

Ατομική Διπλωματική Εργασία

**ΥΛΟΠΟΙΗΣΗ ΚΑΙ ΑΞΙΟΛΟΓΗΣΗ ΣΥΣΤΑΤΙΚΩΝ ΣΤΟΙΧΕΙΩΝ
ΣΥΣΤΗΜΑΤΟΣ ΔΙΑΧΕΙΡΙΣΗΣ ΚΙΝΔΥΝΩΝ ΜΕ ΧΡΗΣΗ
ΥΠΟΛΟΓΙΣΤΙΚΗΣ ΟΡΑΣΗΣ**

Αθηνά Χατζηχριστοδούλου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2022

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Υλοποίηση και Αξιολόγηση Συστατικών Στοιχείων Συστήματος Διαχείρισης Κινδύνων με Χρήση Υπολογιστικής Όρασης

Αθηνά Χατζηχριστοδούλου

Επιβλέπων Καθηγητής

Δημήτρης Ζεϊναλιπούρ

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των
απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του
Πανεπιστημίου Κύπρου

Μάιος 2022

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή Δρ. Δημήτρη Ζεϊναλιπούρ για την συνεχή καθοδήγηση, υποστήριξη και ανεκτίμητη βοήθεια που μου παρείχε καθ' όλη τη διάρκεια της έρευνας και συγγραφής της παρούσας διπλωματικής εργασίας.

Επιπρόσθετα, θα ήθελα να ευχαριστήσω δύο πολύ σημαντικούς ερευνητές του εργαστηρίου DMSL, τον κ. Πασχάλη Μπέη και τον κ. Ιωάννη Ιωαννίδη. Η συνεργασία μας τον τελευταίο χρόνο ήταν άψογη και οι γνώσεις τους πολύτιμες για την επιτυχή ολοκλήρωση της εργασίας αυτής.

Αναγνώριση

Η επίδειξη του συστήματος AnyplaceCV που περιγράφεται σε αυτήν τη διπλωματική εργασία θα παρουσιαστεί στο πιο κάτω συνέδριο:

- AnyplaceCV: Infrastructure-less Localization in Anyplace with Computer Vision, Paschalis Mpeis, Athina Hadjichristodoulou, Ioannis Ioannidis and Demetrios Zeinalipour-Yazti, The 23rd IEEE International Conference on Mobile Data Management, Paphos, Cyprus (online), IEEE Computer Society, June 6 – 9, 2022.

Περίληψη

Τα έξυπνα κινητά τηλέφωνα, που χαρακτηρίζονται ως μικροί υπολογιστές αφού παρέχουν ένα μεγάλο εύρος από λειτουργίες, είναι από τις πιο απαραίτητες συσκευές που μπορεί να έχει ένα άτομο. Μέσω των συσκευών αυτών παρέχονται οι δυνατότητες αποστολής μηνυμάτων, λήψης φωτογραφιών, δημιουργίας κλήσεων, κ.α. Ένας συνδυασμός παροχών που τις κάνουν αναντικατάστατες. Είναι λοιπόν λογική η επιθυμία εγκατάστασης περισσότερων λειτουργιών σε αυτές, έτσι ώστε ο χρήστης να διευκολυνθεί ακόμη περισσότερο στην καθημερινότητα του.

Η έγκαιρη ενημέρωση των ειδικών σε περιπτώσεις έκτακτης ανάγκης, όπως οι εκρήξεις πυρκαγιών, είναι ο πιο αποτελεσματικός τρόπος για την γρήγορη αντιμετώπιση τους και τον περιορισμό της εξάπλωσης τους. Λαμβάνοντας υπόψη ότι τα κινητά τηλέφωνα είναι πλέον οι πιο δημοφιλείς συσκευές και την παραπάνω ανάγκη για έκτακτη ενημέρωση, έγινε ο σχεδιασμός και η ανάπτυξη της εφαρμογής SMAS για κινητά τηλέφωνα με λειτουργικό Android.

Η εφαρμογή SMAS αναπτύχθηκε με σκοπό την ταχεία αντιμετώπιση των πυρκαγιών σε πλοία τύπου roll on/ roll off. Στόχος είναι η γρήγορη εύρεση της περιοχής που ξέσπασε η πυρκαγιά για να μπορέσουν οι πυροσβέστες και το κατάλληλο προσωπικό να πάνε στο σημείο αυτό. Με την γνώση ότι τα πλοία δεν διαθέτουν τον κατάλληλο υπόβαθρο για τον εντοπισμό εσωτερικής τοποθεσίας με την χρήση σημάτων (όπως Wi-Fi, BLE, UWB), αναπτύχθηκε από ερευνητές του εργαστηρίου DMSL του Πανεπιστημίου Κύπρου, ένας νέος αλγόριθμος που αξιοποιεί τις δυνατότητες της υπολογιστικής όρασης και της μηχανικής μάθησης. Η γεωτοποθέτηση γίνεται με την αντιστοίχιση των φυσικών αντικειμένων που αναγνωρίζονται σε ένα χώρο, το πλήθος τους και τις σχετικές τους θέσεις, βάσει ήδη υπαρχόντων χαρτογραφήσεων.

Η παρούσα διπλωματική εργασία στόχο είχε την προσθήκη επιπλέον λειτουργιών στο σύστημα SMAS, όπως ένα κανάλι επικοινωνίας για τους χρήστες του βάσει τοποθεσίας, αλλά και ένα εργαλείο αναζήτησης χώρων του πλοίου. Επιπλέον, ασχολούμαστε με την φάση της εκπαίδευσης του αλγόριθμου εντοπισμού εσωτερικής τοποθεσίας καθώς και με μεθόδους βελτίωσης των αποτελεσμάτων του, όπως το IMU και το Map Matching.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	1
	1.1 Υποκίνηση Εργασίας	1
	1.2 Περίγραμμα Εργασίας	3
	1.3 Συνεισφορές	4
Κεφάλαιο 2	Σχετική Δουλειά και Υπόβαθρο.....	5
	2.1 Σχετικά Συστήματα Ανταλλαγής Μηνυμάτων	5
	2.1.1 Συστήματα Ανταλλαγής Μηνυμάτων Βάσει Τοποθεσίας	6
	2.1.2 Συστήματα Ανταλλαγής Εφήμερων Μηνυμάτων	8
	2.2 Σχετικά Συστήματα Εντοπισμού Εσωτερικής Τοποθεσίας	10
	2.3 Σχετικά Έξυπνα Συστήματα Διαχείρισης Κινδύνων	14
	2.4 Τεχνολογικό Υπόβαθρο	16
	2.4.1 Τεχνολογίες Backend	16
	2.4.2 Τεχνολογίες Fronted	18
	2.4.3 Τεχνολογίες Ελέγχου Έκδοσης Λογισμικού	19
Κεφάλαιο 3	Ανάλυση Συστημάτων Anyplace και SMAS.....	20
	3.1 Ανάγκη για Υπολογισμό Εσωτερικής Τοποθεσίας	20
	3.2 Ανατομία του Συστήματος Anyplace	21
	3.2.1 Anyplace Logger	22
	3.2.2 Anyplace Navigator	24
	3.2.3 Anyplace Architect	24
	3.2.4 Anyplace Viewer	25
	3.3 Περιβάλλοντα Χωρίς Υποδομές	25
	3.4 Εφαρμογή SMAS	26
	3.4.1 Anyplace και SMAS	27
	3.4.2 Τεχνολογίες στο Σύστημα SMAS	28

Κεφάλαιο 4	Υποσύστημα Ανταλλαγής Μηνυμάτων Βάσει Τοποθεσίας και Εργαλείο Αναζήτησης.....	30
4.1	Ιδέα	30
4.2	Διεπαφή Χρήστη για το Υποσύστημα Ανταλλαγής Μηνυμάτων	32
4.2.1	Φιλοσοφία Σχεδιασμού	32
4.2.2	Αναγνώριση Ομιλίας	35
4.2.3	Προσθήκη Φωτογραφιών	35
4.2.4	Αποστολή Τοποθεσίας και Ειδοποιήσεων	36
4.3	Εργαλείο Αναζήτησης	37
Κεφάλαιο 5	Υπολογισμός Εσωτερικής Τοποθεσίας με Χρήση Υπολογιστικής Όρασης.....	39
5.1	Επισκόπηση Αλγόριθμου	39
5.2	Φάση Εκπαίδευσης	40
5.2.1	Μοντέλο LASHCO	42
5.3	Φάση Logging	44
5.4	Φάση Εντοπισμού Εσωτερικής Τοποθεσίας	45
5.5	Παραδείγματα Χρήσης	46
Κεφάλαιο 6	Υποσύστημα IMU και Αλγόριθμος Map Matching.....	48
6.1	Στόχος	48
6.2	Αδρανειακά Συστήματα Μέτρησης	49
6.3	Προηγούμενη Υλοποίηση	50
6.4	Αξιολόγηση των Αισθητήρων	50
6.5	Υλοποίηση IMU με τη Χρήση Αισθητήρων και Dead Reckoning	55
6.6	Αλγόριθμος Map Matching	56
Κεφάλαιο 7	Πειραματική Αξιολόγηση.....	58
7.1	Αξιολόγηση Μετρήσεων στη Φάση Εκπαίδευσης	58
7.2	Αξιολόγηση Υποσυστήματος Ανταλλαγής Μηνυμάτων	61
7.3	Αξιολόγηση Υποσυστήματος IMU	63

Κεφάλαιο 8 Συμπεράσματα.....	65
8.1 Συμπεράσματα	65
8.2 Μελλοντική Δουλειά	66
 Βιβλιογραφία	 68

Κεφάλαιο 1

Εισαγωγή

1.1 Υποκίνηση Εργασίας	1
1.2 Περίγραμμα Εργασίας	3
1.3 Συνεισφορές	4

1.1 Υποκίνηση Εργασίας

Εν έτη 2022, το ποσοστό των συνδρομών με έξυπνα κινητά τηλέφωνα παγκοσμίως είναι στα 82% του συνολικού πληθυσμού και προβλέπεται ότι ο αριθμός αυτός θα φτάσει τα 96% μέσα στα επόμενα 5 χρόνια. Με την συνεχή αύξηση των χρηστών των κινητών τηλεφώνων που παρατηρούμε τα τελευταία χρόνια, οι απαιτήσεις από αυτά έχουν αλλάξει. Δημιουργούνται νέες ανάγκες για συστήματα που μπορούν να αξιοποιηθούν έτσι ώστε να επιλύσουν καθημερινά προβλήματα, να φανούν χρήσιμα για την διεκπεραίωση μιας πολύπλοκης διεργασίας ή να παρέχουν ένα τρόπο ψυχαγωγίας.

Οι εφαρμογές οι οποίες θεωρούνται από τις πιο χρήσιμες και έχουν πλέον ενσωματωθεί στην καθημερινότητα μας είναι αυτές που δίνουν την δυνατότητα χαρτογράφησης δρόμων, πόλεων, χωρών και εύρεσης της θέσης ενός κινούμενου (ή και μη) σώματος πάνω σε αυτά. Η πλοήγηση μεταξύ δύο σημείων είναι πλέον τόσο εύκολη όσο το πάτημα μερικών κουμπιών και η ακολουθία των οδηγιών που δίνονται από τον ψηφιακό οδηγό της εφαρμογής. Τέτοια συστήματα χρησιμοποιούν το Παγκόσμιο Σύστημα Στιγματοθέτησης, ή αλλιώς GPS όπως είναι γνωστό με το ακρώνυμο του.

Το σύστημα GPS κάνει χρήση μιας ομάδας από δορυφόρους που βρίσκονται σε τροχιά γύρω από την Γη, οι οποίοι εκπέμπουν σήματα. Τα σήματα αυτά είναι αρκετά αποτελεσματικά για την εύρεση τοποθεσίας σε εξωτερικούς χώρους, καθώς τα εμπόδια

που συναντούν είναι ελάχιστα. Όμως, η ένταση των σημάτων αυτών δεν είναι καθόλου ικανοποιητική για την εύρεση τοποθεσίας σε εσωτερικούς χρόνους. Η ένταση των σημάτων εξασθενεί καθώς περνάνε μέσα από πολλά επίπεδα τοίχων, μετάλλων και άλλων υλικών που απαρτίζουν κτήρια και υποδομές. Αυτό μειώνει την ακρίβεια της υπολογιζόμενης θέσης αισθητά.

Η ανάγκη για εύρεση της εσωτερικής τοποθεσίας δημιουργήθηκε με την ανάγκη των ανθρώπων να περιφέρονται μέσα σε κτήρια μεγάλης κλίμακας με ευκολία, χωρίς να αποπροσανατολίζονται. Στα σενάρια όπου ένας ταξιδιώτης βρίσκεται σε ένα αεροδρόμιο που δεν έχει ξαναβρεθεί, ένας μαθητής που φθάνει σε μια πανεπιστημιούπολη για πρώτη φορά και ένας τουρίστας που θέλει να επισκεφθεί το εμπορικό κέντρο μιας άλλης χώρας, ένα τέτοιο σύστημα είναι ανεκτίμητο. Για τον λόγο αυτό, η ερευνητική κοινότητα ψάχνει διάφορες μεθοδολογίες για τον εντοπισμό της τοποθεσίας σε κλειστούς χώρους, με την πιο δημοφιλή να είναι η χρήση Wi-Fi σημάτων.

Ο τρόπος που γίνεται η γεωτοποθέτηση με την χρήση των σημάτων Wi-Fi είναι εύκολα κατανοήσιμος. Συγκρίνονται οι εντάσεις των σημάτων που λαμβάνει η συσκευή του χρήστη την συγκεκριμένη στιγμή όπου η γεωτοποθέτηση είναι επιθυμητή, με ένα «χάρτη» από προηγούμενες μετρήσεις που λήφθηκαν από το ίδιο κτήριο και το πιο κοντινό ταίριασμα είναι η υπολογισμένη θέση του χρήστη. Η μέθοδος όμως αυτή δεν είναι εφαρμόσιμη σε χώρους όπου η κάλυψη των Wi-Fi σημάτων δεν είναι τόσο μεγάλη ώστε να μπορέσει να δημιουργηθεί ο χάρτης.

Στα πλαίσια του προγράμματος LASH FIRE αναπτύχθηκε η εφαρμογή SMAS (Smart Alert System). Μια εφαρμογή που θα χρησιμοποιείται από το πλήρωμα των πλοίων για την γρήγορη αντιμετώπιση κινδύνων και εύρεση τοποθεσίας μέσα σε αυτά. Μέσα στα πλοία, τα σήματα Wi-Fi δεν είναι ευρέως διαθέσιμα κάνοντας την μέθοδο που τα χρησιμοποιεί αναξιόπιστη. Λόγω αυτού, έγιναν προσπάθειες για την υλοποίηση και ανάπτυξη ενός συστήματος εύρεσης της τοποθεσίας με διαφορετική μέθοδο, με χρήση υπολογιστικής όρασης και εκπαιδευμένων μοντέλων που μπορούν να αναγνωρίζουν αντικείμενα. Ο τρόπος που λειτουργεί είναι παρόμοιος με τον τρόπο που αναφέραμε προηγουμένως για τα σήματα Wi-Fi. Αντί όμως σημάτων, πλέον το αναγνωριστικό ενός χώρου είναι τα αντικείμενα που βρίσκονται σε αυτόν και η σχετική τους θέση.

Το σύστημα αυτό μπορεί να χρησιμοποιηθεί με επιτυχία στα πλοία αλλά και σε άλλους χώρους όπου έχουν το ίδιο πρόβλημα. Δίνοντας έμφαση στον στόχο του προγράμματος LASH FIRE, εκτός από εντοπισμό της θέσης τους μέσα στο πλοίο, οι χρήστες είναι απαραίτητο να ενημερώνονται έγκαιρα για τυχόν κινδύνους, για παράδειγμα σε περιπτώσεις εκρήξεων πυρκαγιών. Μέσω του συστήματος SMAS, τους παρέχουμε ένα κανάλι επικοινωνίας το οποίο είναι ρυθμιζόμενο έτσι ώστε να στέλνει τα μηνύματα μόνο στο επιθυμητό κοινό, κάνοντας χρήση της τοποθεσίας τους. Τα μηνύματα είναι εφήμερα, δηλαδή διαγράφονται μετά από κάποιο χρονικό πλαίσιο για να κάνουν «χώρο» στα επόμενα. Επίσης, παρέχουμε στους χρήστες την δυνατότητα αναζήτησης σημείων στο πλοίο και οδηγίες για την πλοήγηση τους μέχρι αυτά.

1.2 Περίγραμμα Εργασίας

Η εργασία αυτή αποτελείται από 8 κεφάλαια των οποίων οι περιγραφές έχουν ως ακολούθως. Στο Κεφάλαιο 2 γίνονται αναφορές σε συστήματα και ερευνητικές δουλειές που είναι άμεσα σχετικές με τα υποσυστήματα που απαρτίζουν την εφαρμογή SMAS. Δηλαδή, τα υποσυστήματα ανταλλαγής μηνυμάτων και εσωτερικής γεωτοποθέτησης και η λειτουργία αποστολής ενημερώσεων για έγκαιρη διαχείριση κινδύνων. Αναφέρονται επίσης οι τεχνολογίες που χρησιμοποιήθηκαν κατά την διάρκεια ανάπτυξης των υποσυστημάτων αυτών.

Στο Κεφάλαιο 3 αναλύεται το οικοσύστημα Anyplace, του οποίου η αρχιτεκτονική ήταν η βάση στήριξης για την ανάπτυξη του υποσυστήματος γεωτοποθέτησης. Περιγράφεται η αρχιτεκτονική του και τα συστατικά του στοιχεία. Ακολούθως, δίνεται το περίγραμμα της εφαρμογής SMAS και εξηγείται ο σκοπός του προγράμματος LASH FIRE.

Στο Κεφάλαιο 4 εξηγείται η ιδέα πίσω από την υλοποίηση ενός καναλιού επικοινωνίας βάσει τοποθεσίας και δίνονται περισσότερες λεπτομέρειες για την γραφική διεπαφή και τις λειτουργίες της. Επίσης, γίνεται αναφορά στο εργαλείο αναζήτησης που παρέχεται στους χρήστες της εφαρμογής.

Στο Κεφάλαιο 5 περιγράφονται οι φάσεις του αλγόριθμου υπολογισμού εσωτερικής τοποθεσίας με τη χρήση υπολογιστικής όρασης και εκπαιδευμένων μοντέλων και δίνονται κάποια παραδείγματα χρήσης του στους χώρους του Πανεπιστημίου Κύπρου.

Στο Κεφάλαιο 6 γίνεται αναφορά στο υποσύστημα IMU, που στόχος του είναι η βελτιστοποίηση των αποτελεσμάτων του προαναφερθέν αλγόριθμου, καθώς και στη μέθοδο Map Matching.

Στο Κεφάλαιο 7 παρουσιάζονται τα πειράματα που έγιναν σε διάφορες πτυχές της εφαρμογής SMAS για λόγους αξιολόγησης της εγκυρότητας και αποδοτικότητας της.

Τέλος, στο Κεφάλαιο 8 αναφέρονται τα συμπεράσματα που εξάχθηκαν μέσω της έρευνας που διεξάχθηκε για την παρούσα διπλωματική. Γίνεται αναφορά και σε μερικά στοιχεία του συστήματος SMAS που μπορούν να επεκταθούν σε μεταγενέστερο στάδιο για την περαιτέρω βελτίωση του.

1.3 Συνεισφορές

Μέσω της παρούσας διπλωματικής εργασίας έγιναν συνεισφορές στις τέσσερις ακόλουθες περιοχές του συστήματος SMAS:

- Στο υποσύστημα ανταλλαγής μηνυμάτων, όπου αναπτύχθηκε η γραφική διεπαφή χρήστη και οι ενσωματωμένες λειτουργίες του (frontend),
- Στο εργαλείο αναζήτησης σταθερών και κινούμενων στόχων,
- Στο στάδιο εκπαίδευσης του αλγόριθμου που χρησιμοποιείται για αναγνώριση αντικειμένων και
- Στη μέθοδο βελτίωσης του υπολογισμού της εσωτερικής τοποθεσίας με τη χρήση IMU.

Κεφάλαιο 2

Σχετική Δουλειά και Υπόβαθρο

2.1	Σχετικά Συστήματα Ανταλλαγής Μηνυμάτων	5
2.1.1	Συστήματα Ανταλλαγής Μηνυμάτων Βάσει Τοποθεσίας	6
2.1.2	Συστήματα Ανταλλαγής Εφήμερων Μηνυμάτων	8
2.2	Σχετικά Συστήματα Εντοπισμού Εσωτερικής Τοποθεσίας	10
2.3	Σχετικά Έξυπνα Συστήματα Διαχείρισης Κινδύνων	14
2.4	Τεχνολογικό Υπόβαθρο	16
2.4.1	Τεχνολογίες Backend	16
2.4.2	Τεχνολογίες Frontend	18
2.4.3	Τεχνολογίες Ελέγχου Έκδοσης Λογισμικού	19

Σε αυτό το κεφάλαιο γίνεται η μελέτη και ανάλυση ήδη υπάρχοντων συστημάτων και ερευνών που έχουν άμεση σχέση με το πεδίο ενασχόλησης της παρούσας διπλωματικής εργασίας. Το πρώτο υποκεφάλαιο ασχολείται με σχετικά συστήματα υποστήριξης συνομιλιών μεταξύ δύο ή περισσότερων ατόμων, το δεύτερο με συστήματα με την δυνατότητα υπολογισμού εσωτερικής τοποθεσίας και το τρίτο με εφαρμογές για άμεση διαχείριση κινδύνων. Ακολούθως, γίνονται σύντομες αναφορές στις τεχνολογίες που χρησιμοποιήθηκαν κατά την διάρκεια ανάπτυξης των συστατικών στοιχείων και προγραμμάτων που απαρτίζουν αυτή την διπλωματική.

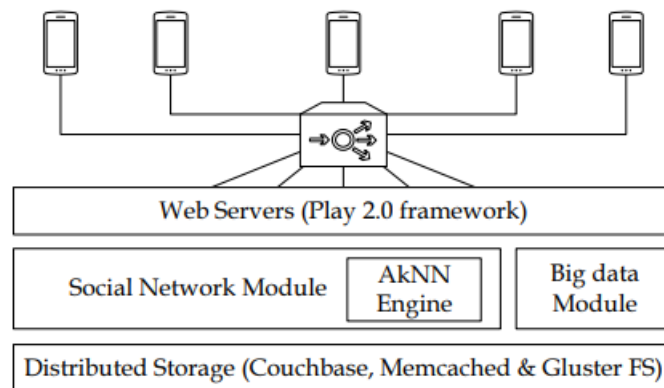
2.1 Σχετικά Συστήματα Ανταλλαγής Μηνυμάτων

Το υποσύστημα ανταλλαγής μηνυμάτων που αναπτύχθηκε έχει ως κύριο χαρακτηριστικό του την διεξαγωγή συνομιλιών βάσει τοποθεσίας των χρηστών. Εκτός αυτού, άλλα σημαντικά στοιχεία του είναι τα εφήμερα μηνύματα και η δυνατότητα αποστολής alerts

στους πιο κοντινούς ανταποκριτές για γρήγορη διαχείριση κινδύνων. Πιο κάτω, γίνονται αναφορές σε συστήματα που έχουν παρόμοιες ιδιότητες για σκοπούς σύγκρισης αλλά και κατανόησης της ανάγκης ενός συστήματος που συνδυάζει όλα τα παραπάνω χαρακτηριστικά.

2.1.1 Συστήματα Ανταλλαγής Μηνυμάτων Βάσει Τοποθεσίας

Η εφαρμογή κοινωνικής δικτύωσης Yik Yak [38] κυκλοφόρησε το 2013, ιδρυόμενη από τους Tyler Droll και Brooks Buffington. Είναι από τις πρώτες πλατφόρμες του είδους της που επιτρέπουν την ανώνυμη δημοσίευση σύντομων κειμένων σε μια τοπική κοινότητα, και πιο συγκεκριμένα, σε άτομα που βρίσκονται στη μέγιστη απόσταση των 8 χιλιομέτρων [4]. Η εφαρμογή έχει σαν στόχο φοιτητές νεαρής ηλικίας, αφού επί το πλείστον χρησιμοποιείται σε πανεπιστημιούπολεις, δίνοντας τους έτσι την δυνατότητα να εκφράζουν ανώνυμα τις σκέψεις τους σε ένα περιβάλλον όπου μόνο συμφοιτητές τους μπορούν να αλληλεπιδράσουν. Η εφαρμογή έλαβε αρκετά σκληρή κριτική την περίοδο του 2015-2016, με την αιτιολόγηση ότι παρότρυνε τον εκφοβισμό, αφού έδινε στους χρήστες της την δυνατότητα ανωνυμίας [23]. Το 2017 η Yik Yak ανακοίνωσε τον τερματισμό της υπηρεσίας της, με την επαναλειτουργία της να ανακοινώνεται το 2021.

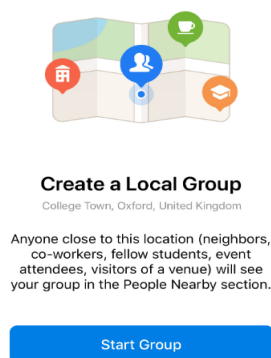


Σχήμα 2.1 : Αρχιτεκτονική Συστήματος Rayzit [11]

Ακόμη μία αξιοσημείωτη υλοποίηση ενός παρόμοιου συστήματος είναι η εφαρμογή Rayzit [32], η οποία αναπτύχθηκε στο Data Management Systems Laboratory (DMSL) του Πανεπιστημίου Κύπρου από ερευνητές και φοιτητές. Το Rayzit είναι μια crowd messaging αρχιτεκτονική, που δίνει την δυνατότητα ανταλλαγής μηνυμάτων με τους

κοντινότερους χρήστες της, κρατώντας τον αποστολέα κάθε μηνύματος ανώνυμο. Χρησιμοποιείται η μέθοδος των K κοντινότερων γειτόνων (K Nearest Neighbors), όπου είναι οι K χρήστες της εφαρμογής με τα πιο παρόμοια χαρακτηριστικά με τον κεντρικό κόμβο. Ανά τακτά χρονικά διαστήματα, παράγεται δυναμικά ένας καινούριος, παγκόσμιος γράφος συνδέσεων λόγω της συνεχούς αλλαγής τοποθεσίας. Το Rayzit βρίσκει εφαρμογή σε προχωρημένες υπηρεσίες που απαιτούν την επίγνωση τοποθεσίας, όπως την αγοραπωλησία προϊόντων, την επικοινωνία κοινού σε κοινωνικές εκδηλώσεις και στην ενημέρωση για φυσικές (και μη) καταστροφές [11].

Η υπηρεσία ανταλλαγής άμεσων μηνυμάτων Telegram [36] εμφανίστηκε το 2013 για εφαρμογές Android και iOS. Βασίζεται στο cloud και πλέον μπορεί να εγκατασταθεί σε πολλαπλές πλατφόρμες. Εκτός από αποστολή μηνυμάτων, υποστηρίζει κρυπτογραφημένες βιντεοκλήσεις, κοινή χρήση αρχείων κ.α. Με την ενημέρωση της εφαρμογής που έγινε τον Ιούνιο του 2019, παρουσιάστηκε μια νέα λειτουργία όπου επιτρέπει την δημιουργία chat rooms βάσει τοποθεσίας. Η ιδέα πίσω από αυτό ήταν η δημιουργία διάφορων δικτύων από άτομα που βρίσκονται στον ίδιο χώρο για την παραχώρηση ενός τρόπου επικοινωνίας, διαφορετικού από τους υπόλοιπους [7]. Με άλλα λόγια, μπορεί να χρησιμοποιηθεί σε εκδηλώσεις πολιτικού θέματος, σε πανεπιστημιούπολεις, για διαχείριση κινδύνων σε μια συγκεκριμένη περιοχή κ.α. Ο τρόπος δημιουργίας μιας συνομιλίας βάσει τοποθεσίας είναι αρκετά εύκολος και μετά, ο οποιοσδήποτε βρίσκεται στην περιοχή όπου η συγκεκριμένη συνομιλία έχει δημιουργηθεί, μπορεί να συμμετέχει σε αυτή, δηλαδή να στέλνει και να βλέπει μηνύματα. Η διαδικασία αυτή γίνεται μόνο όταν το GPS στο κινητό τηλέφωνο είναι ενεργοποιημένο και ως εκ τούτου, η λειτουργία των συνομιλιών αυτών δεν αποτελεί απειλή για την ασφάλεια των χρηστών που δεν επιθυμούν να συμμετέχουν.



Σχήμα 2.2 : Δημιουργία Local Group Chat μέσω του Telegram [36]

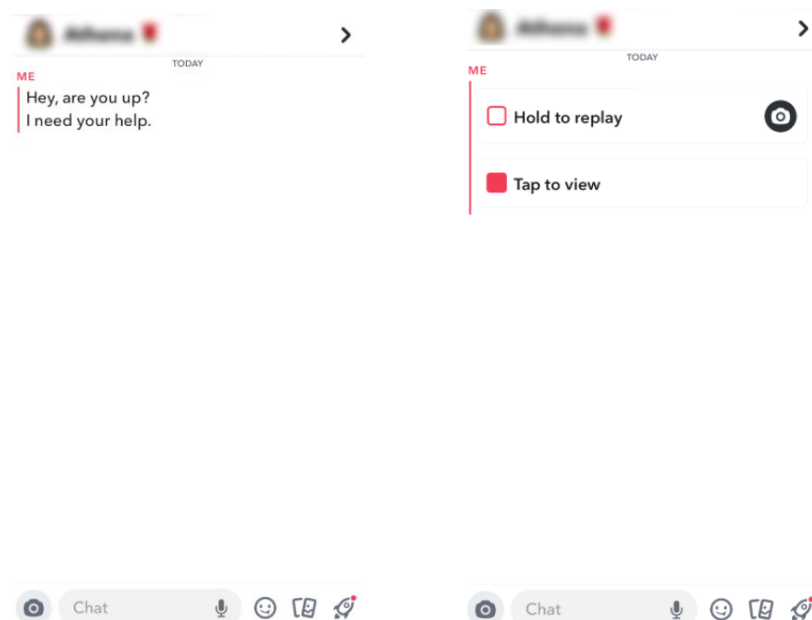
Μελετώντας και αναλύοντας τα παραπάνω 3 παραδείγματα εφαρμογών που κάνουν χρήση της τοποθεσίας για έναρξη συνομιλιών, αλλά και από άλλα που δεν αναφέρονται, μπορούμε να παρατηρήσουμε ότι η πλειονότητα τέτοιων συστημάτων παραχωρούν το δικαίωμα την ανωνυμίας στους χρήστες τους. Αυτό το χαρακτηριστικό είναι κάτι το οποίο δεν είναι απαραίτητο στην λειτουργία ενός έξυπνου συστήματος διαχείρισης κινδύνων. Αυτό γιατί η ταυτότητα και η τοποθεσία του αποστολέα είναι απαραίτητα στοιχεία προκειμένου να πραγματοποιηθεί μια ταχεία ανταπόκριση σε περίπτωση έκτακτης ανάγκης αλλά και για την επικύρωση ενός μηνύματος.

2.1.2 Συστήματα Ανταλλαγής Εφήμερων Μηνυμάτων

Η πρώτη ευρέως χρησιμοποιούμενη υπηρεσία που επικεντρώθηκε γύρω από εφήμερες αλληλεπιδράσεις χρηστών είναι η εφαρμογή για κινητά τηλέφωνα Snapchat [35]. Κυκλοφόρησε το 2011 για λειτουργικά Android και iOS, δημιουργημένη από τους Evan Spiegel, Bobby Murphy και Reggie Brown όταν ακόμη φοιτούσαν στο πανεπιστήμιο του Stanford. Ένα από τα κύρια χαρακτηριστικά της είναι πως, το οποιοδήποτε οπτικοακουστικό υλικό αλλά και τα γραπτά μηνύματα που στέλνονται σε μια συνομιλία, είναι διαθέσιμα μόνο για ένα μικρό χρονικό διάστημα, πριν διαγραφούν διαπαντός από τους διακομιστές της εταιρίας [3]. Η διαφορετική αυτή μορφή επικοινωνίας έμπνευσε αρκετές εταιρίες που διαθέτουν μέσα κοινωνικής δικτύωσης, έτσι ώστε να την υιοθετήσουν και αυτές με την σειρά τους στις εφαρμογές τους.

Μπορεί οι δημιουργοί της εφαρμογής Snapchat να θεωρούνται πρωτοπόροι με την επινόηση τους αυτή και να δέχτηκαν επί το πλείστον αξιόλογες κριτικές, όμως υπάρχουν αρκετά προβλήματα που πολλοί θεωρούν αξιοσημείωτα. Για παράδειγμα, ένα από τα πιο συχνά σχόλια που γίνονται για την εφαρμογή και το χαρακτηριστικό των εφήμερων μηνυμάτων, είναι πως οι χρήστες ξεχνάνε τι είχαν στείλει προηγουμένως στις συνομιλίες που έχουν. Έτσι, δεν έχουν την σωστή ιδέα για το γενικό πλαίσιο της συζήτησης και δεν μπορούν να απαντήσουν κατάλληλα στα εισερχόμενα μηνύματα. Ακόμη ένα μειονέκτημα είναι πως, ο οποιοσδήποτε εκφοβισμός γίνεται με την χρήση του Snapchat δεν μπορεί να αποδειχθεί αφού τα μηνύματα εξαφανίζονται και το ίδιο ισχύει με τις ανεπιθύμητες φωτογραφίες και βίντεο που λαμβάνονται.

Άλλα μέσα κοινωνικής δικτύωσης και επικοινωνίας που έχουν υιοθετήσει το χαρακτηριστικό των μηνυμάτων που μετά από κάποια χρονική διάρκεια εξαφανίζονται, είναι οι τρεις θυγατρικές της Meta (παλαιότερα γνωστή ως και Facebook Inc.), Instagram, Messenger και WhatsApp [24]. Στην εφαρμογή Instagram δίνεται η δυνατότητα στους χρήστες να στέλνουν φωτογραφίες στις συνομιλίες τους, οι οποίες διαγράφονται μόλις ο παραλήπτης τις ανοίξει και τις δει. Στο Messenger δημιουργείται μια κρυφή συνομιλία, για κάθε μια που ήδη υπάρχει, όπου οι δύο συνομιλητές μπορούν να επικοινωνήσουν, γνωρίζοντας ότι όταν απομακρυνθούν από αυτήν τότε όλα τα μηνύματα που έχουν στείλει θα διαγραφούν. Η λειτουργία στην εφαρμογή WhatsApp είναι περίπου όμοια, προσφέροντας στους χρήστες της την δυνατότητα να ρυθμίσουν το χρονικό διάστημα για το οποίο το μήνυμά τους θα είναι διαθέσιμο.



Σχήμα 2.3 : Στιγμιότυπα οθόνης από την εφαρμογή Snapchat

Η ιδιότητα των προσωρινών μηνυμάτων δίνει μια μοντέρνα χροιά σε όποια συστήματα υλοποιείται, αλλάζοντας βέβαια και τον παραδοσιακό τρόπο επικοινωνίας. Παρά τα αρνητικά στοιχεία που έρχονται μαζί με την υλοποίηση του διαφορετικού αυτού τρόπου συνομιλίας, τα εξαφανιζόμενα μηνύματα είναι το επιθυμητό αποτέλεσμα για την εφαρμογή SMAS. Αυτό γιατί όσο περισσότερος χρόνος περνάει από την αποστολή τους, τόσο λιγότερη σημασία και αξία θα έχουν για τα άτομα που θα χρησιμοποιούν την εφαρμογή.

Εφαρμογή	Ανωνυμία	Τοποθεσία	Εφήμερα Μηνύματα	Alerts
Yik Yak	Ναι	Ναι	Όχι	Όχι
Rayzit	Ναι	Ναι	Όχι	Όχι
Telegram	Όχι	Μερική	Όχι	Όχι
Snapchat	Όχι	Όχι	Ναι	Όχι
SMAS	Όχι	Ναι	Ναι	Ναι

Πίνακας 2.1 : Σύγκριση των εφαρμογών που αναφέρθηκαν στο υποκεφάλαιο 2.1

Στον πίνακα 2.1, μπορούμε να δούμε την σύγκριση μεταξύ των εφαρμογών που αναφέρονται πιο πάνω αλλά και της εφαρμογής SMAS που αναπτύξαμε. Συγκρίνονται τέσσερις ιδιότητες: η ανωνυμία, η επικοινωνία βάσει τοποθεσίας, τα εφήμερα μηνύματα και η ρητή λειτουργία για αποστολή alerts.

2.2 Σχετικά Συστήματα Εντοπισμού Εσωτερικής Τοποθεσίας

Τα συστήματα τα οποία προσφέρουν την δυνατότητα εντοπισμού εσωτερικής τοποθεσίας είναι στο στόχαστρο των ερευνητών που επιθυμούν να βελτιστοποιήσουν την πλοήγηση σε εσωτερικούς χώρους. Μελετώνται για περισσότερο από μια δεκαετία και τα αποτελέσματα των ερευνών ανταποκρίνονται στις σύγχρονες ανάγκες των χρηστών. Τέτοιου είδους συστήματα περιλαμβάνουν μοντέλα χώρων, όπως χάρτες και διάφορα σημεία ενδιαφέροντος, μαζί με ασύρματα, μαγνητικά σήματα και σήματα φωτός που χρησιμοποιούνται για να εντοπίσουν την θέση ενός χρήστη. Εκτός από χρήση σημάτων, υπάρχουν και συστήματα που χρησιμοποιούν την κάμερα των κινητών τηλεφώνων καθώς και τις μετρήσεις από τους ενσωματωμένους αισθητήρες. Στο υποκεφάλαιο αυτό περιγράφονται με συντομία οι δουλειές που είναι περισσότερο σχετικές με το πλαίσιο ενασχόλησης της διπλωματικής αυτής.

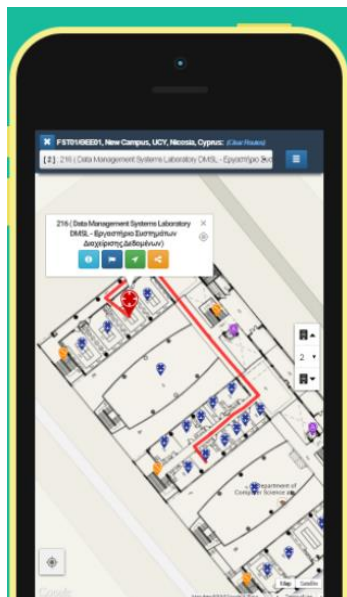
Anyplace

Το Anyplace [1] είναι μια δωρεάν υπηρεσία, ανοικτού κώδικα, για εντοπισμό τοποθεσίας σε εσωτερικά περιβάλλοντα και πλοήγηση σε αυτά. Αναπτύχθηκε από ερευνητές και

φοιτητές στο Data Management System Laboratory του Πανεπιστημίου Κύπρου και είναι προσβάσιμο από ηλεκτρονικούς υπολογιστές αλλά και από κινητά τηλέφωνα με λειτουργικό σύστημα Android .

Δεν κάνει χρήση του Παγκόσμιου Συστήματος Γεωτοποθέτησης (GPS) αλλά άλλων, πιο ακριβές σημάτων σε εσωτερικούς χώρους, όπως Wi-Fi, BLE, Cellular, UBW αλλά και της υπολογιστικής όρασης. Η λειτουργία της γεωτοποθέτησης αξιοποιεί την έννοια των «δακτυλικών» αποτυπωμάτων που αφήνουν τέτοια σήματα στις διάφορες εγκαταστάσεις.

Η διαδικασία εντοπισμού της τοποθεσίας ενός χρήστη έχει ως εξής: στην πρώτη φάση γίνεται η ανάκτηση των αποτυπωμάτων με την βοήθεια μιας συσκευής καταγραφής, που συνήθως είναι ένα κινητό τηλέφωνο. Τα αποτυπώματα αυτά είναι οι λαμβανόμενες ενδείξεις ισχύος του σήματος (RSSI), σε συγκεκριμένες συντεταγμένες (x, y), επισημασμένα στον χάρτη του κτηρίου ανά μερικά μέτρα. Ακολούθως, τα αποτυπώματα ενώνονται σε πίνακες Wi-Fi Radiomaps και ο χρήστης μπορεί να συγκρίνει τις δικές του παρατηρήσεις αποτυπωμάτων έναντι των πινάκων αυτών και να βρει τη καλύτερη αντιστοίχιση μέσω των αλγορίθμων KNN ή WKNN [26].



Σχήμα 2.4 : Το σύστημα Anyplace [1]

Redpin

Το Redpin [5, 33] είναι ακόμη ένα σύστημα ανοικτού κώδικα το οποίο αναπτύχθηκε με στόχο τον εντοπισμό της εσωτερικής τοποθεσίας, κάνοντας χρήση των σημάτων Wi-Fi.

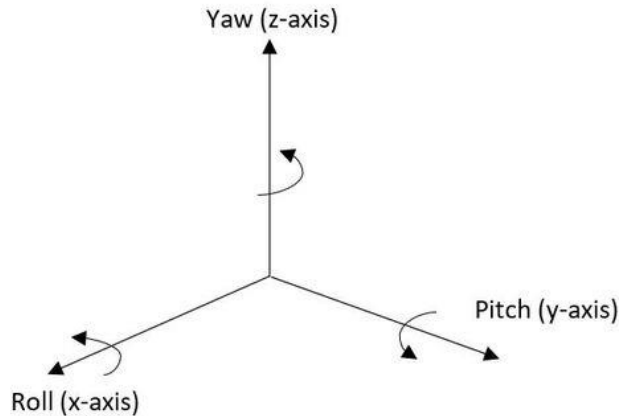
Βασίζεται πάνω στις συνεισφορές της κοινότητας των χρηστών του, αποφεύγοντας έτσι τη χρονοβόρα διαδικασία εκπαίδευσης. Μπορεί να εγκατασταθεί σε κινητά με λειτουργικά Android και iOS.

Τα αποτελέσματα του Redpin δεν είναι γεωγραφικές συντεταγμένες, αλλά το όνομα ή ο αριθμός ενός χώρου (βάση του χάρτη), παραχωρώντας έτσι ακρίβεια σε επίπεδο δωματίου. Τα μοντέλα με τα αποτυπώματα των σημάτων που χρησιμοποιούνται για τον εντοπισμό της τοποθεσίας δεν είναι στατικά, αφού η εφαρμογή δεν διαφοροποιεί την φάση της εκπαίδευσης με την φάση της χρήσης. Έτσι, ο κάθε χρήστης που χρησιμοποιεί την εφαρμογή, δυναμικά στέλνει δεδομένα έτσι ώστε να αναδημιουργηθούν τα μοντέλα αποτυπωμάτων. Αυτό έχει ως αποτέλεσμα την εύκολη προσαρμογή στις αλλαγές των θέσεων σημείων πρόσβασης. Εκτός αυτού, για να αυξηθούν τα επίπεδα ακρίβειας και προσαρμοστικότητας της εφαρμογής, το Redpin αφήνει πολλαπλές μετρήσεις που λαμβάνονται διαδοχικά, να εισάγονται ως το ίδιο αποτύπωμα.

An Indoor Position-Estimation Algorithm Using Smartphone IMU Sensor Data

Οι Alwin Poulouse, Odongo Steven Eyobu και Dong Seog Han μέσα από την επιστημονική μελέτη τους ονόματι « An Indoor Position-Estimation Algorithm Using Smartphone IMU Sensor Data» [31] πρότειναν ένα αλγόριθμο εντοπισμού εσωτερικής τοποθεσίας με χρήση μόνο του κινητού τηλεφώνου. Πιο συγκεκριμένα, ο αλγόριθμος αυτός συνδυάζει τις μετρήσεις από ένα IMU αισθητήρα, δηλαδή από το επιταχυνσιόμετρο, μαγνητόμετρο και γυροσκόπιο, τα οποία μπορούν να βρεθούν ενσωματωμένα στα πλείστα έξυπνα κινητά τηλέφωνα.

Σύμφωνα με τους ερευνητές, υλοποιήθηκαν και αξιολογήθηκαν τρία σημαντικά συστατικά στοιχεία, τα οποία έχουν ως ακολούθως: ένα φίλτρο Kalman για δύο αλγόριθμους που χρησιμοποιούν μετρήσεις από αισθητήρες, ένας αλγόριθμος για ανίχνευση βημάτων και ο συνδυασμός των παραπάνω για την δημιουργία ενός αλγορίθμου εκτίμησης θέσης. Οι δύο αλγόριθμοι που κάνουν χρήση του φίλτρου Kalman υπολογίζουν την κατεύθυνση κίνησης του σώματος καθώς και τις διαστάσεις κινήσεων pitch και roll. Στο σχήμα 2.5 φαίνεται μια απεικόνιση των δύο αυτών διαστάσεων.



Σχήμα 2.5 : Οι τρεις διαστάσεις κινήσεων [34]

Το προτεινόμενο σύστημα ανίχνευσης βημάτων έχει μέσο σφάλμα 2.5%, ενώ το σφάλμα στην κατεύθυνση κίνησης είναι στις 4.72 μοίρες. Με τα αποτελέσματα που έλαβαν από τα πειράματα τα οποία εκτέλεσαν, έβγαλαν το συμπέρασμα ότι ο προτεινόμενος αλγόριθμος έχει υψηλή ακρίβεια στην εύρεση θέσης, που ξεπερνά σημαντικά τις επιδόσεις άλλων συμβατικών μεθόδων εκτίμησης.

Resource-efficient and Automated Image-based Indoor Localization

Ο υπολογισμός της εσωτερικής τοποθεσίας με την χρήση της κάμερας του κινητού τηλεφώνου και φωτογραφιών είναι ακόμη μια περιοχή μελέτης που λαμβάνει αρκετό ενδιαφέρον από ερευνητές. Η τεχνική αυτή βασίζεται κυρίως σε ορόσημα που βρίσκονται σε ένα χώρο, δηλαδή φυσικά αντικείμενα που η ακριβής τοποθεσία τους είναι γνωστή. Η διαφορά με τα συστήματα που κάνουν χρήση σημάτων για εντοπισμό της θέσης, είναι ότι τέτοια αντικείμενα έχουν μεγάλη διάρκεια ζωής στο σύστημα (εβδομάδες ή μήνες) και η περίπτωση μετακίνησης τους είναι ελάχιστη.

Οι εφαρμογές που υιοθετούν την τεχνική αυτή συνήθως χρησιμοποιούν το μοντέλο πελάτη-εξυπηρετητή, όπου οι εξυπηρετητές είναι υπεύθυνοι για να επεξεργάζονται τα queries των χρηστών. Αυτό συνεπάγεται σε ένα μη-επεκτάσιμο αριθμό από χρήστες, καθυστέρηση, προβλήματα ασφάλειας προσωπικών δεδομένων και σύνδεση στο διαδίκτυο που μπορεί να μην είναι πάντοτε εφικτή. Οι Qun Niu και άλλοι, μέσω της έρευνας τους με τίτλο «Resource-efficient and Automated Image-based Indoor

Localization» [28], προτείνουν τον **Highly Automated Image-Based Localization** αλγόριθμο (HAIL) όπου μπορεί να τρέξει σε συσκευές με περιορισμένους διαθέσιμους πόρους.

Διαφορετικό από άλλα συστήματα, το HAIL είναι αυτοματοποιημένο, δηλαδή μπορεί να αναγνωρίσει ένα ορόσημο χωρίς τη ρητή επιβεβαίωση από εικόνες. Πιο συγκεκριμένα, δεν γίνεται αποθήκευση ολόκληρων των εικόνων που χρησιμοποιήθηκαν για την εκπαίδευση αλλά επιλεγμένων, ευδιάκριτων χαρακτηριστικών τους, κάτι το οποίο μειώνει τον απαιτούμενο χώρο αποθήκευσης στη κινητή συσκευή. Εκτός αυτού, για την περεταίρω βελτίωση στην ακρίβεια εντοπισμού τοποθεσίας, χρησιμοποιεί κατόψεις κτηρίων και οροφών και λαμβάνει υπόψη τη κατεύθυνση της πυξίδας, το γυροσκόπιο και τις αποστάσεις μεταξύ των οροσμήμων.

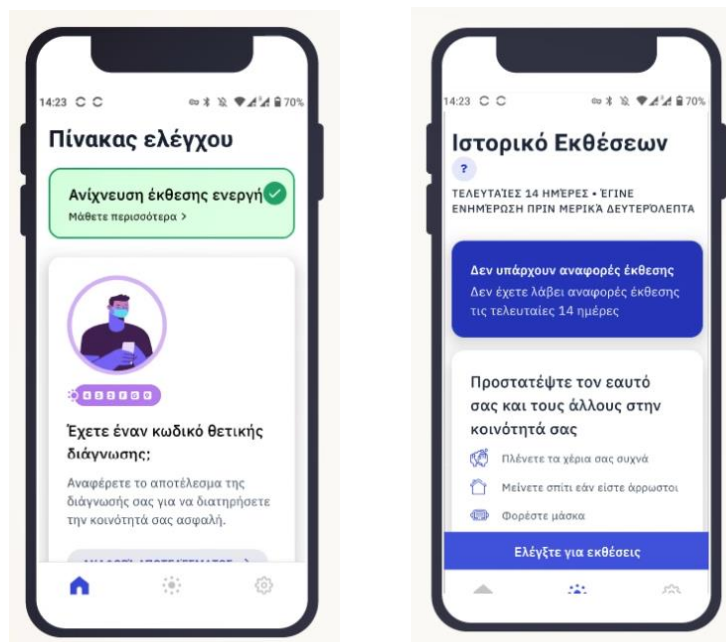
Πραγματοποιώντας εκτεταμένα πειράματα σε πλατείες και μεγάλα πολυκαταστήματα, τα αποτελέσματα που πήραν οι ερευνητές ήταν αρκετά ικανοποιητικά. Είχαν κατά μέσο όρο 20% λιγότερα σφάλματα γεωτοποθέτησης και 40% περισσότερη αποδοτικότητα υπολογιστικής ισχύος σε σχέση με άλλα παρόμοια συστήματα.

2.3 Σχετικά Έξυπνα Συστήματα Διαχείρισης Κινδύνων

Με τις φυσικές καταστροφές που γίνονται όλα και πιο συχνά ανά το παγκόσμιο (πυρκαγιές, σεισμοί, κ.α.), η ύπαρξη συστημάτων για αποστολή άμεσων ειδοποιήσεων στα άτομα που βρίσκονται κοντά, έτσι ώστε να μπορέσουν να φύγουν όσο το δυνατό πιο γρήγορα, είναι απαραίτητη. Επίσης, τέτοιες υλοποιήσεις μπορούν να χρησιμοποιηθούν και για την ταχεία ενημέρωση των ατόμων που θα χρειαστεί να είναι παρόντες στα σημεία αυτά, ώστε να παρέχουν βοήθεια (πυροσβέστες, νοσοκόμοι, κ.α.). Τα τελευταία χρόνια, με το ξέσπασμα του COVID-19, οι ερευνητικές κοινότητες της υγείας και της πληροφορικής έσπευσαν να αναπτύξουν εφαρμογές άμεσων ειδοποιήσεων χρησιμοποιώντας την τοποθεσία των χρηστών. Παρακάτω γίνεται αναφορά σε δύο υλοποιήσεις όπου κοινός τους άξονας είναι η συγκεκριμένη λειτουργία αποστολής ειδοποιήσεων, έτσι ώστε η διαχείριση κινδύνων να γίνεται πιο αποτελεσματικά.

A Privacy-Preserving Location-Based Alert System

Η παρακολούθηση της θέσης των χρηστών των κινητών τηλεφώνων έχει σημαντική εφαρμογή στα συστήματα διαχείρισης κινδύνων και όχι μόνο. Ωστόσο, η θέση, η κίνηση και ο προορισμός ενός χρήστη αποτελούν ευαίσθητες πληροφορίες. Επομένως, η προστασία τους είναι ένα από τα πιο σημαντικά μελήματα που πρέπει να έχουν οι προγραμματιστές τέτοιων συστημάτων. Για τον λόγο αυτό, οι Gabriel Ghinita και Razvan Rughinis, στην έρευνα τους με τίτλο «A Privacy-Preserving Location-Based Alert System» [15], προτείνουν ένα σύστημα όπου η τοποθεσία ενός χρήστη γίνεται ορατή υπό κάποιες προϋποθέσεις. Η λειτουργία αυτή ονομάζεται *searchable encryption* και παραχωρεί εγγύηση ασφάλειας ευαίσθητων πληροφοριών. Η αποκρυπτογράφηση τους γίνεται μόνο όταν η τοποθεσία του χρήστη ικανοποιεί κάποιες συνθήκες, όπως για παράδειγμα την εγγύτητα σε κάποια καταστροφή ή συμβάν, έτσι ώστε ο χρήστης να είναι ενήμερος.



Σχήμα 2.6 : Στιγμιότυπα οθόνης από την εφαρμογή CovTracer-EN [12]

CovTracer-EN

Η επίσημη εφαρμογή της Κυπριακής Κυβέρνησης CovTracer-EN [12], ανεπτυγμένη από προγραμματιστές των ερευνητικών κέντρων KIOS και CYENS, δημιουργήθηκε με σκοπό την ανίχνευση κρουσμάτων COVID-19. Κάνει χρήση των σημάτων Bluetooth, τα οποία ανιχνεύει η εφαρμογή από άλλες συσκευές που έχουν εγκατεστημένη την ίδια

εφαρμογή. Η ανίχνευση γίνεται μόνο σε πολίτες οι οποίοι βρίσκονται στην μέγιστη απόσταση των 2 μέτρων για περισσότερο από 15 λεπτά. Οι επαφές αυτές αποθηκεύονται για 2 εβδομάδες χωρίς να αποκαλύπτεται η οποιαδήποτε προσωπική πληροφορία τους (όνομα, τοποθεσία). Αν κάποιος χρήστης της εφαρμογής διαγνωστεί με κορονοϊό, τότε του δίνεται η δυνατότητα να εισάγει ένα μοναδικό κωδικό στο σύστημα και να σταλεί η πληροφορία στις εφαρμογές των χρηστών με τους οποίους ήρθε σε επαφή. Έτσι, το σύστημα αυτό που αναπτύχθηκε, λειτουργεί σαν ένα μέσο αποστολής ειδοποιήσεων και ενημέρωσης των πολιτών όταν είναι στενές επαφές κρούσματος.

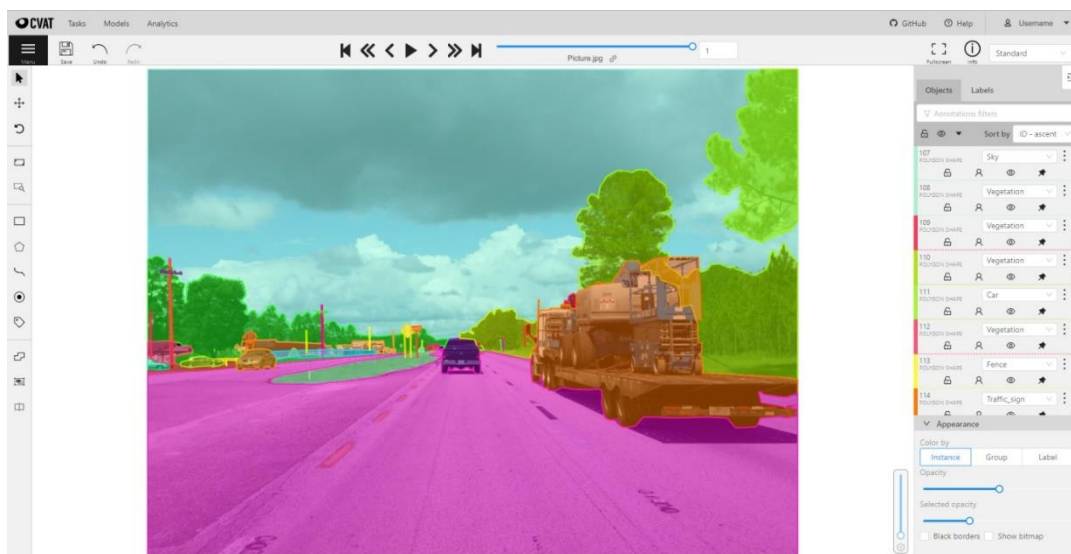
2.4 Τεχνολογικό Υπόβαθρο

Για την ανάπτυξη των υποσυστημάτων στα πλαίσια της διπλωματικής αυτής, η οποία υλοποίηση τους αναλύεται περαιτέρω στα κεφάλαια που ακολουθούν, έγινε χρήση αρκετών μοντέρνων τεχνολογιών. Οι τεχνολογίες αυτές μπορούν να χωριστούν σε 3 κατηγορίες: τεχνολογίες που χρησιμοποιήθηκαν στο backend μέρος των συστημάτων, τεχνολογίες που χρησιμοποιήθηκαν στο frontend μέρος και τεχνολογίες ελέγχου έκδοσης λογισμικού.

2.4.1 Τεχνολογίες Backend

CVAT

Το Computer Vision Annotation Tool (CVAT) [9] είναι ένα δωρεάν, ανοικτού κώδικα εργαλείο που αναπτύχθηκε από την Intel και το οποίο χρησιμοποιείται για σχολιασμό βίντεο. Annotation (ή αλλιώς σχολιασμός) είναι η διαδικασία επεξεργασίας ενός σύνολου από βίντεο, όπου σε κάθε στιγμιότυπο σημειώνεται τι είναι το κάθε αντικείμενο που είναι ορατό. Η διαδικασία γίνεται από μια σωστά εκπαιδευμένη ομάδα ατόμων και μπορεί να πάρει αρκετές εβδομάδες, ανάλογα με το πλήθος των βίντεο που θα τύχουν ανάλυσης. Το CVAT επιταχύνει την διεργασία αυτή αφού προσφέρει ημιαυτόματο σχολιασμό χρησιμοποιώντας μοντέλα βαθιάς μάθησης. Είναι ένα εργαλείο για τα αρχικά στάδια της επιβλεπόμενης μάθησης και τα αποτελέσματα του χρησιμοποιούνται για αλγόριθμους υπολογιστικής όρασης έτσι ώστε η αναγνώριση αντικειμένων να γίνεται σε πραγματικό χρόνο.



Σχήμα 2.7 : Στιγμιότυπο οθόνης από το σύστημα CVAT [9]

YOLO

Το YOLO [39] είναι ένα σύγχρονο, πραγματικού χρόνου, σύστημα ανίχνευσης αντικειμένων ανεπτυγμένο από τον Joseph Redmon. Είναι αρκετά πιο γρήγορο και ακριβές στις ανιχνεύσεις του σε σχέση με όμοια συστήματα και είναι υλοποιημένο με χρήση του TensorFlow 2.0.

TensorFlow

Το TensorFlow [37] είναι μια δωρεάν, ανοικτού κώδικα βιβλιοθήκη για μηχανική μάθηση και τεχνική νοημοσύνη, ανεπτυγμένη από την ομάδα του Google Brain. Παρέχει διάφορες δυνατότητες όπως, προ-επεξεργασία και μεταφορά δεδομένων, εκπαίδευση και εκτίμηση μοντέλων, απεικόνιση και εφαρμογή των μοντέλων σε διάφορες συσκευές. Για παράδειγμα, το TensorFlow Lite δίνει την δυνατότητα στους προγραμματιστές να τρέχουν τα ήδη εκπαιδευμένα τους μοντέλα σε συσκευές που δεν έχουν μεγάλη υπολογιστική δύναμη όπως κινητά τηλέφωνα, ενσωματωμένες συσκευές και συσκευές αιχμής.

Κέλυφος Bash

Το Bash [2] είναι ένα κέλυφος ανοικτού κώδικα για τα λειτουργικά συστήματα Unix, το οποίο είναι σύμφωνο με το πρότυπο POSIX. Είναι ένας επεξεργαστής όπου ο χρήστης μπορεί να τρέξει εντολές χαμηλού επιπέδου και να βλέπει τα αποτελέσματα σε ένα απλό παράθυρο κειμένου. Επίσης, μπορεί να τρέξει και εντολές που βρίσκονται μέσα σε αρχείο, το οποίο αρχείο συνήθως έχει την κατάληξη .sh και ονομάζεται shell script.

2.4.2 Τεχνολογίες Frontend

Kotlin

Η Kotlin [21] είναι μια γενικού σκοπού, αντικειμενοστραφής γλώσσα προγραμματισμού που αναπτύχθηκε από την ομάδα JetBrains το 2011. Τρέχει πάνω στην εικονική μηχανή της Java (JVM) και αξιοποιεί τις υπάρχουσες βιβλιοθήκες της. Ένα πρόγραμμα Kotlin μπορεί να περιέχει κώδικα γραμμένο σε Java και το αντίστροφο. Η Google πλέον θεωρεί την Kotlin σαν την πιο κατάλληλη γλώσσα προγραμματισμού εφαρμογών για Android λειτουργικά συστήματα.

Jetpack Compose

Το Jetpack Compose [20] είναι ένα μοντέρνο Android Framework που αναπτύχθηκε με σκοπό την διευκόλυνση σχεδιασμού διεπαφής χρήστη (UI) μέσα στο περιβάλλον που γίνεται η ανάπτυξη των εφαρμογών. Το κτίσιμο μιας εφαρμογής γίνεται με λιγότερο κώδικα, ισχυρά εργαλεία και συναρτήσεις που είναι εύκολα προσβάσιμες. Επιπλέον, δίνεται η δυνατότητα στον προγραμματιστή να έχει μια γενική εικόνα για το πως θα φαίνεται η διεπαφή που αναπτύσσει χωρίς την ανάγκη να κτίσει ολόκληρο τον κώδικα, με την χρήση του Compose Tooling.

MVVM

Το Model-View-ViewModel (MVVM) είναι ένα αρχιτεκτονικό μοτίβο λογισμικού που επινοήθηκε από τους Ken Cooper και Ted Peters της Microsoft. Διευκολύνει τον διαχωρισμό της ανάπτυξης εφαρμογών με γραφική διεπαφή χρήστη, έτσι ώστε η προβολή

(view) να μην βασίζεται σε κανένα συγκεκριμένο μοντέλο ή πλατφόρμα. Η αρχιτεκτονική αυτή αποτελείται από 3 βασικά επίπεδα. Το μοντέλο (model), την προβολή (view) και το view model. Κάθε ένα από αυτά εξυπηρετεί ένα διαφορετικό σκοπό, ο οποίος πρέπει να τηρείται για μια αποτελεσματική και «καθαρή» υλοποίηση του μοτίβου. Οι ευθύνες του κάθε επιπέδου έχουν ως εξής:

Μοντέλο (Model) → αποτελείται από κλάσεις που δεν χρειάζονται οπτική αναπαράσταση, οι οποίες κρατάνε τα δεδομένα της εφαρμογής μαζί με τον οποιοδήποτε κώδικα λογικής και επικύρωσης.

Προβολή (View) → υπεύθυνη για τον καθορισμό της δομής, εμφάνισης και διάταξης των στοιχείων που βλέπει ο χρήστης στην οθόνη.

View Model → υπεύθυνο για την ενημέρωση του επιπέδου προβολής για οποιαδήποτε αλλαγή κατάστασης στο μοντέλο, καθώς και για τον καθορισμό των λειτουργιών που θα προσφέρονται από την εφαρμογή.

2.4.1 Τεχνολογίες Ελέγχου Έκδοσης Λογισμικού

GitHub

Το GitHub [16] είναι μια πλατφόρμα φιλοξενίας κώδικα και ελέγχου έκδοσης λογισμικού που βρίσκεται υπό ανάπτυξη. Είναι θυγατρική εταιρία της Microsoft και είναι πλέον η μεγαλύτερη πλατφόρμα του είδους της. Παρέχει έλεγχο πρόσβασης στα διάφορα έργα, καθώς και αρκετές δυνατότητες συνεργασίας μεταξύ προγραμματιστών που εργάζονται από κοινού στην ίδια εφαρμογή. Κυρίως, χρησιμοποιείται για την δημοσίευση λογισμικού ανοιχτού κώδικα, έτσι ώστε να μπορέσει ο οποιοσδήποτε ενδιαφερόμενος να έχει πρόσβαση σε αυτό.

Κεφάλαιο 3

Ανάλυση Συστημάτων Anyplace και SMAS

3.1	Ανάγκη για Υπολογισμό Εσωτερικής Τοποθεσίας	20
3.2	Ανατομία του Συστήματος Anyplace	21
3.2.1	Anyplace Logger	22
3.2.2	Anyplace Navigator	24
3.2.3	Anyplace Architect	24
3.2.4	Anyplace Viewer	25
3.3	Περιβάλλοντα Χωρίς Υποδομές	25
3.4	Εφαρμογή SMAS	26
3.4.1	Anyplace και SMAS	27
3.4.2	Τεχνολογίες στο Σύστημα SMAS	28

Στο κεφάλαιο αυτό αναλύουμε την ανάγκη για συστήματα που μπορούν να υπολογίσουν τη θέση σε εσωτερικά περιβάλλοντα. Για τον λόγο αυτό, στη συνέχεια, μελετούμε το οικοσύστημα Anyplace που παρέχει την λειτουργία αυτή, τις βασικές εφαρμογές που το απαρτίζουν και μέρη της αρχιτεκτονικής του. Ακολούθως, παρουσιάζουμε τους λόγους για τους οποίους το Anyplace δεν μπορεί να έχει εφαρμογή σε όλους τους εσωτερικούς χώρους και δίνουμε την λύση παρουσιάζοντας το σύστημα SMAS.

3.1 Ανάγκη για Υπολογισμό Εσωτερικής Τοποθεσίας

Η χρήση συστημάτων για πλοήγηση σε εξωτερικούς χώρους έχει γίνει πλέον ρουτίνα για οδηγούς και πεζούς, ειδικά στις περιπτώσεις όπου βρίσκονται σε άγνωστες περιοχές, σε διαφορετική χώρα ή έχουν χαθεί. Αφού τα πλείστα συστήματα πλέον δεν χρειάζονται σύνδεση στο διαδίκτυο για να δίνουν οδηγίες, η χρήση τους μπορεί να γίνεται από οπουδήποτε. Τέτοια συστήματα κάνουν χρήση του παγκόσμιου συστήματος εντοπισμού

γεωγραφικής θέσης (GPS), το οποίο βασίζεται σε ένα πλέγμα από 24 δορυφόρους της Γης που εκπέμπουν σήματα.

Τα συστήματα πλοήγησης σε εσωτερικούς χώρους είναι απαραίτητα για τους ίδιους λόγους που είναι και αυτά για πλοήγηση σε εξωτερικούς. Για παράδειγμα, στις περιπτώσεις των ταξιδιωτών όπου βρίσκονται σε ένα αεροδρόμιο που δεν γνωρίζουν, μια εφαρμογή η οποία τους δίνει οδηγίες για να φτάσουν στην πύλη, στον έλεγχο διαβατηρίων, στο τερματικό κτλ., θα ήταν πολύ χρήσιμη. Έγιναν αρκετές προσπάθειες για χρήση του GPS σε εσωτερικά περιβάλλοντα χωρίς όμως ικανοποιητικά αποτελέσματα. Αυτό λόγω του εξασθενημένου σήματος, το οποίο πρέπει να περάσει μέσα από επίπεδα τοίχων και μετάλλων και της πολύ μικρής ακρίβειας που παρέχει.

Για τον λόγο αυτό η ερευνητική κοινότητα μελετάει άλλους, διαφορετικούς τρόπους υπολογισμού εσωτερικής τοποθεσίας. Κυρίως, γίνεται χρήση άλλων σημάτων που είναι διαθέσιμα σε εσωτερικούς χώρους και η ισχύς τους είναι ικανοποιητική για τον υπολογισμό της εσωτερικής θέσης [6]. Πιο κάτω, γίνεται ανάλυση του συστήματος Anyplace το οποίο κάνει χρήση των σημάτων Wi-Fi που εκπέμπονται από τα διάφορα Access Points μέσα σε ένα κτήριο.

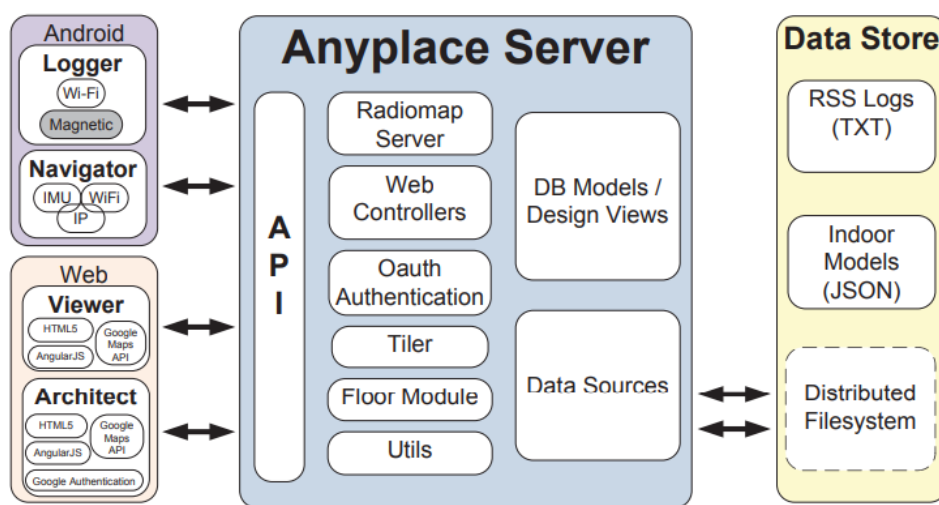
3.2 Ανατομία του Συστήματος Anyplace

Το σύστημα Anyplace [40], στο οποίο γίνεται αναφορά στο υποκεφάλαιο 2.2 «Σχετικά Συστήματα Εντοπισμού Εσωτερικής Τοποθεσίας», συνδυάζει ένα πλήθος από μεθόδους για τον εντοπισμό της εσωτερικής τοποθεσίας ενός ανθρώπου, αντικειμένου ή έξυπνου μηχανήματος. Με την πάροδο των χρόνων όλο και περισσότερες μέθοδοι ενσωματώνονται στο σύστημα, με στόχο να δοθεί η δυνατότητα πλοήγησης σε διάφορα είδη εσωτερικών χώρων καθώς και για την βελτίωση των ήδη υπάρχοντων.

Αποτελείται από τέσσερις βασικές εφαρμογές οι οποίες συνδυαστικά κάνουν την λειτουργία της πλοήγησης εφικτή. Έχουν ως εξής: Logger [14] και Navigator [13] οι οποίες είναι εφαρμογές για κινητά τηλέφωνα και Viewer και Architect οι οποίες είναι εφαρμογές διαδικτύου. Όλες αυτές οι εφαρμογές αλληλεπιδρούν με τον Anyplace Server ο οποίος εξυπηρετεί το Anyplace API, παρέχει τα μοντέλα στις εφαρμογές, επικυρώνει τους χρήστες και κάνει άλλες λειτουργίες σχετικές με το σύστημα. Ο server αυτός

επικοινωνεί με το Anyplace Data Store όπου βρίσκονται αποθηκευμένα τα μοντέλα για τους χώρους, καθώς και οι μετρήσεις RSS.

Οι εντολές πλοήγησης στέλνονται από τον διακομιστή σε μορφή JSON. Το JSON αντιπροσωπεύει το όνομα JavaScript Object Notation και είναι ένας ελαφρύς τρόπος ανταλλαγής δεδομένων, κυρίως μέσω διαδικτύου. Είναι εύκολο τόσο να διαβαστεί από ανθρώπους, όσο και να επεξεργαστεί από τις μηχανές. Ο χρήστης μπορεί να προσδιορίσει αντικείμενα, πίνακες και απλές τιμές στο σώμα του αρχείου, καλύπτοντας έτσι ένα μεγάλο εύρος των χαρακτηριστικών που έχουν οι προγραμματιστικές γλώσσες [8].



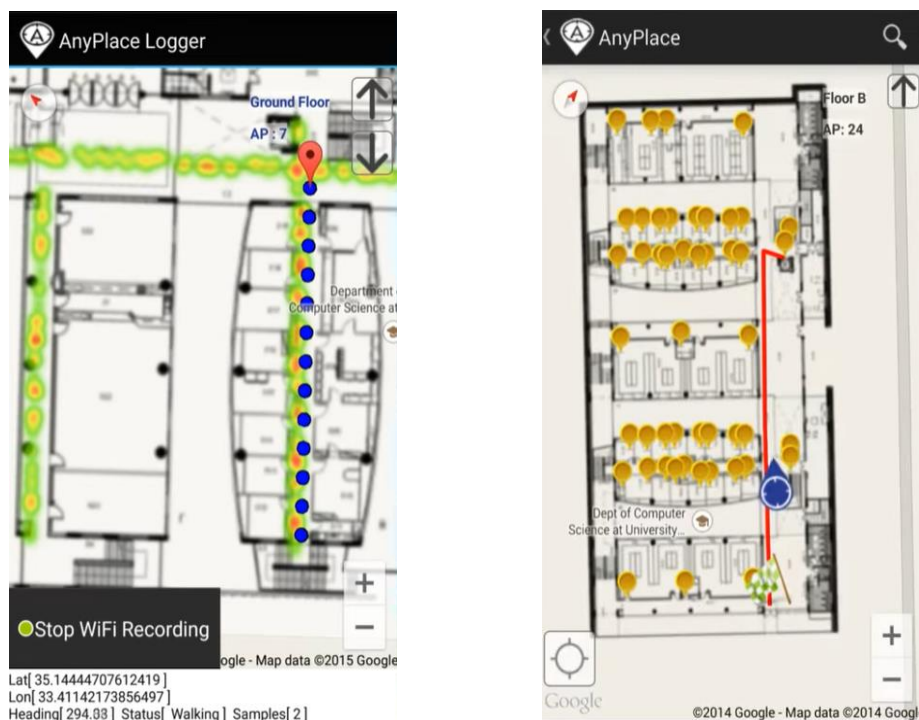
Σχήμα 3.1 : Αρχιτεκτονική του Anyplace [1, 40]

3.2.1 Anyplace Logger

Η εφαρμογή Logger για κινητά τηλέφωνα με λειτουργικό Android επιτρέπει στους εθελοντές να συμβάλλουν στην προσπάθεια βελτίωσης του εντοπισμού τοποθεσίας. Βάσει του χώρου στον οποίο βρίσκονται, μπορούν να επιλέξουν το κτήριο ή τον όροφο από τον οποίον θα συλλέξουν δεδομένα. Τα κτήρια αυτά νοείται ότι υπάρχουν ήδη στην γκάμα των δημόσιων κτηρίων του Anyplace. Αν αυτό δεν ισχύει, τότε μπορούν να τα προσθέσουν με τη χρήση της εφαρμογής Architect στην οποία αναφερόμαστε παρακάτω.

Μετά από την επιλογή, ο χρήστης πρέπει να υποδείξει την τρέχουσα του τοποθεσία πατώντας πάνω στον χάρτη (ή κάτοψη) που φαίνεται στην οθόνη του. Έπειτα, η διαδικασία είναι απλή. Ο χρήστης χρειάζεται να περπατήσει και να καλύψει όση

περισσότερη απόσταση μπορεί από το επιλεγόμενο κτήριο/ όροφο, έχοντας πάντα ενεργή την εφαρμογή. Με τον τρόπο αυτό, η εφαρμογή ανιχνεύει τα σήματα Wi-Fi που εκπέμπουν τα διάφορα Access Points (APs) στο κτήριο και τα αποθηκεύει. Πιο συγκεκριμένα, όπως αναφέρθηκε και προηγουμένως, η μέτρηση που αποθηκεύεται είναι η ένταση που έχει το κάθε σήμα. Έτσι, σε διαφορετικές τοποθεσίες οι εντάσεις των σημάτων θα είναι διαφορετικές. Για παράδειγμα, στην θέση δίπλα από ένα AP η ένταση των Wi-Fi σημάτων θα είναι πιο δυνατή απ' ό τι σε μια θέση με 20 μέτρα απόσταση από το ίδιο AP.



Σχήμα 3.2 : Στιγμιότυπα οθόνης των εφαρμογών Logger (αριστερά - 2015) και Navigator (δεξιά - 2014) [13, 14]

Για να διασφαλιστεί η καλύτερη ποιότητα συλλογής δεδομένων, παρουσιάζεται ένας θερμικός χάρτης στον χρήστη όπου επισημαίνει τα σημεία στο κτήριο όπου δεν πάρθηκαν αρκετές μετρήσεις. Έτσι, ο χρήστης θα μπορεί να μετακινηθεί στα σημεία αυτά ώστε να συμβάλει στην συλλογή. Μετά το τέλος της διαδικασίας αυτής, δίνεται η επιλογή της αποστολής των μετρήσεων στο cloud σε μορφή JSON έτσι ώστε να είναι διαθέσιμες και σε άλλους χρήστες.

3.2.2 Anyplace Navigator

Η εφαρμογή Navigator είναι αυτή που δίνει την δυνατότητα στους χρήστες να δουν την τρέχουσα τοποθεσία τους σε ένα κτήριο με τη χρήση κατόψεων. Η εφαρμογή κατεβάζει τον χάρτη με τις μετρήσεις RSS (του κτηρίου που βρίσκεται ο χρήστης), που πάρθηκαν με την βοήθεια του Logger, και συγκρίνει τις μετρήσεις σημάτων Wi-Fi που λαμβάνει την καθορισμένη χρονική στιγμή με τις μετρήσεις στον χάρτη αυτόν. Έτσι, η τοποθεσία πάνω στον χάρτη με τις μικρότερες διαφορές στις εντάσεις των σημάτων με αυτές που λαμβάνει είναι και η τοποθεσία του χρήστη στο κτήριο.

Εκτός αυτού, το Navigator δίνει και την δυνατότητα πλοήγησης μέσα σε εσωτερικούς χώρους. Ο χρήστης μπορεί να επιλέξει ένα σημείο ενδιαφέροντος που υπάρχει ήδη και να του δοθούν οδηγίες για το πως θα φτάσει στο σημείο αυτό, ανανεώνοντας κάθε φορά την καινούρια του θέση πάνω στην κάτοψη του κτηρίου. Για να βελτιωθεί περεταίρω η ακρίβεια του συστήματος, η τοποθεσία του χρήστη διορθώνεται με την χρήση των ενσωματωμένων αισθητήρων στο κινητό, όπως το γυροσκόπιο, το επιταχυνσιόμετρο και την πυξίδα. Η ακρίβεια του συστήματος είναι στα 1.96 μέτρα από την πραγματική θέση του χρήστη.

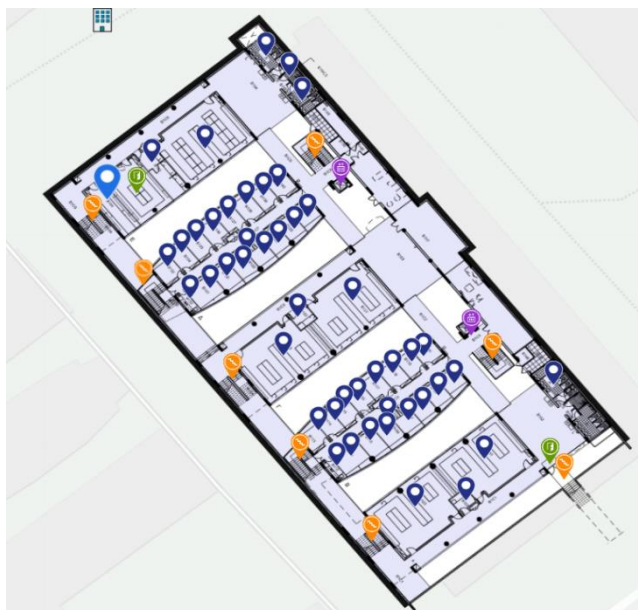
3.2.3 Anyplace Architect

Η εφαρμογή διαδικτύου Architect δίνει την ευκαιρία προσθήκης και διαχείρισης μοντέλων εσωτερικών χώρων που θα γίνουν, ή είναι ήδη διαθέσιμα στο γενικό σύστημα Anyplace για όλες τις λειτουργίες που παρέχει. Ο χρήστης μπορεί να προσθέσει εικόνες αρχιτεκτονικών σχεδίων από ένα κτήριο στον χάρτη, στην ακριβή τους τοποθεσία, ακόμη και πολύωροφων κτηρίων. Ακολουθώντας, ο χρήστης μπορεί να προσθέσει σημεία ενδιαφέροντος πάνω στο κτήριο, όπως εισόδους/ εξόδους, σκάλες, δωμάτια κ.α. και να τα ενώσει, δημιουργώντας έτσι μονοπάτια μέσα στον χώρο. Αργότερα, τα μονοπάτια αυτά θα χρησιμοποιηθούν για να δοθούν οι οδηγίες πλοήγησης μέσω του Navigator και του Viewer.

Τα κτήρια, εκτός από δημόσια, μπορούν να δηλωθούν ως και ιδιωτικά. Τα δημόσια κτήρια είναι ορατά από όλη την κοινότητα, καθώς και τα σημεία ενδιαφέροντος που έχουν δηλωθεί επάνω. Τα ιδιωτικά από την άλλη, μόνο τα άτομα τα οποία διαθέτουν το URL τους μπορούν να τα δουν και να προηγηθούν σε αυτά.

3.2.4 Anyplace Viewer

Το Anyplace Viewer είναι επίσης μια εφαρμογή διαδικτύου στην οποία ο καθένας μπορεί να αναζητήσει κτήρια, να δει τα πλάνα των εσωτερικών τους χώρων και τις θέσεις των σημείων ενδιαφέροντος. Επίσης, δίνονται οδηγίες πλοήγησης από ένα σημείο σε ένα άλλο, δεδομένου ότι δημιουργήθηκαν τα μονοπάτια προηγουμένως με την βοήθεια του Architect.



Σχήμα 3.3 : Παράδειγμα χαρτογραφημένου εσωτερικού χώρου με σημεία ενδιαφέροντος (POIs) [1]

3.3 Περιβάλλοντα Χωρίς Υποδομές

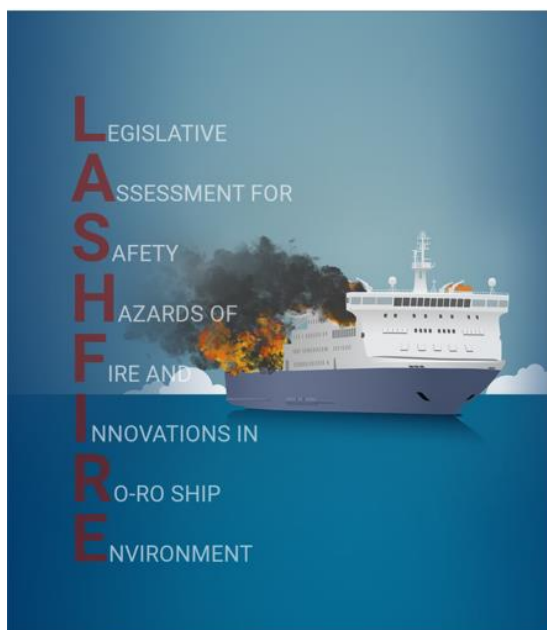
Δεν είναι δεδομένο ότι όλοι οι εσωτερικοί χώροι που θέλουμε να εισάγουμε στο σύστημα Anyplace έχουν τις κατάλληλες υποδομές, έτσι ώστε να μπορεί να υπολογιστεί η τοποθεσία ενός χρήστη σε αυτά. Υπάρχουν αρκετά εσωτερικά περιβάλλοντα που δεν διαθέτουν Wi-Fi σήματα ή άλλου είδους σημάτων όπως BLE, LED, UWB κ.α. για να μπορέσουμε να τα αξιοποιήσουμε. Αυτό κάνει τον υπολογισμό της εσωτερικής τοποθεσίας και πλοήγησης με την χρήση του Anyplace Navigator, αδύνατο.

Για τον λόγο αυτό προχωρήσαμε στην ανάπτυξη ενός συστήματος που χρησιμοποιεί υπολογιστική όραση για τον υπολογισμό της εσωτερικής τοποθεσίας, για χρήση σε χώρους χωρίς υποδομές. Το σύστημα αυτό, σε συνδυασμό με άλλες λειτουργίες,

αποτελούν την εφαρμογή SMAS (**Smart Alert System**) που υλοποιήθηκε με αρχικό στόχο την χρήση της σε πλοία, όπου τα Wi-Fi σήματα δεν είναι τόσο ευρέως διαθέσιμα. Η εφαρμογή περιγράφεται εκτενώς στο υποκεφάλαιο που ακολουθεί 3.4, καθώς και στο κεφάλαιο 5, «Υπολογισμός Εσωτερικής Τοποθεσίας με Χρήση Υπολογιστικής Όρασης».

3.4 Εφαρμογή SMAS

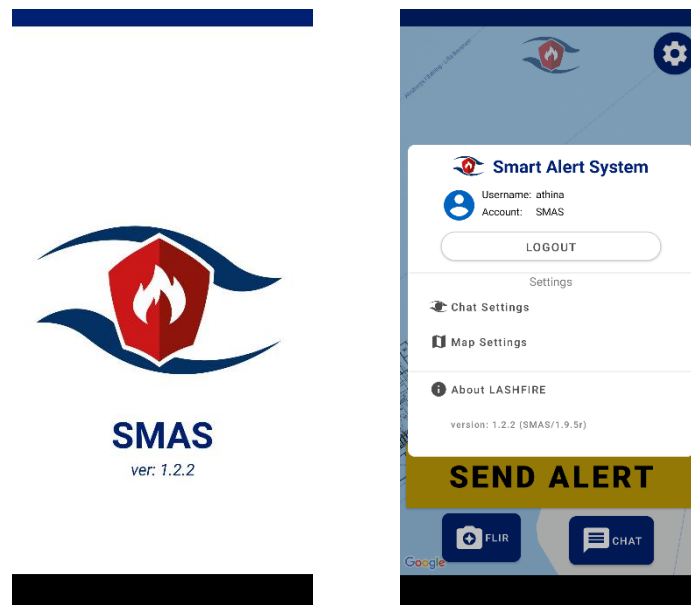
Η εφαρμογή SMAS αναπτύχθηκε στα πλαίσια του προγράμματος LASH FIRE [22] το οποίο στοχεύει στην μείωση πιθανοτήτων εκρήξεων πυρκαγιών σε πλοία. Πιο συγκεκριμένα, στόχος του προγράμματος είναι η ανάπτυξη έξυπνων τεχνολογιών οι οποίες μπορούν να χρησιμοποιηθούν για την αποτελεσματική απόσβεση πυρκαγιών στα αρχικά τους στάδια, καθώς και για έγκαιρη ενημέρωση του πληρώματος του πλοίου και των πυροσβεστών. Επίσης, οι τεχνολογίες αυτές θα πρέπει να παρέχουν ένα τρόπο εντοπισμού της τοποθεσίας άλλων ατόμων μέσα στο πλοίο και ένα κανάλι επικοινωνίας μέσα στην ίδια την εφαρμογή.



Σχήμα 3.4 : Η σημασία του ακρωνύμιου LASH FIRE [22]

Το SMAS αναπτύχθηκε για να ενισχύσει τις προσπάθειες του προγράμματος LASH FIRE και προσφέρει μια γκάμα από λειτουργίες οι οποίες ανταποκρίνονται στις απαιτήσεις του. Παρέχει την δυνατότητα εντοπισμού της τρέχουσας τοποθεσίας του χρήστη μέσα στο

πλοίο, καθώς και τις τοποθεσίες των άλλων ενεργών χρηστών. Αυτό γίνεται με την βοήθεια του αλγόριθμου που κάνει χρήση υπολογιστικής όρασης και νευρωνικών δικτύων. Επίσης, μπορεί να παρέχει οδηγίες πλοήγησης από ένα POI σε ένα άλλο καθώς και αναζήτηση τέτοιων POIs, ή αναζήτηση άλλων ατόμων. Δίνει άμεση πρόσβαση στην θερμική κάμερα FLIR και την ταχεία αποστολή ειδοποιήσεων σε περίπτωση έκτακτης ανάγκης. Τέλος, παρέχει την δυνατότητα αποστολής μηνυμάτων στους υπόλοιπους χρήστες.



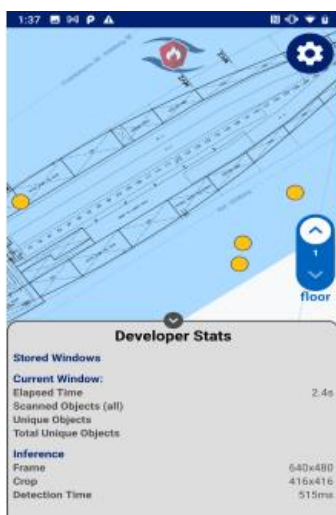
Σχήμα 3.5 : Οθόνη έναρξης (αριστερά) και ρυθμίσεις (δεξιά) της εφαρμογής SMAS

3.4.1 Anyplace και SMAS

Η αρχιτεκτονική της εφαρμογής SMAS βασίζεται πάνω στο σύστημα Anyplace, στο οποίο αναφερθήκαμε στο υποκεφάλαιο 3.2 «Ανατομία του Συστήματος Anyplace». Χρησιμοποιήθηκε η εφαρμογή διαδικτύου Architect για την εισαγωγή του αρχιτεκτονικού σχεδίου της κάτοψης ενός από των πλοίων που ανήκουν στο πρόγραμμα LASH FIRE. Σημειώθηκαν επίσης τα διάφορα σημεία ενδιαφέροντος πάνω στον χάρτη και έγινε η ένωση τους για την δημιουργία μονοπατιών, που αργότερα θα χρησιμοποιηθούν για την λειτουργία πλοήγησης. Η διαφορά του SMAS με τον μέχρι τώρα τρόπο λειτουργίας του Anyplace, είναι το ότι πλέον δεν χρειάζεται η συλλογή

σημάτων Wi-Fi για εντοπισμό της εσωτερικής τοποθεσίας, το οποίο ήταν εφικτό με την εφαρμογή Logger. Αντίστοιχα, αναπτύχθηκε μια νέα εκδοχή του Logger, αποκλειστικά για το SMAS, όπου χρησιμοποιείται για την αντιστοίχιση σταθερών αντικειμένων στον χώρο με τις γεωγραφικές τους συντεταγμένες.

Η μέθοδος υπολογισμού της εσωτερικής θέσης μέσα στο πλοίο αποτελείται από τρία στάδια. Αρχικά γίνεται η εκπαίδευση των νευρωνικών δικτύων με την βοήθεια δεδομένων που εξάχθηκαν από το σύστημα CVAT (βλ. υποκ. 2.4.1). Ακολούθως, γίνεται η διαδικασία του Logging, όπως περιεγράφηκε προηγουμένως, από τα άτομα τα οποία βρίσκονται πάνω στο πλοίο. Τελευταίο στάδιο είναι αυτό του εντοπισμού των γεωγραφικών συντεταγμένων του χρήστη που χρησιμοποιεί το SMAS, με την χρήση της υπολογιστικής όρασης και των προηγούμενων σταδίων.



Σχήμα 3.6 : Στιγμιότυπο οθόνης από την εφαρμογή Logger του συστήματος SMAS

3.4.2 Τεχνολογίες στο Σύστημα SMAS

Η εφαρμογή SMAS είναι γραμμένη στο Android X, στην γλώσσα προγραμματισμού Kotlin. Ακολουθεί το αρχιτεκτονικό μοτίβο MVVM το οποίο δημιουργεί μια ευέλικτη δομή στο σύστημα για μελλοντικές αλλαγές. Λόγω της συχνής επικοινωνίας που χρειάζεται να έχει με τον διακομιστή για να φέρνει τα νέα μηνύματα, αλλά και της βαριάς διαδικασίας της αναγνώρισης αντικειμένων, χρησιμοποιούνται τα coroutines, χαρακτηριστικό της γλώσσας Kotlin. Τα coroutines επιτρέπουν την ασύγχρονη εκτέλεση ενός κώδικα, έτσι ώστε το σύστημα να μην μένει στάσιμο ενώ εκτελείται μια λειτουργία που παίρνει αρκετό χρόνο.

Τα μοντέλα, οι χάρτες και οτιδήποτε άλλο χρειάζεται η εφαρμογή να κατεβάσει για να λειτουργήσει, βρίσκονται ανεβασμένα στο cloud. Αυτό μειώνει τον συνολικό χρόνο έναρξης της. Επίσης, χρησιμοποιείται το σύστημα διαχείρισης σχεσιακών βάσεων SQLite για την αποθήκευση των μηνυμάτων και των ειδοποιήσεων στον διακομιστή και τοπικά στο κινητό τηλέφωνο.

Κεφάλαιο 4

Υποσύστημα Ανταλλαγής Μηνυμάτων Βάσει Τοποθεσίας και Εργαλείο Αναζήτησης

4.1	Ιδέα	30
4.2	Διεπαφή Χρήστη για το Υποσύστημα Ανταλλαγής Μηνυμάτων	32
4.2.1	Φιλοσοφία Σχεδιασμού	32
4.2.2	Αναγνώριση Ομιλίας	35
4.2.3	Προσθήκη Φωτογραφιών	35
4.2.4	Αποστολή Τοποθεσίας και Ειδοποιήσεων	36
4.3	Εργαλείο Αναζήτησης	37

Το κεφάλαιο αυτό ασχολείται με το υποσύστημα ανταλλαγής μηνυμάτων με γνώση τοποθεσίας των χρηστών, καθώς και με το εργαλείο αναζήτησης, τα οποία αναπτύχθηκαν για να ενταχθούν στην εφαρμογή SMAS. Αρχικά παρουσιάζεται η ιδέα πίσω από την υλοποίηση του συστήματος επικοινωνίας και ακολούθως, αναλύεται εκτενώς η φιλοσοφία πίσω από τον σχεδιασμό της διεπαφής της. Έπειτα, με περισσότερες λεπτομέρειες περιγράφονται τα πιο σημαντικά στοιχεία της. Τέλος γίνεται μια σύντομη αναφορά στο εργαλείο αναζήτησης όπου χρησιμοποιείται στην αρχική οθόνη της εφαρμογής.

4.1 Ιδέα

Με την παρουσία χιλιάδων εφαρμογών επικοινωνίας στα συστήματα διανομής λογισμικού, πλέον ο κάθε χρήστης ενός κινητού τηλεφώνου μπορεί εύκολα να επικοινωνήσει με τον οποιονδήποτε, είτε στέλνοντας ένα μήνυμα είτε καλώντας τον. Όλη η επικοινωνία γίνεται χωρίς χρέωση, με μοναδική απαίτηση την σύνδεση στο διαδίκτυο.

Οι εφαρμογές αυτές δίνουν την δυνατότητα σύνδεσης δύο ατόμων που βρίσκονται στα δύο άκρα μιας χώρας, μιας ηπείρου καθώς και διηπειρωτικές διασυνδέσεις. Η τοποθεσία ενός χρήστη δεν επηρεάζει με ποιόν μπορεί ο συγκεκριμένος να επικοινωνήσει, αρκεί τα δύο άκρα της σύνδεσης (δηλαδή οι δύο χρήστες) να έχουν στην διάθεση τους την ίδια εφαρμογή.

Με τελικό στόχο την υλοποίηση ενός συστήματος για την διευκόλυνση της πλοήγησης των ατόμων που βρίσκονται σε πλοία με μεγάλες πιθανότητες έκρηξης πυρκαγιών, αποφασίσαμε να προχωρήσουμε στην παραχώρηση ενός καναλιού επικοινωνίας μέσα στο σύστημα αυτό, για τα άτομα τα οποία βρίσκονται στον ίδιο χώρο. Με τον τρόπο αυτό η επικοινωνία μπορεί να γίνει μέσα στο ίδιο σύστημα το οποίο χρησιμοποιείται για την πλοήγηση χωρίς την ανάγκη χρήσης μιας δευτερεύουσας εφαρμογής. Αυτό δίνει τα προνόμια της ταχύτερης επικοινωνίας, καλύτερης κατανόησης του περιεχομένου και των συμφραζόμενων και ευκολότερη πρόσβαση στις πληροφορίες που σχετίζονται με την πλοήγηση.

Η ιδέα είναι η ανάπτυξη ενός συστήματος ανταλλαγής μηνυμάτων βάσει τοποθεσίας. Δηλαδή, η ανταλλαγή μηνυμάτων να μην γίνεται μεταξύ δύο συγκεκριμένων χρηστών της εφαρμογής, αλλά το μήνυμα που αποστέλλει ένας χρήστης να φθάνει σε όλους τους υπόλοιπους χρήστες που βρίσκονται σε κοντινή απόσταση από αυτόν, βάσει και των ρυθμίσεων που θα επιλεχθούν. Έτσι, αποστολέας ενός μηνύματος είναι ένα άτομο ενώ παραλήπτες του πολλοί. Με τον τρόπο αυτό, σημαντικές και ουσιώδεις πληροφορίες διαδίδονται ευκολότερα και γρηγορότερα στο πλήθος των ατόμων που βρίσκονται από κοινού στον ίδιο χώρο.

Το σύστημα υλοποιεί τέσσερις μεθόδους αποστολής μηνυμάτων, καθορίζοντας ποια άτομα τα οποία βρίσκονται συνδεδεμένα με την εφαρμογή θα παραλάβουν το μήνυμα. Έχουν ως εξής:

- Πρώτη μέθοδος όπου όλοι οι χρήστες είναι παραλήπτες του μηνύματος.
- Δεύτερη μέθοδος όπου παραλήπτες είναι οι χρήστες που βρίσκονται στο ίδιο κατάστρωμα/ όροφο με τον αποστολέα.
- Τρίτη μέθοδος όπου παραλήπτες είναι οι Κ κοντινότεροι γείτονες του αποστολέα βάσει του γεωγραφικού πλάτους, γεωγραφικού ύψους και καταστρώματος/ ορόφου.

- Τέταρτη μέθοδος όπου το μήνυμα αποστέλλεται σε χρήστες που βρίσκονται στη μέγιστη απόσταση των 100 μέτρων από τον αποστολέα.

4.2 Διεπαφή Χρήστη για το Υποσύστημα Ανταλλαγής Μηνυμάτων

Με τον όρο διεπαφή εννοούμε το μέρος ενός συστήματος με το οποίο μπορεί να αλληλεπιδράσει ένας χρήστης. Είναι το σημείο επαφής χρήστη και συστήματος που επιτρέπει την αμφίδρομη επικοινωνία μεταξύ τους. Για την διασφάλιση της ευχαρίστησης του ατόμου αυτού, υπάρχουν κάποια κριτήρια που πρέπει να τηρούνται. Για παράδειγμα, η γλώσσα, τα σύμβολα και οι έννοιες που χρησιμοποιούνται από το σύστημα πρέπει να συσχετίζονται με τον πραγματικό κόσμο. Πρέπει να υπάρχει συνέπεια και τήρηση των προτύπων καθώς και ευελιξία χρήσης. Επίσης, ο σχεδιασμός του συστήματος πρέπει να γίνεται έτσι ώστε να αποτρέπονται τα πιθανά σφάλματα χρηστών, παρέχοντας έτσι περισσότερη σταθερότητα. Αυτοί και μερικοί άλλοι αποτελούν τους 10 κανόνες ευχρηστίας του Jakob Nielsen [27].

Σε αυτό εδώ το υποκεφάλαιο παρουσιάζεται η διεπαφή χρήστη που αναπτύξαμε για το σύστημα συνομιλιών που εντάχθηκε στην εφαρμογή SMAS. Η λειτουργία αυτή είναι προσβάσιμη με το πάτημα ενός κουμπιού που βρίσκεται στην αρχική οθόνη της εφαρμογής. Κάποιες από τις δυνατότητες που παρέχει το υποσύστημα είναι η αποστολή μηνυμάτων κειμένου, αποστολή φωτογραφιών, τοποθεσίας και ειδοποιήσεων και η μετατροπή φωνητικού κειμένου σε γραπτό λόγο.

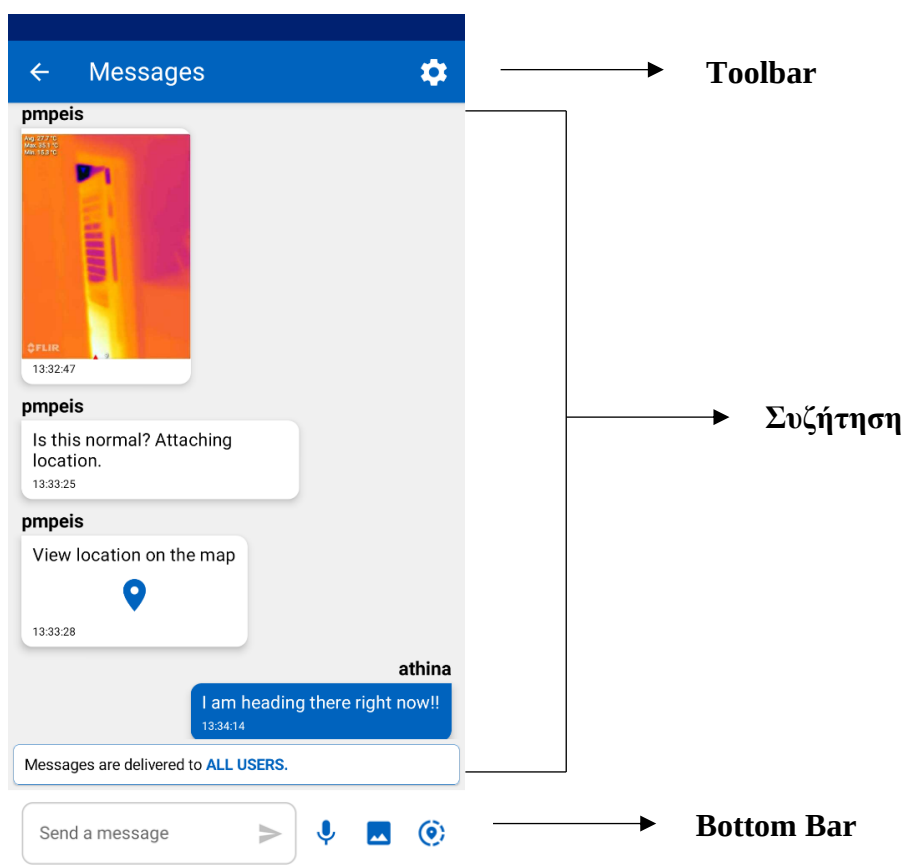
Για την επικοινωνία με τα μοντέλα δεδομένων και την πρόσβαση στις απαντήσεις από τον διακομιστή, χρησιμοποιήθηκε το μοτίβο αρχιτεκτονικής Model-View-ViewModel που αναλύεται στο υποκεφάλαιο 2.4.2, «Τεχνολογίες Frontend». Για την ανάπτυξη των οθονών που βλέπει ο χρήστης χρησιμοποιήθηκε η γλώσσα προγραμματισμού Kotlin και το εργαλείο Android Compose, για τα οποία γίνεται επίσης αναφορά στο προαναφερθέν υποκεφάλαιο.

4.2.1 Φιλοσοφία Σχεδιασμού

Το υποσύστημα ανταλλαγής μηνυμάτων σχεδιάστηκε έτσι ώστε να ακολουθεί την θεματική που υπάρχει σε όλη την εφαρμογή όπου κύριο χρώμα είναι το μπλε.

Αποτελείται από 3 σημαντικά μέρη τα οποία είναι τα εξής: το Toolbar, το κύριο μέρος όπου φαίνεται η συζήτηση και το Bottom Bar. Στο σχήμα 4.1 μπορούμε να δούμε τα τρία αυτά μέρη.

Το Toolbar βρίσκεται στο πάνω μέρος της οθόνης, έχοντας δύο εικονίδια. Το εικονίδιο με το βελάκι που βρίσκεται αριστερά μεταβιβάζει τον χρήστη στη προηγούμενη οθόνη που ήταν ενεργή, σύμφωνα με το ιστορικό πλοήγησης, τερματίζοντας την λειτουργία αυτής. Το εικονίδιο που βρίσκεται στα δεξιά ανοίγει μια νέα οθόνη όπου βρίσκονται οι ρυθμίσεις που είναι σχετικές με τον διακομιστή που εξυπηρετεί την συνομιλία.



Σχήμα 4.1 : Στιγμιότυπο οθόνης υποσυστήματος ανταλλαγής μηνυμάτων

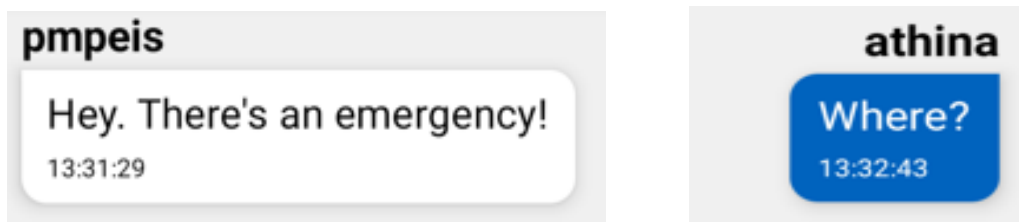
Το κύριο μέρος της οθόνης είναι εκεί όπου φαίνονται τα απεσταλμένα μηνύματα καθώς και τα μηνύματα που λήφθηκαν. Αυτό το μέρος της οθόνης είναι κυλιόμενο, με τελευταίο στοιχείο του το πιο πρόσφατο μήνυμα.

Τα τέσσερα είδη μηνυμάτων που υπάρχουν είναι τα εξής:

- Μηνύματα κειμένου,

- Εικόνες που κωδικοποιούνται σε base64 μορφή πριν αποσταλούν και αποκωδικοποιούνται πίσω στην κανονική μορφή τους όταν παραληφθούν,
- Τοποθεσία ενός χρήστη με γεωγραφικές συντεταγμένες,
- Ειδοποίηση για άμεση δράση σε συνδυασμό με τοποθεσία.

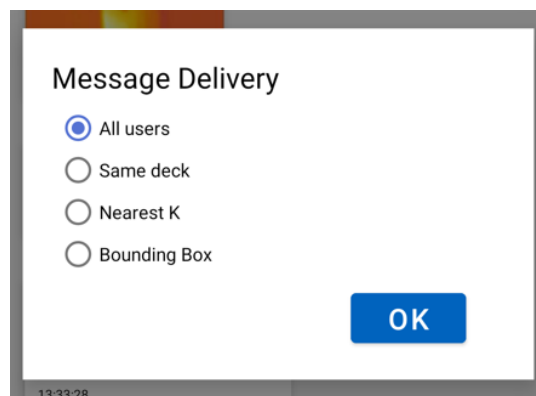
Πάνω από κάθε μήνυμα φαίνεται το όνομα του αποστολέα ενώ μέσα στο ορθογώνιο κουτάκι (βλ. σχήμα 4.2) όπου βρίσκεται το ίδιο, φαίνεται η ώρα αποστολής του (και η ημερομηνία όταν αλλάξει η μέρα).



Σχήμα 4.2 : Αριστερά φαίνεται ένα εισερχόμενο μήνυμα ενώ στα δεξιά ένα απεσταλμένο

Στο Bottom Bar υπάρχουν τα κατάλληλα στοιχεία έτσι ώστε ο χρήστης να μπορέσει να αποστείλει ένα μήνυμα οποιουδήποτε είδους. Υπάρχει το TextBox για την εισαγωγή κειμένου με το πληκτρολόγιο και το εικονίδιο μικροφώνου για εγγραφή φωνής, όπου μετατρέπεται σε κείμενο που εμφανίζεται και πάλι στο TextBox. Επίσης, υπάρχει το εικονίδιο αρχείων για την επιλογή φωτογραφιών και το τελευταίο εικονίδιο στη σειρά όπου είναι για αποστολή της τρέχουσας τοποθεσίας του χρήστη.

Πάνω από αυτά τα στοιχεία, υπάρχουν, σε μορφή λεζάντας, πληροφορίες για το που αποστέλλονται τα μηνύματα του χρήστη, βάσει το τεσσάρων μεθόδων αποστολής που αναφέρθηκαν πιο πάνω (βλ. υποκ. 4.1). Το ορθογώνιο αυτό έχει την δυνατότητα κλικ και επιλογής άλλης μεθόδου αποστολής.



Σχήμα 4.3 : Επιλογή για το ποιος θα παραλάβει το μήνυμα

4.2.2 Αναγνώριση Ομιλίας

Η αναγνώριση ομιλίας ενεργοποιείται με το πάτημα του κουμπιού που φαίνεται στο σχήμα 4.4. Λειτουργεί με και χωρίς σύνδεση στο διαδίκτυο και αυτόματα μετατρέπει τον προφορικό λόγο σε γραπτό.



Σχήμα 4.4 : Κουμπί αναγνώρισης ομιλίας

Κάνει χρήση της κλάσης SpeechRecognizer από την βιβλιοθήκη αντικειμένων της Java. Αν και όπως προαναφέρθηκε μπορεί να λειτουργήσει χωρίς σύνδεση στο διαδίκτυο, η υλοποίηση αυτή πιθανών να κάνει χρήση απομακρυσμένων διακομιστών για την αναγνώριση. Αυτό συνεπάγεται στην μη δυνατή συνεχόμενη αναγνώριση ομιλίας που θα σήμαινε και αρκετή κατανάλωση της μπαταρίας της συσκευής. Η δυνατότητα αναγνώρισης είναι μια πρόταση ανά φορά.

Στην συγκεκριμένη υλοποίηση, η γλώσσα που καταλαβαίνει το σύστημα είναι τα Αγγλικά. Δίνεται όμως η δυνατότητα προσθήκης περισσότερων γλωσσών, έτσι ώστε να αναγνωρίζει ομιλίες από ανθρώπους διαφορετικών εθνικοτήτων.

4.2.3 Προσθήκη Φωτογραφιών

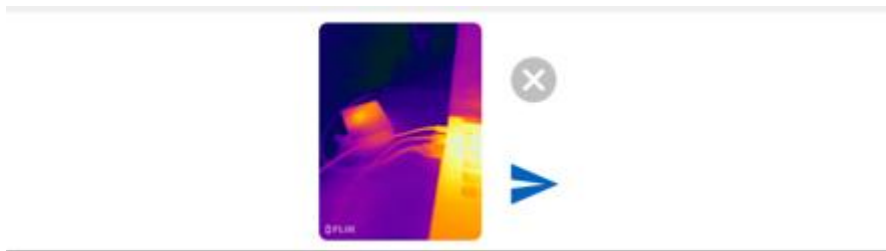
Η αποστολή φωτογραφιών είναι αναπόσπαστο μέρος τέτοιων συστημάτων επικοινωνίας. Το σύστημα SMAS θα χρησιμοποιείται κυρίως από πυροσβέστες και άτομα που βρίσκονται πάνω σε πλοία, έτσι ώστε σε περίπτωση κινδύνων να μπορέσουν να ενημερώσουν τους υπόλοιπους. Αφού μέσω της εφαρμογής τους δίνεται η δυνατότητα πρόσβασης σε θερμική κάμερα για να πάρουν φωτογραφίες, αναπόφευκτο είναι να θέλουν να τις στείλουν στους συνεργάτες τους για να επισημάνουν μια τοποθεσία, ένα αντικείμενο, μια καταστροφή.



Σχήμα 4.5 : Κουμπί επιλογής φωτογραφίας

Με το πάτημα του κουμπιού που φαίνεται στο σχήμα 4.5, δίνεται η ευκαιρία στον χρήστη να επιλέξει μόνο μια φωτογραφία από τις ήδη αποθηκευμένες του. Με την επιλογή της, το Bottom Bar αλλάζει όψη και μετατρέπεται σε αυτό που φαίνεται στο σχήμα 4.6. Με την αλλαγή αυτή ο χρήστης μπορεί να δει την επιλεγμένη φωτογραφία, να ακυρώσει την επιλογή της ή να την στείλει στην συνομιλία.

Οι φωτογραφίες για να σταλούν στον διακομιστή που εξυπηρετεί το chat και μετά να σταλούν στους παραλήπτες τους, πρέπει πρώτα να κωδικοποιηθούν σε μορφή base64. Το base64 είναι μια ομάδα κωδικοποιήσεων ψηφιακών δεδομένων σε κείμενο. Κάθε byte ενός αρχείου αντιστοιχεί σε ένα υποσύνολο από χαρακτήρες ASCII, οι οποίοι είναι εκτυπώσιμοι. Με την παραλαβή των φωτογραφιών γίνεται η αποκωδικοποίηση τους πίσω στον τύπο φωτογραφίας που ήταν αρχικά και εμφανίζονται κανονικά στον χρήστη.



Σχήμα 4.6 : Προβολή επιλεγμένης φωτογραφίας

Για λόγους επίδοσης της διεπαφής ανταλλαγής μηνυμάτων, οι φωτογραφίες οι οποίες στέλνονται επιτυχώς, συμπιέζονται πριν εμφανιστούν στον χρήστη. Αυτό γίνεται γιατί οι φωτογραφίες αποτελούν μεγάλα αρχεία, ειδικά όταν ληφθούν με κάμερα υψηλής ανάλυσης, και καθυστερούν τον συνολικό χρόνο φόρτωσης και κύλισης της συνομιλίας. Παρέχετε όμως η δυνατότητα στον χρήστη να πατήσει επάνω και να δει την κανονική φωτογραφία, πριν την συμπίεση αλλά και να την μεγεθύνει αν θέλει να παρατηρήσει τις λεπτομέρειες τις.

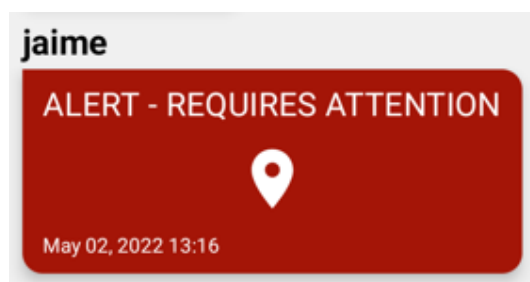
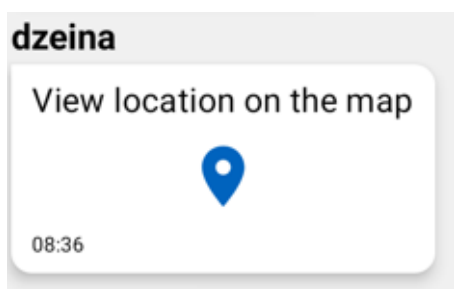
4.2.4 Αποστολή Τοποθεσίας και Ειδοποιήσεων

Με το πάτημα του τελευταίου κουμπιού (βλ. σχήμα 4.7) στο Bottom Bar, δίνεται η δυνατότητα στον χρήστη να αποστείλει την τρέχουσα τοποθεσία του. Αυτό είναι αναγκαίο σε περιπτώσεις όπου ο χρήστης χρειάζεται βοήθεια σε ένα συγκεκριμένο χώρο. Στέλνοντας τις γεωγραφικές του συντεταγμένες στους συνεργάτες του, αυτοί μπορούν με ευκολία να πλοηγηθούν στο σημείο αυτό.



Σχήμα 4.7 : Κουμπί αποστολής τρέχουσας τοποθεσίας

Πριν αποσταλεί η τοποθεσία, εμφανίζεται στην οθόνη ένα μήνυμα τύπου ειδοποίησης όπου ζητά από τον χρήστη να επιβεβαιώσει ότι θέλει να στείλει την θέση του. Τα μηνύματα τύπου τοποθεσίας φαίνονται στην συνομιλία όπως το σχήμα 4.8. Το εικονίδιο που βρίσκεται στο μέσο του ορθογωνίου είναι με δυνατότητα κλικ. Όταν πατηθεί, η τρέχουσα οθόνη τερματίζει και ο χρήστης μεταφέρεται στην οθόνη όπου είναι ο χάρτης του πλοίου. Η λειτουργία αυτή, εκτός από μετάβαση μεταξύ οθονών, μεταφέρει το γεωγραφικό πλάτος και μήκος σε αυτή την οθόνη, εμφανίζοντας στον χάρτη την ακριβή τοποθεσία που αντιστοιχεί πάνω στο πλοίο.



Σχήμα 4.8 : Μήνυμα τύπου τοποθεσίας (αριστερά) και τύπου ειδοποίησης (δεξιά)

Τα μηνύματα τύπου ειδοποίησης είναι παρόμοια με τα μηνύματα τύπου τοποθεσίας, με διαφορές στα χρώματα και στο κείμενο που αναγράφεται μέσα στο κουτί. Αυτού του είδους μηνύματα δεν μπορούν να σταλούν μέσω της διεπαφής του chat, αλλά μέσω της αρχικής οθόνης που βρίσκεται ο χάρτης του πλοίου, πατώντας το κουμπί SEND ALERT.

4.3 Εργαλείο Αναζήτησης

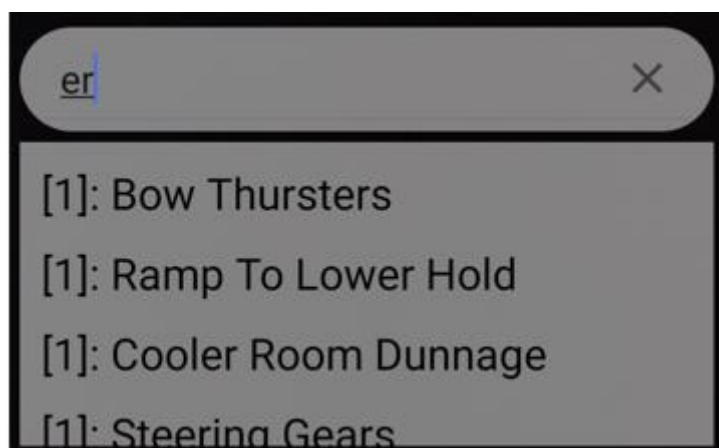
Για σκοπούς διευκόλυνσης των χρηστών της εφαρμογής, εκτός από το κανάλι επικοινωνίας τους παρέχουμε και την λειτουργία αναζήτησης. Από την στιγμή που η τοποθεσία κάθε χρήστη σε σύνδεση είναι γνωστή μέσω του χάρτη που φαίνεται στην αρχική οθόνη, τότε και η αναζήτηση τους βάση ονόματος είναι εφικτή. Εκτός αυτού,

επιτρέπεται και η αναζήτηση σημείων ενδιαφέροντος πάνω στο πλοίο καθώς και άλλων σταθερών στοιχείων.



Σχήμα 4.9 : Το εργαλείο αναζήτησης

Όταν ο χρήστης αρχίσει να πληκτρολογεί ένα όνομα, του παρουσιάζονται επιλογές που αντιστοιχούν στο κείμενο που πληκτρολογεί. Είναι ένα είδους αυτόματης συμπλήρωσης όπου τα διαθέσιμα, αποθηκευμένα ονόματα εμφανίζονται μόνο εάν έχουν το κείμενο του χρήστη σαν υποσυμβολοσειρά τους. Εάν δεν γίνει καμία αντιστοίχιση τότε ο χρήστης ενημερώνεται ανάλογα.



Σχήμα 4.10 : Αυτόματη συμπλήρωση από τις διαθέσιμες επιλογές

Με το που ο χρήστης διαλέξει μια από τις προτεινόμενες επιλογές, εμφανίζεται η τοποθεσία της πάνω στον χάρτη, στιγματισμένη με έναν marker. Ο χρήστης μπορεί να επιλέξει να του δοθεί μια διαδρομή για πλοήγηση ως προς την τοποθεσία, από την δική του τρέχουσα. Η πλοήγηση γίνεται μόνο μέσω σημείων ενδιαφέροντος (POIS) που είναι γνωστά μέσα στο πλοίο.

Για όλη αυτή την διαδικασία, γίνεται χρήση του Google Maps API και όλων των δυνατοτήτων που έρχονται με αυτό. Για παράδειγμα, η επισήμανση τοποθεσίας με marker, η χρήση polylines και η κλάση LatLng για τις γεωγραφικές συντεταγμένες.

Κεφάλαιο 5

Υπολογισμός Εσωτερικής Τοποθεσίας με Χρήση Υπολογιστικής Όρασης

5.1	Επισκόπηση Αλγόριθμου	39
5.2	Φάση Εκπαίδευσης	40
5.2.1	Μοντέλο LASHCO	42
5.3	Φάση Logging	44
5.4	Φάση Εντοπισμού Εσωτερικής Τοποθεσίας	45
5.5	Παραδείγματα Χρήσης	46

Στο κεφάλαιο αυτό περιγράφουμε τον αλγόριθμο του συστήματος SMAS, ο οποίος μπορεί να υπολογίζει την τοποθεσία σε εσωτερικά περιβάλλοντα με χρήση υπολογιστικής όρασης. Βλέπουμε αναλυτικά τις τρεις φάσεις που τον απαρτίζουν και στο τέλος παρουσιάζουμε μερικά παραδείγματα χρήσης του στους χώρους του Πανεπιστημίου Κύπρου.

5.1 Επισκόπηση Αλγόριθμου

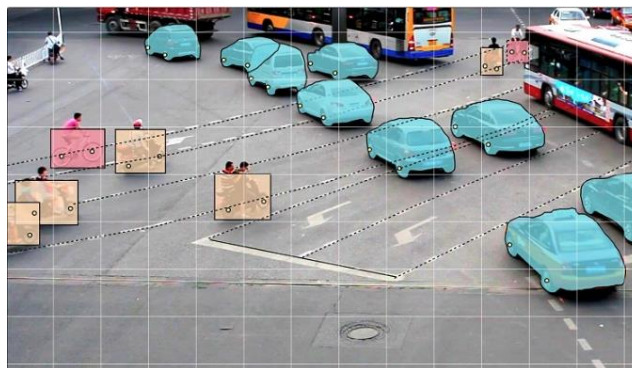
Ο υπολογισμός της εσωτερικής τοποθεσίας σε πλοία όπου τα σήματα Wi-Fi είναι εξασθενημένα λόγω των μεταλλικών υποδομών είναι αρκετά δύσκολος έως αδύνατος. Όχι μόνο αυτό, αλλά τα Wi-Fi σήματα δεν είναι διαθέσιμα σε όλους τους χώρους ενός πλοίου όπως είναι σε κτήρια στα οποία είναι περισσότερο αναγκαία. Για τον λόγο αυτό, αναπτύχθηκε ένας νέος αλγόριθμος εντοπισμού της εσωτερικής θέσης, κάνοντας χρήση των φυσικών αντικειμένων σε ένα χώρο και τις θέσεις τους μέσα σε αυτόν. Χρησιμοποιεί ένα συνδυασμό από μηχανική μάθηση και υπολογιστική όραση. Πιο συγκεκριμένα, ο αλγόριθμος αποτελείται από τρεις φάσεις. Στην πρώτη φάση γίνεται η εκπαίδευση των

νευρωνικών δικτύων που θα πραγματοποιούν την αναγνώριση των αντικειμένων. Στην δεύτερη φάση γίνεται η διαδικασία του Logging, η οποία γίνεται από άτομα που έχουν πρόσβαση πάνω στο πλοίο και τέλος, στην τρίτη φάση μπορεί να γίνει ο εντοπισμός της εσωτερικής τοποθεσίας.

Είναι σημαντικό να αναφερθεί ότι ο πιο πάνω αλγόριθμος μπορεί να προσαρμοστεί έτσι ώστε να αναγνωρίζει κι άλλα αντικείμενα, εκτός από αντικείμενα που βρίσκονται μέσα σε ένα πλοίο. Δηλαδή, ο υπολογισμός εσωτερικής θέσης δεν περιορίζεται μόνο σε πλοία. Αρκεί φυσικά να γίνει η κατάλληλη εκπαίδευση του, ανάλογα με το κτήριο στο οποίο επιθυμούμε να λειτουργεί. Επίσης, είναι απαραίτητο ο χρήστης του συστήματος SMAS, το οποίο αξιοποιεί τον αλγόριθμο αυτόν, να διαθέτει ένα κινητό τηλέφωνο με κάμερα και επαρκή χώρο αποθήκευσης.

5.2 Φάση Εκπαίδευσης

Η φάση της εκπαίδευσης είναι η πιο χρονοβόρα από όλες, αφού για να μπορέσουν τα νευρωνικά δίκτυα να αναγνωρίζουν τις διάφορες κλάσεις (αντικείμενα) που επιθυμούμε, απαιτούν μεγάλο όγκο δεδομένων εισόδου. Αυτός ο όγκος δεδομένων εξάχθηκε με την βοήθεια του συστήματος CVAT, στο οποίο καταχωρήσαμε συνολικά 15 βίντεο από το εσωτερικό του πλοίου, διάφορων διαρκειών. Έπειτα, περάσαμε όλα τα βίντεο ένα-ένα, ανά μερικά καρέ την φορά για να επισημάνουμε σε αυτά την κλάση του κάθε ορατού αντικειμένου. Αποφασίσαμε ότι δεν θα εκπαιδεύαμε το δίκτυο έτσι ώστε να μπορεί να αναγνωρίζει ανθρώπους ή αντικείμενα που πιθανών να μετακινούνταν (όπως αυτοκίνητα), γιατί αυτό δεν θα βοηθούσε καθόλου στην φάση της εύρεσης τοποθεσίας μέσα στο πλοίο. Η όλη διαδικασία διήρκεσε τέσσερις εβδομάδες, με την βοήθεια πέντε ατόμων και την συνεχή προσθήκη κλάσεων για τον εμπλουτισμό του μοντέλου.

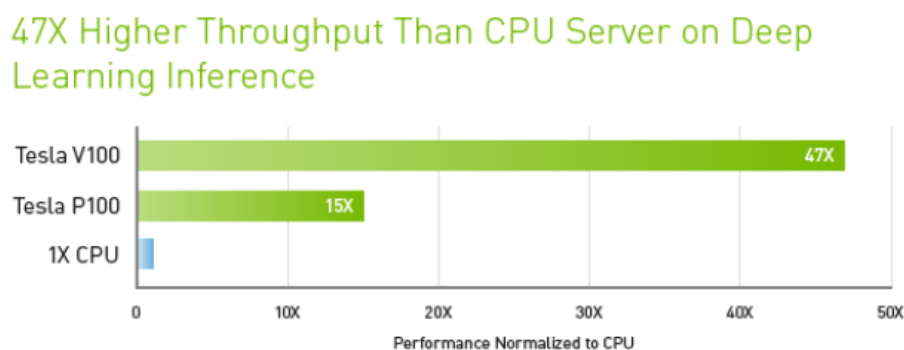


Σχήμα 5.1 : Παράδειγμα διαδικασίας επισήμανσης στο εργαλείο CVAT [9]

Ακολούθως, τα δεδομένα που εξάχθηκαν από το CVAT στάλθηκαν στο σύστημα YOLO. Τα νευρωνικά δίκτυα του εκπαιδεύτηκαν με αυτά τα δεδομένα, για να μπορούν αργότερα να αναγνωρίζουν, σε πραγματικό χρόνο, αντικείμενα μέσα στον χώρο με την χρήση υπολογιστικής όρασης. Με την ολοκλήρωση της εκπαίδευσης μετατρέψαμε το παραγόμενο μοντέλο σε TensorFlow Lite. Έτσι, μπορέσαμε να το φορτώσουμε σε κινητά τηλέφωνα όπου οι διαθέσιμοι πόροι είναι πολύ λιγότεροι από αυτούς στα υπολογιστικά συστήματα που χρησιμοποιούσαμε για την εκπαίδευση.

Για να μικρύνει το μέγεθος του δικτύου, είχαμε την επιλογή να εφαρμόσουμε την μέθοδο του quantization κατά την διάρκεια της εκπαίδευσης. Η μέθοδος αυτή έχει να κάνει με το πόσα bits αποτελούνται οι αριθμοί float. Υπό κανονικές συνθήκες, όλες οι πράξεις FPN (Floating Point Number) γίνονται με 32-bits αριθμούς. Όμως, με το quantization σε float16 καταφέραμε να μετατρέψουμε τους αριθμούς αυτούς σε μόλις 16-bits. Αυτό είχε ως αποτέλεσμα την μείωση του μεγέθους του παραγόμενου μοντέλου κατά 50% σε σχέση με τα μοντέλα που είχαν 32-bit αριθμούς. Με την μέθοδο του quantization, η ακρίβεια του μοντέλου μειώνεται ελάχιστα, σε σημείο που δεν επηρεάζει καθόλου τα αποτελέσματα που έχει αργότερα όταν εφαρμοστεί.

Το CVAT, καθώς και ο αλγόριθμος μηχανικής μάθησης, τρέχουν σε διακομιστή ο οποίος έχει ενσωματωμένη την κάρτα GPU NVIDIA Tesla V100 [29]. Μια κάρτα σχεδιασμένη για να επιταχύνει αλγόριθμους τεχνητής νοημοσύνης, βαριές διεργασίες που απαιτούν πολλούς πόρους, διεργασίες σχετικές με την επιστήμη δεδομένων και γραφικά. Αυτή η κάρτα μας επέτρεψε την μείωση του συνολικού χρόνου που χρειάστηκε το δίκτυο για να εκπαιδευτεί με τα δεδομένα που του παρείχαμε, από βδομάδες σε μόλις μερικές ώρες. Επίσης, με την επεξεργασία των δεδομένων τοπικά, δεν χρειάστηκε να αντιμετωπίσουμε την οποιαδήποτε συμφόρηση σε δίκτυο που πιθανόν να υπήρχε.



Σχήμα 5.2 : Σύγκριση των Tesla V100, Tesla P100 και CPU στην απόδοση τους σε εργασίες σχετικές με Deep Learning [29]

5.2.1 Μοντέλο LASHCO

Το μοντέλο που παράχθηκε με την βοήθεια των βίντεο από το εσωτερικό του πλοίου ονομάστηκε LASHCO και αποτελείται συνολικά από 99 κλάσεις αντικειμένων. Παρουσιάζονται στον πίνακα που ακολουθεί.

sign	deskInfo	doorRed	specificPlaygroundH use
signTruck	slotMachine	wallFireExtinguisherEmb edded	restaurantTheatreChai rs
signGather	benchWood	wallFireExtinguisherEmb eddedOCR	handleIron
signExitPersonLett ers	announcementSp eaker	emergencyEscapeRedSig n	wallCorridorCabinLigh t
wallLettersOCR	lifesaverRing	fireExtinguisherBoxed	specificMap
wallNumberOCR	poleUpperDeck	redLineXsingle	wallHealthKitOrange
signRestaurant	finUpperDeck	doorElevator	floorPatternCorridor
stairs	lightUpperDeck	ramp	signExitLettersOnly
stairsBlue	truck	signCCTV	signExitPersonArrowD ownLeft
stairsBlueIron	redLineXdouble	specificEngineroomSigns	signExitPersonArrowD ownRight
stairsUpperDeck	pipeWhite	wallPlugBlue	signExitPersonArrowU pLeft
trashcanRed	phoneBlack	signPower	signExitPersonArrowU pRight
trashcanMulticolou r	monitor	wallBoxGrey	signExitPersonArrowL eft

trashcanIron	boilerRed	signMultiple	signExitPersonArrowRight
trashcanYellow	emergencyFirePumpBlue	wallCapWhiteOCR	signArrow
wall	door	wallBoxGreyButtons	signNumberOCR
wallFirehose	doorRedIron	ceilingParking	specificTankDeck
wallFireExtinguisher	doorBlue	floorPatternReception	wallSingleDigitOCR
wallEmergencyButtonRed	doorBlueOCR	signLetterOCR	signLifeSaver
wallEmergencyBottleGreen	doorBlueWindow	trashcanInterior	signFireSafetyLever
restaurantDivider	doorWood	handleDoorBlueInterior	signFireGlassBreak
restaurantTableRound	doorRedOCR	stairsInterior	signFireExtinguisher
restaurantChair	doorBarred	restaurantChairLegs	signBathroomDisabled
handleWood	wallPatternRedWhite	specificRestaurantFreezer	symbolNoSmoking
handleBlue	wallPatternYellowBlack	deckOutsideFence	

Πίνακας 5.1 : Οι κλάσεις του μοντέλου LASHCO

Οι κατηγορίες έχουν ως εξής:

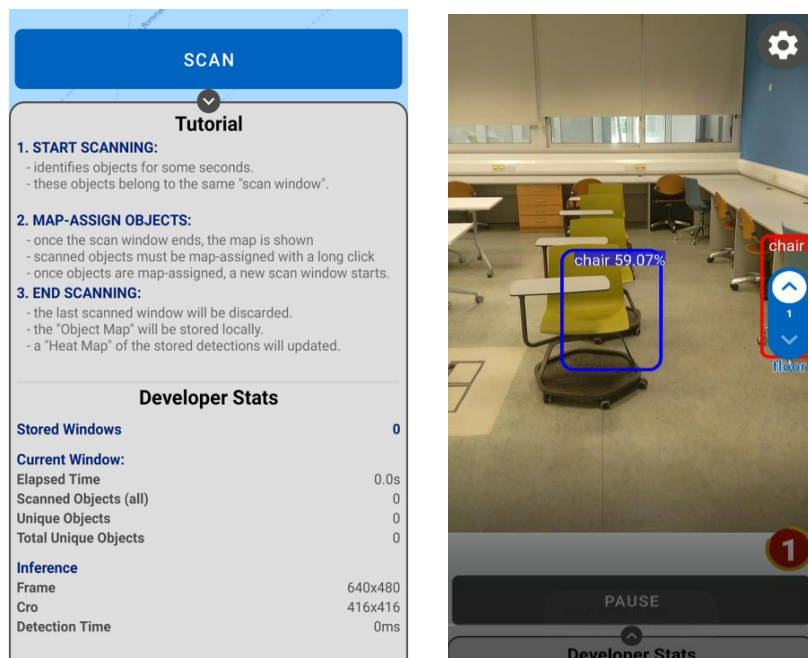
- Άσπρο: πινακίδες
- Κίτρινο: πόρτες
- Πράσινο: αντικείμενα στο εστιατόριο του πλοίου
- Κόκκινο: τοίχοι, πατώματα, οροφές και αντικείμενα που βρίσκονται σε αυτά
- Γκρι: σκάλες
- Μπλε: λοιπά αντικείμενα

5.3 Φάση Logging

Στην φάση αυτή γίνεται χρήση της εφαρμογής CV Logger, η οποία χρησιμοποιεί υπολογιστική όραση σε συνδυασμό με το παραγόμενο μοντέλο. Στα πλαίσια του προγράμματος LASH FIRE χρησιμοποιείται από το πλήρωμα του πλοίου από το οποίο πάρθηκαν τα βίντεο εκπαίδευσης.

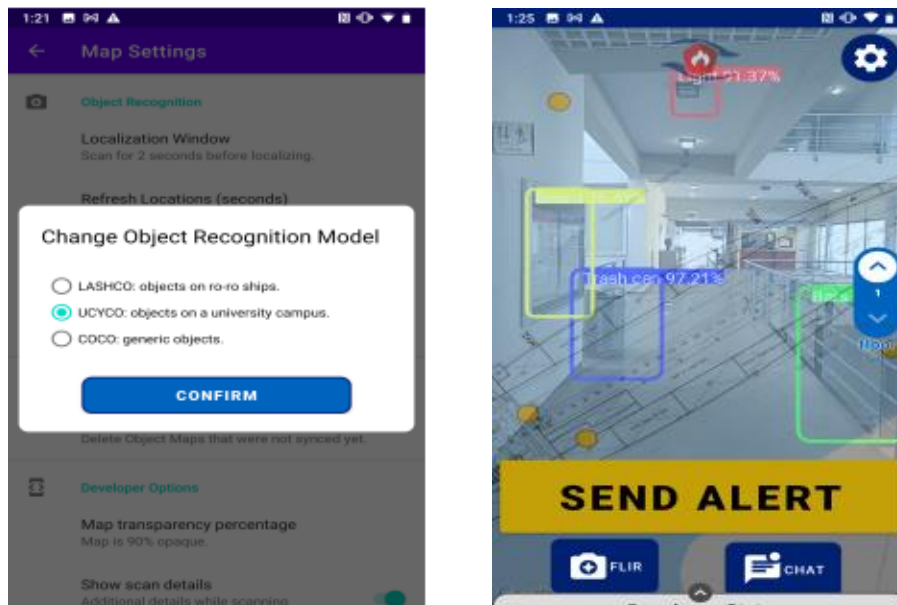
Αρχικά, τα άτομα αυτά πρέπει να έχουν εγκατεστημένη την εφαρμογή CV Logger σε κινητά τηλέφωνα με κάμερα στο πίσω μέρος. Ακολουθώντας, καθώς βρίσκονται μέσα στο πλοίο και σε ένα χώρο που θέλουν να χαρτογραφήσουν, ενεργοποιούν την εφαρμογή και ξεκινούν την λειτουργία του scanning. Η λειτουργία αυτή ενεργοποιεί την κάμερα του κινητού τηλεφώνου και για κάποια δευτερόλεπτα αναγνωρίζει αντικείμενα μέσα στον χώρο, σύμφωνα με το μοντέλο που επιλέχθηκε. Νοείται ότι ο χρήστης στέκετε σε λογική απόσταση από τα αντικείμενα-στόχους που επιθυμεί το σύστημα να αναγνωρίσει.

Όταν τελειώσει το χρονικό πλαίσιο για αναγνώριση, εμφανίζεται ο χάρτης του πλοίου στον χρήστη. Ο ίδιος πρέπει με κλικ επάνω του να επισημάνει την τοποθεσία στο πλοίο όπου είναι τα αντικείμενα που μόλις αναγνωρίστηκαν. Μετά από αυτό, η λειτουργία του scanning ξανά ξεκινάει. Με το τέλος της, αποθηκεύεται ο νέος χάρτης και ο θερμικός χάρτης με τις νέες ανιχνεύσεις ανανεώνεται.



Σχήμα 5.3: Στιγμιότυπα οθόνης από την εφαρμογή CV Logger

Εκτός από το μοντέλο LASHCO, υπάρχουν ακόμη δύο τα οποία μπορούν να χρησιμοποιηθούν στην φάση του logging για χαρτογράφηση κτηρίων. Το πρώτο είναι το μοντέλο COCO, το οποίο αποτελείται από 80 κατηγορίες και 1.5 εκατομμύρια περιπτώσεις αντικειμένων που μπορεί να αναγνωρίσει. Το δεύτερο είναι το UCYCO, ένα μικρού μεγέθους μοντέλο που είναι ειδικά για τους χώρους του Πανεπιστημίου Κύπρου.



Σχήμα 5.4 : Τα τρία μοντέλα που μπορεί να επιλέξει ο χρήστης (αριστερά) και το μοντέλο UCYCO σε εφαρμογή στο Π.Κ (δεξιά)

5.4 Φάση Εντοπισμού Εσωτερικής Τοποθεσίας

Μετά το τέλος της φάσης του logging, τα αντικείμενα που αναγνωρίστηκαν μαζί με τις θέσεις τους (αποτυπώματα) πάνω στον χάρτη ενώνονται σε ένα πίνακα $N \times M$ διαστάσεων. Η διάσταση N είναι ο αριθμός των μοναδικών αποτυπωμάτων και M ο συνολικός αριθμός των αντικειμένων που αναγνωρίστηκαν με την υπολογιστική όραση.

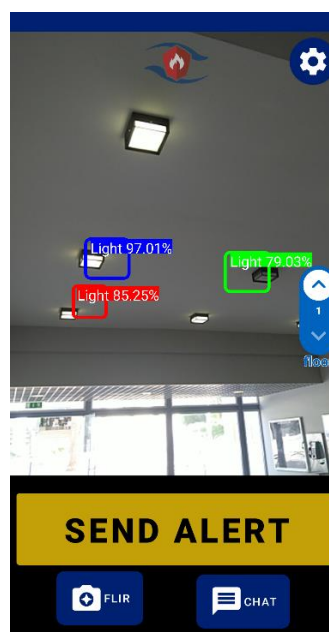
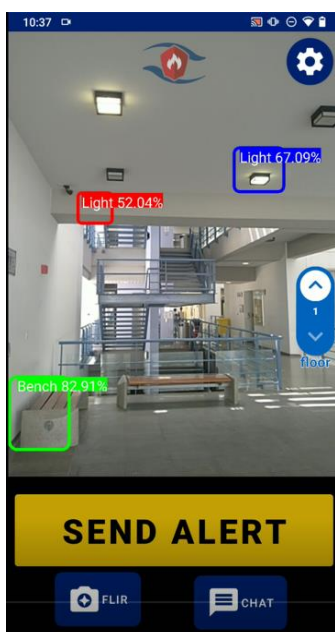
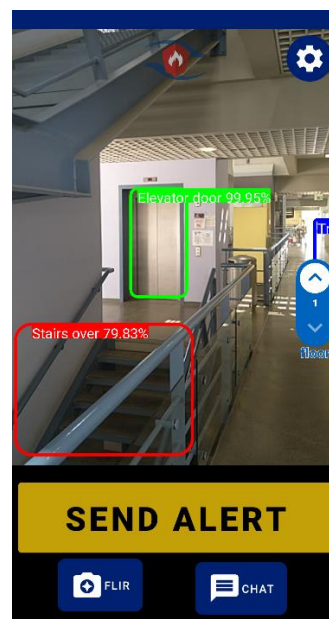
Ο υπολογισμός της εσωτερικής τοποθεσίας γίνεται ως εξής: ο χρήστης περιφέρεται μέσα στο πλοίο με την εφαρμογή SMAS ενεργή. Καθώς κινείται, γίνεται αναγνώριση των αντικειμένων γύρω του με την βοήθεια της υπολογιστικής όρασης και του κατάλληλου μοντέλου. Ο χρήστης συγκρίνει τα αποτυπώματα που λαμβάνει την στιγμή εκείνη με τον πίνακα $N \times M$ που υπάρχει ήδη, για να βρει το πιο κοντινό ταίριασμα. Αυτό είναι εφικτό

με τους αλγόριθμους KNN και WKNN. Το αποτέλεσμα του αλγόριθμου είναι η θέση του χρήστη μέσα στο πλοίο.

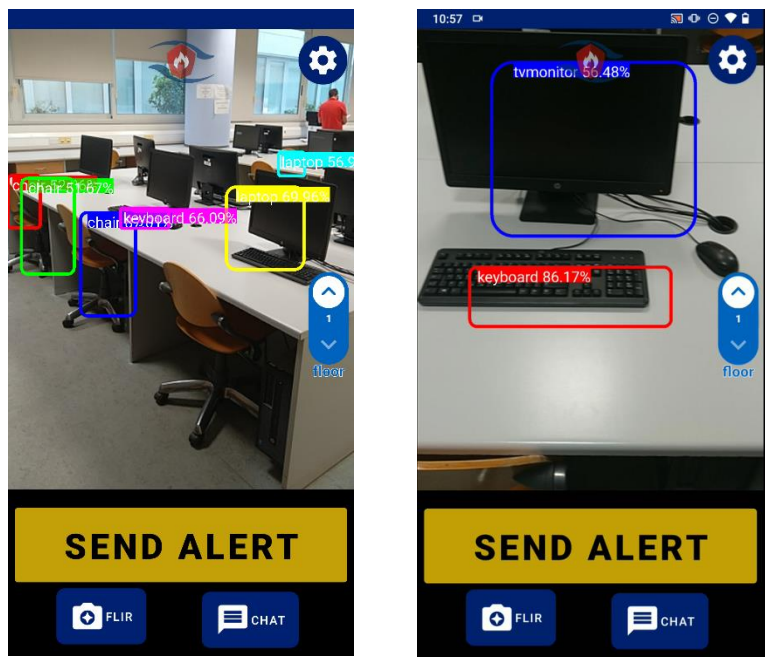
5.5 Παραδείγματα Χρήσης

Ακολουθούν κάποια παραδείγματα χρήσης του αλγόριθμου στους χώρους του Πανεπιστημίου Κύπρου για τα μοντέλα UCYCO και COCO.

UCYCO



COCO



Κεφάλαιο 6

Υποσύστημα IMU και Αλγόριθμος Map Matching

6.1 Στόχος	48
6.2 Αδρανειακά Συστήματα Μέτρησης	49
6.3 Προηγούμενη Υλοποίηση	50
6.4 Αξιολόγηση των Αισθητήρων	50
6.5 Υλοποίηση IMU με τη Χρήση Αισθητήρων και Dead Reckoning	55
6.6 Αλγόριθμος Map Matching	56

Το κεφάλαιο αυτό μελετάει τα αδρανειακά συστήματα μέτρησης (IMU) και πως μπορούν να χρησιμοποιηθούν για να ενισχύσουν την ακρίβεια του αλγόριθμου που χρησιμοποιεί υπολογιστική όραση. Γίνεται μια μικρή αναφορά στην προηγούμενη υλοποίηση που χρησιμοποιείται στο σύστημα Anyplace Navigator. Ακολουθώς, περιγράφεται η διαδικασία της αξιολόγησης των μετρήσεων από τους αισθητήρες κινητού τηλεφώνου και πως επιλέχθηκαν οι κατάλληλοι για την ανάπτυξη του υποσυστήματος IMU. Στην συνέχεια, αναλύεται ο αλγόριθμος που αναπτύχθηκε με την χρήση του IMU και της μεθόδου Dead Reckoning και τέλος αναφέρεται η χρήση του αλγόριθμου Map Matching.

6.1 Στόχος

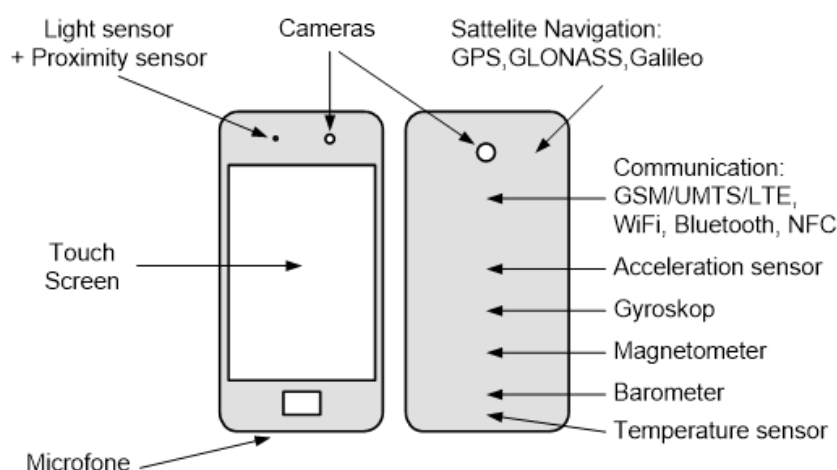
Στόχος μας ήταν η υλοποίηση ενός ανιχνευτή, ή αλλιώς Tracker όπως μπορεί να βρεθεί στον κώδικα, ο οποίος μπορεί να ιχνηλατήσει το κάθε βήμα ενός χρήστη καθώς και την κατεύθυνση που ακολουθεί. Η συνεχής ανίχνευση του κάθε βήματος του δημιουργεί την έννοια της συνεχόμενης κίνησης στο επίπεδο της εφαρμογής και ο ίδιος ο χρήστης μπορεί

ανά πάσα στιγμή να γνωρίζει που βρίσκεται σε ένα χώρο, δεδομένου ότι είναι ήδη γνωστή η αρχική του θέση.

Σκοπός είναι η ενσωμάτωση του στο μεγαλύτερων διαστάσεων σύστημα γεωτοποθέτησης SMAS μαζί με την τεχνική Map Matching. Ο συνδυασμός του υπολογισμού της εσωτερικής τοποθεσίας με χρήση υπολογιστικής όρασης και του ανιχνευτή βοηθάει στην ύπαρξη καλύτερων αποτελεσμάτων και σε πιο ακριβής στιγματοθέτηση του χρήστη στον χώρο καθώς κινείται. Είναι γνωστό ότι η χρήση του Παγκόσμιου Συστήματος Στιγματοθέτησης (GPS) για πλοήγηση καταναλώνει αρκετή ενέργεια, κάτι ανεπιθύμητο για τους χρήστες των κινητών τηλεφώνων. Με τον συνδυασμό αυτό αποφεύγουμε την μεγάλη αυτή κατανάλωση.

6.2 Αδρανειακά Συστήματα Μέτρησης

Τα αδρανειακά συστήματα μέτρησης (Inertial Measurement Units – IMUs) είναι ηλεκτρονικές συσκευές που παρέχουν πληροφορίες για την κίνηση και την κατεύθυνση ενός σώματος. Συνήθως αποτελούνται από τρεις αισθητήρες, ένα επιταχυνσιόμετρο, ένα γυροσκόπιο και ένα μαγνητόμετρο. Το επιταχυνσιόμετρο μετράει την γραμμική επιτάχυνση του σώματος, το γυροσκόπιο την γωνιακή του ταχύτητα και το μαγνητόμετρο μετράει στοιχεία σχετικά με τον γήινο μαγνητισμό [30].



Σχήμα 6.1 : Οι διάφοροι αισθητήρες που υπάρχουν σε ένα κινητό τηλέφωνο [25]

Τέτοιου είδους συστήματα μπορούμε να βρούμε ενσωματωμένα στα πλείστα έξυπνα κινητά τηλέφωνα αφού χρησιμοποιούνται στα παιχνίδια, για περιστροφή της οθόνης από portrait mode σε landscape mode κ.α. Βασιζόμενοι στο ότι τα κινητά τηλέφωνα έχουν ενσωματωμένο ένα τέτοιο σύστημα, μπορούμε να χρησιμοποιήσουμε τις μετρήσεις από τους αισθητήρες του για να φτιάξουμε ένα σύστημα σε μορφή λογισμικού που να μπορεί να ανιχνεύει την κίνηση του χρήστη που κρατάει το κινητό τηλέφωνο και την κατεύθυνση του, κάτι το οποίο έχει υλοποιηθεί και εξηγείται στα παρακάτω υποκεφάλαια.

Ένα μεγάλο μειονέκτημα της πλοήγησης με την χρήση IMU είναι το σφάλμα που συσσωρεύεται με την πάροδο του χρόνου, το οποίο επηρεάζει συνολικά την απόδοση ολόκληρου του συστήματος. Οι μετρήσεις των αισθητήρων δεν είναι πάντα σωστές. Ακόμη και μια μικρή απόκλιση από τις σωστές τιμές μπορεί να τοποθετήσει το κινούμενο σώμα (π.χ. χρήστης) σε μια εντελώς λανθασμένη θέση, μακριά από την πραγματική του.

6.3 Προηγούμενη Υλοποίηση

Στην προηγούμενη έκδοση της εφαρμογής Anyplace, το Αδρανειακό Σύστημα Μέτρησης βρίσκεται υλοποιημένο στην γλώσσα προγραμματισμού Java και αξιοποιεί το Android API για Sensors. Οι μετρήσεις από τους αισθητήρες φιλτράρονται μέσω του Kalman filter και υπολογίζεται η νέα θέση του χρήστη βάση του αριθμού των βημάτων του και του προσανατολισμού του. Όταν ο χρήστης σταματήσει την κίνηση του, οι μετρήσεις του συστήματος αποθηκεύονται σε ένα πίνακα από τον οποίο υπολογίζεται η μέση τιμή (ή ενδιαμέση) από όλες τις αποθηκευμένες. Με το που αρχίσει ο χρήστης ξανά την κίνηση του, η διαδικασία αυτή επαναλαμβάνεται [10].

6.4 Αξιολόγηση των Αισθητήρων

Το Android API για Sensors δίνει την δυνατότητα πρόσβασης σε πάρα πολλούς αισθητήρες, τόσο υλικού όσο και λογισμικού. Τα δεδομένα που λαμβάνονται από τις μετρήσεις και τις συναφείς συναρτήσεις της διεπαφής είναι ακατέργαστα, κάτι το οποίο δίνει την δυνατότητα στον οποιοδήποτε να τις αξιοποιήσει όπως επιθυμεί. Οι αισθητήρες χωρίζονται σε κατηγορίες ανάλογα με το είδος τους, από τις οποίες μερικές είναι κίνησης

[17], θέσης [19] και περιβάλλοντος. Εμάς μας έχουν απασχολήσει περισσότερο οι αισθητήρες κίνησης αφού αυτό ακριβώς είναι που θέλουμε να επιτύχουμε.

Δεν είναι όλοι οι αισθητήρες (λογισμικού) συμβατοί με όλα τα κινητά με λειτουργικό Android και επιπλέον δεν υπάρχουν όλοι οι αισθητήρες στο επίπεδο υλικού σε όλα τις συσκευές. Για παράδειγμα, τα Android 4.0 (με API Level 14) είναι συμβατά με το γυροσκόπιο ενώ τα Android 2.2 (με API Level 8) όχι. Με την ραγδαία όμως εξέλιξη της τεχνολογίας και την ανάγκη των ανθρώπων να ανταποκριθούν σε αυτήν, ευελπιστούμε ότι οι περισσότεροι χρήστες θα διαθέτουν τον κατάλληλο εξοπλισμό για την χρήση της εφαρμογής. Αν όχι, τότε το σύστημα με τους αισθητήρες δεν θα δουλεύει αφού δεν θα έχει δεδομένα για επεξεργασία.

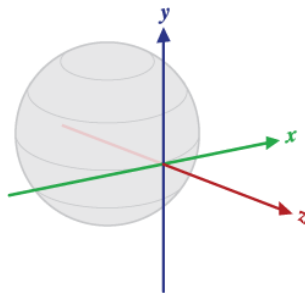
Αισθητήρας	Τύπος	Περιγραφή
TYPE_STEP_COUNTER	Λογισμικό	Ο αριθμός βημάτων του χρήστη μετά από την τελευταία επανεκκίνηση της συσκευής
TYPE_STEP_DETECTOR	Λογισμικό	Ο αριθμός βημάτων του χρήστη από την εκκίνηση της εφαρμογής
TYPE_ROTATION_VECTOR	Λογισμικό ή Υλικό	Ο προσανατολισμός της συσκευής στους 3 άξονες περιστροφής
TYPE_GYROSCOPE	Υλικό	Ο ρυθμός περιστροφής της συσκευής στο σύστημα των 3 αξόνων (rad/s)
TYPE_ACCELEROMETER	Υλικό	Η επιτάχυνση της συσκευής στο σύστημα των 3 αξόνων, συμπεριλαμβανομένου και της βαρύτητας (m/s^2)
TYPE_LINEAR_ACCELERATION	Λογισμικό ή Υλικό	Η επιτάχυνση της συσκευής στο σύστημα των 3 αξόνων, χωρίς τη βαρύτητα (m/s^2)

Πίνακας 6.1 : Οι αισθητήρες που αξιολογήθηκαν και η περιγραφή τους [18]

Αφού το προαναφερόμενο API δίνει την δυνατότητα αλληλεπίδρασης με ένα μεγάλο εύρος από αισθητήρες, απαραίτητη είναι η αξιολόγηση όσων φαινομενικά είναι κατάλληλοι για το αποτέλεσμα που επιδιώκουμε. Για αυτό τον λόγο αναπτύχθηκε μια απλή εφαρμογή για Android κινητά, όπου ο χρήστης μπορεί να δει τις μετρήσεις από τους αισθητήρες. Στα αρχικά στάδια, η εφαρμογή αυτή χρησιμοποιήθηκε για την αξιολόγηση των αισθητήρων, έτσι ώστε να βρεθούν αυτοί με τις πιο ακριβείς και συνεπείς μετρήσεις, οι οποίες θα χρησιμοποιούνταν αργότερα για την υλοποίηση του Tracker. Οι αισθητήρες που αξιολογήθηκαν με την βοήθεια της εφαρμογής, καθώς και οι περιγραφές τους φαίνονται στον πίνακα 6.1.

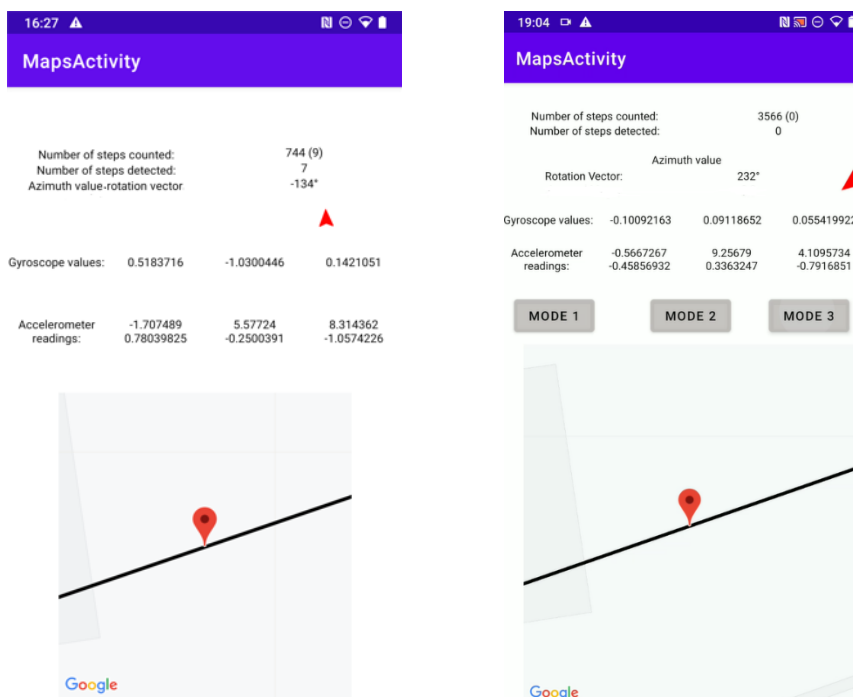
Το σύστημα των 3 αξόνων για την θέση και κίνηση της συσκευής έχει ως εξής [17]:

- Ο άξονας X είναι παράλληλος με το έδαφος και δείχνει ανατολικά.
- Ο άξονας Y είναι παράλληλος με το έδαφος και δείχνει προς τον μαγνητικό Βόρειο Πόλο.
- Ο άξονας Z είναι κάθετος ως προς το επίγειο επίπεδο και δείχνει προς τον ουρανό.



Σχήμα 6.2 : Σύστημα συντεταγμένων [17]

Για να λειτουργήσει το Αδρανειακό Σύστημα Μέτρησης χρειαζόμαστε τα βήματα του χρήστη και την κατεύθυνση που ακολουθεί. Η ανίχνευση των βημάτων μπορεί να γίνει από τους αισθητήρες TYPE_STEP_COUNTER, TYPE_STEP_DETECTOR, TYPE_ACCELEROMETER και TYPE_LINEAR_ACCELEROMETER. Για την ανίχνευση βημάτων με την χρήση των δύο τελευταίων αισθητήρων, απαραίτητη είναι η διαμόρφωση και επεξεργασία των μετρήσεων τους. Αντίστοιχα, ο προσανατολισμός της συσκευής καθορίζει την κατεύθυνση που ακολουθείται από τον χρήστη μέσα στο σύστημα, γι' αυτό δύναται ο ίδιος ο χρήστης να κρατάει την συσκευή στραμμένη ως προς την κατεύθυνση που περπατάει και βλέπει. Οι αισθητήρες που μπορούν να μας δώσουν την μέτρηση αυτή είναι οι TYPE_GYROSCOPE και TYPE_ROTATION_VECTOR.



Σχήμα 6.3 : Στιγμιότυπα οθόνης από την αρχική υλοποίηση και αξιολόγηση του συστήματος IMU

Στο αριστερό μέρος του σχήματος 6.3 φαίνεται η πρώτη υλοποίηση της εφαρμογής αξιολόγησης, με την κάθε μέτρηση να αντιστοιχεί στον τύπο αισθητήρα που αναγράφεται δίπλα της. Ο αριθμός που βρίσκεται μέσα σε παρενθέσεις, δίπλα από την μέτρηση του αισθητήρα TYPE_STEP_COUNTER, είναι ο αριθμός των κλήσεων που έγιναν από το σύστημα για τον αισθητήρα αυτόν από την εκκίνηση της εφαρμογής. Η μέτρηση αυτή πάρθηκε έτσι ώστε να αξιολογήσουμε κατά πόσο οι αισθητήρες TYPE_STEP_COUNTER και TYPE_STEP_DETECTOR συμπεριφέρονται το ίδιο. Μέσα και από άλλες μετρήσεις και αξιολογήσεις των δύο αυτών αισθητήρων, συμπεράναμε ότι, τις πλείστες φορές συμπεριφέρονται το ίδιο, με κάποιες εξαιρέσεις (όπως φαίνεται και στο σχήμα 6.3).

Μετατρέποντας το διάνυσμα μετρήσεων που παρέχει ο αισθητήρας TYPE_ROTATION_VECTOR σε ένα πίνακα περιστροφής και ακολούθως εξάγοντας από αυτόν τον προσανατολισμό του σώματος (δηλαδή της συσκευής), λαμβάνουμε τρεις μετρήσεις στο σύστημα των τριών διαστάσεων. Οι μετρήσεις αυτές έχουν ως εξής:

- Το αζιμούθιο (azimuth), όπου είναι η γωνία που σχηματίζεται μεταξύ της νοητής γραμμής του Μαγνητικού Βορρά και της τρέχουσας κατεύθυνσης της συσκευής. Για παράδειγμα, το αζιμούθιο ισούται με 0° όταν η συσκευή δείχνει ως προς τον Μαγνητικό Βορρά και με 180° όταν δείχνει νότια.

- Η κλίση pitch, όπου είναι η γωνία μεταξύ ενός επιπέδου παράλληλου με το έδαφος και ενός επιπέδου παράλληλου με την οθόνη της συσκευής.
- Η κλίση roll, όπου είναι η γωνία μεταξύ ενός επιπέδου κάθετου ως προς το επίγειο επίπεδο και ενός επιπέδου κάθετου με την οθόνη της συσκευής [19].

Η γωνία που χρειαζόμαστε για να βρούμε την κατεύθυνση/ προσανατολισμό της συσκευής, και συνεπώς του χρήστη, είναι το αζιμούθιο, το οποίο φαίνεται στα αριστερά του σχήματος 6.3 με τιμές από $[-180, 180]$ και στα δεξιά με τιμές $[0, 360]$.

```
val vector: FloatArray = event.values
val targetMatrix = FloatArray( size: 16)
var orientation = FloatArray( size: 3)

//Converts the vector into a rotation matrix
SensorManager.getRotationMatrixFromVector(targetMatrix, vector)
//Converts the rotation matrix into a set of radian values
SensorManager.getOrientation(targetMatrix, orientation)
var azimuth = Math.toDegrees(orientation[0].toDouble())
if (azimuth < 0)
    azimuth += 360
```

Σχήμα 6.4 : Κώδικας ο οποίος εξάγει την τιμή του αζιμούθιου

Οι τιμές των αισθητήρων TYPE_GYROSCOPE και TYPE_ACCELEROMETER παρουσιάζονται με την σειρά x, y, z. Στα Accelerometer readings, πάνω έχουμε τις τιμές από το TYPE_ACCELEROMETER και κάτω από το TYPE_LINEAR_ACCELEROMETER.

Στα δεξιά του σχήματος 6.3 βλέπουμε μία δεύτερη έκδοση της εφαρμογής αξιολόγησης, όπου με διαφορετικούς τρόπους λειτουργίας (modes) αξιολογούνται οι μετρήσεις και το IMU που δημιουργείται μέσω αυτών. Αξιολογήθηκε το σύστημα IMU με ανίχνευση βημάτων με την χρήση του επιταχυνσιόμετρου και η εύρεση κατεύθυνσης έγινε με τη χρήση γυροσκοπίου. Η δεύτερη αξιολόγηση ήταν με την χρήση των αισθητήρων TYPE_STEP_COUNTER/ TYPE_STEP_DETECTOR για την ανίχνευση των βημάτων και το αζιμούθιο από το TYPE_ROTATION_VECTOR για την κατεύθυνση.

Μετά από διάφορες δοκιμές σε εξωτερικούς και εσωτερικούς χώρους, καταλήξαμε ότι ο συνδυασμός TYPE_STEP_DETECTOR για ανίχνευση βημάτων και το αζιμούθιο από τον αισθητήρα TYPE_ROTATION_VECTOR για τον προσανατολισμό, δίνουν τα πιο ικανοποιητικά αποτελέσματα. Αργότερα ίσως συνδυαστεί και με το σύστημα που

χρησιμοποιεί το επιταχυνσιόμετρο, έτσι ώστε η ανιχνεύσεις των βημάτων να είναι πιο ακριβής.

6.5 Υλοποίηση του IMU με τη Χρήση Αισθητήρων και Dead Reckoning

Dead Reckoning είναι η διαδικασία υπολογισμού της νέας θέσης ενός κινούμενου αντικειμένου βάσει της προηγούμενης του η οποία είναι γνωστή. Στην περίπτωση του Tracker είναι απαραίτητο να γνωρίζουμε τουλάχιστον την αρχική θέση του σώματος για να μπορέσουμε να υπολογίσουμε τις επόμενες του. Στις εφαρμογές που αναπτύχθηκαν για τις δοκιμές του συστήματος, ο ίδιος ο χρήστης επισημαίνει πάνω στον χάρτη την θέση του. Αργότερα, όταν το σύστημα θα ενσωματωθεί στην εφαρμογή SMAS, η θέση του χρήστη θα υπολογίζεται με τον αλγόριθμο που κάνει χρήση της υπολογιστικής όρασης και των μοντέλων μηχανικής μάθησης.

Η ανίχνευση των βημάτων γίνεται με τον αισθητήρα TYPE_STEP_DETECTOR όπως αναφέραμε και πιο πάνω. Ορίσαμε μια μεταβλητή για τις μετρήσεις του, τύπου MutableLiveData η οποία επιτρέπει την παρακολούθηση της. Έτσι, κάθε φορά που η τιμή της μεταβλητής αυτής αυξάνεται, δίνει το «σήμα» ότι ένα νέο βήμα γίνεται από τον χρήστη. Για κάθε νέο ανιχνευτόμενο βήμα καλείται η μέθοδος που υπολογίζει την νέα θέση του χρήστη. Στο σχήμα 6.5 βλέπουμε το κομμάτι του κώδικα που το υλοποιεί, όπου position είναι η αρχική θέση του χρήστη που ανανεώνεται κάθε φορά, azimuth το αζιμούθιο από τον αισθητήρα TYPE_ROTATION_VECTOR και mMap ο χάρτης.

```
viewModel.stepsDetected.observe(owner: this, Observer { it: Int!
    if(!stopObserver1) {
        //Generate a beep sound every time the sensor is triggered
        val toneGenerator = ToneGenerator(AudioManager.STREAM_NOTIFICATION, volume: 100)
        toneGenerator.startTone(ToneGenerator.TONE_PROP_BEEP);

        position = viewModel.findNewPos(position, azimuth, mMap)
    }
})
```

Σχήμα 6.5 : Κομμάτι κώδικα το οποίο παρακολουθεί τις αλλαγές στην τιμή της μεταβλητής stepsDetected

Για να υπολογιστεί η νέα θέση αξιοποιούμε τις πιο κάτω γνωστές τιμές και μαθηματικές εξισώσεις:

$$\begin{aligned}
&\text{earthRadius} = 6371000 \quad \text{stepSize} = 0.37 \\
&\text{lat} = \text{lat} * \pi/180 \quad \text{lon} = \text{lon} * \pi/180 \quad \text{azimuthRad} = \text{azimuth} * \pi/180 \\
&\text{newLat} = \text{asin}(\sin(\text{lat}) * \cos(\text{stepSize}/\text{earthRadius}) + \cos(\text{lat}) * \sin(\text{stepSize}/\text{earthRadius}) * \cos(\text{azimuthRad})) \\
&\text{newLon} = \text{lon} + \text{atan2}(\sin(\text{azimuthRad}) * \sin(\text{stepSize}/\text{earthRadius}) * \cos(\text{lat}), \cos(\text{stepSize}/\text{earthRadius}) - \sin(\text{lat}) * \sin(\text{newLat})) \\
&\text{newLat} = \text{newLat} * 180/\pi \quad \text{newLon} = \text{newLon} * 180/\pi
\end{aligned}$$

Όπου earthRadius η ακτίνα της Γης σε μέτρα, stepSize το μήκος ενός βήματος ενός ανθρώπου, lat και lon οι προηγούμενες συντεταγμένες (προηγούμενη θέση), azimuthRad το αζιμούθιο από τον αισθητήρα TYPE_ROTATION_VECTOR σε rads και newLat, newLon οι νέες γεωγραφικές συντεταγμένες.

6.6 Αλγόριθμος Map Matching

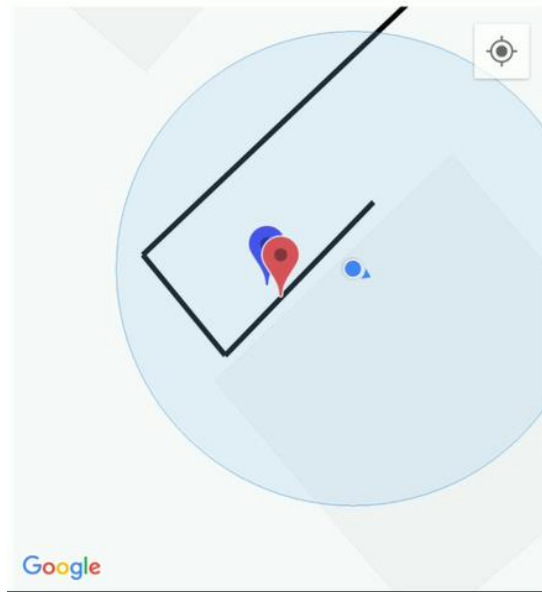
Ο αλγόριθμος που περιεγράφηκε πιο πάνω με την χρήση του IMU και της τεχνικής Dead Reckoning θα χρησιμοποιηθεί στις περιπτώσεις όπου οι χρήστες του συστήματος SMAS θέλουν να πλοηγηθούν από ένα σημείο σε ένα άλλο μέσα στο πλοίο. Θα συνδυαστεί με τον αλγόριθμο της υπολογιστικής όρασης έτσι ώστε να βελτιώνει τον εντοπισμό της νέας θέσης αλλά και της κατεύθυνσης της κίνησης.

Για παράδειγμα, όταν ο χρήστης ψάξει ένα POI με την χρήση του εργαλείου αναζήτησης (βλ. υποκ. 4.3) και αποφασίσει ότι θέλει να πάει σε αυτό, τότε πάνω στον χάρτη θα εμφανιστεί η διαδρομή με την χρήση των polylines. Polyline είναι μέρη γραμμών ενωμένα, έτσι ώστε να σχηματίζουν μια διαδρομή, ένα μονοπάτι ή άλλου είδους ενώσεων μεταξύ σημείων πάνω σε χάρτη. Καθώς θα περπατάει ο χρήστης κατά μήκος της νοητής αυτής διαδρομής πάνω στο πλοίο, η θέση του στον χάρτη θα ανανεώνεται με κάθε νέο του βήμα. Όμως, το σημάδι του χρήστη που επισημαίνει την θέση του δεν θα ακολουθεί την σχεδιασμένη διαδρομή, είτε λόγω σφαλμάτων από το IMU, είτε σφαλμάτων γεωτοποθέτησης από τον αλγόριθμο υπολογιστικής όρασης. Για τον λόγο αυτόν, μελετήσαμε και αξιοποιήσαμε τον αλγόριθμο Map Matching.



Σχήμα 6.6 : Μια διαδρομή με χρήση polylines

Σκοπός του αλγόριθμου Map Matching είναι να αντιστοιχήσει τις νέες υπολογιζόμενες γεωγραφικές συντεταγμένες με συντεταγμένες που ανήκουν στο πιο κοντινό polyline. Έτσι, το σημάδι του χρήστη δεν φεύγει από τα όρια της διαδρομής που σχηματίζεται. Ο αλγόριθμος έχει ως εξής: αρχικά για κάθε polyline παίρνουμε πέντε σημεία του, την αρχή, το τέλος, την μέση, και τα σημεία που βρίσκονται στα δύο τέταρτα του. Έπειτα, υπολογίζουμε την απόσταση από τα σημεία αυτά, κρατώντας την μικρότερη. Η μικρότερη απόσταση από όλα τα polylines και τα σημεία τους, είναι και η θέση του χρήστη στην διαδρομή.



Σχήμα 6.7 : Χάρτης με polylines και markers, χωρίς map matching (μπλε), με map matching (κόκκινο)

Κεφάλαιο 7

Πειραματική Αξιολόγηση

7.1 Αξιολόγηση Αποτελεσμάτων της Φάσης Εκπαίδευσης	58
7.2 Αξιολόγηση του Υποσυστήματος Ανταλλαγής Μηνυμάτων	61
7.3 Αξιολόγηση του Υποσυστήματος IMU	63

Στο κεφάλαιο αυτό γίνεται αναφορά στα διάφορα πειράματα και στις αξιολογήσεις που έγιναν στα συστήματα με τα οποία ασχολείται η παρούσα διπλωματική εργασία. Πιο συγκεκριμένα, αναφέρονται κάποιοι παράμετροι που επηρέασαν το παραγόμενο μοντέλο LASHCO, αξιολογείται το κανάλι επικοινωνίας που υλοποιήθηκε και τέλος, παρουσιάζεται ένα μικρό πείραμα που εφαρμόστηκε στο υποσύστημα IMU.

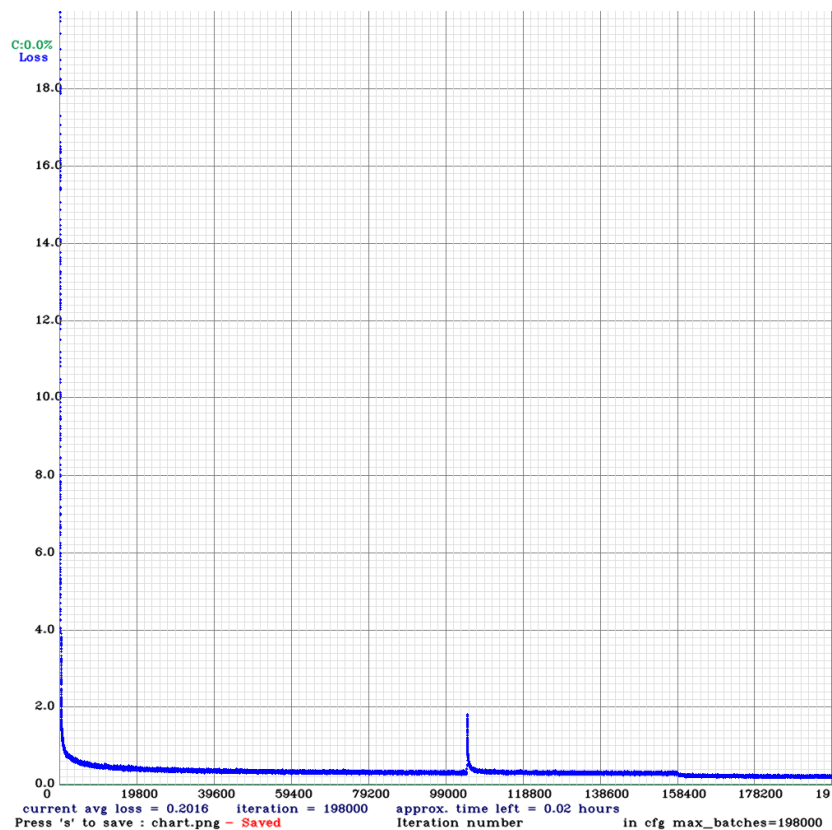
7.1 Αξιολόγηση Μετρήσεων στη Φάση Εκπαίδευσης

Κατά τη διάρκεια της φάσης εκπαίδευσης των νευρωνικών δικτύων είχαμε την ευκαιρία να αλλάξουμε μερικές παραμέτρους ώστε να ρυθμίσουμε το παραγόμενο μοντέλο βάσει των αποτελεσμάτων που επιθυμούσαμε. Στο υποκεφάλαιο 5.2 «Φάση Εκπαίδευσης», γίνεται αναφορά στην μέθοδο quantization με την οποία το μέγεθος του μοντέλου μπορεί να μειωθεί. Στο σημείο αυτό θα εξετάσουμε τρεις άλλες παραμέτρους που είχαμε την ευκαιρία να αλλάξουμε. Τον αριθμό των επαναλήψεων, το μέγεθος batch και το μέγεθος των subdivisions.

Ο αριθμός των επαναλήψεων είναι μια από τις πιο κρίσιμες παραμέτρους οι οποίες επηρεάζουν σημαντικά το τελικό αποτέλεσμα. Είναι ο αριθμός των φορών που μια δέσμη από δεδομένα περνάει μέσα από τον αλγόριθμο. Ο αριθμός αυτός επηρεάζει την ακρίβεια του μοντέλου και κατά πόσο θα μπορεί να αναγνωρίζει τα αντικείμενα με μεγάλη

βεβαιότητα. Επιπλέον, έχει επίδραση στον χρόνο της εκπαίδευσης αφού οι φορές για τις οποίες πρέπει να τρέξει ο αλγόριθμος μειώνονται ή αυξάνονται αναλόγως.

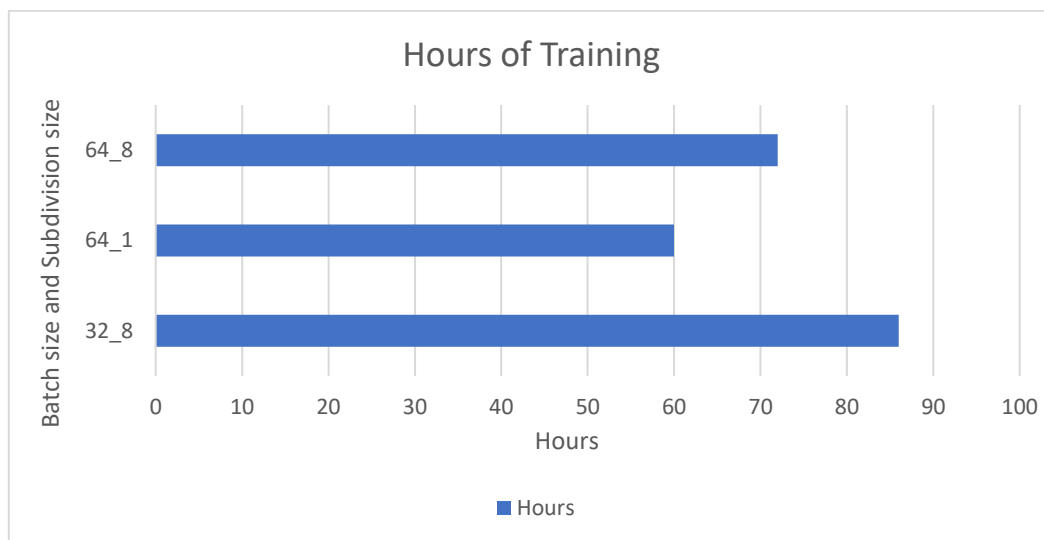
Όπως μπορούμε να δούμε στο σχήμα 7.1, η γραφική παρουσιάζει την αντιστοίχιση μεταξύ του loss και του αριθμού των επαναλήψεων. Το loss είναι ο αριθμός που δείχνει το ποσοστό των λάθος προβλέψεων του δικτύου για τα δεδομένα που παίρνει σαν είσοδο. Παρατηρώντας την γραφική, βλέπουμε ότι περίπου το μοντέλο εκπαιδεύεται με 198 χιλιάδες επαναλήψεις, στις οποίες η μέση τιμή του loss είναι στα 0.2016. Μπορούμε επίσης να δούμε ότι η μείωση της τιμής του loss στο περιθώριο 0 – 19800 επαναλήψεων είναι δραματική και στην περίπτωση που σταματούσαμε την εκπαίδευση στο σημείο αυτό, τα αποτελέσματα του μοντέλου δεν θα ήταν καθόλου ικανοποιητικά.



Σχήμα 7.1 : Γραφική παράσταση του loss με τον αριθμό επαναλήψεων όπου το avg. loss είναι στα 0.2016

Οι παράμετροι batch-size και subdivision-size σχετίζονται μεταξύ τους. Η παράμετρος batch αναφέρεται στο μέγεθος των δεσμών που θα χωρίσουμε τα δεδομένα μας σε μια επανάληψη. Η ποσότητα αυτή είναι η ποσότητα των εικόνων που θα επεξεργαστούμε πριν να αλλάξουν τα βάρη του δικτύου. Δηλαδή, στην περίπτωση που έχουμε ένα dataset

με 320 εικόνες και το batch-size είναι 64, οι εικόνες θα χωριστούν σε 5 δέσμες των 64 εικόνων η κάθε μια. Άρα σε μια επανάληψη, τα βάρη θα αλλάξουν 5 φορές, δηλαδή κάθε 64 εικόνες. Έτσι, καταλαβαίνουμε ότι το μέγεθος των δεσμών επηρεάζει τον χρόνο επεξεργασίας, αφού αν αλλάζαμε τα βάρη μετά το πέρας κάθε 32 εικόνων (batch-size = 32) τότε η αλλαγή των βαρών θα γινόταν πολύ πιο συχνά, αρά θα χρειαζόταν και πολύ περισσότερο χρόνο επεξεργασίας.



Σχήμα 7.2 : Ώρες εκπαίδευσης σε σύγκριση με το μέγεθος των δεσμών και των subdivisions

Το μέγεθος των subdivisions αναφέρεται στα πόσα mini batches θα έχουμε ανά δέσμη. Επηρεάζει τον αριθμό των εικόνων που στέλνονται στην GPU για παράλληλη επεξεργασία. Άρα, στην περίπτωση των δεσμών μεγέθους 64 με 8 subdivisions ανά δέσμη, 8 εικόνες μπορούν να επεξεργάζονται παράλληλα. Τα subdivisions πρέπει να ρυθμιστούν ανάλογα με την GPU που είναι διαθέσιμη. Επηρεάζουν και αυτά τον χρόνο επεξεργασίας όπως φαίνεται στο σχήμα 7.2.

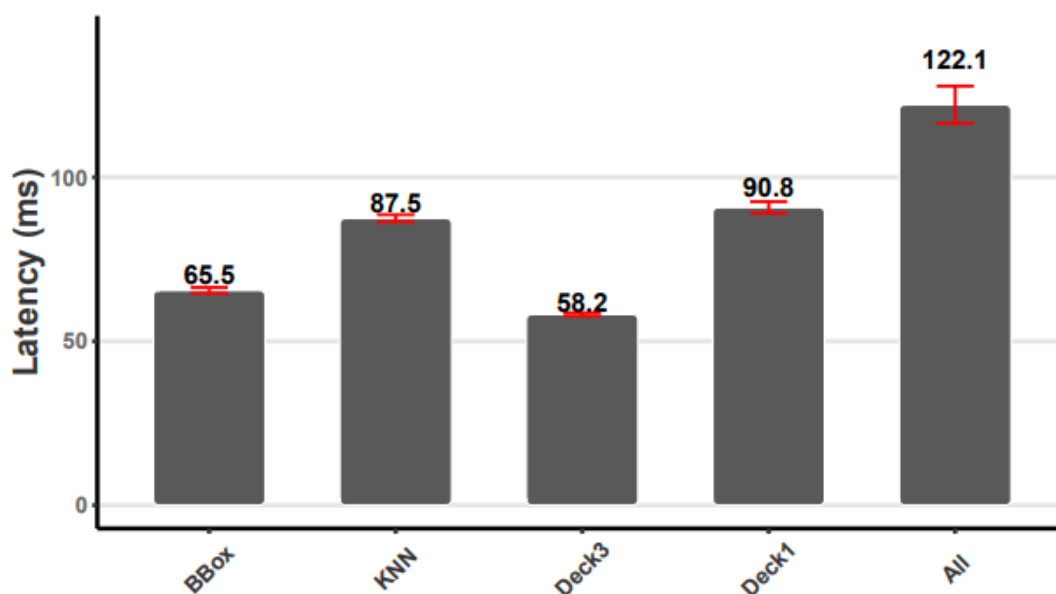
Από την γραφική στο σχήμα 7.2 μπορούμε να διαπιστώσουμε ότι η πιο γρήγορη εκπαίδευση γίνεται με batch-size = 64 και subdivisions = 1 αφού διήρκεσε 60 ώρες. Η εκπαίδευση με batch-size = 64 και subdivisions = 8 πήρε περίπου 72 ώρες, ενώ η τρίτη εκπαίδευση με μέγεθος δέσμης 32 και 8 subdivisions διήρκεσε πολύ περισσότερο από τις υπόλοιπες, ξεπερνώντας τις 80 ώρες. Στην επίσημη ιστοσελίδα του συστήματος YOLO, ο δημιουργός του προτείνει και χρησιμοποιεί την εκπαίδευση 64_8 γι' αυτό και αυτήν αξιοποιήσαμε.

7.2 Αξιολόγηση Υποσυστήματος Ανταλλαγής Μηνυμάτων

Για την αξιολόγηση του υποσυστήματος ανταλλαγής μηνυμάτων διεκπεραιώθηκαν δύο πειράματα. Αξιολογήθηκαν οι χρόνοι παράδοσης των μηνυμάτων ανάλογα με την μέθοδο που επιλέχθηκε και η απόδοση του κυλιόμενου σκέλους της συζήτησης.

Χρόνοι Παράδοσης Μηνυμάτων

Στο σχήμα 7.3 μπορούμε να δούμε τους χρόνους παράδοσης σε ms για τις διάφορες μεθόδους οι οποίες επηρεάζουν ποια άτομα θα παραλάβουν το μήνυμα. Συγκρίνονται πέντε μέθοδοι. Το BBox το οποίο είναι ρυθμισμένο να στέλνει τα μηνύματα σε άτομα που βρίσκονται σε απόσταση μικρότερη των 100 μέτρων. Το KKN στο οποίο το μήνυμα στέλνεται στους κοντινότερους K γείτονες. Τα Deck3 και Deck1 στα οποία τα μηνύματα στέλνονται μόνο στα άτομα που βρίσκονται στο αντίστοιχο κατάστρωμα και τέλος, το All όπου όλοι οι χρήστες λαμβάνουν το μήνυμα ανεξάρτητα από το που βρίσκονται.



Σχήμα 7.3 : Σύγκριση μεθόδων αποστολής μηνυμάτων

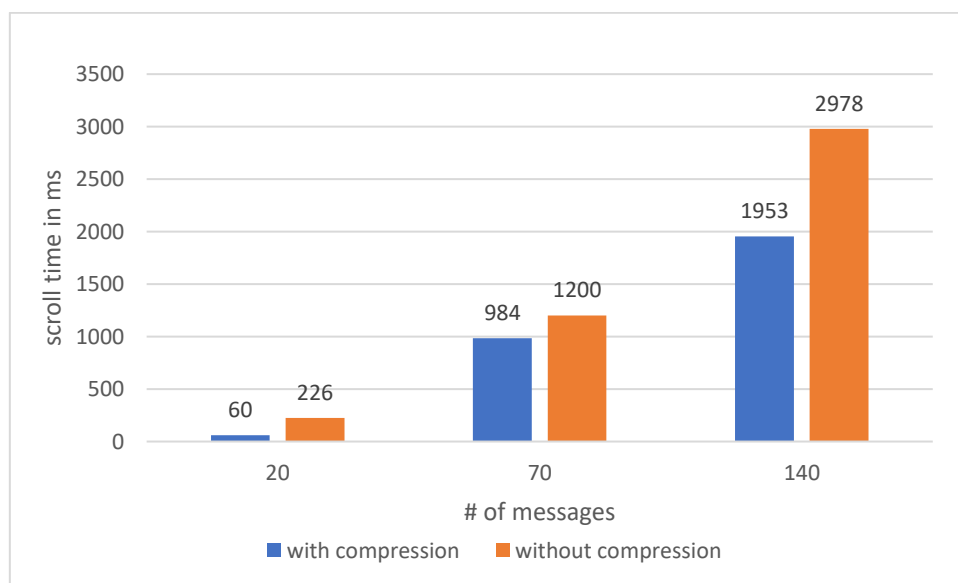
Για την εκτέλεση του πειράματος, χρησιμοποιήθηκε ένας μικρός αριθμός χρηστών και για την κάθε μέθοδο στάλθηκαν 1000 μηνύματα. Από την γραφική στο σχήμα 7.3 μπορούμε να δούμε ότι η μέθοδος που στέλνει τα μηνύματα σε όλους τους χρήστες είναι και η πιο αργή, φτάνοντας τα 122 ms. Όταν περιορίστηκε η αποστολή μηνυμάτων στα

άτομα που βρίσκονταν στα 100 μέτρα, το latency μειώθηκε κατά περίπου 50% σε σχέση με την προηγούμενη μέθοδο. Οι μέθοδοι που στέλνουν το μήνυμα μόνο στα άτομα που βρίσκονται στο ίδιο κατάστρωμα έχουν αρκετή διαφορά στο χρόνο τους, με την μέθοδο Deck3 να φτάνει τα 58 ms και την μέθοδο Deck1 τα 91 ms. Αυτό γιατί στο κατάστρωμα 1 υπήρχαν περισσότερα άτομα. Με την χρήση της μεθόδου KNN παρατηρήσαμε 4% μείωση του latency σε σχέση με την αποστολή των μηνυμάτων στα άτομα του καταστρώματος 1.

Τα αποτελέσματα αυτά μας δείχνουν πόσο απαραίτητο είναι να περιορίζουμε την αποστολή των μηνυμάτων στα άτομα που πρέπει να τα παραλάβουν. Σε περιπτώσεις έκτακτης ανάγκης, είναι σημαντικό η αντιμετώπιση τους και η ενημέρωση των κατάλληλων ατόμων να γίνονται όσο το δυνατόν πιο γρήγορα, χρησιμοποιώντας την πιο σωστή μέθοδο.

Απόδοση του Κυλιόμενου Μέρους της Συζήτησης

Στο πείραμα αυτό μετρήθηκε ο χρόνος κύλισης από το πιο πρόσφατο μήνυμα στο πιο παλιό της συζήτησης. Μέσω του εργαλείου Jetpack Compose δίνεται η δυνατότητα αυτόματης κύλισης, οπότε στους χρόνους που παρουσιάζουμε δεν εμπεριέχονται οι χρόνοι αντίδρασης και αλληλεπίδρασης του χρήστη. Μέσω του πειράματος, λήφθηκαν υπόψη τρεις παράμετροι: ο συνολικός αριθμός των μηνυμάτων, το πλήθος των εικόνων και αν οι εικόνες έτυχαν συμπίεσης ή όχι.



Σχήμα 7.4 : Απόδοση του κυλιόμενου σκέλους του υποσυστήματος ανταλλαγής μηνυμάτων

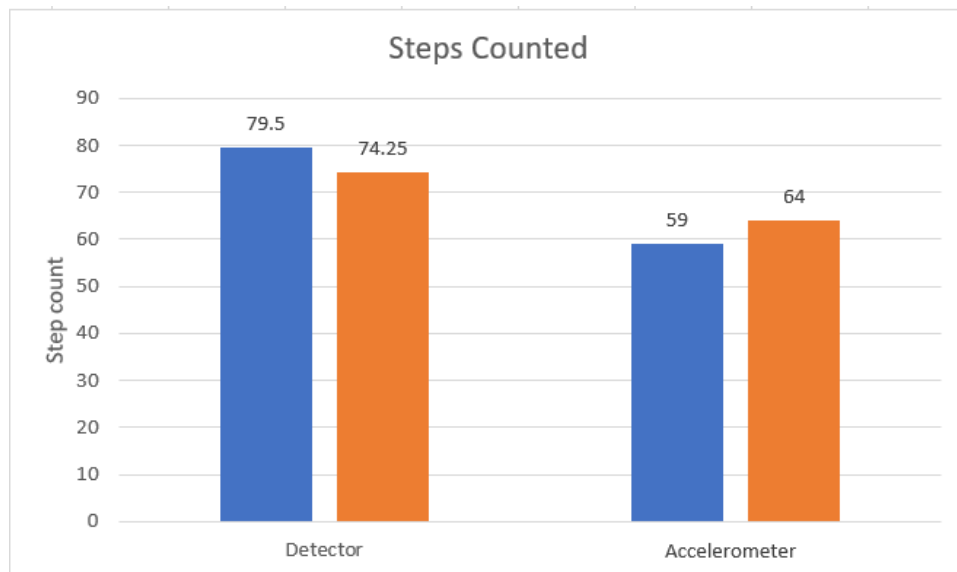
Στην γραφική που φαίνεται στο σχήμα 7.4 μπορούμε να εντοπίσουμε τις εμφανείς διαφορές στον χρόνο κύλισης όταν το πλήθος των μηνυμάτων είναι 20, 70 και 140. Κάτι το οποίο είναι αναμενόμενο, αφού με την παρουσία περισσότερων μηνυμάτων, μεγαλώνει και το μήκος του ανάλογου σκέλους. Την διαφορά στους χρόνους αυτούς επηρέασε και το ποσοστό των εικόνων ανάμεσα στα μηνύματα. Οι εικόνες γενικότερα είναι πολύ μεγαλύτερα αρχεία, οπότε η φόρτωση τους παίρνει περισσότερο χρόνο. Όταν το πλήθος των μηνυμάτων ήταν 20, μόνο μια φωτογραφία υπήρχε ανάμεσα τους. Όταν ήταν 70, υπήρχαν 6 και όταν ήταν 140 υπήρχαν 10.

Η πιο σημαντική παράμετρος από αυτές που εξετάσαμε είναι η παράμετρος της συμπίεσης. Αυτό γιατί δεν θα μπορούμε να διαχειριζόμαστε το πλήθος των μηνυμάτων που θα στέλνονται με το σύστημα, ούτε μπορούμε να περιορίσουμε το ποσοστό των εικόνων που θα υπάρχουν. Αυτά θα γίνονται από την πλευρά του χρήστη. Αυτό που μπορούμε όμως σαν προγραμματιστές να ελέγξουμε έτσι ώστε να αυξήσουμε την απόδοση είναι η συμπίεση των εικόνων. Με την συμπίεση της εικόνας, μειώνεται το μέγεθος της, άρα μειώνεται και ο χρόνος φόρτωσης της.

Μπορούμε να δούμε και στα τρία πειράματα που φαίνονται στην γραφική 7.4, την σημαντική διαφορά στις τιμές του χρόνου όταν οι εικόνες της συνομιλίας έτυχαν συμπίεσης ή όχι. Στο πείραμα με τα 140 μηνύματα και τις 10 εικόνες, ο χρόνος κύλισης με συμπιεσμένες εικόνες είναι 1953 ms ενώ χωρίς συμπίεση 2978 ms. Δηλαδή περίπου 1000 ms παραπάνω. Η διαφορά που υπάρχει στα 70 μηνύματα με 6 φωτογραφίες είναι περίπου στα 220 ms, ενώ στα 20 μηνύματα, η διαφορά δεν είναι τόσο αισθητή όσο στα προηγούμενα λόγω της ύπαρξης μόνο μίας εικόνας.

7.3 Αξιολόγηση Υποσυστήματος IMU

Όπως αναφέρθηκε και στο κεφάλαιο 6 «Υποσύστημα IMU και αλγόριθμος Map Matching», χρειάστηκε να γίνει πειραματική αποτίμηση των μετρήσεων των αισθητήρων που είναι ενσωματωμένοι μέσα σε ένα κινητό τηλέφωνο. Πιο κάτω συγκρίνουμε τις μετρήσεις των βημάτων από δύο αισθητήρες, τον TYPE_STEP_DETECTOR και τον TYPE_ACCELEROMETER (οι οποίες πάρθηκαν με την βοήθεια ενός συστήματος ανοικτού κώδικα: <https://github.com/Lunecaps/SensorsCollector2>).



Σχήμα 7.5 : Σύγκριση των μετρήσεων των δύο αισθητήρων

Στη γραφική, στο σχήμα 7.5, το μπλε χρώμα αναπαριστά τον πραγματικό αριθμό βημάτων και το πορτοκαλί τον αριθμό των βημάτων που μετρήθηκαν με τους αισθητήρες. Όλες οι τιμές είναι ο μέσος όρος ενός συνόλου από μετρήσεων. Μπορούμε να παρατηρήσουμε ότι οι μετρήσεις και των δύο αισθητήρων δεν είναι ακριβείς. Οι μετρήσεις των βημάτων από τον αισθητήρα Step Detector έχουν κατά περίπου 6% απόκλιση από τις πραγματικές. Από την άλλη, οι μετρήσεις από τον αισθητήρα Accelerometer έχουν απόκλιση περίπου 8.5%. Είναι προφανές ότι ο αισθητήρας Detector χάνει κάποια βήματα, ενώ ο αισθητήρας Accelerometer είναι πιο ευαίσθητος, υπολογίζοντας παραπάνω από την πραγματική τιμή.

Για αρχή, αξιοποιούνται μόνο οι μετρήσεις από τον αισθητήρα Detector, αλλά σε μελλοντική δουλειά μπορεί να γίνει ο συνδυασμός των δύο έτσι ώστε να βρεθεί ένα κοινό σημείο μεταξύ των ψηλότερων και χαμηλότερων μετρήσεων.

Κεφάλαιο 8

Συμπεράσματα

8.1 Συμπεράσματα	65
8.2 Μελλοντική Δουλειά	66

8.1 Συμπεράσματα

Στην παρούσα διπλωματική εργασία υλοποιήθηκαν επιπλέον λειτουργίες για το σύστημα SMAS, έτσι ώστε να μπορεί να χρησιμοποιηθεί αποτελεσματικά από το πλήρωμα των πλοίων που ανήκουν στο πρόγραμμα LASH FIRE. Στόχος μας είναι η ικανοποίηση από την αποδοτικότητα και τις λειτουργικές παροχές της εφαρμογής αλλά και η αποτελεσματικότητα των ενεργειών της.

Ο αλγόριθμος που χρησιμοποιείται στην εφαρμογή για τον εντοπισμό της εσωτερικής τοποθεσίας αξιοποιεί τις δυνατότητες της υπολογιστικής όρασης αλλά και της μηχανικής μάθησης. Το παραγόμενο μοντέλο LASHCO για το εσωτερικό των πλοίων έτυχε εκπαίδευσης με δεδομένα που εξάχθηκαν με την βοήθεια του εργαλείου CVAT, μέσω ανασκόπησης πολλών διάφορων βίντεο. Επίσης, αξιολογήθηκαν και οι επιρροές που είχαν κάποιοι παράμετροι κατά την διάρκεια της εκπαίδευσης των νευρωνικών δικτύων με το σύστημα YOLO.

Επίσης, αναπτύχθηκε η διεπαφή, καθώς και οι λειτουργίες του συστήματος ανταλλαγής μηνυμάτων που είναι ενσωματωμένο στην εφαρμογή SMAS. Οι χρήστες μπορούν να στείλουν γραπτό κείμενο, φωτογραφίες, την τοποθεσία τους μέσα στο πλοίο καθώς και ειδοποιήσεις έκτακτης ανάγκης στους υπόλοιπους χρήστες. Είναι επίσης ρυθμιζόμενο το ποια άτομα θα παραλάβουν το μήνυμα, βάσει της τοποθεσίας τους. Υπάρχουν τέσσερις μέθοδοι για τις οποίες αξιολογούνται οι χρόνοι παράδοσης των μηνυμάτων.

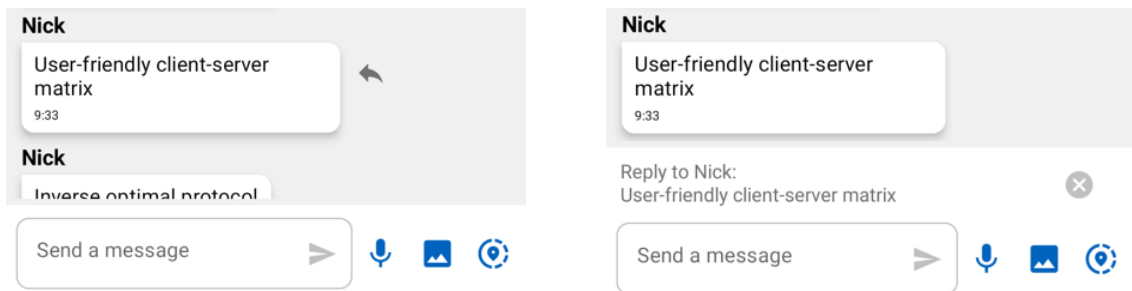
Στους χρήστες παρέχεται επίσης και η δυνατότητα αναζήτησης χώρων του πλοίου, οι οποίοι είναι χαρτογραφημένοι, αλλά και η αναζήτηση άλλων χρηστών. Με την επιλογή τους πάνω στον χάρτη, δίνονται οδηγίες στον χρήστη για το πως να πλοηγηθεί μέχρι το σημείο αυτό, με την θέση του να διορθώνεται με τη χρήση του IMU και του Map Matching.

Το σύστημα IMU, όπως αναφέρεται και παραπάνω, αναπτύχθηκε με σκοπό την διόρθωση της θέσης του χρήστη καθώς περιφέρεται μέσα στο πλοίο, ακολουθώντας οδηγίες. Το IMU κάνει χρήση των μετρήσεων από ένα σύνολο αισθητήρων και βάσει αυτών μπορεί να ανιχνευτεί η κίνηση του χρήστη και να υπολογισθεί η κατεύθυνση της κίνησης του. Σε συνδυασμό με τον αλγόριθμο Map Matching, η γεωτοποθέτηση του χρήστη μέσα στο πλοίο είναι πιο ακριβής.

8.2 Μελλοντική Δουλειά

Η εφαρμογή SMAS βρίσκεται ακόμη στα αρχικά της στάδια και υπάρχει αρκετό περιθώριο για προσθήκη λειτουργιών αλλά και βελτίωσης της. Ξεκινώντας από τον αλγόριθμο που αξιοποιείται, μελλοντικά θα ήταν χρήσιμη η εκπαίδευση περισσότερων μοντέλων. Μέχρι στιγμής υπάρχουν τρία μοντέλα, το LASHCO, το UCYCO και το COCO. Με την εκπαίδευση περισσότερων μοντέλων, ο εντοπισμός την εσωτερικής τοποθεσίας μέσα σε χώρους όπου δεν έχουν τις κατάλληλες υποδομές θα είναι πιο εύκολος. Αυτό όμως απαιτεί το τράβηγμα αρκετών βίντεο από το εσωτερικό του χώρου που μας ενδιαφέρει, την χαρτογράφηση του, την επισήμανση των σημείων ενδιαφέροντος του καθώς και το πέρασμα των βίντεο μέσα από το σύστημα CVAT.

Στο υποσύστημα ανταλλαγής μηνυμάτων μπορεί να προστεθεί η λειτουργία του «reply to a message». Δηλαδή ο χρήστης να μπορεί να απαντήσει απευθείας πάνω σε ένα μήνυμα στην συνομιλία και οι άλλοι χρήστες να μπορούν να το δουν. Η λειτουργία αυτή από την μεριά της διεπαφής έχει υλοποιηθεί, μένει μόνο η υλοποίηση και στην πλευρά της βάσης δεδομένων. Στο σχήμα 8.1 μπορούμε να δούμε στιγμιότυπα οθόνης όπου χρησιμοποιείται και πως φαίνεται στον χρήστη. Επίσης, θα μπορούσε να προστεθεί και η δυνατότητα αποστολής κι άλλων αρχείων μέσα στην συνομιλία, όπως βίντεο, pdf κ.α.



Σχήμα 8.1 : Η λειτουργία «reply to a message»

Όπως αναφέρθηκε στο κεφάλαιο 6 «Υποσύστημα IMU και αλγόριθμος Map Matching» αλλά και κατά την αξιολόγηση του συστήματος στο κεφάλαιο 8, η πιο ακριβής ανίχνευση βημάτων του χρήστη με το σύστημα IMU μπορεί να γίνει με το συνδυασμό των μετρήσεων από δύο αισθητήρες που παρέχει το Android API, παρά με ένα.

Βιβλιογραφία

1. AnyPlace | Indoor Navigation Service, AnyPlace. Available at: <https://anyplace.cs.ucy.ac.cy/>.
2. Bash - GNU project - free software foundation, Gnu.org. Available at: <https://www.gnu.org/software/bash/> (Accessed: April 29, 2022).
3. Bayer, J. B. et al. (2016) “Sharing the small moments: ephemeral social interaction on snapchat,” in Information, Communication & Society, pp. 956–977.
4. Black, E. W., Mezzina, K. and Thompson, L. A. (2016) “Anonymous social media—understanding the content and context of yik yak,” in Computers in Human Behavior, pp. 17–22.
5. Bolliger, P. (2008) Redpin -adaptive, zero-configuration indoor localization through user collaboration, Ethz.ch. Available at: <http://www.vs.inf.ethz.ch/publ/papers/bolligph-redpin2008.pdf> (Accessed: May 2, 2022).
6. Bolliger, P. et al. (2009) “Improving location fingerprinting through motion detection and asynchronous interval labeling,” in Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 37–51.
7. Bovas (2019) How the “Location Based Chat” Feature in Telegram Messenger could use to fight Corona Virus ?, Medium. Available at: <https://medium.com/@bovasjohn/why-the-new-location-based-chat-in-telegram-messenger-could-be-a-game-changer-in-these-five-e83b8434a5c> (Accessed: May 1, 2022).
8. Bray, T. (2014) The JavaScript object notation (JSON) data interchange format. Edited by T. Bray. RFC Editor.
9. Computer Vision Annotation Tool (CVAT). Available at: <https://github.com/opencv/opencv4/cvat> (Accessed: April 29, 2022).
10. Constambeys, T. (2014) Υλοποίηση και Αποτίμηση Υποσυστημάτων Γεωπλοήγησης σε Εσωτερικούς Χώρους. University of Cyprus.

11. Costa, C. et al. (2015) “Rayzit: An Anonymous and Dynamic Crowd Messaging Architecture,” in Proceedings of the 3rd IEEE International Workshop on Mobile Data Management, Mining, and Computing on Social Networks, pp. 98–103.
12. COVTRACER - republic of Cyprus (2021) Gov.cy. Available at: <https://covtracer.dmrld.gov.cy/dmrld/covtracer/covtracer.nsf/home/home?OpenDocument> (Accessed: May 3, 2022).
13. Data Management Systems Laboratory (2014) Anyplace Navigator. Youtube. Available at: <https://www.youtube.com/watch?v=MO-473oWSfE> (Accessed: May 6, 2022).
14. Dmsl (2015) Anyplace Logger. Youtube. Available at: <https://www.youtube.com/watch?v=8EvioLZ6hvg> (Accessed: May 6, 2022).
15. Ghinita, G. et al. (2013) “A privacy-preserving location-based alert system,” Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 432–435.
16. GitHub: Where the world builds software. Available at: <https://github.com/> (Accessed: April 29, 2022).
17. Google (2022) Motion sensors, Android Developers. Available at: https://developer.android.com/guide/topics/sensors/sensors_motion (Accessed: March 21, 2022).
18. Google (2022) Sensors overview, Android Developers. Available at: https://developer.android.com/guide/topics/sensors/sensors_overview (Accessed: March 21, 2022).
19. Google, Position sensors, Android Developers. Available at: https://developer.android.com/guide/topics/sensors/sensors_position (Accessed: March 21, 2022).
20. Jetpack compose, Android Developers. Available at: <https://developer.android.com/jetpack/compose> (Accessed: April 29, 2022).

21. Kotlin programming language (2011) Kotlin. Available at: <https://kotlinlang.org/> (Accessed: April 29, 2022).

22. Lashfire.Eu, Lashfire.eu. Available at: <https://lashfire.eu/> (Accessed: May 7, 2022).

23. Marketwired (2014) Hyper-local social messaging app yik Yak raises \$1.5 million in funding, Yahoo Finance. Available at: https://finance.yahoo.com/news/hyper-local-social-messaging-app-120000737.html?guccounter=1&guce_referrer=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnLw&guce_referrer_sig=AQAAAEsi8WV_YWBWF930ZLV0m04Zg1BuJfrHG3d3zfw4MXo3hkVANtqSgRWHHcBZjWuBMrKgw6VaPlnlZSQC09G5nr_3TP-eBWFyPHs2f4lg93F62YNq99uLUTVcFsYr0US-RlddE9ZsR547hozU1L-AVnmtjEoFF4y7sSyghnabUbd2 (Accessed: April 30, 2022).

24. Meta Platforms and Technologies | Meta, Facebook.com. Available at: <https://about.facebook.com/technologies/> (Accessed: May 1, 2022).

25. Mobile sensing (2020) movisens GmbH. Available at: <https://www.movisens.com/en/solutions/mobile-sensing/> (Accessed: May 10, 2022).

26. Mpeis, P. et al. (2020) “The Anyplace 4.0 IoT Localization Architecture,” in Proceedings of the 21st IEEE International Conference on Mobile Data Management. IEEE Computer Society, pp. 218–225. Available at: <https://www.cs.ucy.ac.cy/~dzeina/papers/mdm20-a4iot.pdf> (Accessed: May 2, 2022).

27. Nielsen, J. (2020) 10 usability heuristics for user interface design, Nielsen Norman Group. Available at: <https://www.nngroup.com/articles/ten-usability-heuristics/> (Accessed: May 4, 2022).

28. Niu, Q. et al. (2019) “Resource-efficient and automated image-based indoor localization,” ACM transactions on sensor networks, 15(2), pp. 1–31.

29. NVIDIA Tesla V100, NVIDIA. Available at: <https://www.nvidia.com/en-gb/data-center/tesla-v100/> (Accessed: May 9, 2022).

30. Or, B. (2021) What is IMU?, Towards Data Science. Available at: <https://towardsdatascience.com/what-is-imu-9565e55b44c> (Accessed: March 21, 2022).
31. Poullose, A., Eyobu, O. S. and Han, D. S. (2019) “An indoor position-estimation algorithm using smartphone IMU sensor data,” IEEE access: practical innovations, open solutions, 7, pp. 11165–11177.
32. Rayzit: Rayz Your Message (2014) Ucy.ac.cy. Available at: <https://rayzit.cs.ucy.ac.cy/> (Accessed: May 1, 2022).
33. Redpin - indoor positioning for the rest of us, Redpin.org. Available at: <http://redpin.org/index.html> (Accessed: May 2, 2022).
34. Sears-Collins, A. (2020) “Yaw, pitch, and roll diagrams using 2D coordinate systems.” Available at: <https://automaticaddison.com/yaw-pitch-and-roll-diagrams-using-2d-coordinate-systems/> (Accessed: May 2, 2022).
35. Snapchat - The fastest way to share a moment! (2011) Snapchat.com. Available at: <https://www.snapchat.com/> (Accessed: May 1, 2022).
36. Telegram – a new era of messaging (2013) Telegram. Available at: <https://telegram.org/> (Accessed: May 1, 2022).
37. TensorFlow: Large-scale machine learning on heterogeneous systems (2015) TensorFlow. Available at: <https://www.tensorflow.org/> (Accessed: April 29, 2022).
38. Yik Yak, Inc, Yik Yak, Yikyak.com. Available at: <https://yikyak.com/> (Accessed: May 6, 2022).
39. YOLO: Real-time object detection, Pjreddie.com. Available at: <https://pjreddie.com/darknet/yolo/> (Accessed: April 29, 2022).
40. Zeinalipour-Yazti, D. and Laoudias, C. (2017) “The Anatomy of the Anyplace Indoor Navigation Service,” in ACM SIGSPATIAL Special. ACM Press, pp. 3–10. Available at: <https://www.cs.ucy.ac.cy/~dzeina/papers/sigspatial17-letter.pdf> (Accessed: May 2, 2022).