για την άντληση των δεδομένων, από το GPS καί από τον εγκέφαλο του κάθε οχήματος, μέχρι την παρουσίαση τους από το σύστημα παρουσιάζεται παρακάτω. Σε κάθε περίπτωση, για την επιλογή του τρόπου με τον οποίο γίνεται ο χρονοπρογραμματισμός των εργασιών, έπαιξαν σημαντικό ρόλο οι ανάγκες της υλοποίησης μας, καθώς επίσης και οι προδιαγραφές του τεχνικού εξοπλισμού που χρησιμοποιήθηκε.

Πιο συγκεκριμένα, στις περισσότερες συσκευές στις μέρες μας, η συχνότητα που επιτρέπει ο τεχνικός εξοπλισμός (hardware) να γίνονται ανανεώσεις για το GPS είναι 1Hz, δηλαδή κάθε ένα δευτερόλεπτο. Συχνότερες ανανεώσεις από ότι μπορεί να διαχειριστεί η εφαρμογή, θα μπορούσε να προκαλέσει υπερχειλίση δεδομένων ή ακόμα και προβλήματα στην διεπαφή της εφαρμογής.



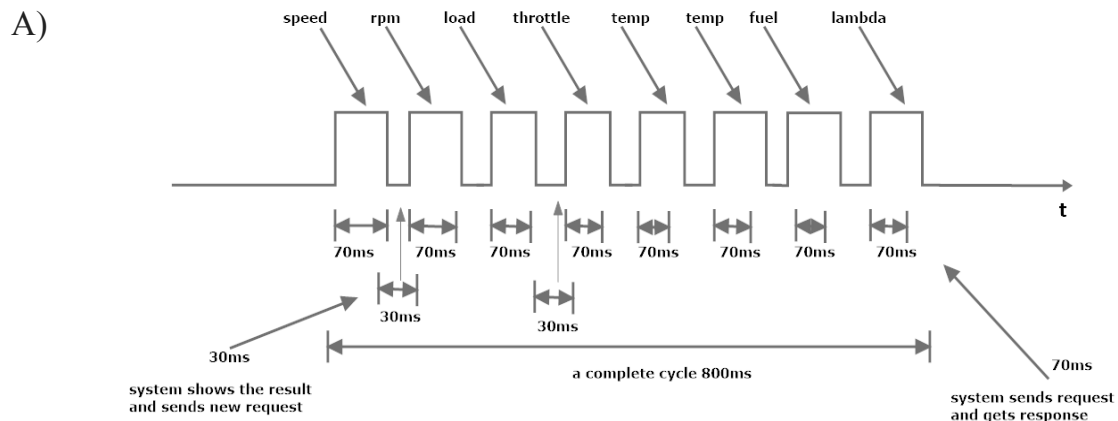
Εικόνα 19. Χρονοπρογραμματισμός διεργασιών για το GPS.

Ένα από τα σημαντικότερα ζητήματα που αντιμετωπίσαμε είναι το πώς μπορούμε να βελτιώσουμε το ρυθμό συλλογής των δεδομένων πραγματικού χρόνου από τον εγκέφαλο του αυτοκινήτου. Δεν υπάρχει τίποτα που μπορούμε να κάνουμε για το πόσο γρήγορα μπορεί ένα όχημα να αποκριθεί σε μια αίτηση μας. Το μόνο που μπορούμε να κάνουμε είναι να

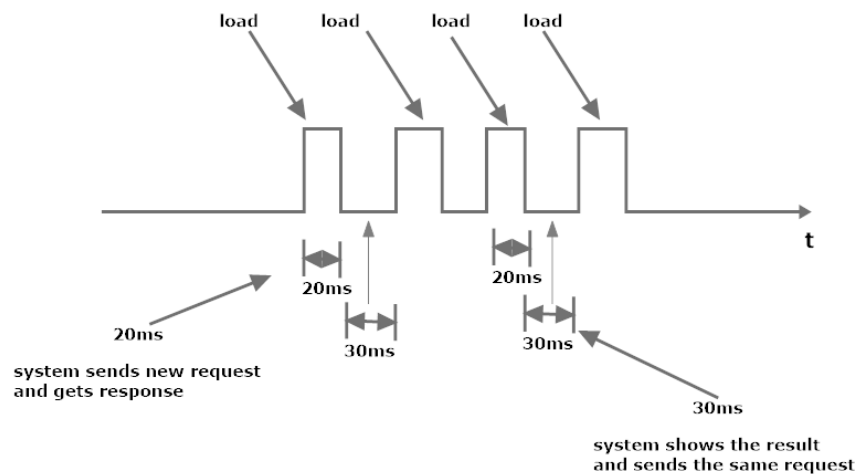
βελτιώσουμε το πώς ο μικροελεγκτής ELM327 μπορεί να χειριστεί το αίτημα μας. Συνεπώς είναι ο μόνος χρόνος που μπορούμε να βελτιώσουμε για την ανανέωση των δεδομένων μας. Για ότι αφορά το όχημα και την συσκευή που χρησιμοποιούμε δεν μπορούμε να κάνουμε κάτι παραπάνω για να βελτιώσουμε την ταχύτητα της συλλογής των δεδομένων. Πιο συγκεκριμένα, οι χρόνοι που παρουσιάζονται παρακάτω μπορεί να διαφέρουν από χρήστη σε χρήστη εξαιτίας του διαφορετικού πρωτοκόλλου που πιθανόν να χρησιμοποιεί το όχημα της δοκιμής, καθώς και της υπολογιστικής ισχύς της συσκευής που χρησιμοποιείται.

Σε ότι αφορά τώρα το μικροελεγκτής ELM327, αφού στείλει λοιπόν ένα αίτημα, περιμένει για περίπου 150ms (από 100ms έως 200ms) για να λάβει μια απάντηση. Αν μέσα σε αυτό το διάστημα δε λάβει κάποια απάντηση, τερματίζει την αναμονή και επιστρέφει την τιμή 'NO DATA'.

Μετά τη λήψη της απάντησης, περιμένει να δει αν θα ακολουθήσει και κάποια ακόμα. Το χρονικό αυτό όριο υπολογίζεται περίπου 150 ms. Δηλαδή σε ένα τυπικό όχημα ο ταχύτερος ρυθμός συλλογής υπολογίζεται σε 5 αιτήματα ανά δευτερόλεπτο, αν σκεφτεί κανείς ότι χρειάζονται 50ms για να σταλεί κάθε αίτημα και άλλα 150ms για το χρονικό όριο μέχρι να ληφθεί η απάντηση. Για να βελτιώσουμε το ρυθμό αυτό λοιπόν, το μόνο πράγμα που μπορούμε να κάνουμε είναι να μειώσουμε το χρονικό όριο αναμονής, μόνο αν γνωρίζουμε πόσες απαντήσεις περιμένουμε. Ο τρόπος που το πετύχαμε αυτό λοιπόν είναι στέλνοντας αντί για το αίτημα 01 0D, για την ταχύτητα για παράδειγμα, να στείλουμε το αίτημα 01 0D 2. Με αυτόν τον τρόπο λέμε στο κύκλωμα να στείλει τον κωδικό 01 0D και να επιστρέψει αμέσως αφού λάβει δύο απαντήσεις. Επομένως δεν μπορούμε να επιταχύνουμε μια αργή ηλεκτρονική μονάδα ελέγχου (ECU), αλλά να μειώσουμε την τελική αναμονή αφού γνωρίζουμε πόσες απαντήσεις να αναμένουμε. Αυτή η αλλαγή μπορεί να μας δώσει μέχρι 10 περίπου απαντήσεις ανά δευτερόλεπτο αντί για 5 ωριότερα.



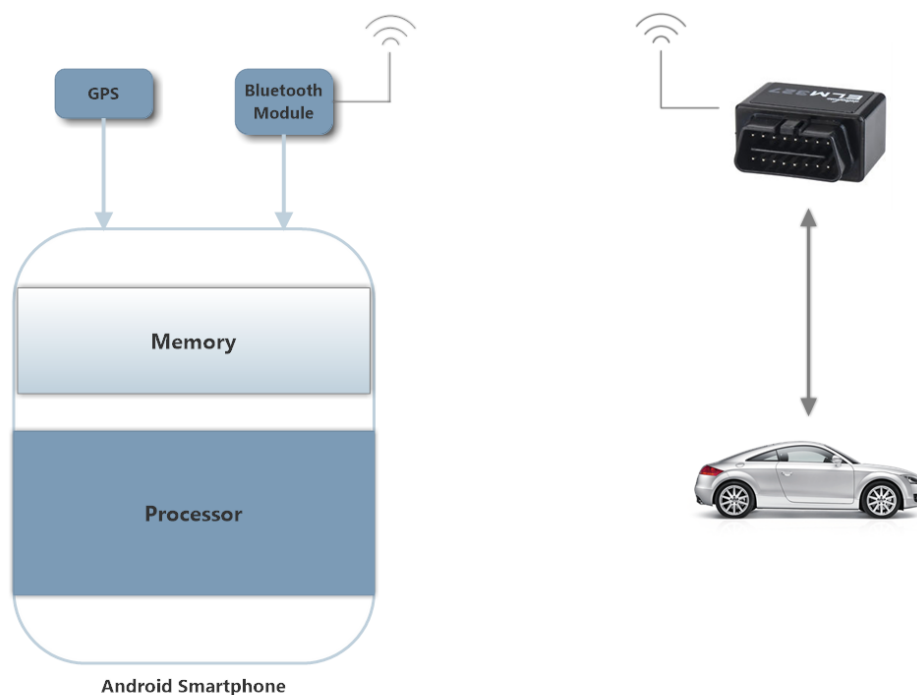
B)



Εικόνα 20. Χρονοπρογραμματισμός διεργασιών για Bluetooth στην οθόνη που εμφανίζονται  
 Α) όλα τα στοιχεία.  
 Β) και στην οθόνη που εμφανίζεται μόνο το ένα από αυτά (πχ. engine load).

## 4.2 Αρχιτεκτονική Συστήματος

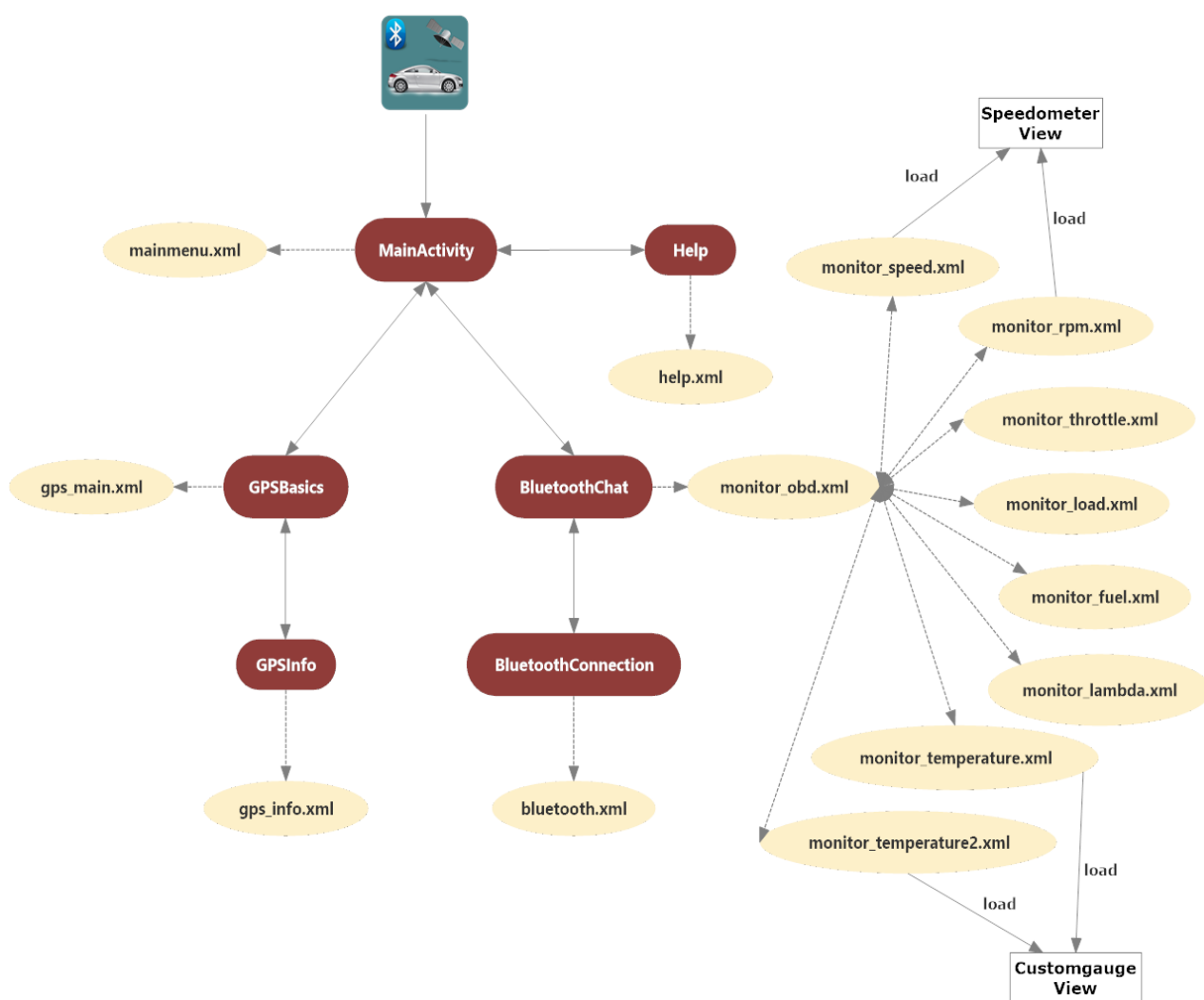
Συνθέτοντας τα επιμέρους τμήματα, δημιουργείται η πλήρης αρχιτεκτονική του συστήματος. Τα τμήματα επικοινωνούν μεταξύ τους και ανταλλάζουν δεδομένα σύμφωνα με την λογική της εφαρμογής μας.



Εικόνα 21. Σχεδιάγραμμα Αρχιτεκτονικής Συστήματος.

Όπως φαίνεται και από την παραπάνω εικόνα, η μονάδα Bluetooth της συσκευής Android επικοινωνεί και ανταλλάσσει δεδομένα με τον μικροελεγκτή ELM327, ο οποίος συνδέεται μέσω της θύρας OBD με το δίκτυο μεταφοράς δεδομένων του οχήματος. Ακόμα ο δέκτης GPS ενημερώνεται ανά τακτά χρονικά διαστήματα για την ακριβή τοποθεσία της συσκευής και παρέχει χρήσιμες πληροφορίες για αυτήν.

Στη συνέχεια και πιο συγκεκριμένα στην εικόνα 22 υπάρχει ένα λεπτομερές σχεδιάγραμμα του συστήματος που υλοποιήθηκε. Στο σχεδιάγραμμα αυτό μπορεί κανείς να δει όλα τα επιμέρους τμήματα της εφαρμογής και τη σύνδεση μεταξύ τους



Εικόνα 22. Αναλυτικό Σχεδιάγραμμα Συστήματος.

### 4.3 Σχεδίαση Εφαρμογής

Η υλοποίηση μιας νέας εφαρμογής στο Android όπως και στα υπόλοιπα λειτουργικά συστήματα, ξεκινάει από τις λειτουργικές απαιτήσεις, δηλαδή τις δυνατότητες και λειτουργίες που θα υποστηρίξει η εφαρμογή, και συνεχίζει με το σχεδιασμό του UI layout που θα δίνει πρόσβαση στον χρήστη, στις παραπάνω λειτουργίες. Ο σχεδιασμός λοιπόν έχει μεγαλύτερη σημασία από τις ίδιες τις λειτουργίες της εφαρμογής, μιας και είναι το σημείο πρόσβασης προς αυτές. Μια κακοσχεδιασμένη εφαρμογή η οποία κρύβει τις λειτουργίες της πίσω από πολλά κουμπιά και μενού, ενδέχεται να μπερδέψει τον χρήστη και ίσως να έχει τα αντίθετα αποτελέσματα από τα επιθυμητά.

Κατά την σχεδίαση της εφαρμογής λάβαμε σοβαρά υπόψη μας ότι η εφαρμογή μπορεί να χρησιμοποιείται από τον οδηγό ενός οχήματος. Για αυτό το λόγο προσπαθήσαμε η υλοποίηση μας να είναι όσο το δυνατόν πιο απλή στην χρήση της και τα δεδομένα που εμφανίζουμε να είναι αρκετά ευδιάκριτα, ώστε να μην αποσπούν την προσοχή του χρήστη.

#### 4.3.1 Λειτουργικές Απαιτήσεις

Η εφαρμογή που υλοποιήσαμε, απαιτεί για τη λειτουργία της, συσκευή με λειτουργικό σύστημα Android. Η συσκευή αυτή είναι απαραίτητο να διαθέτει λειτουργικό σύστημα Android 3.0 ή νεότερο, οθόνη αφής, Bluetooth λειτουργία αλλά και δέκτη GPS.

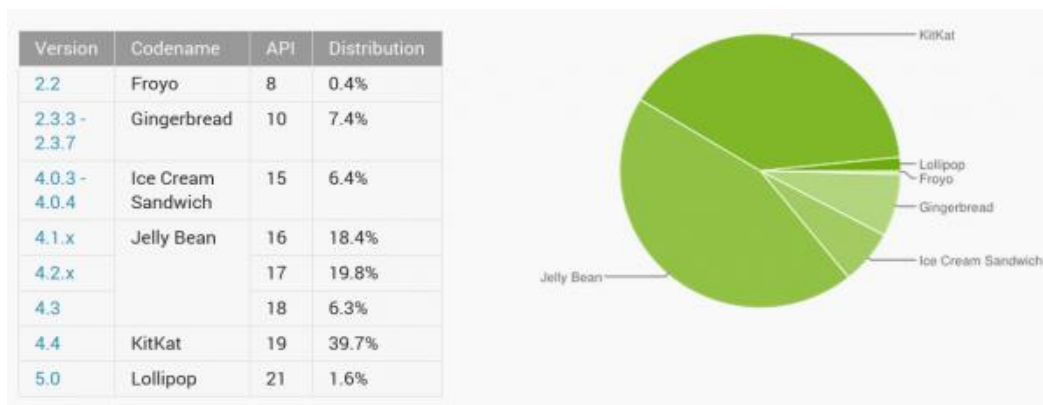
Όσον αφορά τον δέκτη GPS, υπάρχει ενσωματωμένος στη συσκευή που χρησιμοποιούμε και συλλέγει δεδομένα τα οποία επεξεργαζόμαστε και εμφανίζουμε στον χρήστη. Η ταχύτητα με την οποία κινούμαστε, το γεωγραφικό μήκος και πλάτος καθώς και το υψόμετρο που βρισκόμαστε, αποτελούν τα στοιχεία που λαμβάνουμε απευθείας από το δέκτη GPS της συσκευής. Με τη βοήθεια των στοιχείων αυτών και ύστερα από την κατάλληλη επεξεργασία τους, εμφανίζουμε και μερικά ακόμα χρήσιμα στοιχεία για το χρήστη της εφαρμογής. Τα στοιχεία αυτά μπορούν να δώσουν χρήσιμες πληροφορίες στον χρήστη, που αφορούν στην διαδρομή που διένυσε, όπως είναι το μήκος της συνολικής διαδρομής, η μέγιστη και μέση ταχύτητα της αλλά και ο συνολικός χρόνος που διήρκεσε.

Ο μικροελεγκτής ELM327 αποτελεί απαραίτητο εργαλείο για τον χρήστη της εφαρμογής μας. Το εργαλείο αυτό λειτουργεί ως γέφυρα μεταξύ της ηλεκτρονικής μονάδας ελέγχου (ECU) του αυτοκινήτου και της συσκευής μας. Ο ELM327 επικοινωνεί με το OBDII σύστημα του οχήματος και στη συνέχεια εκτελεί μεταφορά των δεδομένων μέσω του Bluetooth. Τα δεδομένα αυτά πραγματικού χρόνου τα συλλέγουμε και αφού τα επεξεργαστούμε κατάλληλα τα εμφανίζουμε και αυτά στην οθόνη της συσκευής. Μερικά από τα δεδομένα αυτά είναι διαθέσιμα και από το ταμπλό του αυτοκινήτου. Όμως με την χρήση της εφαρμογής που υλοποιήσαμε ενημερωνόμαστε και για στοιχεία του αυτοκινήτου που δεν ήταν

διαθέσιμα νωρίτερα και εκτός από αυτό, μας δίνεται η δυνατότητα καταγραφής τους σε αρχείο στην μνήμη της συσκευής για περαιτέρω μελέτη. Με αυτόν τον τρόπο μπορούμε να μελετήσουμε και να εξάγουμε χρήσιμα συμπεράσματα για την συμπεριφορά ορισμένων μεταβλητών του οχήματος.

Λειτουργικές απαιτήσεις της εφαρμογής μας

- Κινητό τηλέφωνο ή tablet με λειτουργικό σύστημα Android 3.0 ή νεότερο
- Οθόνη αφής
- Συνδεσιμότητα με Bluetooth, GPS
- Ελάχιστη χωρητικότητα 3MB
- ELM 327



Εικόνα23. Ποσοστά χρήσης Android εκδόσεων 1 Φεβρουαρίου 2015.<sup>[46]</sup>

### 4.3.2 Υποστήριξη Πολλαπλών Συσκευών

Το λειτουργικό σύστημα Android τρέχει σε μια πληθώρα συσκευών οι οποίες μπορεί να έχουν πολύ διαφορετικές προδιαγραφές ή μία από την άλλη. Η διαφοροποίηση των συσκευών εντοπίζεται:

- Στις πολλές εκδόσεις του Android που υπάρχουν.
- Στην μεγάλη ποικιλία hardware που κυκλοφορεί στην αγορά. Υπάρχουν συσκευές με πολύ διαφορετικά χαρακτηριστικά, όσον αφορά την υπολογιστική ισχύ του επεξεργαστή και την μνήμη ram της συσκευής. Εκτός από τις διαφορές σε επίπεδο microchip η κύρια διαφορά μεταξύ των συσκευών εντοπίζεται στη μεγάλη ποικιλία διαστάσεων οθόνης και ανάλυσης.

Έτσι λοιπόν για να κάνουμε την εφαρμογή προσβάσιμη σε όσο τον δυνατόν περισσότερες συσκευές χρηστών, λάβαμε σοβαρά υπόψη τις 2 παραπάνω παραμέτρους και

σχεδιάσαμε την εφαρμογή έτσι ώστε να υποστηρίζει την πλειονότητα των συσκευών. Βέβαια αυτό σημαίνει συνεχή προσαρμογή στις νέες συνθήκες που μπορεί να προκύψουν, αλλά και χρήση των νέων δυνατοτήτων που ενδεχομένως θα παρέχει μια νέα έκδοση του λειτουργικού, χωρίς να επηρεάζεται η υποστήριξη στις παλαιότερες συσκευές.

Αυτή η συνεχής εξέλιξη της πλατφόρμας αποτελεί πλεονέκτημα αλλά και πρόκληση για τον προγραμματιστή ο οποίος θα πρέπει να ακολουθεί τις εξελίξεις και να χρησιμοποιεί τις νέες δυνατότητες που του προσφέρει η κάθε έκδοση, χωρίς να παραγκωνίζει την υποστήριξη στις παλαιότερες εκδόσεις του Android. Αυτό είναι ένα σημαντικό πρόβλημα καθότι κάποιες νέες λειτουργίες δεν υποστηρίζονται από παλιότερες εκδόσεις και άρα καθιστούν αδύνατη τη χρήση τους σε κάποια παλαιότερη έκδοση του Android.

Μια τέτοια λειτουργία, που απαιτούσε τουλάχιστον την έκδοση Android 4.0.x ή νεότερη, είχαμε αρχικά χρησιμοποιήσει για την εμφάνιση των δεδομένων από τον εγκέφαλο του αυτοκινήτου. Πιο συγκεκριμένα με την χρήση της λειτουργίας αυτής θα μπορούσε ο χρήστης να περιηγηθεί μεμονωμένα σε κάθε ένα από τα διαθέσιμα στοιχεία κινώντας το χέρι του προς τα αριστερά ή προς τα δεξιά της οθόνης για εναλλαγή. Τελικά όμως η λειτουργία αυτή δεν ενσωματώθηκε στην υλοποίηση μας, διότι προστέθηκαν και άλλα στοιχεία στα ήδη διαθέσιμα, γεγονός που θα έκανε την εναλλαγή τους κατά αυτόν τον τρόπο δυσκολότερη και πιο χρονοβόρα.

### 4.3.3 Σχεδίαση Διεπαφής Χρήστη

Στην ενότητα αυτή περιγράφονται οι στρατηγικές που ακολουθήθηκαν για την σχεδίαση των γραφικών διεπαφών του συστήματος μας. Βασική αρχή που ακολουθήσαμε είναι η απλότητα στις επιλογές, για να καλύψουμε χρήστες με μικρότερη εξοικείωση σε εφαρμογές συσκευών με λειτουργικό σύστημα Android.

Ακολουθούν οι κανόνες που ακολουθήσαμε για την πρωταρχική σχεδίαση των User Interfaces. Προφανώς τα User Interfaces διαφοροποιούνται σε μικρό βαθμό στις διάφορες οθόνες (views) του συστήματος τηρώντας όμως σε έναν μεγάλο βαθμό την αρχή της σταθερότητας (consistency).

Αναλυτικότερα οι κανόνες:

- Λίγα κουμπιά και επιλογές σε κάθε οθόνη.
- Ευκολία και απλότητα στην πλοήγηση μεταξύ των οθονών.
- Ευδιάκριτα στοιχεία και πληροφορίες για να μην αποσπούν την προσοχή του χρήστη, καθώς η εφαρμογή μπορεί να χρησιμοποιείται κατά την διάρκεια της οδήγησης. (Δεν συνιστάται)

- Συγκέντρωση των βασικών λειτουργιών στην ActionBar που χρησιμοποιείται σχεδόν σε όλες τις οθόνες του συστήματος, ενώ οι λιγότερο χρησιμοποιούμενες εμφανίζονται πατώντας το πλήκτρο του μενού.

Οι οθόνες του συστήματος μας:

1. Αρχική οθόνη – Main Menu
2. Οθόνη βοήθειας – Help
3. Οθόνη εμφάνισης δεδομένων από τον εγκέφαλο του αυτοκινήτου – Monitor OBD
4. Οθόνη εμφάνισης ταχύτητας – Monitor Speed
5. Οθόνη εμφάνισης ανά λεπτό στροφών του κινητήρα – Monitor RPM
6. Οθόνη εμφάνισης του επί τοις εκατό ποσοστού της θέσης πεταλούδας γκαζιού – Monitor Throttle
7. Οθόνη εμφάνισης του επί τοις εκατό ποσοστού φορτίου του κινητήρα – Monitor Load
8. Οθόνη εμφάνισης θερμοκρασίας αντιψυκτικού της μηχανής – Monitor Temperature
9. Οθόνη εμφάνισης θερμοκρασίας του αέρα εισαγωγής στον κινητήρα – Monitor Temperature2
10. Οθόνη εμφάνισης στιγμιαίας κατανάλωσης καυσίμου – Monitor Fuel
11. Οθόνη εμφάνισης τιμής του αισθητήρα λ – Monitor Lambda
12. Οθόνη σύνδεσης μέσω Bluetooth - Bluetooth
13. Οθόνη εμφάνισης δεδομένων από το GPS της συσκευής – GPS Basics
14. Οθόνη εμφάνισης πληροφοριών για το GPS της συσκευής – GPS Info

#### 4.4 Ρυθμός Συλλογής Δεδομένων

Σημαντικό μέρος στη συλλογή των δεδομένων πραγματικού χρόνου παίζει και το γεγονός πως θέλουμε να υπάρχει ακρίβεια στα αποτελέσματα που εμφανίζουμε. Δεν θέλουμε σε καμία περίπτωση να εμφανίζουμε πάρα πολλά στοιχεία, διότι αυτό όχι μόνο θα είχε αντίθετα αποτελέσματα σε ότι αφορά τον ρυθμό με τον οποίο συλλέγονται τα δεδομένα, αλλά και κάθε χρονική στιγμή θα δίνουμε τόσες πολλές πληροφορίες στον χρήστη που δεν θα μπορούσε να τις αξιοποιήσει κατάλληλα. Οι χρόνοι που αναφέρονται παρακάτω μπορεί να διαφέρουν ανάλογα με την συσκευή Android και το μικροελεγκτή ELM327 που χρησιμοποιούμε.

#### 4.4.1 Bluetooth

Στην περίπτωση μας για κάθε στοιχείο η διαδικασία της απαίτησης κάποιου κωδικού και της επιστροφής του αποτελέσματος μέχρι και την εμφάνισή του στην οθόνη του χρήστη διαρκεί περίπου 0,1 δευτερόλεπτο (100ms). Συνεπώς για την ανανέωση όλων των αποτελεσμάτων, στην οθόνη που εμφανίζονται όλα τα στοιχεία μαζί, χρειάζεται περίπου 800ms. Ο χρόνος αυτός υπολογίζεται από τα οχτώ κύρια στοιχεία που εμφανίζονται στη κύρια οθόνη, δηλαδή τους οχτώ κωδικούς που στέλνονται.

Αντίθετα ο χρόνος συλλογής των δεδομένων στην οθόνη που στέλνεται και εμφανίζεται μόνο το ένα στοιχείο είναι πολύ μικρότερος (50ms) πράγμα το οποίο σημαίνει πως το συγκεκριμένο στοιχείο ανανεώνεται πάρα πολύ συχνά. Με το μεγαλύτερο αυτό ρυθμό συλλογής δεδομένων δίνεται η δυνατότητα στον χρήστη να ελέγχει κάποιο στοιχείο που επιθυμεί με πολύ μεγαλύτερη ακρίβεια αποτελεσμάτων, καθώς τα δείγματα που συγκεντρώνει μπορεί να φτάσουν τα 20 για κάθε δευτερόλεπτο.

Η μόνη διαφορά στο ρυθμό συλλογής υπάρχει στην οθόνη όπου εμφανίζεται ένα μονάχα στοιχείο και αυτό είναι η στιγμιαία κατανάλωση καυσίμου. Αυτό έγκειται στο γεγονός ότι για τον υπολογισμό της, λαμβάνουμε υπόψη την στιγμιαία ταχύτητα του αυτοκινήτου. Έτσι για να εξάγουμε κάθε φορά το σωστό αποτέλεσμα για την κατανάλωση θα πρέπει να υπολογίσουμε εκ νέου την στιγμιαία ταχύτητα, γεγονός που σημαίνει ότι στέλνουμε δύο κωδικούς κάθε φορά. Επομένως ο χρόνος μεταξύ διαδοχικών ανανεώσεων των τιμών της στιγμιαίας κατανάλωσης καυσίμου είναι περίπου 0,2 δευτερόλεπτα (200ms).

#### 4.4.2 GPS

Ο ρυθμός συλλογής που αφορά το GPS διαφέρει αρκετά σε σχέση με τον προηγούμενο ρυθμό και κυρίως στον τρόπο με τον οποίο συλλέγουμε τα δεδομένα. Η συλλογή των δεδομένων στο κομμάτι αυτό, σε αντίθεση με το Bluetooth, πραγματοποιείται κάθε χρονική στιγμή που έχουμε ορίσει εμείς (στην περίπτωση μας κάθε ένα δευτερόλεπτο). Κάθε χρονική στιγμή εκτελείται ένα μέρος του κώδικα που έχει να κάνει με την ενημέρωση όλων των στοιχείων που εμφανίζονται στην οθόνη και στην συνέχεια σταματάει, μέχρις ότου εκτελεσθεί ξανά την επόμενη χρονική στιγμή.

Το ένα δευτερόλεπτο που έχει οριστεί είναι ένα πολύ καλό χρονικό διάστημα για ανανέωση των στοιχείων που εμφανίζονται, καθώς δεν επιβαρύνεται το σύστημα με συνεχείς ανανεώσεις πολύ συχνότερα και συνεπώς καταναλώνει λιγότερη ενέργεια, αλλά και κάτι τέτοιο δεν θα είχε ιδιαίτερο νόημα καθώς οι τιμές των στοιχείων δεν θα είχαν αισθητές αποκλίσεις.

## 4.5 Βιβλιοθήκες

### 4.5.1 Bluetooth API

Η πλατφόρμα Android περιλαμβάνει υποστήριξη για το Bluetooth, η οποία επιτρέπει σε μια συσκευή να ανταλλάσσει δεδομένα ασύρματα με άλλες συσκευές μέσω Bluetooth. Το Application Framework παρέχει πρόσβαση στη λειτουργικότητα του Bluetooth με την βοήθεια της βιβλιοθήκης Bluetooth API. Η βιβλιοθήκη αυτή επιτρέπει στις εφαρμογές να συνδέονται ασύρματα με άλλες συσκευές Bluetooth και να ανταλλάσσουν δεδομένα.

Χρησιμοποιώντας την συγκεκριμένη βιβλιοθήκη κατάλληλα, μια εφαρμογή Android μπορεί να εκτελέσει τα εξής:<sup>[29]</sup>

- Σάρωση για άλλες συσκευές Bluetooth
- Αντιστοίχιση δύο συσκευών Bluetooth
- Σύνδεση με άλλες συσκευές μέσω της λειτουργίας ανακάλυψης
- Μεταφορά δεδομένων μεταξύ συσκευών
- Διαχειριστή πολλαπλών συνδέσεων

Για την ανακάλυψη, σύνδεση και το ταίριασμα συσκευών Bluetooth απαιτούνται σχετικές άδειες (BLUETOOTH, BLUETOOTH\_ADMIN), τις οποίες και προσθέσαμε στο Manifest.xml αρχείο μας.

### 4.5.2 LocationManager

Η κλάση αυτή ανήκει στις μεθόδους της βιβλιοθήκης Location API και μας παρέχει πρόσβαση στις υπηρεσίες εντοπισμού του συστήματος. Οι υπηρεσίες αυτές επιτρέπουν στις εφαρμογές να αποκτήσουν περιοδικές ενημερώσεις της γεωγραφικής θέσης της συσκευής, ή ακόμα και να εκτελέσουν κάποια λειτουργία σε περίπτωση που παρατηρηθεί αλλαγή της γεωγραφικής θέσης.

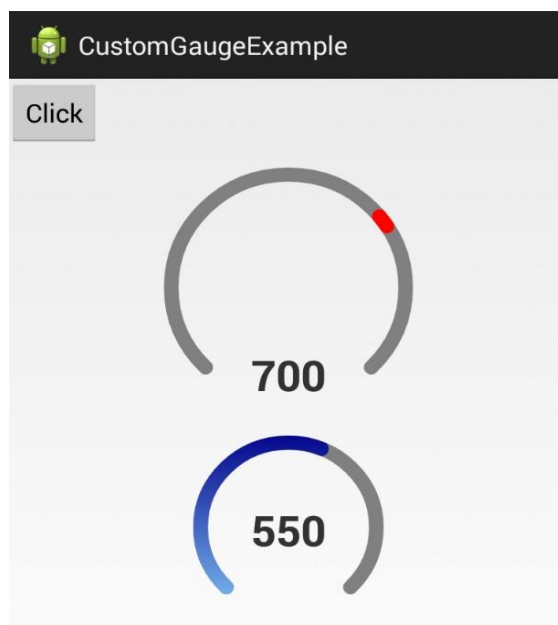
Για μπορέσουμε να αντλήσουμε πληροφορίες για την γεωγραφική θέση της συσκευής απαιτείται σχετική άδεια (ACCESS\_FINE\_LOCATION), την οποία προσθέσαμε στο Manifest.xml αρχείο μας.

### 4.5.3 CustomGauge

Η βιβλιοθήκη αυτή χρησιμοποιείται στη υλοποίηση μας για να εμφανίζει στοιχεία που αφορούν το κομμάτι της εμφάνισης δεδομένων πραγματικού χρόνου από τον εγκέφαλο του αυτοκινήτου και έχουν να κάνουν με στοιχεία θερμοκρασίας, όπως είναι:

- η θερμοκρασία αντιψυκτικού της μηχανής

- η θερμοκρασία του αέρα εισαγωγής στον κινητήρα



Εικόνα 24. Αρχική εικόνα βιβλιοθήκης CustomGauge.

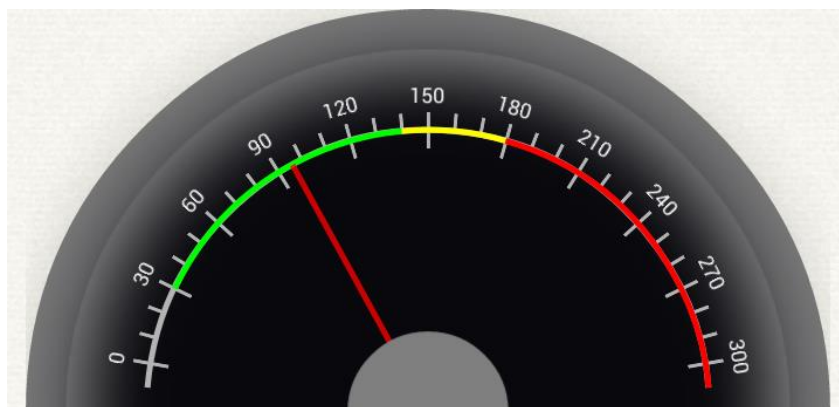
#### 4.5.4 SpeedometerView

Η βιβλιοθήκη αυτή χρησιμοποιείται στη υλοποίηση μας και αφορά το κομμάτι της εμφάνισης αποτελεσμάτων από τον εγκέφαλο του αυτοκινήτου για τα εξής στοιχεία:

- την ταχύτητα με την οποία κινείται το αυτοκίνητο
- τις ανά λεπτό στροφές του κινητήρα

Όπως θα δείτε και στην συνέχεια της εργασίας και πιο συγκεκριμένα στο κεφάλαιο της παρουσίασης της εφαρμογής, η συγκεκριμένη βιβλιοθήκη επεξεργάστηκε αρκετά μέχρι να καταλήξει στην τελική της μορφή, διότι έπρεπε να προσαρμοστεί, εκτός από τις απαιτήσεις της υλοποίησης μας και στο κομμάτι της εμφάνισης.

Στην εικόνα 25 βλέπουμε την αρχική εικόνα της βιβλιοθήκης πριν την επεξεργασία μας.



Εικόνα 25. Αρχική εικόνα βιβλιοθήκης SpeedometerView.



## Κεφάλαιο 5

### Παρουσίαση Εφαρμογής CarSystem

#### 5.1 Android Manifest.xml

Όλες οι εφαρμογές Android χρειάζεται να έχουν ένα αρχείο XML, στη διεύθυνση root της εφαρμογής, το οποίο ονομάζεται Android Manifest.xml. Αυτό το αρχείο χρησιμοποιείται από διάφορες παροχές του συστήματος προκειμένου να λαμβάνονται βασικές πληροφορίες για την εφαρμογή. Στο αρχείο Android Manifest προκαθορισμένοι τύποι στοιχείων επιτρέπεται να καθοριστούν, όπως το όνομα της εφαρμογής, το στοιχείο της εφαρμογής, διάφορες άδειες, απαιτούμενες βιβλιοθήκες, και διάφορα φίλτρα.

Ως όνομα της εφαρμογής μας καθορίσαμε το CarSystem και παρακάτω παραθέτουμε και το αντίστοιχο εικονίδιο.



Εικόνα 26. Εικονίδιο της εφαρμογής μας CarSystem.

Στην παρούσα εργασία ακόμα καθορίζονται άδειες για την ανακάλυψη, σύνδεση και το ταίριασμα συσκευών Bluetooth (BLUETOOTH, BLUETOOTH\_ADMIN), άδεια για την αποθήκευση στην εξωτερική μνήμη (SD) της συσκευής (WRITE\_EXTERNAL\_STORAGE) καθώς και άδειες για την ακριβή τοποθεσία από το GPS της συσκευής (ACCESS\_FINE\_LOCATION) και το δίκτυο (INTERNET).

## 5.2 Δραστηριότητες του Συστήματος

Οι δραστηριότητες (Activities) του συστήματος αποτελούν ουσιαστικά τις οθόνες της εφαρμογής. Η Android εφαρμογή που δημιουργήσαμε αποτελείται από 6 επιμέρους δραστηριότητες.

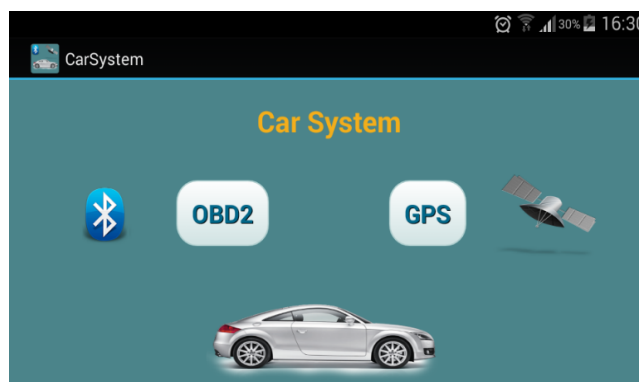
Η εφαρμογή μας υποστηρίζει και τους δύο προσανατολισμούς της οθόνης (portrait, landscape). Μόλις όμως ο χρήστης επιλέξει μια από τις δύο βασικές επιλογές που εμφανίζονται στο κύριο μενού της εφαρμογής, τότε θέτουμε ως βασικό προσανατολισμό την τρέχουσα διάταξη της οθόνης, για τις εξής δραστηριότητες: MainActivity, GPSBasics, BluetoothConnection και BluetoothChat.

Για την υλοποίησή μας καταλήξαμε σ' αυτόν τον τρόπο, διότι αντιμετωπίσαμε αρκετά προβλήματα στην σταθερότητα και την αξιοπιστία της εφαρμογής. Πιο συγκεκριμένα αντιμετωπίσαμε προβλήματα με την διατήρηση της σύνδεσης Bluetooth, καθώς επίσης με τη διατήρηση των στοιχείων του GPS κατά την εναλλαγή του προσανατολισμού της οθόνης. Στις υπόλοιπες δραστηριότητες της εφαρμογής η αλλαγή του προσανατολισμού της συσκευής πραγματοποιείται κανονικά.

Στη συνέχεια θα παραθέσουμε και τις δύο διαφορετικές οθόνες, ανάλογα με τον προσανατολισμό της συσκευής που χρησιμοποιούμε, για την κάθε διαφορετική δραστηριότητα.

### 5.2.1 MainActivity

Ανοίγοντας την εφαρμογή εμφανίζεται η ακόλουθη σελίδα, η οποία αποτελεί και το κεντρικό μενού της εφαρμογής μας.



Εικόνα 27. Κεντρικό μενού της εφαρμογής σε οριζόντιο και κάθετο προσανατολισμό της οθόνης.

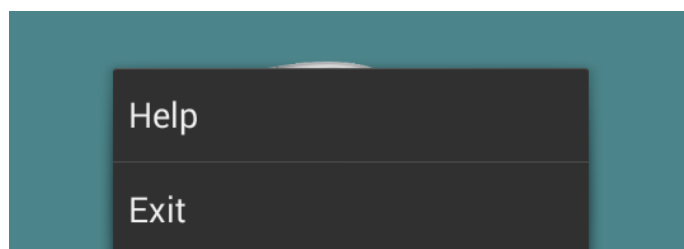
Σε αυτήν την σελίδα εμφανίζονται δύο κουμπιά, ένα για κάθε βασική επιλογή που διαθέτει ο χρήστης. Η πρώτη επιλογή αφορά στο κομμάτι της εφαρμογής που ενσωματώνει την σύνδεση της συσκευής μας με τον OBD-II αντάπτορα μέσω Bluetooth, καθώς επίσης και την συλλογή και καταγραφή των δεδομένων πραγματικού χρόνου που λαμβάνονται από τον εγκέφαλο του αυτοκινήτου. Η δεύτερη επιλογή ενσωματώνει το κομμάτι της συλλογής δεδομένων πραγματικού χρόνου από το GPS της συσκευής μας και την δυνατότητα υπολογισμού στοιχείων ταξιδιού.

Επιπλέον πατώντας το πλήκτρο του μενού εμφανίζονται δύο ακόμα επιλογές που έχει ο χρήστης: Help και Exit.

- Η επιλογή Help, δίνει πληροφορίες για την εφαρμογή μας καθώς και οδηγίες χρήσης.
- Η επιλογή Exit, τερματίζει όλες τις ενεργές δραστηριότητες της εφαρμογής και απελευθερώνει τυχόν δεσμευμένους πόρους της συσκευής από αυτήν.

### 5.2.2 HelpActivity

Στην κεντρική σελίδα και ειδικότερα πατώντας το πλήκτρο του μενού και ύστερα την επιλογή Help,



Εικόνα 28. Εικόνα για βοήθεια και έξοδο από το σύστημα.

εμφανίζονται κάποιες οδηγίες και διευκρινήσεις που αφορούν στην λειτουργία της εφαρμογής μας και βοηθούν τον χρήστη στην ομαλή προσαρμογή του.

Αναλυτικότερα εμφανίζονται ορισμένες πληροφορίες για την τοποθέτηση του αντάπτορα OBD-II στη σωστή θύρα του αυτοκινήτου και την διαδικασία που πρέπει να ακολουθήσουμε έτσι ώστε να συνδεθούμε επιτυχώς με την συσκευή που χρησιμοποιούμε.

Επιπλέον μας δίνονται μερικές οδηγίες που αφορούν τις λειτουργίες της εφαρμογής. Αυτές είναι:

- Η καταγραφή των στοιχείων και η επισήμανση τους σε αρχείο στην μνήμη της συσκευής μας, σε περίπτωση που ο χρήστης το επιθυμεί.
- Η δυνατότητα που παρέχεται από την εφαρμογή, για αποστολή επιπλέον κωδικού για την λήψη περεταίρω πληροφοριών.
- Μια λειτουργία, η οποία καθαρίζει τους κωδικούς σφαλμάτων που είναι καταχωρημένοι στην ηλεκτρονική μονάδα ελέγχου (ECU) του αυτοκινήτου.

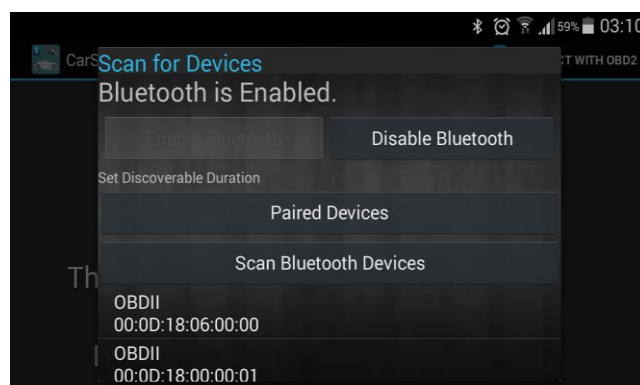
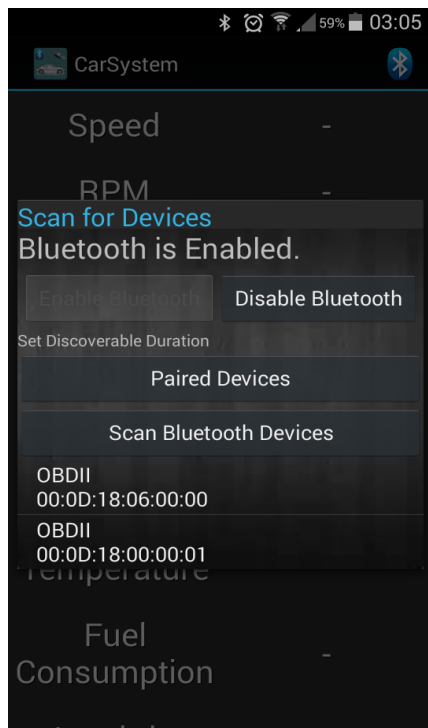
### 5.2.3 BluetoothConnection

Δραστηριότητα η οποία δίνει στον χρήστη τη δυνατότητα να ενεργοποιήσει ή να απενεργοποιήσει το Bluetooth, να του εμφανίσει τις ήδη ταιριασμένες συσκευές αλλά και να σαρώσει για όλες τις συσκευές που βρίσκονται σε κοντινή εμβέλεια. Εμφανίζεται στο προσκήνιο αφού ο χρήστης επιλέξει το εικονίδιο του Bluetooth που εμφανίζεται στην Action Bar της BluetoothChat δραστηριότητας, στο πάνω δεξιά μέρος της οθόνης.



Εικόνα 29. Bluetooth Button.

Αφού ο χρήστης επιλέξει το εργαλείο διάγνωσης OBD-II, πραγματοποιείται η προσπάθεια σύνδεσης των δύο συσκευών και αντίστοιχο μήνυμα εμφανίζεται στην οθόνη. Αυτή την προσπάθεια σύνδεσης των δύο συσκευών και την επιστροφή του αντίστοιχου αποτελέσματος, διαχειρίζεται η Java κλάση μας ManageConnection. Η συγκεκριμένη κλάση περιέχει ένα thread, το οποίο “ακούει” για εισερχόμενες συνδέσεις, ένα για την σύνδεση με μια συσκευή και ένα που εκτελεί μεταφορές δεδομένων, εφ’ όσον υπάρχει επιτυχής σύνδεση μεταξύ των δύο συσκευών (connected).



Εικόνα 30. Οθόνες σάρωσης και σύνδεσης συσκευής μέσω Bluetooth.

### 5.2.4 BluetoothChat

Επιλέγοντας την πρώτη επιλογή από το κεντρικό μενού της εφαρμογής μας, εμφανίζεται στον χρήστη η εξής οθόνη:



Εικόνα 31. Οθόνη εμφάνισης στοιχείων από τον εγκέφαλο του αυτοκινήτου.

Επειδή τα δεδομένα που εμφανίζονται σε αυτήν την οθόνη είναι αρκετά, για να είναι ευδιάκριτα στον χρήστη σε περίπτωση που η οθόνη της συσκευής του δεν είναι αρκετά μεγάλη, χρησιμοποιήσαμε μια Scrollbar.

Με το που εισέρχεται ο χρήστης στη συγκεκριμένη δραστηριότητα, ελέγχεται αν το Bluetooth είναι ενεργοποιημένο και αν όχι, εμφανίζεται αντίστοιχο μήνυμα στον χρήστη με επιλογή για την άμεση ενεργοποίησή του. Σε αυτό το σημείο πρέπει ο αντάπτορας OBD-II να προσαρμοστεί στην κατάλληλη υποδοχή του αυτοκινήτου και στη συνέχεια ο χρήστης από τη συσκευή που χρησιμοποιεί, να συνδεθεί με τη συσκευή OBD-II έτσι ώστε να αρχίσει η λήψη των δεδομένων πραγματικού χρόνου από τον εγκέφαλο του αυτοκινήτου.

Η διαδικασία αυτή επιτυγχάνεται με την βοήθεια της BluetoothConnection δραστηριότητας που περιγράφουμε παραπάνω.

Μόλις οι δύο συσκευές συνδεθούν μεταξύ τους, ξεκινάει η δραστηριότητα να στέλνει κωδικούς μέσω του Bluetooth της συσκευής και με τη βοήθεια του ELM327 στο δίκτυο του αυτοκινήτου, σύμφωνα με το πρωτόκολλο (CAN) που έχουμε ορίσει. Το σύστημα αναγνωρίζει τους κωδικούς και επιστρέφει το ζητούμενο δεδομένο κάθε φορά σε δεκαεξαδική μορφή, το οποίο αφού το μετατρέψουμε στο δεκαδικό σύστημα αρίθμησης και το εισάγουμε στην

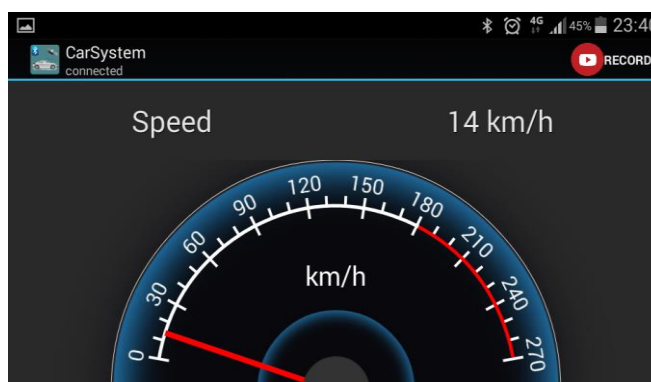
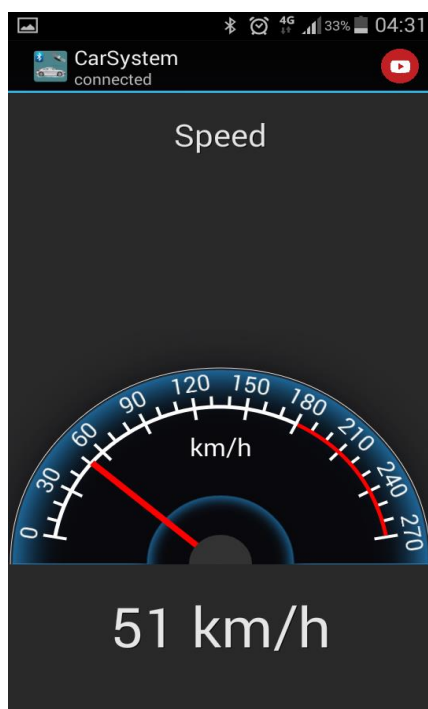
κατάλληλη εξίσωση που ορίζει ο κατασκευαστής, το εμφανίζουμε στην οθόνη. Αυτό συμβαίνει διαδοχικά για όλους τους κωδικούς που έχουμε ορίσει στην δραστηριότητά μας.

Εκτός από την οθόνη που παραθέσαμε νωρίτερα, ο χρήστης μπορεί να εστιάσει σε όποιο από τα διαθέσιμα στοιχεία επιθυμεί. Πατώντας λοιπόν κάποιο από τα στοιχεία που εμφανίζονται, μεταφέρεται σε μια νέα οθόνη, στην οποία εμφανίζεται μεμονωμένα το στοιχείο που επέλεξε. Η λειτουργία αυτή υλοποιήθηκε ώστε τα αποτελέσματα που λαμβάνονται για κάθε στοιχείο που επιθυμεί ο χρήστης, να είναι πολύ πιο ευδιάκριτα καθώς και η συχνότητα ανανέωσής τους πολύ μεγαλύτερη, όπως θα εξηγήσουμε παρακάτω στο κομμάτι που αναφέρεται στον ρυθμό συλλογής δεδομένων.

Τα στοιχεία που εμφανίζονται στην οθόνη, τα οποία μπορεί ο χρήστης να παρακολουθεί μαζικά αλλά και μεμονωμένα εστιάζοντας σε κάποιο από αυτά, είναι τα εξής:

- η ταχύτητα με την οποία κινείται το αυτοκίνητο (Vehicle speed). Για τον υπολογισμό της ταχύτητας, αρχικά στέλνεται ο κωδικός 01 0D και μας επιστρέφεται ως απάντηση ένα byte σαν αποτέλεσμα. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και την εμφανίζουμε απευθείας στην οθόνη χωρίς να την εισάγουμε σε κάποια εξίσωση στην περίπτωση της ταχύτητας. Η αναπαράσταση της ταχύτητας γίνεται σε χιλιόμετρα ανά δευτερόλεπτο.

Speed 48 km/h



Εικόνα 32. Οθόνες εμφάνισης ταχύτητας οχήματος.

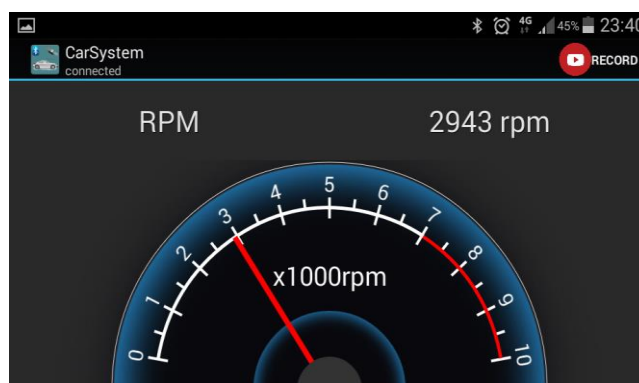
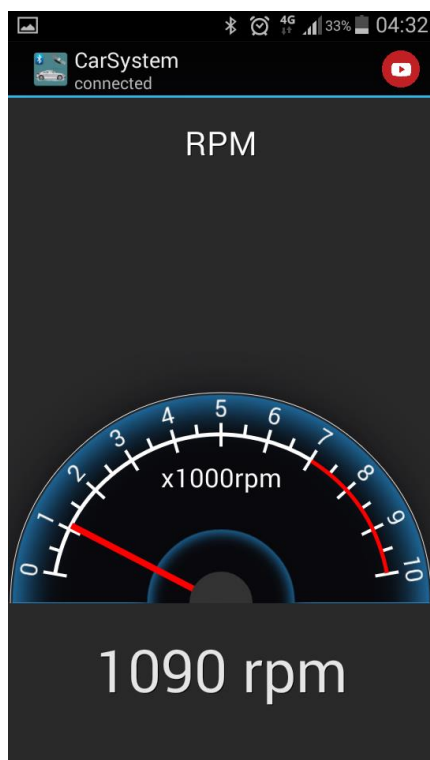
- Οι ανά λεπτό στροφές του κινητήρα (Engine RPM). Για τον υπολογισμό των στροφών του κινητήρα, στέλνεται ο κωδικός 01 0C και μας επιστρέφονται ως απάντηση δύο byte αυτήν την φορά σαν αποτέλεσμα, αφού η αναπαράσταση της τιμής των στροφών του κινητήρα ανά λεπτό δεν χωράει να αναπαρασταθεί με λιγότερα . Μετατρέπουμε τις δύο τιμές που λάβαμε στο δεκαδικό σύστημα αρίθμησης και τις εισάγουμε στην παρακάτω εξίσωση

$$((A * 256) + B)/4$$

όπου A η πρώτη κατά σειρά τιμή που μας επιστρέφεται και B η δεύτερη, και εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα. Η αναπαράσταση των στροφών του κινητήρα γίνεται σε rpm (Revolutions per minute).

RPM

1868 rpm



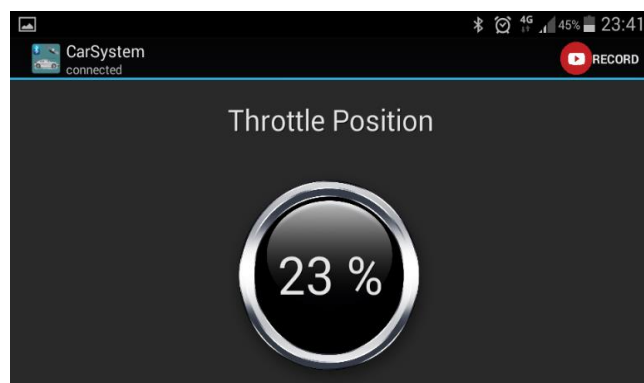
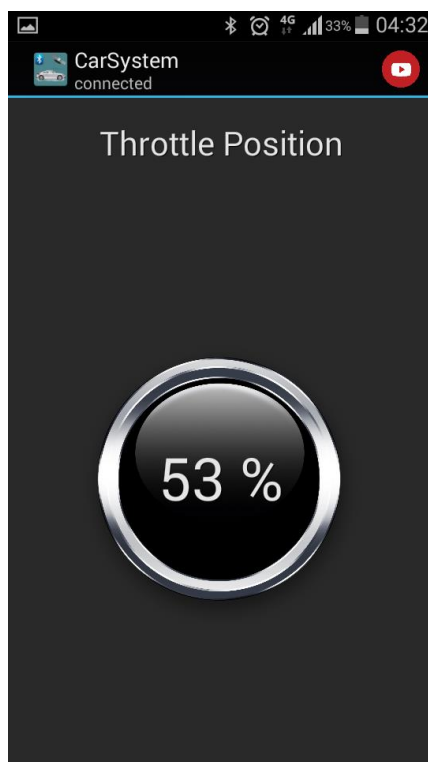
Εικόνα 33. Οθόνες εμφάνισης στροφών κινητήρα.

- το επί τοις εκατό ποσοστό της θέσης πεταλούδας γκαζιού (Throttle position). Για τον υπολογισμό της θέσης πεταλούδας γκαζιού, στέλνεται ο κωδικός 01 11 και μας επιστρέφεται ως απάντηση ένα byte σαν αποτέλεσμα. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και βάζοντάς την στην παρακάτω εξίσωση

$$(A * 100)/255$$

όπου A η τιμή που μας επιστρέφεται, εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

Η αναπαράσταση της συγκεκριμένης τιμής γίνεται σε % ποσοστό.



Εικόνα 34. Οθόνες εμφάνισης ποσοστού θέσης της πεταλούδας γκαζιού.

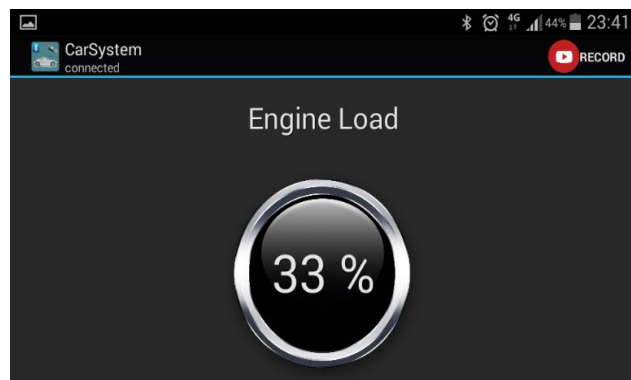
- το επί τοις εκατό ποσοστό φορτίο του κινητήρα (Calculated engine load value). Για τον υπολογισμό του φορτίο του κινητήρα, στέλνεται ο κωδικός 01 04 και μας επιστρέφεται ως απάντηση ένα byte σαν αποτέλεσμα. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και βάζοντας της στην παρακάτω εξίσωση

$$(A * 100)/255$$

όπου A η τιμή που μας επιστρέφεται, εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

Η αναπαράσταση της συγκεκριμένης τιμής γίνεται σε % ποσοστό.

Engine Load      32 %



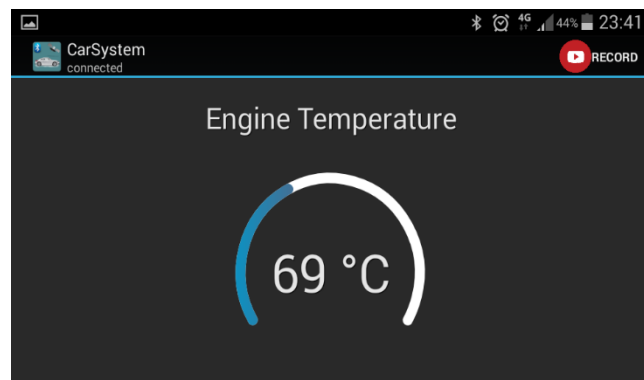
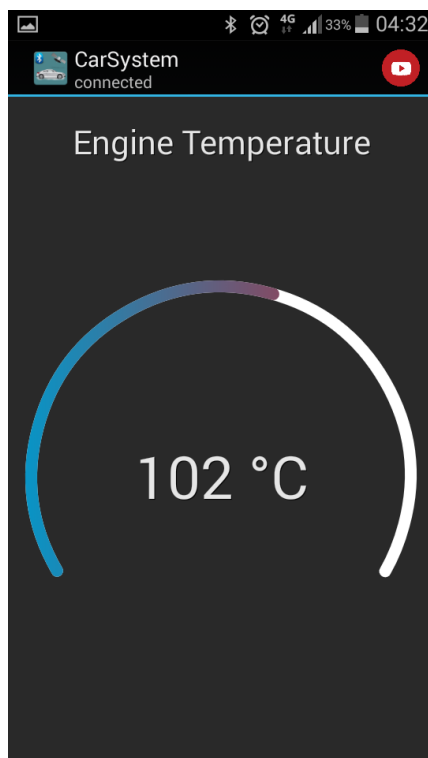
Εικόνα 35. Οθόνες εμφάνισης ποσοστού φορτίου κινητήρα.

- η θερμοκρασία αντιψυκτικού της μηχανής (Engine coolant temperature). Για τον υπολογισμό της θερμοκρασίας αντιψυκτικού της μηχανής, στέλνεται ο κωδικός 01 05 και μας επιστρέφεται ως απάντηση ένα byte σαν αποτέλεσμα. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και βάζοντας της στην παρακάτω εξίσωση

$$A - 40$$

όπου A η τιμή που μας επιστρέφεται, εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα. Η αναπαράσταση της συγκεκριμένης τιμής γίνεται σε °C (βαθμούς κελσίου).

Engine Temperature 102 °C



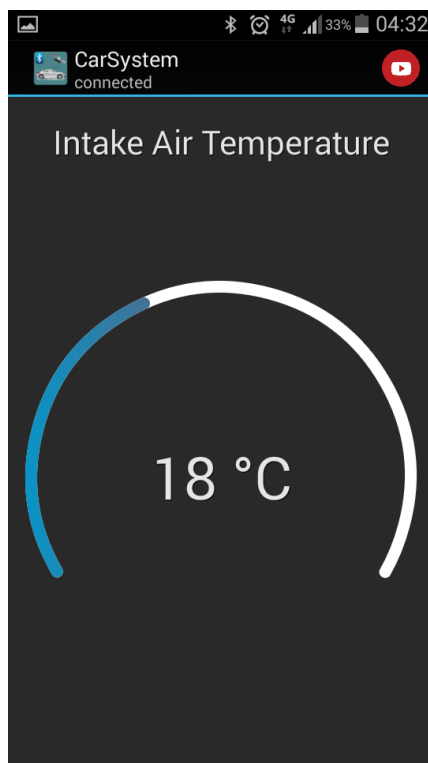
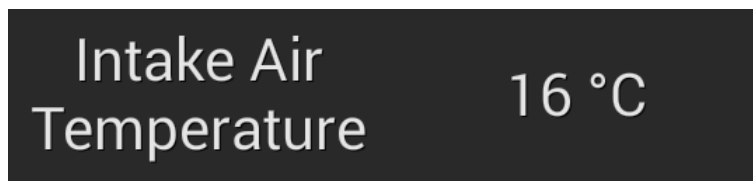
Εικόνα 36. Οθόνες εμφάνισης θερμοκρασίας αντιψυκτικού μηχανής.

- η θερμοκρασία του αέρα εισαγωγής στον κινητήρα (Intake air temperature). Για τον υπολογισμό της θερμοκρασίας του αέρα εισαγωγής στον κινητήρα, στέλνεται ο κωδικός 01 0F και μας επιστρέφεται ως απάντηση ένα byte σαν αποτέλεσμα. Μετατρέπουμε την τιμή αυτή που λάβαμε στο δεκαδικό σύστημα αρίθμησης και βάζοντας την στην παρακάτω εξίσωση

$$A - 40$$

όπου A η τιμή που μας επιστρέφεται, εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

Η αναπαράσταση της συγκεκριμένης τιμής γίνεται σε °C (βαθμούς κελσίου).



Εικόνα 37. Οθόνες εμφάνισης θερμοκρασίας αέρα εισαγωγής.

- η στιγμιαία κατανάλωση καυσίμου (Fuel Consumption). Ο υπολογισμός της στιγμιαίας κατανάλωσης καυσίμου είναι κάπως πιο περίπλοκος. Αρχικά στέλνεται ο κωδικός 01 10, ο οποίος αφορά την τιμή του αισθητήρα μέτρησης της ροής μάζας αέρα στην εισαγωγή ενός κινητήρα εσωτερικής καύσης (MAF air flow rate) και μας επιστρέφονται ως απάντηση δύο byte αυτήν την φορά σαν αποτέλεσμα. Μετατρέπουμε τις δύο τιμές που λάβαμε στο δεκαδικό σύστημα αρίθμησης και τις εισάγουμε στην παρακάτω εξίσωση

$$((A * 256) + B) / 100$$

όπου A η πρώτη κατά σειρά τιμή που μας επιστρέφεται και B η δεύτερη. Το αποτέλεσμα που λαμβάνουμε (MAF) είναι σε grams/sec. Τώρα παίρνοντας αυτό το αποτέλεσμα και βάζοντας το σε μια δεύτερη εξίσωση,

$$\frac{VSS * 710.7}{(MAF * 100)}$$

όπου VSS η στιγμιαία ταχύτητα, μπορούμε να υπολογίσουμε τα απαιτούμενο MPG (Miles per gallon). Τέλος για την μετατροπή της στιγμιαίας κατανάλωσης σε λίτρα ανά εκατό χιλιόμετρα διαιρούμε με την παρακάτω εξίσωση και εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα.

$$235.214/MPG$$

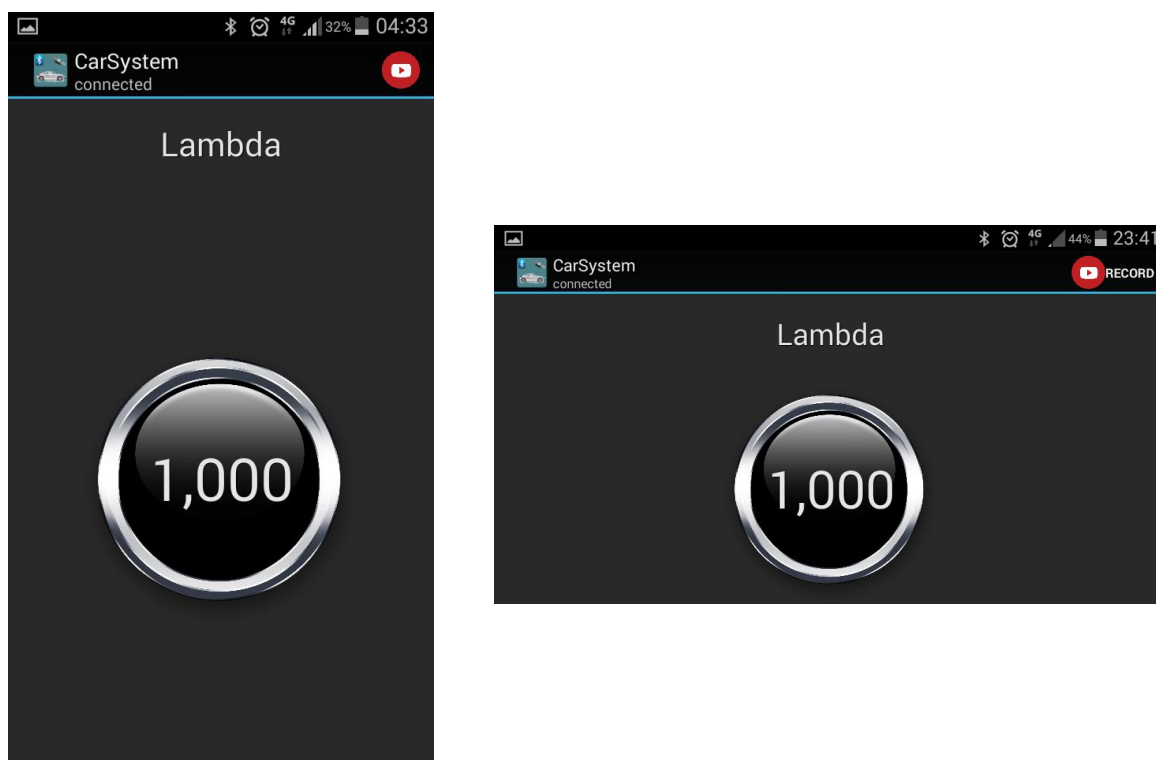
- η τιμή του αισθητήρα λ, ο οποίος φροντίζει ώστε τα ποσοστά των ρύπων στα καυσαέρια να παραμένουν κάτω από τα επιτρεπτά όρια τιμών. Για τον υπολογισμό της τιμής του αισθητήρα λ, στέλνεται ο κωδικός 01 34 και μας επιστρέφονται ως απάντηση τέσσερα byte αυτήν την φορά σαν αποτέλεσμα, όμως τα δύο πρώτα είναι αυτά που χρειαζόμαστε. Μετατρέπουμε τις δύο τιμές που λάβαμε στο δεκαδικό σύστημα αρίθμησης και τις εισάγουμε στην παρακάτω εξίσωση

$$((A * 256) + B)/32768$$

όπου A η πρώτη κατά σειρά τιμή που μας επιστρέφεται και B η δεύτερη, και εμφανίζουμε στην οθόνη το τελικό αποτέλεσμα. Η αναπαράσταση της τιμής του αισθητήρα λ είναι καθαρή τιμή.

Lambda

0,000



Εικόνα 38. Οθόνες εμφάνισης τιμής αισθητήρα λ.

Όσον αφορά τώρα το αποτέλεσμα του αισθητήρα λ, η κυριότερη προϋπόθεση για τον περιορισμό των ρύπων στα καυσαέρια, σε κινητήρα που είναι εφοδιασμένος με τριοδικό καταλυτικό μετατροπέα είναι να λειτουργεί ο κινητήρας στη στοιχειομετρική αναλογία ( $\lambda=1$ ) ή με πολύ μικρή απόκλιση (μικρότερη του 1%). Αυτό δεν είναι δυνατό να επιτευχθεί ακόμα και σε κινητήρα, ο οποίος διαθέτει το πλέον σύγχρονο σύστημα ψεκασμού του καυσίμου μείγματος, αν δεν υπάρχει ένα κλειστό σύστημα το οποίο να ρυθμίζει συνέχεια το μείγμα αέρα-καυσίμου, ανάλογα με τις τιμές των ρύπων στα καυσαέρια. Ο προσδιορισμός των τιμών των ρύπων στα καυσαέρια από τον αισθητήρα λ γίνεται με έμμεσο τρόπο. Δηλαδή, δε μετράει απευθείας τις τιμές τους, αλλά τις προσδιορίζει μετρώντας τη συγκέντρωση των μορίων του οξυγόνου, που περιέχονται στα καυσαέρια. Έτσι αν ανιχνεύσει μεγάλη ποσότητα οξυγόνου, αυτό σημαίνει ότι το μείγμα που κάηκε ήταν «φτωχό» ( $\lambda>1$ ), ενώ αν ανιχνεύσει ελάχιστη ως μηδενική ποσότητα οξυγόνου, αυτό σημαίνει ότι το μείγμα που κάηκε ήταν «πλούσιο» ( $\lambda<1$ ). Επειδή λοιπόν, ο αισθητήρας λ μετράει την ποσότητα του οξυγόνου στα καυσαέρια λέγεται και αισθητήρας οξυγόνου. Αρχικά μάλιστα λεγόταν αισθητήρας οξυγόνου αερίων εξαγωγής

#### 5.2.4.1 Καταγραφή Αποτελεσμάτων σε Αρχείο

Όλα τα αποτελέσματα μπορούν να καταγραφούν σε ένα συγκεκριμένο αρχείο που δημιουργείται στην μνήμη της συσκευής απλά με την επιλογή του αντίστοιχου εικονιδίου που εμφανίζεται στην Action Bar της συγκεκριμένης δραστηριότητας αφού η κατάσταση της σύνδεσης είναι connected (συνδεδεμένο).



Εικόνα 39. Rec Button.

Τότε όλα τα στοιχεία που μπορεί να δει ο χρήστης στην συγκεκριμένη οθόνη καθώς και η ακριβής ώρα, αρχίζουν να καταγράφονται παράλληλα σε ένα αρχείο .txt με περίοδο ενός δευτερολέπτου.

Αφού η καταγραφή έχει ενεργοποιηθεί δίνεται η δυνατότητα στο χρήστη σε περίπτωση που θεωρήσει πως κάτι στο αυτοκίνητο υπολειτουργεί, να πατήσει ένα κουμπί "Alert!" (Η επιλογή αυτή εμφανίζεται στην Action Bar της δραστηριότητας με το που αρχίσει η καταγραφή των στοιχείων). Για όση ώρα η επιλογή αυτή είναι ενεργοποιημένη όλες οι τιμές των στοιχείων σημειώνονται στο αρχείο με διαφορετικό τρόπο ώστε να είναι ευδιάκριτο σε όποιον επιθυμεί να τις ελέγξει εκ των υστέρων.



Εικόνα 40.  
Alert Button.



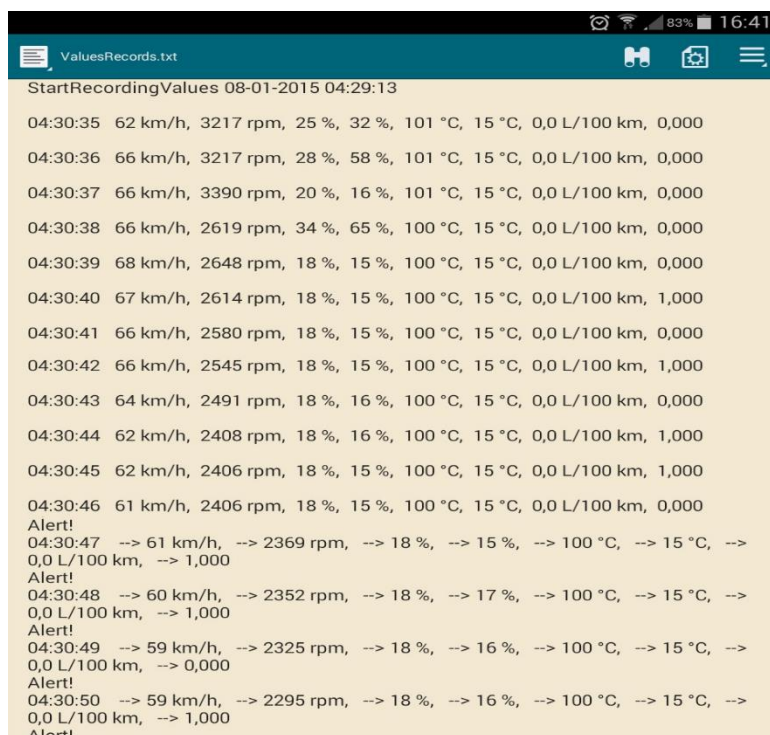
Εικόνα 41.  
Stop Alert Button.

Ο χρήστης μπορεί ανά πάσα στιγμή να σταματήσει την καταγραφή των στοιχείων, πράγμα το οποίο συμβαίνει και αυτομάτως με την έξοδο από την συγκεκριμένη δραστηριότητα.



Εικόνα 42.  
Stop Rec Button.

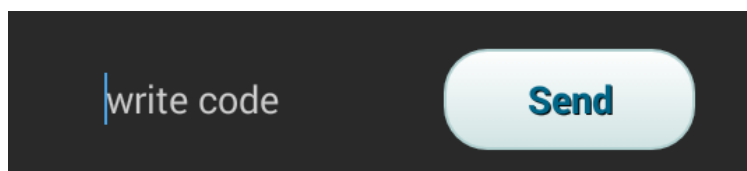
Στην Εικόνα 43 μπορεί κανείς να δει το αρχείο το οποίο η εφαρμογή μας δημιουργεί και περιέχει τα καταγεγραμμένα δεδομένα τα οποία συλλέγονται από τον εγκέφαλο του οχήματος που χρησιμοποιείται.



Εικόνα 43. Παράδειγμα καταγραφής αποτελεσμάτων σε αρχείο.

#### 5.2.4.2 Αποστολή Επιπλέον Κωδικού

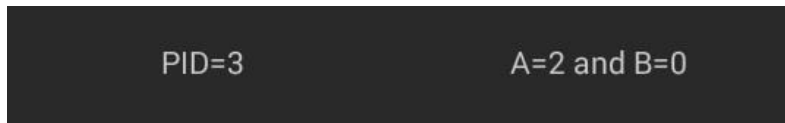
Εκτός από αυτές τις πληροφορίες, ο χρήστης έχει την δυνατότητα να λάβει αποτελέσματα και για όποιο επιπλέον στοιχείο επιθυμεί και είναι διαθέσιμο από τον εγκέφαλο του αυτοκινήτου. Αυτό επιτυγχάνεται, εισάγοντας στο κατάλληλο πεδίο τον κωδικό που επιθυμεί, με την σωστή μορφή που απαιτείται και πατώντας το κουμπί Send.



Εικόνα 44. Πλαίσιο αποστολής επιπλέον κωδικού και κουμπί αποστολής.

Με την επιλογή αυτή ο κωδικός προστίθεται στους ήδη υπάρχοντες κωδικούς που στέλνονται σειριακά. Για παράδειγμα, αν θελήσουμε να δούμε το αποτέλεσμα για το στοιχείο με κωδικό 03 θα πληκτρολογήσουμε 01 03 και θα πατήσουμε Send. Το αντίστοιχο αποτέλεσμα εμφανίζεται στην οθόνη όπως φαίνεται παρακάτω. Αριστερά εμφανίζεται ο κωδικός (PID) για

την εντολή που στάλθηκε και δίπλα οι τιμές που επιστρέφονται (A,B ανάλογα με το πόσα byte επιστρέφονται 1 ή 2).

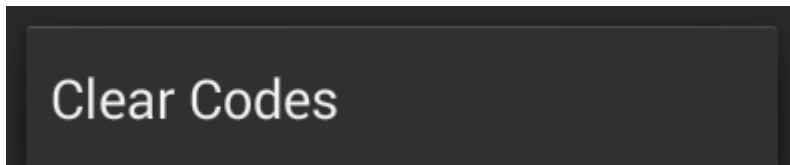


Εικόνα 45. Πλαίσιο λήψης αποτελέσματος επιπλέον κωδικού.

Για να εξάγουμε το τελικό αποτέλεσμα του στοιχείου που στάλθηκε, πρέπει να εισάγουμε τις τιμές που λάβαμε στην κατάλληλη εξίσωση κάθε φορά. Η λειτουργία αυτή ενσωματώθηκε για την περίπτωση που κάποιος χρήστης (μηχανικός, ειδικός) θα ήθελε να εξετάσει περαιτέρω στοιχεία του αυτοκινήτου από τα ήδη υπάρχοντα.

### 5.2.4.3 Εκκαθάριση Κωδικών Σφαλμάτων

Τέλος υπάρχει και μια τελευταία λειτουργία που αφορά το Mode 4 του OBD-II, η οποία ενεργοποιείται πατώντας το πλήκτρο του μενού και ύστερα την επιλογή Clear Codes από τον χρήστη και καθαρίζει τους κωδικούς σφαλμάτων που είναι καταχωρημένοι στην ηλεκτρονική μονάδα ελέγχου (ECU) του αυτοκινήτου. Αυτό επιτυγχάνεται στέλνοντας τον κωδικό 04 στο δίκτυο του αυτοκινήτου.

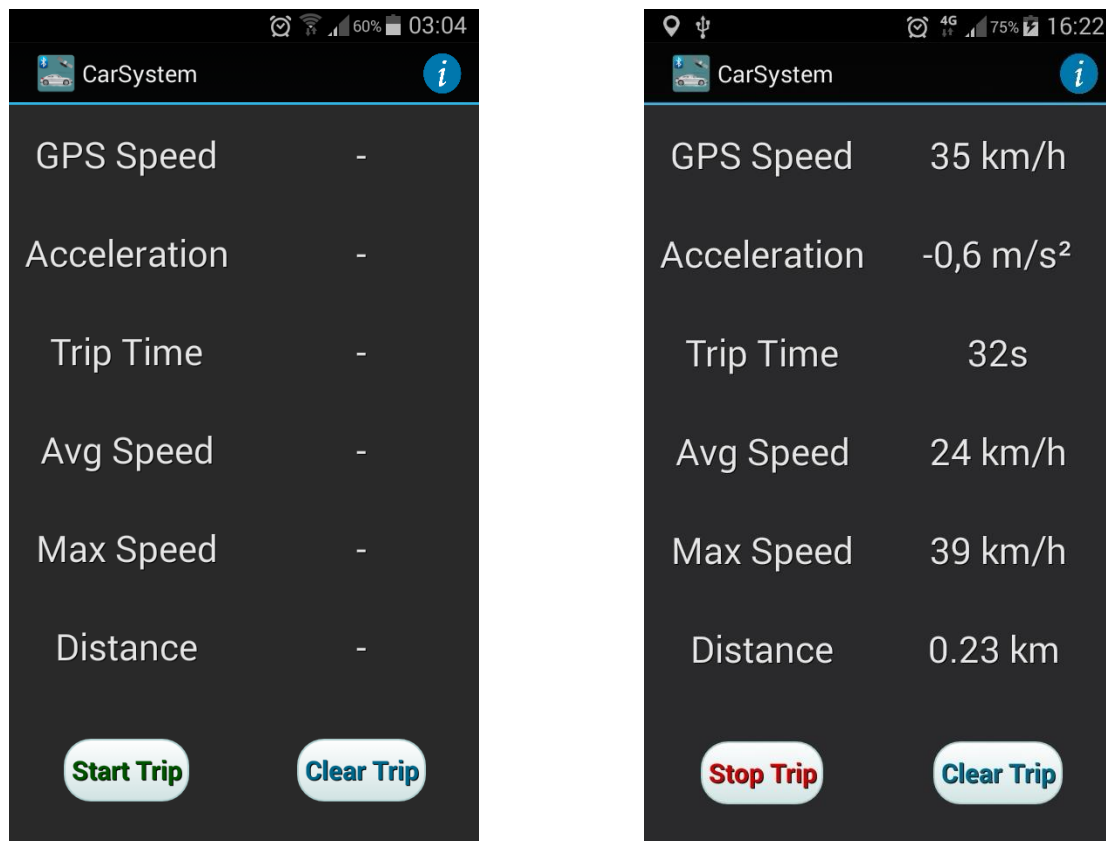


Εικόνα 46. Επιλογή για εκκαθάριση κωδικών σφαλμάτων.

### 5.2.5 GPS\_Basics

Επιλέγοντας την δεύτερη επιλογή από το κεντρικό μενού της εφαρμογής μας, το κομμάτι δηλαδή που αφορά την λήψη δεδομένων πραγματικού χρόνου από το GPS, εμφανίζεται στον χρήστη η οθόνη της εικόνας 47.

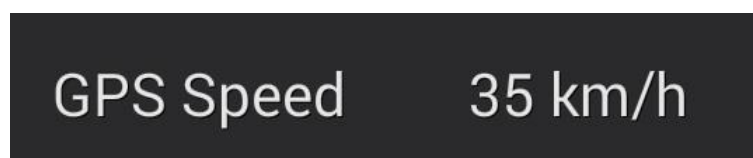
Αρχικά για την σωστή λειτουργία της δραστηριότητας αυτής θα πρέπει να είναι ενεργοποιημένο το GPS, αλλιώς σε αντίθετη περίπτωση εμφανίζεται αντίστοιχο μήνυμα. Κατά την εκκίνηση της δραστηριότητας και αφού έχει ενεργοποιηθεί το GPS, παίρνουν τιμές και εμφανίζονται στην οθόνη κάποια στοιχεία που έχουμε ορίσει. Η ανανέωση των στοιχείων αυτών πραγματοποιείται κάθε δευτερόλεπτο όπως το έχουμε ορίσει.



Εικόνα 47. Οθόνες στοιχείων από το GPS.

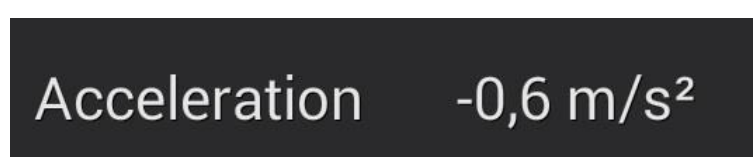
Κάθε χρονική στιγμή εμφανίζονται τα εξής στοιχεία:

- ταχύτητα. Μας παρέχεται κάθε χρονική στιγμή από το GPS και την εμφανίζουμε σε χιλιόμετρα ανά δευτερόλεπτο.



Εικόνα 48. Πλαίσιο αποτελέσματος ταχύτητας.

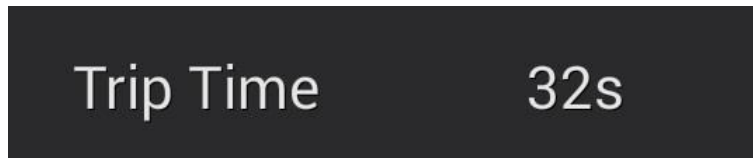
- στιγμιαία επιτάχυνση. Υπολογίζεται από την διαφορά της ταχύτητας δύο διαφορετικών χρονικών στιγμών και για την εμφάνισή της, σε μέτρα ανά δευτερόλεπτο εις στο τετράγωνο, διαιρούμε την προηγούμενη διαφορά με το 3,6.



Εικόνα 49. Πλαίσιο αποτελέσματος επιτάχυνσης.

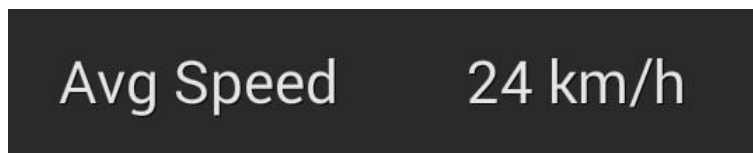
Επίσης σε περίπτωση που η επιλογή start trip (εκκίνηση ταξιδιού) έχει ενεργοποιηθεί παίρνουν τιμές και τα υπόλοιπα στοιχεία που υπάρχουν στην οθόνη, όπως είναι:

- χρόνος ταξιδιού. Με την ενεργοποίηση του start trip ξεκινάει μια συνάρτηση η οποία μετατρέπει τα χιλιοστά δευτερολέπτου σε αναγνώσιμη μορφή ώρας για τον χρήστη, ώστε να εμφανίζουμε τον ακριβή χρόνο ταξιδιού στον χρήστη με ακρίβεια δευτερολέπτου.



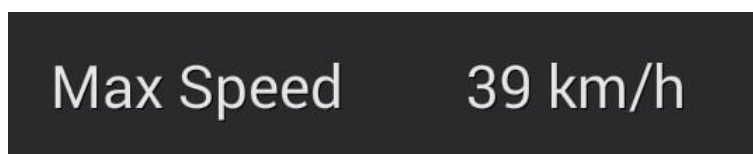
Εικόνα 50. Πλαίσιο αποτελέσματος διάρκειας ταξιδιού.

- μέση ταχύτητα. Υπολογίζεται παίρνοντας την συνολική απόσταση που έχουμε διανύσει σε μέτρα και διαιρώντας την ως προς τον συνολικό χρόνο που χρειαστήκαμε σε δευτερόλεπτα, αφού πρώτα πολλαπλασιάσουμε με 3,6 ώστε να την εμφανίζουμε σε χιλιόμετρα ανά δευτερόλεπτο.



Εικόνα 51. Πλαίσιο αποτελέσματος μέσης ταχύτητας.

- μέγιστη ταχύτητα. Κάθε χρονική στιγμή συγκρίνουμε την στιγμιαία ταχύτητα με την μέγιστη ταχύτητα (την οποία έχουμε αρχικοποιήσει στην τιμή μηδέν) και αν είναι μεγαλύτερη τότε ορίζουμε ως μέγιστη ταχύτητα την υπάρχουσα τιμή της στιγμιαίας ταχύτητας. Την μέγιστη ταχύτητα την εμφανίζουμε και αυτήν σε χιλιόμετρα ανά δευτερόλεπτο.



Εικόνα 52. Πλαίσιο αποτελέσματος μέγιστης ταχύτητας.

- συνολική απόσταση. Χρησιμοποιώντας μια συνάρτηση που υπολογίζει τη διαφορά σε μέτρα, δύο διαδοχικών τοποθεσιών που έχουμε λάβει από το GPS, παίρνουμε το αποτέλεσμα που θέλουμε. Κάθε δευτερόλεπτο που πραγματοποιείται η ανανέωση της τοποθεσίας μας, προσθέτουμε τη νέα διαφορά που υπολογίζεται στην συνολική απόσταση που έχουμε διανύσει και την εμφανίζουμε σε χιλιόμετρα με την ακρίβεια δύο δεκαδικών στοιχείων, δηλαδή δέκα μέτρων.



Εικόνα 53. Πλαίσιο αποτελέσματος συνολικής απόστασης.

Ο χρήστης έχει ακόμα την δυνατότητα να πατήσει την επιλογή stop trip, η οποία σταματάει τον υπολογισμό των στοιχείων ταξιδιού, μέχρις ότου η επιλογή start trip ενεργοποιηθεί και πάλι. Τέλος υπάρχει και η επιλογή clear trip, η οποία καθαρίζει τα μέχρι πρότινος δεδομένα και ο χρήστης μπορεί να ξαναρχίσει την καταγραφή τους από την αρχή για την δημιουργία νέου ταξιδιού.

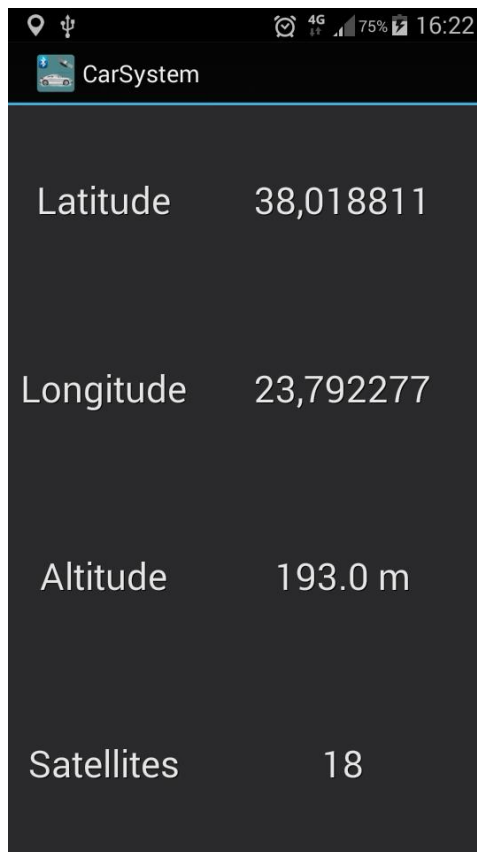
### 5.2.6 GPS\_Info

Πατώντας το εικονίδιο πληροφοριών που εμφανίζεται στην Action Bar της προηγούμενης δραστηριότητας, στο πάνω δεξιά μέρος της οθόνης, εμφανίζονται κάποιες ακόμα πληροφορίες που αφορούν στην τοποθεσία της.



Εικόνα 54. GPS Info Button.

Τέτοιες είναι το γεωγραφικό πλάτος και μήκος, το υψόμετρο που βρισκόμαστε αλλά και οι συνδεδεμένοι δορυφόροι από τους οποίους λαμβάνουμε τις συγκεκριμένες πληροφορίες. Τα τρία πρώτα δεδομένα τα παίρνουμε κατευθείαν από το GPS και τα εμφανίζουμε στην οθόνη, όπως φαίνεται παρακάτω στην εικόνα. Το γεωγραφικό πλάτος και μήκος μας εμφανίζονται με ακρίβεια 6 δεκαδικών, δηλαδή περίπου 10 εκατοστών, ενώ το υψόμετρο που βρισκόμαστε με ακρίβεια ενός μέτρου. Όσον αφορά τους συνδεδεμένους δορυφόρους, υπολογίζονται με την βοήθεια μιας ειδικής συνάντησης, στην οποία προσθέτουμε όλους τους δορυφόρους που μας βοήθησαν στην εξαγωγή των αποτελεσμάτων που εμφανίζονται στην οθόνη του χρήστη.



Εικόνα 55. Οθόνη εμφάνισης πληροφοριών GPS.

### 5.3 Πιθανά Προβλήματα

Η εφαρμογή έχει υλοποιηθεί ώστε να υποστηρίζει μεγάλο μέρος των συσκευών με λειτουργικό σύστημα Android, καθώς επίσης να είναι συμβατή σε όσο το δυνατόν περισσότερες οθόνες, όσον αφορά τις διαστάσεις αλλά και την ανάλυση της. Παρόλα αυτά παρουσιάζονται κάποια πιθανά προβλήματα όπως:

- η λανθασμένη εμφάνιση των γραφικών στοιχείων σε κάποιο συγκεκριμένο είδος οθόνης.
- κάθε όχημα δεν υποστηρίζει όλους τους κωδικούς για την διάγνωση ελέγχου του αυτοκινήτου. Έτσι ενδέχεται κάποιο στοιχείο στην οθόνη που αφορά το κομμάτι του Bluetooth και της συλλογής δεδομένων πραγματικού χρόνου, να μην λαμβάνει κάποια τιμή, καθώς δεν είναι διαθέσιμο από τον εγκέφαλο του αυτοκινήτου.
- το κομμάτι της σύνδεσης της συσκευής Android με το διαγνωστικό εργαλείο ELM327. Υπάρχουν μερικές περιπτώσεις που η σύνδεση μεταξύ των δύο αυτών συσκευών δεν υλοποιείται επιτυχώς ή επιτυγχάνεται δίχως να συλλέγονται τα ζητούμενα δεδομένα στην οθόνη.

Σε περίπτωση που το ενδεχόμενο αυτό συμβεί, για την επίλυση του θα πρέπει ακολουθήσουμε τα παρακάτω βήματα:

1. Τερματισμό της εφαρμογή μας.
2. Απενεργοποίηση του Bluetooth.
3. Αποσύνδεση του προσαρμογέα ELM327 από την θύρα του αυτοκινήτου.
4. Επανασύνδεση του προσαρμογέα
5. Επανεναρξη της εφαρμογής
6. Ενεργοποίηση του Bluetooth
7. Σύνδεση των δύο συσκευών



## Κεφάλαιο 6

### Επιβεβαίωση Λειτουργίας

Στο κεφάλαιο αυτό αναφέρονται τα βήματα που ακολουθήθηκαν για την επιβεβαίωση της ορθής λειτουργία της εφαρμογής. Αρχικά επιβεβαιώνεται ξεχωριστά κάθε ένα από τα κομμάτια της υλοποίησης, ενώ στην συνέχεια επιβεβαιώνονται οι λειτουργίες της ολοκληρωμένης εφαρμογής με όλα τα συστατικά μέρη της συνενωμένα.

#### 6.1 Επιβεβαίωση Λειτουργίας Bluetooth

Αρχικά η υλοποίηση της εφαρμογής μας ξεκίνησε με την προσπάθεια σύνδεσης της συσκευής μας με λειτουργικό σύστημα Android και του μικροελεγκτή ELM327 μέσω Bluetooth. Το πρώτο πράγμα που επαληθεύθηκε ήταν η ενεργοποίηση και απενεργοποίηση του Bluetooth της συσκευής και στην συνέχεια η επιβεβαίωση συνεχίστηκε με τις λειτουργίες που αφορούν την εμφάνιση των ήδη συζευγμένων συσκευών καθώς και την αναζήτηση για νέες συσκευές.

Αφού επιλέχθηκε η συσκευή OBD-II που επιθυμούμε να συνδεθούμε, από την παραπάνω λίστα για νέες συσκευές, τότε πραγματοποιείται και η προσπάθεια σύνδεσης των δύο συσκευών. Αυτό είναι και το αμέσως επόμενο πράγμα, για το οποίο επαληθεύσαμε την λειτουργία του. Για να συμβεί αυτό τοποθετήσαμε την συσκευή ELM327 στην κατάλληλη υποδοχή του αυτοκινήτου, ανοίξαμε την εφαρμογή μας, ενεργοποιήσαμε το Bluetooth και πατήσαμε την επιλογή για αναζήτηση νέων συσκευών. Τα αποτελέσματα της αναζήτησης εμφανίστηκαν στην οθόνη μας και επιλέγοντας την συσκευή που θέλουμε να συνδεθούμε επιβεβαίωσαμε την σωστή επικοινωνία των δύο συσκευών.

#### 6.2 Επιβεβαίωση Εμφάνισης Στοιχείων Μηχανής

Αφού ολοκληρώθηκε ο έλεγχος για την επιτυχή σύνδεση των δύο συσκευών το αμέσως επόμενο κομμάτι που υλοποιήσαμε ήταν η προσπάθεια προβολής, αρχικά, ενός στοιχείου από τον εγκέφαλο του αυτοκινήτου στην οθόνη του κινητού.

Το πρώτο στοιχείο που προσπαθήσαμε να αντλήσουμε από τον εγκέφαλο ήταν οι στροφές του κινητήρα. Ο λόγος που επιλέχθηκε αυτό το στοιχείο ήταν ότι για την επαλήθευση του δεν χρειαζόταν να πραγματοποιήσουμε βόλτα με το αυτοκίνητο παρά μόνο να

ενεργοποιήσουμε τον κινητήρα και να συγκρίνουμε τις τιμές που λαμβάνουμε με αυτές του στροφόμετρου. Έτσι λοιπόν αφού η ενημέρωση μας για το πρώτο στοιχείο ήταν επιτυχής και επαληθεύθηκε από το στροφόμετρο του αυτοκινήτου, αρχίσαμε να αντλούμε και άλλες πληροφορίες από τον εγκέφαλο, προσθέτοντας επιπλέον κωδικούς.

Τα αμέσως επόμενα στοιχεία που καταφέραμε να εμφανίσουμε ήταν η ταχύτητα με την οποία κινείται το όχημα και το επί τοις εκατό ποσοστό της θέσης πεταλούδας γκαζιού. Η σωστή τιμή της ταχύτητα επαληθεύτηκε πραγματοποιώντας επίσης δοκιμές και συγκρίνοντας την με το κοντέρ του αυτοκινήτου, καθώς επίσης και αργότερα με την αντίστοιχη τιμή της που λαμβάναμε από το GPS.

Με αυτό τον τρόπο επαληθεύθηκε η σωστή άντληση πληροφοριών για στοιχεία της μηχανής τα οποία επιστρέφουν 1 byte (ταχύτητα) αλλά και 2 bytes (στροφές κινητήρα). Έτσι και τα υπόλοιπα αποτελέσματα που λαμβάνονται είναι σωστά καθώς δεν στέλνονται λανθασμένα από τον εγκέφαλο του αυτοκινήτου και το μόνο που πρέπει να προσέξουμε είναι η σωστή μετατροπή και εμφάνιση τους.

### 6.3 Επιβεβαίωση Καταγραφής Στοιχείων

Έπειτα την επαληθευμένα σωστή εμφάνιση των στοιχείων πραγματικού χρόνου στην οθόνη της συσκευής μας, υλοποιήθηκε η λειτουργία καταγραφής τους σε αρχείο στην μνήμη της συσκευής μας.

Η λειτουργία αυτή δεν ήταν δύσκολο να επαληθευτεί καθώς, σε περίπτωση που η επιλογή της καταγραφής είναι ενεργοποιημένη, κάθε δευτερόλεπτο καταγράφουμε τις τιμές των στοιχείων σε αρχείο στην μνήμη της συσκευής μαζί με την ακριβή ώρα. Έτσι λοιπόν πραγματοποιώντας μια απλή βόλτα με το όχημα μας και κατά την διάρκεια της κάποια στιγμή ενεργοποιηθεί η καταγραφή των δεδομένων, μπορεί να επαληθευτεί η ορθότητα τους ανοίγοντας απλά το αρχείο και μελετώντας τις καταγεγραμμένες τιμές που περιέχει.

### 6.4 Επιβεβαίωση Λειτουργίας GPS

Μετά την επιτυχή ολοκλήρωση και επιβεβαίωση των τμημάτων της εφαρμογής που αφορούν το Bluetooth, ασχοληθήκαμε με το κομμάτι εκείνο της εφαρμογής που αφορά την λήψη δεδομένων πραγματικού χρόνου από το GPS της συσκευής μας.

Το πρώτο πράγμα που ελέγξαμε ήταν η σωστή συλλογή και εμφάνιση των συντεταγμένων που λαμβάνουμε, οι οποίες είναι το γεωγραφικό μήκος και πλάτος. Αφού ελέγξαμε λοιπόν ότι οι γεωγραφικές συντεταγμένες εμφανίζονται επιτυχώς στην οθόνη

ακολούθησε η προσθήκη και η εμφάνιση και άλλων στοιχείων από το GPS, όπως είναι το υψόμετρο, η ταχύτητα με την οποία κινούμαστε καθώς και την απόσταση που διανύουμε.

Όσον αφορά την τιμή που εμφανίζεται για το υψόμετρο, μπορεί να παρατηρηθεί μια μικρή απόκλιση από την πραγματική τιμή, πράγμα το οποίο συμβαίνει λόγω του βαθμού δυσκολίας του υπολογισμού της. Η σωστή εμφάνιση της ταχύτητας επιβεβαιώθηκε και από το κοντέρ του αυτοκινήτου αλλά και από την τιμή της ταχύτητας που λαμβάνουμε από τον εγκέφαλο του αυτοκινήτου. Για την επαλήθευση της τιμής της απόστασης που διανύουμε βοηθηθήκαμε και πάλι από το καντράν του αυτοκινήτου διανύοντας μεγάλες αποστάσεις.

Τέλος οι υπόλοιπες τιμές, όπως η μέση ταχύτητα, η μέγιστη ταχύτητα και η στιγμιαία επιτάχυνση αποτελούν αποτελέσματα πράξεων των προηγούμενων στοιχείων και η ορθότητα τους επιτεύχθηκε ύστερα από πολλές δοκιμές και συγκρίσεις σε αυτοκίνητα που διαθέτουν υπολογιστή ταξιδιού με τα συγκεκριμένα στοιχεία.

## 6.5 Τελική Δοκιμή

Μετά την ολοκλήρωση της υλοποίησης, ακολούθησαν εκτενείς δοκιμές για την συμπεριφορά της ολοκληρωμένης εφαρμογής με όλα τα συστατικά μέρη συνενωμένα. Μετά από τις δοκιμές, η εφαρμογή δόθηκε σε φίλους και φοιτητές του πολυτεχνείου για αξιολόγηση. Αφού δοκίμασαν και χρησιμοποίησαν εκτενώς την εφαρμογή, τους ζητήθηκε να την σχολιάσουν. Η γνώμη τους βοήθησε στην περαιτέρω βελτιστοποίηση της εφαρμογής, όχι μόνο στο γραφικό αλλά και στο λειτουργικό κομμάτι, καθώς η εφαρμογή δοκιμάστηκε σε διαφορετικές εκδόσεις Android αλλά και σε διαφορετικές οθόνες με βάση την ανάλυση και το μέγεθός τους.

Η τελική δοκιμή της εφαρμογής πραγματοποιήθηκε κατά την διάρκεια της διαδρομής Αθήνα – Ναύπλιο, όπου όλα κύλισαν ομαλά και τα αποτελέσματα θα τα δούμε παρακάτω.

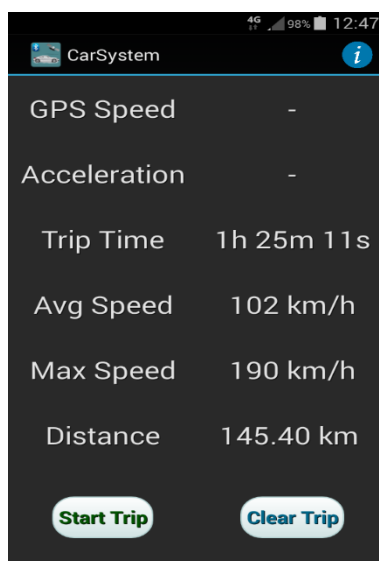
Αρχικά αφού ενεργοποιήσαμε το Bluetooth συνδεθήκαμε επιτυχώς με τον προσαρμογέα ELM327 που βρίσκεται συνδεδεμένος στην OBD-II θύρα του αυτοκινήτου και τα δεδομένα πραγματικού χρόνου της μηχανής άρχισαν να εμφανίζονται στην οθόνη μας. Όσον αφορά το πρώτο κομμάτι του Bluetooth, κατά την διάρκεια της διαδρομής δοκιμάστηκαν εκτενώς και οι λειτουργίες που ενσωματώνει και αφορούν την αποστολή ενός περαιτέρω στοιχείου καθώς και την καταγραφή των στοιχείων σε αρχείο. Στην συνέχεια ενεργοποιήσαμε και το GPS για την εύρεση της ακριβούς τοποθεσίας μας και αρχίσαμε τον υπολογισμό στοιχείων του ταξιδιού μας. Η παράλληλη σωστή λειτουργία και των δύο επιμέρους κομματιών της εφαρμογής μας επαληθεύθηκε καθ' όλη την διάρκεια του ταξιδιού. Τα περισσότερα στοιχεία που περιλαμβάνονται στην υλοποίηση μας επαληθεύθηκαν από τα όργανα που υπάρχουν στο καντράν του αυτοκινήτου. Για το λόγω αυτό χρησιμοποιήθηκε και συγκεκριμένο

αυτοκίνητο, το οποίο εμφανίζει πολλές πληροφορίες της μηχανής, καθώς και περιλαμβάνει υπολογιστή για τον υπολογισμό παραμέτρων ταξιδιού.

Αξίζει να σημειωθεί, πως η εφαρμογή που υλοποιήθηκε, συνεχίζει κανονικά την λειτουργία της ακόμα και στην περίπτωση που χρησιμοποιηθεί κάποια άλλη εφαρμογή της συσκευής ή διακοπεί προσωρινά λόγω εισερχόμενης κλήσης.

Η μόνη περίπτωση να διακοπεί απρόσμενα κάποια λειτουργία της, είναι όταν εκτελούμε μια άλλη εφαρμογή της συσκευής και δεν υπάρχουν επαρκείς διαθέσιμοι πόροι για να διασφαλιστεί η σταθερότητα της σύμφωνα με το λειτουργικό σύστημα Android. Στην περίπτωση αυτή τυχόν δεσμευμένοι πόροι από τις εφαρμογές που βρίσκονται στο παρασκήνιο θα ελευθερωθούν και αυτό μπορεί να σημαίνει αστάθεια στην σύνδεση Bluetooth ή ακόμα και τερματισμό των δραστηριοτήτων της εφαρμογής. Για αυτό το λόγο, καλό θα ήταν πριν την έναρξη της εφαρμογής που υλοποιήσαμε, να κλείνουμε όλες τις εφαρμογές που δεν μας χρειάζονται. Έτσι θα αποφευχθούν παρόμοιες συμπεριφορές.

Αποτελέσματα της δοκιμής θα παραθέσουμε μόνο για το κομμάτι που αφορά το GPS, καθώς το κομμάτι του Bluetooth εμφανίζει μόνο δεδομένα πραγματικού χρόνου. Η εμφάνιση της καταγραφής τους από το αρχείο, κατά την διάρκεια της διαδρομής, θα ήταν δύσκολο να παρουσιαστεί, διότι τα στοιχεία ήταν πάρα πολλά και δεν θα είχε ιδιαίτερο νόημα.



Εικόνα 56. Εμφάνιση αποτελεσμάτων διαδρομής

Πιο συγκεκριμένα, παρατηρώντας τα αποτελέσματα της διαδρομής βλέπουμε τα στοιχεία που εμφανίζονται στην παραπάνω εικόνα. Η διαδρομή διήρκεσε περίπου μία ώρα και είκοσι πέντε λεπτά, με μέση ταχύτητα τα εκατόν δύο χιλιόμετρα ανά ώρα και μέγιστη τα εκατόν ενενήντα χιλιόμετρα ανά ώρα, ενώ η συνολική απόσταση που διανύσαμε ξεπερνά για λίγο τα εκατόν σαράντα πέντε χιλιόμετρα. Η ορθότητα των αποτελεσμάτων για ορισμένα από τα στοιχεία, όπως η μέση ταχύτητα και η συνολική απόσταση, επιβεβαιώθηκαν και από το καντράν του αυτοκινήτου.

## Κεφάλαιο 7

### Συμπεράσματα - Μελλοντικές Επεκτάσεις

#### 7.1 Συμπεράσματα

Με την ραγδαία ανάπτυξη των έξυπνων κινητών συσκευών, δίνεται η δυνατότητα για άμεση επικοινωνία και αλληλεπίδραση του χρήστη με το αυτοκίνητο, οποιαδήποτε χρονική στιγμή. Οι κινητές συσκευές πλέον έχουν αυξημένη υπολογιστική ισχύ και διαθέτουν πολλές δυνατότητες, ενώ η δημοτικότητα τους ανεβαίνει μέρα με την μέρα.

Στην παρούσα διπλωματική εργασία παρουσιάστηκε η υλοποίηση της εφαρμογής CarSystem. Ακόμα αναλύθηκε η μοντελοποίηση και η αρχιτεκτονική του συστήματος, καθώς επίσης τα εργαλεία και οι βιβλιοθήκες που χρειάστηκαν για την υλοποίηση της. Για την χρήση της εφαρμογής υπάρχει ένας σημαντικός περιορισμός. Απαιτείται οι χρήστες να έχουν συσκευή με λειτουργικό σύστημα Android 3.0 ή νεότερο, καθώς επίσης η συσκευή τους να διαθέτει οθόνη αφής, Bluetooth αλλά και GPS. Τέλος ο μικροελεγκτής ELM327 αποτελεί απαραίτητο εργαλείο για την χρήση της εφαρμογής.

Μέσω της εφαρμογής αυτής οι χρήστες μπορούν να λαμβάνουν δεδομένα και πληροφορίες πραγματικού χρόνου του αυτοκινήτου και της τοποθεσίας τους. Η εφαρμογή προσφέρει επίσης λειτουργίες όπως:

- Την καταγραφή των δεδομένων του αυτοκινήτου σε αρχείο στην μνήμη της συσκευής, την ειδική καταγραφή σε περίπτωση προβλήματος,
- Τον καθαρισμό όλων των κωδικών σφαλμάτων που είναι καταχωρημένοι στην ηλεκτρονική μονάδα ελέγχου (ECU) του αυτοκινήτου,
- Τον υπολογισμό πληροφοριών ταξιδιού με τη βοήθεια του GPS.

Κάποια χρήσιμα συμπεράσματα τα οποία εξάγαμε κατά την διάρκεια εκπόνησης της εργασίας είναι ότι η εφαρμογή μας είναι συμβατή με τις περισσότερες εκδόσεις Android και με μεγάλο πλήθος οθονών. Ακόμα επιτεύχθηκε παράλληλη λειτουργία για την συλλογή των δεδομένων και από το Bluetooth, αλλά και από το δέκτη GPS της συσκευής και τέλος τα αποτελέσματα, τα οποία συλλέγονται και εμφανίζονται στο χρήστη από το αυτοκίνητο, παρουσιάζουν μεγάλη ακρίβεια λόγω της συχνής ανανέωσης τους.

Κλείνοντας πιστεύουμε πως η εφαρμογή αυτή μπορεί να αποτελέσει ένα σημαντικό εργαλείο όχι μόνο για την απλή πληροφόρηση των χρηστών σχετικά με δεδομένα πραγματικού χρόνου του αυτοκινήτου και της τοποθεσίας της συσκευής τους, αλλά και για τη διάγνωση και πρόληψη τυχόν προβλημάτων του αυτοκινήτου.

### **7.1.1 Απόκτηση Τεχνογνωσίας**

Κατά την διάρκεια εκπόνησης της παρούσας εργασίας, μάθαμε αρκετά χρήσιμα πράγματα. Αρχικά αποκτήσαμε την απαραίτητη τεχνογνωσία που απαιτείται για την σχεδίαση και την υλοποίηση μιας ολοκληρωμένης Android εφαρμογής. Επιπλέον εμβαθύνουμε στην συλλογή δεδομένων σε πραγματικό χρόνο και μέσω Bluetooth από το όχημα μας, αλλά και από το δέκτη GPS της συσκευής μας. Τέλος, μάθαμε αρκετά χρήσιμα πράγματα για το πώς η ηλεκτρονική μονάδα ελέγχου ενός οχήματος συλλέγει δεδομένα από τους διάφορους αισθητήρες και από άλλα υποσυστήματα του, καθώς επίσης και με ποιον τρόπο μπορούμε να επικοινωνήσουμε με αυτήν και να ενημερωθούμε για όλες αυτές τις παραμέτρους.

### **7.1.2 Κατανομή Χρόνου**

Το μεγαλύτερο μέρος του χρόνου που χρειάστηκε για να αναπτυχθεί το σύστημα μας, απαιτήθηκε για την κατανόηση και την υλοποίηση, του τρόπου με τον οποίο πραγματοποιείται η επικοινωνία και η ανταλλαγή των δεδομένων με το δίκτυο του οχήματος, μέχρι την τελική εμφάνιση τους στην οθόνη του χρήστη. Πιο συγκεκριμένα, σημαντικό ήταν να κατανοήσουμε σε βάθος την λογική με την οποία επικοινωνούμε με το όχημα μας, στέλνοντας αρχικά την λειτουργία (1 byte) και τον κωδικό (1 byte) του στοιχείου που θέλουμε να μας επιστραφεί. Το σύστημα ανταποκρίνεται και μας επιστρέφει μια σειρά από bytes ανάλογα με το στοιχείο που επιλέξαμε. Σε αυτό το σημείο πραγματοποιείται φιλτράρισμα στην τιμή που λαμβάνεται για τυχόν κενούς χαρακτήρες και στην συνέχεια γίνεται έλεγχος αν μορφή της απάντηση είναι αποδεκτή σαν αποτέλεσμα. Η απάντηση που λαμβάνεται περιέχει 1 byte που αφορά την λειτουργία που ζητήσαμε, ακόμα ένα για να μας ενημερώσει για ποιο στοιχείο θα μας επιστρέψει την τιμή και τέλος ακολουθεί ένας αριθμός από bytes ως τιμή του στοιχείου που ζητήσαμε. Την τιμή αυτή την επεξεργαζόμαστε ανάλογα με τον κωδικό του στοιχείου που λαμβάνουμε, καθώς για κάθε στοιχείο πραγματοποιείται διαφορετική επεξεργασία πριν την τελική του εμφάνιση του στο χρήστη.

Αντίθετα ο χρόνος για την επικοινωνία και τη συλλογή δεδομένων από το δέκτη GPS της συσκευής ήταν πολύ μικρότερος, καθώς η βιβλιοθήκη Location Manager με τις λειτουργίες που μας παρέχει, απλοποιεί αρκετά τα πράγματα. Μεγάλο χρονικό διάστημα απαιτήθηκε

επίσης για την βελτίωση της διεπαφής του χρήστη με την εφαρμογή μας, μέσω του γραφικού περιβάλλοντος που δημιουργήσαμε, καθώς προτεραιότητα μας εκτός από την ταχύτητα ανανέωσης των δεδομένων, ήταν και η όσο το δυνατόν καλύτερη αλληλεπίδραση του χρήστη με την εφαρμογή μας.

### 7.1.3 Υποστήριξη Διαφορετικών Πρωτοκόλλων και Κεφαλών

Η εφαρμογή που υλοποιήσαμε, θεωρητικά υποστηρίζει όλα τα διαθέσιμα πρωτόκολλα OBD-II που μπορεί να συναντήσει κανείς σε διαφορετικά οχήματα, όμως μόνο για το πρωτόκολλο CAN έχει επαληθευτεί η ορθή λειτουργία του. Για την επιλογή του πρωτοκόλλου που χρησιμοποιείται κάθε φορά, η εφαρμογή μας στέλνει αντίστοιχο μήνυμα προς τον μικροελεγκτή (“AT SP 0”), ο οποίος με την σειρά του πραγματοποιεί αυτόματη αναζήτηση και με βάση το πρωτόκολλο του οχήματος καθορίζεται η καταλληλότερη επιλογή.

Ο τρόπος που αντλούνται τα δεδομένα είναι ο ίδιος για όλα τα οχήματα. Έτσι λοιπόν το μόνο που διαφέρει και μπορεί να μην υποστηρίζει η εφαρμογή μας είναι η συμβατότητα της, σε περίπτωση που χρησιμοποιηθεί διαφορετικός μικροελεγκτής αντί του ELM327(και του STN1110 που είναι απόλυτα συμβατός με τον ELM327). Σε τέτοια περίπτωση το μόνο που πρέπει να ελεγχθεί στον κώδικα της εφαρμογής μας είναι ο τρόπος με τον οποίο ρυθμίζουμε το πρωτόκολλο που θα χρησιμοποιηθεί κατά την έναρξη της επικοινωνίας των δύο συσκευών. Όλα τα υπόλοιπα τμήματα του κώδικα μας είναι απολύτως συμβατά με όλους τους μικροελεγκτές.

### 7.1.4 Ανάπτυξη νέων εφαρμογών

Με την ολοκλήρωση της παρούσας εφαρμογής έχουν μπει κάποιες σημαντικές υποδομές για την ανάπτυξη και άλλων εφαρμογών. Μερικά παραδείγματα εφαρμογών που μπορούν να υλοποιηθούν βασισμένες στην παρούσα εργασία είναι:

- Η ανάπτυξη εφαρμογής που σκοπό έχει τη καταγραφή της διαδρομής και της ταχύτητας με την οποία κινούνται τα σχολικά λεωφορεία μιας σχολικής μονάδας. Μια τέτοια εφαρμογή θα ήταν χρήσιμη για το έλεγχο της ασφάλειας των παιδιών που μεταφέρονται, αλλά και για το διοικητικό προσωπικό για την καλύτερο έλεγχο των οδηγών.
- Ακόμα μπορούν να υλοποιηθούν εφαρμογές για την παρακολούθηση και τον έλεγχο και άλλων οχημάτων που ανήκουν σε δημόσιες ή ιδιωτικές επιχειρήσεις.

## 7.2 Μελλοντικές Επεκτάσεις

Η εργασία αυτή μπορεί να αποτελέσει σημείο αφετηρίας για μετέπειτα διπλωματικές εργασίες, με στόχο την πρόσθεση περισσότερων λειτουργιών και βελτίωσης της εφαρμογής. Κλείνοντας λοιπόν θα προταθούν ορισμένες βελτιώσεις που μπορούν να γίνουν τόσο στα συστήματα OBD όσο και προτάσεις μελλοντικής έρευνας των διαγνωστικών συστημάτων.

Όσον αφορά το σύστημα OBD, ενώ στο αυτοκίνητο δοκιμής ο υπολογιστής ταξιδιού υπολογίζει τη μέση κατανάλωση, η πληροφορία αυτή δεν είναι διαθέσιμη σε μας. Θα ήταν ευκαίριο τα σύγχρονα συστήματα OBD να παρέχουν πλήρη πληροφόρηση της κατάστασης των οχημάτων, γεγονός που θα απλοποιούσε κατά πολύ την ανάπτυξη εργαλείων διάγνωσης, και θα εξάλειφε πιθανές αποκλίσεις μετρήσεων. Η αυστηροποίηση και ομογενοποίηση λοιπόν των διάφορων προτύπων μόνο θετικά αποτελέσματα θα μπορούσε να έχει στη συμβατότητα και τη διάγνωση των διαγνωστικών εργαλείων και μεθόδων.

Επιπλέον εργασία που αφορά την επέκταση της παρούσας διπλωματικής μπορεί να γίνει σε διάφορους τομείς. Μερικές ιδέες για περαιτέρω μελέτη συνοψίζονται παρακάτω:

1. Η ανάπτυξη λογισμικού για άλλες φορητές ή και σταθερές πλατφόρμες, καθώς ακόμα και η μεταφορά της εφαρμογής σε άλλες γλώσσες προγραμματισμού. Παρότι το Android είναι με διαφορά το δημοφιλέστερο λειτουργικό για κινητές συσκευές, θα ήταν χρήσιμη η υλοποίηση της εφαρμογής και για άλλα λειτουργικά συστήματα.
2. Ακόμα μπορούν να προστεθούν στην υπάρχουσα εφαρμογή λειτουργίες, όπως:
  - Δυνατότητα εμφάνισης στατιστικών των στοιχείων της μηχανής του αυτοκινήτου ως γραφήματα σε συσχέτιση με το χρόνο.
  - Δυνατότητα ταυτόχρονης παρακολούθησης των δεδομένων πραγματικού χρόνου από έναν δεύτερο χρήστη σε κάποιο απομακρυσμένο χώρο.
  - Δυνατότητα αποθήκευσης στοιχείων διαδρομών χρονολογικά σε βάση δεδομένων.
3. Επίσης μπορεί μέσω της χρήσης του GPS και ειδικότερα δίνοντας τη διεύθυνση του προορισμού του, να υπολογίζει την πιο σύντομη για αυτόν διαδρομή και να τον κατευθύνει με φωνητικές οδηγίες. Ταυτόχρονα, θα ενημερώνεται και για το υπόλοιπο της βενζίνης που διαθέτει, οπότε σε περίπτωση που δεν είναι αρκετό για να καλύψει τη διαδρομή του, να μπορεί να πληροφορείται για το πιο κοντινό πρατήριο βενζίνης.
4. Μέσω της μπροστινής κάμερας της συσκευής που χρησιμοποιείται, μπορεί να ελέγχεται αν ο χρήστης έχει ανοιχτά τα μάτια του, δηλαδή είναι συγκεντρωμένος στο

δρόμο, αν τα κλείνει αρκετά συχνά λόγω της συσσωρευμένης κόπωσης ή ακόμα αν τα έχει κλειστά πράγμα που σημαίνει πως πρέπει να τον ειδοποιήσει επειγόντως με κάποιον έντονο ήχο.

5. Σε περίπτωση που οι αισθητήρες κίνησης της συσκευής ανιχνεύσουν μια έντονη δραστηριότητα, όπως για παράδειγμα τη σύγκρουση του οχήματος, να μπορεί να στείλει κάποιο μήνυμα με τις συντεταγμένες του οχήματος, ή ακόμα και να καλέσει κατευθείαν για βοήθεια. Μια τέτοια λειτουργία θα μπορούσε να είναι πολύ χρήσιμη ίσως και σωτήρια σε περίπτωση ατυχήματος.
6. Έλεγχος της ορθής λειτουργίας του συστήματος μας και για άλλες κεφαλές Bluetooth, καθώς λόγω οικονομικών περιορισμών δεν μας δόθηκε η ευκαιρία για την διαπίστωση της συμβατότητας της εφαρμογής μας με αυτές.



## Παράρτημα OBD-II Modes and PIDs

Στο παράρτημα υπάρχουν οι λειτουργίες καθώς και όλοι οι διαθέσιμοι κωδικοί όπως περιγράφονται στο τελευταίο OBD-II πρότυπο SAE J1979

### Modes

Υπάρχουν 10 λειτουργίες (modes of operation). Αυτές είναι οι ακόλουθες:

Mode (hex)	Description
01	Show current data
02	Show freeze frame data
03	Show stored Diagnostic Trouble Codes
04	Clear Diagnostic Trouble Codes and stored values
05	Test results, oxygen sensor monitoring (non CAN only)
06	Test results, other component/system monitoring (Test results, oxygen sensor monitoring for CAN only)
07	Show pending Diagnostic Trouble Codes (detected during current or last driving cycle)
08	Control operation of on-board component/system
09	Request vehicle information
0A	Permanent Diagnostic Trouble Codes (DTCs) (Cleared DTCs)

## PIDs

Ο παρακάτω πίνακας παρουσιάζει όλα τα PIDs OBD-II όπως ορίζονται από το πρότυπο SAE J1979. Η αναμενόμενη απάντηση για κάθε PID δίνεται μαζί με πληροφορίες για το πώς να μεταφράσει την απάντηση σε σημαντικά δεδομένα. Όλα τα PIDs δεν υποστηρίζονται απ' όλα τα οχήματα και δεν μπορεί να υπάρξει κατασκευαστής που να ορίζει διαφορετικά PIDs από αυτά που ορίζονται στο πρότυπο OBD-II.

Αξίζει να αναφέρουμε ότι οι Λειτουργίες 1 και 2 είναι ουσιαστικά ταυτόσημες, εκτός από το ότι η Λειτουργία 1 παρέχει τις τρέχουσες πληροφορίες, ενώ η Λειτουργία 2 παρέχει μια συνοπτική εικόνα των ίδιων δεδομένων που λαμβάνονται την στιγμή που ορίστηκε ο τελευταίος κωδικός βλάβης διαγνωστικού ελέγχου. Οι εξαιρέσεις είναι: PID 01, η οποία είναι διαθέσιμη μόνο στη λειτουργία 1, και PID 02, η οποία είναι διαθέσιμη μόνο στη λειτουργία 2. Εάν η Λειτουργία 2 PID 02 επιστρέφει μηδέν, τότε δεν υπάρχει στιγμιότυπο και όλα τα άλλα δεδομένα της λειτουργίας 2 δεν έχουν νόημα.

Όταν χρησιμοποιούμε κωδικοποιημένα bit, όπως το C4, εννοούμε το τέταρτο bit από την ψηφιολέξη δεδομένων (byte) C. Κάθε bit αριθμείται από το 0 έως το 7, έτσι ώστε το 7 να είναι το πιο σημαντικό bit και το 0 να είναι το λιγότερο σημαντικό bit.

A								B								C								D							
A7	A6	A5	A4	A3	A2	A1	A0	B7	B6	B5	B4	B3	B2	B1	B0	C7	C6	C5	C4	C3	C2	C1	C0	D7	D6	D5	D4	D3	D2	D1	D0

## Mode 01

PID (hex)	Data bytes returned	Description	Min value	Max value	Units	Formula <sup>[a]</sup>
00	4	PIDs supported [01 - 20]				Bit encoded [A7..D0] == [PID \$01..PID \$20]See below
01	4	Monitor status since DTCs cleared. (Includes malfunction indicator lamp (MIL) status and number of DTCs.)				Bit encoded. See below
02	2	Freeze DTC				
03	2	Fuel system status				Bit encoded. See below
04	1	Calculated engine load value	0	100	%	$A * 100 / 255$
05	1	Engine coolant temperature	-40	215	°C	$A - 40$
06	1	Short term fuel % trim—Bank 1	-100 Subtracting Fuel (Rich Condition)	99.22 Adding Fuel (Lean Condition)	%	$(A - 128) * 100 / 128$
07	1	Long term fuel % trim—Bank 1	-100 Subtracting Fuel (Rich Condition)	99.22 Adding Fuel (Lean Condition)	%	$(A - 128) * 100 / 128$
08	1	Short term fuel % trim—Bank 2	-100 Subtracting Fuel (Rich Condition)	99.22 Adding Fuel (Lean Condition)	%	$(A - 128) * 100 / 128$
09	1	Long term fuel % trim—Bank 2	-100 Subtracting Fuel (Rich Condition)	99.22 Adding Fuel (Lean Condition)	%	$(A - 128) * 100 / 128$
0A	1	Fuel pressure	0	765	kPa (gauge)	$A * 3$
0B	1	Intake manifold absolute pressure	0	255	kPa (absolute)	A
0C	2	Engine RPM	0	16,383.75	rpm	$((A * 256) + B) / 4$
0D	1	Vehicle speed	0	255	km/h	A

0E	1	Timing advance	-64	63.5	° relative to #1 cylinder	(A-128)/2
0F	1	Intake air temperature	-40	215	°C	A-40
10	2	MAF air flow rate	0	655.35	grams/sec	((A*256)+B) / 100
11	1	Throttle position	0	100	%	A*100/255
12	1	Commanded secondary air status				Bit encoded. See below
13	1	Oxygen sensors present				[A0..A3] == Bank 1, Sensors 1-4. [A4..A7] == Bank 2...
14	2	Bank 1, Sensor 1: Oxygen sensor voltage, Short term fuel trim	0 -100(lean)	1.275 99.2(rich)	Volts %	A/200 (B-128) * 100/128 (if B==\$FF, sensor is not used in trim calc)
15	2	Bank 1, Sensor 2: Oxygen sensor voltage, Short term fuel trim	0 -100(lean)	1.275 99.2(rich)	Volts %	A/200 (B-128) * 100/128 (if B==\$FF, sensor is not used in trim calc)
16	2	Bank 1, Sensor 3: Oxygen sensor voltage, Short term fuel trim	0 -100(lean)	1.275 99.2(rich)	Volts %	A/200 (B-128) * 100/128 (if B==\$FF, sensor is not used in trim calc)
17	2	Bank 1, Sensor 4: Oxygen sensor voltage, Short term fuel trim	0 -100(lean)	1.275 99.2(rich)	Volts %	A/200 (B-128) * 100/128 (if B==\$FF, sensor is not used in trim calc)
18	2	Bank 2, Sensor 1: Oxygen sensor voltage, Short term fuel trim	0 -100(lean)	1.275 99.2(rich)	Volts %	A/200 (B-128) * 100/128 (if B==\$FF, sensor is not used in trim calc)
19	2	Bank 2, Sensor 2: Oxygen sensor voltage, Short term fuel trim	0 -100(lean)	1.275 99.2(rich)	Volts %	A/200 (B-128) * 100/128 (if B==\$FF, sensor is not used in trim calc)
1A	2	Bank 2, Sensor 3: Oxygen sensor voltage, Short term fuel trim	0 -100(lean)	1.275 99.2(rich)	Volts %	A/200 (B-128) * 100/128 (if B==\$FF, sensor is not used in trim calc)
1B	2	Bank 2, Sensor 4: Oxygen sensor voltage, Short term fuel trim	0 -100(lean)	1.275 99.2(rich)	Volts %	A/200 (B-128) * 100/128 (if B==\$FF, sensor is not used in trim calc)
1C	1	OBD standards this vehicle conforms to				Bit encoded. See below
1D	1	Oxygen sensors present				Similar to PID 13, but [A0..A7] == [B1S1, B1S2, B2S1, B2S2,

						B3S1, B3S2, B4S1, B4S2]
1E	1	Auxiliary input status				A0 == Power Take Off (PTO) status (1 == active) [A1..A7] not used
1F	2	Run time since engine start	0	65,535	seconds	(A*256)+B
20	4	PIDs supported [21 - 40]				Bit encoded [A7..D0] == [PID \$21..PID \$40]See below
21	2	Distance traveled with malfunction indicator lamp (MIL) on	0	65,535	km	(A*256)+B
22	2	Fuel Rail Pressure (relative to manifold vacuum)	0	5177.265	kPa	((A*256)+B) * 0.079
23	2	Fuel Rail Pressure (diesel, or gasoline direct inject)	0	655,350	kPa (gauge)	((A*256)+B) * 10
24	4	O2S1_WR_lambda(1): Equivalence Ratio Voltage	0 0	1.999 7.999	N/A V	((A*256)+B)*2/65535 or ((A*256)+B)/32768 ((C*256)+D)*8/65535 or ((C*256)+D)/8192
25	4	O2S2_WR_lambda(1): Equivalence Ratio Voltage	0 0	2 8	N/A V	((A*256)+B)*2/65535 ((C*256)+D)*8/65535
26	4	O2S3_WR_lambda(1): Equivalence Ratio Voltage	0 0	2 8	N/A V	((A*256)+B)*2/65535 ((C*256)+D)*8/65535
27	4	O2S4_WR_lambda(1): Equivalence Ratio Voltage	0 0	2 8	N/A V	((A*256)+B)*2/65535 ((C*256)+D)*8/65535
28	4	O2S5_WR_lambda(1): Equivalence Ratio Voltage	0 0	2 8	N/A V	((A*256)+B)*2/65535 ((C*256)+D)*8/65535
29	4	O2S6_WR_lambda(1): Equivalence Ratio Voltage	0 0	2 8	N/A V	((A*256)+B)*2/65535 ((C*256)+D)*8/65535
2A	4	O2S7_WR_lambda(1): Equivalence Ratio Voltage	0 0	2 8	N/A V	((A*256)+B)*2/65535 ((C*256)+D)*8/65535
2B	4	O2S8_WR_lambda(1): Equivalence Ratio Voltage	0 0	2 8	N/A V	((A*256)+B)*2/65535 ((C*256)+D)*8/65535
2C	1	Commanded EGR	0	100	%	A*100/255

2D	1	EGR Error	-100	99.22	%	(A-128) * 100/128
2E	1	Commanded evaporative purge	0	100	%	A*100/255
2F	1	Fuel Level Input	0	100	%	A*100/255
30	1	# of warm-ups since codes cleared	0	255	N/A	A
31	2	Distance traveled since codes cleared	0	65,535	km	(A*256)+B
32	2	Evap. System Vapor Pressure	-8,192	8,192	Pa	((A*256)+B)/4 (A and B are two's complement signed)
33	1	Barometric pressure	0	255	kPa (Absolute)	A
34	4	O2S1_WR_lambda(1): Equivalence Ratio Current	0 -128	1.999 127.99	N/A mA	((A*256)+B)/32,768 ((C*256)+D)/256 - 128
35	4	O2S2_WR_lambda(1): Equivalence Ratio Current	0 -128	2 128	N/A mA	((A*256)+B)/32,768 ((C*256)+D)/256 - 128
36	4	O2S3_WR_lambda(1): Equivalence Ratio Current	0 -128	2 128	N/A mA	((A*256)+B)/32768 ((C*256)+D)/256 - 128
37	4	O2S4_WR_lambda(1): Equivalence Ratio Current	0 -128	2 128	N/A mA	((A*256)+B)/32,768 ((C*256)+D)/256 - 128
38	4	O2S5_WR_lambda(1): Equivalence Ratio Current	0 -128	2 128	N/A mA	((A*256)+B)/32,768 ((C*256)+D)/256 - 128
39	4	O2S6_WR_lambda(1): Equivalence Ratio Current	0 -128	2 128	N/A mA	((A*256)+B)/32,768 ((C*256)+D)/256 - 128
3A	4	O2S7_WR_lambda(1): Equivalence Ratio Current	0 -128	2 128	N/A mA	((A*256)+B)/32,768 ((C*256)+D)/256 - 128
3B	4	O2S8_WR_lambda(1): Equivalence Ratio Current	0 -128	2 128	N/A mA	((A*256)+B)/32,768 ((C*256)+D)/256 - 128
3C	2	Catalyst Temperature Bank 1, Sensor 1	-40	6,513.5	°C	((A*256)+B)/10 - 40
3D	2	Catalyst Temperature Bank 2, Sensor 1	-40	6,513.5	°C	((A*256)+B)/10 - 40
3E	2	Catalyst Temperature Bank 1, Sensor 2	-40	6,513.5	°C	((A*256)+B)/10 - 40
3F	2	Catalyst Temperature Bank 2, Sensor 2	-40	6,513.5	°C	((A*256)+B)/10 - 40

40	4	PIDs supported [41 - 60]				Bit encoded [A7..D0] == [PID \$41..PID \$60]See below
41	4	Monitor status this drive cycle				Bit encoded. See below
42	2	Control module voltage	0	65.535	V	$((A*256)+B)/1000$
43	2	Absolute load value	0	25,700	%	$((A*256)+B)*100/255$
44	2	Fuel/Air commanded equivalence ratio	0	2	N/A	$((A*256)+B)/32768$
45	1	Relative throttle position	0	100	%	$A*100/255$
46	1	Ambient air temperature	-40	215	°C	A-40
47	1	Absolute throttle position B	0	100	%	$A*100/255$
48	1	Absolute throttle position C	0	100	%	$A*100/255$
49	1	Accelerator pedal position D	0	100	%	$A*100/255$
4A	1	Accelerator pedal position E	0	100	%	$A*100/255$
4B	1	Accelerator pedal position F	0	100	%	$A*100/255$
4C	1	Commanded throttle actuator	0	100	%	$A*100/255$
4D	2	Time run with MIL on	0	65,535	minutes	$(A*256)+B$
4E	2	Time since trouble codes cleared	0	65,535	minutes	$(A*256)+B$
4F	4	Maximum value for equivalence ratio, oxygen sensor voltage, oxygen sensor current, and intake manifold absolute pressure	0, 0, 0, 0	255, 255, 255, 2550	, V, mA, kPa	A, B, C, D*10
50	4	Maximum value for air flow rate from mass air flow sensor	0	2550	g/s	A*10, B, C, and D are reserved for future use
51	1	Fuel Type				From fuel type table see below
52	1	Ethanol fuel %	0	100	%	$A*100/255$
53	2	Absolute Evap system Vapor Pressure	0	327.675	kPa	$((A*256)+B)/200$
54	2	Evap system vapor pressure	-32,767	32,768	Pa	$((A*256)+B)-32767$

55	2	Short term secondary oxygen sensor trim bank 1 and bank 3	-100	99.22	%	$(A-128)*100/128$ $(B-128)*100/128$
56	2	Long term secondary oxygen sensor trim bank 1 and bank 3	-100	99.22	%	$(A-128)*100/128$ $(B-128)*100/128$
57	2	Short term secondary oxygen sensor trim bank 2 and bank 4	-100	99.22	%	$(A-128)*100/128$ $(B-128)*100/128$
58	2	Long term secondary oxygen sensor trim bank 2 and bank 4	-100	99.22	%	$(A-128)*100/128$ $(B-128)*100/128$
59	2	Fuel rail pressure (absolute)	0	655,350	kPa	$((A*256)+B) * 10$
5A	1	Relative accelerator pedal position	0	100	%	$A*100/255$
5B	1	Hybrid battery pack remaining life	0	100	%	$A*100/255$
5C	1	Engine oil temperature	-40	210	°C	A - 40
5D	2	Fuel injection timing	-210.00	301.992	°	$((A*256)+B)-26,880/128$
5E	2	Engine fuel rate	0	3212.75	L/h	$((A*256)+B)*0.05$
5F	1	Emission requirements to which vehicle is designed				Bit Encoded
60	4	PIDs supported [61 - 80]				Bit encoded [A7..D0] == [PID \$61..PID \$80]See below
61	1	Driver's demand engine - percent torque	-125	125	%	A-125
62	1	Actual engine - percent torque	-125	125	%	A-125
63	2	Engine reference torque	0	65,535	Nm	$A*256+B$
64	5	Engine percent torque data	-125	125	%	A-125 Idle B-125 Engine point 1 C-125 Engine point 2 D-125 Engine point 3 E-125 Engine point 4
65	2	Auxiliary input / output supported				Bit Encoded
66	5	Mass air flow sensor				
67	3	Engine coolant temperature				
68	7	Intake air temperature sensor				

69	7	Commanded EGR and EGR Error				
6A	5	Commanded Diesel intake air flow control and relative intake air flow position				
6B	5	Exhaust gas recirculation temperature				
6C	5	Commanded throttle actuator control and relative throttle position				
6D	6	Fuel pressure control system				
6E	5	Injection pressure control system				
6F	3	Turbocharger compressor inlet pressure				
70	9	Boost pressure control				
71	5	Variable Geometry turbo (VGT) control				
72	5	Wastegate control				
73	5	Exhaust pressure				
74	5	Turbocharger RPM				
75	7	Turbocharger temperature				
76	7	Turbocharger temperature				
77	5	Charge air cooler temperature (CACT)				
78	9	Exhaust Gas temperature (EGT) Bank 1				Special PID. See below
79	9	Exhaust Gas temperature (EGT) Bank 2				Special PID. See below
7A	7	Diesel particulate filter (DPF)				
7B	7	Diesel particulate filter (DPF)				
7C	9	Diesel Particulate filter (DPF) temperature				

7D	1	NOx NTE control area status				
7E	1	PM NTE control area status				
7F	13	Engine run time				
80	4	PIDs supported [81 - A0]				Bit encoded [A7..D0] == [PID \$81..PID \$A0]See below
81	21	Engine run time for Auxiliary Emissions Control Device(AECD)				
82	21	Engine run time for Auxiliary Emissions Control Device(AECD)				
83	5	NOx sensor				
84		Manifold surface temperature				
85		NOx reagent system				
86		Particulate matter (PM) sensor				
87		Intake manifold absolute pressure				
A0	4	PIDs supported [A1 - C0]				Bit encoded [A7..D0] == [PID \$A1..PID \$C0]See below
C0	4	PIDs supported [C1 - E0]				Bit encoded [A7..D0] == [PID \$C1..PID \$E0]See below
C3	?	?	?	?	?	Returns numerous data, including Drive Condition ID and Engine Speed*
C4	?	?	?	?	?	B5 is Engine Idle Request B6 is Engine Stop Request*
<b>PID (hex)</b>	<b>Data bytes returned</b>	<b>Description</b>	<b>Min value</b>	<b>Max value</b>	<b>Units</b>	<b>Formula<sup>[a]</sup></b>

## Βιβλιογραφία

- [1] Stefan Brahler, Analysis of the Android Architecture, Kalsruher Institut fur Technologie, 2010
- [2] Implementation of Location based Services in Android using GPS and Web Services, Manav Singhal<sup>1</sup> , Anupam Shukla<sup>2</sup>, IJCSI International Journal of Computer Science Issues January 2012
- [3] OBD-II Electric Control Fault Diagnostics for Automobiles Based on Bluetooth, Yongqing Ji, YuanMing Gong School of Automotive Engineering, Shanghai University of Engineering Science, Shanghai,China,
- [4] Research and Development of Hybrid Electric Vehicles CAN-Bus Data Monitor and Diagnostic System through OBD-II and Android-Based Smartphones, 2013,Yalian Yang, Baolin Chen, Lin Su, and Datong Qin
- [5] R. Subramanya, Byung K. Yi, User Interfaces for Mobile Content, S, LGE Mobile Research, LG Electronics, April 2006
- [6] Paul Deitel, Harvey Deitel, Abbey Deitel, Michael Morgano 2011. Android for Programmers, An App-Driven Approach. 1st Edition, Prentice Hall, USA.
- [7] Reto Meier 2012. Professional Android 4 Application Development. Updated Edition, John Wiley & Sons, USA
- [8] Android: Java Programming for the New Generation of Mobile Devices. 1st Edition, O'Reilly, USA
- [9] Jason Morris 2011. Android User Interface Development: Beginner's Guide. 1st Edition, Packt Publishing, USA
- [10] Providing Accident Detection in Vehicular Networks Through OBD-II Devices and Android- J. Friesen, 2010, "Learn Java for Android Development", Apress
- [11] W.M. Lee, 2011, "Beginning Android Application Development", Wrox
- [12] M. Murphy, 2011, "Android Programming Tutorials, 3rd Edition", CommonsWare

- [13] J. Morris, 2011, “Android User Interface Development Beginner's Guide”, Pakt Publishingbased Smartphones
- [14] Κωνσταντίνος Καλοβρέκτης, 2012. Βασικές Δομές Ενσωματωμένων Συστημάτων
- [15] Smartphones - Wikipedia, the free encyclopedia,  
<http://en.wikipedia.org/wiki/Smartphone>
- [16] Android Operating System  
[http://en.wikipedia.org/wiki/Android\\_\(operating\\_system\)#Features](http://en.wikipedia.org/wiki/Android_(operating_system)#Features)
- [17] Bluetooth - Wikipedia, the free encyclopedia,  
<http://en.wikipedia.org/wiki/Bluetooth>
- [18] On Board Diagnostics - Wikipedia, the free encyclopedia,  
[http://en.wikipedia.org/wiki/On-board\\_diagnostics](http://en.wikipedia.org/wiki/On-board_diagnostics)
- [19] Global Positioning System - Wikipedia, the free encyclopedia,  
[http://en.wikipedia.org/wiki/Global\\_Positioning\\_System](http://en.wikipedia.org/wiki/Global_Positioning_System)
- [20] OBD-II PIDs - Wikipedia, the free encyclopedia,  
[http://en.wikipedia.org/wiki/OBD-II\\_PIDs](http://en.wikipedia.org/wiki/OBD-II_PIDs)
- [21] ELM327 - Wikipedia, the free encyclopedia,  
<http://en.wikipedia.org/wiki/ELM327>
- [22] Google Android Developers,  
<http://developer.android.com/develop/index.html>
- [23] Stack Overflow,  
<http://www.stackoverflow.com>
- [24] Smartphone OS Market Share,  
<http://www.idc.com/prodserv/smartphone-os-market-share.jsp>
- [25] Oxygen or Lambda Sensor,  
[http://en.wikipedia.org/wiki/Oxygen\\_sensor](http://en.wikipedia.org/wiki/Oxygen_sensor)
- [26] Android GPS, Location Manager Tutorial,  
<http://www.androidhive.info/2012/07/android-gps-location-manager-tutorial/>
- [27] CAN Protocol - Understanding the Controller Area Network Protocol  
<http://www.engineersgarage.com/article/what-is-controller-area-network>

- [28] Reading Real-Time Data  
<http://www.obdsol.com/knowledgebase/obd-software-development/reading-real-time-data>
- [29] <http://developer.android.com/guide/topics/connectivity/bluetooth.html>
- [30] <https://github.com/pkleczko/CustomGauge>
- [31] <https://github.com/ntoskrnl/AndroidWidgets>
- [32] <http://developer.android.com/guide/components/activities.html>
- [33] <http://elmelectronics.com>

Πηγές εικόνων:

- [34] <http://www.thinkvidya.com/a/career-embedded-systems>
- [35] <http://www.wingnity.com/blog/beginners-guide-android-architecture/>
- [36] <http://www.extremetech.com/computing/89870-nasty-windows-bluetooth-vulnerability-nixed>
- [37] <http://www8.garmin.com/aboutGPS/>
- [38] <http://www.vx220.org.uk/forums/topic/135152-obd2-wiring-diagram/>
- [39] <http://www.xenonlights.gr/can-bus.html>
- [40] <http://shopotam.ru/catalog/131185522985-item.html>
- [41] <http://www.obdsol.com/knowledgebase/on-board-diagnostics/what-is-obd/>
- [42] <http://www.portalprogramas.com/java/8-jdk-32bits/>
- [43] <http://openui5.blogspot.com/p/ide.html>
- [44] <http://www.android.pk/blog/general/best-android-apps-development-tools/>
- [45] <http://www.thinkdope.com/illustration/6-major-photoshop-cc-2014-features/>
- [46] <http://www.androidauthority.com/chrome-42-last-version-ice-cream-sandwich-592280/>

- [47] <https://github.com/pires/android-obd-reader>
- [48] <https://play.google.com/store/apps/details?id=org.prowl.torque&hl=el>
- [49] <https://play.google.com/store/apps/details?id=OCTech.Mobile.Applications.OBDLink&hl=el>
- [50] <https://play.google.com/store/apps/details?id=org.obddroidscan>
- [51] <https://play.google.com/store/apps/details?id=com.pnn.obdcardoctor>





ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ  
TECHNICAL UNIVERISTY OF CRETE