



Πανεπιστήμιο Δυτικής Μακεδονίας  
Τμήμα Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών

## Διπλωματική Εργασία

# Σχεδιασμός και Υλοποίηση Συστήματος Τηλεχειρισμού οχημάτων μέσω Wi-Fi με την Χρήση Κάμερας

Design and Implementation of a Wi-Fi Remote Control  
Vehicle Using a Camera

**Πατερνάς Κωνσταντίνος**

Επιβλέπων Καθηγητής: **Δρ. Μηνάς Δασυγένης**

Εργαστήριο Ρομποτικής, Ενσωματωμένων & Ολοκληρωμένων Συστημάτων

Κοζάνη, Οκτώβριος 2020



## Περιεχόμενα

Κατάλογος Εικόνων.....	7
Κατάλογος Πινάκων .....	9
Περίληψη.....	11
Abstract .....	13
Δήλωση Πνευματικών Δικαιωμάτων.....	14
Ευχαριστίες.....	15
Κεφάλαιο 1 <sup>ο</sup> – Εισαγωγή .....	17
1.1 Σημερινή Κατάσταση.....	17
1.2 Σκοπός του Έργου.....	18
1.3 Παρόμοιες Εργασίες.....	19
1.4 Σύνοψη Κεφαλαίου .....	24
Κεφάλαιο 2 <sup>ο</sup> – Θεωρητικό Υπόβαθρο .....	25
2.1 Διαδικτυακός Προγραμματισμός .....	25
2.1.1 HTML .....	25
2.1.2 CSS .....	26
2.1.3 JavaScript .....	27
2.1.4 jQuery .....	28
2.2 Προγράμματα ανάπτυξης λογισμικού .....	29
2.2.1 Πρωτόκολλο δεδομένων χρήστη UDP .....	29
2.2.2 Microsoft Visual Studio.....	30
2.2.3 Visual C# και .NET πλαίσιο .....	30
2.2.4 Δομή AForge.NET.....	31
2.2.5 Arduino IDE.....	32
2.2.6 MIT App Inventor.....	33
2.2.7 NotePad ++.....	33

2.3 Λογισμικό σχεδίασης τυπωμένων κυκλωμάτων (PCB) .....	34
2.3.1 Πλακέτα τυπωμένου Κυκλώματος (PCB) .....	34
2.3.2 KiCad .....	35
2.3.2.1 Επεξεργαστής σχηματικής διάταξης (Eeschema).....	36
2.3.2.2 Επεξεργαστής διάταξης τυπωμένων κυκλωμάτων (Pcb Layout Editor) .....	37
2.3.2.3 Αποτυπώματα (Footprints).....	38
2.4 Μικροελεγκτής ESP32-Cam .....	39
2.5 Ρυθμιστής Υποβιβασμού Τάσης LM2596s (Step Down Voltage Regulator) .....	40
2.6 Οδηγός Κινητήρων L9110s (Motor Driver) .....	40
2.7 Σύνοψη Κεφαλαίου .....	41
Κεφάλαιο 3 <sup>ο</sup> – Δομικά μέρη συστήματος .....	43
3.1 Απαιτήσεις Συστήματος (System Requirements) .....	43
3.2 Περιπτώσεις Χρήσης (Use Cases).....	44
3.3 Υλικό μέρος .....	45
3.3.1 Ενεργειακές Απαιτήσεις (Energy Requirements).....	45
3.4 Υλοποίηση Τυπωμένου Κυκλώματος PCB .....	47
3.4.1 Τελική μορφή πλακέτας PCB.....	51
3.5 Κόστος κατασκευής (Bill Of Materials) .....	53
3.6 Σύνοψη Κεφαλαίου .....	54
Κεφάλαιο 4 – Ανάλυση Λογισμικού .....	55
4.1 Λογισμικό Μικροελεγκτή .....	55
4.1.2 Λειτουργίες Κάμερας.....	57
4.1.3 Κώδικας Μικροελεγκτή .....	58
4.2 Android Εφαρμογή .....	62
4.3 Εφαρμογή Ιστού (Web App).....	66
4.3.1 Κώδικας Ιστοχώρου.....	70
4.4 Εφαρμογή Η/Υ (Desktop App) .....	75

4.4.1 Κώδικας Εφαρμογής (Desktop App).....	77
4.4.2 Τρόπος Επικοινωνίας Εφαρμογής.....	81
4.5 Σύνοψη Κεφαλαίου .....	82
Κεφάλαιο 5 – Επίλογος.....	83
5.1 Σύνοψη.....	83
5.2 Μετρικά Συστήματος.....	84
5.2.1 Μετρικά Οχήματος .....	84
5.2.2 Μετρικά Μικροελεγκτή .....	85
5.2.3 Μετρικά Εφαρμογής Ιστοχώρου .....	85
5.2.4 Μετρικά Εφαρμογής Windows .....	86
5.3 Ανάλυση SWOT.....	86
5.3.1 Δυνατά Σημεία .....	86
5.3.2 Αδυναμίες.....	87
5.3.3 Ευκαιρίες.....	87
5.3.4 Απειλές.....	88
5.4 Ασφάλεια .....	88
5.5 Μελλοντικές Επεκτάσεις .....	88
5.6 Προβλήματα που αντιμετωπίστηκαν.....	89
5.7 Σύνοψη Κεφαλαίου .....	90
Βιβλιογραφία .....	91
Παράρτημα Α – Οδηγίες Εγκατάστασης Εφαρμογής Windows.....	97
Παράρτημα Β – Οδηγίες Εγκατάστασης Εφαρμογής Android.....	101



## Κατάλογος Εικόνων

Εικόνα 1: Ενδεικτική εικόνα πρώτου αντίστοιχου συστήματος .....	19
Εικόνα 2: Ενδεικτική εικόνα δεύτερου αντίστοιχου συστήματος.....	20
Εικόνα 3: Ενδεικτική εικόνα τρίτου αντίστοιχου συστήματος .....	21
Εικόνα 4: Ενδεικτική εικόνα τέταρτου αντίστοιχου συστήματος .....	22
Εικόνα 5: Παράδειγμα HTML.....	26
Εικόνα 6: Παράδειγμα CSS.....	27
Εικόνα 7: Παράδειγμα JavaScript .....	28
Εικόνα 8: Παράδειγμα jQuery.....	28
Εικόνα 9: Βασική Χρήση UDP .....	29
Εικόνα 10: Ενδεικτική εικόνα του Arduino IDE.....	32
Εικόνα 11: Κεντρική Σελίδα KiCad .....	35
Εικόνα 12: Εφαρμογή EESchema .....	36
Εικόνα 13: Εφαρμογή Pcb Layout Editor. ....	37
Εικόνα 14: Επεξεργαστής αποτυπωμάτων.....	38
Εικόνα 15: Μικροελεγκτής ESP32-Cam .....	39
Εικόνα 16: Ρυθμιστής Τάσης LM2596s .....	40
Εικόνα 17: Οδηγός Κινητήρων L9110s.....	41
Εικόνα 18: Κυκλωματική απεικόνιση του συστήματος.....	46
Εικόνα 19: Σχηματικό πλακέτας.....	47
Εικόνα 20: Κύκλωμα Ρυθμιστή Τάσης LM2596s Adj .....	48
Εικόνα 21: Κύκλωμα Ρυθμιστή Τάσης LM2596s 5V .....	49
Εικόνα 22: Κύκλωμα Οδηγού Κινητήρων L9110s.....	50
Εικόνα 23: Απεικόνιση PCB Layout .....	51
Εικόνα 24: 3D Απεικόνιση Πάνω Όψης.....	52
Εικόνα 25: 3D Απεικόνιση Κάτω Όψης .....	52
Εικόνα 26: Συνάρτηση WheelAct .....	58
Εικόνα 27: Επικοινωνία Μικροελεγκτή με Windows Εφαρμογή (Server Side).....	59
Εικόνα 28: Συνάρτηση Μικροελεγκτή (esp_err_t car_handler).....	60
Εικόνα 29: Συνέχεια Συνάρτησης esp_err_t car_handler .....	61
Εικόνα 30: Αρχική Οθόνη Android Εφαρμογής .....	62

Εικόνα 31: Οθόνη Διαθέσιμων Συσκευών.....	63
Εικόνα 32: Σύνδεση σε Δίκτυο.....	64
Εικόνα 33: Διαδικασία Σύνδεσης σε Ασύρματο Δίκτυο.....	65
Εικόνα 34: Αρχική Οθόνη Ιστοχώρου.....	66
Εικόνα 35: Μενού Ρυθμίσεων Κάμερας.....	68
Εικόνα 36: Μετάδοση Εικόνας.....	69
Εικόνα 37: Κομμάτι Κώδικα του Μενού.....	70
Εικόνα 38: Κώδικας Κουμπιών Χειρισμού.....	71
Εικόνα 39: Συνάρτηση Χειρισμού του Οχήματος Car-do.....	72
Εικόνα 40: Συνάρτηση KeydownFunction.....	73
Εικόνα 41: Συνάρτηση KeyupFunction.....	74
Εικόνα 42: Κύρια Οθόνη Εφαρμογής (Desktop App).....	75
Εικόνα 43: Ενεργοποίηση Κάμερας.....	76
Εικόνα 44: Συνάρτηση Σύνδεσης Εφαρμογής (Connection).....	77
Εικόνα 45: Διαχειριστής Γεγονότος Ποντικιού (MouseEventHandler).....	78
Εικόνα 46: Κώδικας για Χειρισμό Από Πληκτρολόγιο.....	79
Εικόνα 47: Κώδικας MJPEG Stream.....	80
Εικόνα 48: Συνάρτηση Request.....	80
Εικόνα 49: Ενδεικτική Εικόνα Εγκατάστασης Εφαρμογής.....	97
Εικόνα 50: Ενδεικτική Εικόνα Εγκατάστασης Εφαρμογής.....	98
Εικόνα 51: Ενδεικτική Εικόνα Εγκατάστασης Εφαρμογής.....	99
Εικόνα 52: Ενδεικτική Εικόνα Εγκατάστασης Εφαρμογής.....	100



## Κατάλογος Πινάκων

Πίνακας 1: Τεχνικά Χαρακτηριστικά ESP32-Cam .....	39
Πίνακας 2: Τεχνικά Χαρακτηριστικά LM2596s .....	40
Πίνακας 3: Τεχνικά Χαρακτηριστικά L9110s.....	41
Πίνακας 4: Κόστος Υλικών Κατασκευής .....	53
Πίνακας 5: Εντολές Κίνησης Οχήματος (Desktop App) .....	56
Πίνακας 6: Εντολές Κίνησης Οχήματος (Web App).....	56
Πίνακας 7: Λειτουργίες Κάμερας.....	57
Πίνακας 8: Πίνακας Εντολών Χειριστηρίου .....	67
Πίνακας 9: Μετρήσεις Οχήματος .....	84
Πίνακας 10: Μετρήσεις Μικροελεγκτή .....	85
Πίνακας 11: Μετρήσεις Κώδικα Εφαρμογής Ιστού .....	85
Πίνακας 12: Μετρικά Εφαρμογής Windows.....	86



## Περίληψη

Στις μέρες μας, σύμφωνα με τα νέα και ευμετάβλητα δεδομένα της εποχής, υπάρχει μια διαρκής και ολοένα αυξανόμενη τάση της χρήσης ποικίλων φορητών συσκευών ή εξαρτημάτων και της διασύνδεσης αυτών μέσω διαδικτύου, το οποίο μετατρέπει την ζωή μας σε αισθητά ευκολότερη. Το διαδίκτυο αυτό ονομάζεται διαδίκτυο των πραγμάτων (Internet of Things IoT) και έχει συμβάλλει σημαντικά σε διάφορους τομείς της καθημερινότητας του ανθρώπου, όπως στο έξυπνο σπίτι, στις βιομηχανικές ενέργειες, στη μεταφορά και τη διανομή αγαθών, σε θέματα σχετικά με την ενέργεια και με τις εφαρμογές, στο δημόσιο τομέα αλλά και σε αυτόν της υγείας.

Στην παρούσα διπλωματική εργασία, σχεδιάστηκε και υλοποιήθηκε ένα σύστημα που επιτρέπει τον χειρισμό οχημάτων μέσω διαδικτύου. Πιο συγκεκριμένα, το όχημα αποτελείται από έναν μικροελεγκτή χαμηλού κόστους και κατανάλωσης με ενσωματωμένο Wi-Fi, Bluetooth, και κάμερα. Η ενσωματωμένη κάμερα του θα παρέχει ζωντανή μετάδοση εικόνας και έτσι ο κάθε χρήστης θα μπορεί να χειριστεί το όχημα από την οθόνη του υπολογιστή του χωρίς να υπάρχει φυσική οπτική επαφή. Για τον καλύτερο και ευκολότερο χειρισμό του οχήματος υλοποιήθηκαν δυο εφαρμογές, μια διαδικτυακή και μια Windows εφαρμογή, όπου ο εκάστοτε χρήστης θα μπορεί επιλέξει όποια από τις δυο επιθυμεί για να χρησιμοποιήσει. Και οι δυο εφαρμογές προσφέρουν στο χρήστη λειτουργίες για την ρύθμιση της κάμερας αλλά και led flash για σημεία με χαμηλότερο φωτισμό. Επίσης, για μεγαλύτερη ευελιξία του συστήματος αναπτύχθηκε Android εφαρμογή η οποία λαμβάνει από τον μικροελεγκτή μέσω Bluetooth τα διαθέσιμα ασύρματα δίκτυα και στην συνέχεια, ο χρήστης μπορεί να επιλέξει και να συνδεθεί σε ένα από αυτά. Τέλος, σχεδιάστηκε και υλοποιήθηκε πλακέτα τυπωμένου κυκλώματος (PCB), η οποία προσαρμόστηκε πάνω στο όχημα με σκοπό να αποφευχθεί η πολλαπλή χρήση καλωδίων, αλλά επίσης και για προσφέρει σταθερότητα και αξιοπιστία σε τυχόν συγκρούσεις.

**Λέξεις κλειδιά:** Διαδίκτυο των Πραγμάτων, Wi-Fi, Προγραμματισμός Διαδικτύου, Διεπαφή προγραμματισμού εφαρμογών (API), Κάμερα, Μικροελεγκτής, Πλακέτα Τυπωμένου Κυκλώματος (PCB), Πελάτης-Εξυπηρετητής (Client-Server), Μηχανοκίνητος Αθλητισμός, Android



## Abstract

Nowadays, according to the new and variable data of the time, there is an ever-increasing trend of using various mobile devices or accessories and connecting them via the internet, which makes our lives much easier. This network is called the Internet of Things (Internet of Things IoT) and has contributed significantly to various areas of human daily life, such as the smart home, industrial operations, the transportation and distribution of goods, energy issues and applications, in the public sector but also in health.

In the present dissertation, a system that allows the operation of vehicles via the Internet was designed and implemented. More specifically, the vehicle consists of a low cost and low power microcontroller with integrated Wi-Fi, Bluetooth, and a camera. Its built-in camera will provide live video transmission so each user will be able to operate the vehicle from their computer screen without any physical eye contact. For the best and easiest handling of the vehicle, two applications were implemented, an online and a Windows application, where the user will be able to choose which of the two he wishes to use. Both applications offer the user functions for adjusting the camera and led flash for points with lower lighting. Also, for greater flexibility of the system, an Android application was developed which receives from the microcontroller via Bluetooth the available wireless networks, and then the user can select and connect to one of them. Finally, a printed circuit board (PCB) was designed and implemented, which was adapted to the vehicle to avoid multiple uses of cables, but also to offer stability and reliability in the event of a collision.

**Keywords:** Internet Of Things, Wi-Fi, Internet Programming, Camera, Microcontroller, PCB board, Motor Sports, Application Programming Interface (API), Client-Server, Android

## Δήλωση Πνευματικών Δικαιωμάτων

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ.3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο «Σχεδιασμός και Υλοποίηση Συστήματος Τηλεχειρισμού οχημάτων μέσω Wi-Fi με την Χρήση Κάμερας» καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Μηνά Δασυγένη, αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

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

Copyright © Πατερνάς Κωνσταντίνος, Μηνάς Δασυγένης, 2020, Κοζάνη

## Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω την οικογένεια μου για την αμέριστη υλική και ηθική συμπαράσταση τους κατά την διάρκεια των σπουδών μου, καθώς και την στήριξη της σε κάθε μου επιλογή.

Στην συνέχεια, θα ήθελα να προσθέσω ότι είμαι ευγνώμων στο κύριο Μηνά Δασυγένη, επιβλέπων καθηγητή της εργασίας μου, για την μεθοδική καθοδήγηση, την συνεργασία, την πολύτιμη βοήθεια του, καθώς και την εμπιστοσύνη που μου έδειξε για την ανάθεση της παρούσας διπλωματικής εργασίας.

Τέλος, ευχαριστώ την φίλη μου που μου συμπαραστάθηκε και με βοήθησε, με όποιον τρόπο μπορούσε.





## Κεφάλαιο 1<sup>ο</sup> – Εισαγωγή

Στο κεφάλαιο αυτό περιγράφεται λεπτομερώς το αντικείμενο της συγκεκριμένης διπλωματικής εργασίας. Αρχικά, αναφέρεται στην υπάρχουσα κατάσταση που επικρατεί καθώς και τις συνθήκες που προκύπτουν από την συνεχή και αδιάκοπη χρήση του διαδικτύου. Στην συνέχεια, παρουσιάζονται οι προκλήσεις και ο σκοπός που οδήγησαν στην υλοποίηση της παρούσης διπλωματικής εργασίας. Επιπλέον, αναλύονται αντίστοιχα συστήματα που έχουν πραγματωθεί από άλλους ερευνητές, ενώ στην συνέχεια συγκρίνονται το κάθε ένα ξεχωριστά με την συγκεκριμένη διπλωματική εργασία και λαμβάνεται ένα ολοκληρωμένο συμπέρασμα. Τέλος, πραγματοποιείται μια συνοπτική παρουσίαση της δομής της εργασίας.

### 1.1 Σημερινή Κατάσταση

Στην εποχή μας η ολοένα και εντονότερη ανάγκη του ανθρώπου για εξέλιξη έχει επιφέρει μεγάλη τεχνολογική ακμή σε πολλούς τομείς στην ζωή του αλλά και στην κοινωνία ευρύτερα. Αρχικά από την επικοινωνία, τον αθλητισμό, την εκπαίδευση, την μετακίνηση, την εργασία, την ψυχαγωγία, και πολλούς άλλους παράγοντες, η ποιότητα ζωής των ανθρώπων έχει μεταβληθεί εξαιρετικά προς το καλύτερο και αυτό καθιστά ως ανάγκη την συνεχή βελτιστοποίηση συσκευών και λογισμικών, που επικοινωνούν μέσω διαδικτύου, υλοποιημένα με τέτοιο τρόπο ώστε να είναι προσβάσιμα και εύκολα διαχειρίσιμα από τον άνθρωπο, επιδιώκοντας να εξυπηρετούν τις ανάγκες και υποχρεώσεις του, αποφορτίζοντας έτσι το πρόγραμμα της καθημερινότητας του, δίνοντας του ελεύθερο χρόνο και βελτιώνοντας την ποιότητα ζωής του.

Ένας τεχνολογικός όρος που έχει αρχίσει σταδιακά να «κυριαρχεί» στη ζωή μας είναι το «Διαδίκτυο των Πραγμάτων» (Internet of Things) [17]. Το Διαδίκτυο των Πραγμάτων είναι μια έννοια που αφορά τα αντικείμενα της καθημερινότητάς μας που ποικίλουν από βιομηχανικές μηχανές, μέχρι φορητές συσκευές που χρησιμοποιούν ενσωματωμένους αισθητήρες για τη συλλογή δεδομένων και την ανάληψη κάποιας δράσης σε αυτά, μέσω της χρήσης του διαδικτύου. Η έννοια «Πραγμάτων» δεν είναι αυστηρά συνδεδεμένη με ορισμένα προϊόντα. Αναφέρεται σε μία ευρεία ποικιλία συσκευών εντελώς διαφορετικά μεταξύ τους, όπως για παράδειγμα αυτοκίνητα με ενσωματωμένους αισθητήρες, κάμερες, κλιματιστικά, φώτα, συστήματα ασφαλείας, smartwatches ακόμα και αυτοκίνητα των οποίων οι περίπλοκοι

αισθητήρες εντοπίζουν αντικείμενα στην πορεία τους. Είναι μερικά από τα πολλά προϊόντα τεχνολογίας. Αναλυτικότερα, συμπεραίνουμε ότι ο εκάστοτε χρήστης μπορεί να ελέγχει αυτές τις συσκευές απομακρυσμένα και ταυτόχρονα να παρακολουθεί σε πραγματικό χρόνο την ανάδραση που εκτελούν, να επικοινωνεί και να ανταλλάσσει δεδομένα με αυτές.

## 1.2 Σκοπός του Έργου

Η συνεχής ανάπτυξη της τεχνολογίας αλλά και οι γρήγοροι ρυθμοί προόδου των τηλεπικοινωνιών έχουν συνδράμει ευρύτερα στο διαδίκτυο και στις υπηρεσίες του και καθιστά αναγκαίο στον άνθρωπο να το συμβουλευεται ολοένα και περισσότερο στην καθημερινότητά του προκειμένου να την κάνει ευκολότερη. Συμπεριλαμβάνοντας τα παραπάνω στοιχεία και μετά από πολλαπλές έρευνες σε διαδικτυακούς τόπους, γεννήθηκε η ιδέα για την δημιουργία ενός συστήματος τηλεχειρισμού οχημάτων μέσω Wi-Fi με την χρήση κάμερας. Βασική προϋπόθεση του συστήματος της παρούσας διπλωματικής εργασίας αποτελεί η ακλόνητη και αδιάκοπη επικοινωνία του εκάστοτε οχήματος με τον χρήστη. Για τον λόγο αυτό χρησιμοποιείται ένας μικροελεγκτής με ενσωματωμένο Wi-Fi και κάμερα, και βρίσκεται σε συνεχή επικοινωνία, ανταλλάσσοντας δεδομένα με μια διαδικτυακή εφαρμογή, καθώς και με μια Windows εφαρμογή.

Η βασική ιδέα και ο σκοπός αυτού του συστήματος, που μας οδήγησαν στον σχεδιασμό και στην υλοποίησή του, είναι προαγωγή του μηχανοκίνητου αθλητισμού μέσα από την οθόνη του υπολογιστή αλλά και με την χρήση του διαδικτύου, να προωθήσει τις ακαταμέτρητες λειτουργίες που δύναται να εκτελέσει βάσει της σταθερής και προγραμματισμένης καθοδήγησής του από τον χρήστη-παίχτη. Η τεχνολογία έχει τον ρόλο ενός υποκατάστατου στους αμέτρητους τομείς στις μέρες μας, συμπεριλαμβανομένου και τις ψυχαγωγίας. Είναι πραγματικά εκπληκτικό συγκριτικά με τον προηγούμενο αιώνα το πόσες εναλλακτικές έχει πλέον ο άνθρωπος στην διάθεσή του για να ψυχαγωγηθεί, όμως αυτήν στην οποία θα εστιάσουμε στην προκειμένη περίπτωση είναι ο αθλητισμός. Όπως είναι προφανές, ο αθλητισμός μέσω του υπολογιστή δεν έχει τα σωματικά οφέλη που προσφέρει η φυσική άσκηση, όμως οι αξίες, τα συναισθήματα και τα μηνύματα, που παρέχει είναι εξίσου χρήσιμα και απηχούν σε βάθος ιδίως στους σημερινούς νέους. Συνεπώς, ο μηχανοκίνητος αθλητισμός μέσω υπολογιστή έχει σχεδιαστεί για να διασκεδάξει με εναλλακτικό τρόπο τους παίκτες οι

οποίοι ενδέχεται να μην έχουν την δυνατότητα να ψυχαγωγηθούν (π.χ. λόγω κάποιας αναπηρίας) και να αποκομίσουν την αθλητική εμπειρία με την τυπική μορφή, μεταδίδοντας συναισθήματα και τις ίδιες θεμελιώδεις αξίες που κρύβονται πίσω από το παιχνίδι, καθώς και τους δίνει την ευκαιρία να διαχειρίζονται το διαδίκτυο σωστότερα και να μαθαίνουν ανακαλύπτοντας καινούριες πλευρές του τεχνολογικού πλαισίου. Όλη αυτή η διαδικασία θα αποδειχτεί ωφέλιμη για ένα μέλλον που το διαδίκτυο θα αντικαταστήσει περαιτέρω ανθρώπινες ενέργειες και η δυνατότητα τηλεχειρισμού του θα γίνει κυριολεκτικά πολύτιμη.

### 1.3 Παρόμοιες Εργασίες

Μετά από μια ενδελεχή έρευνα σε διάφορους διαδικτυακούς τόπους εκπαιδευτικού και τεχνολογικού ενδιαφέροντος, παρουσιάζεται μια λίστα με παρόμοιες εφαρμογές, που εξυπηρετούν τον σκοπό της παρούσης διπλωματικής εργασίας. Παρουσιάζονται διαφορετικοί τρόποι υλοποίησης με διαφορετικές τεχνολογίες και υλικά. Συγκεκριμένα, μαζί με την παρουσίαση τους, γίνεται και μια συνοπτική αναφορά σχετικά με ποιες είναι οι περιπτώσεις που καλύπτουν. Στο τέλος, γίνεται σύγκριση όλων των εφαρμογών με την εφαρμογή που υλοποιήθηκε στην παρούσα διπλωματική εργασία, όπως και ποια είναι τα πλεονεκτήματα έναντι αυτών.



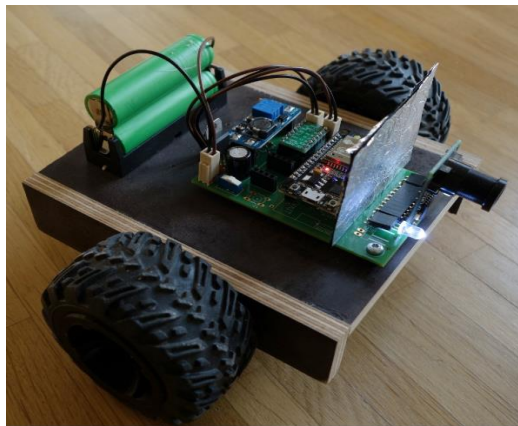
*Εικόνα 1: Ενδεικτική εικόνα πρώτου αντίστοιχου συστήματος*

#### 1) **ESP32-Cam RC Car with Camera and Mobile Phone Controller**

- Περιγραφή: Πρόκειται για μια υλοποίηση [1], η οποία αποτελείται από ένα υλοποιημένο ενσωματωμένο σύστημα που παρέχεται η δυνατότητα χειρισμού ενός τηλεκατευθυνόμενου οχήματος, μέσω ενός μικροελεγκτή. Πιο συγκεκριμένα, το όχημα

επικοινωνεί μέσω Wi-Fi με τον χρήστη, υποβοηθούμενος από έναν ιστότοπο, όπου ταυτόχρονα πραγματοποιείται και live stream μετάδοση. Ο χρήστης με την βοήθεια του προγράμματος περιήγησης (Browser) και της κατάλληλης διεύθυνσης, εισέρχεται στον ιστότοπο, όπου έχει την δυνατότητα να χειριστεί το όχημα. Η συγκεκριμένη εφαρμογή εναρμονίζεται περισσότερο σε συσκευές αφής, μιας και ο χειρισμός του οχήματος γίνεται με την χρήση ενός touch interface. Στόχος της συγκεκριμένης εργασίας είναι ο τηλεχειρισμός οχήματος, μέσω Wi-Fi επικοινωνίας για συσκευές αφής.

- Υλικά hardware: Το σύστημα αποτελείται από έναν μικροελεγκτή τύπου ESP-32 Cam, ο οποίος έχει ενσωματωμένο Wi-Fi, Bluetooth, αλλά και μια κάμερα 2 mega pixel ονομασίας OV2640. Επιπρόσθετα, για τον χειρισμό των κινητήρων, γίνεται χρήση ενός χειριστή κινητήρων συνεχούς ρεύματος (DC Motor Driver) τύπου L298N. Για την πτώση τάσης, χρησιμοποιείται ένας ρυθμιστής υποβιβασμού τάσης συνεχούς ρεύματος (Step-Down Voltage Regulator) τύπου S7V8F3, ενώ για τον έλεγχο του τιμονιού, ένας κινητήρας Servo SG-90 με δυο πλευρικούς διακόπτες ορίου (Limit Switches). Τέλος, για την παροχή ενέργειας, γίνεται χρήση μπαταριών 4xAA Ni-MH 4.8V.



*Εικόνα 2: Ενδεικτική εικόνα δεύτερου αντίστοιχου συστήματος*

## 2) Remote Wi-Fi video Car

- Περιγραφή: Η συγκεκριμένη υλοποίηση [2] αποτελείται από ένα ενσωματωμένο σύστημα, που συντελεί στην κίνηση ενός αυτοσχέδιου οχήματος απομακρυσμένα. Ο χειρισμός του πραγματοποιείται με την χρήση ενός μικροελεγκτή, καθώς και μιας κάμερας. Πιο αναλυτικά, από την πλευρά του χρήστη, η επικοινωνία γίνεται μέσω Wi-Fi, αλλά και μιας Android εφαρμογής,

που μεταδίδει live την εικόνα από το όχημα. Μέσα από το περιβάλλον της εφαρμογής, ο χρήστης μπορεί εύκολα να καθορίσει την πορεία του οχήματος, μέσω ενός touch interface, που έχει υλοποιηθεί κατάλληλα για να εξυπηρετεί τις ανάγκες κίνησης του. Ταυτόχρονα, με την χρήση κουμπιών, μπορεί να ανάψει ένα led λαμπάκι, αλλά και να ενεργοποιήσει ένα φίλτρο, που εμποδίζει το φως υπέρυθρων κατά τη διέλευση ορατού φωτός (IR filter). Σκοπός αυτής της εφαρμογής είναι τηλεχειρισμός, αλλά και η πλοήγηση του οχήματος στον χώρο από συσκευές Android.

- Υλικά hardware: Το σύστημα αποτελείται από έναν μικροελεγκτή τύπου ESP-32, ο οποίος έχει ενσωματωμένο Wi-Fi και Bluetooth. Το εξάρτημα για την μετάδοση εικόνας είναι μια Arducam με ένα τσιπ κάμερας τύπου OV2640. Για την ρύθμιση της τάσης, ένας ρυθμιστής υποβιβασμού τάσης συνεχούς ρεύματος (Step-Down Voltage Regulator). Ένας χειριστής κινητήρων συνεχούς ρεύματος τύπου A4988 (DC Motor Driver).



Εικόνα 3: Ενδεικτική εικόνα τρίτου αντίστοιχου συστήματος

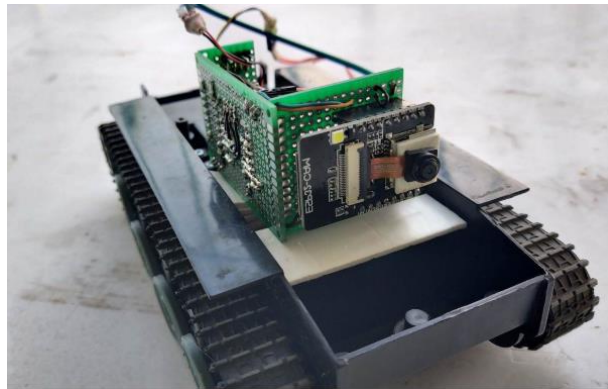
### 3) Wi-Fi RC Car

- Περιγραφή: Μια ακόμη παρόμοια κατασκευή είναι αυτό το ενσωματωμένο σύστημα [3], που παρουσιάζεται στην **Εικόνα 3**. Όπως φαίνεται, αποτελείται από ένα τηλεκατευθυνόμενο όχημα, με ενσωματωμένο έναν μικροελεγκτή, που του παρέχει τον έλεγχο απομακρυσμένα. Αναλυτικότερα, ο χειρισμός του οχήματος από τον χρήστη γίνεται μέσω Wi-Fi επικοινωνίας και συγκεκριμένα, μέσω UDP πρωτοκόλλου, το όχημα επικοινωνεί με μια desktop εφαρμογή, όπου με την χρήση των κατάλληλων

κουμπιών, γίνεται ο χειρισμός του. Ένα ακόμα χαρακτηριστικό του συγκεκριμένου συστήματος είναι το ενσωματωμένο RGB led, όπου αλλάζει χρώμα ανάλογα με την πορεία του οχήματος, καθώς επίσης και η χρήση κόρνας, που επιλέγεται με το πάτημα του κατάλληλου κουμπιού. Τέλος η συγκεκριμένη εφαρμογή, απευθύνεται μόνο στην χρήση από ηλεκτρονικό υπολογιστή, αλλά ο χειρισμός δεν μπορεί να γίνει απομακρυσμένα, μιας και δεν υπάρχει κάμερα.

- Υλικά hardware: Το σύστημα αποτελείται από έναν μικροελεγκτή τύπου ESP-8266, ο οποίος έχει ενσωματωμένο Wi-Fi. Επιπρόσθετα, για τον χειρισμό των κινητήρων, γίνεται χρήση ενός χειριστή κινητήρων συνεχούς ρεύματος (DC Motor Driver) τύπου L298N. Για τον έλεγχο του RGB led, χρησιμοποιείται μια έξυπνη συσκευή ελέγχου RGB led με ονομασία WS2812B. Τέλος, για την λειτουργία της κόρνας, γίνεται χρήση ενός buzzer τύπου HR0288.

#### 4) ESP-32-Cam FPV Arduino Wi-Fi Control Tank



Εικόνα 4: Ενδεικτική εικόνα τέταρτου αντίστοιχου συστήματος

- Περιγραφή: Μια άλλη παρόμοια υλοποίηση, είναι το ενσωματωμένο σύστημα [4], που απεικονίζεται στην **Εικόνα 4** και, όπως φαίνεται, είναι ένα ερπυστριοφόρο όχημα με έναν ενσωματωμένο μικροελεγκτή, που ελέγχει το όχημα. Ειδικότερα, μέσω μιας εφαρμογής και Wi-Fi επικοινωνίας, γίνεται η live μετάδοση εικόνας του οχήματος και ο χειρισμός από τον χρήστη. Χαρακτηριστικό της συγκεκριμένης εφαρμογής είναι η

χρήση led-flash και η πληθώρα ρυθμίσεων, που διαθέτει η κάμερα, με σκοπό την διευκόλυνση του χειρισμού του οχήματος σε όλες τις συνθήκες περιβάλλοντος. Επίσης, μια άλλη δυνατότητα είναι η ρύθμιση της ταχύτητας του οχήματος. Τέλος, η παραπάνω εφαρμογή απευθύνεται σε συσκευές που περιλαμβάνουν οθόνη αφής, μιας και διαθέτει ένα Joystick αφής, για την κατεύθυνση του οχήματος

- Υλικά hardware: Το σύστημα αποτελείται από έναν μικροελεγκτή τύπου ESP-32 Cam, ο οποίος έχει ενσωματωμένο Wi-Fi, Bluetooth, αλλά και μια κάμερα 2 mega pixel ονομασίας OV2640. Επιπλέον, για τον χειρισμό των κινητήρων, γίνεται χρήση ενός χειριστή κινητήρων συνεχούς ρεύματος (DC Motor Driver) τύπου L298N. Για την ρύθμιση της τάσης στον μικροελεγκτή, ένας ρυθμιστής υποβιβασμού τάσης συνεχούς ρεύματος (Step-Down Voltage Regulator). Όλα τα προαναφερόμενα εξαρτήματα είναι συνδεδεμένα πάνω σε μια PCB πλακέτα για εξασφαλισμένη σταθερότητα.

Μελετώντας με προσοχή όλες τις προαναφερθείσες διαδικτυακές εφαρμογές ως προς τις δυνατότητες που προσφέρουν, διαπιστώθηκαν τα εξής συμπεράσματα:

- Όλες οι εφαρμογές για την επικοινωνία, αλλά και αποτελεσματικότερη και ταχύτερη απόκριση του συστήματος, αξιοποιούν την χρήση του Wi-Fi.
- Όλες έχουν ένα περιβάλλον χρήσης κατάλληλα διαμορφωμένο, με τέτοιο τρόπο, ώστε να γίνεται κατανοητό και εύκολα διαχειρίσιμο, ακόμα και από άτομα, τα οποία έχουν ελάχιστες γνώσεις και εμπειρία στην χρήση υπολογιστή, αλλά και φορητών συσκευών αφής.
- Όλες οι εφαρμογές, με εξαίρεση την εφαρμογή ονόματι Wi-Fi RC Car, διαθέτουν κάμερα που κάνει live μετάδοση στην οθόνη του χρήστη, καθώς και flash led για χειρισμό του οχήματος σε σημεία με χαμηλότερο φωτισμό. Επιπρόσθετα, καμία εφαρμογή δεν παρέχει ένδειξη led ανάλογα με την πορεία του οχήματος, σε αντίθεση με την Wi-Fi RC Car που διαθέτει αυτήν την λειτουργία.
- Αντιθέτως με τις υπόλοιπες, η εφαρμογή ESP-32-Cam FPV Wi-Fi Tank παρέχει πολυάριθμη επιλογή ρυθμίσεων στην κάμερα. Εστιάζει δηλαδή, στην καλύτερη ποιότητα εικόνας, για να καλύψει ακόμη και τις πιο δυσμενείς συνθήκες λειτουργίας του συστήματος.

- Όλες οι εφαρμογές, με εξαίρεση την Wi-Fi RC Car, έχουν σταθερές συνδέσεις που είναι προσαρμοσμένες με κολλήσεις στον εξοπλισμό του οχήματος, ή πάνω σε μια πλακέτα PCB.

Έπειτα από τον απολογισμό της ανάλυσης των εφαρμογών που αναφέρθηκαν, αδιαμφισβήτητα, διαπιστώνεται ότι το σύστημα, που αναπτύχθηκε στην συγκεκριμένη διπλωματική εργασία, είναι εξίσου καλό με τα παραπάνω συστήματα, αφού καλύπτει δυναμικά τις δυνατότητες που παρέχουν. Ειδικότερα, το παρόν σύστημα της διπλωματικής έχει το μοναδικό πλεονέκτημα ότι δίνει τη δυνατότητα στους χρήστες να χειριστούν τα οχήματα από δυο εφαρμογές (Desktop, Διαδικτυακή). Ένα ακόμη προνόμιο της συγκεκριμένης εφαρμογής, είναι ότι διαθέτει ρυθμίσεις κάμερας και μπορεί να προσαρμοστεί και σε περιοχές με λιγότερο φωτισμό. Επίσης, είναι η μόνη υλοποίηση που διαθέτει Android εφαρμογή που λειτουργεί μέσω Bluetooth και διευκολύνει την σύνδεση του οχήματος σε κάθε ασύρματο δίκτυο χωρίς να χρειάζεται προγραμματισμό ο μικροελεγκτής. Τέλος, παρέχει χειρισμό και από το πληκτρολόγιο, γεγονός που το καθιστά εύχρηστο και ευχάριστο στην χρήση του.

#### 1.4 Σύνοψη Κεφαλαίου

Στο συγκεκριμένο κεφάλαιο συνοψίστηκαν οι προβληματισμοί που οδήγησαν στην εκπόνηση της συγκεκριμένης διπλωματικής, οι προϋποθέσεις που είναι αναγκαίο να καλύπτονται, τα οφέλη που παρέχονται, καθώς και η δομή του συγκεκριμένου κειμένου. Στην συνέχεια του επόμενου κεφαλαίου, πραγματοποιείται ειδική ανάλυση του θεωρητικού υποβάθρου των τεχνολογιών στις οποίες στηρίζεται η υλοποίηση της διπλωματικής εργασίας. Μετέπειτα αναφέρονται τα εργαλεία προγραμματισμού με την χρήση των οποίων αναπτύχθηκε το λογισμικό και τα προγράμματα σχεδιασμού.



## Κεφάλαιο 2<sup>ο</sup> – Θεωρητικό Υπόβαθρο

Στο προκείμενο κεφάλαιο της διπλωματικής εργασίας, θα παρουσιαστεί η πλήρης ανάλυση δομής, ανάπτυξης και λειτουργίας του αντίστοιχου Πληροφοριακού Συστήματος που υλοποιήθηκε επιτυχώς. Θα γίνει αναφορά στις γλώσσες προγραμματισμού που χρησιμοποιήθηκαν, καθώς και στα λογισμικά εργαλεία μέσω των θεωρητικών εννοιών τους και γενικότερα οποιουδήποτε είδους τεχνολογικό υλικό συντέλεσε στην ολοκληρωμένη μορφή της εφαρμογής.

### 2.1 Διαδικτυακός Προγραμματισμός

Μια από τις πιο σημαντικές χρήσεις της πληροφορικής, είναι η υλοποίηση διαδικτυακών ιστοχώρων. Ο προγραμματισμός διαδικτύου είναι ένας όρος που σχετίζεται με την κατασκευή αυτών. Η ανάπτυξη ενός ιστοχώρου διαφοροποιείται σε πολλούς τομείς που εξαρτώνται από την χρήση του, την απλότητα του, και την αποδοτικότητα του. Όσον αφορά το δομικό κομμάτι του διαδικτυακού προγραμματισμού, υπάρχουν δυο βασικές κατηγορίες που τον χαρακτηρίζουν: ο πελάτης (web client), ο εξυπηρετητής (web server). Ο εξυπηρετητής είναι υπεύθυνος για την εκτέλεση των αιτημάτων που του αναθέτει ο χρήστης. Στον εξυπηρετητή εκτελούνται γλώσσες προγραμματισμού εξυπηρετητή (server-side programming) για την παραγωγή HTML εγγράφων. Ο χρήστης μέσω ενός πελάτη (web client), επικοινωνεί αποστέλλοντας και λαμβάνοντας αιτήματα με τον εξυπηρετητή μέσω ενός προγράμματος περιήγησης (web browser).

#### 2.1.1 HTML

Ο όρος HTML (HyperText Markup Language ή Γλώσσα Σήμανσης Κειμένου) [5] αναφέρεται σε μια βασική γλώσσα σήμανσης που αποτελεί βασικό πυλώνα για την υλοποίηση ιστοχώρων. Η HTML δεν αποτελεί μια γλώσσα προγραμματισμού αλλά ουσιαστικά είναι ένα έγγραφο με συγκεκριμένους κανόνες έτσι ώστε οι περιηγητές (browsers) να εμφανίσουν το

σωστό αποτέλεσμα στους χρήστες. Πιο αναλυτικά, με την χρήση ετικετών (tags), δηλαδή κάποιες λέξεις ή γράμματα που τοποθετούνται μέσα σε γωνιώδεις αγκύλες, διαμορφώνεται η βασική δομή του ιστοχώρου. Στην **Εικόνα 5** που παρουσιάζεται ένα παράδειγμα χρήσης τους.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>All about rabbits!!</title>
  </head>
  <body>

    <h1>All about rabbits!</h1>

    <h2>Basic info</h2>

    <p>My Paragraph</p>

  </body>
</html>
```

*Εικόνα 5: Παράδειγμα HTML*

### 2.1.2 CSS

Ένα ακόμη βασικό συστατικό του διαδικτυακού προγραμματισμού είναι το CSS (Cascading Style Sheets ή Διαδοχικά φύλλα στυλ) [6]. Αναλυτικότερα, το CSS όπως και η HTML εκτελείται από τον περιηγητή του χρήστη, και είναι αυτό που καθορίζει πως εμφανίζονται στον επισκέπτη ενός ιστοχώρου τα διάφορα στοιχεία της HTML. Ένα μεγάλο πλεονέκτημα των CSS είναι ότι μπορούν να αποθηκευτούν σε εξωτερικό αρχείο με κατάληξη .css και να χρησιμοποιηθούν από κοινού σε περισσότερες από μια HTML ιστοσελίδες, εξοικονομώντας με τον τρόπο αυτό χρόνο και κόπο υλοποίησης. Αυτό πρακτικά σημαίνει πως μια αλλαγή σε αυτό το αρχείο θα ενημερώσει όλες τις HTML σελίδες με τις οποίες συνδέεται. Όσον αφορά το δομικό κομμάτι του αποτελείται από τον επιλογέα και μια ομάδα από ορισμούς. Ο επιλογέας μπορεί να είναι μια ετικέτα HTML την οποία θέλουμε να μορφοποιήσουμε ή ένα δικό μας όνομα το οποίο θα το εφαρμόσουμε σε κάποιο σημείο στον κώδικα HTML. Οι ορισμοί δηλώνονται ανάμεσα στα σύμβολα με τα άγκιστρα {}. Κάθε ορισμός αφορά μια ιδιότητα, την

οποία θέλουμε να μεταβάλουμε και την τιμή την οποία παίρνει. Μεταξύ αυτών των δύο χρησιμοποιείται το σύμβολο άνω-κάτω τελεία :. Στο τέλος κάθε ορισμού υπάρχει το ελληνικό ερωτηματικό ;. Μπορούμε να βάλουμε όσους ορισμούς χρειαζόμαστε για τον κάθε έναν επιλογή. Στην **Εικόνα 6** φαίνεται αναλυτικά η λειτουργία του.

```
p {  
  color:blue;  
  text-align:center;  
}  
  
H1 {  
  font-family: "Times New Roman", Times, serif;  
}
```

*Εικόνα 6: Παράδειγμα CSS*

### 2.1.3 JavaScript

Η JavaScript (JS συντομογραφία) [7] είναι μια γλώσσα προγραμματισμού σεναρίου, (Scripting) που τυποποιείται από την ECMA International (ένωση ειδικευμένη στην τυποποίηση των συστημάτων πληροφοριών), με την οποία κάνουμε πολλά πράγματα σε μία ιστοσελίδα. Υποστηρίζεται από όλους τους δημοφιλείς περιηγητές, όπως οι Firefox, Safari, Opera, Google Chrome, κ.τ.λ. Η γλώσσα αυτή είναι client side που σημαίνει ότι εκτελείται και παράγεται το αποτέλεσμα στον browser του επισκέπτη. Επιτρέπει δηλαδή στον επισκέπτη να κάνει κάποια πράγματα δυναμικά σε μία ιστοσελίδα, σε μικρό βαθμό και άμεσα, χωρίς να επικοινωνήσει το σύστημα με τον Server για να εκτελέσει το αίτημά του. Αν και το όνομα της μας παραπέμπει στη Java στην πραγματικότητα, αν εξαιρεθεί ότι οι δυο γλώσσες έχουν

παρόμοια σύνταξη, δεν έχουν κανένα άλλο κοινό μιας και η Java υποστηρίζει κυρίως αντικειμενοστρέφια.

Παρακάτω στην **Εικόνα 7** βλέπουμε ένα script με την λειτουργία της γλώσσας αυτής.

```
<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Paragraph changed.";
}
</script>
```

*Εικόνα 7: Παράδειγμα JavaScript*

#### 2.1.4 jQuery

Η jQuery [8] είναι ένα ελεύθερο λογισμικό ανοικτού κώδικα (open source) και πιο συγκεκριμένα πρόκειται για μια βιβλιοθήκη JavaScript που έχει σχεδιαστεί για να απλοποιεί τη διασταύρωση και τη διαχείριση των σεναρίων HTML, καθώς και τον χειρισμό συμβάντων, όπως η δημιουργία κινούμενης εικόνας (animation) την αλλαγή ιδιοτήτων στην HTML και στο CSS του κώδικα, βάσει των ενεργειών του χρήστη. Τέλος, αξίζει να σημειωθεί ότι η jQuery είναι η πιο δημοφιλής βιβλιοθήκη JavaScript και χρησιμοποιείται ευρέως ακόμα και από τις μεγαλύτερες εταιρείες όπως Google, Microsoft, IBM κλπ.

```
<script>
$(document).ready(function(){
  $("button").click(function(){
    $("p").hide();
  });
});
</script>
```

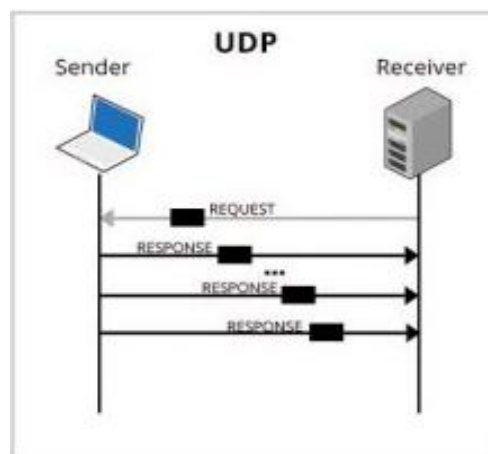
*Εικόνα 8: Παράδειγμα jQuery*

## 2.2 Προγράμματα ανάπτυξης λογισμικού

Η επικοινωνία του οχήματος με τον υπολογιστή επιτυγχάνεται με δύο τρόπους, μέσω ενός ιστοχώρου αλλά και μιας Windows εφαρμογής. Η συγκεκριμένη εφαρμογή επικοινωνεί με το όχημα χρησιμοποιώντας κυρίως το πρωτόκολλο UDP (User Datagram Protocol ή Πρωτόκολλο δεδομένων χρήστη). Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε η γλώσσα προγραμματισμού Visual C# σε συνδυασμό με την χρήση του προγράμματος Visual Studio και άλλων τεχνολογιών όπως η επέκταση .NET, AForge.Video.

### 2.2.1 Πρωτόκολλο δεδομένων χρήστη UDP

Το πρωτόκολλο User Datagram Protocol (UDP) [9] είναι ένα από τα βασικά πρωτόκολλα που χρησιμοποιούνται στο Διαδίκτυο. Ένα από τα κύρια χαρακτηριστικά του UDP είναι η μεγάλη ταχύτητα που διαθέτει για την αποστολή των πακέτων. Όσον αφορά τα πακέτα UDP, που στέλνονται από έναν υπολογιστή σε κάποιον άλλον, υπάρχει μεγάλη πιθανότητα να φτάσουν στον παραλήπτη με λανθασμένο τρόπο, λανθασμένη σειρά, διπλά ή ακόμα να μην παραληφθούν ποτέ, λόγω υπερβολικού φορτίου του δικτύου. Ωστόσο, η μη διάθεση των μηχανισμών αυτών καθιστά το πρωτόκολλο UDP αποτελεσματικότερο και ταχύτερο, όσον αφορά τις εφαρμογές που η αξιόπιστη επικοινωνία μεταξύ των υπολογιστών δεν είναι απαραίτητη. Οι εφαρμογές audio και video streaming χρησιμοποιούν κατά κόρον πακέτα UDP. Για τις εφαρμογές αυτές απαιτείται τα πακέτα που θα παραδοθούν στον παραλήπτη να φτάσουν σύντομα, ώστε να μην υπάρξει διακοπή στην ροή του ήχου ή της εικόνας.



Εικόνα 9: Βασική Χρήση UDP

### 2.2.2 Microsoft Visual Studio

Το Microsoft Visual Studio [10], γνωστό και ως ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) της Microsoft, καθίσταται χρήσιμο για την υλοποίηση προγραμμάτων ηλεκτρονικών υπολογιστών, ιστοσελίδων, εφαρμογών web, διαδικτυακών υπηρεσιών και εφαρμογών για κινητά. Συγκεκριμένα, αξιοποιεί πλατφόρμες ανάπτυξης λογισμικού Microsoft, όπως Windows API, Windows Forms, Windows Presentation Foundation, Windows Store και Microsoft Silverlight. Το Visual Studio είναι κατάλληλα σχεδιασμένο για να διαβάζει 36 διαφορετικές γλώσσες προγραμματισμού και να επιτρέπει στον επεξεργαστή κώδικα και το πρόγραμμα εντοπισμού σφαλμάτων να υποστηρίζει (σε διαφορετικούς βαθμούς) σχεδόν οποιαδήποτε γλώσσα προγραμματισμού, εφόσον η υπηρεσία διαθέτει τη συγκεκριμένη γλώσσα. Η τρέχουσα υποστηριζόμενη έκδοση του Visual Studio 2019 είναι η v16.8.0.

### 2.2.3 Visual C# και .NET πλαίσιο

Η Visual C# [11] είναι μια αντικειμενοστρεφής γλώσσα προγραμματισμού γενικού τύπου που δημιουργήθηκε από την Microsoft. Αρχικά, εμφανίστηκε μέσω της πλατφόρμας .NET με σκοπό τη δημιουργία λογισμικού και έπειτα αναγνωρίστηκε επισήμως από την Ecma (ECMA-334) και την ISO (ISO/IEC 23270:2018). Το .NET Framework είναι ένα πλαίσιο λογισμικού (software framework) που προορίζεται για την πλατφόρμα των Windows. Αποτελούμενο από μια μεγάλη βιβλιοθήκη κλάσεων, υποστηρίζει μια πλειάδα γλωσσών προγραμματισμού με τη δυνατότητα η μια να μπορεί να χρησιμοποιηθεί από την άλλη. Τα προγράμματα που σχεδιάζονται ειδικά για το .NET Framework, εκτελούνται σε ένα περιβάλλον εκτέλεσης ονόματι Common Language Runtime (CLR), ενός λογισμικού, κατάλληλα διαμορφωμένου να υποστηρίζει την εκτέλεση προγραμμάτων και την απρόσκοπτη συνεργασία με το λειτουργικό σύστημα. Το CLR περιλαμβάνει μια εικονική μηχανή (virtual machine) που διαχειρίζεται την εκτέλεση ενός προγράμματος όπως και εξασφαλίζει την παροχή μιας σειράς σημαντικών υπηρεσιών όπως ασφάλεια, διαχείριση μνήμης και διαχείριση εξαιρέσεων. Η γλώσσα C# αποτελεί μια από τις γλώσσες που υποστηρίζει το .NET. Ο μεταγλωττιστής της C# στοχεύει ειδικά στο .NET κάτι που σημαίνει ότι τα προγράμματα γραμμένα σε C# θα τρέχουν πάντα στο .NET Framework. Αυτό έχει δυο σημαντικές συνέπειες για τη γλώσσα:

- Η αρχιτεκτονική της και οι μεθοδολογίες της αντικατοπτρίζουν τη δομή του .NET.
- Σε πολλές περιπτώσεις, ειδικά χαρακτηριστικά της γλώσσας εξαρτώνται στην πραγματικότητα από τα χαρακτηριστικά του .NET ή τις βασικές κλάσεις του .NET

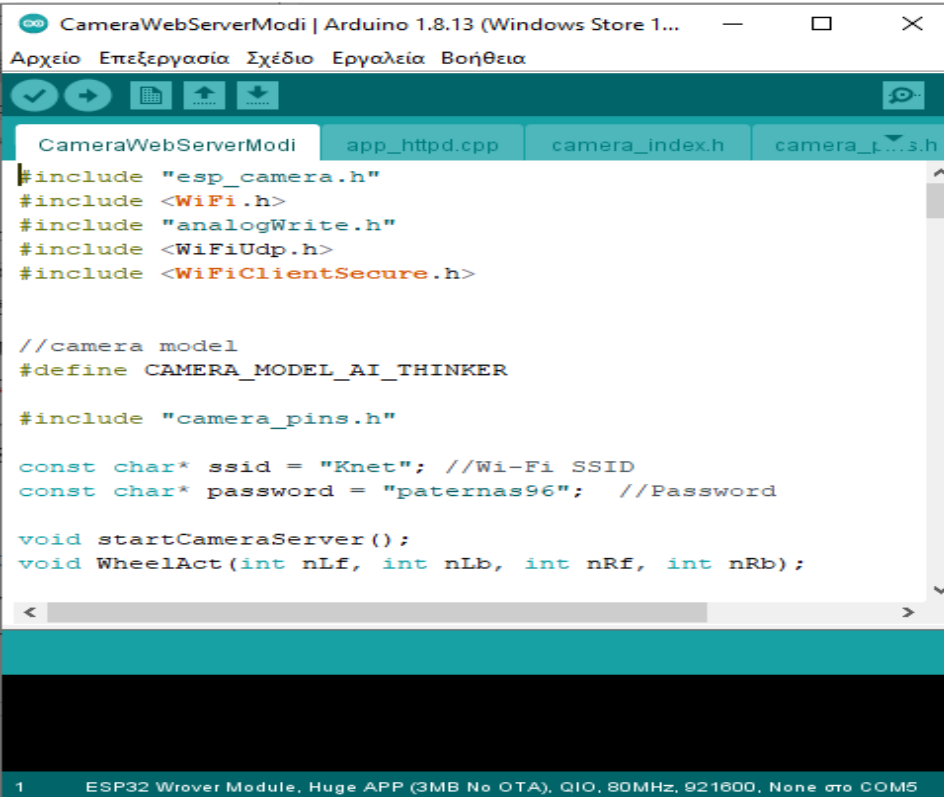
Στην παρούσα διπλωματική εργασία για την υλοποίηση της εφαρμογής ελέγχου του οχήματος χρησιμοποιήθηκε το πλαίσιο .NET v4.0 έτσι ώστε να είναι συμβατό σε πιθανές μελλοντικές επεκτάσεις.

#### 2.2.4 Δομή AForge.NET

Η βιβλιοθήκη AForge [13] είναι μια βασική βιβλιοθήκη οπτικών υπολογιστών (Computer Vision) και τεχνητής νοημοσύνης (Artificial Intelligence) του πλαισίου AForge.NET. Το πλαίσιο αποτελείται από το σύνολο βιβλιοθηκών που καλύπτουν ένα ευρύ φάσμα λειτουργιών επεξεργασία εικόνας, νευρωνικών δικτύων γενετικών αλγόριθμων, ασαφή λογική, μηχανική μάθηση και ρομποτική. Το έργο για τη βελτίωση του πλαισίου βρίσκεται συνεχώς σε εξέλιξη, που σημαίνει ότι νέα χαρακτηριστικά και λειτουργίες προστίθενται συνεχώς. Το πλαίσιο παρέχεται όχι μόνο με διαφορετικές βιβλιοθήκες και τις πηγές τους, αλλά και με πολλά δείγματα εφαρμογών, που αποδεικνύουν τη χρήση αυτού του πλαισίου, και με αρχεία βοήθειας τεκμηρίωσης, τα οποία παρέχονται σε μορφή βοήθειας HTML.

## 2.2.5 Arduino IDE

Το Arduino Integrated Development Environment (IDE) [16] είναι ένα ολοκληρωμένο περιβάλλον που λειτουργεί στα περισσότερα λειτουργικά συστήματα (Windows, macOS, Linux). Χρησιμοποιείται για την σύνταξη, την μεταφόρτωση και την μεταγλώττιση του κώδικα στις πλακέτες Arduino, αλλά επίσης με την εγκατάσταση βιβλιοθηκών τρίτων και σε άλλες πλακέτες ανάπτυξης. Το Arduino IDE υποστηρίζει τις γλώσσες C και C++ χρησιμοποιώντας ειδικούς κανόνες δομής κώδικα. Στην παρούσα υλοποίηση όπως φαίνεται και στην **Εικόνα 10** χρησιμοποιήθηκε η έκδοση v1.8.13.



```
CameraWebServerModi | Arduino 1.8.13 (Windows Store 1...
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια

CameraWebServerModi app_httpd.cpp camera_index.h camera_...s.h
#include "esp_camera.h"
#include <WiFi.h>
#include "analogWrite.h"
#include <WiFiUdp.h>
#include <WiFiClientSecure.h>

//camera model
#define CAMERA_MODEL_AI_THINKER

#include "camera_pins.h"

const char* ssid = "Knet"; //Wi-Fi SSID
const char* password = "paternas96"; //Password

void startCameraServer();
void WheelAct(int nLf, int nLb, int nRf, int nRb);

1 ESP32 Wrover Module, Huge APP (3MB No OTA), QIO, 80MHz, 921600, None στο COM5
```

Εικόνα 10: Ενδεικτική εικόνα του Arduino IDE



### 2.2.6 MIT App Inventor

Το MIT App Inventor [30] είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών που παρέχεται και συντηρείται από το Τεχνολογικό Ινστιτούτο της Μασαχουσέτης (MIT). Χρησιμοποιεί μια γραφική διεπαφή χρήστη (GUI), η οποία επιτρέπει στους χρήστες να σύρουν και να αποθέσουν οπτικά αντικείμενα για να δημιουργήσουν μια εφαρμογή που μπορεί να εκτελεστεί σε συσκευές Android. Το εργαλείο αυτό βασίζεται σε μπλοκ και διευκολύνει τη δημιουργία σύνθετων εφαρμογών υψηλού αντίκτυπου σε πολύ λιγότερο χρόνο από τα παραδοσιακά περιβάλλοντα προγραμματισμού. Στην παρούσα διπλωματική εργασία, χρησιμοποιήθηκε η έκδοση nb185a.

### 2.2.7 NotePad ++

Για την συγγραφή κώδικα της εφαρμογής ιστού, χρησιμοποιήθηκε ο κειμενογράφος Notepad++ v7.8.6. Το Notepad ++ [31] είναι ένα δωρεάν πρόγραμμα επεξεργασίας πηγαίου κώδικα που υποστηρίζει πολλές γλώσσες προγραμματισμού. Εκτελείται σε περιβάλλον Windows, η χρήση του διέπεται από το GNU General Public License. Τέλος, προσφέρει αυτόματη στοίχιση και χρωματισμό στα διάφορα στοιχεία της γλώσσας κάνοντας έτσι την συγγραφή και ανάγνωση του κώδικα ευκολότερη.

## 2.3 Λογισμικό σχεδίασης τυπωμένων κυκλωμάτων (PCB)

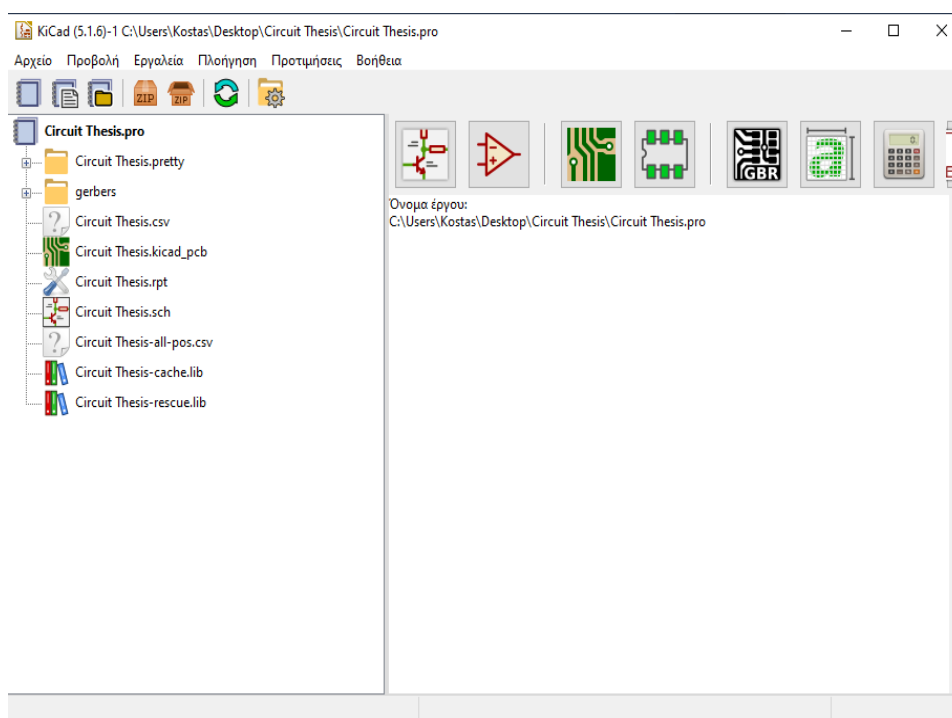
Στην παρούσα διπλωματική εργασία σχεδιάστηκε και υλοποιήθηκε μια πλακέτα τυπωμένου κυκλώματος (PCB) με σκοπό την εξασφάλιση της σταθερότητας των συνδέσεων, αλλά και ολοκλήρου του συστήματος. Για την υλοποίηση και τον σχεδιασμό της πλακέτας χρησιμοποιήθηκε το λογισμικό KiCad το οποίο αναλύεται στην ενότητα αυτή.

### 2.3.1 Πλακέτα τυπωμένου Κυκλώματος (PCB)

Μια πλακέτα τυπωμένου κυκλώματος (PCB) είναι ένα ηλεκτρικό κύκλωμα του οποίου τα εξαρτήματα και οι αγωγοί περιέχονται σε μια μηχανική δομή. Κάθε πλακέτα τυπωμένου κυκλώματος είναι κατασκευασμένη από εναλλασσόμενα στρώματα αγωγίμου χαλκού με στρώματα μη αγωγίμου μονωτικού υλικού. Οι πλακέτες μπορούν να είναι μονής όψης, διπλής όψης ή πολλαπλών στρώσεων. Κατά τη διάρκεια της κατασκευής, τα εσωτερικά στρώματα χαλκού χαρασσονται αφήνοντας προβλεπόμενα ίχνη χαλκού για τη σύνδεση εξαρτημάτων κυκλώματος. Μόλις χαραχτεί το μονωτικό υλικό ελασματοποιείται στα στρώματα χαλκού έως ότου ολοκληρωθεί η πλακέτα τυπωμένου κυκλώματος.

### 2.3.2 KiCad

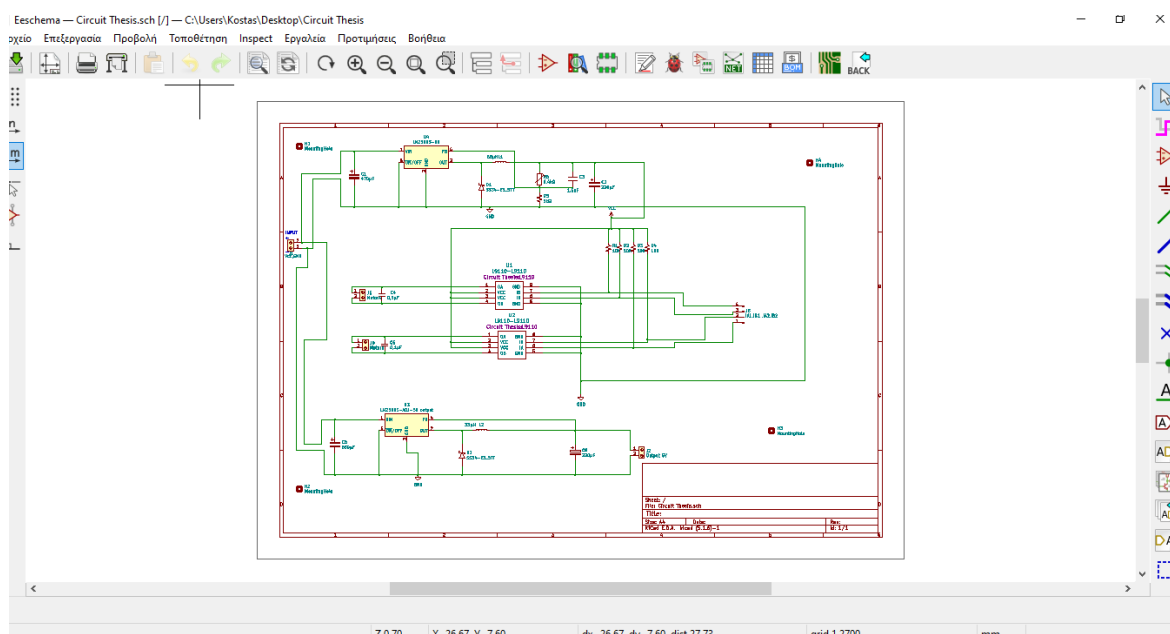
Το KiCad [12] είναι ένα ολοκληρωμένο λογισμικό που διανέμεται δωρεάν και χρησιμοποιείται για τον σχεδιασμό ηλεκτρονικών κυκλωμάτων (EDA-Electronic Design Automation). Το λογισμικό αρχικά κυκλοφόρησε το 1992 από τον Jean-Pierre Charras αλλά τώρα βρίσκεται υπό συνεχή ανάπτυξη από ομάδα προγραμματιστών του KiCad. Με το KiCad είναι εφικτή η σχεδίαση σχηματικών και τυπωμένων ηλεκτρονικών κυκλωμάτων, χρησιμοποιώντας διαφορετικές εφαρμογές. Κάθε έργο αποτελεί και ένα αρχείο .pro όπως φαίνεται στην **Εικόνα 11** στην οποία απεικονίζεται η κεντρική σελίδα του λογισμικού σε συνδυασμό με τις βιβλιοθήκες του.



Εικόνα 11: Κεντρική Σελίδα KiCad

### 2.3.2.1 Επεξεργαστής σχηματικής διάταξης (Eeschema)

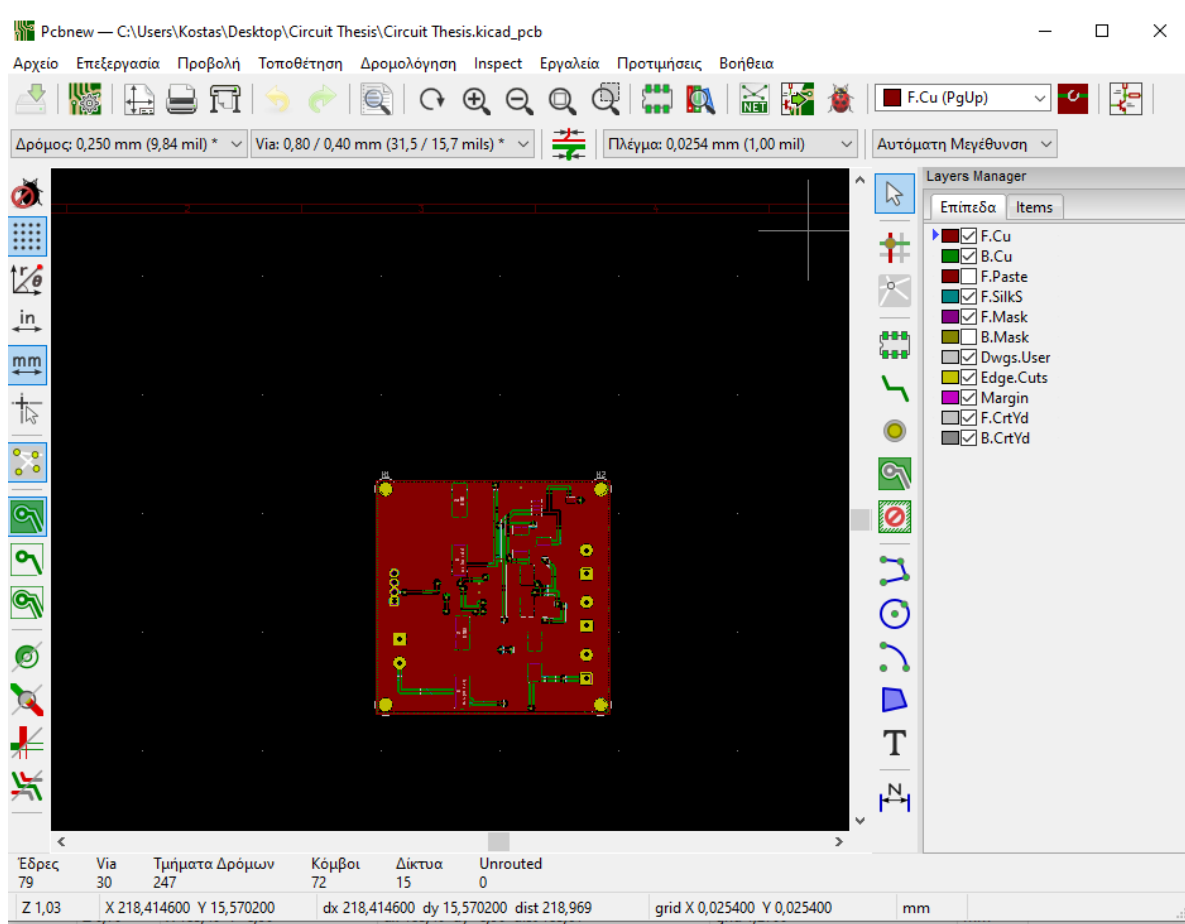
Τα σχηματικά κυκλώματα σχεδιάζονται στο KiCad με την εφαρμογή Eeschema. Αποτελούνται από εξαρτήματα (για παράδειγμα ένας πυκνωτής, αντίσταση) συνδέσεις μεταξύ τους, και άλλα στοιχεία. Τα σχηματικά κυκλώματα είναι οργανωμένα σε σχηματικά φύλλα. Κάθε σχηματικό φύλλο αποτελεί και ένα αρχείο .sch στον υπολογιστή του χρήστη. Επίσης, η λίστα δικτύων κάθε κυκλώματος αποτελεί ένα αρχείο .net στον υπολογιστή του χρήστη. Τα εξαρτήματα των σχηματικών μπορεί να ανήκουν σε μία βιβλιοθήκη του KiCad ή μπορεί να τα έχει σχεδιάσει από το μηδέν ο χρήστης χρησιμοποιώντας τη σχετική εφαρμογή του KiCad που ονομάζεται Επεξεργαστής Βιβλιοθήκης Εξαρτημάτων. Κάθε βιβλιοθήκη με εξαρτήματα αποτελεί και ένα αρχείο .lib στον υπολογιστή του χρήστη. Στην **Εικόνα 12** που ακολουθεί φαίνεται το περιβάλλον εργασίας του χρήστη.



Εικόνα 12: Εφαρμογή EEschema

### 2.3.2.2 Επεξεργαστής διάταξης τυπωμένων κυκλωμάτων (Pcb Layout Editor)

Για την σχεδίαση των τυπωμένων κυκλωμάτων στο KiCad υπάρχει η εφαρμογή Pcb Layout Editor. Στην εφαρμογή αυτή καθορίζεται η μορφή και οι διαστάσεις κάθε πλακέτας καθώς και οι θέσεις των αποτυπωμάτων πάνω σε αυτή. Έπειτα πραγματοποιούνται οι συνδέσεις μεταξύ τους μέσω δρόμων, από τρύπες, via, κ.α. Κάθε πλακέτα αποτελεί και ένα αρχείο .kicad\_pcb στον υπολογιστή του χρήστη. Τα τυπωμένα κυκλώματα όταν έχουν ολοκληρωθεί απεικονίζονται και σε άλλα αρχεία που μπορεί να παράξει ο χρήστης με το KiCad. Αυτά τα αρχεία είναι υπεύθυνα κυρίως για την κατασκευή της πλακέτας. Τέτοια αρχεία είναι τα gerber files (.gtl, .gbl, .gtr, .gto, .gbs, .gts, .gbr, .gml, κα), τα αρχεία διάτρησης (.drl), και τα αρχεία τοποθέτησης (.pos). Μία βιβλιοθήκη αποτυπωμάτων είναι ένας φάκελος αρχείων στον υπολογιστή του χρήστη, με επέκταση .pretty, ο οποίος φάκελος περιέχει αρχεία .kicad\_mod (ένα αρχείο .kicad\_mod για κάθε αποτύπωμα). Παρακάτω (Εικόνα 13) απεικονίζεται η εφαρμογή Pcb Layout Editor.

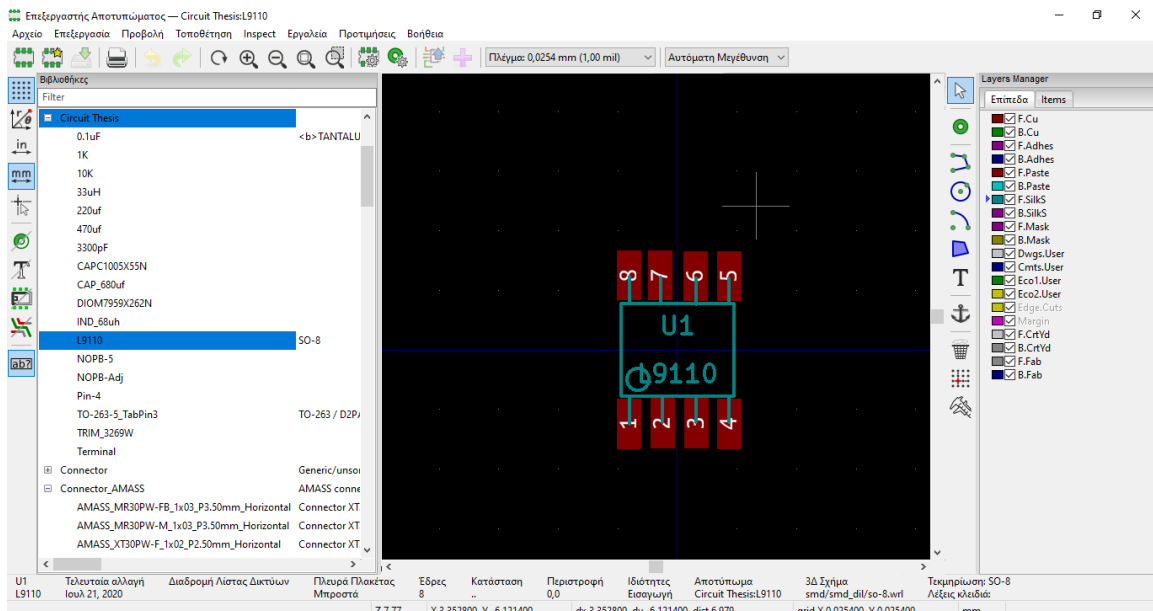


Εικόνα 13: Εφαρμογή Pcb Layout Editor

### 2.3.2.3 Αποτυπώματα (Footprints)

Μία βιβλιοθήκη αποτυπωμάτων είναι ένας φάκελος αρχείων στον υπολογιστή του χρήστη, με επέκταση .pretty, ο οποίος φάκελος περιέχει αρχεία .kicad\_mod (ένα αρχείο .kicad\_mod για κάθε αποτύπωμα). Τα αποτυπώματα μπορεί να έχουν τρισδιάστατη απεικόνιση ώστε ο χρήστης να μπορεί να απεικονίσει τρισδιάστατα το κύκλωμά του και να γνωρίζει πως θα είναι, πριν το κατασκευάσει.


Το KiCad προσφέρει τρισδιάστατες απεικονίσεις (3D σχήματα) για κάποια τυπικά εξαρτήματα, αλλά ο χρήστης έχει την δυνατότητα να σχεδιάσει και τα δικά του 3D σχήματα από το μηδέν, με κάποιο πρόγραμμα τρισδιάστατης σχεδίασης. Κάθε τρισδιάστατη απεικόνιση αποτελεί και ένα αρχείο .wrl στον υπολογιστή του χρήστη. Κάθε εξάρτημα αντιστοιχεί και σε ένα αποτύπωμα (συνήθως πολλά εξαρτήματα έχουν το ίδιο αποτύπωμα). Αυτή η αντιστοίχιση αποτυπώματος-εξαρτήματος γίνεται από το εργαλείο Cnrcb. Το σύνολο του κυκλώματος (σηματικό, πλακέτα) οργανώνεται από το KiCad σε έργα. Κάθε έργο αποτελεί και ένα αρχείο .pro στον υπολογιστή του χρήστη. Τέλος, βοηθητικά, ο χρήστης συνιστάται να διατηρεί μαζί με τα αρχεία του κυκλώματος και όλα τα εγχειρίδια (ή τα φύλλα δεδομένων) των εξαρτημάτων που χρησιμοποιεί ώστε να μπορεί να αναφέρεται σε αυτά ως προς τις συνδέσεις που απαιτούν, τις έδρες που χρησιμοποιούν κα. Συνήθως κάθε εγχειρίδιο είναι συνήθως ένα αρχείο .pdf.



Εικόνα 14: Επεξεργαστής αποτυπωμάτων.

## 2.4 Μικροελεγκτής ESP32-Cam


Το ESP32-CAM [14] είναι ένας πλήρης μικροελεγκτής που διαθέτει επίσης ενσωματωμένη κάμερα Wi-Fi, Bluetooth και υποδοχή κάρτας microSD. Δημιουργήθηκε και αναπτύχθηκε από την Espressif Systems. Είναι κατάλληλος για οικιακές έξυπνες συσκευές και εφαρμογές IoT. Κύριο χαρακτηριστικό του είναι ότι επιτυγχάνει εξαιρετικά χαμηλή κατανάλωση ενέργειας σε συνδυασμό με την δυναμική κλιμάκωση της ισχύος. Το ESP32-Cam μπορεί να λειτουργήσει ως πλήρες αυτόνομο σύστημα ή ως δευτερεύουσα συσκευή σε έναν κεντρικό υπολογιστή, μειώνοντας την εναέρια στοίβα επικοινωνίας στον κύριο επεξεργαστή εφαρμογών. Πιο συγκεκριμένα, αναλύονται τα χαρακτηριστικά του στον παρακάτω πίνακα (Πίνακα 1).

Συσκευή	Τεχνικά Χαρακτηριστικά
 <i>Εικόνα 15: Μικροελεγκτής ESP32-Cam</i>	<p>Μνήμη RAM: 520KB SRAM +4M PSRAM</p> <p>Εύρος τροφοδοσίας: 5V</p> <p>Θερμοκρασία λειτουργίας: -20 C° ~ 85 C°</p> <p>Ασύρματη Επικοινωνία: 802.11b/g/n</p> <p>Μέγεθος : 27*40.5*4.5 (±0.2) mm</p> <p>IO θύρες: 9</p>

Πίνακας 1: Τεχνικά Χαρακτηριστικά ESP32-Cam

## 2.5 Ρυθμιστής Υποβιβασμού Τάσης LM2596s (Step Down Voltage Regulator)

Το LM2596s [15] είναι ένα ολοκληρωμένο κύκλωμα που παρέχει όλες τις ενεργές λειτουργίες για τον υποβιβασμό της τάσης (buck). Έχει την δυνατότητα να μεταφέρει μέχρι και η φορτίο των 3A. Αυτή η συσκευή είναι διαθέσιμη σε τρεις εκδόσεις με σταθερή έξοδο τάσης 3,3V, 5V, 12V και μια έκδοση με ρυθμιζόμενη έξοδο. Απαιτώντας έναν ελάχιστο αριθμό εξωτερικών συστατικών, αυτοί οι ρυθμιστές είναι απλοί στη χρήση και περιλαμβάνουν εσωτερική αντιστάθμιση συχνότητας και ταλαντωτή σταθερής συχνότητας. Η σειρά LM2596 λειτουργεί σε συχνότητα μεταγωγής 150 kHz, επιτρέποντας έτσι μικρότερο μέγεθος φίλτρου. Στον **Πίνακα 2** παρουσιάζονται αναλυτικά τα χαρακτηριστικά της συγκεκριμένης συσκευής.

Συσκευή	Τεχνικά Χαρακτηριστικά
 <p><i>Εικόνα 16: Ρυθμιστής Τάσης LM2596s</i></p>	<p>Τάση Εισόδου: 3.2-40Volt</p> <p>Τάση Εξόδου: 1.3-35Volt</p> <p>Ένταση Εξόδου: 3Amp</p> <p>Θερμοκρασία λειτουργίας -45 °C ~ +85 °C</p> <p>Αποδοτικότητα μετατροπής: 92%</p>


*Πίνακας 2: Τεχνικά Χαρακτηριστικά LM2596s*

## 2.6 Οδηγός Κινητήρων L9110s (Motor Driver)

Η μονάδα οδηγού κινητήρα L9110s δύο καναλιών είναι μια συμπαγής πλακέτα που μπορεί να χρησιμοποιηθεί για την οδήγηση κινητήρων. Αυτή η μονάδα, διαθέτει δύο ανεξάρτητα τσιπ οδηγού κινητήρα L9110s όπου το καθένα να ανεβάσει συνεχώς ρεύμα 800mA. Η πλακέτα μπορεί να λειτουργήσει από 2.5V έως 12V επιτρέποντας τη χρήση αυτής της μονάδας με μικροελεγκτές 3.3V και 5V. Ένα σετ θηλυκών καρφίτσων κεφαλίδας χρησιμοποιείται για τη σύνδεση αυτής της μονάδας σε έναν μικροελεγκτή. Οι κινητήρες συνδέονται μέσω δύο σετ από κλέμες. Ένα σήμα διαμόρφωσης πλάτους παλμού (PWM) χρησιμοποιείται για τον έλεγχο της ταχύτητας ενός κινητήρα και μια ψηφιακή έξοδος



χρησιμοποιείται για να αλλάξει την κατεύθυνση του. Αυτή η μονάδα μπορεί επίσης να χρησιμοποιηθεί για την οδήγηση ενός μονοφασικού ή διφασικού κινητήρα stepper. Πιο αναλυτικά, παρουσιάζονται στον παρακάτω πίνακα (Πίνακα 3) τα χαρακτηριστικά της προαναφερθείσας πλακέτας.

Συσκευή	Τεχνικά Χαρακτηριστικά
 <p data-bbox="343 701 718 728">Εικόνα 17: Οδηγός Κινητήρων L9110s</p>	<p data-bbox="850 477 1244 504">Διαστάσεις: 28.4x21.6x14mm</p> <p data-bbox="850 548 1204 575">Τάση εισόδου: 2.5-12V DC</p> <p data-bbox="850 620 1428 647">Μέγιστο συνεχές ρεύμα ανά κανάλι: 800 mA</p> <p data-bbox="850 692 1125 719">Κανάλια κινητήρα: 2</p>

Πίνακας 3: Τεχνικά Χαρακτηριστικά L9110s

## 2.7 Σύνοψη Κεφαλαίου

Σε αυτό το κεφάλαιο, παρουσιάστηκαν οι τεχνολογίες που συνέβαλαν στο υλικό και λογισμικό μέρος της υλοποίησης της παρούσας διπλωματικής εργασίας. Επιπλέον, αναφέρθηκαν τα εργαλεία που απαιτούνταν η χρήση τους για αυτήν την υλοποίηση. Στο επακόλουθο κεφάλαιο, θα αναλυθούν περαιτέρω τα υλικά μέρη του συστήματος, η μεταξύ τους συνδεσμολογία και ο τρόπος λειτουργίας του υλικού μέρους.



## Κεφάλαιο 3<sup>ο</sup> – Δομικά μέρη συστήματος

Σε αυτό το κεφάλαιο, παρουσιάζονται οι απαιτήσεις του συστήματος, καθώς και τα δομικά του μέρη. Αρχικά, θα αναλυθούν οι απαιτήσεις και οι προδιαγραφές του συστήματος, καθώς και οι περιπτώσεις χρήσης του (use cases). Στην συνέχεια, ακολουθεί η ανάλυση του υλικού μέρους του οχήματος. Τέλος, θα γίνει αναλυτική αναφορά στον τρόπο σχεδιασμού και υλοποίησης της πλακέτας τυπωμένου κυκλώματος του οχήματος.

### 3.1 Απαιτήσεις Συστήματος (System Requirements)

Ένα από το πιο σημαντικά κομμάτια για την επιτυχή δημιουργία ενός συστήματος, είναι ο καθορισμός των απαιτήσεων του. Με τον όρο αυτό, αναφερόμαστε ουσιαστικά στις προδιαγραφές που πρέπει να τηρούνται, ώστε το σύστημα να είναι διαχειρίσιμο και λειτουργικό. Οι προδιαγραφές και οι ανάγκες του λογισμικού αναλύονται στη συνέχεια.

Το σύστημα πρέπει να είναι συνδεδεμένο στο διαδίκτυο. Βασική προϋπόθεση λειτουργίας του συστήματος μας είναι η σύνδεση του μέσω Wi-Fi στο διαδίκτυο. Όλη η επικοινωνία του οχήματος με τον υπολογιστή γίνεται μέσω Wi-Fi, συνεπώς ο χώρος στον οποίο βρίσκεται η πίστα του οχήματος πρέπει να διαθέτει καλή ποιότητα σήματος, έτσι ώστε να αποφευχθούν παρεμβολές από άλλες ασύρματες συσκευές.

Το σύστημα υλοποιήθηκε με κλιμακούμενο τρόπο, έτσι ώστε να μπορούν να συνδεθούν πολλά οχήματα σε μελλοντικές του επεκτάσεις.

Το λογισμικό είναι σχεδιασμένο και υλοποιημένο, έτσι ώστε να αποφευχθούν περιπτώσεις κατάρρευσης μετά από πολύωρη χρήση και ταυτόχρονα να υπάρχει όσο το δυνατόν καλύτερος χρόνος απόκρισης. Το σύστημα μας απαιτεί μια συνεχή επικοινωνία με τον υπολογιστή μιας και μεταφέρει δεδομένα για την κίνηση του οχήματος, αλλά και την μετάδοση video.

Το όχημα είναι υλοποιημένο με τέτοιο τρόπο, έτσι ώστε να αποτελεί μια σταθερή κατασκευή. Στο εσωτερικό του οχήματος, οι συνδέσεις πρέπει να είναι σταθερές, ώστε αν υπάρχουν συγκρούσεις να μην επηρεάσουν την σωστή λειτουργία του συστήματος.

Το όχημα και η εφαρμογή πρέπει να λειτουργούν με πολύ σταθερό τρόπο. Κατά τη διάρκεια του χρόνου που συνδέεται με τον υπολογιστή για τον έλεγχο, δεν πρέπει να καθυστερεί ή να χάνει το σήμα λόγω της υπερθέρμανσης τσιπ ελεγκτή.

Το περιβάλλον χρήσης της εφαρμογής είναι δομημένο και υλοποιημένο με τέτοιο τρόπο, ώστε να γίνεται κατανοητό και εύκολα διαχειρίσιμο, ακόμα και από άτομα που έχουν ελάχιστες γνώσεις και εμπειρία στην χρήση υπολογιστή. Στόχος της απλότητας του είναι όχι μόνο η ευκολία στην χρήση ενός μικροελεγκτή, αλλά και η καθοδήγηση του χρήστη στην εύρεση των λειτουργιών που αναζητεί. Τέλος, η αποκριτική του σχεδίαση επιτρέπει την διατήρηση της ποιότητας, της εμφάνισης και των λειτουργιών της εφαρμογής, από οποιαδήποτε συσκευή χρησιμοποιείται, οποιωνδήποτε διαστάσεων.

### 3.2 Περιπτώσεις Χρήσης (Use Cases)

Στην συγκριμένη παράγραφο παρουσιάζονται οι περιπτώσεις χρήσης του συστήματος, που σχεδιάστηκε και υλοποιήθηκε στην παρούσα διπλωματική εργασία, δείχνει όλες τις λειτουργίες τις οποίες μπορεί να υλοποιήσει ο χρήστης.

Παρακάτω αναλύονται τα βασικότερα σενάρια χρήσης του συστήματος.

**Είσοδος στο σύστημα.** Αρχικά, για να αποκτήσει κάποιος πρόσβαση στο σύστημα πρέπει να συνδεθεί στο τοπικό δίκτυο που βρίσκεται ο μικροελεγκτής. Αυτό προϋποθέτει την χρήση του κατάλληλου κωδικού. Η σύνδεση πραγματοποιείται μέσω της Android εφαρμογής που υλοποιήθηκε.

**Προβολή περιοχών.** Ο χρήστης, μετά την σύνδεση του έχει την δυνατότητα πλέον να περιηγηθεί στην σελίδα, καθώς και να εγκλιματιστεί με τα πλήκτρα χειρισμού με τα οποία μπορεί να κατευθύνει το όχημα.

**Επεξεργασία και ρύθμιση Κάμερας.** Στην συνέχεια, ο χρήστης αφού ενεργοποιήσει την κάμερα πατώντας το κουμπί Start Stream, μπορεί να επεξεργαστεί τα χαρακτηριστικά που θέλει να έχει όπως η ανάλυση, η ποιότητα, η φωτεινότητα, ο κορεσμός, καθώς και να ενεργοποιήσει το led flash, την λειτουργία εξισορρόπησης του λευκού χρώματος (AWB) και την λειτουργία ενίσχυσης εικόνας (AGC) που συμβάλει στην καθαρότερη εικόνα .

**Χειρισμός Οχήματος.** Αφού γίνουν οι απαραίτητες ρυθμίσεις από την κάμερα, ο χρήστης είναι σε θέση να χειριστεί το όχημα από το πληκτρολόγιο με την χρήση συγκεκριμένων πλήκτρων

αλλά και με το ποντίκι. Πιο αναλυτικά, οι κινήσεις που μπορεί να κάνει κάθε όχημα είναι αυτές που αναφέρονται στην συνέχεια και πραγματοποιούνται με τα εξής πλήκτρα:

Μπροστά (W), Πίσω (S), Δεξιά (D), Αριστερά (A), Μπροστά-Αριστερά (Q), Μπροστά-Δεξιά (E), Πίσω-Δεξιά (C), Πίσω-Αριστερά (Z)

### 3.3 Υλικό μέρος

Στην παρούσα ενότητα αναλύεται το σύστημα που έχει υλοποιηθεί στο όχημα. Αρχικά, ο βασικός πυλώνας επικοινωνίας ολόκληρου του συστήματος είναι ο μικροελεγκτής ESP-32 Cam. Ο μικροελεγκτής αυτός είναι υπεύθυνος και για την κίνηση του οχήματος, αφού ελέγχει και τον οδηγό των κινητήρων (Motor Driver). Επιπρόσθετα, χρησιμοποιούνται δύο ρυθμιστές υποβιβασμού τάσης προκειμένου να επιτευχθεί η σωστή λειτουργία των ηλεκτρονικών υλικών, αλλά και η καλύτερη απόκριση του συστήματος. Πιο συγκεκριμένα, το σύστημα του οχήματος περιλαμβάνει τα εξής υλικά στοιχεία.

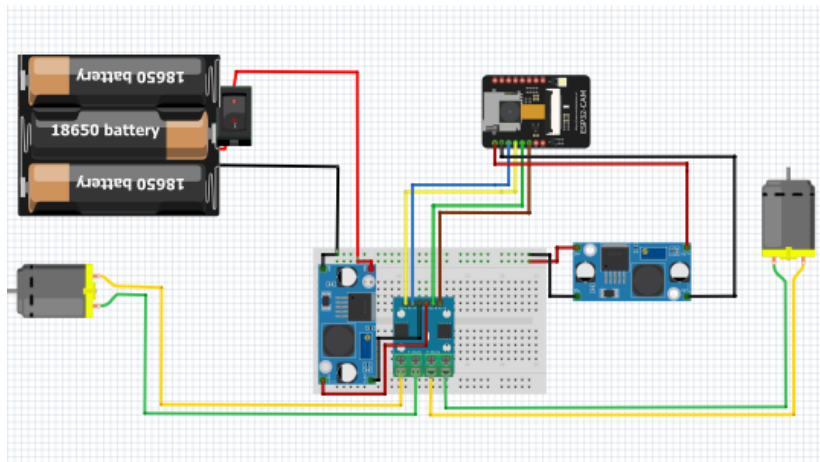
- 1x Μικροελεγκτής Esp-32 Cam
- 2x Step Down Voltage Regulator LM2596s
- 1x Motor Driver L9110S
- 1x Διακόπτης οχήματος
- 2x Ηλεκτροκινητήρας συνεχούς ρεύματος 9V
- 3x Μπαταριά 18650 3.6V 2900 mAh
- 1x Μπαταριοθήκη 18650 τριών θέσεων

#### 3.3.1 Ενεργειακές Απαιτήσεις (Energy Requirements)

Ένα από τα πιο σημαντικά κομμάτια για τη σωστή του λειτουργία του συστήματος, είναι η τροφοδοσία του με ηλεκτρικό ρεύμα. Οι ενεργειακές ανάγκες του συστήματος που υλοποιήθηκε, αναφέρονται αναλυτικά στην συγκεκριμένη ενότητα. Επομένως, για την κίνηση του οχήματος και την τροφοδοσία όλου του συστήματος χρησιμοποιούνται τρεις μπαταρίες Samsung 18650 με χωρητικότητα 2900 mAh. Η επιλογή των συγκεκριμένων μπαταριών, έγινε με γνώμονα την μεγαλύτερη αυτονομία και καλύτερη απόκριση του συστήματος, αλλά και την εξοικονόμηση χώρου. Αναλυτικότερα, οι δυο κινητήρες των 9V του οχήματος εκτιμάται ότι καταναλώνουν 500-600mAh στις μέγιστες στροφές συνυπολογίζοντας το βάρος του οχήματος. Επιπλέον, ο μικροελεγκτής καταναλώνει μέχρι και 310mAh σε λειτουργία με την χρήση του

led flash, αλλά η μέση κατανάλωση του εκτιμάται στα 180mAh. Συνεπώς, η αυτονομία του κάθε οχήματος μπορεί να αγγίξει και τις δυο ώρες συνεχούς χρήσης. Επίσης, προκειμένου να υποβιβαστεί η τάση στις επιθυμητές τιμές για την σωστή λειτουργία του μικροελεγκτή και των κινητήρων, χρησιμοποιήθηκαν δυο ρυθμιστές υποβιβασμού τάσης (Step Down Voltage Regulator), όπου ο ένας θα ρυθμίζει την τάση στα 5V για τον μικροελεγκτή, ενώ ο δεύτερος θα υποβιβάζει την τάση στα 9V για τους κινητήρες.

Στην **Εικόνα 18**, φαίνεται το κυκλωματικό ισοδύναμο του συστήματος πριν από τον σχεδιασμό της πλακέτας.

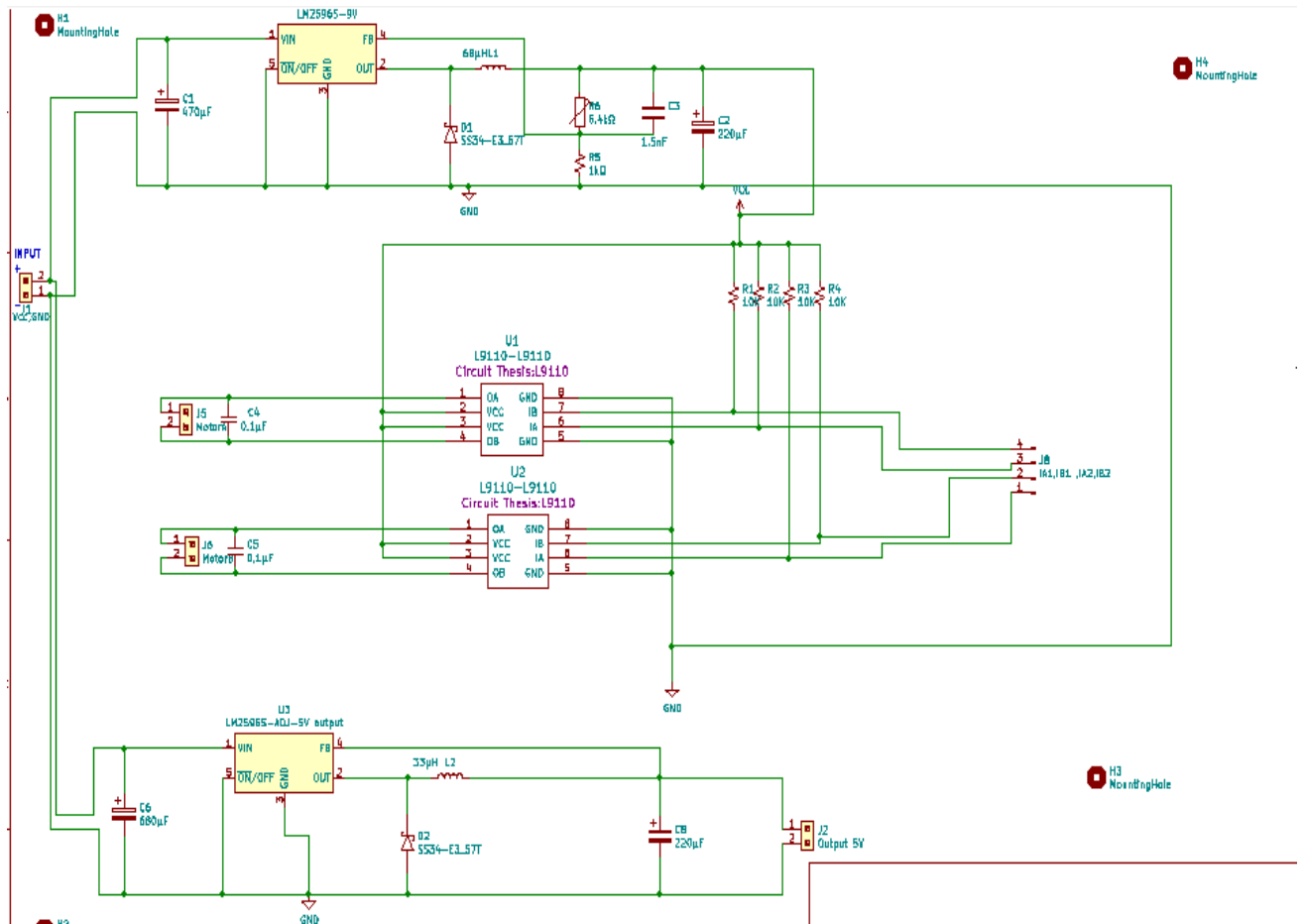


*Εικόνα 18: Κυκλωματική απεικόνιση του συστήματος*

Όπως παρουσιάζεται και στην **Εικόνα 18**, από τις μπαταρίες το ρεύμα κατευθύνεται προς τους μετατροπείς συνεχούς τάσης. Ο ένας τροφοδοτεί τον μικροελεγκτή με 5V και ο άλλος με 9V τους κινητήρες μέσω του οδηγού των κινητήρων L9110s.

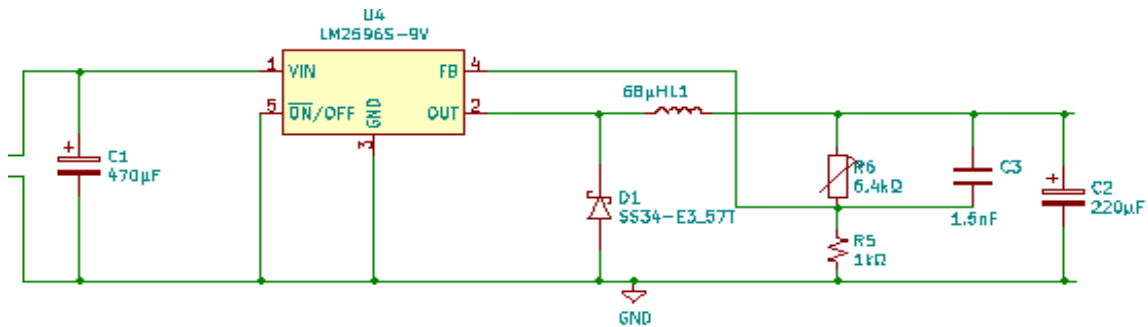
### 3.4 Υλοποίηση Τυπωμένου Κυκλώματος PCB

Έπειτα από την δοκιμή της ορθής λειτουργίας του συστήματος (Εικόνα 18), παρατηρήθηκε ότι υπήρχε μεγάλος όγκος καλωδίων και σε συνδυασμό με τον περιορισμένο χώρο που υπάρχει στο κάθε όχημα, επιλέχθηκε να σχεδιαστεί και να υλοποιηθεί μια PCB πλακέτα διπλής όψεως με τις επιθυμητές διαστάσεις (50mm x 50mm) για να καλύψει τις ανάγκες του κάθε οχήματος. Η λειτουργία της πλακέτας ουσιαστικά αφορά τον χειρισμό των κινητήρων αλλά και την παροχή δυο εξόδων τάσης, μια 5V για τον μικροελεγκτή, και στα 9V για τους κινητήρες. Στην Εικόνα 19 παρακάτω, φαίνεται πιο αναλυτικά το σχηματικό του κυκλώματος της πλακέτας.



Εικόνα 19: Σχηματικό πλακέτας

Όπως φαίνεται στην **Εικόνα 19**, υπάρχει μια είσοδος (Input) όπου τροφοδοτείται το κύκλωμα και στην συνέχεια μοιράζεται στα δυο μικρότερα κυκλώματα των ρυθμιστών τάσης. Στην παρακάτω εικόνα (**Εικόνα 20**) παρουσιάζεται το κύκλωμα του ρυθμιζόμενου ρυθμιστή.

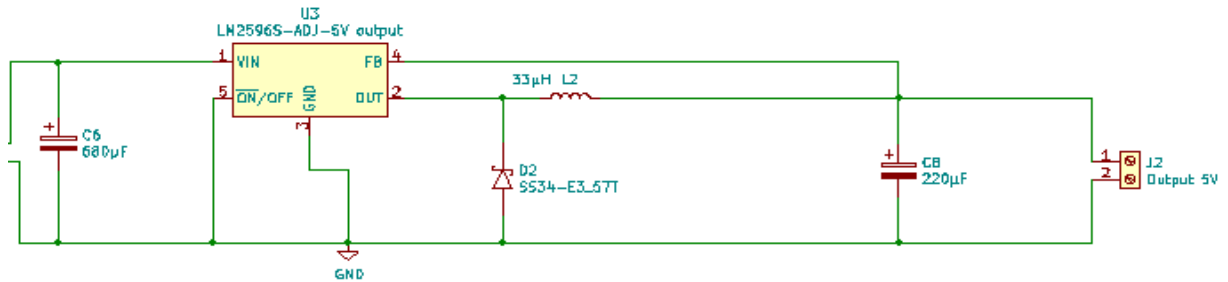


Εικόνα 20: Κύκλωμα Ρυθμιστή Τάσης LM2596s Adj

Ο πρώτος ρυθμιστής τάσης LM2596s-Adj (**Εικόνα 20**), είναι αυτός που τροφοδοτεί τους κινητήρες του κάθε οχήματος. Αναλυτικότερα, αποτελείται από δυο ηλεκτρολυτικούς πυκνωτές έναν στην είσοδο ( $C_1=470\mu\text{F}$ ) και έναν στην έξοδο ( $C_2=220\mu\text{F}$ ) όπου διατηρούν σταθερή την τάση εισόδου και εξόδου αντίστοιχα. Στην συνέχεια, υπάρχει μια δίοδος Schottky που διασφαλίζει την ροή του ρεύματος προς μία κατεύθυνση, και ένα πηνίο ( $L_1=68\mu\text{H}$ ) οπού λειτουργεί ως ηλεκτρονικό φίλτρο και διατηρεί την συχνότητα στα επιθυμητά επίπεδα (150kHz). Έπειτα, σημαντικό ρόλο παίζουν οι δυο αντιστάσεις, εκ των οποίων η μία λειτουργεί σαν ποτενσιόμετρο προκειμένου να ρυθμίζει την τάση εξόδου στις επιθυμητές τιμές. Τέλος, ένας μικρός πυκνωτής ( $C_3=1,5\text{nF}$ ) έχει τοποθετηθεί με σκοπό να διατηρεί σταθερή την ανατροφοδότηση τάσης.



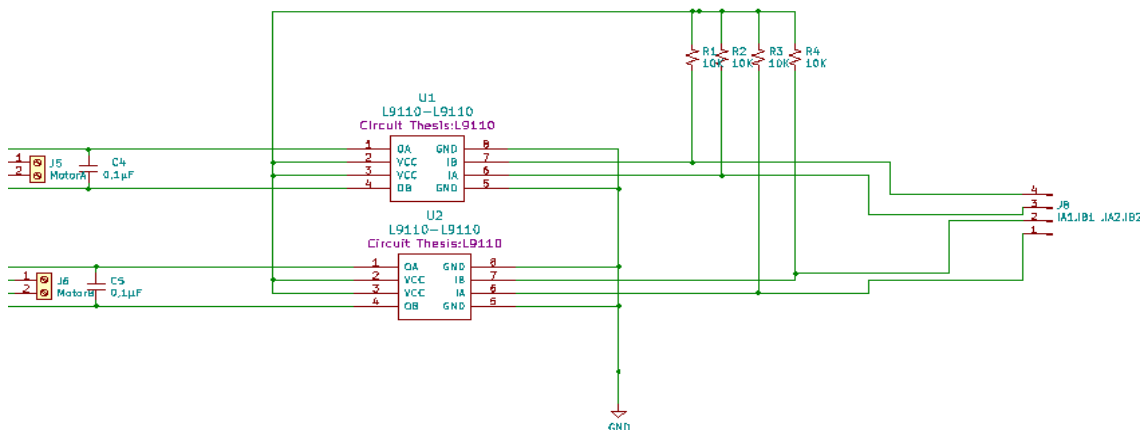
Στην συνέχεια, παρουσιάζεται ο δεύτερος ρυθμιστής τάσης, που χρησιμοποιείται για την τροφοδοσία του μικροελεγκτή και έχει σταθερή έξοδο 5V.



Εικόνα 21: Κύκλωμα Ρυθμιστή Τάσης LM2596s 5V

Ο δεύτερος ρυθμιστής τάσης LM2596s-5V (Εικόνα 21), αποτελείται και αυτός από δυο ηλεκτρολυτικούς πυκνωτές έναν στην είσοδο ( $C_6=680\mu\text{F}$ ) και έναν στην έξοδο ( $C_8=220\mu\text{F}$ ) όπου διατηρούν σταθερή την τάση εισόδου και εξόδου αντίστοιχα. Στην συνέχεια, υπάρχει μια διάδος Schottky που διασφαλίζει την ροή του ρεύματος προς μία κατεύθυνση, και ένα πηνίο ( $L_2=33\mu\text{H}$ ) οπου λειτουργεί ως ηλεκτρονικό φίλτρο και διατηρεί την συχνότητα στα επιθυμητά επίπεδα (150kHz).

Έπειτα, το τελευταίο κομμάτι είναι ο οδηγός κινητήρων L9110s όπου έχει την ιδιότητα να ρυθμίζει την φορά στρέψης των κινητήρων αλλάζοντας την πολικότητα τους. Πιο συγκεκριμένα στην εικόνα (**Εικόνα 22**) που ακολουθεί φαίνεται το σχηματικό του κυκλώματος αυτού.

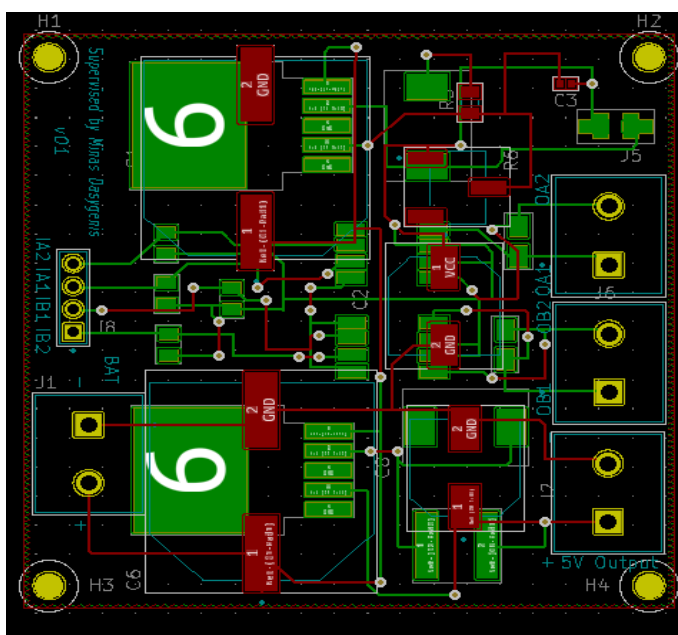


Εικόνα 22: Κύκλωμα Οδηγού Κινητήρων L9110s

Όπως παρατηρείται στο παραπάνω κύκλωμα (**Εικόνα 22**), όλες οι εισοδοι συνδέονται με την τροφοδοσία μέσω τεσσάρων pull-up αντιστάσεων 10kΩ, έτσι ώστε όταν ο κεντρικός διακόπτης του συστήματος είναι ανοικτός να βρίσκεται σε λογικό HIGH ενώ όταν κλείσει σε λογικό LOW. Ως εκ τούτου, οι έξοδοι του κινητήρα είναι σε κατάσταση λογικού LOW. Στην συνέχεια ακολουθούν τα δυο τσιπ L9110s ακολουθούμενα από δυο πυκνωτές (C<sub>4</sub>, C<sub>5</sub>=0,2µF) όπου διατηρούν σταθερή την τάση στην έξοδο του κυκλώματος.

### 3.4.1 Τελική μορφή πλακέτας PCB

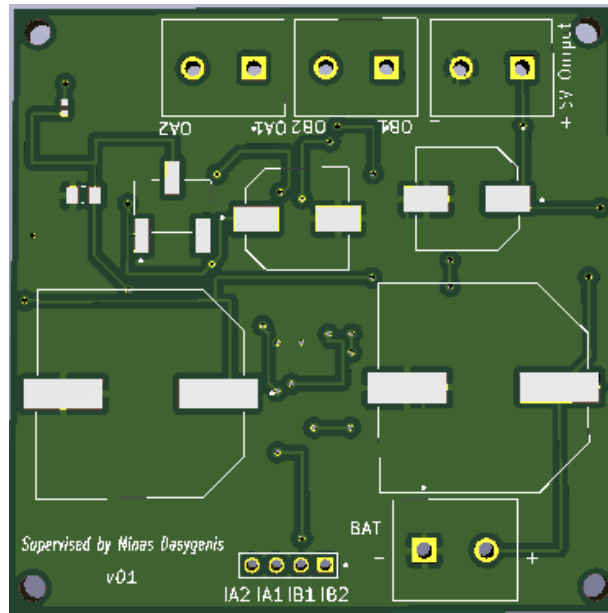
Μετά την υλοποίηση των επιμέρους κυκλωμάτων ακολούθησε ο σχεδιασμός του μοντέλου PCB στην πλατφόρμα του KiCad. Αρχικά, μελετήθηκε το μέγεθος την πλακέτας και στην συνέχεια τοποθετήθηκαν οι δρόμοι χαλκού που συνδέουν τα εξαρτήματα μεταξύ τους. Μάλιστα, λόγω του μικρού μεγέθους (50 mm x 50 mm) αξιοποιήθηκαν και οι δυο όψεις της πλακέτας για την τοποθέτηση εξαρτημάτων.



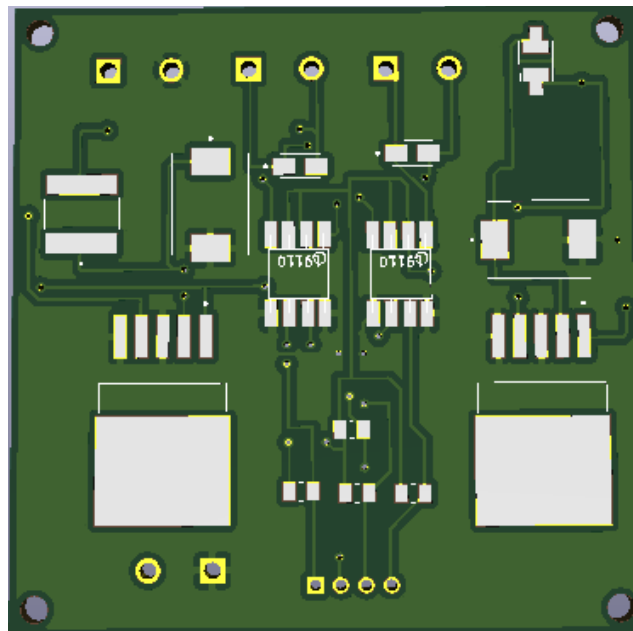
Εικόνα 23: Απεικόνιση PCB Layout

Στην εικόνα παραπάνω (**Εικόνα 23**), διακρίνονται με κόκκινο χρώμα οι δρόμοι χαλκού μεταξύ των εξαρτημάτων της πάνω όψης και με πράσινο χρώμα της κάτω όψης. Επίσης, σε κάθε γωνία της πλακέτας τοποθετήθηκαν μικρές τρύπες διαμέτρου 2,1mm για πιο σταθερή προσαρμογή πάνω στο όχημα.

Στην συνέχεια, ακολουθεί μια προεπισκόπηση της (**Εικόνα 24**, **Εικόνα 25**) πλακέτας πριν από την εκτύπωση της. Στην κατάσταση αυτή φαίνεται ξεκάθαρα η τελική της μορφή, οι δρόμοι και τα νία που συνδέουν τις δυο όψεις καθώς και οι επιγραφές που έχουν οριστεί για να διευκολύνουν το χρήστη.



Εικόνα 24: 3D Απεικόνιση Πάνω Όψης



Εικόνα 25: 3D Απεικόνιση Κάτω Όψης

### 3.5 Κόστος κατασκευής (Bill Of Materials)

Βασική προϋπόθεση στην υλοποίηση του συστήματος είναι ο συνδυασμός της αξιοπιστίας με το χαμηλότερο κόστος κατασκευής. Για αυτόν τον λόγο, είναι επιτακτική η ανάγκη για την αξιοποίηση ήδη υπαρχόντων πόρων, καθώς και η ερευνά αγοράς οικονομικών υλικών. Πιο αναλυτικά, παρουσιάζεται το κόστος κατασκευής στον παρακάτω πίνακα (Πίνακας 4).

Υλικό	Τιμή Μονάδας €	Ποσό €
2x Τηλεκατευθυνόμενο Αυτοκίνητο	12	24
1x ESP32-Cam	8	8
3x Μπαταρίες 18650 2900mAh	4,2	12,6
1x Μπαταριοθήκη τριών θέσεων	1,5	1,5
2x PCB πλακέτες 50 mm x 50 mm	3,2	6,4
4x Schottky Δίοδος SS34	0,41	1,64
2x Ηλεκτρολυτικός πυκνωτής AEC-Q200 680uF	1,96	3,92
2x Αντίσταση SMD Thick Film 1kΩ	0,41	0,82
8x Αντίσταση SMD Thick Film 10kΩ	0,135	1,08
2x Κεραμικός πυκνωτής X7R 0805	0,315	0,63
8x Κλέμες	0,495	3,96
2x Ηλεκτρολυτικός πυκνωτής AEC-Q200 470uF	1,58	3,16
4x Πυκνωτής Τανταλίου SMD 0,1uF	0,486	1,944
2x Πηνίο SMD 33uH	0,441	0,882
2x Σετ Pin 4 θέσεων	0,774	1,548
2x Ποτενσιόμετρο SMD 20kΩ	4,56	9,12
2x Πηνίο SMD 68uH	0,324	0,648
2x Τσιπ LM2596s -5V	4,30	8,60
2x Τσιπ LM2596s -Adj	5,16	10,32
4x Ηλεκτρολυτικός πυκνωτής AEC-Q200 220uF	0,576	2,304
4x Τσιπ L9110s	0,51	2,04
<b>Σύνολο</b>		<b>105,116 €</b>

Πίνακας 4: Κόστος Υλικών Κατασκευής

### 3.6 Σύνοψη Κεφαλαίου

Στο προκείμενο κεφάλαιο, αναλύθηκε λεπτομερώς το υλικό μέρος του οχήματος, οι λειτουργικές απαιτήσεις όπως και οι περιπτώσεις χρήσης του συστήματος. Επιπλέον, εξηγήθηκε ο τρόπος λειτουργίας και ο σκοπός του κάθε υλικού. Παράλληλα, έγινε η αναφορά και η ανάλυση του τρόπου κατασκευής της πλακέτας τυπωμένου κυκλώματος (PCB). Τέλος, παραδόθηκε μία λίστα με όλα τα υλικά που χρησιμοποιήθηκαν και το κόστος τους του καθενός. Ακολουθεί το τέταρτο κεφάλαιο, το οποίο αφορά το λογισμικό που υλοποιήθηκε, ποιες λειτουργίες περιλαμβάνει και τι απαιτήσεις καλύπτει. Επιπλέον, περιγράφεται η λειτουργία του και ο τρόπος επικοινωνίας μεταξύ των συσκευών. Τέλος, επισημαίνονται οι τρόποι χρήσης του συστήματος, οι περιορισμοί του και οι συμβάσεις που έγιναν.

## Κεφάλαιο 4 – Ανάλυση Λογισμικού

Αφού καλύφθηκαν οι απαραίτητες γνώσεις για το υλικό μέρος του συστήματος, στο παρόν κεφάλαιο επεξηγείται η ολοκλήρωση του λογισμικού. Σε τούτο το κεφάλαιο, αναλύονται τα λογισμικά που υλοποιήθηκαν σε όλες τις επιμέρους συσκευές, και η μεταξύ τους επικοινωνία. Η επιμέρους ανάλυση ξεκινάει από τα μικρότερα και απλούστερα υποσυστήματα, καταλήγοντας στα μεγαλύτερα και πολυπλοκότερα. Παράλληλα, αναγράφονται τρόποι λειτουργίας, τα ποιοτικά χαρακτηριστικά του συστήματος, όπως και συμβάσεις που έγιναν κατά την υλοποίηση της παρούσας διπλωματικής εργασίας.

### 4.1 Λογισμικό Μικροελεγκτή

Όπως αναφέρθηκε στην προηγούμενη ενότητα, κύριος πυλώνας επικοινωνίας των εφαρμογών με το όχημα αποτελεί ο μικροελεγκτής ESP32-Cam. Βασική προϋπόθεση λειτουργίας του συστήματος που υλοποιήθηκε είναι σύνδεση του με το διαδίκτυο. Για τον λόγο αυτό, χρησιμοποιήθηκε η βιβλιοθήκη WiFi.h που επιτρέπει στον μικροελεγκτή την σύνδεση στο διαδίκτυο [18]. Η βιβλιοθήκη αυτή, υποστηρίζει κρυπτογράφηση WEP και WPA2 και είναι υπεύθυνη για την σύνδεση στο διαδίκτυο. Στην συνέχεια, για την χρήση κάμερας και την αξιοποίηση όλων των λειτουργιών της χρησιμοποιήθηκε η βιβλιοθήκη esp\_camera.h. Η βιβλιοθήκη αυτή περιλαμβάνει την εφαρμογή της κάμερας καθώς και ένα πλήθος λειτουργιών που διαθέτει. Έπειτα, για την λειτουργία της ιστοσελίδας (web page) καλείται η βιβλιοθήκη esp\_http\_server.h [29]. Στόχος της βιβλιοθήκης αυτής είναι να παρέχει τη δυνατότητα εκτέλεσης διακομιστή ιστού (Web Server) και να διασφαλίζει την επικοινωνία του με τους πελάτες. Τέλος, μια ακόμη σημαντική βιβλιοθήκη που χρησιμοποιείται για την επικοινωνία της desktop εφαρμογής με τον μικροελεγκτή είναι η WiFiUdp.h [19]. Ο ρόλος της βιβλιοθήκης αυτής είναι να αναδείξει την επικοινωνία UDP μεταξύ ενός διακομιστή και ενός εξωτερικού πελάτη. Αυτό επιτυγχάνεται, μέσα στην δομή loop με τον έλεγχο άφιξης πακέτου και έπειτα όταν λαμβάνεται ένα έγκυρο πακέτο, ένα πακέτο αναγνώρισης αποστέλλεται πίσω στον πελάτη στην ίδια θύρα που έχει σταλεί. Κάθε πακέτο περιέχει ένα αλφαριθμητικό που ουσιαστικά κωδικοποιεί μια εντολή για την κίνηση του οχήματος. Οι εντολές διαφέρουν ανάλογα με την εφαρμογή που έχει επιλέξει ο χρήστης να χρησιμοποιήσει. Με την κωδικοποίηση αυτή των εντολών αυξάνεται η ασφάλεια του συστήματος από κακόβουλους χρήστες που θα επιχειρήσουν να χειριστούν το όχημα.

Στον παρακάτω πίνακα (**Πίνακας 5**) παρουσιάζονται οι εντολές για τον χειρισμό του οχήματος από την desktop εφαρμογή που υλοποιήθηκε .

Εντολή (Αλφαριθμητικό)	Ενέργεια
f	Μπροστά
b	Πίσω
r	Δεξιά
l	Αριστερά
fl	Μπροστά και Αριστερά
fr	Μπροστά και Δεξιά
bl	Πίσω και Αριστερά
br	Πίσω και Δεξιά

*Πίνακας 5: Εντολές Κίνησης Οχήματος (Desktop App)*

Στην συνέχεια, ακολουθεί ο πίνακας (**Πίνακας 6**) όπου αναλύονται οι εντολές που εκτελούνται για την κίνηση του οχήματος από την εφαρμογή ιστού (Web App) που υλοποιήθηκε.

Εντολή (Αλφαριθμητικό)	Ενέργεια
forward	Μπροστά
backward	Πίσω
right	Δεξιά
left	Αριστερά
fleft	Μπροστά και Αριστερά
fright	Μπροστά και Δεξιά
bleft	Πίσω και Αριστερά
bright	Πίσω και Δεξιά

*Πίνακας 6: Εντολές Κίνησης Οχήματος (Web App)*



#### 4.1.2 Λειτουργίες Κάμερας

Όπως έχει αναφερθεί στα προηγούμενα κεφάλαια, η κάμερα που βρίσκεται ενσωματωμένη πάνω στον μικροελεγκτή διαθέτει αρκετές λειτουργίες που αξιοποιούνται για να καλύψουν τις ανάγκες της συγκεκριμένης διπλωματικής εργασίας. Πιο συγκεκριμένα, οι λειτουργίες αυτές όπως και οι εντολές για την κίνηση του οχήματος λειτουργούν με την χρήση αλφαριθμητικού, και έτσι επιτυγχάνεται μια κωδικοποίηση που εξαρτάται από δύο παραμέτρους για κάθε λειτουργία. Η πρώτη παράμετρος αφορά το τύπο της λειτουργίας (π.χ. ποιότητα, ανάλυση, αντίθεση) και ορίζεται ως var (variable) και η δεύτερη παράμετρος ορίζεται ως val (value) και αφορά την τιμή που παίρνει η πρώτη.

Πιο αναλυτικά παρουσιάζονται στον πίνακα (**Πίνακας 7**) παρακάτω οι λειτουργίες καθώς και οι τιμές των παραμέτρων τους.

Λειτουργία	Variable (var)	Value (val)	Ενέργεια
Ανάλυση (Resolution)	framesize	0→QQVGA(160x120) 3→HQVGA(240x176) 4→ QVGA(320x240) 5→ CIF(400x296) 6→ VGA(640x480) 7→ SVGA(800x600)	Αλλάζει η ανάλυση της οθόνης στις επιθυμητές διαστάσεις.
Ποιότητα (Quality)	quality	Min= 10 Max= 63	Ρύθμιση της ποιότητας της εικόνας.
Φωτεινότητα	brightness	Min= -2 Max= 2	Ρύθμιση φωτεινότητας της εικόνας.
Αντίθεση (Contrast)	contrast	Min= -2 Max= 2	Ρύθμιση αντίθεσης της εικόνας.
Κορεσμός (Saturation)	saturation	Min= -2 Max= 2	Ρύθμιση του κορεσμού του χρώματος της εικόνας.
Flash	flash	Ενεργό= flashon Ανενεργό= flashoff	Ενεργοποίηση/Απενεργοποίηση Led flash της συσκευής.
Εξισορρόπηση Λευκού (AWB, Auto White Balance)	awb_gain	0→ Auto 1→ Sunny 2→ Cloudy 3→ Office 4→ Home	Ρύθμιση της εξισορρόπησης λευκού χρώματος.
AGC (Auto Gain Celling)	gaincelling	Min= 0 Max=6	Ρύθμιση του κέρδους με σκοπό την αύξηση της φωτεινότητας σε περιοχές με λιγότερο φωτισμό.

Πίνακας 7: Λειτουργίες Κάμερας

### 4.1.3 Κώδικας Μικροελεγκτή

Στην παρούσα υποενότητα, παρουσιάζονται τα πιο σημαντικά κομμάτια κώδικα που έπαιξαν καθοριστικό ρόλο στην υλοποίηση του συστήματος.

Αρχικά, ένα σημαντικό κομμάτι είναι εκείνο όπου επιτυγχάνεται η επικοινωνία μέσω του πρωτοκόλλου UDP με την εφαρμογή. Πιο αναλυτικά, Οδημιουργήθηκε ένας πίνακας ακεραίων (buffer) 50 θέσεων όπου με την συνάρτηση memset [28] αρχικοποιούμε την τιμή 0 σε όλες τις θέσεις του. Έπειτα, η συνάρτηση parsePacket [25] ελέγχει την παρουσία ενός πακέτου UDP και αναφέρει το μέγεθος του, και στην συνέχεια, η συνάρτηση read [26] διαβάζει τα δεδομένα UDP από το καθορισμένο buffer. Τέλος, η συνάρτηση strcmp [27] συγκρίνει τα δεδομένα του buffer με το αλφαριθμητικό που αντιστοιχεί σε κάθε εντολή και έπειτα, εκτελεί την εντολή κίνησης με την συνάρτηση WheelAct (**Εικόνα 26**). Παρακάτω, φαίνεται πιο αναλυτικά η συγκεκριμένη υλοποίηση (**Εικόνα 27**).

```
void WheelAct(int nLf, int nLb, int nRf, int nRb)
{

    digitalWrite(gpLf, nLf);
    digitalWrite(gpLb, nLb);
    digitalWrite(gpRf, nRf);
    digitalWrite(gpRb, nRb);
}
```

Εικόνα 26: Συνάρτηση WheelAct

```

uint8_t buffer[50];
memset(buffer, 0, 50); //processing incoming packet, must be called before reading the buffer

udp.parsePacket();

if(udp.read(buffer, 50) > 0){
  Serial.print("Server to Car: ");
  Serial.println((char *)buffer);
  if(strcmp( (char *)buffer, "f" ) == 0 ){
    Serial.println("Forward");
    WheelAct(HIGH, LOW , LOW, LOW);
  }else if(strcmp( (char *)buffer, "fl") == 0){
    Serial.println("F-Left");
    WheelAct(HIGH,LOW , LOW, HIGH);
  }else if(strcmp( (char *)buffer, "fr") == 0){
    Serial.println("F-Right");
    WheelAct(HIGH, LOW, HIGH, LOW);
  }else if(strcmp( (char *)buffer, "l") == 0){
    Serial.println("Left");
    WheelAct(LOW, LOW, LOW, HIGH);
  }else if(strcmp( (char *)buffer, "r") == 0){
    Serial.println("Right");
    WheelAct(LOW, LOW, HIGH, LOW);
  }
}

```

*Εικόνα 27: Επικοινωνία Μικροελεγκτή με Windows Εφαρμογή (Server Side)*

Στην συνέχεια, ένα ακόμη σημαντικό κομμάτι κώδικα που αξίζει να αναφερθεί είναι εκείνο της διαχείρισης των εντολών κίνησης από την εφαρμογή ιστοχώρου. Πιο συγκεκριμένα, έχει υλοποιηθεί μια συνάρτηση με όνομα `esp_err_t car_handler` οπου διαβάζει τα `jquery` που στέλνει η εφαρμογή ιστού και πραγματοποιεί έλεγχο του αλφαριθμητικού για την εύρεση της εντολής.

Έπειτα, εκτελεί την συνάρτηση WheelAct (**Εικόνα 26**) όπου εκκινεί τους κινητήρες του οχήματος για την εκτέλεση της αντίστοιχης κίνησης. Στις εικόνες (**Εικόνα 28**, **Εικόνα 29**) που ακολουθούν, παρουσιάζεται η συνάρτηση που περιεγράφηκε παραπάνω.

```
static esp_err_t car_handler(httpd_req_t *req){

    char* buf;
    size_t buf_len;
    char variable[32] = {0,};
    char value[32] = {0,};

    buf_len = httpd_req_get_url_query_len(req) + 1;
    if (buf_len > 1) {
        buf = (char*)malloc(buf_len);
        if(!buf){
            httpd_resp_send_500(req);
            return ESP_FAIL;
        }
        if (httpd_req_get_url_query_str(req, buf, buf_len) == ESP_OK) {
            if (httpd_query_key_value(buf, "var", variable, sizeof(variable)) == ESP_OK &&
                httpd_query_key_value(buf, "val", value, sizeof(value)) == ESP_OK) {
            } else {
                free(buf);
                httpd_resp_send_404(req);
                return ESP_FAIL;
            }
        } else {
            free(buf);
            httpd_resp_send_404(req);
            return ESP_FAIL;
        }
    }
    free(buf);
} else {
    httpd_resp_send_404(req);
    return ESP_FAIL;
}
```

Εικόνα 28: Συνάρτηση Μικροελεγκτή (esp\_err\_t car\_handler)

```

Serial.printf(value);
Serial.printf(variable);
if(!strcmp(variable, "move"))
{
  last_move_millis = millis();
  Serial.printf("Inside");
  if(!strcmp(value, "forward"))
  {
    Serial.println("Forward");

    WheelAct(HIGH, LOW , LOW, LOW);

  }
  else if(!strcmp(value, "back"))
  {
    Serial.println("Backward");

    WheelAct(LOW,HIGH , LOW, LOW);

  }
  else if(!strcmp(value, "left"))
  {
    Serial.println("Left");

    WheelAct(LOW, LOW, LOW, HIGH);

  }
  else if(!strcmp(value, "fleft"))
  {
    Serial.println("F-Left");
  }
}

```

*Εικόνα 29: Συνέχεια Συνάρτησης esp\_err\_t car\_handler*

## 4.2 Android Εφαρμογή

Στην ενότητα αυτή, παρουσιάζεται η εφαρμογή Android που αναπτύχθηκε στην παρούσα διπλωματική εργασία, για να καλύψει την ανάγκη του οχήματος να μπορεί να συνδεθεί σε ασύρματα δίκτυα χωρίς να χρειάζεται ο προγραμματισμός του. Πιο αναλυτικά, η εφαρμογή που υλοποιήθηκε, επικοινωνεί μέσω Bluetooth με τον μικροελεγκτή του οχήματος και λαμβάνει τα διαθέσιμα ασύρματα δίκτυα. Στην συνέχεια, ο χρήστης επιλέγει το δίκτυο που επιθυμεί και πληκτρολογεί τον κωδικό του. Τέλος, επιστρέφει την κατάσταση της σύνδεσης και την διεύθυνση IP στην οποία συνδέθηκε το όχημα.

Πιο συγκεκριμένα στην εικόνα που ακολουθεί (**Εικόνα 30**) φαίνεται η αρχική οθόνη της εφαρμογής πριν από την σύνδεση της με το όχημα.



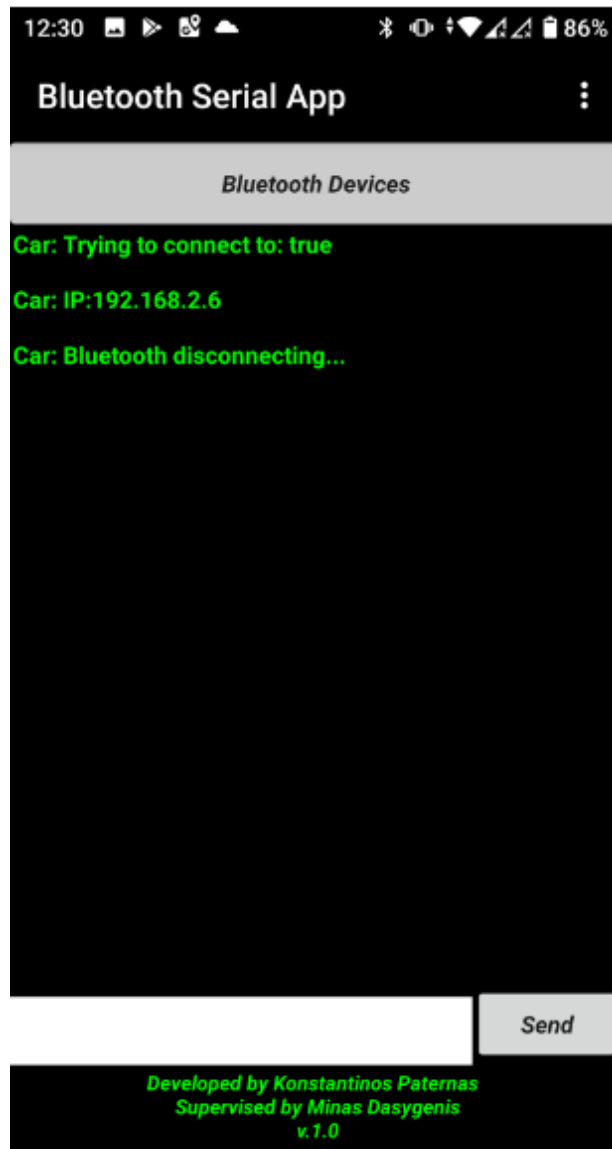
Εικόνα 30: Αρχική Οθόνη Android Εφαρμογής

Έπειτα, πατώντας το κουμπί Bluetooth Devices που βρίσκεται στο πάνω μέρος της οθόνης (**Εικόνα 30**) και εφόσον έχουμε ενεργοποιήσει το Bluetooth της συσκευής μας εμφανίζονται οι διαθέσιμες συσκευές (**Εικόνα 31**). Στην συνέχεια, επιλέγουμε την συσκευή με όνομα Car Bluetooth.



*Εικόνα 31: Οθόνη Διαθέσιμων Συσκευών*

Όταν επιλεγθεί η συσκευή, πραγματοποιείται προσπάθεια σύνδεσης και στην συνέχεια εμφανίζεται η κατάσταση της σύνδεσης στην αρχική οθόνη. Πιο συγκεκριμένα, εάν έχει γίνει επιτυχής σύνδεση επιστρέφει την τιμή true ενώ αντίθετα την τιμή false. Έπειτα, κάνει προσπάθεια σύνδεσης στο τελευταίο αποθηκευμένο δίκτυο και εάν συνδεθεί επιστρέφει την διεύθυνση IP και απενεργοποιεί το Bluetooth. Στην παρακάτω εικόνα (Εικόνα 32), φαίνεται η συγκεκριμένη υλοποίηση.



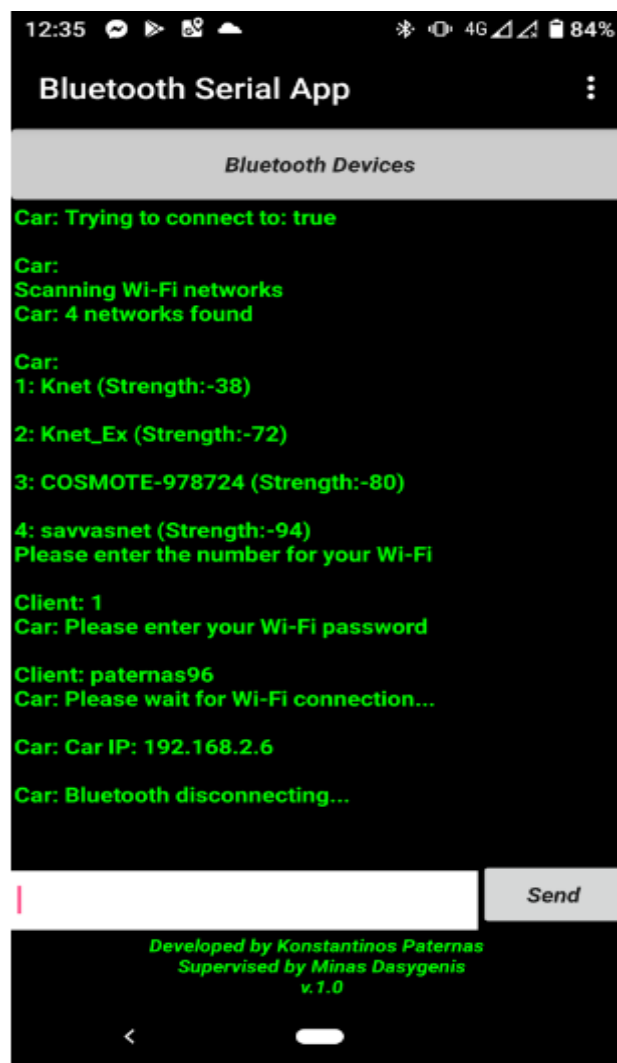
Εικόνα 32: Σύνδεση σε Δίκτυο

Στην συνέχεια, αν το τελευταίο αποθηκευμένο δίκτυο δεν είναι διαθέσιμο, πραγματοποιείται αναζήτηση και εμφανίζονται τα διαθέσιμα δίκτυα. Κάθε δίκτυο που εμφανίζεται αποτελείται από έναν αριθμό, τον οποίο πληκτρολογεί και στέλνει ο χρήστης για



να δηλώσει το δίκτυο που επιθυμεί να συνδεθεί. Έπειτα, ζητείται να συμπληρωθεί ο κωδικός του δικτύου και πραγματοποιείται έλεγχος εγκυρότητας. Στην περίπτωση που είναι λάθος επαναλαμβάνεται η παραπάνω διαδικασία. Αντίθετα, εμφανίζεται η IP διεύθυνση του δικτύου και απενεργοποιείται το Bluetooth.

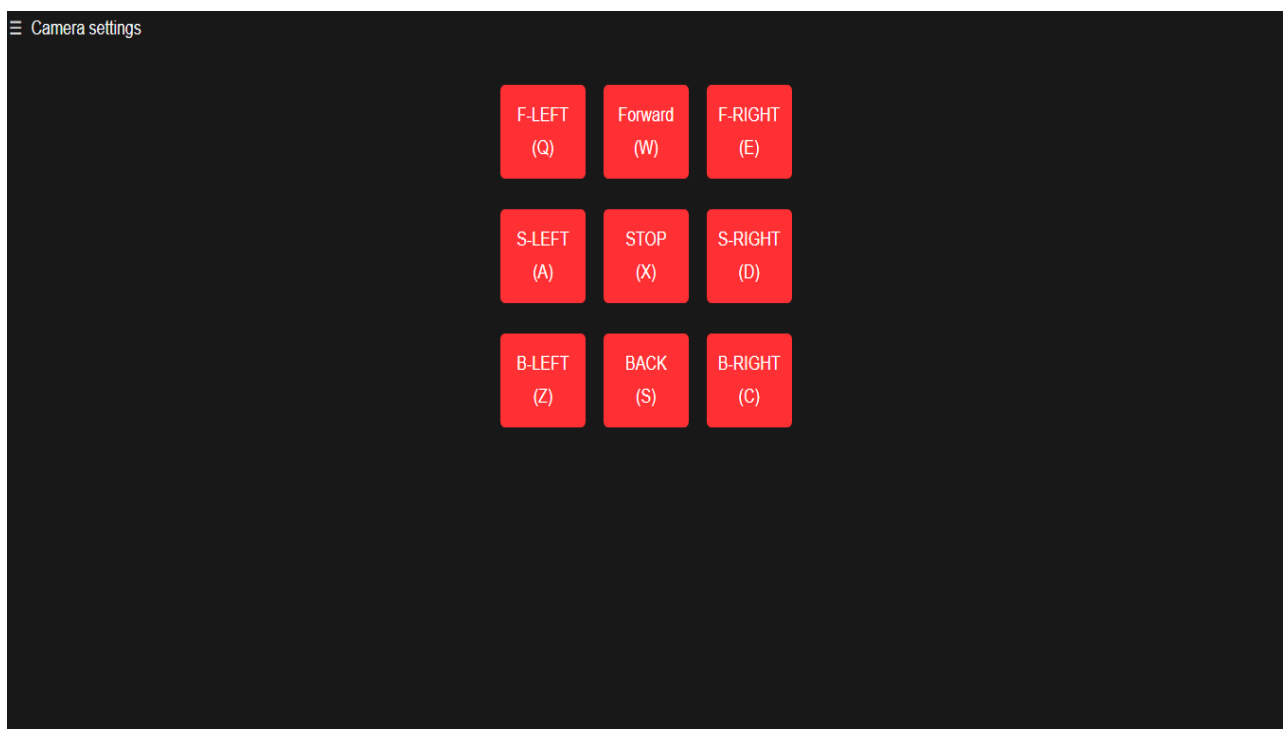
Πιο αναλυτικά, φαίνεται στην παρακάτω εικόνα (**Εικόνα 33**) η συγκεκριμένη υλοποίηση.



Εικόνα 33: Διαδικασία Σύνδεσης σε Ασύρματο Δίκτυο

### 4.3 Εφαρμογή Ιστού (Web App)

Στην συγκεκριμένη ενότητα, αναλύεται η εφαρμογή ιστού που υλοποιήθηκε στην παρούσα διπλωματική εργασία. Πρόκειται επί της ουσίας, για μια ιστοσελίδα όπου ο χρήστης χειρίζεται το όχημα του χρησιμοποιώντας το πρόγραμμα περιήγησης (browser) που διαθέτει. Ο κώδικάς της συγκεκριμένης ιστοσελίδας βρίσκεται συμπιεσμένος με gzip συμπίεση και έχει μορφή δεκαεξαδικού αριθμού [20]. Το gzip είναι μια μέθοδος συμπίεσης αρχείων που είναι ιδιαίτερα χρήσιμη για τις ταχύτερες μεταφορές του δικτύου. Η συμπίεση αυτή, επιτρέπει στον μικροελεγκτή (διακομιστή ιστού) να παρέχει μικρότερο μέγεθος αρχείο και έτσι να φορτώνει ταχύτερα την ιστοσελίδα στον χρήστη. Αναλυτικότερα, παρακάτω (**Εικόνα 34**) φαίνεται η αρχική μορφή της ιστοσελίδας πριν την ενεργοποίηση της κάμερας.



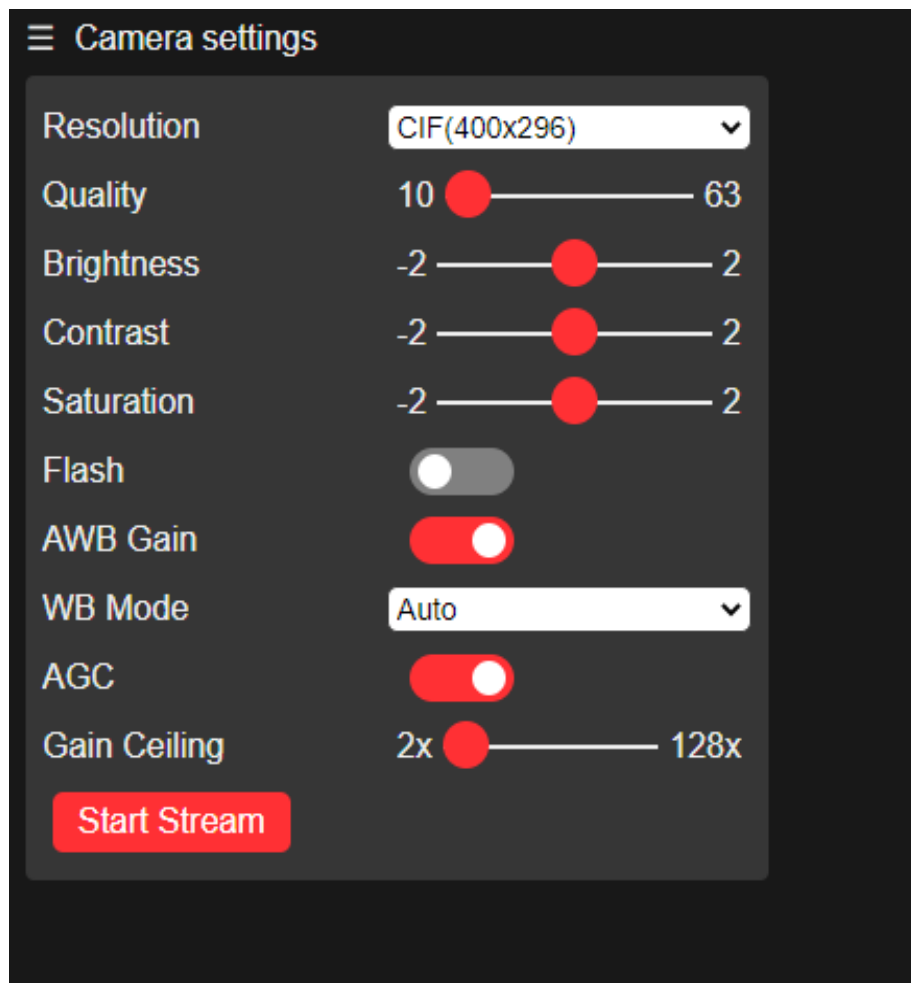
Εικόνα 34: Αρχική Οθόνη Ιστοχώρου

Όπως φαίνεται στην **Εικόνα 34**, η αρχική οθόνη της εφαρμογής περιέχει το χειριστήριο του οχήματος και πάνω σε κάθε κουμπί του αναγράφεται και το αντίστοιχο πλήκτρο για τον χειρισμό από το πληκτρολόγιο. Στον παρακάτω πίνακα (**Πίνακας 8**), αναγράφονται πιο αναλυτικά οι λειτουργίες του χειριστηρίου για κάθε κουμπί.

Κουμπί	Πληκτρολόγιο	Ενέργεια
Forward	W	Μπροστά
Back	S	Πίσω
S-Left	A	Αριστερά
S-Right	D	Δεξιά
F-Left	Q	Μπροστά Αριστερά
F-Right	E	Μπροστά Δεξιά
B-Left	Z	Πίσω Αριστερά
B-Right	C	Πίσω Δεξιά

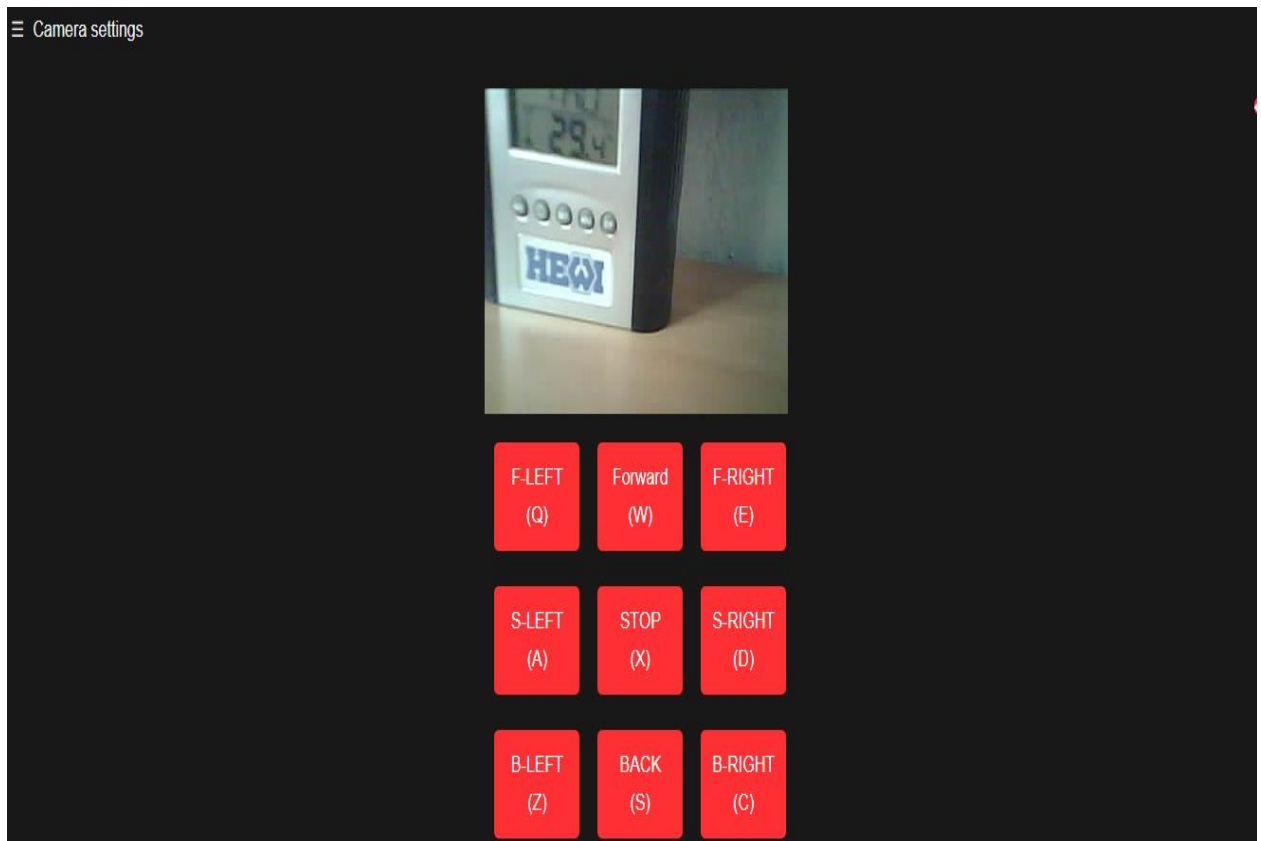
*Πίνακας 8: Πίνακας Εντολών Χειριστηρίου*

Στην συνέχεια, (Εικόνα 35) παρουσιάζεται το μενού της εφαρμογής που βρίσκονται οι ρυθμίσεις της κάμερας (Πίνακας 7), ο διακόπτης για την λειτουργία με φλας και το κουμπί για την έναρξη της μετάδοσης εικόνας.



Εικόνα 35: Μενού Ρυθμίσεων Κάμερας

Πιο συγκεκριμένα, με το πάτημα του κουμπιού Start Stream ξεκινάει η ζωντανή μετάδοση εικόνας από το όχημα. Στην συνέχεια, ο χρήστης αφού έχει ολοκληρώσει τις ρυθμίσεις που επιθυμεί για την κάμερα, είναι έτοιμος για τον χειρισμό του οχήματος. Αναλυτικότερα, φαίνεται στην παρακάτω εικόνα (**Εικόνα 36**) η μορφή του ιστοχώρου κατά την μετάδοση εικόνας.



*Εικόνα 36: Μετάδοση Εικόνας*

Στην συνέχεια, στην υποενότητα **4.3.1** αναλύονται σημαντικά κομμάτια του κώδικα της εφαρμογής ιστού που παρουσιάστηκε στην παρούσα ενότητα.

### 4.3.1 Κώδικας Ιστοχώρου

Στην παρούσα υποενότητα, παρουσιάζονται τα πιο σημαντικά κομμάτια κώδικα που παίζουν ένα καθοριστικό ρόλο στην υλοποίηση της ιστοσελίδας. Αρχικά, η ιστοσελίδα που αναπτύχθηκε αποτελείται από ένα αρχείο κώδικα HTML που ονομάζεται index.html. Το αρχείο αυτό όπως προαναφέρθηκε είναι συμπιεσμένο έτσι ώστε να γίνεται ταχύτερη η διαχείριση του από τον διακομιστή.

```
<input type="checkbox" id="nav-toggle-cb" checked="checked">
<nav id="menu">
  <div class="input-group" id="framesize-group">
    <label for="framesize">Resolution</label>
    <select id="framesize" class="default-action">
      <option value="6">VGA (640x480) </option>
      <option value="5" selected="selected">CIF (400x296) </option>
      <option value="4">QVGA (320x240) </option>
      <option value="3">HQVGA (240x176) </option>
      <option value="0">QQVGA (160x120) </option>
    </select>
  </div>
  <div class="input-group" id="quality-group">
    <label for="quality">Quality</label>
    <div class="range-min">10</div>
    <input type="range" id="quality" min="10" max="63" value="10" class="default-action">
    <div class="range-max">63</div>
  </div>
  <div class="input-group" id="brightness-group">
    <label for="brightness">Brightness</label>
    <div class="range-min">-2</div>
    <input type="range" id="brightness" min="-2" max="2" value="0" class="default-action">
    <div class="range-max">2</div>
  </div>
  <div class="input-group" id="contrast-group">
    <label for="contrast">Contrast</label>
    <div class="range-min">-2</div>
    <input type="range" id="contrast" min="-2" max="2" value="0" class="default-action">
    <div class="range-max">2</div>
  </div>
  <div class="input-group" id="saturation-group">
    <label for="saturation">Saturation</label>
    <div class="range-min">-2</div>
    <input type="range" id="saturation" min="-2" max="2" value="0" class="default-action">
    <div class="range-max">2</div>
  </div>
</div>
```

Εικόνα 37: Κομμάτι Κώδικα του Μενού

Στην παραπάνω εικόνα (Εικόνα 37), παρουσιάζεται ένα κομμάτι κώδικά που δείχνει τον τρόπο με τον οποίο είναι δομημένο το μενού του ιστοχώρου. Πιο συγκεκριμένα, σε κάθε ενότητα (div) υπάρχει και μια από τις ρυθμίσεις της κάμερας του μικροελεγκτή καθώς και το όνομα της κλάσης στην οποία βρίσκεται. Τέλος, φαίνεται το id κάθε λειτουργίας αλλά και οι τιμές που παίρνει.

Έπειτα, ακολουθεί μια παρουσίαση των κουμπιών χειρισμού του οχήματος (**Εικόνα 38**) καθώς και του τρόπου επικοινωνίας με τον μικροελεγκτή.

```
<p align=center>
  <button style="width:90px;height:80px;display: inline-block" onmousedown="javascript:car_do('move','left');"
  onmouseup="javascript:car_do('move','stop');" onkeydown="javascript:keydownFunction();"
  onkeyup="javascript:keyupFunction();">F-LEFT (Q) </button>&nbsp;

  <button style="width:90px;height:80px;display: inline-block" onmousedown="javascript:car_do('move','forward');"
  onmouseup="javascript:car_do('move','stop');" onkeydown="javascript:keydownFunction();"
  onkeyup="javascript:keyupFunction();">Forward (W) </button>&nbsp;

  <button style="width:90px;height:80px;display: inline-block" onmousedown="javascript:car_do('move','fright');"
  onmouseup="javascript:car_do('move','stop');" onkeydown="javascript:keydownFunction();"
  onkeyup="javascript:keyupFunction();">F-RIGHT (E) </button>&nbsp;

</p>

<p align=center>
  <button style="width:90px;height:80px;display: inline-block" onmousedown="javascript:car_do('move','left');"
  onmouseup="javascript:car_do('move','stop');" onkeydown="javascript:keydownFunction();"
  onkeyup="javascript:keyupFunction();">S-LEFT (A) </button>&nbsp;

  <button style="width:90px;height:80px;display: inline-block" onmousedown="javascript:car_do('move','stop');"
  onmouseup="javascript:car_do('move','stop');" onkeydown="javascript:keydownFunction();"
  onkeyup="javascript:keyupFunction();">STOP (X) </button>&nbsp;

  <button style="width:90px;height:80px;display: inline-block" onmousedown="javascript:car_do('move','right');"
  onmouseup="javascript:car_do('move','stop');" onkeydown="javascript:keydownFunction();"
  onkeyup="javascript:keyupFunction();">S-RIGHT (D) </button>&nbsp;
```

Εικόνα 38: Κώδικας Κουμπιών Χειρισμού

Όπως φαίνεται και στην παραπάνω εικόνα (**Εικόνα 38**), κατά το πάτημα ενός κουμπιού ενεργοποιείται η συνάρτηση `car_do` που με την χρήση `jquery` εκτελεί εντολές για την κίνηση του οχήματος. Πιο συγκεκριμένα, η συνάρτηση `car_do` είναι η συνάρτηση που δίνει τις εντολές για την κίνηση του οχήματος. Με την χρήση `jquery`, επιστρέφει στον μικροελεγκτή δυο παραμέτρους: την ενέργεια `action` (π.χ `move`) και την τιμή `val` (π.χ `right`) (**Εικόνα 39**). Η συνάρτηση αυτή, καλείται να κινήσει το όχημα όταν το αντίστοιχο κουμπί είναι επιλεγμένο με το ποντίκι (`onmousedown`) ή όταν το αντίστοιχο πλήκτρο από το πληκτρολόγιο είναι πατημένο (`onkeydown`). Την στιγμή που αποδεσμεύεται το κουμπί το όχημα σταματάει να κινείται (`onmouseup`, `onkeyup`). Η ενέργεια αυτή προστέθηκε έτσι ώστε να γίνει πιο ρεαλιστικός ο χειρισμός του οχήματος.

```
function car_do(action, val)
{
  if(action!=null)
  {
    const query = `/car?var=${action}&val=${val}`

    fetch(query)
      .then(response => {
        console.log(`request to ${query} finished, status: ${response.status}`)
      })
  }
}
```

Εικόνα 39: Συνάρτηση Χειρισμού του Οχήματος `Car-do`



Το `onmousedown` [21] είναι ένα συμβάν (event) που εκτελείται όταν ένας χρήστης πατά ένα κουμπί ποντικιού πάνω από ένα στοιχείο. Αντίστοιχα, το `onmouseup` [22] εκτελείται όταν ένας χρήστης απελευθερώνει ένα κουμπί ποντικιού πάνω από ένα στοιχείο. Όσον αφορά τα κουμπιά του πληκτρολογίου υπάρχουν και εκεί τα αντίστοιχα συμβάντα.

Το `onkeydown` [23] εκτελείται όταν ο χρήστης πιέζει ένα πλήκτρο στο πληκτρολόγιο. Αντίθετα το `onkeyup` [24], συμβαίνει όταν ο χρήστης απελευθερώνει ένα πλήκτρο.

```
function keydownFunction() { //on key press control

    if (event.keyCode === 87) {
        if(i== false){
            car_do('move','forward'); // keyCode for W is 65
            //alert("Forward");
            i=true;}
        }

        else if (event.keyCode === 65){ // keyCode for A is 65
            if(i== false){
                car_do('move','left');
                //alert("Left");
                i=true;}
            }
        else if (event.keyCode === 68){ // keyCode for D is 68
            if(i== false){
                car_do('move','right');
                //alert("Right");
                i=true;}
            }
        else if(event.keyCode === 83){ // keyCode for S is 83
            if(i== false){
                car_do('move','back');
                //alert("Back");
                i=true;}
            }
        }
    }
```

Εικόνα 40: Συνάρτηση `KeydownFunction`

Ο ρόλος της συνάρτησης `keydownFunction` που φαίνεται παραπάνω (**Εικόνα 40**) είναι να ελέγχει ποιο από τα πλήκτρα είναι πατημένο. Πιο αναλυτικά, όπως είδαμε στην **Εικόνα 38**, η συνάρτηση αυτή καλείται όταν ένα πλήκτρο είναι πατημένο (`onkeydown`) και στην συνέχεια πραγματοποιεί έλεγχο με τον κωδικό αριθμό του κάθε πλήκτρου (`keyCode`), έτσι ώστε να εντοπίσει ποιο πλήκτρο είναι πατημένο και να στείλει μέσω της `car_do` την κατάλληλη εντολή για την κίνηση του οχήματος. Αντίθετα, η συνάρτηση `keyupFunction` (**Εικόνα 41**) καλείται όταν απελευθερωθεί το πατημένο πλήκτρο (`onkeyup`) και μέσω της `car_do` ακινητοποιεί το όχημα.

```
function keyupFunction() {  
  
    if (event.keyCode === 87) {  
  
        car_do('move','stop');          // keyCode for W is 65  
        //alert("Forward");  
        i=false;  
    }  
  
    else if (event.keyCode === 65){ // keyCode for A is 65  
        car_do('move','stop');  
        //alert("Left");  
        i=false;  
    }  
    else if (event.keyCode === 68){ // keyCode for D is 68  
        car_do('move','stop');  
        //alert("Right");  
        i=false;  
    }  
    else if(event.keyCode === 83){ // keyCode for S is 83  
        car_do('move','stop');  
        //alert("Back");  
        i=false;  
    }  
    else if(event.keyCode === 81 ){ // keyCode for Q is 81  
        car_do('move','stop');  
        //alert("F-LEFT");  
        i=false;  
    }  
    else if(event.keyCode === 69 ){ // keyCode for E is 69  
        car_do('move','stop');  
        //alert("F-RIGHT");  
        i=false;  
    }  
}
```

Εικόνα 41: Συνάρτηση `KeyUpFunction`

#### 4.4 Εφαρμογή Η/Υ (Desktop App)

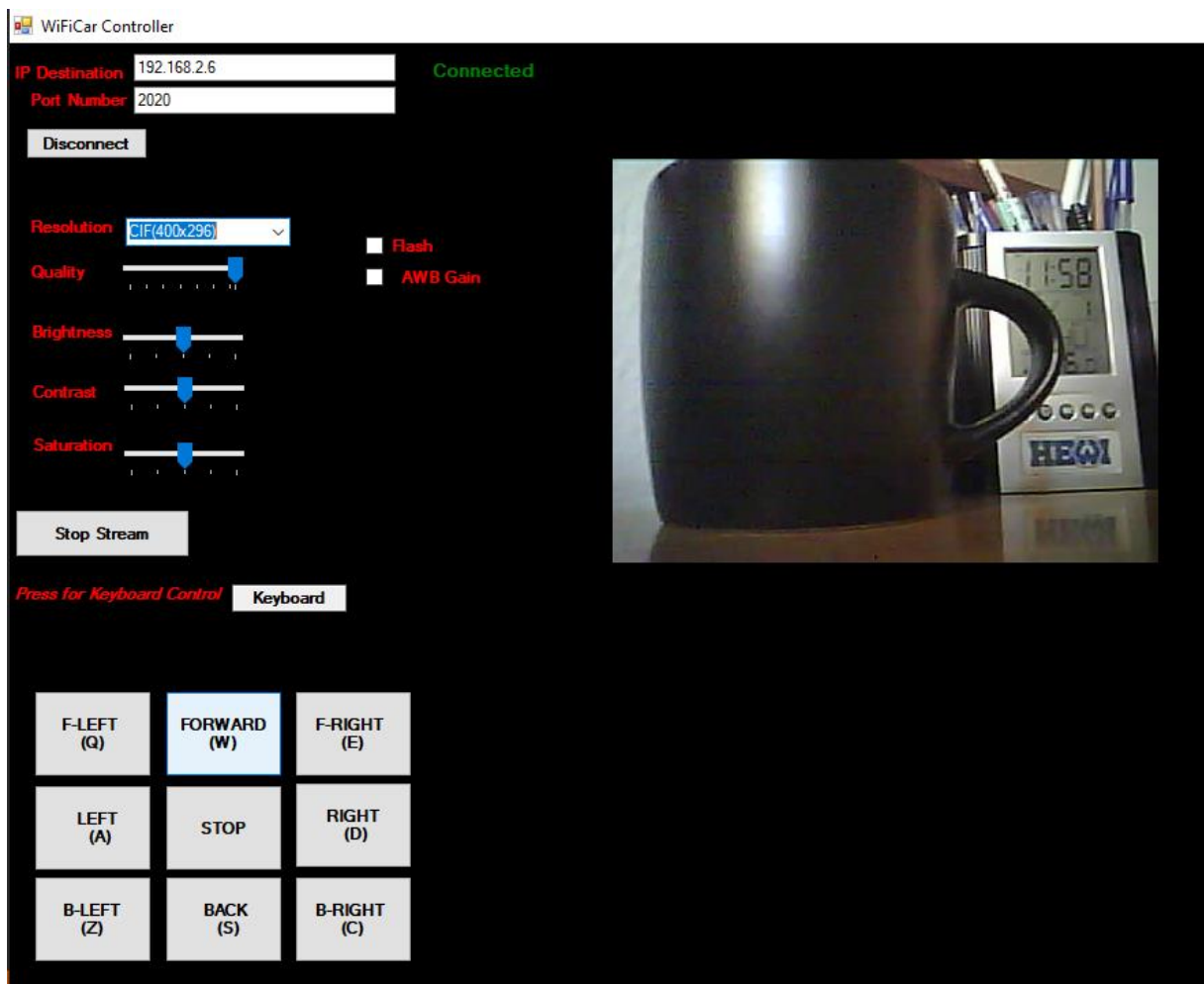
Για τον έλεγχο του οχήματος, εκτός από την εφαρμογή ιστού που είδαμε στην προηγούμενη ενότητα, υλοποιήθηκε και μια εφαρμογή σε Windows χρησιμοποιώντας την ολοκληρωμένη πλατφόρμα του Visual Studio. Στην ενότητα αυτήν, παρουσιάζεται ο τρόπος χρήσης της εφαρμογής αυτής και αναλύεται η λειτουργία της.

Πιο αναλυτικά, υλοποιήθηκε το εκτελέσιμο αρχείο WiFiCarController.exe. Ταυτόχρονα, δημιουργήθηκε και ένας εγκαταστάτης (installer) που εγκαθιστά την εφαρμογή και προσθέτει κάποια αρχεία βιβλιοθηκών δυναμικής σύνδεσης (dll) που απαιτεί για την αποτελεσματική της λειτουργία. Επίσης, τοποθετεί το εκτελέσιμο αρχείο στον φάκελο που επιλέγει ο χρήστης και εξάγει στο ίδιο σημείο τα dll αρχεία. Κατά την εκτέλεση της, εμφανίζεται η κύρια οθόνη (Εικόνα 42).



Εικόνα 42: Κύρια Οθόνη Εφαρμογής (Desktop App)

Όπως φαίνεται στην παραπάνω **Εικόνα 42**, και συγκεκριμένα πάνω αριστερά βρίσκεται το πλαίσιο, στο οποίο μπορούμε να εισάγουμε την IP διεύθυνση του οχήματος και ακριβώς από κάτω την θύρα επικοινωνίας (port). Στην συνέχεια, υπάρχουν οι λειτουργίες για την ρύθμιση της κάμερας όπως ακριβώς και στην εφαρμογή ιστοχώρου που αναλύθηκε στην προηγούμενη ενότητα. Έπειτα, υπάρχει το κουμπί για την ενεργοποίηση της μετάδοσης εικόνας και από κάτω ακόμη ένα κουμπί που ενεργοποιεί την χρήση του πληκτρολογίου για τον χειρισμό του οχήματος. Τέλος, είναι τα κουμπιά για τον χειρισμό του οχήματος με την χρήση του ποντικιού.



Εικόνα 43: Ενεργοποίηση Κάμερας

Στην **Εικόνα 43**, παρουσιάζεται η κύρια οθόνη της εφαρμογής με την κάμερα ενεργοποιημένη. Στην κατάσταση αυτή, η εφαρμογή έχει συνδεθεί με το όχημα και είναι έτοιμη

για χρήση. Ο χρήστης, μπορεί να αλλάξει τις ρυθμίσεις της κάμερας και στην συνέχεια να επιλέξει τον τρόπο που επιθυμεί να χειριστεί το όχημα (ποντίκι, πληκτρολόγιο).

#### 4.4.1 Κώδικας Εφαρμογής (Desktop App)

Στην υποενότητα αυτή, παρουσιάζονται κάποια από τα πιο σημαντικά κομμάτια κώδικα της Windows εφαρμογής που υλοποιήθηκε. Αρχικά, το αρχείο που περιλαμβάνει τον κώδικα της συγκεκριμένης υλοποίησης είναι το Form1.cs και έχει παραχθεί αυτόματα από την σχεδιαστή περιβάλλοντος του Visual Studio.

Όπως είχε αναφερθεί και στις προηγούμενες ενότητες, η επικοινωνία του οχήματος με την εφαρμογή Windows επιτυγχάνεται κυρίως με την χρήση του πρωτοκόλλου UDP. Στην εικόνα που ακολουθεί (**Εικόνα 44**) φαίνεται ο τρόπος σύνδεσης της εφαρμογής με το όχημα. Πιο αναλυτικά, πραγματοποιείται έλεγχος εγκυρότητας της διεύθυνσης και στην συνέχεια, στέλνοντας Ping σε αυτή, πραγματοποιείται έλεγχος για την κατάσταση της σύνδεσης. Έπειτα, με την μέθοδο Connect γίνεται η επικοινωνία με τον εξυπηρετητή και πραγματοποιείται η σύνδεση.

```
private void Connection()
{
    String url = textBox_ip.Text.Trim();
    bool flag = IPAddress.TryParse(url, out IP);
    string[] splitValues = url.Split('.');
    if (flag==true && splitValues.Length == 4){
        int Port = 2020;
        Ping p1 = new Ping();

        PingReply PR = p1.Send(url);
        if(PR.Status.ToString().Equals("Success")&& Convert.ToInt16(textBox_port.Text) == Port ){
            udpClient.Connect(url,Convert.ToInt16(textBox_port.Text));

            disconnect_button.Visible = true;
            Connect.Visible = false;
            Status.Text = "Connected";
            Status.ForeColor = System.Drawing.Color.Green;
        }else{
            MessageBox.Show("Connection Failed!\n Try again.", "Important Message", MessageBoxButtons.OK, MessageBoxIcon.Error);
            disconnect_button.Visible = false;
            Connect.Visible = true;
            Status.Text = "Disconnected";
            Status.ForeColor = System.Drawing.Color.Red;
        }
    }
}
```

Εικόνα 44: Συνάρτηση Σύνδεσης Εφαρμογής (Connection)

Κατά την ανάπτυξη της εφαρμογής, δημιουργήθηκε η ανάγκη χειρισμού του οχήματος με την χρήση του ποντικιού αλλά και με την χρήση του πληκτρολογίου. Αρχικά, στην πρώτη περίπτωση αυτό επιτυγχάνεται χρησιμοποιώντας έναν διαχειριστή γεγονότος ποντικιού (MouseEventHandler) και στην συνέχεια με την μέθοδο Send, η οποία στέλνει την κάθε εντολή, αφού πρώτα έχει κωδικοποιηθεί καταλλήλα το κάθε αλφαριθμητικό. Ακολουθεί, η **Εικόνα 45** που φαίνεται πιο αναλυτικά η λειτουργία του.

```
private void F_LEFT_MouseDown(object sender, MouseEventArgs e)
{
    if(disconnect_button.Visible==true){

        Byte[] senddata = Encoding.ASCII.GetBytes("f1");
        udpClient.Send(senddata, senddata.Length);
    }
}

private void F_LEFT_MouseUp(object sender, MouseEventArgs e)
{
    if(disconnect_button.Visible==true){

        Byte[] senddata = Encoding.ASCII.GetBytes("s");
        udpClient.Send(senddata, senddata.Length);
    }
}
```

*Εικόνα 45: Διαχειριστής Γεγονότος Ποντικιού (MouseEventHandler)*

Στην δεύτερη περίπτωση, για την χρήση του πληκτρολογίου απαιτείται αρχικά η επιλογή της περιοχής Keyboard (**Εικόνα 42**). Έπειτα, πραγματοποιείται έλεγχος με βάση τον κωδικό αριθμό του κάθε πλήκτρου (keycode) και αντίστοιχα γίνεται η αποστολή της κατάλληλης εντολής. Βέβαια προτού ολοκληρωθεί η αποστολή εντολής, πραγματοποιείται έλεγχος με μια μεταβλητή τύπου Boolean προκειμένου να αποσταλεί μια φορά στο διάστημα που το πλήκτρο κρατείται πατημένο. Στην παρακάτω εικόνα (**Εικόνα 46**) φαίνεται ένα κομμάτι κώδικα της συγκεκριμένης υλοποίησης.

```
bool flag = false;

private void textBox1_KeyDown_1(object sender, KeyEventArgs e)
{
    if (e.KeyCode == Keys.W)
    {
        if(flag==false)
        {
            flag = true;

            Byte[] senddata = Encoding.ASCII.GetBytes("f");
            udpClient.Send(senddata, senddata.Length);
        }
    }
    else if (e.KeyCode == Keys.Q)
    {
        if(flag==false)
        {
            flag = true;

            Byte[] senddata = Encoding.ASCII.GetBytes("f1");
            udpClient.Send(senddata, senddata.Length);
        }
    }
}
```

Εικόνα 46: Κώδικας για Χειρισμό Από Πληκτρολόγιο

Στην συνέχεια, αναλύεται ο τρόπος λειτουργίας της μετάδοσης εικόνας καθώς και των ρυθμίσεων της κάμερας που διαθέτει η συγκεκριμένη υλοποίηση. Αρχικά, η μετάδοση οθόνης πραγματοποιείται με την χρήση του πλαισίου AForge.net. Το πλαίσιο αυτό, χρησιμοποιεί τον MJPEG μεταδοτή και παίρνει σαν παράμετρο την διεύθυνση και την θύρα που μας δίνει ο μικροελεγκτής για την μετάδοση εικόνας. Παρακάτω (**Εικόνα 47**) φαίνεται πιο αναλυτικά η δομή της συγκεκριμένης υλοποίησης.

```
Stream = new MJPEGStream("http://" + url + ":81/stream");
Stream.NewFrame += stream_NewFrame;
Stream.Start();
button_start.Visible = false;
stopStream.Visible = true;
```

Εικόνα 47: Κώδικας MJPEG Stream

Όσον αφορά τις ρυθμίσεις της κάμερας, χρησιμοποιούνται Http αιτήματα (requests) για την επικοινωνία με τον μικροελεγκτή (**Εικόνα 48**). Πιο συγκεκριμένα, έχει υλοποιηθεί η συνάρτηση Request όπου δημιουργεί Http αιτήματα προς την διεύθυνση που διαχειρίζεται τις ρυθμίσεις τις κάμερας. Στην διεύθυνση αυτή, υπάρχουν δύο παράμετροι (var, val) που όπως έχει αναφερθεί στην ενότητα που αφορούν την λειτουργία και την τιμή της. Αναλυτικότερα, απεικονίζεται παρακάτω το αντίστοιχο κομμάτι κώδικα.

```
private void Request(String var, int val)
{
    String url = textBox_ip.Text.Trim();
    String url_control = "http://" + url + "/control?var=" + var + "&val=" + val;

    if (disconnect_button.Visible==true)
    {
        HttpWebRequest request = WebRequest.Create(url_control) as HttpWebRequest;

        HttpWebResponse response = request.GetResponse() as HttpWebResponse;
        Stream stream = response.GetResponseStream();
    }
}
```

Εικόνα 48: Συνάρτηση Request



#### 4.4.2 Τρόπος Επικοινωνίας Εφαρμογής

Η επίτευξη της σύνδεσης της εφαρμογής με το όχημα επιτυγχάνεται με την επιλογή του πρωτοκόλλου UDP. Το πρωτόκολλο αυτό, χαρακτηρίζεται από την ταχύτητα που προσφέρει και θεωρείται ιδανικό για χρήση από εφαρμογές που ανταλλάσσουν δεδομένα σε πραγματικό χρόνο (real time app). Στην ενότητα αυτή, αναφέρεται ο τρόπος επικοινωνίας διακομιστή-πελάτη και αναλύονται τα πακέτα που λαμβάνει ο διακομιστής για την εκτέλεση κάθε εντολής.

Πιο συγκεκριμένα, κάθε πακέτο αποτελείται από 43 Bytes εκ των οποίων τα 42 Byte αφορούν τα χαρακτηριστικά του πακέτου και το 1 Byte την εντολή κίνησης. Κάθε αλφαριθμητικό που αποστέλλεται είναι μεταφρασμένο σε ASCII κωδικοποίηση. Αρχικά για την εντολή Forward (Μπροστά) αποστέλλεται το αλφαριθμητικό f και μεταφράζεται ως 66. Για την εντολή, Backwards (Πίσω) αποστέλλεται το αλφαριθμητικό b που μεταφράζεται ως 62. Έπειτα, η εντολή Left (Αριστερά) αντιστοιχεί στο αλφαριθμητικό l που μεταφράζεται 6C και η εντολή δεξιά με αλφαριθμητικό r που μεταφράζεται ως 72. Τέλος, η εντολή Stop αντιστοιχεί στο αλφαριθμητικό s με ASCII τιμή 73.

Εξαίρεση αποτελούν, οι εντολές που εκτελούν δυο ενέργειες ταυτόχρονα. Οι εντολές αυτές είναι οι: Μπροστά και Αριστερά, Μπροστά και Δεξιά, Πίσω και Δεξιά, Πίσω και Αριστερά. Συγκεκριμένα οι εντολές αυτές έχουν μέγεθος 2 Byte και το συνολικό τους πακέτο είναι 44 Byte. Πιο αναλυτικά η εντολή F-Left (Μπροστά και Αριστερά) αντιστοιχεί στο αλφαριθμητικό fl και μεταφράζεται ως 666c. Η εντολή F-Right (Μπροστά Δεξιά) αντιστοιχεί στο αλφαριθμητικό fr και μεταφράζεται ως 6672. Έπειτα, η εντολή B-Left (Πίσω και Αριστερά) με αλφαριθμητικό bl μεταφράζεται ως 626c. Τέλος, η εντολή B-Right (Πίσω και Δεξιά) με αλφαριθμητικό br μεταφράζεται 6272.

Τέλος, η λειτουργία της κάμερας επιτυγχάνεται με την χρήση Http πρωτοκόλλου. Πιο συγκεκριμένα, η μετάδοση εικόνας από το εξυπηρετητή γίνεται στην θύρα 81 της διεύθυνσης του προσθέτοντας “:81/stream”. Οπότε, όταν πραγματοποιηθεί σύνδεση και πατηθεί το κουμπί Start Stream καλείται η διεύθυνση αυτή από την μέθοδο Stream και μεταδίδεται η εικόνα.

#### 4.5 Σύνοψη Κεφαλαίου

Συνοψίζοντας στο παρόν κεφάλαιο, αναλύθηκε η συνολική λειτουργία του λογισμικού του συστήματος, καθώς και των επιμέρους υποσυστημάτων. Αναλύθηκαν και επεξηγήθηκαν τα σημαντικότερα κομμάτια κώδικα της υλοποίησης. Στην συνέχεια, ακολουθεί το τελευταίο κεφάλαιο του συγγράμματος, το οποίο παρουσιάζει τα γενικά χαρακτηριστικά της όλης υλοποίησης. Παράλληλα αναλύονται τα πλεονεκτήματα και μειονεκτήματα στις συγκεκριμένης υλοποίησης. Επιπλέον, αναφέρονται τα σημαντικότερα προβλήματα που προέκυψαν και οι λύσεις που δόθηκαν. Τέλος, δίνονται τα συμπεράσματα που προέκυψαν από την διεξαγωγή της παρούσας εργασίας, και αναφέρονται ιδέες για μεταγενέστερη εξέλιξη και αναβάθμιση του υπάρχοντος συστήματος.

## Κεφάλαιο 5 – Επίλογος

Μετά την ανάλυση του λογισμικού μέρους που πραγματοποιήθηκε στο προηγούμενο κεφάλαιο, έχει ολοκληρωθεί η παρουσίαση του συστήματος. Στο κεφάλαιο αυτό, παρουσιάζεται μια ανακεφαλαίωση με τα γενικά χαρακτηριστικά του συστήματος. Επίσης, παρουσιάζονται οι μετρήσεις σχετικά με το λογισμικό και υλικό μέρος του οχήματος και πραγματοποιείται ανάλυση SWOT (Strengths, Weaknesses, Opportunities, Threats). Τέλος, εξάγονται τα συμπεράσματα που προέκυψαν μετά την υλοποίηση του συστήματος αλλά και οι μελλοντικές επεκτάσεις που θα μπορούσαν να υπάρξουν.

### 5.1 Σύνοψη

Η λειτουργία ελέγχου του οχήματος είναι εφικτή, στην περίπτωση που το όχημα έχει πρόσβαση στο διαδίκτυο. Ο χρήστης, για τον έλεγχο του οχήματος μπορεί να κάνει χρήση και του ποντικιού και του πληκτρολογίου. Το όχημα με χρήση της κάμερας, που είναι εξοπλισμένος ο μικροελεγκτής, παρέχει τη δυνατότητα παρακολούθησης μέσω αυτής. Για τον χειρισμό μπορεί να χρησιμοποιηθεί, είτε η υλοποιημένη Windows εφαρμογή του συστήματος WiFiCar Controller, είτε η διαδικτυακή εφαρμογή στο πρόγραμμα περιήγησης του χρήστη. Και οι δυο εφαρμογές προσφέρουν τις ίδιες δυνατότητες και λειτουργίες. Επίσης, η Android εφαρμογή που υλοποιήθηκε διευκολύνει την σύνδεση σε ασύρματα δίκτυα χρησιμοποιώντας την Bluetooth επικοινωνία.

Οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν για την υλοποίηση του ιστοχώρου του συστήματος είναι οι: HTML, CSS, JavaScript, JQuery. Στην συνέχεια για το λογισμικό του μικροελεγκτή χρησιμοποιήθηκαν οι γλώσσες C και C++. Επιπλέον, η εφαρμογή Windows η αναπτύχθηκε σε γλώσσα C# και χρησιμοποιήθηκαν τα πλαίσια .NET και το AForge.

Όσον αφορά το υλικό μέρος του οχήματος σχεδιάστηκε και υλοποιήθηκε πλακέτα τυπωμένου κυκλώματος (PCB) που παρέχει δύο εξόδους τάσης 5V και 9V και ταυτόχρονα λειτουργεί και σαν οδηγός κινητήρων για το όχημα. Η υλοποίηση αυτή αποτελεί μια σταθερή κατασκευή που εγγυάται την σταθερότητα του οχήματος σε τυχόν συγκρούσεις και την καλύτερη απόκριση του συστήματος.

## 5.2 Μετρικά Συστήματος

Το σύστημα αποτελείται τόσο από υλικό, όσο και από λογισμικό μέρος. Για την καλύτερη παρουσίαση των δυνατοτήτων και της ποιότητας του, πραγματοποιήθηκε ένα σύνολο από μετρήσεις. Οι μετρήσεις αυτές αφορούν και τα δύο μέρη του συστήματος (υλικό, λογισμικό) τόσο μεμονωμένα, όσο και συνδυαστικά.

### 5.2.1 Μετρικά Οχήματος

Συγκεκριμένα, όλη η κατασκευή που υλοποιήθηκε έχει βάρος 600gr και μπορεί να κινηθεί οριακά με φορτίο μέχρι 400gr. Η μέγιστη ταχύτητα που μπορεί να φτάσει είναι τα 20km/h. Βέβαια η ταχύτητα κίνησης του επηρεάζεται σημαντικά από ένα μεγάλο πλήθος παραγόντων, όπως το βάρος του οχήματος, η κλίση και η πρόσφυση της επιφάνειας κίνησης. Η αυτονομία κίνησης του οχήματος μπορεί να φτάσει περίπου 100 λεπτά και ο χρόνος φόρτισης των μπαταριών είναι περίπου 2 ώρες (120 λεπτά).

Μέτρηση	Αριθμός
Βάρος	600 gr
Μέγιστη Ταχύτητα	20 km/h
Αυτονομία	100 min
Χρόνος φόρτισης	120 min

Πίνακας 9: Μετρήσεις Οχήματος

### 5.2.2 Μετρικά Μικροελεγκτή

Το λογισμικό του μικροελεγκτή αποτελείται από δύο αρχεία πηγαίου κώδικα. Στα αρχεία αυτά γίνεται χρήση δύο γλωσσών προγραμματισμού Wiring και C++. Πιο αναλυτικά, το 31% του κώδικα είναι Wiring και το υπόλοιπο 69% είναι C++. Όσον αφορά το μέγεθος του κώδικα είναι 820 kB και συνολικός αριθμός γραμμών είναι 645. Στον πίνακα παρακάτω (**Πίνακας 10**), παρουσιάζονται πιο αναλυτικά οι μετρήσεις της συγκεκριμένης υλοποίησης.

Μέτρηση	Αριθμός
Αρχεία	2
Συνολικό Μέγεθος	820kB
Συνολικές γραμμές	645

Πίνακας 10: Μετρήσεις Μικροελεγκτή

### 5.2.3 Μετρικά Εφαρμογής Ιστοχώρου

Η εφαρμογή ιστοχώρου που αναπτύχθηκε στην παρούσα διπλωματική εργασία, αποτελείται από ένα αρχείο κώδικα HTML το οποίο είναι συμπιεσμένο σε gzip μορφή για ταχύτερη εκτέλεση από τον διακομιστή και εξοικονόμηση χώρου. Το αρχείο αυτό, περιλαμβάνει όλες τις Script συναρτήσεις (JavaScript, JQuery) που χρησιμοποιήθηκαν καθώς και το CSS της ιστοσελίδας. Ο συνολικός αριθμός γραμμών κώδικα του είναι 804 γραμμές και το συνολικό του μέγεθος του 24kB. Ακολουθεί ο πίνακας (**Πίνακας 11**) που φαίνονται αναλυτικά οι παραπάνω μετρήσεις.

Μέτρηση	Αριθμός
Αρχεία	1
Συνολικό Μέγεθος	24kB
Συνολικές Γραμμές	804

Πίνακας 11: Μετρήσεις Κώδικα Εφαρμογής Ιστού

## 5.2.4 Μετρικά Εφαρμογής Windows

Η εφαρμογή Windows που αναπτύχθηκε στην παρούσα διπλωματική εργασία, αποτελείται από ένα αρχείο κώδικα ονόματι Form1.cs. Το αρχείο αυτό περιλαμβάνει τον κώδικα C# καθώς και όλες τις συναρτήσεις που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής. Ο συνολικός αριθμός γραμμών κώδικα του είναι 696 γραμμές και το συνολικό του μέγεθος του 9,59MB. Ακολουθεί ο πίνακας (Πίνακας 12) που φαίνονται αναλυτικά οι παραπάνω μετρήσεις.

Μέτρηση	Αριθμός
Αρχεία	1
Συνολικό Μέγεθος	9,59MB
Συνολικές Γραμμές	696

Πίνακας 12: Μετρικά Εφαρμογής Windows

## 5.3 Ανάλυση SWOT

Στην ενότητα αυτή, πραγματοποιείται η ανάλυση SWOT της παρούσας διπλωματικής εργασίας. Πρόκειται ουσιαστικά, για ένα χρήσιμο εργαλείο, το οποίο εξετάζει τα Δυνατά Σημεία (Strengths), τις Αδυναμίες (Weaknesses), τις Ευκαιρίες (Opportunities) αλλά και τις Απειλές (Threats) του συστήματος. Αυτό βοηθάει σημαντικά, στον καθορισμό των μελλοντικών επεκτάσεων αλλά και των απαιτήσεων του.

### 5.3.1 Δυνατά Σημεία

Το σύστημα που υλοποιήθηκε στην συγκεκριμένη διπλωματική εργασία παρουσιάζει ένα πλήθος προτερημάτων έναντι άλλων παρόμοιων συστημάτων. Μερικά από τα δυνατά σημεία του συστήματος αναλύονται παρακάτω.

Αρχικά, η ιστοσελίδα μπορεί να χρησιμοποιηθεί από οποιονδήποτε υπολογιστή ή άλλη φορητή συσκευή αρκεί αυτή να έχει πρόσβαση στο διαδίκτυο. Επίσης, είναι συμβατή με όλα τα λειτουργικά συστήματα, αφού εκτελείται μέσω του περιηγητή διαδικτύου (browser). Μάλιστα έχει ελάχιστες απαιτήσεις υπολογιστικών πόρων συνεπώς μπορεί να εκτελεστεί από οποιοδήποτε μηχάνημα.

Η εφαρμογή Windows που υλοποιήθηκε, μπορεί να διευκολύνει την χρήση σε υπολογιστές που δεν έχουν πρόσβαση σε κάποιον περιηγητή. Προσφέρει της ίδιες δυνατότητες και λειτουργίες με την ιστοσελίδα, καθώς δεν έχει απαιτήσεις υπολογιστικών πόρων.

Επίσης, και οι δύο εφαρμογές ελέγχου του οχήματος μπορούν να χρησιμοποιηθούν από οποιαδήποτε τοποθεσία και οποιονδήποτε χρήστη. Αποτελούν εύκολες και φιλικές εφαρμογές ως προς τον χρήστη και δεν απαιτούν ιδιαίτερες γνώσεις από αυτόν για την εκτέλεση τους. Επιπλέον, και οι δύο εφαρμογές παρέχονται δωρεάν συνεπώς δεν υπάρχει κάποια οικονομική επιβάρυνση.

Η Android εφαρμογή αποτελεί σημαντικό προνόμιο της συγκεκριμένης διπλωματικής εργασίας αφού προσφέρει ευκολία σύνδεσης του οχήματος στα ασύρματα δίκτυα. Αποτελεί εύκολη και φιλική εφαρμογή που μπορεί εύκολα ο κάθε χρήστης να εκτελέσει.

Όσον αφορά την κατασκευή του οχήματος αποτελεί μια σταθερή κατασκευή που εγγυάται την αξιοπιστία σε βάθος χρόνου. Επίσης, η πλακέτα τυπωμένου κυκλώματος που υλοποιήθηκε προσφέρει μεγαλύτερη απόδοση στο όχημα, λιγότερο βάρος και σταθερότητα στις συνδέσεις των καλωδίων. Επίσης, αναγράφει τις εξόδους και τις εισόδους για το κάθε τι έτσι ώστε αν αποσυναρμολογηθεί από το όχημα, ο χρήστης να μπορεί εύκολα να την συναρμολογήσει.

### 5.3.2 Αδυναμίες

Η κύρια αδυναμία του συστήματος που υλοποιήθηκε, είναι το γεγονός ότι δεν έχει δοκιμαστεί σε μεγάλο αριθμό χρηστών και υπολογιστών. Επίσης, οι κινητήρες του οχήματος είναι οι εργοστασιακοί συνεπώς, δεν γνωρίζουμε την ποιότητα τους, αν δηλαδή είναι αξιόπιστοι για πολύωρη χρήση.

### 5.3.3 Ευκαιρίες

Την σύγχρονη εποχή που διανύουμε, οι αυξανόμενες ανάγκες μειώνουν σημαντικά τον ελεύθερο χρόνο των ανθρώπων ώστε να πραγματοποιήσουν τα χόμπι τους για τον μηχανοκίνητο αθλητισμό. Η κατάσταση δημιουργεί όλο και περισσότερες ευκαιρίες για τη χρήση του συστήματος, αφού ο χρήστης μπορεί εύκολα και γρήγορα να καλύψει τις ανάγκες των χρηστών.

Επίσης, μέσα από την υλοποίηση αυτή δίνεται η ευκαιρία σε ανθρώπους με αναπηρία να γνωρίσουν όσο καλύτερα γίνεται την έννοια του μηχανοκίνητου αθλητισμού καθώς και να ψυχαγωγηθούν.

#### 5.3.4 Απειλές

Η βασική απειλή που παρουσιάζεται είναι ο ανταγωνισμός, καθώς υπάρχουν πολλές παρόμοιες εφαρμογές, με πολλές παρόμοιες δυνατότητες ακόμα και εφαρμογές παρεμφερές πεδίου που όμως έχουν αρχίσει να υποστηρίζουν τέτοιου είδους υπηρεσίες.

#### 5.4 Ασφάλεια

Στην παρούσα διπλωματική εργασία χρησιμοποιείται ένα πλήθος μεθόδων ασφάλειας και προστασίας, τόσο του συστήματος, όσο και του γύρω περιβάλλοντος. Η ασφάλεια που εφαρμόστηκε είναι πολυεπίπεδη και έχει την έννοια τόσο του αγγλικού όρου Security, όσο και του Safety. Αρχικά, πραγματοποιήθηκαν ενέργειες έτσι ώστε να εξασφαλίζεται η προστασία του υλικού μέρους, αλλά και των ατόμων που έρχονται σε επαφή με αυτό. Αυτό επιτεύχθηκε με την προσθήκη κεντρικού διακόπτη για την τροφοδοσία του οχήματος, αλλά και με την χρήση καλά μονωμένων καλωδίων. Με αυτόν τον τρόπο προστατεύεται, τόσο το υλικό του οχήματος, όσο και το περιβάλλον γύρω του. Παράλληλα, χρησιμοποιήθηκαν τεχνικές όπως η κωδικοποίηση των αλφαριθμητικών, οι οποίες δεν επιτρέπουν τη χρήση του συστήματος από άτομα που δεν έχουν εξουσιοδότηση. Τέλος, πρέπει να αναφερθεί ότι όλα τα μέτρα ασφαλείας που υλοποιήθηκαν είναι στον προαπαιτούμενο βαθμό και σίγουρα υπάρχουν περιθώρια βελτίωσης.

#### 5.5 Μελλοντικές Επεκτάσεις

Στο σύστημα που αναπτύχθηκε όπως και σε κάθε σύστημα, πάντα υπάρχει χώρος για βελτίωση και μελλοντικές επεκτάσεις είτε περισσότερες είτε λιγότερες. Παρακάτω, αναφέρονται μερικές που αφορούν την παρούσα διπλωματική εργασία.

Μια σημαντική επέκταση που αφορά το υλικό κομμάτι του συστήματος που υλοποιήθηκε, είναι η προσθήκη ενός μεγαλύτερου κινητήρα 12V στον πίσω άξονα του οχήματος. Αυτή η τροποποίηση θα πρόσφερε μεγαλύτερη ταχύτητα υψηλότερο όριο βάρους



καθώς και ποιοτικότερη κίνηση του οχήματος. Τέλος, για τον καλύτερο έλεγχο του τιμονιού προτείνεται η αντικατάσταση του DC κινητήρα με έναν σερβοκινητήρα.

Επίσης, μια αξιόλογη προσθήκη θα μπορούσε να ήταν η ενσωμάτωση GPS έτσι ώστε να γίνεται υπολογισμός της απόστασης και της διαδρομής χρησιμοποιώντας τις συντεταγμένες του οχήματος.

Επιπλέον, θα μπορούσε να υλοποιηθεί πίστα στον χώρο που βρίσκεται το όχημα που να είναι σχεδιασμένη έτσι ώστε να αξιοποιούνται όλες οι δυνατότητες του καθώς επίσης και να διεξάγονται αγώνες.

Μια ακόμη δυνατή προσθήκη, είναι η ανάπτυξη εφαρμογής για κινητά τηλέφωνα (π.χ. Android) με σκοπό τον έλεγχο του οχήματος. Η υλοποίηση μιας τέτοιας εφαρμογής, θα εξασφάλιζε ακόμα έναν τρόπο για τον έλεγχο του οχήματος που θα αύξανε σημαντικά τον αριθμό των χρηστών του συστήματος.

Τέλος, μια σημαντική προσθήκη θα ήταν η υλοποίηση ενός σταθμού επαναφόρτισης του οχήματος. Πιο αναλυτικά, θα μπορούσε να υπάρχει ένας χώρος όπου ο χρήστης θα κατεύθυνε το όχημα και θα πραγματοποιούνταν ταχεία ασύρματη φόρτιση. Η υλοποίηση αυτή, θα αυτοματοποιούσε σε μεγάλο βαθμό το σύστημα καθώς και θα κέντριζε το ενδιαφέρον των χρηστών να το δοκιμάσουν.

## 5.6 Προβλήματα που αντιμετωπίστηκαν

Κατά την υλοποίηση του συστήματος προέκυψαν αρκετά προβλήματα. Τα προβλήματα αυτά αφορούν τόσο τον τομέα του υλικού, όσο και του λογισμικού. Για την επίτευξη καλύτερη και ποιοτικότερη λειτουργία του συστήματος ήταν απαραίτητη η επίλυση τους.

Αρχικό πρόβλημα ήταν η κακή ποιότητα της καλωδίωσης του οχήματος. Αυτό είχε ως αποτέλεσμα την συνεχή διακοπή της επικοινωνίας με το όχημα και την καθυστέρηση εκτέλεσης εντολών. Το πρόβλημα αυτό λύθηκε με την αντικατάσταση των καλωδίων με άλλα καλύτερης ποιότητας.

Επίσης, ένα ακόμη πρόβλημα που προέκυψε ήταν ότι το όχημα δεν εκτελούσε την εντολή “stop” κατά την απελευθέρωση ενός πλήκτρου που ήταν παρατεταμένα πατημένο. Αίτιο αυτού το προβλήματος ήταν πολλαπλή εκτέλεση της εντολής κίνησης όσο το κουμπί ήταν

πατημένο. Η επίλυση αυτού του προβλήματος έγινε με την χρήση μιας Boolean μεταβλητής που αλλάζει μετά την πρώτη εκτέλεση της κάθε εντολής μέχρι να εκτελεστεί η επόμενη.

## 5.7 Σύνοψη Κεφαλαίου

Εν κατακλείδι, στο κεφάλαιο αυτό παρουσιάστηκε το σύστημα καθώς και οι λειτουργίες του. Παρουσιάστηκαν οι μετρήσεις του καθώς έγινε και η ανάλυση SWOT. Παράλληλα, επισημάνθηκαν ορισμένα συμπεράσματα που προέκυψαν από την παρούσα υλοποίηση. Στη συνέχεια, αναφέρθηκαν τα προβλήματα που προέκυψαν κατά την εκπόνηση της παρούσας διπλωματική εργασίας, ενώ τέλος δόθηκε μια λίστα με πιθανές μελλοντικές επεκτάσεις του συστήματος.

## Βιβλιογραφία

- [1] Αντίστοιχη εφαρμογή 1: ESP32-Cam RC Car with Camera and Mobile Phone Controller  
Ιστότοπος: <https://robotzero.one/esp32-cam-rc-car/>  
Αύγουστος, 2020
- [2] Αντίστοιχη εφαρμογή 2: Remote Wi-Fi video Car  
Ιστότοπος: <https://hackaday.io/project/30400-remote-wifi-video-car>  
Αύγουστος, 2020
- [3] Αντίστοιχη εφαρμογή 3: Wi-Fi RC Car  
Ιστότοπος: <https://www.instructables.com/id/DIY-WIFI-RC-Car-With-ESP8266-and-Arduino-IDE/>  
Αύγουστος, 2020
- [4] Αντίστοιχη εφαρμογή 4: ESP-32-Cam FPV Arduino Wi-Fi Control Tank  
Ιστότοπος: <https://www.instructables.com/id/ESP32-CAM-FPV-Arduino-Wifi-Control-Tank-With-WebAp/>  
Αύγουστος, 2020
- [5] HyperText Markup Language (HTML)  
Ιστότοπος: [https://www.w3schools.com/html/html\\_intro.asp](https://www.w3schools.com/html/html_intro.asp)  
Αύγουστος, 2020
- [6] Cascading Style Sheets (CSS)  
Ιστότοπος: [https://www.w3schools.com/css/css\\_intro.asp](https://www.w3schools.com/css/css_intro.asp)  
Αύγουστος, 2020
- [7] Javascript (JS)  
Ιστότοπος: <https://www.codecademy.com/catalog/language/javascript>  
Αύγουστος, 2020
- [8] jQuery  
Ιστότοπος: <https://www.codecademy.com/catalog/language/javascript>  
Αύγουστος, 2020
- [9] Πρωτόκολλο δεδομένων χρήστη UDP  
Ιστότοπος: <https://www.cloudflare.com/learning/ddos/glossary/user-datagram-protocol-udp/>  
Αύγουστος, 2020
- [10] Microsoft Visual Studio  
Ιστότοπος: [https://en.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://en.wikipedia.org/wiki/Microsoft_Visual_Studio)

Αύγουστος, 2020

- [11] Visual C# και .NET πλαίσιο

Ιστότοπος: <https://www.capitaitresourcing.co.uk/blogs/whats-the-difference-between-c-and-net-81122210121>

Αύγουστος, 2020

- [12] Kicad

Ιστότοπος: <https://kicad-pcb.org/about/kicad/>

Αύγουστος, 2020

- [13] AForge.NET

Ιστότοπος: <http://www.aforgenet.com/framework/>

Αύγουστος, 2020

- [14] ESP32-Cam

Ιστότοπος: [https://www.espressif.com/en/news/ESP32\\_CAM](https://www.espressif.com/en/news/ESP32_CAM)

Αύγουστος, 2020

- [15] LM2596s

Ιστότοπος: <https://www.ti.com/lit/ds/symlink>

Αύγουστος, 2020

- [16] Arduino IDE

Ιστότοπος: <https://www.arduino.cc/en/main/software>

Αύγουστος, 2020

- [17] Internet of Things  
Ιστότοπος: <https://ieeexplore.ieee.org/abstract/document/6714496>  
Αύγουστος, 2020
- [18] Βιβλιοθήκη WiFi.h  
Ιστότοπος: <https://www.arduino.cc/en/Reference/WiFi>  
Αύγουστος, 2020
- [19] Βιβλιοθήκη WiFiUDP.h  
Ιστότοπος: [https://arduino-esp8266.readthedocs.io/en/latest/esp8266wifi/udp\\_examples.html](https://arduino-esp8266.readthedocs.io/en/latest/esp8266wifi/udp_examples.html)  
Αύγουστος, 2020
- [20] Συμπίεση Gzip  
Ιστότοπος: <https://www.imperva.com/learn/performance/gzip/>  
Αύγουστος, 2020
- [21] Συμβάν onkeydown  
Ιστότοπος: [https://www.w3schools.com/jsref/event\\_onmousedown.asp](https://www.w3schools.com/jsref/event_onmousedown.asp)  
Αύγουστος, 2020
- [22] Συμβάν onkeyup  
Ιστότοπος: [https://www.w3schools.com/JSREF/event\\_onmouseup.asp](https://www.w3schools.com/JSREF/event_onmouseup.asp)  
Αύγουστος, 2020
- [23] Συμβάν onkeydown  
Ιστοτοπός: [https://www.w3schools.com/jsref/event\\_onkeydown.asp](https://www.w3schools.com/jsref/event_onkeydown.asp)  
Αύγουστος, 2020

- [24] Συμβάν onkeyup  
Ιστότοπος: [https://www.w3schools.com/JSREF/event\\_onkeyup.asp](https://www.w3schools.com/JSREF/event_onkeyup.asp)  
Αύγουστος, 2020
- [25] Συνάρτηση ParsePacket  
Ιστότοπος: <https://www.arduino.cc/en/Reference/EthernetUDPParsePacket>  
Αύγουστος, 2020
- [26] Συνάρτηση Read  
Ιστότοπος: <https://www.arduino.cc/en/Reference/EthernetUDPRead>  
Αύγουστος, 2020
- [27] Συνάρτηση Strcmp  
Πηγή: Νίκος Χατζηγιαννάκης «Η γλώσσα C σε Βάθος» Εκδόσεις Κλειδάριθμος  
Αύγουστος, 2020
- [28] Συνάρτηση Memset  
Πηγή: Νίκος Χατζηγιαννάκης «Η γλώσσα C σε Βάθος» Εκδόσεις Κλειδάριθμος  
Αύγουστος, 2020
- [29] Βιβλιοθήκη Esp\_http\_server  
Ιστότοπος: [https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/protocols/esp\\_http\\_server.html](https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/protocols/esp_http_server.html)  
Αύγουστος, 2020
- [30] MIT App Inventor  
Ιστότοπος: <https://appinventor.mit.edu/about-us>  
Αύγουστος, 2020

[31] NotePad ++

Ιστότοπος: <https://notepad-plus-plus.org/>

Αύγουστος, 2020



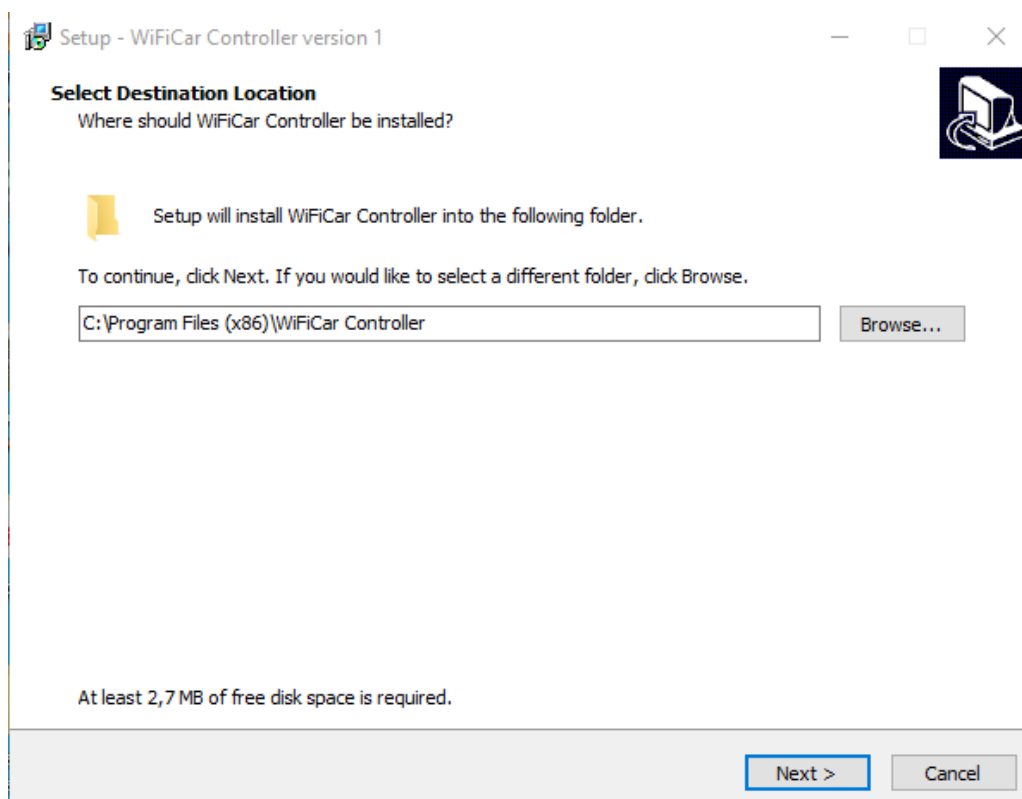


## Παράρτημα Α – Οδηγίες Εγκατάστασης Εφαρμογής Windows

Στο παράρτημα αυτό, παρουσιάζεται η διαδικασία εγκατάστασης της Windows εφαρμογής που αναπτύχθηκε, για να γίνει δυνατή η χρήση της.

Αρχικά, ο χρήστης εκτελεί το αρχείο installer.exe και στην συνέχεια επιλέγει το κουμπί «επόμενο». Έπειτα, περιμένει μέχρι να ολοκληρωθεί η εγκατάσταση και συνέχεια επιλέγει το κουμπί «Τέλος». Η εγκατάσταση πλέον έχει ολοκληρωθεί, και ο χρήστης μπορεί να εκτελέσει το WiFiCar Controller.exe για τον έλεγχο του οχήματος.

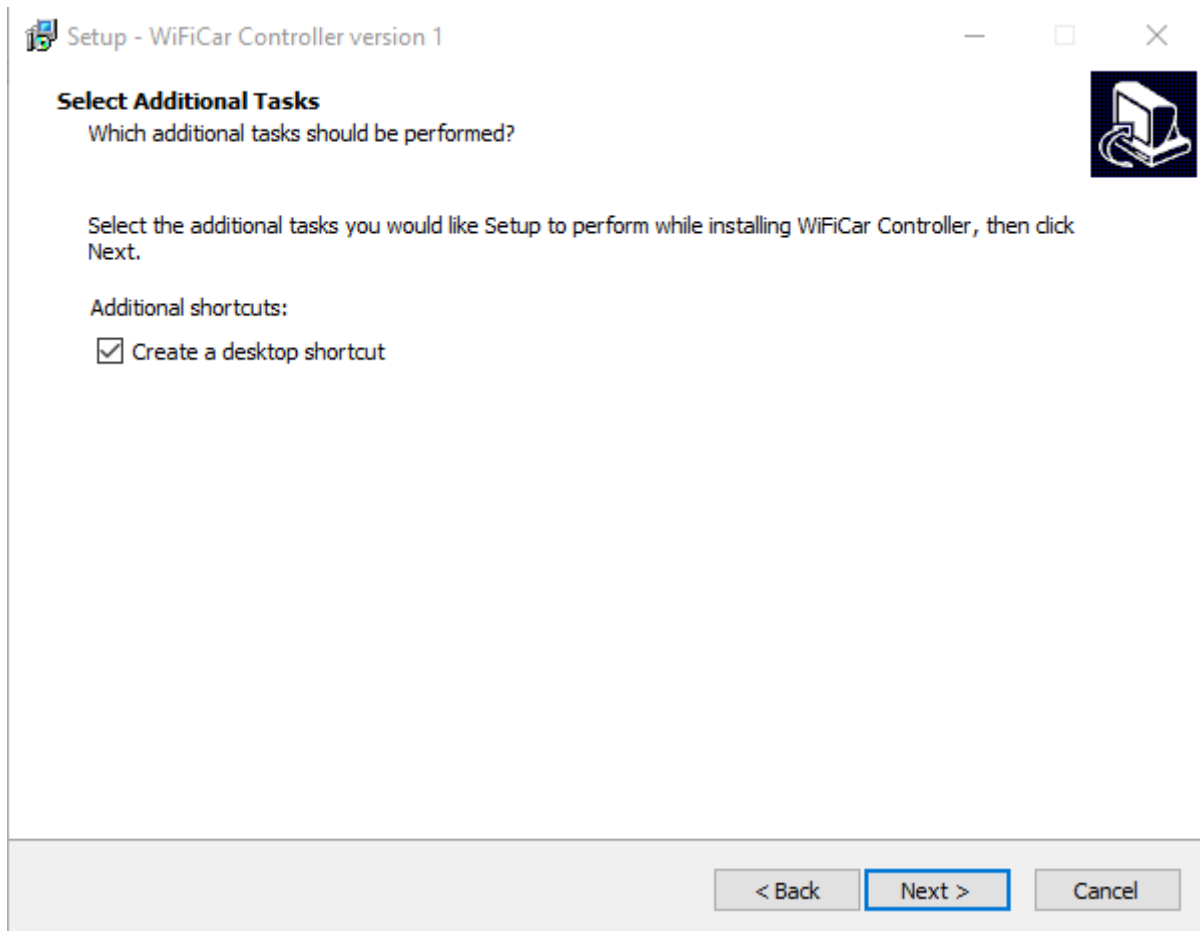
Αναλυτικότερα, παρουσιάζεται στις παρακάτω εικόνες (**Εικόνα 50**, **Εικόνα 51**, **Εικόνα 52**, **Εικόνα 49**) η διαδικασία της εγκατάστασης.



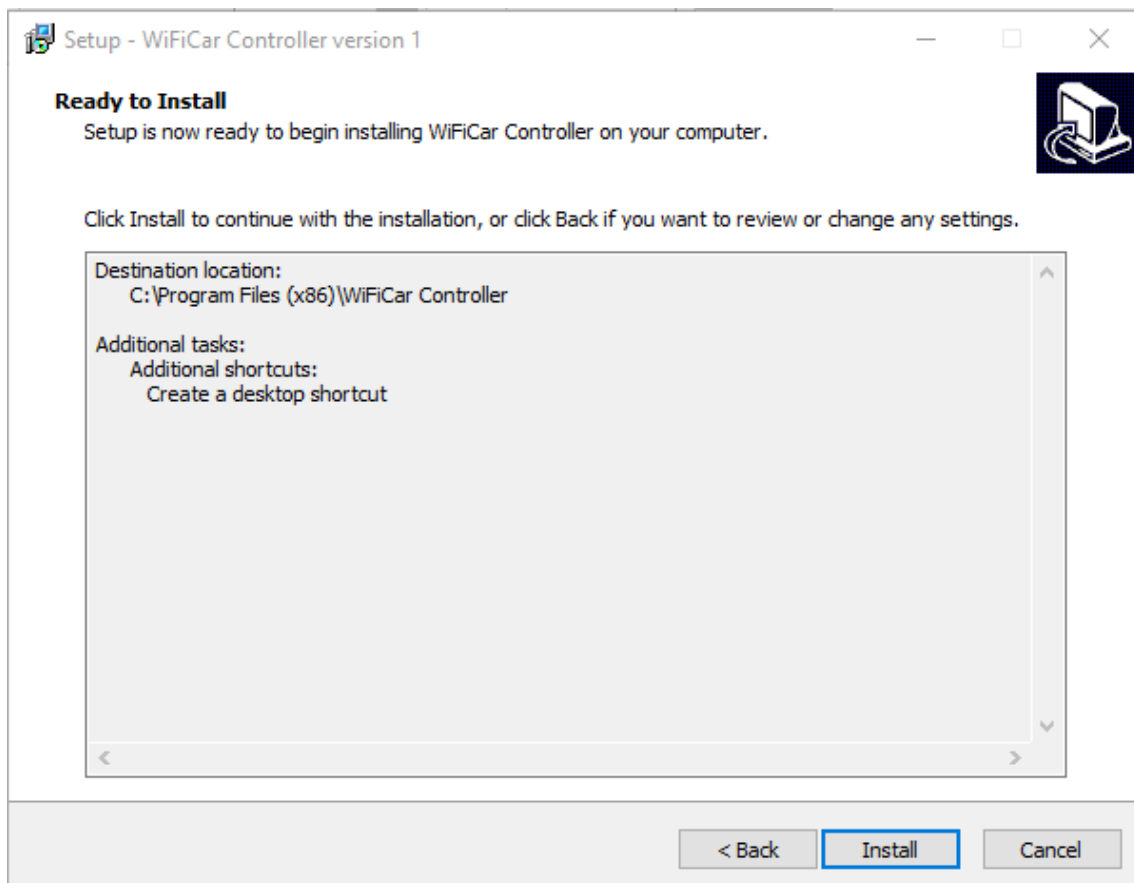
Εικόνα 49: Ενδεικτική Εικόνα Εγκατάστασης Εφαρμογής

Στην **Εικόνα 49** επιλέγουμε την τοποθεσία του φακέλου WiFiCar Controller στον υπολογιστή μας και στην συνέχεια επιλέγουμε το κουμπί Next.

Έπειτα, **Εικόνα 50** μπορούμε να επιλέξουμε αν θέλουμε να δημιουργηθεί συντόμευση στην επιφάνεια εργασίας του υπολογιστή μας.



Εικόνα 50: Ενδεικτική Εικόνα Εγκατάστασης Εφαρμογής



Εικόνα 51: Ενδεικτική Εικόνα Εγκατάστασης Εφαρμογής

Επιλέγουμε το κουμπί Install (**Εικόνα 51**) και περιμένουμε μέχρι να ολοκληρωθεί η εγκατάσταση. Έπειτα, επιλέγουμε το κουμπί Finish (**Εικόνα 52**) και τότε η εφαρμογή εγκαταστάθηκε επιτυχώς.



Εικόνα 52: Ενδεικτική Εικόνα Εγκατάστασης Εφαρμογής

## Παράρτημα Β – Οδηγίες Εγκατάστασης Εφαρμογής Android

Στο παράρτημα αυτό παρουσιάζεται η διαδικασία εγκατάστασης της Εφαρμογής Android που αναπτύχθηκε, για να γίνει δυνατή η χρήση της.

Αρχικά, ο χρήστης πρέπει να έχει μία έξυπνη κινητή συσκευή με λογισμικό Android τουλάχιστον στην έκδοση 4.2. Στην συνέχεια πρέπει να κατεβάσει και να εγκαταστήσει το apk (Android Application Package) αρχείο της εφαρμογής. Πλέον η εφαρμογή είναι εγκατεστημένη στη συσκευή. Με το πάτημα του εικονιδίου η εφαρμογή μπορεί να αρχίσει τη λειτουργία της.