



Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών

Τομέας Τεχνολογίας Πληροφορικής  
και Υπολογιστών

**Ανάπτυξη εφαρμογής επαυξημένης  
πραγματικότητας για τη διαδικασία επιλογής  
προϊόντων σε μεγάλους αποθηκευτικούς χώρους**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΝΙΚΟΛΑΟΣ ΗΛΙΑΚΗΣ**

**Επιβλέπουσα :** Δήμητρα - Θεοδώρα Κακλαμάνη  
Καθηγήτρια Ε.Μ.Π.

Αθήνα, Οκτώβριος 2019





Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών

Τομέας Τεχνολογίας Πληροφορικής  
και Υπολογιστών

**Ανάπτυξη εφαρμογής επαυξημένης  
πραγματικότητας για τη διαδικασία επιλογής  
προϊόντων σε μεγάλους αποθηκευτικούς χώρους**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΝΙΚΟΛΑΟΣ ΗΛΙΑΚΗΣ**

**Επιβλέπουσα :** Δήμητρα - Θεοδώρα Κακλαμάνη  
Καθηγήτρια Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 30η Οκτωβρίου 2019.

.....  
Δήμητρα - Θεοδώρα Κακλαμάνη  
Καθηγήτρια Ε.Μ.Π.

.....  
Ιάκωβος Βενιέρης  
Καθηγητής Ε.Μ.Π.

.....  
Γεώργιος Ματσόπουλος  
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2019

.....  
**Νικόλαος Ηλιάκης**

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

Copyright © Νικόλαος Ηλιάκης, 2019.  
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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

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

## Περίληψη

Σκοπός της παρούσας διπλωματικής εργασίας ήταν η υλοποίηση μίας εφαρμογής για τον ευκολότερο εντοπισμό αντικειμένων σε μεγάλους αποθηκευτικούς χώρους, χρησιμοποιώντας τεχνολογίες επαυξημένης πραγματικότητας ως βασικό εργαλείο.

Η εφαρμογή πρόκειται για ένα εργαλείο πλοήγησης μέσα σε αποθηκευτικούς χώρους, όπως, και βοήθημα εντοπισμού αντικειμένων στα ράφια. Κύριο χαρακτηριστικό της εφαρμογής είναι το σκανάρισμα των barcodes μέσω των γυαλιών, χωρίς επιπλέον εξαρτήματα, όπως και οι φωνητικές εντολές που μπορεί να δώσει ο χρήστης για να έχει τα χέρια αποδεσμευμένα. Για την ανάπτυξη της χρησιμοποιήθηκε η μηχανή ανάπτυξης παιχνιδιών Unity και η πλατφόρμα ανάπτυξης Android Studio. Κύριο εργαλείο, ακόμα, αποτέλεσε και η Vuforia, μία μηχανή που συνδέεται στην Unity και είναι υπεύθυνη για το κομμάτι της επαυξημένης πραγματικότητας.

Ακολουθεί η περιγραφή της εφαρμογής, λεπτομέρειες για το σχεδιασμό και την ανάπτυξη της, όπως, και κάποιες από τις δυσκολίες που παρουσιάστηκαν.

## Λέξεις κλειδιά

Επαυξημένη Πραγματικότητα, Indoor Navigation, Μηχανή Ανάπτυξης Παιχνιδιών, Unity, Vuforia, Barcodes, Logistics, Voice Commands, Android, Android Studio, Java, C#



# **Abstract**

The purpose of this diploma thesis was to implement an application to help users locate objects in large warehouses, by using augmented reality technologies.

The application is a navigation tool in warehouses and a tool to locate objects in the shelves, as well. The key features of the application are scanning of barcodes using the smart glasses, without extra equipment and the voice commands that can be used to interact with the application, which makes it hands free. For the implementation we used the Unity game engine and the Android Studio development platform. The Vuforia sdk is another main tool used, which is responsible for the augmented reality features.

Below, we will see the description of the application, details of the design and the development and the difficulties we encountered.

## **Key words**

Augmented Reality, Indoor Navigation, Game Engine, Unity, Vuforia, Barcodes, Logistics, Voice Commands, Android, Android Studio, Java, C#





## Ευχαριστίες

Ευχαριστώ θερμά την επιβλέπουσα καθηγήτρια αυτής της διπλωματικής εργασίας, κ. Δήμητρα Κακλαμάνη, για την εμπιστοσύνη που μου έδειξε κατά την ανάθεση. Τις θερμές μου ευχαριστίες στον Σπύρο, που τόσα χρόνια μετά ακόμα συμβάλλει στην εκπαίδευση μου, για την ανεκτίμητη βοήθεια του. Θέλω να ευχαριστήσω ακόμα τον Νίκο, ο οποίος με βοήθησε με κάθε τρόπο στα διάφορα στάδια αυτής της εργασίας, όπως, και όλη την ομάδα του VR για την υποστήριξη και τις συμβουλές τους. Δε θα μπορούσα να παραλείψω σε καμία περίπτωση να ευχαριστήσω τους φίλους μου, και ιδιαίτερα την οικογένειά μου και κυρίως τους γονείς μου, οι οποίοι με υποστήριξαν και έκαναν δυνατή την απερίσπαστη ενασχόλησή μου τόσο με την εκπόνηση της διπλωματικής εργασίας μου, όσο και συνολικά με τις σπουδές μου.

Νικόλαος Ηλιάκης,  
Αθήνα, 30η Οκτωβρίου 2019



# Περιεχόμενα

Περίληψη . . . . .	5
Abstract . . . . .	7
Ευχαριστίες . . . . .	9
Περιεχόμενα . . . . .	11
Κατάλογος πινάκων . . . . .	15
Κατάλογος σχημάτων . . . . .	17
<b>1. Σκοπός της Διπλωματικής Εργασίας . . . . .</b>	<b>19</b>
<b>2. Επαυξημένη Πραγματικότητα . . . . .</b>	<b>21</b>
2.1 Εισαγωγή . . . . .	21
2.2 Μικτή Πραγματικότητα . . . . .	22
2.3 Ιστορική Αναδρομή . . . . .	23
2.4 Τρέχουσα Τεχνολογία και Προβλήματα . . . . .	24
2.4.1 Φορητές Συσκευές Απεικόνισης . . . . .	24
2.4.2 Φορητές Συσκευές . . . . .	27
2.4.3 Προβολικές Συσκευές . . . . .	27
2.4.4 Συντονισμός . . . . .	28
2.5 Εφαρμογές . . . . .	29
2.6 Εφαρμογές στα logistics . . . . .	30
2.6.1 Pick-by-Vision: Βελτιωμένη Συλλογή . . . . .	30
2.6.2 Σχεδιασμός Αποθήκης . . . . .	31
2.6.3 Βελτιστοποίηση Μεταφοράς . . . . .	32
<b>3. Η μηχανή ανάπτυξης παιχνιδιών Unity . . . . .</b>	<b>35</b>
3.1 Γενικά . . . . .	35
3.2 Προγραμματισμός σε Unity - Δομή του πρότζεκτ . . . . .	36
3.2.1 Assets . . . . .	36
3.2.2 Hierarchy . . . . .	36
3.2.3 Scene View . . . . .	37
3.2.4 Game View . . . . .	37
3.2.5 Inspector . . . . .	37
3.3 Στοιχεία αντικειμένων (Components) . . . . .	37
3.3.1 Transform . . . . .	37
3.3.2 Camera . . . . .	38

3.3.3	Meshes	38
3.3.4	Rigidbody	39
3.3.5	Collider	39
3.3.6	Materials - Shaders	39
3.3.7	Scripts	40
3.4	Prefabs	41
<b>4.</b>	<b>Vuforia SDK</b>	<b>43</b>
4.1	Εισαγωγή	43
4.2	Χαρακτηριστικά της Vuforia	43
4.2.1	Tracking	43
4.2.2	Targets	44
4.2.3	Extended Tracking	44
4.2.4	Αναγνώριση Εικόνας	45
4.2.5	Αναγνώριση Αντικειμένων	46
4.2.6	Αναγνώριση Κειμένου	46
4.3	Vuforia στη Unity	47
4.3.1	ARCamera	48
4.3.2	Image	48
4.3.3	Multi Targets	50
4.3.4	Ground Plane - Mid Air	50
<b>5.</b>	<b>Android</b>	<b>51</b>
5.1	Εισαγωγή	51
5.2	Η αρχιτεκτονική του Android	51
5.2.1	Linux Kernel	51
5.2.2	Libraries	52
5.2.3	Applications	53
5.3	Ανατομία μίας Android εφαρμογής	53
5.3.1	Συστατικά εφαρμογών	53
5.3.2	Ενεργοποίηση Συστατικών	54
5.3.3	Το Αρχείο Manifest	54
5.3.4	Resources	55
5.4	Κύκλος ζωής Activity	55
5.5	Android Studio	57
<b>6.</b>	<b>Πλοήγηση Εσωτερικού Χώρου</b>	<b>59</b>
6.1	Εισαγωγή	59
6.1.1	Inertial navigation system	59
6.1.2	Θέση μέσω του Wi-fi	59
6.1.3	Bluetooth	60
6.1.4	Computer Vision	60
6.2	Σχεδιασμός και Υλοποίηση	60
6.2.1	Σχεδιασμός	60
6.2.2	Υλοποίηση	61

<b>7. Η εφαρμογή</b> . . . . .	67
7.1 Εισαγωγή . . . . .	67
7.2 Περιγραφή της διαδικασίας picking . . . . .	67
7.3 Παρουσίαση της εφαρμογής . . . . .	69
7.3.1 Σκανάρισμα του barcode της παραγγελίας . . . . .	69
7.3.2 Αλλαγή πακέτου προς συλλογή . . . . .	71
7.3.3 Αλλαγή παραγγελίας . . . . .	72
7.3.4 Χάρτης της αποθήκης . . . . .	72
7.3.5 Αλλαγή ποσότητας . . . . .	74
7.3.6 Σκανάρισμα των κωδικών των προϊόντων . . . . .	76
7.3.7 Ρυθμίσεις της εφαρμογής . . . . .	77
7.3.8 Φωνητικές Εντολές . . . . .	79
<b>8. Σύνοψη και μελλοντικές επεκτάσεις</b> . . . . .	81
<b>Βιβλιογραφία</b> . . . . .	83



## **Κατάλογος πινάκων**





## Κατάλογος σχημάτων

2.1	Προσομοίωση επαυξημένης πραγματικότητας στην ιατρική . . . . .	22
2.2	Απλοποιημένη αναπαράσταση του εικονικού συνεχούς . . . . .	23
2.5	Ένα σχέδιο των δύο συσκευών . . . . .	27
2.6	Το γνωστό AR παιχνίδι PokemonGo που κυκλοφόρησε για smartphones . . . . .	28
2.7	Μία σύνοψη των πλεονεκτημάτων του AR στα logistics . . . . .	31
2.8	Μετάφραση κειμένου σε πραγματικό χρόνο . . . . .	33
3.1	Το περιβάλλον ανάπτυξης της Unity . . . . .	36
3.2	Transform settings . . . . .	37
3.3	Camera settings . . . . .	38
3.4	Mesh settings . . . . .	38
3.5	Rigidbody settings . . . . .	39
3.6	Collider settings . . . . .	39
3.7	Mesh settings . . . . .	40
4.1	Τέσσερα σχήματα που δηλώνουν τι είναι φυσικό χαρακτηριστικό κατά τη Vuforia. Αξίζει να σημειωθεί ότι στο τέταρτο σχήμα δεν υπάρχει χαρακτηριστικά αφού δεν υπάρχουν γωνίες. . . . .	44
4.2	Το μενού που επιτρέπει να προσθέσεις Targets . . . . .	45
4.3	Με το extended Tracking ενεργοποιημένο, η Vuforia μπορεί να κάνει track τους targets ακόμα και όταν είναι εκτός του πλάνους της κάμερας. . . . .	45
4.4	Η εικόνα ενός ασθενοφόρου με υψηλό δείκτη εντοπισμού. Τα κίτρινα σημεία είναι τα φυσικά χαρακτηριστικά. . . . .	46
4.5	Τοποθετώντας το αντικείμενο που πρόκειται να σκαναριστεί πάνω στον scanner target, η Vuforia μπορεί να προσδιορίσει τους XYZ άξονες και να βρει τα σημεία ενδιαφέροντος. Το πλήθος τους φαίνεται στην πάνω αριστερή γωνία. . . . .	46
4.6	Μία απλή επαύξηση με τον εντοπισμό του προηγούμενου αντικειμένου χωρίς τη βοήθεια κάποιου target. . . . .	47
4.7	Η Vuforia μπορεί να αναγνωρίσει και να ακολουθεί κείμενο σαν να ήταν ένας οποιοσδήποτε target. . . . .	47
4.8	Δύο εικόνες με διαφορετικό πλήθος χαρακτηριστικών και πως αυτές αξιολογούνται. . . . .	48
4.9	A Multi Target . . . . .	50
4.10	Ένα 3D μοντέλο αστροναύτη εμφανίζεται στο δάπεδο. . . . .	50
5.1	Η αρχιτεκτονική Android . . . . .	52
5.2	Κύκλος ζωής ενός Activity . . . . .	56
5.3	Το περιβάλλον ανάπτυξης Android Studio . . . . .	57
6.1	Wifi Fingerprinting . . . . .	59

6.2	Bluetooth Beacons . . . . .	60
6.3	Ο Χάνσελ και η Γκρέτελ στο ομώνυμο παραμύθι αφήνουν πίσω τους ψίχουλα για να βρουν τον δρόμο της επιστροφής. . . . .	61
6.4	Οι διαθέσιμες επιλογές του World Center Mode. . . . .	62
6.5	Ο κώδικας που παίρνει τις συντεταγμένες του prefab που τοποθετήθηκε και τις στέλνει στο Firebase . . . . .	63
6.6	Αποθήκευση δεδομένων σε Firebase που αποτελεί No-SQL βάση. . . . .	63
7.1	Μία λίστα συλλογής - Picking List . . . . .	68
7.2	Η πρώτη οθόνη της εφαρμογής . . . . .	69
7.4	Σκανάρισμα του barcode της παραγγελίας . . . . .	70
7.5	Αρχικοποίηση μίας κλάσης γραμμένης σε Java μέσω Unity api . . . . .	70
7.6	Οι πληροφορίες που απεικονίζονται στον χρήστη μέχρι να φτάσει στη θέση του πακέτου που πρέπει να συλλέξει . . . . .	71
7.7	Ο χρήστης μπορεί να δει όλα τα πακέτα προς συλλογή της παραγγελίας και μπορεί να διαλέξει ποίο θέλει να συλλέξει πρώτα. . . . .	72
7.8	Ο χρήστης μπορεί να σταματήσει την τρέχουσα παραγγελία και να σκανάρει το barcode κάποιας άλλης για να την ξεκινήσει . . . . .	73
7.9	Ο χάρτης της αποθήκης στον οποίο φαίνεται η σχετική θέση του χρήστη με το πακέτο προς συλλογή. . . . .	74
7.10	Η οθόνη που επιλέγει ο χρήστης την ποσότητα που θα συλλέξει. . . . .	75
7.11	Η ερώτηση για επιβεβαίωση διακοπής της διαδικασίας σκαναρίσματος με λιγότερα πακέτα συλλεγμένα. . . . .	75
7.13	Όταν δεν απαιτείται σειριακό σκανάρισμα, αρκεί το σκανάρισμα ενός μονάχα έγκυρου barcode. . . . .	77
7.14	Οι ρυθμίσεις της εφαρμογής. . . . .	77
7.15	Μία συσκευή ring scanner. . . . .	78

## Κεφάλαιο 1

# Σκοπός της Διπλωματικής Εργασίας

Σκοπός της παρούσας διπλωματικής εργασίας είναι η βελτίωση της διαδικασίας συλλογής αντικειμένων σε μεγάλους αποθηκευτικούς χώρους (order picking). Τη βελτίωση αυτή θα προσπαθήσουμε να την πετύχουμε εντάσσοντας τεχνολογίες επαυξημένης πραγματικότητας στην ήδη υπάρχουσα διαδικασία. Πιο συγκεκριμένα, αναπτύξαμε μία εφαρμογή επαυξημένης πραγματικότητας η οποία τρέχει σε ένα ζευγάρι έξυπνων γυαλιών και δίνει τη δυνατότητα στους χρήστες να σκανάρουν barcodes, όπως, και να αλληλεπιδρούν μαζί της μέσω φωνητικών εντολών για να έχουν τα χέρια αποδεσμευμένα για να επικεντρωθούν στη συλλογή αντικειμένων.

Όσον αφορά τη δομή της εργασίας, αρχικά θα μελετήσουμε την Επαυξημένη Πραγματικότητα για να καταλάβουμε τι ακριβώς είναι και τι δυνατότητες μας προσφέρει. Στα επόμενα κεφάλαια θα μελετήσουμε τα προγραμματιστικά εργαλεία που χρησιμοποιήσαμε. Πιο αναλυτικά, στο κεφάλαιο τρία θα μιλήσουμε για τη Unity, μία πλατφόρμα ανάπτυξης παιχνιδιών και εφαρμογών. Στο κεφάλαιο τέσσερα θα δούμε τη Vuforia, η οποία είναι η βιβλιοθήκη η οποία είναι υπεύθυνη για το κομμάτι της Επαυξημένης Πραγματικότητας. Στο επόμενο κεφάλαιο θα παρουσιάσουμε το Android Studio, την επίσημη πλατφόρμα ανάπτυξης Android εφαρμογών. Στη συνέχεια, θα ακολουθήσει η περιγραφή μίας υλοποίησης για πλοήγηση σε εσωτερικούς χώρους η οποία, ωστόσο, δεν ενσωματώθηκε στην τελική εφαρμογή, αλλά, φάνηκε χρήσιμο να αναφερθεί. Το επόμενο κεφάλαιο αφορά την περιγραφή του σχεδιασμού της εφαρμογής, όπως, και μία περιγραφή αυτής και των δυνατοτήτων της. Τέλος, θα γίνει μία μικρή σύνοψη και θα παρουσιαστούν κάποιες δυνατές επεκτάσεις.



## Κεφάλαιο 2

# Επαυξημένη Πραγματικότητα

## 2.1 Εισαγωγή

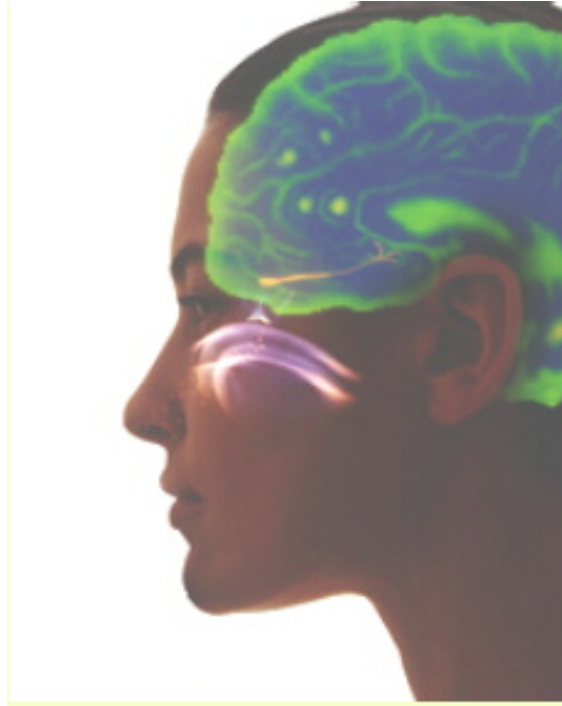
Με τον όρο Επαυξημένη Πραγματικότητα (Augmented Reality) ονομάζουμε τον εμπλουτισμό του πραγματικού κόσμου με εικονικά μοντέλα τα οποία παράγονται από τον υπολογιστή. Δηλαδή, ο χρήστης μπορεί να αισθανθεί πληροφορία η οποία πραγματικά δεν υπάρχει. Ο όρος πολλές φορές συγχέεται με αυτόν της Εικονικής Πραγματικότητας. Η λέξη «εμπλουτισμός» είναι η λέξη κλειδί που βοηθάει στη σύγχυση.

Όσο συναφείς, τόσο ως τεχνολογίες όσο και ως ιδέες, αποτελούν δύο διαφορετικές μεταξύ τους έννοιες. Στην εικονική πραγματικότητα ο χρήστης έρχεται σε επαφή με ένα αλληλεπιδραστικό τρισδιάστατο περιβάλλον, το οποίο παράγεται εξ' ολοκλήρου από υπολογιστή. Κατά τη διαδικασία αυτή υπάρχει πλήρης αποκοπή του χρήστη από τον πραγματικό κόσμο και δημιουργείται η ψευδαίσθηση πως βρίσκεται σε κάποιον άλλο κόσμο. Αντίθετα, στην Επαυξημένη Πραγματικότητα τα μοντέλα που παράγονται από τον υπολογιστή συνδυάζονται αρμονικά με τον πραγματικό κόσμο, σαν να είναι μέρος αυτού. Η Επαυξημένη πραγματικότητα (Augmented Reality) είναι ένα ταχέως αναπτυσσόμενο ερευνητικό πεδίο στο χώρο της εικονικής ή ιδεατής πραγματικότητας (Virtual Reality) [Vall]. Ο όρος επινοήθηκε από τον Tom Caudell, το 1990 και αναφέρεται στην προσθήκη εικονικής πληροφορίας, με τη χρήση κατάλληλων συσκευών, στο περιβάλλον που αντιλαμβάνεται ο άνθρωπος μέσω των αισθήσεων του. Η λέξη «προσθήκη» σηματοδοτεί ότι το πραγματικό περιβάλλον δεν υποκαθίσταται αλλά αντίθετα ενισχύεται και επαυξάνεται με πρόσθετες πληροφορίες από εκείνες που λαμβάνονται αποκλειστικά μέσω των ανθρώπινων αισθήσεων (όραση, ακοή κλπ).

Ο όρος «εικονική πραγματικότητα» αποδίδεται στον Jason Lanier, τον ιδρυτή της VPL Research, μια από τις πρώτες εταιρείες που ασχολήθηκαν με την κατασκευή τέτοιων συστημάτων. Μπορεί να οριστεί ως ένα αλληλεπιδραστικό (interactive), το οποίο παράγει εξ ολοκλήρου από υπολογιστή και στο οποίο ο χρήστης μπορεί να εμπυθιστεί [Auks92]. Στην εικονική πραγματικότητα σκοπός είναι η πλήρης αποκοπή του χρήστη από τον πραγματικό περίγυρο και η αντικατάσταση αυτού από μιας απολύτως ρεαλιστικής ψευδαίσθησης παραγμένης από τον υπολογιστή. Η δυνατότητα να αλληλεπιδρά με το τεχνητό περιβάλλον με φυσικό και άμεσο τρόπο είναι απαραίτητη προϋπόθεση για να παραμείνει ρεαλιστική η ψευδαίσθηση.

Ακολουθούν οι βασικές διαφορές και ομοιότητες μεταξύ εικονικής και επαυξημένης πραγματικότητας.

- Το περιβάλλον που εμβαθύνεται ο χρήστης στην εικονική πραγματικότητα είναι αποκλειστικά παραγμένο από τον υπολογιστή και ελέγχεται από αυτόν. Αντίθετα, στην επαυξημένη πραγματικότητα, ο χρήστης χρειάζεται να διατηρεί την επαφή με το πραγματικό περιβάλλον του. Στην εικονική πραγματικότητα ο χρήστης αλληλεπιδρά αποκλειστικά με εικόνες που παράγει ο υπολογιστής, ενώ στην επαυξημένη αυτές συνδυάζονται με την πραγματική



**Σχήμα 2.1:** Προσομοίωση επαυξημένης πραγματικότητας στην ιατρική

εικόνα του περίγυρου.

- Στην εικονική πραγματικότητα, η ρεαλιστική και πιστή απεικόνιση ενός μη υπαρκτού κόσμου είναι ένα από τα κυριότερα προβλήματα. Ωστόσο, στην επαυξημένη οι δύο κόσμοι πρέπει να συνδυαστούν σωστά, ώστε η σύνδεση τους να είναι αρμονική (μια διαδικασία που λέγεται συντονισμός ή εγγραφή).

- Αμφότερες οι τεχνολογίες έχουν εστιάσει μέχρι στιγμής κυρίως στην όραση, με τις υπόλοιπες αισθήσεις να έχουν ακόμα δευτερεύοντα ρόλο. Για το λόγο αυτό, οι κατεξοχήν χρησιμοποιούμενες συσκευές απεικόνισης είναι τα Head Mounted Displays - HMD με διαφορετικό σκοπό για κάθε μία (αποκοπή από περιβάλλον και επαύξηση). Η άνθηση της αγοράς των smartphones, τα τελευταία χρόνια, έχει οδηγήσει σε μεγαλύτερη διάδοση της επαυξημένης πραγματικότητας και την σταδιακή διεξόδυση της στην ανθρώπινη καθημερινότητα.

## 2.2 Μικτή Πραγματικότητα

Ο Milgram το 1994 όρισε το συνεχές πραγματικότητας-εικονικότητας (reality-virtuality continuum). Στο ένα άκρο βρίσκεται το πραγματικό περιβάλλον ενώ στο άλλο ένα πλήρως εικονικό, όπως παρουσιάζεται στην Εικόνα 2.2 [Kish94]. Ανάμεσα στα δύο αυτά άκρα υπάρχουν οι αποχρώσεις της μικτής πραγματικότητας (mixed reality). Αυτή μπορεί να αναλυθεί περαιτέρω σε επαυξημένη πραγματικότητα και «επαυξημένη εικονικότητα» (Augmented Virtuality, ένας όρος που δημιουργήθηκε από τον Milgram). Η πρώτη βρίσκεται κοντά στο άκρο της πραγματικότητας, καθώς η κυρίαρχη αντίληψη που μεταφέρεται προς τον χρήστη είναι αυτή του πραγματικού κόσμου, επαυξημένου με δεδομένα από υπολογιστή. Αντίθετα η επαυξημένη εικονικότητα, βρίσκεται πλησιέστερα στην εικονική πραγματικότητα και περιγράφει συστήματα τα οποία παρουσιάζουν κυρίως συνθετικές εικόνες με προσθήκη κάποιων στοιχείων από το πραγματικό περιβάλλον. Η επαυξημένη εικονικότητα δεν έχει ουσιαστικές εφαρμογές και αναμένεται να εκλείψει ως κατηγορία, όσο η τεχνολογία βελτιώνεται και αυ-

ξάνεται η ρεαλιστικότητα των εικονικών σκηνών. Το σημαντικό στοιχείο στην ταξινόμηση αυτή είναι το γεγονός ότι οι κατηγορίες της μικτής πραγματικότητας δεν αντιμετωπίζονται ως διακριτά σημεία, αλλά σαν περιοχές ενός συνεχούς.



Σχήμα 2.2: Απλοποιημένη αναπαράσταση του εικονικού συνεχούς

## 2.3 Ιστορική Αναδρομή

Η αρχή της Επαυξημένης πραγματικότητας φαίνεται να έγινε μεταξύ 1957 και 1962, όταν ο κινηματογραφιστής Morton Heiling εφηύρε το Sensorama, ένα προσομοιωτή που περιελάμβανε ήχο, δόνηση και οσμές. Αργότερα, το 1965 ο Ivan Sutherland δημοσίευσε το άρθρο του «The Ultimate Display», όπου περιγράφει πως η απόλυτη οθόνη δε θα είναι τίποτα άλλο, παρά ένα δωμάτιο εντός του οποίου η ύπαρξη της ύλης θα ελέγχεται από τον υπολογιστή [Suth65].

Λίγο αργότερα, με τον φοιτητή του, Bob Sproull, κατασκεύασε την πρώτη φορητή συσκευή απεικόνισης (Head-Mounted Display) εικονικής και επαυξημένης πραγματικότητας [Suth68]. Επρόκειτο για μία συσκευή που ήταν σχεδιασμένη να φοριέται στο κεφάλι και να απεικονίζει σε κάθε μάτι μια δισδιάστατη στερεοσκοπική εικόνα, με στόχο ο εγκέφαλος να τις συνδυάσει σε μία τρισδιάστατη προοπτική. Καθώς ο χρήστης θα κινούνταν, θα ανιχνεύονταν η θέση και ο προσανατολισμός του κεφαλιού και θα μεταβάλλοντας αντίστοιχα η προβαλλόμενη εικόνα. Όμως, το βάρος της έκανε απαγορευτική τη χρήση της και χρειάστηκε να κρεμιέται από το ταβάνι. Ακόμη, τα τεχνικά μέσα της εποχής επέτρεπαν την απεικόνιση μόνο των περιγραμμάτων των αντικειμένων ενώ ο υπολογιστής δεν μπορούσε να κρύψει αντικείμενα ή τμήματα αντικειμένων τα οποία δεν θα έπρεπε να φαίνονται (occlusion) με αποτέλεσμα ο χρήστης να δυσκολεύεται να προσδιορίσει το σχήμα και τη θέση τους.

Στις επόμενες δεκαετίες μικρές ομάδες από την Αμερικανική Πολεμική Αεροπορία, τη NASA, και κάποια πανεπιστημιακά ιδρύματα ασχολήθηκαν με την επαυξημένη πραγματικότητα χωρίς να ονομάζεται έτσι [Wiki]. Το 1975 ο Myron Krueger δημιούργησε για πρώτη φορά ένα εργαστήριο τεχνητής πραγματικότητας με ονομασία Videoplace, το οποίο περιβάλλει του χρήστες και αποκρίνεται στις κινήσεις τους, χωρίς να απαιτείται η χρήση ειδικών γυαλιών. Το 1989 ο Jaron Lanier εισήγαγε τον όρο εικονική πραγματικότητα ενώ τρία χρόνια αργότερα ο Tom Caudell χρησιμοποίησε για πρώτη φορά τον όρο επαυξημένη πραγματικότητα. Την ίδια χρονιά ο L.B. Rosenberg ανέπτυξε το πρώτο σύστημα επαυξημένης πραγματικότητας. Η τεχνολογική εξέλιξη της τελευταίας εικοσαετίας αφαίρεσε πολλά εμπόδια και μετέτρεψε την επαυξημένη πραγματικότητα σε ένα αυτόνομο ερευνητικό πεδίο. Το 1997, ο Ronald Azuma δημοσίευσε το άρθρο του με τίτλο «A survey of Augmented Reality» οριοθετώντας τον χώρο και συνοψίζοντας τα μέχρι τότε επιτεύγματα και προβλήματα [Azum97]. Έκτοτε έχουν γίνει μεγάλα βήματα προόδου τα οποία περιλαμβάνουν, μεταξύ άλλων, φο-

ρητά παιχνίδια, ταξιδιωτικούς οδηγούς σε κινητά τηλέφωνα νέας γενιάς αλλά και συστήματα πλοήγησης που στηρίζονται στην τεχνολογία της επαυξημένης πραγματικότητας [hitl].

## 2.4 Τρέχουσα Τεχνολογία και Προβλήματα

Ο σκοπός της επαυξημένης πραγματικότητας, όσον αφορά αποκλειστικά στην οπτική επαύξηση μιας σκηνής, δεν περιορίζεται μόνο στην προσθήκη αντικειμένων σε ένα πραγματικό τοπίο αλλά ενίοτε περιλαμβάνει και την απόκρυψη αντικειμένων του τοπίου, σε πραγματικό χρόνο, δίνοντας ταυτόχρονα στο χρήστη τη δυνατότητα αλληλεπίδρασης με το εικονικό περιβάλλον. Για να γίνουν αυτά, οποιοδήποτε σύστημα επαυξημένης πραγματικότητας πρέπει να εκτελέσει τα εξής:

- Παρουσίαση του πραγματικού περιβάλλοντος.
- Δημιουργία των κατάλληλων εικονικών αντικειμένων, τα οποία προβάλλονται επικαλύπτοντας το πραγματικό σκηνικό, αποσκοπώντας είτε στην προβολή επιπλέον αντικειμένων/πληροφοριών, είτε στην απόκρυψη πραγματικών αντικειμένων.
- Συντονισμός του πραγματικού σκηνικού με τα εικονικά αντικείμενα, δηλαδή, με τη θέση όπου περιμένει ο χρήστης να δει τα εικονικά αντικείμενα.
- Απεικόνιση της εικονικής πληροφορίας πάνω στην πραγματική.
- Ανίχνευση της κίνησης και της θέσης του χρήστη ή των επιλογών που αυτός κάνει και ανάλογη μεταβολή της απεικόνισης.

Από τα παραπάνω προκύπτει ότι ο πλέον απαραίτητος εξοπλισμός ενός συστήματος εικονικής πραγματικότητας είναι η συσκευή απεικόνισης, η οποία εμφανίζει τη μίξη του εικονικού και πραγματικού περιβάλλοντος. Οι συσκευές αυτές διακρίνονται στις εξής κατηγορίες: φορητές, φορητές και προβολικές.

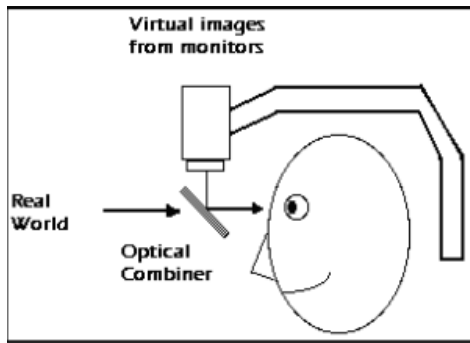
### 2.4.1 Φορητές Συσκευές Απεικόνισης

Πρόκειται για την πιο διαδεδομένη συσκευή απεικόνισης. Είναι οι συσκευές που φοριούνται στο κεφάλι και προβάλλουν εικόνες στα μάτια του χρήστη. Υπάρχουν δύο τύποι: οι οπτικές συσκευές και οι βιντεοσυσκευές.

Οι οπτικές συσκευές απεικόνισης (optical see-through displays), έχουν μια διάφανη οθόνη και επιτρέπουν στον χρήστη να δει το πραγματικό περιβάλλον όπως είναι, προβάλλοντας τα εικονικά αντικείμενα πάνω της. Η συνηθέστερη προσέγγιση για την κατασκευή οπτικών συσκευών περιλαμβάνει τη χρήση ενός ημιανακλαστικού κατόπτρου το οποίο μπορεί ταυτόχρονα να ανακλά και να μεταδίδει το φως. Αν τοποθετηθεί σωστά μπροστά από το μάτι του χρήστη, μπορεί να ανακλά μια εικόνα από μια οθόνη υπολογιστή μέσα στο οπτικό πεδίο του, ενώ ταυτόχρονα επιτρέπει τη διέλευση του φωτός του περιβάλλοντα χώρου. Με την τοποθέτηση κατάλληλων φακών μεταξύ του ημιανακλαστικού κατόπτρου και της οθόνης του υπολογιστή, η εικόνα εστιάζεται σε μια βολική για τον χρήστη απόσταση. Εάν για κάθε μάτι χρησιμοποιηθεί ένα ξεχωριστό σύστημα ημιανακλαστικού κατόπτρου, δίνεται η δυνατότητα στο χρήστη να βλέπει στερεοσκοπικά.

Οι βιντεοσυσκευές απεικόνισης (video see-through displays) όπως παρουσιάζονται στις εικόνες που ακολουθούν, αποκόπτουν τελείως τον χρήστη από το περιβάλλον. Συνδυάζουν την εικόνα που καταγράφεται από μία κάμερα που φοράει ο χρήστης με γραφικά από υπολογιστή. Το πραγματικό περιβάλλον καταγράφεται από την κάμερα, ψηφιοποιείται και παράγεται τελικά μια επαυξημένη εικόνα του περιβάλλοντος. Η συνδυασμένη εικόνα του κόσμου θα προσεγγίζει τη συνηθισμένη οπτική γωνία του χρήστη. Όπως και στις οπτικές συσκευές,



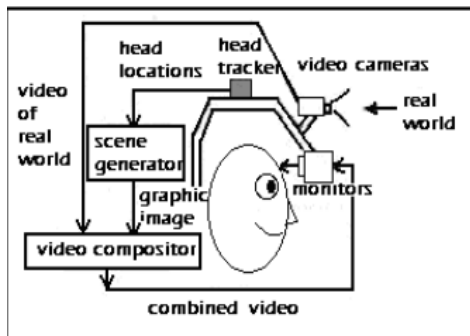


(a) Ο τρόπος λειτουργίας μιας οπτικής συσκευής



(b) Μία οπτική συσκευή

αν υπάρχει ξεχωριστό σύστημα (κάμερα-οθόνη) για κάθε μάτι, τότε ο χρήστης θα έχει τη δυνατότητα στερεοσκοπικής όρασης, όπως και στην κανονική ζωή.



(a) Ο τρόπος λειτουργίας μιας βίντεο συσκευής



(b) Μία βίντεο συσκευή

Η κάθε προσέγγιση έχει συγκεκριμένα πλεονεκτήματα και μειονεκτήματα. Οι οπτικές συσκευές υπερτερούν στα ακόλουθα σημεία:

- **Απλότητα:** η οπτική μίξη είναι απλούστερη και φθηνότερη ως κατασκευή, από τη μίξη δύο σημάτων βίντεο. Τα δύο σήματα πρέπει να είναι συγχρονισμένα, αλλιώς θα υπάρχει χρονική παραμόρφωση. Επίσης η ψηφιοποίηση των εικόνων του πραγματικού κόσμου είναι χρονοβόρος διαδικασία, με αποτέλεσμα να υπάρχει καθυστέρηση (της τάξης των χιλιοστών του δευτερολέπτου) ανάμεσα σε αυτό που βλέπει ο χρήστης και σε αυτό που πραγματικά καταγράφεται από την κάμερα.

- **Ανάλυση:** στις βιντεοσυσκευές απεικόνισης η ανάλυση (ευκρίνεια και λεπτομέρεια) της εικόνας που βλέπει ο χρήστης περιορίζεται στην ανάλυση των συσκευών καταγραφής και απεικόνισης, η οποία μέχρι στιγμής είναι μικρότερη από της αναλυτική ικανότητα του

αμφιβληστροειδούς χιτώνα του ματιού. Έτσι, η τελική εικόνα δε φαίνεται «φυσική». Στις οπτικές συσκευές, η εικόνα των αντικειμένων του πραγματικού περιβάλλοντος δεν υποβαθμίζεται ούτε αλλοιώνεται.

- **Ασφάλεια:** αν σε μια συσκευή βίντεο υπάρξει κάποια βλάβη που να προκαλέσει την παύση της λειτουργίας της, ο χρήστης αποκόπτεται εντελώς από το περιβάλλον του και καθίσταται τυφλός. Σε κάποιες εφαρμογές, αυτό δημιουργεί μεγάλα προβλήματα ασφαλείας. Αντίθετα, σε αντίστοιχη περίπτωση, μια οπτική συσκευή θα γινόταν απλώς ένα βαρύ ζευγάρι γυαλιά, επιτρέποντας στον χρήστη να μένει σε επαφή με το πραγματικό περιβάλλον, έστω και χωρίς την επαυξημένη πραγματικότητα.

- **Δεν υπάρχει μετατόπιση του οπτικού πεδίου:** στις συσκευές βίντεο η οπτική του χρήστη είναι αυτή που παρέχεται από τις κάμερες. Αυτό σημαίνει ότι η θέση των ματιών του μετατοπίζεται στη θέση των καμερών. Καθώς οι κάμερες δεν βρίσκονται συνήθως στη θέση των ματιών και η απόσταση μεταξύ τους μπορεί να διαφέρει από αυτή των ματιών, δημιουργείται μια μετατόπιση ανάμεσα σε αυτό που βλέπει ο χρήστης σε σχέση με αυτό που είχε συνηθίσει να βλέπει. Αυτό μπορεί να αποφευχθεί με χρήση κατάλληλων τεχνικών που επιτρέπουν στις κάμερες να «δουν» όπως θα έβλεπε ο χρήστης αυξάνοντας όμως το κόστος και την πολυπλοκότητα της συσκευής.

- **Ευελιξία στη σύνθεση του πραγματικού με το εικονικό.** Στις οπτικές συσκευές, τα εικονικά αντικείμενα δεν μπορούν να αποκρύψουν πλήρως τα πραγματικά, καθώς επιτρέπουν στο φως από το περιβάλλον να περνά. Η κατασκευή μιας οπτικής συσκευής που να επιτρέπει στην επιλεκτική δίοδο του φωτός είναι πολύ δύσκολη και ακριβή. Ως αποτέλεσμα, οι οπτικές συσκευές δεν μπορούν ουσιαστικά να αφαιρέσουν πραγματικά αντικείμενα και τα εικονικά αντικείμενα εμφανίζονται ημιδιαφανή, κάτι το οποίο υποσκάπτει την αίσθηση του ρεαλισμού και την πιστότητα. Αντίθετα, στις βιντεοσυσκευές όπου και οι δύο κόσμοι υπάρχουν σε ψηφιακή μορφή, είναι πολύ πιο εύκολη η επεξεργασία και αποτελεσματική σύνθεση της τελικής εικόνας.

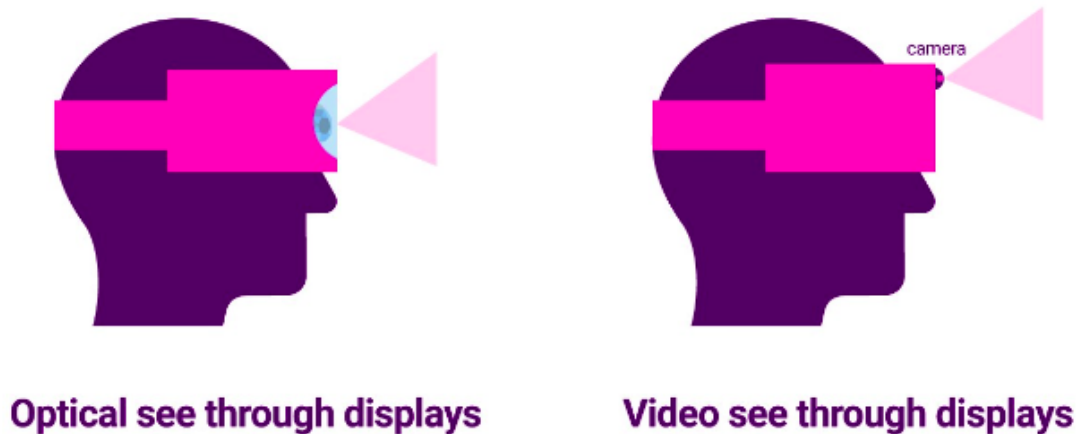
- **Χρονικές καθυστερήσεις:** οποιεσδήποτε χρονικές διαφορές μεταξύ πραγματικών και εικονικών σκηνών μπορούν να απαλειφθούν στις βιντεοσυσκευές, καθώς η εικόνα του πραγματικού περιβάλλοντος μπορεί να καθυστερήσει μέχρι να ολοκληρωθεί η επεξεργασία του εικονικού και να προβληθούν ταυτόχρονα και συγχρονισμένα, κάτι που δεν μπορεί να γίνει στις οπτικές συσκευές.

- **Συντονισμός:** το μοναδικό στοιχείο που διαθέτει ένα οπτικό σύστημα για τον συντονισμό (registration) του εικονικού με το πραγματικό περιβάλλον είναι η θέση του κεφαλιού και η τοποθεσία του χρήστη. Αντίθετα, μια βιντεοσυσκευή έχει στη διάθεση της και την ψηφιοποιημένη εικόνα του περιβάλλοντος. Αυτό σημαίνει πως με ειδικές τεχνικές επεξεργασίας εικόνας, αναγνώρισης προτύπων (pattern recognition) κλπ., μπορούν να αντληθούν επιπλέον στοιχεία για την βελτίωση της διαδικασίας συντονισμού.

- **Αντίθεση και φωτεινότητα:** ιδανικά θα έπρεπε η φωτεινότητα των εικονικών και των πραγματικών αντικειμένων να είναι ίδια. Καμία συσκευή απεικόνισης σήμερα δεν έχει τέτοιο δυναμικό εύρος αντίστοιχο με αυτό του ματιού, με αποτέλεσμα σε μια οπτική συσκευή να μη φαίνονται καθόλου τα εικονικά αντικείμενα, αν το περιβάλλον φως είναι πολύ έντονο, ενώ αν είναι πολύ σκοτεινό τα πραγματικά αντικείμενα να είναι δυσδιάκριτα. Το πρόβλημα είναι λιγότερο στις βιντεοσυσκευές καθώς και οι δύο κόσμοι προβάλλονται μέσα από μία μονάδα απεικόνισης και τα πάντα μετατρέπονται στο δικό της δυναμικό εύρος, χωρίς τα πολύ φωτεινά ή τα πολύ σκοτεινά αντικείμενα να διαφέρουν ιδιαίτερα μεταξύ τους σε ένταση [Prygo98].

Μια εντελώς διαφορετική προσέγγιση είναι η απεικόνιση στον αμφιβληστροειδή (virtual retinal display), η οποία σχηματίζει εικόνες απευθείας πάνω από τον αμφιβληστροειδή του ματιού.

Ανεξάρτητα από τον τρόπο προβολής των φορητών συσκευών, ο τελικός σκοπός είναι να μην είναι βαρύτερες από ένα ζευγάρι γυαλιά. Κάποια πρωτότυπα μοντέλα έχουν το μέγεθος και τις διαστάσεις γυαλιών οράσεως, στα κρύσταλλα των οποίων έχει ενσωματωθεί ένα πρίσμα που αντανακλά την εικόνα μιας πολύ μικρής, κατάλληλα τοποθετημένης οθόνης LCD. Ο υπόλοιπος βλέπει μόνο ένα διαφανή φακό, χωρίς να υπάρχει καμία ένδειξη ότι προβάλλεται πάνω του κάποια εικόνα.



Σχήμα 2.5: Ένα σχέδιο των δύο συσκευών

### 2.4.2 Φορητές Συσκευές

Πρόκειται για φορητές επίπεδες οθόνες οι οποίες έχοντας μια ενσωματωμένη κάμερα παρέχουν επαύξηση του περιβάλλοντος. Ο πραγματικός κόσμος λαμβάνεται μέσω μιας κάμερας και προβάλλεται στην οθόνη, η οποία παρουσιάζει και τα εικονικά αντικείμενα επικαλύπτοντας τα πραγματικά. Αυτό σημαίνει ότι ο χρήστης δεν εμβυθίζεται σε ένα επαυξημένο περιβάλλον, αλλά, παρατηρεί τη σύνθεση εικονικού και πραγματικού μέσω της οθόνης του συστήματος.

Όλες οι συσκευές αυτής της κατηγορίας υλοποιούν οπτικές τεχνικές για την υπέρθεση των εικονικών πληροφοριών στον πραγματικό κόσμο. Διάφοροι αισθητήρες όπως πυξίδες, επιταχυνσιόμετρα και GPS ενσωματώνονται για την ανίχνευση του περιβάλλοντος. Τα κινητά τηλέφωνα νέας γενιάς πληρούν όλες τις προϋποθέσεις για υπέρθεση μιας και ενσωματώνουν πληθώρα αισθητήρων.

### 2.4.3 Προβολικές Συσκευές

Πρόκειται για συσκευές οι οποίες προβάλλουν απευθείας την εικονική πληροφορία πάνω στα φυσικά αντικείμενα. Στην απλούστερη περίπτωση η πληροφορία προβάλλεται πάνω στο αντικείμενο, απλό ένα συνηθισμένο προβολικό μηχάνημα (Projector). Στο άλλο άκρο της κατηγορίας βρίσκονται συστήματα, όπως το CAVE, όπου, σε ένα ειδικό δωμάτιο υπάρχουν πολλαπλά προβολικά που μπορούν να καλύψουν την επιφάνεια των τοίχων, χρησιμοποιώντας αυτόματους προβολικούς μηχανισμούς, ώστε να μεταβάλλεται η κάλυψη, η εστίαση κλπ [Rask99]. Καταγράφοντας με κάμερες το πραγματικό περιβάλλον και προβάλλοντας τον



**Σχήμα 2.6:** Το γνωστό AR παιχνίδι PokemonGo που κυκλοφόρησε για smartphones

συνδυασμό πραγματικού και εικονικού, που παράγεται από υπολογιστές, στις επιφάνειες των τοίχων μέσω των προβολικών, ο χρήστης μπορεί να έχει μια πολύ ρεαλιστική επαύξηση της πραγματικότητας, η οποία τον περικλείει. Με κατάλληλο χειριστήριο δίνεται στο χρήστη η δυνατότητα αλληλεπίδρασης, επιτρέποντας του να ελέγχει και να δίνει εντολές στο σύστημα.

Μια ενδιαφέρουσα εφαρμογή των προβολικών συσκευών είναι ο συνδυασμός τους με ειδικά ανακλαστικά υλικά. Χρησιμοποιώντας ένα τέτοιο ανακλαστικό επίχρισμα σε ένα πραγματικό αντικείμενο και προβάλλοντας μια εικόνα του περιβάλλοντος χωρίς το αντικείμενο αυτό, μπορούμε σχεδόν να αποκρύψουμε το πραγματικό αντικείμενο, καθιστώντας το ημιδιαφανές.

#### 2.4.4 Συντονισμός

Ένα από τα βασικότερα προβλήματα που αντιμετωπίζει η επαυξημένη πραγματικότητα αφορά στη σωστή ταύτιση της εικονικής πληροφορίας με τα αντικείμενα του πραγματικού κόσμου, όπως τον βιώνει ο χρήστης. Σε αντίθετη περίπτωση, η ψευδαίσθηση της συνύπαρξης των δύο κόσμων καταστρέφεται. Το φαινόμενο αυτό είναι πολύ πιο έντονο στην επαυξημένη παρά στη εικονική πραγματικότητα, γιατί η πηγή του προβλήματος έγκειται στη ευαισθησία του ανθρώπινου οπτικού συστήματος. Αντίθετα, στην εικονική πραγματικότητα ο χρήστης βλέπει μόνο τον εικονικό κόσμο και οι δυσκολίες προκύπτουν από τις διαφορές οπτικού και κιναισθητικού συστήματος.

Για να επιτευχθεί συντονισμός, το σύστημα πρέπει να γνωρίζει κάθε στιγμή τη θέση και την κατεύθυνση του κεφαλιού του χρήστη καθώς και τη γενικότερη θέση του, αν αυτός βρίσκεται σε εξωτερικό περιβάλλον. Για το σκοπό αυτό χρησιμοποιούνται ανιχνευτές αδρανείας (γυροσκόπια και επιταχυνσιόμετρα), σε συνδυασμό με ειδικούς ανιχνευτές, οι οποίοι λαμβάνουν σήματα από κατάλληλους σηματοδότες, όπως LED ή υπέρηχους, τοποθετημένους σε

διάφορα σημεία στον χώρο, όταν πρόκειται για εσωτερικά περιβάλλοντα. Συνδυάζοντας τα δεδομένα από αυτές τις κατηγορίες ανιχνευτών, η ακρίβεια του εντοπισμού της θέσης και επακόλουθα ο συντονισμός βελτιώνονται κατά πολύ και επιτυγχάνονται ιδιαίτερα ικανοποιητικά αποτελέσματα.

Όλα αυτά λειτουργούν πολύ καλά όταν πρόκειται για κλειστούς χώρους, στους οποίους έχουν τοποθετηθεί κατάλληλοι σηματοδότες και αισθητήρες. Όσον αφορά στους εξωτερικούς χώρους, έχουν χρησιμοποιηθεί γυροσκόπια και επιταχυνσιόμετρα σε συνδυασμό με μαγνητόμετρο, το οποίο μετρά το μαγνητικό πεδίο της γης [Fein97]. Η εύρεση της θέσης γίνεται με έναν δέκτη υψηλής ακρίβειας του γνωστού συστήματος GPS. Η ακρίβεια του εμπορικά διαθέσιμου GPS, το οποίο στηρίζεται στη λήψη σημάτων από δορυφόρους, είναι της τάξης των μερικών μέτρων, γεγονός που το καθιστά ακατάλληλο. Για το λόγο αυτό έχει χρησιμοποιηθεί μια τεχνική γνωστή ως «διαφορικό GPS» στην οποία ο δέκτης δέχεται, εκτός από τα σήματα των δορυφόρων, και σήματα από έναν άλλον δέκτη GPS και από έναν επίγειο σταθμό. Με τον τρόπο αυτό επιτυγχάνεται ακρίβεια της των εκατοστών. Το πρόβλημα αυτής της μεθόδου έγκειται στο ότι τα σήματα των δορυφόρων εξασθενούν ή χάνονται όταν ο χρήστης βρίσκεται σε κλειστούς χώρους ή περιβάλλεται από ψηλά κτίρια.

## 2.5 Εφαρμογές

Το πεδίο εφαρμογής της επαυξημένης πραγματικότητας μπορεί να περιλαμβάνει πολλούς τομείς της επιστήμης, της τεχνολογίας, αλλά και της καθημερινής ζωής.

- Στην ιατρική: οι περισσότερες προσπάθειες εφαρμογής στο χώρο της ιατρικής στρέφονται στην υποβοήθηση των εγχειρήσεων. Εικόνες από ακτινογραφίες ή αξονικές/μαγνητικές τομογραφίες μπορούν να προβάλλονται στο σωστό σημείο του ασθενούς καθώς η εγχείρηση βρίσκεται σε εξέλιξη. Επίσης, ο γιατρός μπορεί να βλέπει μια τρισδιάστατη απεικόνιση ενός εμβρύου. Η εικόνα θα εμφανίζεται σαν να μπορούσε να δει ο γιατρός μέσα στην κοιλιά και θα αλλάζει κατάλληλα ανάλογα με τη γωνία παρατήρησης. Οι μελλοντικές εξελίξεις όμως στην επαυξημένη πραγματικότητα μπορεί να συμπεριλαμβάνουν την επαύξηση και άλλων αισθήσεων.

- Στη διασκέδαση/ενημέρωση: η επαυξημένη πραγματικότητα έχει ήδη εφαρμοστεί από τους τηλεοπτικούς σταθμούς στα δελτία καιρού. Οι χάρτες που βλέπουμε είναι όλοι ψηφιακοί, ο παρουσιαστής στέκεται μπροστά σε μια μπλε ή πράσινη οθόνη και το σύστημα συνδυάζει κατάλληλα το πραγματικό με τη ψηφιακό ώστε να έχουμε μια ζωντανή πρόγνωση.

- Η συσκευασμένη τεχνολογία εφαρμόζεται ήδη και κατά την αναμετάδοση αγώνων. Ο ηλεκτρονικός πίνακας διαφημίσεων έχει αναπτύξει ένα σύστημα επαυξημένης πραγματικότητας, που επιτρέπει στους σταθμούς να παρεμβάλουν διαφημίσεις σε συγκεκριμένες περιοχές της μεταδιδόμενης εικόνας. Αρχικά απαιτείται η βαθμονόμηση του σταδίου λαμβάνοντας εικόνες από χαρακτηριστικές γωνίες και εστιάσεις καμερών προκειμένου να δημιουργηθεί ένας χάρτης του χώρου συμπεριλαμβανομένων και των θέσεων όπου θα προβληθούν οι διαφημίσεις. Με τη χρήση των προκαθορισμένων σημείων αναφοράς του γηπέδου, το σύστημα καθορίζει αυτόματα τη γωνία της κάμερας που χρησιμοποιείται και παρεμβάλλει τη διαφήμιση στη σωστή θέση.

- Τα παιχνίδια σε υπολογιστές είναι ένας ακόμα τομέας στον οποίο η επαυξημένη πραγματικότητα θα μπορεί να προσφέρει ένα άλλο επίπεδο ρεαλισμού, όπου πραγματικοί παίκτες και εικονικοί κόσμοι θα συνδυάζονται σε μια σκηνή. Ακόμη και στον τομέα της εκπαίδευσης μπορεί να χρησιμοποιηθεί για τη δημιουργία εικονικών αντικειμένων σε μουσεία, εκθέσεις ακόμα και βιβλία.

- Στον στρατό: εφοδιάζοντας τους μελλοντικούς στρατιώτες με ειδικά κράνη επαυξημένες πραγματικότητας, μπορούν να υπερτεθούν πληροφορίες στον πραγματικό κόσμο:

- Για τη μορφολογία τους εδάφους
- Για τον αντίπαλο σχηματισμό
- Για μη ορατές από την συγκεκριμένη οπτική λεπτομέρειες όπως πληροφορίες από δορυφόρους ή από άλλους στρατιώτες.
- Για κρίσιμα σημεία και αδυναμίες του αντιπάλου

- Στον βιομηχανικό σχεδιασμό, στις επισκευές και στις κατασκευές: δίνοντας τη δυνατότητα σε μια ομάδα μηχανικών σε διαφορετικά σημεία του πλανήτη να εργάζονται πάνω στο ίδιο μοντέλο, βλέποντας ο καθένας μια απεικόνιση του μαζί με τις τροποποιήσεις που προτείνουν οι υπόλοιποι. Ένας συντηρητής θα μπορεί να βλέπει το ελαττωματικό εξάρτημα μιας μηχανής με έντονο χρώμα, σε σχέση με τα υπόλοιπα, ενώ δίπλα εμφανίζονται οι τεχνικές προδιαγραφές τους.

- Στην ρομποτική και στην τηλε-ρομποτική: ένας χειριστής τηλε-ρομποτικού μηχανισμού μπορεί να χρησιμοποιήσει της εικόνα από έναν μακρινό χώρο εργασίας για να καθοδηγήσει ένα ρομπότ. Δεδομένου ότι συχνά η άποψη της μακρινής εικόνας είναι μονοσκοπική, η επαύξηση με συρματομόρφα μοντέλα μπορεί να μετατρέψει την μακρινή εικόνα σε τρισδιάστατη γεωμετρία. Έτσι όταν ο χειριστής κάνει μία κίνηση, αυτή θα μπορεί να ασκηθεί σε ένα εικονικό ρομπότ που απεικονίζεται σαν επαύξηση στην πραγματική σκηνή. Αφού δει τα αποτελέσματα, ο χειριστής μπορεί να αποφασίσει να συνεχίσει με μια επόμενη κίνηση. Στην συνέχεια, θα μπορεί να εκτελεστεί άμεσα η κίνηση από το ρομπότ.

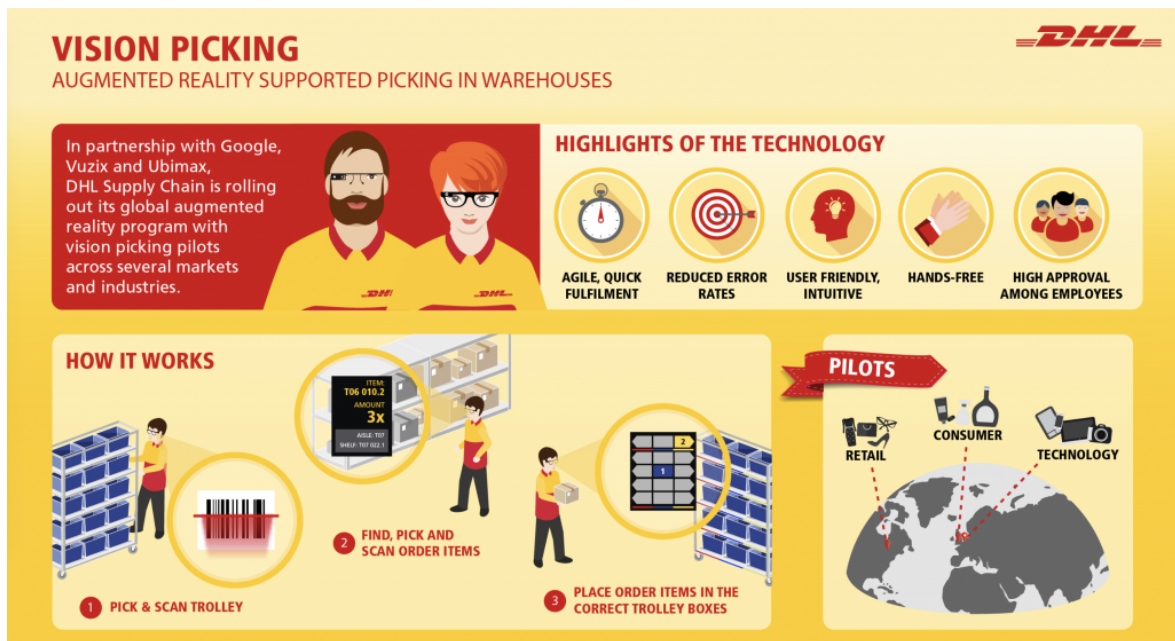
## 2.6 Εφαρμογές στα logistics

Το AR φαίνεται ως τώρα αρκετά υποσχόμενο για logistics και λειτουργίες αποθηκών. Οι λειτουργίες αυτές εκτιμάται ότι κοστίζουν είκοσι τις εκατό των συνολικών εξόδων για logistics, και η διαδικασία εύρεσης και φόρτωσης αντικειμένων να κοστίζει από πενήντα πέντε έως εξήντα πέντε τις εκατό των λειτουργιών αυτών. Αυτό υποδηλώνει ότι το AR έχει τη δυνατότητα να μειώσει σημαντικά το κόστος βελτιώνοντας τη διαδικασία εύρεσης και φόρτωσης. Μπορεί ακόμα να βοηθήσει στην εκπαίδευση των νέων εργαζομένων και στο προσωρινό προσωπικό, όπως και στον προγραμματισμό της αποθήκης [DHLL].

### 2.6.1 Pick-by-Vision: Βελτιωμένη Συλλογή

Στα logistics, στο AR οι πιο απτές λύσεις είναι συστήματα για τη βελτίωση της συλλογής των αντικειμένων. Η συντριπτική πλειοψηφία των αποθηκών στον αναπτυσσόμενο κόσμο ακόμα χρησιμοποιούν χαρτί για τη συλλογή αντικειμένων. Αλλά, κάθε διαδικασία βασισμένη στο χαρτί είναι αργή και επιρρεπής σε λάθη. Ακόμα, αυτή η δουλειά αναλαμβάνεται συχνά από προσωρινό προσωπικό που χρειάζεται ακριβή εκπαίδευση για να βεβαιωθεί ότι είναι αποτελεσματικοί και αποφεύγουν τα λάθη.

Συστήματα αναπτυσσόμενα από την Knap, την SAP και την Ubimax βρίσκονται στις τελευταίες φάσεις ελέγχου σε ρεαλιστικές συνθήκες και αποτελούνται από κινητά AR συστήματα όπως head mounted displays, κάμερες, φορητό PC και πακέτα μπαταριών που αντέχουν τουλάχιστον μία βάρδια. Το λογισμικό οπτικής συλλογής προσφέρει αναγνώριση αντικειμένων



Σχήμα 2.7: Μία σύνοψη των πλεονεκτημάτων του AR στα logistics

σε πραγματικό χρόνο, διάβασμα barcode, πλοήγηση εσωτερικού χώρου και ενσωμάτωση με το διαχείρισης της αποθήκης (WMS). Ένα κύριο πλεονέκτημα είναι η δυνατότητα παροχής πληροφοριών χωρίς να δεσμεύονται τα χέρια τους όσο εκτελούν λειτουργίες συλλογής.

Χρησιμοποιώντας ένα τέτοιο σύστημα, κάθε εργαζόμενος μπορεί να δει μία ψηφιακή λίστα με τα αντικείμενα που πρέπει να συλλέξει και χάρη στο σύστημα εσωτερικής πλοήγησης να βρίσκει την γρηγορότερη διαδρομή μειώνοντας το χρόνο μεταφοράς μέσα στην αποθήκη. Χρησιμοποιώντας τις δυνατότητες αυτόματου διαβάσματος barcode, το σύστημα αναγνώρισης εικόνων μπορεί να ελέγξει αν ο εργαζόμενος έχει φτάσει στο σωστό σημείο και να το οδηγήσει γρηγορότερα στον εντοπισμό του σωστού αντικειμένου στο ράφι.

Στη συνέχεια, μπορεί ο εργαζόμενος να σκανάρει το αντικείμενο και να καταγράψει αυτόματα τη διαδικασία στο WMS, πραγματοποιώντας ενημερώσεις στη βάση σε πραγματικό χρόνο. Επίσης, τέτοια συστήματα μπορούν να μειώσουν τον χρόνο που χρειάζεται να εκπαιδευτούν οι νέοι υπάλληλοι, όπως και να λυθούν γλωσσικά χάσματα όταν πρόκειται για μετανάστες εργαζομένους.

Οι έλεγχοι αυτών των συστημάτων έχουν αποδείξει σημαντικές βελτιώσεις στις λειτουργίες. Για παράδειγμα, το σφάλμα στη συλλογή μπορεί να μειωθεί έως και σαράντα τις εκατό. Ενώ ακόμα και σήμερα, ακόμα και με χαρτί, το ποσοστό είναι ήδη αρκετά χαμηλό, ωστόσο κάθε σφάλμα πρέπει να αποφεύγεται καθώς προκύπτουν ζημίες για την εταιρεία.

## 2.6.2 Σχεδιασμός Αποθήκης

Το AR είναι πιθανό να επηρεάσει και τις διαδικασίες σχεδιασμού των αποθηκών. Σήμερα, οι αποθήκες δε χρησιμοποιούνται μόνο ως χώροι αποθήκευσης και διανομής, όλο και περισσότερες αυξάνουν τις υπηρεσίες τους. Οι υπηρεσίες αυτές αφορούν τη συναρμολόγηση προϊόντων, τιτλοφόρηση, συσκευασία και επιδιορθώσεις.

Αυτό σημαίνει ότι πρέπει να ξανασχεδιαστούν για να φιλοξενήσουν αυτές τις υπηρεσίες. Το AR μπορεί να χρησιμοποιηθεί για να οπτικοποιηθεί κάθε σχεδιασμένη εκ νέου διευθέτηση σε κανονική κλίμακα, αφού γίνεται δυνατό να τοποθετηθούν διαδραστικές ψηφιακές αναπαραστάσεις των προτεινόμενων μελλοντικών τροποποιήσεων στο τωρινό, πραγματικό

περιβάλλον την αποθήκης. Οι σχεδιαστές μπορούν να δοκιμάσουν αν οι μετρήσεις των σχεδιασμένων τροποποιήσεων ταιριάζουν στον χώρο και να δοκιμάσουν νέες ροές εργασιών.

### **2.6.3 Βελτιστοποίηση Μεταφοράς**

Κατά την τελευταία δεκαετία, η χρήση προηγμένων τεχνολογιών στα logistics έχει επηρεάσει σοβαρά την αποτελεσματικότητα, την αξιοπιστία και την ασφάλεια στη μεταφορά φορτίων. Το AR έχει τη δυνατότητα να βοηθήσει ακόμα περισσότερο την μεταφορά φορτίων σε τομείς όπως, έλεγχοι πληρότητας, παγκόσμιο εμπόριο, πλοήγηση οδηγών και τη φόρτωση.

#### **Έλεγχοι πληρότητας**

Το AR μπορεί να επιτύχει πιο αποτελεσματικό στη συλλογή προϊόντων. Ένας υπάλληλος εξοπλισμένος με μία συσκευή AR θα μπορούσε να κοιτάξει απλώς ένα φορτίο και να ελέγξει αν είναι ολοκληρωμένο. Προς το παρόν, απαιτείται χειρονακτική καταμέτρηση ή διάβασμα των barcode με κάποια συσκευή χειρός. Στο μέλλον, με μία φορητή AR συσκευή θα μπορούσε να χρησιμοποιηθεί ένας συνδυασμός από σκάνερς και αισθητήρες βάθους για να αποφανθεί το πλήθος των αντικειμένων ή ο όγκος τους. Αυτές οι μετρήσεις θα συγκρίνονται με τις προκαθορισμένες τιμές και το αποτέλεσμα θα προβάλλεται στον συλλέκτη. Αυτό το σύστημα θα μπορούσε να σκανάρει τα αντικείμενα για να εντοπίσει τυχόν ζημιές ή λάθη.

#### **Παγκόσμιο Εμπόριο**

Με τις περισσότερες περιοχές του κόσμου να ευδοκιμούν οικονομικά, οι μεταφορές που ρέουν από και προς αυτές τις αναδύμενες αγορές αυξάνονται σημαντικά. Αυτή η αύξηση αντιπροσωπεύει μία μεγάλη ευκαιρία για παρόχους logistics, αλλά, παράλληλα αυξάνει την πολυπλοκότητα, καθώς υπάρχουν σημαντικές αποκλίσεις στους εμπορικούς κανονισμούς και τις απαιτήσεις σε όλο τον κόσμο.

Το AR είναι πιθανό να αποδειχθεί πολύτιμο για τους παρόχους του παγκόσμιου εμπορίου. Πριν από μία αποστολή, ένα σύστημα AR θα μπορούσε να βοηθήσει εξασφαλίζοντας ότι είναι σύμφωνη με τις σχετικές ρυθμίσεις εισαγωγής και εξαγωγής, ή ότι τα εμπορικά έγγραφα έχουν συμπληρωθεί σωστά. Θα μπορούσε να ανιχνεύει τα εμπορικά έγγραφα ή τα εμπορεύματα για λέξεις κλειδιά και αυτόματα να αλλάζει ή να προτείνει τον κωδικό ταξινόμησης του προϊόντος.

Μετά την αποστολή, θα βοηθούσε να μειωθεί σημαντικά ο χρόνος στα λιμάνια και τι αποθήκες μεταφράζοντας τα εμπορικά έγγραφα σε πραγματικό χρόνο (πχ Word Lens).

#### **Δυναμική Κυκλοφοριακή Υποστήριξη**

Η Κυκλοφοριακή συμφόρηση συχνά αποτρέπει την ομαλή λειτουργία πολλών οικονομικών διαδικασιών. Υπολογίζεται ότι η συμφόρηση κοστίζει περίπου το ένα τις εκατό του ακαθάριστου εγχώριου προϊόντος (GDP) κάθε χρόνο. Όσο αυξάνεται η κίνηση υπάρχει μεγάλη ζήτηση για λύσεις που βελτιώνουν της ακρίβεια.

Στο μέλλον, θα βλέπουμε αυξημένη χρήση δυναμικής υποστήριξης με δεδομένα κίνησης πραγματικού χρόνου που θα βελτιστοποιεί τις διαδρομές και θα τις επαναπροσδιορίζει σε πραγματικό χρόνο. Μία εφαρμογή AR που βοηθάει βοηθάει τον οδηγό θα μπορούσε να χρησιμοποιηθεί για να παρέχει πληροφορίες για το οπτικό πεδίο του. Πιο συγκεκριμένα, τα AR συστήματα θα είναι οι απόγονοι των σημερινών συστημάτων πλοήγησης με το πλεονέκτημα





**Σχήμα 2.8:** Μετάφραση κειμένου σε πραγματικό χρόνο

κλειδί ότι ο οδηγός δεν πρέπει να απομακρύνει το βλέμμα του από τον δρόμο. Επίσης, θα μπορούσε να ενημερώνει τον οδηγό για κρίσιμες πληροφορίες για το όχημα και το φορτίο.



## Κεφάλαιο 3

# Η μηχανή ανάπτυξης παιχνιδιών Unity

### 3.1 Γενικά

Η Unity είναι μία μηχανή ανάπτυξης παιχνιδιών για κάθε είδος πλατφόρμας (cross-platform). Ωστόσο, το εύρος δυνατοτήτων της την κάνει ιδανική για ανάπτυξη όχι μόνο παιχνιδιών, αλλά, και εφαρμογών του βιομηχανικού ή εκπαιδευτικού τομέα. Η μηχανή αυτή κρίθηκε ιδανική, πρωτίστως, γιατί υποστηρίζει τη ανάπτυξη εφαρμογών για το hardware μας (vuzix m300 glasses). Όμως, η πληθώρα χρήσιμων εργαλείων ανάπτυξης, το εύχρηστο περιβάλλον εργασίας και η μεγάλη και ενεργή κοινότητα χρηστών έπαιξαν σημαντικό ρόλο στην επιλογή της.

Η Unity διατίθεται σε δύο εκδόσεις: την δωρεάν και την επαγγελματική (Unity Pro). Η δεύτερη προσφέρει περισσότερες δυνατότητες, οι οποίες αναφέρονται κυρίως σε εταιρείες που ασχολούνται με την ανάπτυξη μεγάλων και περίπλοκων παιχνιδιών. Η μεταφερσιμότητα είναι ένα από τα σημαντικότερα χαρακτηριστικά της Unity, αφού προσφέρεται η δυνατότητα έκδοσης παιχνιδιών για υπολογιστές (Windows/Mac), κονσόλες (PS4, Xbox), smartphones (iOS, Android), ακόμη και παιχνίδια φυλλομετρητών (browser games) με ελάχιστες αλλαγές στον κώδικα του παιχνιδιού. Το περιβάλλον της Unity είναι ιδιαίτερα εύχρηστο. Προσφέρονται όλα τα απαραίτητα εργαλεία για την κατασκευή του κόσμου της εφαρμογής. Νέο περιεχόμενο (Asset) μπορεί να εισαχθεί εύκολα με «σύρσιμο και απόθεση» (drag & drop) ενώ υπάρχει συμβατότητα με την πλειοψηφία των προγραμμάτων τρισδιάστατων γραφικών (Maya, 3DStudioMax, κα). Ακόμη, ανά πάσα στιγμή υπάρχει η δυνατότητα προεπισκόπησης της εφαρμογής. Κατά τη διάρκεια της προεπισκόπησης, όλες οι public μεταβλητές του κώδικα είναι προσβάσιμες και μπορούν άμεσα να επηρεαστούν, μεταβάλλοντας ταυτόχρονα και τη συμπεριφορά της εφαρμογής.

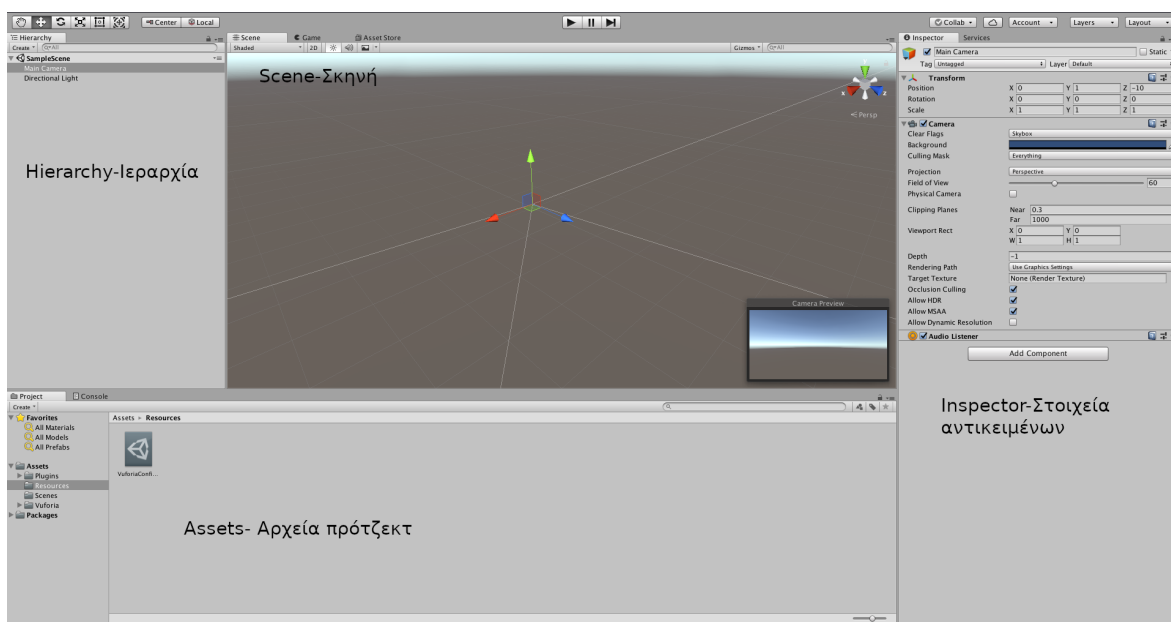
Στην ίδια την πλατφόρμα παρέχονται μεγάλος αριθμός tutorials, καθώς και παραδείγματα έτοιμων παιχνιδιών, που βοηθούν στην εξοικείωση μαζί της. Αξίζει ακόμα να σημειωθεί πως γύρω από το Unity έχει συσπειρωθεί μία πολύ ενεργή κοινότητα δημιουργών, πρόθυμη να παράσχει βοήθεια σε αρχάριους χρήστες. Ενδεικτικό της προσπάθειας αυτής είναι η ιστοσελίδα Unity Answers όπου χρήστες μπορούν να εκφράσουν τις απορίες τους.

Η ίδια η μηχανή Unity είναι τελευταίας τεχνολογίας και ανταγωνίζεται ακριβότερες μηχανές ανάπτυξης παιχνιδιών της αγοράς σε όλους τους τομείς (rendering, φωτισμός, επεξεργασία/ παραγωγή ήχου, μηχανή φυσικής, κλπ) Γενικά, αποτελεί είναι ένα από τα κορυφαία εργαλεία ανάπτυξης παιχνιδιών. Μάλιστα, χρησιμοποιείται και στον τομέα της παιδείας, σε προγράμματα εκπαίδευσης, σε εφαρμογές εικονικής πραγματικότητας και σε παρουσιάσεις.

## 3.2 Προγραμματισμός σε Unity - Δομή του πρότζεκτ

Μία εφαρμογή που αναπτύσσεται στο περιβάλλον της Unity ονομάζεται πρότζεκτ (Project). Ένα πρότζεκτ αποτελείται από μία ή περισσότερες σκηνές (Scenes), οι οποίες περιέχουν ένα ή περισσότερα αντικείμενα γνωστά ως GameObjects. Τα αντικείμενα βρίσκονται σε μία ιεραρχία στο πρότζεκτ. Η κατάλληλη τοποθέτηση των αντικειμένων στη σκηνή δημιουργεί το γραφικό περιβάλλον της εφαρμογής που αναπτύσσεται. Κάθε αντικείμενο αποτελείται από ένα ή περισσότερα στοιχεία (Components). Τα στοιχεία αυτά καθορίζουν τις ιδιότητες του αντικειμένου, την αλληλεπίδρασή του με το περιβάλλον της εφαρμογής και τη συμπεριφορά που θα έχει κατά τη διάρκεια εκτέλεσής της. Όλα τα αρχεία που μπορεί να χρειαστούν για την υλοποίηση του πρότζεκτ (π.χ. αρχεία ήχου, εικόνες, αρχεία τρισδιάστατων μοντέλων, κ.λπ.) αποθηκεύονται στον φάκελο Assets. Στο σχήμα 2.1 παρουσιάζεται το περιβάλλον της μηχανής γραφικών κατά τη δημιουργία ενός νέου πρότζεκτ.

Το κεντρικό περιβάλλον είναι ο Editor ο οποίος όπως βλέπουμε από διάφορα μικρότερα παράθυρα, τα λεγόμενα views. Καλή γνώση της λειτουργίας του κάθε view οδηγεί σε σωστή, γρήγορη και αποδοτική χρήση της Unity.



Σχήμα 3.1: Το περιβάλλον ανάπτυξης της Unity

### 3.2.1 Assets

Στα Assets εμφανίζονται όλα τα αρχεία που χρησιμοποιεί η εφαρμογή ταξινομημένα σε φακέλους. Μέσω των Assets γίνεται η οργάνωση του περιεχομένου είτε πρόκειται για scripts, γραφικά, μοντέλα, ήχους ή και ολόκληρες σκηνές (Scenes).

### 3.2.2 Hierarchy

Στο Hierarchy περιλαμβάνονται όλα τα GameObjects που απαρτίζουν την ενεργό σκηνή. Αν κάποιο GameObject προστεθεί ή διαγραφεί από τη σκηνή, θα προστεθεί ή θα διαγραφεί και από το Hierarchy. Από εδώ μπορούν να επιλεγθούν τα GameObject της σκηνής, να εντοπιστούν στον κόσμο του παιχνιδιού και να διαμορφωθούν. Το Unity χρησιμοποιεί την

έννοια του «Parenting», η οποία σημαίνει πως κάποιο GameObject μπορεί να οριστεί ως «παιδί» ή «πατέρας» κάποιου άλλου. Αυτό υλοποιείται πολύ απλά στο Hierarchy με σύρσιμο και απόθεση του παιδιού στον πατέρα. Το παιδί κληρονομεί το Transform του πατέρα του και αποκτά θέση σχετική με αυτόν. Ακόμη, μπορεί να αλληλεπιδράσει μαζί του μέσω ειδικών συναρτήσεων.

### 3.2.3 Scene View

Το Scene View είναι ίσως το πιο σημαντικό παράθυρο στου Editor. Όποτε φορτώνεται μία σκηνή, αυτή εμφανίζεται στο Scene View. Μέσω των εργαλείων που προσφέρει γίνεται η δημιουργία και επεξεργασία της σκηνής, τοποθετούνται όλα τα μοντέλα, οι κάμερες, τα φώτα κοκ.

### 3.2.4 Game View

Το Game View εμφανίζει μία προεπισκόπηση εκτέλεσης της υπό επεξεργασία εφαρμογής, μόλις πατηθεί το Play. Δίνεται έτσι η δυνατότητα στο δημιουργό να τρέξει την εφαρμογή μέσα από τον Editor χωρίς να κατασκευαστεί εκτελέσιμο αρχείο. Η οπτική γωνία προέρχεται από τις κάμερες που έχουν τοποθετηθεί στη σκηνή. Όλες οι αλλαγές που γίνονται στο παιχνίδι όσο είναι πατημένο το Play είναι προσωρινές και αναιρούνται μόλις σταματήσει η εκτέλεση.

### 3.2.5 Inspector

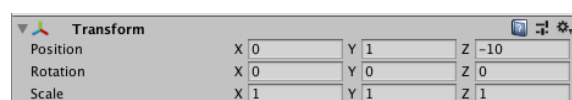
Τα παιχνίδια στο Unity αποτελούνται από πολλά GameObjects που περιλαμβάνουν με τη σειρά τους στοιχεία (Components) με τη μορφή κώδικα, ήχου ή άλλων ιδιοτήτων. Ο Inspector δείχνει λεπτομερείς πληροφορίες για τις ιδιότητες και τα στοιχεία του επιλεγμένου GameObject. Εδώ μπορούμε να διαμορφώσουμε τη συμπεριφορά ενός αντικειμένου μέσα στον κόσμο του παιχνιδιού προσθέτοντας, αφαιρώντας ή τροποποιώντας τα Components από τα οποία απαρτίζεται. Οποιαδήποτε ιδιότητα που φαίνεται στον Inspector μπορεί να τροποποιηθεί άμεσα χωρίς συγγραφή κώδικα. Οι public μεταβλητές των scripts είναι ορατές στον Inspector. Έτσι, μία μεταβλητή που ορίζεται στον κώδικα ενός script μπορεί να αρχικοποιηθεί μέσω του Inspector. Ακόμα, και κατά τη διάρκεια της προεσκόπησης, δίνεται η δυνατότητα επεξεργασίας των μεταβλητών χωρίς αλλαγές στον κώδικα, επιτρέποντας έτσι την εύρεση της τιμής στην οποία το εκάστοτε αντικείμενο έχει την επιθυμητή συμπεριφορά.

## 3.3 Στοιχεία αντικειμένων (Components)

Στην ενότητα αυτή θα περιγραφούν μερικά από τα πιο βασικά στοιχεία που μπορεί να περιέχονται σε κάποιο GameObject. Τα στοιχεία αυτά μπορούν να εμφανιστούν και να επεξεργαστούν μέσω του Inspector.

### 3.3.1 Transform

Αποτελεί ίσως το πιο βασικό χαρακτηριστικό ενός GameObject, δημιουργείται μαζί με το GameObject και βρίσκεται υποχρεωτικά σε αυτό. Το Transform είναι υπεύθυνο



Σχήμα 3.2: Transform settings

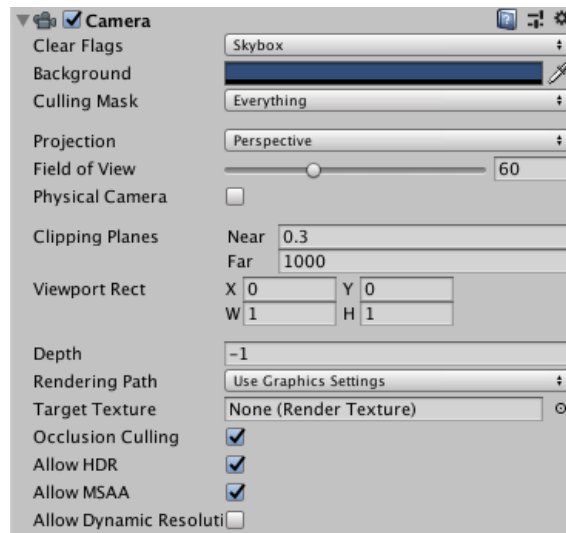
για την θέση, την περιστροφή και το μέγεθος του για τους τρεις άξονες x,y και z.Αξίζει να σημειωθεί πως αν το GameObject είναι παιδί κάποιου άλλου τότε και τα τρία στοιχεία είναι σχετικά προς αυτό.

### 3.3.2 Camera

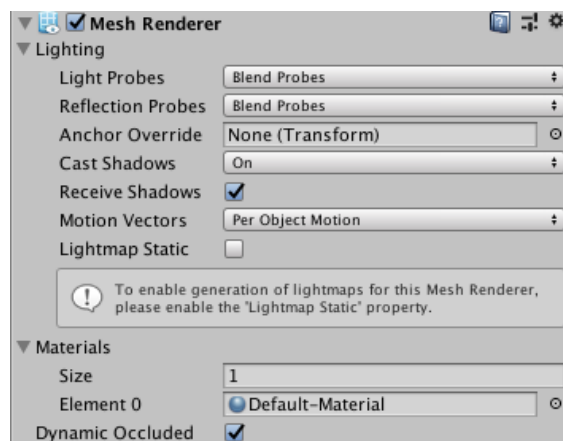
Η κάμερα είναι είναι το GameObject που κάνει δυνατή την προβολή της σκηνής στον χρήστη. Περιέχει μία σειρά από ρυθμίσεις οι οποίες αλλάζουν ανάλογα με τις ανάγκες της εφαρμογής. Αξίζει να σημειωθεί ότι μπορούν να υπάρχουν παραπάνω από μία στη σκηνή, αλλά, ανά πάσα στιγμή μόνο μία είναι ενεργή. Έτσι, η εμπειρία της εφαρμογής γίνεται πιο πλήρης και πλούσια.

### 3.3.3 Meshes

Η προβολή τρισδιάστατων μοντέλων στη σκηνή της Unity γίνεται μέσω δικτύων γεωμετρίας (Meshes). Αυτά σχεδιάζονται συνήθως με τη βοήθεια ειδικών προγραμμάτων μοντελοποίησης, όπως τα 3ds Max, Blender, Autodesk Inventor ή Cinema 4D. Η Unity υποστηρίζει συγκεκριμένους τύπους αρχείων γεωμετρίας, οι οποίοι είναι .obj, .fbx, .max, .blend, .dae και .dxf. Κατά τη δημιουργία ενός τρισδιάστατου αντικειμένου στη σκηνή της Unity, δημιουργούνται αυτόματα σε αυτό τα στοιχεία MeshFilter και MeshRenderer. Το πρώτο από αυτά είναι υπεύθυνο για το σχηματισμό της γεωμετρίας του μοντέλου, ενώ το δεύτερο για την προβολή του στη σκηνή και τις ιδιότητες εμφάνισης του (φωτισμός, αντανακλάσεις, σκιές, κλπ)



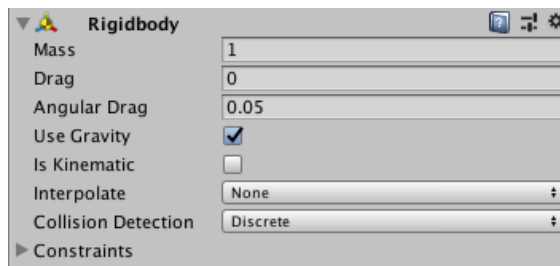
Σχήμα 3.3: Camera settings



Σχήμα 3.4: Mesh settings

### 3.3.4 Rigidbody

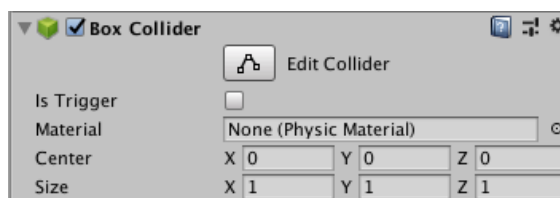
Το Rigidbody είναι το κύριο component που εκθέτει το GameObject στη μηχανή φυσικής της Unity. Η μηχανή φυσικής είναι ο μηχανισμός που κάνει τα GameObject να αποκρίνονται στους νόμους της φυσικής με ρεαλιστικό τρόπο, ώστε να υπάρχει αλληλεπίδραση μεταξύ τους, αλλά, και με τη σκηνή. Μόλις προστεθεί στο GameObject τότε αυτό κατευθείαν βρίσκεται υπό την επήρεια της βαρύτητας και των συγκρούσεων με άλλα GameObjects. Όπως, τα περισσότερα components προσφέρει μία γκάμα από ρυθμίσεις για τις εκάστοτε ανάγκες.



Σχήμα 3.5: Rigidbody settings

### 3.3.5 Collider

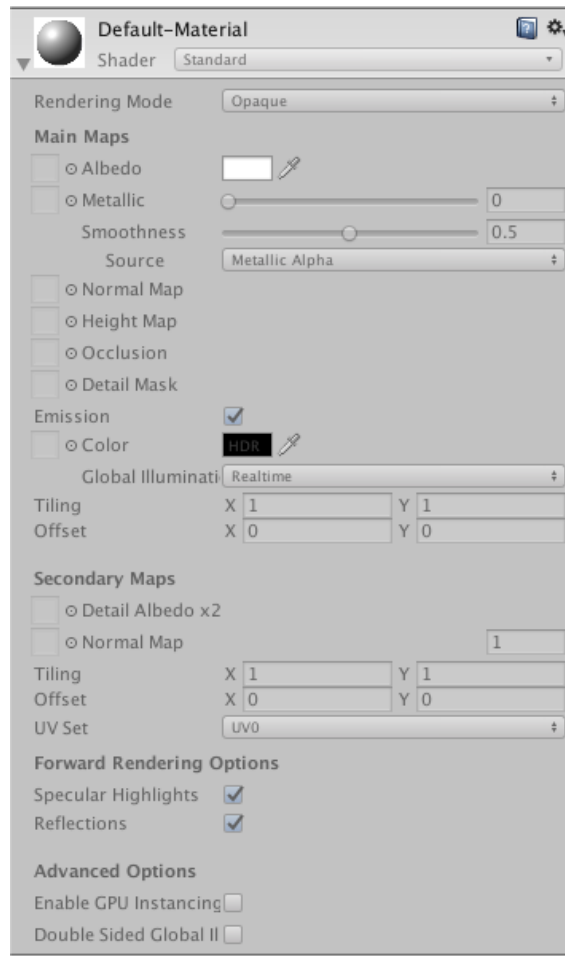
Το στοιχείο Collider καθορίζει τα όρια του αντικειμένου για την ανίχνευση συγκρούσεων. Ένας collider είναι αόρατος και δεν έχει κάποια σχέση με το δίκτυο γεωμετρίας του GameObject. Υπάρχουν διάφορα είδη από colliders, μερικά από τα οποία είναι οι Box Collider, Sphere Collider και Capsule Collider. Η χρήση επιπλέον ορίων για την ανίχνευση συγκρούσεων συμβαίνει επειδή συνήθως η γεωμετρία των GameObjects είναι αρκετά πολύπλοκη, γεγονός το οποίο θα οδηγούσε σε μεγάλη σπατάλη πόρων. Στις περισσότερες περιπτώσεις, η απλή γεωμετρία των στοιχείων collider είναι αρκετή για να προσεγγίζει ικανοποιητικά την πραγματική γεωμετρία του GameObject. Σε αντίθετες περιπτώσεις χρησιμοποιείται το στοιχείο Mesh Collider, το οποίο αντιστοιχεί ακριβώς στη πραγματική γεωμετρία του μοντέλου. Η περιορισμένη χρήση του στοιχείου Mesh Collider αρκετές φορές επιφέρει καλύτερα αποτελέσματα χωρίς να δημιουργεί προβλήματα έλλειψης υπολογιστικών πόρων. Σημειώνεται ότι είναι δυνατή η χρήση περισσότερων του ενός Collider σε ένα αντικείμενο, σε περιπτώσεις όπου η χρήση του ενός δεν δημιουργεί την επιθυμητή γεωμετρία.



Σχήμα 3.6: Collider settings

### 3.3.6 Materials - Shaders

Τα components Material και Shader παίζουν πολύ μεγάλο ρόλο στην εμφάνιση του αντικειμένου στη σκηνή. Αυτά τα δύο λειτουργούν σε συνδυασμό. Πιο συγκεκριμένα, το στοιχείο Material είναι υπεύθυνο για το χρώμα και την υφή που θα έχει ένα αντικείμενο, ενώ το στοιχείο Shader περιέχει μαθηματικούς υπολογισμούς που αφορούν τον υπολογισμό του χρώματος σε κάθε pixel του αντικειμένου, βασισμένο στον εκάστοτε φωτισμό της σκηνής και στις ρυθμίσεις του στοιχείου Material. Κατά την εισαγωγή ενός στοιχείου Material ορίζεται ποιο Shader θα χρησιμοποιηθεί. Συνεπώς, αναλόγως την επιλογή του Shader, οι ρυθμίσεις για το στοιχείο Material είναι διαφορετικές. Για τα περισσότερα GameObject της σκηνής, το Shader που χρησιμοποιείται είναι το Standard Shader. Το Shader αυτό παρέχει μία πληθώρα ρυθμίσεων το οποίο το καθιστά κατάλληλο για τη ρεαλιστική απεικόνιση των περισσότερων αντικειμένων. Για εξειδικευμένες περιπτώσεις χρησιμοποιούνται διαφορετικά Shaders τα οποία προσδίδουν ιδιαίτερα εφέ στο GameObject.



Σχήμα 3.7: Mesh settings

### 3.3.7 Scripts

Το βασικότερο component για τη συμπεριφορά του εκάστοτε GameObject είναι το Script. Τα Scripts ουσιαστικά αποτελούν κλάσεις στις οποίες περιέχονται κομμάτια κώδικα τα οποία συνήθως συντάσσει ο χρήστης. Οποιαδήποτε συμπεριφορά και έλεγχος αντικειμένου ή στοιχείου, μπορεί να υλοποιηθεί μέσω κώδικα. Οι υποστηριζόμενες γλώσσες στο περιβάλλον της Unity είναι η C# και η Javascript. Η βασική σύνδεση ενός script με τη Unity γίνεται μέσω της κλάσης MonoBehaviour. Για κάθε κλάση-script που δημιουργείται δηλώνεται αυτόματα ότι η κλάση αυτή κληρονομεί τη MonoBehaviour. Αυτό έχει ως αποτέλεσμα την αυτόματη δημιουργία των συναρτήσεων Start και Update στο εκάστοτε Script.

Η συνάρτηση Start() χρησιμοποιείται για την αρχικοποίηση διαφόρων μεταβλητών στο script. Η κλήση της συνάρτησης αυτής γίνεται μόνο μία φορά κατά την εκτέλεση της εφαρμογής και συγκεκριμένα όταν το script ενεργοποιηθεί για πρώτη φορά. Η Unity εξασφαλίζει ότι όλες οι συναρτήσεις Start θα έχουν κληθεί πριν από οποιαδήποτε εκτέλεση κάποιας συνάρτησης Update.

Η συνάρτηση Update() καλείται σε κάθε frame της εφαρμογής – παιχνιδιού. Ο αριθμός των frames ανά δευτερόλεπτο σε μία εφαρμογή ονομάζεται Frame Rate ή Frames Per Second (FPS). Ως FPS ορίζεται ο αριθμός των εικόνων που προβάλλονται στην οθόνη κάθε δευτερόλεπτο. Επομένως, η συνάρτηση Update() χρησιμοποιείται για την υλοποίηση συμπεριφορών στις οποίες είναι απαραίτητος ο συνεχής έλεγχος κάποιων καταστάσεων.

Για παράδειγμα, η μετακίνηση ενός παίκτη με τη χρήση εισόδου από το πληκτρολόγιο,



απαιτεί συνεχώς να ελέγχεται αν κάποιο από τα κουμπιά μετακίνησης έχει πατηθεί. Εάν ο έλεγχος αυτός γίνεται στη συνάρτηση Update(), το αποτέλεσμα θα είναι μία ομαλή εκτέλεση του παιχνιδιού, όπως είναι επιθυμητό.

Η κλάση MonoBehaviour προσφέρει επιπλέον κάποιες σημαντικές συναρτήσεις που μπορούν να χρησιμοποιηθούν, όμως, δεν δηλώνονται αυτόματα κατά τη δημιουργία των script. Μερικές από αυτές είναι οι Awake, FixedUpdate, LateUpdate, OnGUI, OnEnable και OnDisable. Εκτός από τις παραπάνω συναρτήσεις, ο χρήστης μπορεί να δημιουργήσει τις δικές του, τις οποίες θα πρέπει να καλέσει ξεχωριστά.

## 3.4 Prefabs

Τα Prefabs είναι πρότυπα αντικείμενα που αποθηκεύονται κάτω από τον ομώνυμο φάκελο μέσα στα Assets. Η σημαντικότερη ιδιότητα τους είναι η δυνατότητα δημιουργίας πολλαπλών στιγμιοτύπων (instances) του αρχικού αντικειμένου, τα οποία διατηρούν όλες τις ιδιότητες του πρωτοτύπου. Η δημιουργία των στιγμιοτύπων μπορεί να γίνει είτε με χρήση κώδικα, είτε μέσω του γραφικού περιβάλλοντος. Το βασικότερο πρόβλημα που λύνεται μέσω των prefabs είναι ότι οποιαδήποτε αλλαγή χρειαστεί θα γίνει μόνο μία φορά σε αντίθεση με την περίπτωση των πολλαπλών αντιγράφων των GameObjects.



## Κεφάλαιο 4

# Vuforia SDK

### 4.1 Εισαγωγή

Η Vuforia είναι ένα AR SDK (software development kit) για κινητά και άλλες παρόμοιες συσκευές, που μας επιτρέπει να φτιάχνουμε AR εφαρμογές σε βίντεο που καταγράφεται και αναπαράγεται στη συσκευή σε πραγματικό χρόνο. Το SDK αυτό χρησιμοποιεί τις δυνατότητες της τεχνολογίας της όρασης υπολογιστών (computer vision) για να αναγνωρίζει και ανίχνευση (tracking) αντικειμένων και εικόνων που καταγράφονται από την κάμερα.

Η Vuforia δίνει τη δυνατότητα στον προγραμματιστή να τοποθετεί εικονικά αντικείμενα, κυρίως 3D, στη σκηνή και να επαναπροσδιορίζει τη θέση και τον προσανατολισμό τους ανάλογα με την εικόνα που καταγράφεται. Μετά, την εμφάνιση τους τα εικονικά αντικείμενα ανταποκρίνονται πλήρως στις κινήσεις της συσκευής ώστε η θέση τους να ταιριάζει με τη θέση και τον προσανατολισμό σε σχέση με τον πραγματικό κόσμο, άρα και του χρήστη. Έτσι τα αντικείμενα φαίνεται σα να είναι μέρη του πραγματικού κόσμου.

Για να εμφανιστούν τα εικονικά αντικείμενα στην οθόνη είναι απαραίτητο να να σκαναριστούν συγκεκριμένα αντικείμενα, τα οποία τα ορίζουμε ως στόχους. Υπάρχουν διάφοροι τύποι στόχων, 2D και 3D, αναγνώριση πολλών στόχων ταυτόχρονα, κυλινδρικές εικόνες. Η Vuforia προσφέρει τη δυνατότητα να δημιουργήσει ο χρήστης δικούς του προσαρμοσμένους targets οι οποίοι μοιάζουν με QR και εμπεριέχουν επιπλέον πληροφορία. Άλλη, μία ενδιαφέρουσα δυνατότητα είναι να δημιουργήσει ο χρήστης (στόχους) targets από 3D αντικείμενα χρησιμοποιώντας μία εφαρμογή που ονομάζεται Model Target Generator και σκανάρει αντικείμενα. Στις τελευταίες, εκδόσεις υπάρχει η δυνατότητα για markerless AR, δηλαδή, χωρίς να εντοπιστεί κάποιος συγκεκριμένος target. Έτσι, μπορούν να εμφανιστούν αντικείμενα είτε στο πάτωμα είτε στον αέρα και να προσφέρουν πλουσιότερες εμπειρίες στους χρήστες.

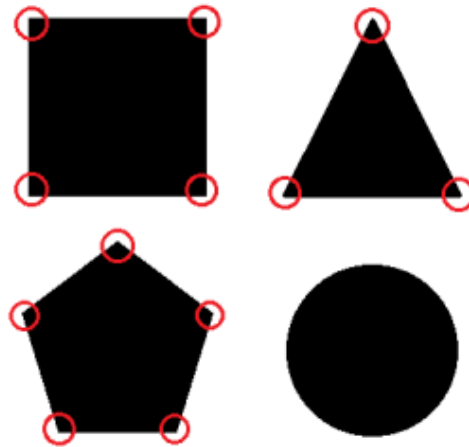
Η Vuforia προσφέρει επιπλέον μία διεπαφή προγραμματισμού ή, όπως, είναι ο αγγλικός όρος API, (Application Programming Interface) σε Java, C++ και C# μέσω Unity. Έτσι, ο προγραμματιστής μπορεί να δημιουργήσει πιο προσαρμοσμένες συμπεριφορές από τα εικονικά αντικείμενα ανάλογα με τις ανάγκες της εφαρμογής. Επιπλέον, από τις γλώσσες που είναι διαθέσιμο το API γίνεται σαφές ότι μπορεί να αναπτυχθούν εφαρμογές και σε IOS και Android.

### 4.2 Χαρακτηριστικά της Vuforia

#### 4.2.1 Tracking

Μη όντας ανοιχτού κώδικα η Vuforia δεν ξέρουμε τι αλγόριθμοι χρησιμοποιούνται για το tracking, αλλά, κοιτάζοντας το εγχειρίδιο [Inca] της καταλαβαίνουμε ότι χρησιμοποιείται κάποια παραλλαγή tracking που εκμεταλλεύεται τα φυσικά χαρακτηριστικά του αντικειμέ-

νου (natural feature tracking). Τέτοια φυσικά χαρακτηριστικά αναφέρονται κυρίως σε γωνίες υψηλής αντίθεσης στις κορυφές τριγώνων.



**Σχήμα 4.1:** Τέσσερα σχήματα που δηλώνουν τι είναι φυσικό χαρακτηριστικό κατά τη Vuforia. Αξίζει να σημειωθεί ότι στο τέταρτο σχήμα δεν υπάρχει χαρακτηριστικά αφού δεν υπάρχουν γωνίες.

Αυτή η τεχνική πρωτοεμφανίστηκε γύρω στο 1980 και βασίζεται στην ιδέα ότι κάθε εικόνα χωρίζεται διάφορα κομμάτια. Κάθε κομμάτι αλλάζει μερικές κατευθύνσεις και μετριέται η RGB τιμή κάθε κομματιού. Για παράδειγμα, αν ένα κομμάτι περιέχει μία ευθεία γραμμή, στην κατεύθυνση της γραμμής θα παρατηρηθούν ελάχιστες διαφορές στην RGB τιμή. Αντίθετα, στην κάθετη διεύθυνση θα παρατηρηθούν μεγάλες αλλαγές. Μία γωνία, θα προκαλέσει μεγάλες αλλαγές στις μετρήσεις σε όλες τις διευθύνσεις. Λαμβάνοντας υπόψιν αυτές τις έντονες αλλαγές ο αλγόριθμος αποφασίζει τι είναι χαρακτηριστικό και το τοποθετεί στη θέση του στη συνολική εικόνα.

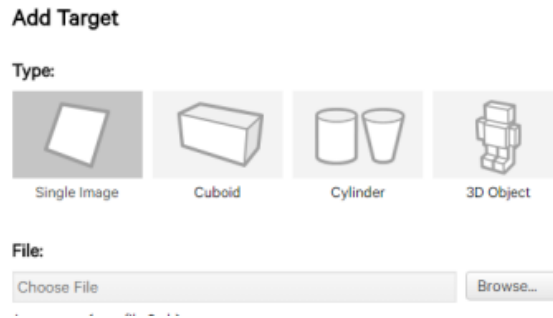
Ως αποτέλεσμα, ο αλγόριθμος φτιάχνει πατρών από χαρακτηριστικά. Στη Vuforia, το πατρών αυτό το ονομάζουμε Target. Όταν ένα target εντοπιστεί από την εφαρμογή, αντιστοιχίζεται σε κάποια ορισμένα πατρών και έτσι υπολογίζεται η πόζα (pose) του target. Ο όρος pose είναι αρκετά διαδεδομένος στο χώρο του computer vision.

## 4.2.2 Targets

Σε όλη τη διάρκεια αυτής της εργασίας θα αναφερόμαστε συχνά σε διάφορους targets, για αυτό είναι σημαντικό να ορίσουμε τι ακριβώς είναι. Ο target είναι ένα πατρών από φυσικά χαρακτηριστικά που έχει προκαθοριστεί από τον developer ή από κάποιον χρήστη. Με άλλα λόγια, όταν η vuforia ψάχνει για μία συγκεκριμένη εικόνα ή αντικείμενο, είναι απαραίτητο να γνωρίζει τι πατρών χρειάζεται να ψάξει. Για αυτό το σκοπό υπάρχει το Vuforia Target Manager. Προσφέρεται ένα web service για να δημιουργεί targets από 2D εικόνες, κυβοειδή αντικείμενα, κυλίνδρους και 3D αντικείμενα.

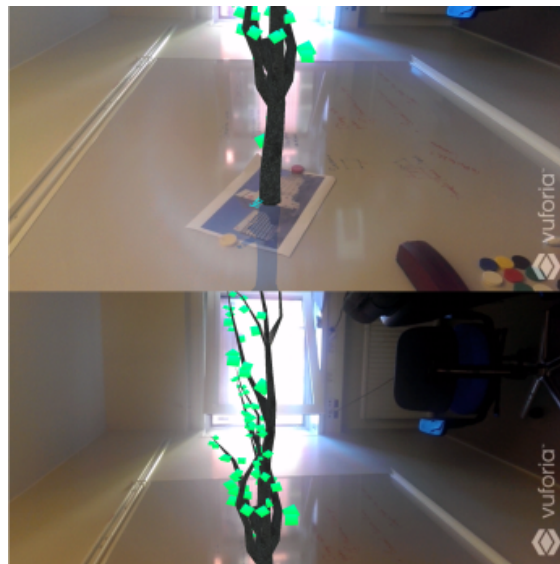
## 4.2.3 Extended Tracking

Το Extended Tracking είναι μία δυνατότητα της Vuforia που επιτρέπει να κάνει track τους targets χρησιμοποιώντας να φυσικά χαρακτηριστικά του χώρου που βρίσκεται. Με αυτά τα χαρακτηριστικά φτιάχνετε ένας χάρτης από χαρακτηριστικά που περιτριγυρίζουν τον target



**Σχήμα 4.2:** Το μενού που επιτρέπει να προσθέσεις Targets

και διατηρούν τον εντοπισμό του ακόμα και όταν δεν είναι ορατός από την κάμερα, όπως φαίνεται και στην εικόνα.



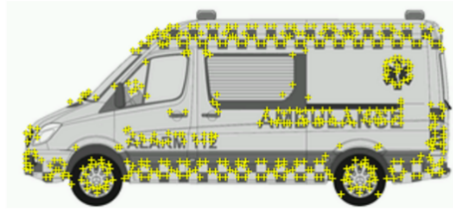
**Σχήμα 4.3:** Με το extended Tracking ενεργοποιημένο, η Vuforia μπορεί να κάνει track τους targets ακόμα και όταν είναι εκτός του πλάνους της κάμερας.

#### 4.2.4 Αναγνώριση Εικόνας

Χρησιμοποιώντας τον Target Manager κάποιος μπορεί να ορίσει έναν target διαλέγοντας μία οποιαδήποτε εικόνα. Το αν η εικόνα αποτελεί έναν καλό target είναι διαφορετικό ερώτημα. Η σύνθεση και το πλήθος των χαρακτηριστικών είναι καθοριστικοί παράγοντες για το πόσο εύκολα η Vuforia μπορεί να εντοπίζει έναν target.

Πιο συγκεκριμένα, ένας καλός target αποτελείται από υψηλό πλήθος από χαρακτηριστικά διατεταγμένα όσο πιο μοναδικά γίνεται. Μία εικόνα ενός δισδιάστατου πλέγματος μπορεί να περιέχει πολλά φυσικά χαρακτηριστικά, αλλά, καθώς το πατρών είναι επαναλαμβανόμενο, κάθε κομμάτι της εικόνας δεν θα είναι αναγνωρίσιμο από την Vuforia. Έτσι, δε θα μπορεί να αναγνωριστεί τι είναι πάνω, κάτω, δεξιά και αριστερά, αποτελώντας έναν κακό target [Inca].

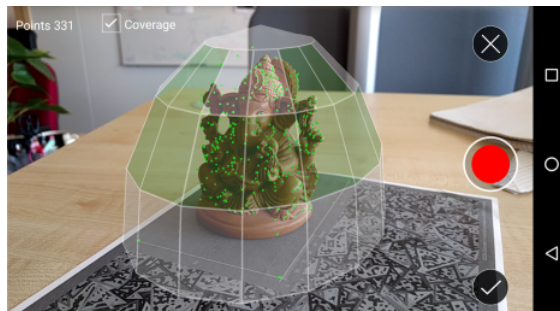
Εικόνες αποτελούμενες από στρογγυλά σχήματα χωρίς γωνίες αποτελούν και αυτές κακούς targets, καθώς δε περιέχουν χαρακτηριστικά προς αναγνώριση.



**Σχήμα 4.4:** Η εικόνα ενός ασθενοφόρου με υψηλό δείκτη εντοπισμού. Τα κίτρινα σημεία είναι τα φυσικά χαρακτηριστικά.

#### 4.2.5 Αναγνώριση Αντικειμένων

Ενώ στον πυρήνα της η αναγνώριση 3D αντικειμένων χρησιμοποιεί και αυτή αναγνώριση φυσικών χαρακτηριστικών, η διαδικασία δημιουργίας των Object Targets είναι διαφορετική από αυτή των εικόνων. Σε ένα Image Target αντιστοιχίζοντας 2D χαρακτηριστικά αναλύοντας τα pixel της εικόνας. Ένα 3D αντικείμενο έχει όμως και έναν τρίτο άξονα να λάβουμε υπόψιν. Τα Object Targets δίνουν τη δυνατότητα στο χρήστη να βλέπει τον target από διάφορες οπτικές γωνίες διατηρώντας την επαύξηση ενεργή. Για να επιτευχθεί κάτι τέτοιο η Vuforia έχει αναπτύξει μία εφαρμογή για σκανάρισμα μικρών σχετικά 3D αντικειμένων. Κατά το σκανάρισμα, το αντικείμενο τοποθετείται πάνω σε μία επιφάνεια στόχο scanning plane ώστε γίνεται αντιληπτός ο προσανατολισμός του αντικειμένου.



**Σχήμα 4.5:** Τοποθετώντας το αντικείμενο που πρόκειται να σκαναριστεί πάνω στον scanner target, η Vuforia μπορεί να προσδιορίσει τους XYZ άξονες και να βρει τα σημεία ενδιαφέροντος. Το πλήθος τους φαίνεται στην πάνω αριστερή γωνία.

Ενώ δεν έχουμε στοιχεία για τους αλγόριθμους τους σκαναρίσματος, κοιτώντας την εφαρμογή καταλαβαίνουμε ότι επικεντρώνεται στο tracking χαρακτηριστικών, χρησιμοποιώντας Interest Point Detection. Αυτή η τεχνική λειτουργεί έχοντας ένα σετ από πρότυπες εικόνες που έχουν την εμφάνιση του αντικειμένου και τα πατρών των χαρακτηριστικών. Καθώς τα χαρακτηριστικά εντοπίζονται, συγκρίνονται με αυτά των εικόνων - τα σημεία ενδιαφέροντος- και υπολογίζεται ο προσανατολισμός [Incb].

#### 4.2.6 Αναγνώριση Κειμένου

Η αναγνώριση κειμένου της Vuforia μπορεί να εντοπίσει 100.000 αγγλικές λέξεις στις πιο διαδεδομένες γραμματοσειρές. Αυτός ο αριθμός μπορεί να αυξηθεί, κατά 10.000, με χρήση επιπλέον λιστών λέξεων. Η αναγνώριση λέξεων βασίζεται στην UTF-8 κωδικοποίηση και μπορεί να αναγνωρίσει μόνο τους χαρακτήρες Aa-Zz [Incc].

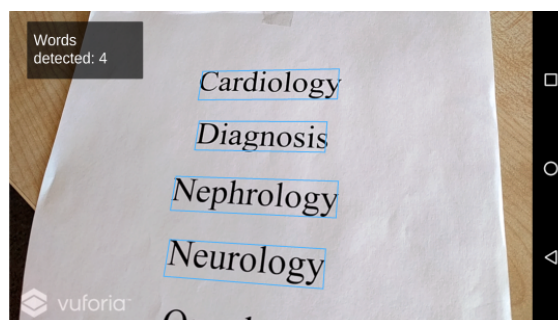


**Σχήμα 4.6:** Μία απλή επαύξηση με τον εντοπισμό του προηγούμενου αντικειμένου χωρίς τη βοήθεια κάποιου target.

Στον πυρήνα αυτής της δυνατότητας βρίσκεται η μηχανή αναγνώρισης κειμένου (text recognition engine). Ωστόσο, δεν υπάρχουν, όπως, και στις άλλες δυνατότητες, συγκεκριμένες προδιαγραφές. Αυτό που είναι γνωστό είναι ότι μία τέτοια δυνατότητα στα κινητά γίνεται υλοποιήσιμη χάρη στο Optical Character recognition, γι' αυτό χρήζει σκόπιμο να ακολουθηθεί μία περιγραφή.

Το Optical Character Recognition, OCR για συντομία, ονομάζεται η διαδικασία κατά την οποία αναγνωρίζονται γράμματα σε εικόνες [Ch15]. Υπάρχουν πολλοί αλγόριθμοι για αυτό, αλλά, στο λειτουργικό Android όλοι εμπίπτουν σε συγκεκριμένα βήματα.

Αρχικά, μία εικόνα δίνεται ως είσοδος και υφίσταται μία προεργασία κατά την οποία το κείμενο διαφοροποιείται από άλλα γραφικά στοιχεία που μπορεί να υπάρχουν. Εφόσον, η εικόνα προέρχεται από την κάμερα θα υπάρχει και θόρυβος. Η προεργασία βοηθάει να μειωθεί ο θόρυβος αυτός και να βελτιωθεί η αναγνώριση του κειμένου. Στη συνέχεια, τα στοιχεία που σχετίζονται με κείμενο αφαιρούνται στο Android, συνήθως, δίνονται ως είσοδος στο ανοιχτού κώδικα sdk της Google, το Tesseract. Εκεί συγκρίνονται με γνωστούς χαρακτήρες.



**Σχήμα 4.7:** Η Vuforia μπορεί να αναγνωρίσει και να ακολουθεί κείμενο σαν να ήταν ένας οποιοσδήποτε target.

### 4.3 Vuforia στη Unity

Η Vuforia προσφέρεται ως Unity package και περιέχει κάποια GameObjects και scripts που κάνουν ευκολότερη της ανάπτυξη AR εφαρμογών. Σε αυτήν την ενότητα θα μελετήσουμε τα σημαντικότερα από αυτά.

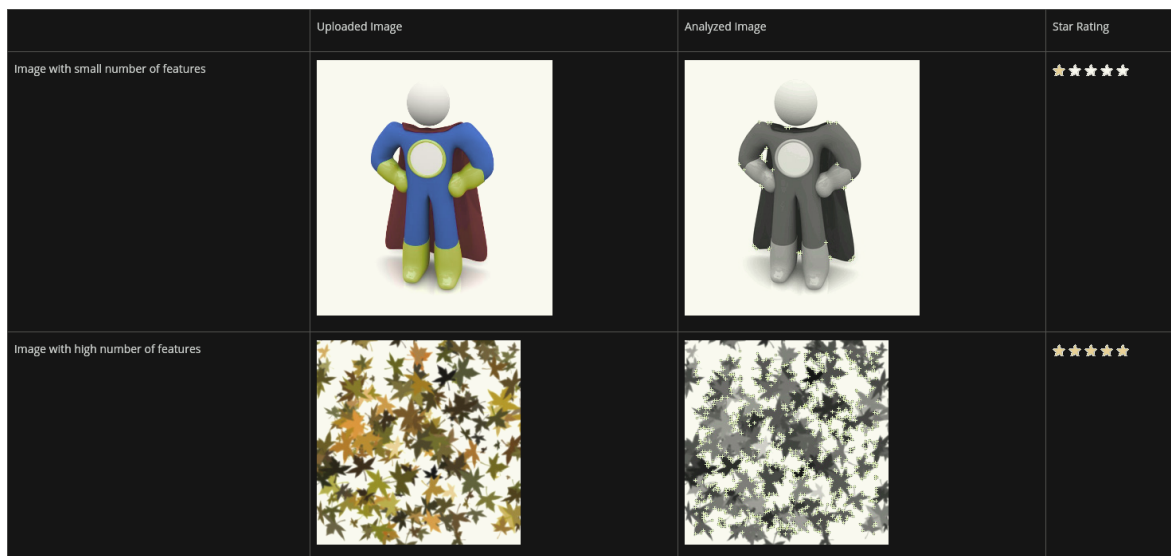
### 4.3.1 ARCamera

Το GameObject ARCamera είναι ίσως το σημαντικότερο από όλα αφού είναι αυτό που μετατρέπει μία απλή εφαρμογή σε AR. Αρχικά πρέπει να αντικαταστήσει την απλή camera που προσφέρεται από τη Unity, έπειτα δίνει τη δυνατότητα καταγραφής του περιβάλλοντος και το πέρασμα του στο Vuforia Engine όπου γίνεται επεξεργασία. Στον inspector, μπορούμε να δούμε ότι προσφέρονται διάφορες ρυθμίσεις όπως το App License Key, το οποίο είναι το αναγνωριστικό για την εφαρμογή, το Camera Device Mode, που διαλέγουμε την ποιότητα καταγραφής, το World Center Mode, που καθορίζει τη σχετική θέση που έχει η συσκευή στο χώρο, κα.

### 4.3.2 Image

Το Image είναι το GameObject που χρησιμοποιούμε περισσότερο. Όπως, προδίδει και το όνομα του, στις ρυθμίσεις αυτουνού αντιστοιχίζουμε εικόνες (Image Targets) τις οποίες η Vuforia μπορεί να εντοπίσει και στη συνέχεια να ακολουθεί(track). Οι εικόνες δε χρειάζεται να είναι κάποιου ειδικού τύπου, όπως είναι τα QR, για να αναγνωριστούν. Η Vuforia χρησιμοποιεί μία σειρά από αλγορίθμους για να βρεί χαρακτηριστικά στοιχεία που βρίσκονται στις εικόνες και στη συνέχεια ψάχνει για αυτά σε μία βάση δεδομένων με γνωστά σε αυτή στοιχεία. Μετά τον εντοπισμό μίας εικόνας η Vuforia την ακολουθεί σε όλο το οπτικό πεδίο της κάμερας.

Τα Image Targets δημιουργούνται online στο Target Management System (TMS) από εικόνες τύπου JPG ή PNG. Κατά τη διαδικασία αυτή εντοπίζονται τα ιδιαίτερα χαρακτηριστικά της εικόνας και αξιολογείται με βάση αυτά. Η αξιολόγηση αυτή μας δείχνει πόσο εύκολο είναι να εντοπιστεί η εικόνα σε πραγματικό χρόνο (runtime).



**Σχήμα 4.8:** Δύο εικόνες με διαφορετικό πλήθος χαρακτηριστικών και πως αυτές αξιολογούνται.

Η Vuforia μπορεί να εντοπίζει παραπάνω από μία εικόνες ταυτόχρονα στο runtime ανάλογα με τον εκάστοτε επεξεργαστή και την κάρτα γραφικών. Το όριο που δίνεται από τη Vuforia αυτή τη στιγμή είναι ότι στην εφαρμογή μπορούν να δοθούν μέχρι 1000 Image Targets προς εντοπισμό. Ωστόσο, αυτό τον περιορισμό μπορούμε να το ξεπεράσουμε χωρίζοντας τους Targets σε Datasets και να διαχειριζόμαστε δυναμικά αυτά στο runtime.



## **Datasets**

Το Dataset είναι ένα σύνολο από Targets που το δημιουργήσαμε και κατεβάσαμε από το TMS. Το SDK της Vuforia μας επιτρέπει να φορτώνουμε, να ενεργοποιούμε να απενεργοποιούμε και να φορτώνουμε από το cloud datasets στο runtime. Τα datasets μπορούν να περιέχουν Image Targets, Multi Targets, κυλίνδρους και 3D. Το φόρτωμα τους χρειάζεται κάποιο χρόνο οπότε είναι συνιστάται να γίνεται στο background.

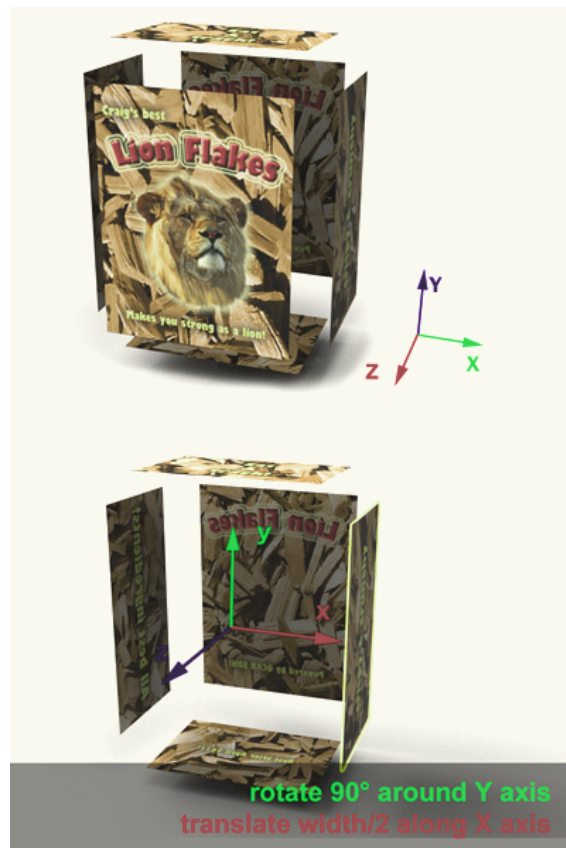
## **Target Size**

Το Target size είναι η παράμετρος που δείχνει το 3D μέγεθος του Target στη σκηνή, στη μονάδα της Unity τα units. Αυτή η παράμετρος προσδιορίζεται στο TMS κατά τη δημιουργία του Target. Είναι πολύ σημαντικό να δώσουμε σωστές αναλογίες αφού όλα τα εικονικά μοντέλα και υπολογισμοί θα λάβουν υπόψιν αυτό το μέγεθος.

### 4.3.3 Multi Targets

Ένας Multi Target αποτελείται από πολλά Image Targets οι οποίες έχουν μία χωρική συσχέτιση. Όταν η κάμερα εντοπίζει ένα κομμάτι του Multi Target, όλα τα υπόλοιπα είναι σαν να εντοπίστηκαν στην προκαθορισμένη θέση και προσανατολισμό. Έτσι, δεν χρειάζεται να εντοπιστούν όλα τα μέρη του. Η διαφορά πολλών Image Targets και ενός Multi είναι ότι το δεύτερο συμπεριφέρεται σαν ένα μόνο GameObject με ένα Transform.

Υπάρχουν δύο τρόποι να δημιουργηθεί ένα Multi Target. Ο πρώτος είναι να γίνει online από το TMS χρησιμοποιώντας εικόνες και ο δεύτερος στο runtime μέσω το api. Στο runtime μπορούμε, ακόμα, δυναμικά να προσθέτουμε, αφαιρούμε μέρη και να αλλάζουμε τη χωρική διάταξη.

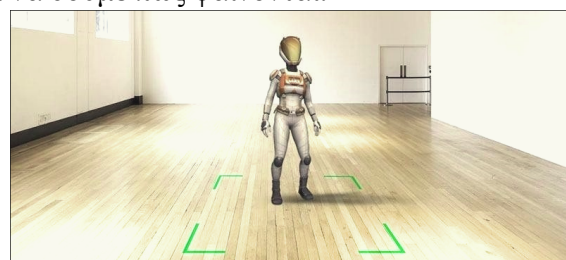


Σχήμα 4.9: A Multi Target

### 4.3.4 Ground Plane - Mid Air

Το Ground Plane επιτρέπει στο χρήστη να τοποθετεί εικονικά μοντέλα πάνω σε οριζόντιες επιφάνειες στο περιβάλλον, όπως το δάπεδο και τα τραπεζία. Εκτός από τον εντοπισμό τους επιτρέπει και παρακολούθηση τους, ώστε να μπορούμε να δημιουργούμε μοντέλα στον αέρα με την χρήση anchor points, χωρίς να χρειαζόμαστε κάποιων Target. Έτσι μας δίνεται η δυνατότητα να δημιουργήσουμε χωρικές AR εμπειρίες αφού τα μοντέλα μας μπορούν να μείνουν σταθερά στο χώρο, δίνοντας πολλές πιθανές εφαρμογές. Μία από τις πιο γνωστές εφαρμογές είναι της IKEA που επιτρέπει να τοποθετούμε εικονικά έπιπλα στο χώρο για να δούμε πως φαίνονται.

Το Ground Plane (και το Mid Air) χωρίζεται σε δύο GameObjects. Το πρώτο είναι το Ground Plane Stage (Mid Air Stage) και λειτουργεί ως ο πατέρας και το περιεχόμενο που θέλουμε θα είναι το παιδί. Το Ground Plane Stage έχει μία οπτική σήμανση που υποδηλώνει ότι είναι 100 τετραγωνικά εκατοστά. Αυτή η σήμανση είναι ορατή μόνο κατά τη διάρκεια του development και βοηθάει στην αντιστοίχιση της κλίμακας πραγματικού κόσμου και Unity. Επίσης, η σχετική θέση του παιδιού με το Stage θα είναι και η σχετική θέση που θα έχει το εικονικό μοντέλο με την πραγματική επιφάνεια.



Σχήμα 4.10: Ένα 3D μοντέλο αστροναύτη εμφανίζεται στο δάπεδο.

Το δεύτερο GameObject είναι το Plane Finder (Mid Air Positioner). Το Plane Finder είναι υπεύθυνο για να δέχεται κάποιο input από τον χρήστη, όπως ένα πάτημα στην οθόνη, και στη συνέχεια να ψάχνει για μία κατάλληλη επιφάνεια και να τοποθετεί το μοντέλο. Στον Inspector το Finder δέχεται ως παράμετρο ένα Anchor Stage. Σε αυτό πρέπει να περάσουμε το GameObject του Ground Plane Stage.

## Κεφάλαιο 5

# Android

## 5.1 Εισαγωγή

Το Android είναι λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας και άλλες συσκευές το οποίο τρέχει τον πυρήνα του λειτουργικού Linux. Αρχικά αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με τις γλώσσες προγραμματισμού Java και Kotlin, ελέγχοντας την συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από την Google. Στόχος των δημιουργών ήταν η δημιουργία ενός λειτουργικού, ευέλικτου καθώς και αναβαθμίσιμου.

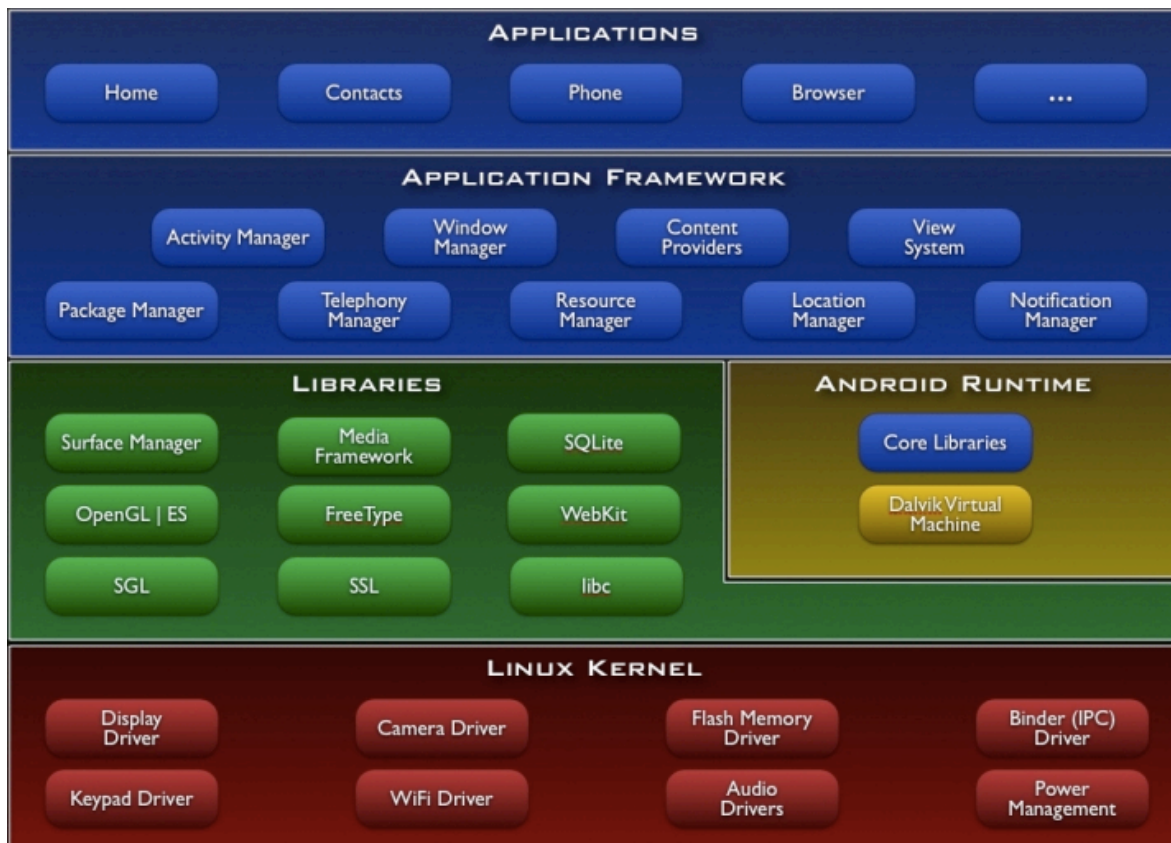
Η πρώτη παρουσίαση της πλατφόρμας Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance, μιας κοινοπραξίας 48 τηλεπικοινωνιακών εταιρειών, εταιρειών λογισμικού καθώς και κατασκευής hardware, οι οποίες είναι αφιερωμένες στην ανάπτυξη και εξέλιξη ανοιχτών προτύπων στις συσκευές κινητής τηλεφωνίας. Τον Σεπτέμβριο του 2008 η T-Mobile ανακοίνωσε την διαθεσιμότητα του T-Mobile G1, του πρώτου Smartphone βασισμένο στο λειτουργικό Android. Λίγες μέρες αργότερα η Google ανακοίνωσε την διαθεσιμότητα του Android SDK Release Candidate 1.0. Το Android ήταν διαθέσιμο σαν opensource λογισμικό από τον Οκτώβριο του 2008. Η Google δημοσίευσε το μεγαλύτερο μέρος του κώδικα του Android υπό τους όρους της Apache License, μιας ελεύθερης άδειας λογισμικού. μέσω του οποίου, ιδιωτικές εταιρείες μπορούσαν να προσθέσουν τις δικές τους εφαρμογές και επεκτάσεις και να τις πουλήσουν χωρίς να υποχρεούνται να τις υποβάλλουν στην open-source κοινότητα. Στα τέλη του 2008 η Google ανακοίνωσε μια συσκευή με το όνομα Android Dev Phone 1 που είχε την δυνατότητα να τρέχει Android εφαρμογές χωρίς να είναι άμεσα συνδεδεμένη με κάποιο δίκτυο κινητής τηλεφωνίας. Ο στόχος της συσκευής αυτής ήταν να επιτρέψει στους προγραμματιστές να πειραματιστούν με μια πραγματική συσκευή με Android χωρίς κάποιο συμβόλαιο με εταιρεία κινητής τηλεφωνίας.

## 5.2 Η αρχιτεκτονική του Android

Σε αυτήν την υποενότητα παρουσιάζεται η αρχιτεκτονική της πλατφόρμας Android, η οποία όπως φαίνεται στην Εικόνα 5.1 αποτελείται από τα layers: Linux Kernel, Libraries, Android Runtime, Application Framework και Applications.

### 5.2.1 Linux Kernel

Το Android έχει χτιστεί πάνω στον Linux Kernel και επομένως χρησιμοποιείται Linux Power Management για την επικοινωνία με το hardware της συσκευής. Συγκεκριμένα χρησιμοποιείται Linux για τους drivers της συσκευής, την διαχείριση μνήμης, την διαχείριση



**Σχήμα 5.1:** Η αρχιτεκτονική Android

διεργασιών και τη δικτύωση. Ωστόσο ο προγραμματιστής εφαρμογών δεν θα χρειαστεί ποτέ να προγραμματίσει σε αυτό το επίπεδο.

## 5.2.2 Libraries

Στο επόμενο επίπεδο βρίσκονται οι βιβλιοθήκες. Το Android περιλαμβάνει ένα σετ από βιβλιοθήκες, που χρησιμοποιούνται από διάφορα components του συστήματος και είναι γραμμένες σε C/C++. Αυτές οι δυνατότητες διαθέτονται στους developers μέσα από το Android Application Framework. Μερικές από τις βασικές βιβλιοθήκες είναι οι παρακάτω:

- **System C library:** είναι μια BSD-παραγόμενη εφαρμογή της standard C system library (libc), συντονισμένη για embedded Linux-based συσκευές.
- **Media Libraries:** βασισμένες στο PacketVideo OpenCore. Οι βιβλιοθήκες υποστηρίζουν playback και recording σε πολλά δημοφιλή audio και video formats, όπως MPEG4, H.264, MP3, AAC, AMR, JPG και PNG.
- **Surface Manager:** συνθετικός window manager παρόμοιος με αυτόν των Windows ή του Comriz, αλλά πολύ πιο απλό. Οι εντολές σχεδιασμού πηγαίνουν εκτός οθόνης bitmap που μετέπειτα συνδυάζονται με άλλα bitmaps για να σχηματίσουν την οθόνη που βλέπει ο χρήστης. Αυτό επιτρέπει στο σύστημα να δημιουργήσει πολλά εφέ όπως διάφανα παράθυρα και εντυπωσιακά transitions.
- **LibWebCore:** ένας μοντέρνος browser που χρησιμοποιεί την WebKit βιβλιοθήκη για την γρήγορη προβολή HTML περιεχομένου. Η ίδια μηχανή χρησιμοποιείται και στο Google Chrome browser, στον Safari της Apple και στα Nokia S60.
- **SGL:** μηχανή για 2D γραφικά

- **3D libraries:** βασίζονται στα OpenGL ES 1.0 APIs. Οι βιβλιοθήκες χρησιμοποιούν 3D υλικό αν η συσκευή το διαθέτει αλλιώς έναν γρήγορο software rasterizer.
- **SQLite:** μια δυναμική και ελαφριά βάση δεδομένων, που χρησιμοποιείται από εφαρμογές για μόνιμη αποθήκευση δεδομένων, παρόμοια με αυτή που χρησιμοποιείται στον Firefox και στο iPhone.

### 5.2.3 Applications

Το τελευταίο στρώμα στην αρχιτεκτονική του Android είναι τα applications και είναι αυτά που αντιλαμβάνεται ο χρήστης. Πρόκειται για εφαρμογές όπως email client, πρόγραμμα για SMS, ημερολόγιο, χάρτες, browser και άλλα. Όλες οι εφαρμογές είναι προγραμματισμένες σε Java και Kotlin. Ωστόσο, μια Java εφαρμογή για Android δεν είναι συμβατή με προγράμματα Java γραμμένα για Java SE και Java ME πλατφόρμες.

## 5.3 Ανατομία μίας Android εφαρμογής

Υπάρχουν τέσσερα διαφορετικά είδη συστατικών μερών μιας Android εφαρμογής. Κάθε είδος εξυπηρετεί συγκεκριμένο σκοπό και έχει συγκεκριμένο κύκλο ζωής που ορίζει πότε το συστατικό θα δημιουργηθεί και θα καταστραφεί. Τα τέσσερα αυτά building blocks είναι τα: Activities, Services, Content Providers, Broadcast receivers και αναλύονται στην συνέχεια.

### 5.3.1 Συστατικά εφαρμογών

#### Activities

Ένα activity αντιπροσωπεύει μια οθόνη με ένα user interface. Για παράδειγμα, μια εφαρμογή για email μπορεί να έχει μια activity που δείχνει μια λίστα από νέα emails, μια άλλη activity για να δημιουργείς email και μια άλλη για να διαβάζεις emails. Παρόλο που οι activities συνεργάζονται για να δημιουργήσουν μια συνεκτική εμπειρία για τον χρήστη, κάθε μια είναι ανεξάρτητη από τις υπόλοιπες. Έτσι, μια διαφορετική εφαρμογή μπορεί να θέσει σε λειτουργία οποιαδήποτε από αυτές τις activities (αν η εφαρμογή για email το επιτρέπει). Για παράδειγμα, μια κάμερα μπορεί να θέσει σε λειτουργία την activity που μπορείς να συνθέσεις email, έτσι ώστε ο χρήστης να μπορέσει να την διαμοιραστεί.

#### Services

Ένα service είναι ένα συστατικό που τρέχει στο background για να διεκπεραιώσει μεγάλης χρονικής διάρκειας διαδικασίες, ή για να εκτελέσει εργασίες απομακρυσμένων διεργασιών. Ένα service δεν παρέχει user interface. Για παράδειγμα, ένα service μπορεί να παίζει μουσική στο background ενώ ο χρήστης έχει ανοιχτή μια άλλη εφαρμογή. Ένα άλλο συστατικό, όπως ένα activity, μπορεί να θέσει σε λειτουργία ένα service και να το αφήσει να τρέχει ή να το δεσμεύσει ώστε να επικοινωνεί μαζί του.

#### Content Providers

Ένας content provider διαχειρίζεται ένα σύνολο από διαμοιραζόμενα δεδομένα της εφαρμογής. Έτσι τα δεδομένα μπορούν να αποθηκευτούν στο σύστημα αρχείων, στην SQLite

βάση δεδομένων, στο web, ή σε οποιοδήποτε χώρο αποθήκευσης έχει πρόσβαση η εφαρμογή. Μέσω του content provider, άλλες εφαρμογές μπορούν να ζητήσουν ή να τροποποιήσουν τα δεδομένα αν αυτός το επιτρέπει. Για παράδειγμα το λειτουργικό σύστημα Android προσφέρει έναν content provider που διαχειρίζεται της πληροφορίες των επαφών του χρήστη. Έτσι οποιαδήποτε εφαρμογή με την κατάλληλη άδεια μπορεί να ζητήσει από ένα μέρος του content provider να διαβάσει ή να γράψει πληροφορία για μια συγκεκριμένη επαφή.

Οι content providers είναι επίσης χρήσιμοι για την ανάγνωση και εγγραφή δεδομένων που είναι ιδιωτικά για την εφαρμογή μας και όχι διαμοιραζόμενα. Για παράδειγμα μια εφαρμογή Note Pad χρησιμοποιεί τον content provider για να αποθηκεύσει σημειώσεις.

## Broadcast Receivers

Ένας broadcast receiver είναι ένα συστατικό που απαντά σε system-wide broadcast ανακοινώσεις. Πολλά broadcasts προέρχονται από το σύστημα, για παράδειγμα μια broadcast ανακοίνωση ότι η οθόνη έκλεισε, η μπαταρία είναι χαμηλή, ή ότι τραβήχτηκε μια φωτογραφία. Τα broadcasts μπορούν ακόμα να αρχικοποιηθούν από εφαρμογές, για παράδειγμα το να επιτρέπεται άλλες εφαρμογές να ξέρουν ότι κάποια δεδομένα έχουν φορτωθεί στην συσκευή και είναι διαθέσιμα για χρήση. Παρόλο που οι broadcast receivers δεν έχουν user interface, μπορούν να δημιουργήσουν μια ειδοποίηση στο status bar, για να ενημερώσουν το χρήστη όταν ένα broadcast συμβεί.

### 5.3.2 Ενεργοποίηση Συστατικών

Τρία από τα τέσσερα building blocks - activities, services, και broadcast receivers - ενεργοποιούνται από ένα ασύγχρονο μήνυμα που ονομάζεται intent. Τα intents δεσμεύουν μεμονωμένα συστατικά μεταξύ τους κατά την διάρκεια της εκτέλεσης, είτε το συστατικό ανήκει στην εφαρμογή μας είτε σε άλλη.

Για τα activities και τα services, ένα intent ορίζει την μια ενέργεια που πρέπει να εκτελεστεί (για παράδειγμα να δεις  ή να στείλεις  κάτι) και μπορεί να διευκρινίζει το URI των δεδομένων πάνω στο οποίο θα γίνει η ενέργεια. Για παράδειγμα, ένα intent μπορεί να μεταβιβάσει μια αίτηση σε ένα activity για να δείξει μια εικόνα ή να ανοίξει μια ιστοσελίδα. Σε μερικές περιπτώσεις υπάρχει η δυνατότητα να τεθεί σε λειτουργία ένα activity για να λάβουμε ένα αποτέλεσμα το οποίο στη συγκεκριμένη περίπτωση μπορεί να επιστραφεί σε ένα intent από το activity.

Για τους broadcast receivers, το intent ορίζει την ανακοίνωση που πρόκειται να γίνει broadcast.

### 5.3.3 Το Αρχείο Manifest

Πριν το Android θέσει σε λειτουργία ένα συστατικό μιας εφαρμογής, το σύστημα πρέπει να γνωρίζει ότι το συστατικό υπάρχει διαβάζοντας το αρχείο AndroidManifest.xml της εφαρμογής. Η εφαρμογή πρέπει να ορίζει όλα τα συστατικά της σε αυτό το αρχείο, το οποίο πρέπει να βρίσκεται στο root του project directory της εφαρμογής.

Το αρχείο manifest προκειμένου να ορίσει τα συστατικά των εφαρμογών κάνει τα εξής:

- Προσδιορίζει κάθε άδεια του χρήστη που απαιτεί η εφαρμογή, όπως πρόσβαση στο Internet ή αναγνωσιμότητα των επαφών του χρήστη.
- Ορίζει το ελάχιστο API Level που απαιτείται από την εφαρμογή, και βασίζεται σε ποια APIs αυτή χρησιμοποιεί.

- Ορίζει τα χαρακτηριστικά του hardware και του software που χρησιμοποιούνται ή απαιτούνται από την εφαρμογή, όπως κάμερα, bluetooth services, ή multitouch οθόνη.
- Δηλώνει τις API βιβλιοθήκες με τις οποίες η εφαρμογή χρειάζεται να συνδεθεί, για παράδειγμα η Google Maps βιβλιοθήκη.

### 5.3.4 Resources

Μια Android εφαρμογή δεν είναι μόνο κώδικας αλλά και resources που είναι εικόνες, audio αρχεία και οτιδήποτε σχετικό με την εμφάνιση της εφαρμογής. Για παράδειγμα, ο προγραμματιστής πρέπει να ορίσει animations, menus, styles, χρώματα και την εμφάνιση της activity με XML αρχεία. Χρησιμοποιώντας application resources είναι εύκολο να αναβαθμιστούν πολλά χαρακτηριστικά μιας εφαρμογής χωρίς να τροποποιηθεί ο κώδικας.

Όλα τα resources γίνονται compile μέσα στο application και αποθηκεύονται στον φάκελο res μέσα στο project. Υπάρχουν ξεχωριστοί φάκελοι για τις εικόνες, τα strings, τα xml και άλλα. Ο resource compiler συμπιέζει και πακετάρει τα resources και μετά δημιουργεί μια κλάση R που περιέχει τους identifiers που χρησιμοποιούνται στο πρόγραμμα για να ανακληθούν τα resources στο πρόγραμμα. Έτσι το Android αναγνωρίζει ότι όλες οι αναφορές σε resources είναι έγκυρες και σώζει χώρο καθώς δεν αποθηκεύει όλα τα resource keys.

## 5.4 Κύκλος ζωής Activity

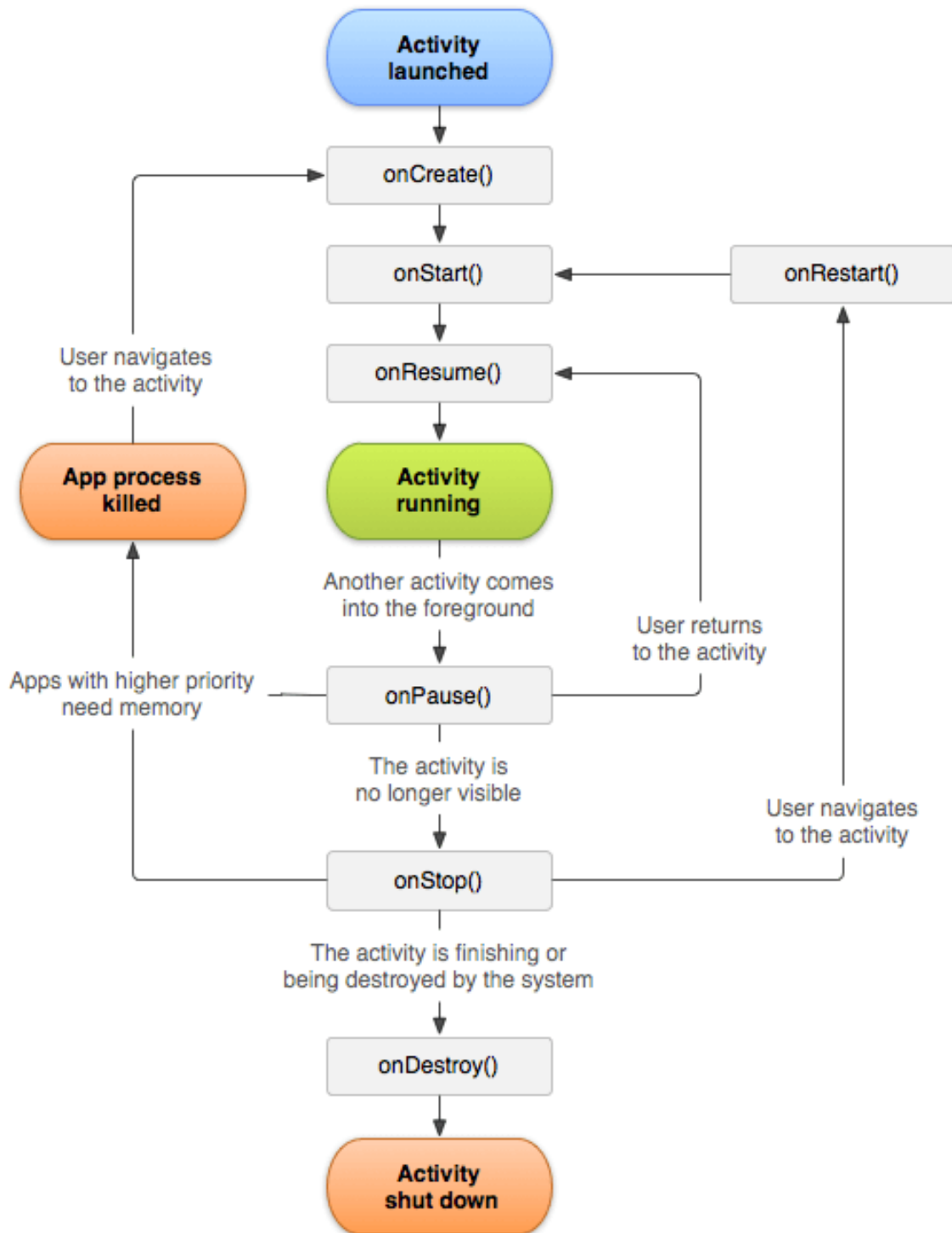
Τα activities στο σύστημα, διαχειρίζονται σαν μια στοίβα από activities. Όταν ένα activity ξεκινήσει, τότε τοποθετείται στην κορυφή της στοίβας και γίνεται το τρέχον activity και το προηγούμενο activity παραμένει από κάτω στην στοίβα, η οποία έρχεται μόνο στο προσκήνιο αν το καινούργιο activity τελειώσει.

Ένα activity μπορεί ουσιαστικά να βρεθεί σε τέσσερις καταστάσεις:

- Αν ένα activity βρίσκεται στο προσκήνιο, δηλαδή εμφανίζεται στην οθόνη (είναι στην κορυφή της στοίβας), τότε είναι στην κατάσταση active ή running.
- Αν ένα activity δεν είναι focused αλλά είναι ακόμα ορατό (δηλαδή μια νέο activity που δεν είναι full-sized είναι focused πάνω από το πρώτο activity), τότε βρίσκεται στην κατάσταση paused. Σε αυτήν την κατάσταση το activity είναι ζωντανό (διατηρεί όλη την κατάσταση), αλλά μπορεί να κλείσει από το σύστημα σε περίπτωση πολύ χαμηλής μνήμης.
- Αν ένα activity είναι πλήρως κρυμμένο από άλλο activity, τότε βρίσκεται στην κατάσταση stopped. Διατηρεί όλη την κατάσταση και τις πληροφορίες, ωστόσο δεν είναι πλέον ορατό στον χρήστη, έτσι το παράθυρό του είναι κρυμμένο και συνήθως κλείνει από το σύστημα όταν υπάρχει ανάγκη για μνήμη.
- Αν ένα είναι στην κατάσταση paused ή stopped, το σύστημα μπορεί να την βγάλει από την μνήμη, είτε ρωτώντας την να το σταματήσει, είτε απλώς κλείνοντας την διεργασία του. Όταν εμφανίζεται ξανά στο χρήστη, πρέπει να ξεκινήσει πάλι και να αποκαταστήσει την προηγούμενη του κατάσταση.

Το διάγραμμα που ακολουθεί δείχνει τα σημαντικά μονοπάτια κατάστασης ενός Activity. Τα τετραγωνισμένα παραλληλόγραμμα δείχνουν τις callback μεθόδους που μπορούν να εφαρμοστούν για τις διάφορες διεργασίες όταν ένα Activity κινείται μεταξύ διαφόρων καταστάσεων. Τα χρωματιστά στρογγυλεμένα παραλληλόγραμμα δείχνουν τις κύριες καταστάσεις στις οποίες μπορεί να βρεθεί ένα Activity.

Υπάρχουν τρία loops που μπορείτε να διαχειριστείτε μέσα σε μια activity:



**Σχήμα 5.2:** Κύκλος ζωής ενός Activity

- Το loop entire lifetime μιας activity διενεργείται μεταξύ της πρώτης κλήσης onCreate(Bundle) μέχρι μιας τελικής κλήσης onDestroy(). Ένα activity θα κάνει όλες τις αρχικοποιήσεις στην κλήση onCreate(), και θα απελευθερώσει όλα τα resources στην onDestroy().

- Το loop visible lifetime ενός activity ουσιαστικά είναι μεταξύ μιας κλήσης onStart() και της αντίστοιχης κλήσης onStop(). Κατά την διάρκεια αυτού του loop ο χρήστης μπορεί να δει την activity στην οθόνη, παρόλο που μπορεί να μην είναι στο προσκήνιο και να αλληλεπι-

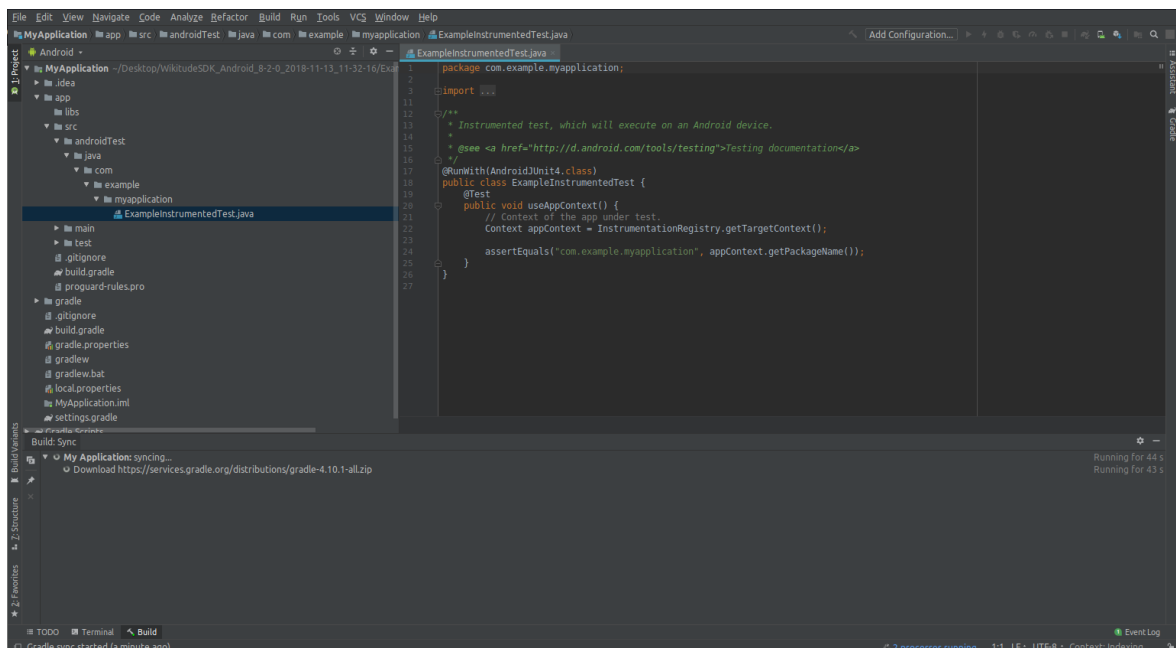


δρά με τον χρήστη. Μεταξύ των δυο αυτών μεθόδων μπορούν να διατηρούνται τα resources που χρειάζονται για να δείχνουν την activity στο χρήστη. Αυτές οι δυο μέθοδοι μπορούν να καλούνται πολλές φορές, καθώς ένα activity εμφανίζεται ή είναι κρυμμένο στο χρήστη.

- Το foreground lifetime loop ενός activity διενεργείται μεταξύ μιας κλήσης onResume() και της αντίστοιχης κλήσης onPause(). Κατά τη διάρκεια αυτή το activity βρίσκεται μπροστά από όλα τα υπόλοιπα activities και αλληλεπιδρά με τον χρήστη. Ένα activity μπορεί συχνά να μεταβεί μεταξύ των δυο αυτών μεθόδων, για παράδειγμα όταν η συσκευή τίθεται σε sleep mode όταν φτάνει ένα αποτέλεσμα ενός activity μέσω ενός intent. Για αυτό το λόγο ο κώδικας σε αυτές τις μεθόδους πρέπει να είναι αρκετά ελαφρύς.

## 5.5 Android Studio

Στο σημείο αυτό έφτασε η ώρα να γίνει και η σύνδεση της γλώσσας Java με το λειτουργικό σύστημα Android αλλά και με το εξειδικευμένο πρόγραμμα για την δημιουργία εφαρμογών Android Studio που είναι το προεπιλεγμένο εργαλείο ανάπτυξης εφαρμογών από την Google. Η Java είναι σημαντικό συστατικό του λειτουργικού Android το οποίο είναι χτισμένο πάνω στον πυρήνα του Linux μεγάλα κομμάτια του οποίου είναι γραμμένα σε C, αλλά το Android Support Development Kit χρησιμοποιεί την Java σαν βάση για τις εφαρμογές Android, χωρίς να χρησιμοποιεί το virtual machine της Java αλλά μεταφράζοντας των κώδικα σε Dalvik bytecode. Όλα αυτά τα συστατικά πλέον συνοψίζονται στο βασικό εργαλείο ανάπτυξης εφαρμογών που λέγεται Android Studio. Για να χρησιμοποιήσει κανείς το Android Studio πρέπει να έχει κάποια έκδοση της Java εγκατεστημένη στον υπολογιστή του και το πρόγραμμα αυτόματα αποκτά πρόσβαση σε όλες τις βιβλιοθήκες της Java κάνοντας αρκετά πιο εύκολη την ανάπτυξη των εφαρμογών. Είναι δηλαδή ένα ολοκληρωμένο περιβάλλον ανάπτυξης από το οποίο διατίθεται υπό την άδεια Apache License 2.0 δηλαδή είναι ένα ελεύθερο προϊόν. Η κατασκευή του περιβάλλοντος ξεκίνησε από την εταιρία IntelliJ IDEA και πλέον έχει γίνει το εργαλείο για την ανάπτυξη εφαρμογών.



Σχήμα 5.3: Το περιβάλλον ανάπτυξης Android Studio

Το περιβάλλον προσφέρει πολλά πλεονεκτήματα και σαφή οργάνωση των αρχείων ώστε

να είναι εύκολη η ανάπτυξη εφαρμογών. Περιλαμβάνει editor για τον κώδικα που είναι εξαιρετικά αποδοτικός σε σημείο που δεν χρειάζεται καν να σκεφτείς την δομή κάθε εντολής καθώς οι οδηγίες έρχονται πακέτο μαζί με την σύνταξη της, debugger ο οποίος χρησιμοποιείται για να μπορούμε εύκολα να διακρίνουμε τα λάθη στον κώδικα, το στοιχείο Android Virtual Device με βάση το οποίο μπορεί κανείς να δημιουργήσει εικονικές συσκευές και να τις τρέξει (στον emulator) για να προσομοιώσει σε αυτές την εφαρμογή του ή και να περάσει απευθείας τον κώδικα του στη δικιά του συσκευή και να δει την εφαρμογή του εκεί. Εκτός από τον editor του κώδικα περιλαμβάνει και ένα γραφικό περιβάλλον δημιουργίας της εκάστοτε οθόνης διεπαφής προς το χρήστη όπου με την μέθοδο drag and drop μπορείς να επιλέγεις διάφορα στοιχεία και να τα επισυνάπτεις στην οθόνη. Επί της ουσίας αυτό που λαμβάνει χώρα είναι το κάλεσμα μιας κλάσης και η δημιουργία συγκεκριμένου στιγμιότυπου αυτής το οποίο γίνεται αυτόματα και εμείς δεν χρειάζεται να γράψουμε κώδικα για αυτό.

## Κεφάλαιο 6

# Πλοήγηση Εσωτερικού Χώρου

## 6.1 Εισαγωγή

Με τη ραγδαία πρόοδο στην τεχνολογία των smartphones, η δυνατότητα της απόκτησης της θέσης της συσκευής και του ατόμου είναι κύριο χαρακτηριστικό για τη λειτουργία πολλών εφαρμογών. Το GPS είναι διαθέσιμο σε κάθε smartphone και χρησιμοποιείται σε κάθε κατηγορίας εφαρμογή, πέρα από τις παραδοσιακές εφαρμογές που χρησιμοποιούν χάρτη. Στις μέρες μας η τοποθεσία χρησιμοποιείται από τον υπολογισμό της τοπικής ώρας, τον καιρό ή οποιαδήποτε άλλη πληροφορία είναι χρήσιμη για τον εκάστοτε χρήστη. Επίσης το gps, με την ακρίβεια που διαθέτει, είναι η καλύτερη λύση για πλοήγηση εξωτερικού χώρου. Ωστόσο, σε εσωτερικούς χώρους το ασύρματο σήμα γρήγορα αποδυναμώνεται λόγω του πάχους των τοίχων, της οροφής και από τα κοντινά αντικείμενα. Η γνώση του προσανατολισμού και της τοποθεσίας της συσκευής θα έδινε τη δυνατότητα να φτάνει στο χρήστη πληροφορία ανάλογα με τις ανάγκες του. Από αυτή την άποψη, οι εξαιρετικές επιδόσεις σε οπτικό υπολογισμό (visual computing) επιτρέπει την ανίχνευση και παρακολούθηση δεικτών και αντικειμένων σε πραγματικό χρόνο. Αυτή η πληροφορία μπορεί να χρησιμοποιηθεί για την αναγνώριση θέσης και περιοχές που το GPS είναι ανακριβές ή μη διαθέσιμο.

Ωστόσο, μέχρι πρόσφατα οι κινητές ηλεκτρονικές συσκευές δεν είχαν τέτοια υπολογιστική ισχύ. Έτσι, στον εντοπισμό (localization) σε εσωτερικούς χώρους κυριαρχούσαν λύσεις που βασίζονταν σε ασύρματη επικοινωνία σε συνδυασμό με αδρανειακούς αισθητήρες κα. Ας δούμε κάποιες από αυτές τις μεθόδους.

### 6.1.1 Inertial navigation system

Το Inertial navigation system(INS) είναι ένα σύστημα πλοήγησης 6 αξόνων αποτελούμενο από ένα επιταχυνσιόμετρο και ένα γυροσκόπιο. Ο ρόλος του είναι να υπολογίζει συνεχώς την θέση μέσω τον ανιχνευτή κίνησης (επιταχυνσιόμετρο) και την κατεύθυνση μέσω του γυροσκοπίου. Η τρέχουσα θέση είναι γνωστή βάση της προηγούμενης.

### 6.1.2 Θέση μέσω του Wi-fi

Σε αυτή την προσέγγιση τρία σημεία πρόσβασης είναι απαραίτητα (access points). Μέσω ασύρματου δικτύου, ο χρήστης συνδέεται και σύμφωνα με την ένταση του σήματος υπολογίζεται η απόσταση του από κάθε σημείο πρόσβασης. Τα αποτελέσματα συγκρίνονται με κάποια προκαθορι-



Σχήμα 6.1: Wifi Fingerprinting

σμένα στη βάση δεδομένων και υπολογίζεται η θέση του χρήστη.

### 6.1.3 Bluetooth

Αυτή η τεχνική είναι ιδιαίτερα αποτελεσματική και όχι επεξεργαστικά απαιτητική για μεγάλους χώρους. Η απόσταση υπολογίζεται από την άφιξη του σήματος και την θέση του μέσω της επικοινωνίας που έχουν τα bluetooth beacon και η συσκευή του χρήστη. Τα bluetooth beacon έχουν έναν ράδιο εκπομπό και μία μπαταρία, καθένα το οποίο επικοινωνεί με τον χρήστη μεταδίδει το μοναδικό του id στο σύστημα, το οποίο στη συνέχεια μεταφράζεται σε συντεταγμένες.



Σχήμα 6.2: Bluetooth Beacons

### 6.1.4 Computer Vision

Σε μία πιο απλοποιημένη έκδοση, targets έχουν τοποθετηθεί σε διάφορα σημεία του χώρου με αποτέλεσμα μόλις σκαναριστούν να γνωρίζει η συσκευή σε ποίο σημείο βρίσκεται ο χρήστης. Βέβαια, υπάρχουν και πιο αναπτυγμένες τεχνικές, όπως, το Visual-Inertial Odometry (VIO) κατά την οποία οι εικόνες της κάμερας επεξεργάζονται από το σύστημα και γίνεται γνωστή η θέση της συσκευής και ο προσανατολισμός της. Μία, ακόμα, γνωστή τεχνική στο χώρο της όρασης υπολογιστή είναι το Simultaneous Localization and mapping (SLAM) η οποία περιγράφει το πρόβλημα της κατασκευής και ενημέρωσης ενός χάρτη σε ένα άγνωστο περιβάλλον, ενώ, παράλληλα κρατιέται η θέση της συσκευής μέσα στον χάρτη αυτόν.

## 6.2 Σχεδιασμός και Υλοποίηση

### 6.2.1 Σχεδιασμός

Από την έβδομη έκδοση της Vuforia εισήχθη το Vuforia Fusion το οποίο δίνει τη δυνατότητα στις αναπτυσσόμενες εφαρμογές δυνατότητες SLAM. Έτσι έχουμε τη δυνατότητα να καταλαβαίνουμε την κίνηση της συσκευής στους τρεις άξονες XYZ και να έχουμε γνώση της θέσης της στον κόσμο. Όπως είδαμε προηγουμένως χάρη στο Extended Tracking μπορούμε να κρατάμε σχετικές θέσεις των αντικειμένων στον πραγματικό κόσμο. Αυτές οι δυνατότητες οδήγησαν στην ανάπτυξη της λύσης η οποία βασίζεται σε computer vision και στο INS αφού γίνεται χρήση των αισθητήρων της Android συσκευής.

Η ιδέα βασίζεται στο SLAM και τον χάρτη που δημιουργείται. Εφόσον, η Vuforia δημιουργεί έναν εικονικό χάρτη για το που βρίσκεται ο χρήστης ενώ μετακινείται γιατί να μην παράγει τον ίδιο χάρτη κάθε φορά που τρέχει η εφαρμογή. Αν οι χάρτες είχαν κοινό σημείο αναφοράς, όλες οι μετακινήσεις του χρήστη θα κατέληγαν στις ίδιες συντεταγμένες.

Χρησιμοποιώντας τα παραπάνω θα χωρίσουμε την εφαρμογή σε δύο μέρη. Κύρια ενέργεια και στα δύο μέρη με την εκκίνηση τους είναι το σκανάρισμα ενός target που αποτελεί το σημείο αναφοράς τους. Το πρώτο είναι το μέρος του σκαναρίσματος του χώρου. Δηλαδή, ο χρήστης αφού σκανάρει τον target, και ουσιαστικά δηλώσει στην εφαρμογή πως βρίσκεται στο (0,0,0), προχωράει και χρησιμοποιώντας το ground plane αφήνει στίγματα στις συντεταγμένες από τις οποίες πέρασε. Εδώ, φαίνεται χρήσιμο το Extended Tracking αφού τα στίγματα αυτά θα παραμείνουν στο ίδιο σημείο σε όλη τη διάρκεια της λειτουργίας της εφαρμογής. Οπότε, αυτό που τελικά πρέπει να κάνει ο χρήστης είναι να περάσει από όλα τα πιθανά μονοπάτια που μπορεί να βρεθεί και να αφήσει στίγματα.

Το δεύτερο μέρος είναι το λειτουργικό και περιγράφει την κανονική λειτουργία της εφαρμογής. Μόλις ο χρήστης σκανάρει τον target και επιλέξει ένα σημείο στο οποίο θέλει να πάει, η εφαρμογή ξέρει όλα τα πιθανά σημεία από τα οποία μπορεί να περάσει για να καταλήξει στο επιθυμητό σημείο μέσω των στιγμάτων. Στη συνέχεια υπολογίζεται με κάποιον αλγόριθμο της επιλογής μας ένα μονοπάτι και εμφανίζονται στην οθόνη του χρήστη οδηγίες για ποιο δρόμο να ακολουθήσει.

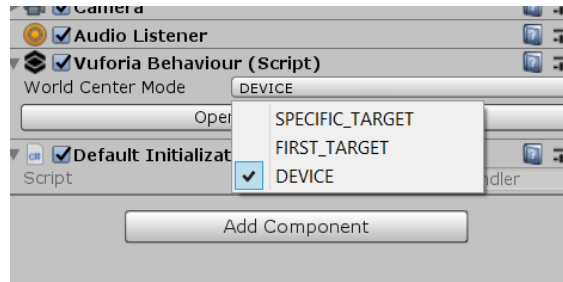


**Σχήμα 6.3:** Ο Χάνσελ και η Γκρέτελ στο ομώνυμο παραμύθι αφήνουν πίσω τους ψίχουλα για να βρουν τον δρόμο της επιστροφής.

## 6.2.2 Υλοποίηση

### Δημιουργία του χάρτη

Όπως είπαμε πρέπει ο χάρτης που δημιουργείται από το SLAM της Vuforia πρέπει να είναι κάθε φορά ο ίδιος. Για αυτό θα χρησιμοποιήσουμε το Positional Tracker της, μία λειτουργία που ενεργοποιεί το Ground Plane και το Extended Tracking για βελτιωμένες AR εμπειρίες. Η ρύθμιση που θα μας βοηθήσει να υλοποιήσουμε το στόχο μας είναι το World Center Mode στο Vuforia Behaviour script του ARCamera component. Με αυτή μπορούμε να δηλώσουμε ποιο θα είναι το κέντρο του κόσμου που δημιουργεί η Vuforia. Οι διαθέσιμες επιλογές είναι Device, που το κέντρο του κόσμου είναι η συσκευή και φαίνεται τα targets να κινούνται γύρω από αυτήν, FIRST TARGET, που το κέντρο του κόσμου είναι ο πρώτος γνωστός target στη Vuforia που θα σκαναριστεί ή SPECIFIC TARGET, που δηλώνουμε ως κέντρο έναν target από τη διαθέσιμη βάση. Η επιλογή που καλύπτει τις ανάγκες μας είναι το specific, έτσι όπου και να εκκινήσει το δεύτερο μέρος της εφαρμογής μόλις σκαναριστεί ο target ο κόσμος της εφαρμογής θα κεντραριστεί με αυτόν του πρώτου.



**Σχήμα 6.4:** Οι διαθέσιμες επιλογές του World Center Mode.

Σύμφωνα, με την επιλογή που κάναμε στο world center mode φαίνεται να κινούνται όλα γύρω από τον target, το ίδιο και το ARCamera. Το ARCamera component αντιπροσωπεύει την κάμερα της συσκευής που χρησιμοποιείται και τις κινήσεις αυτής. Οπότε αυτό που χρειαζόμαστε είναι οι συντεταγμένες του component στον κόσμο της Vuforia και θα έχουμε τα μέρη από τα οποία περνάει. Σε αυτό θα μας βοηθήσει το API της Unity. Μέσα στο script που είναι component του GameObject που θέλουμε, αναφερόμενοι στο transform.position παίρνουμε ως αποτέλεσμα μία τούπλα με τρία στοιχεία τύπου float που δίνουν τις συντεταγμένες του GameObject στους τρεις άξονες.

Τώρα πρέπει να βρούμε με ποιον τρόπο θα αφήσουμε τα εικονικά στίγματα για να καταγράψουμε όλα τα πιθανά μονοπάτια. Για αυτό θα χρησιμοποιήσουμε το mid air positioner της Vuforia, (η μόνη διαφορά με το ground plane είναι ότι τα αντικείμενα αιωρούνται και θα είναι πιο διακριτά στην πορεία του χρήστη). Έτσι, καθώς ο χρήστης κινείται θα μπορεί όπου θελήσει και υπάρχει επιφάνεια-δρόμο να αφήνει ένα στίγμα.

Θα φτιάξουμε ένα script (location.cs) που θα παίρνει τη θέση του gameobject και επιλέγοντας ένα απλό GameObject θα δημιουργήσουμε ένα prefab. Αυτό το prefab θα χρησιμοποιείται όποτε ο χρήστης αφήνει ένα στίγμα στο μονοπάτι.

## Βάσεις δεδομένων

Κάπου εδώ γίνεται φανερό ότι θα χρειαστούμε κάποιου είδους βάση δεδομένων. Αρχικά γιατί χρειάζεται να αποθηκεύσουμε τα σημεία μας για να μπορούμε να εκμεταλλευτούμε μελλοντικά, αλλά, και για να μπορούν να περαστούν από το πρώτο μέρος της εφαρμογής στο δεύτερο.

Για να υπάρχει η δυνατότητα να τρέξουν τα δύο μέρη ως μεμονωμένες εφαρμογές θεωρήθηκε σωστό η χρήση μίας διαδικτυακής βάσης. Η επιλογή μας ήταν το Firebase [Fire], λόγω της ευκολίας στο στήσιμο και στην χρήση του, καθώς παρέχει υποστηριγμένη έκδοση για unity και παραμένει δωρεάν για το πλήθος δεδομένων που θέλουμε να αποθηκεύσουμε. Το Firebase είναι ένα προϊόν της Google που χρησιμοποιείται κυρίως σε εφαρμογές κινητών και πέρα από αποθήκευση προσφέρει και πιο αναπτυγμένες δυνατότητες όπως analytics, crashlytics, user authentication κα. Είναι χρήσιμο να αναφέρουμε ότι το μόνο που χρειάστηκε από μέρους μας ήταν το integration του sdk. Σε περίπτωση που είχαμε επιλέξει μία παραδοσιακή βάση δεδομένων θα χρειαζόταν να δεσμεύσουμε κάποιο μηχάνημα καθώς και να γραφτεί κώδικας στο backend.

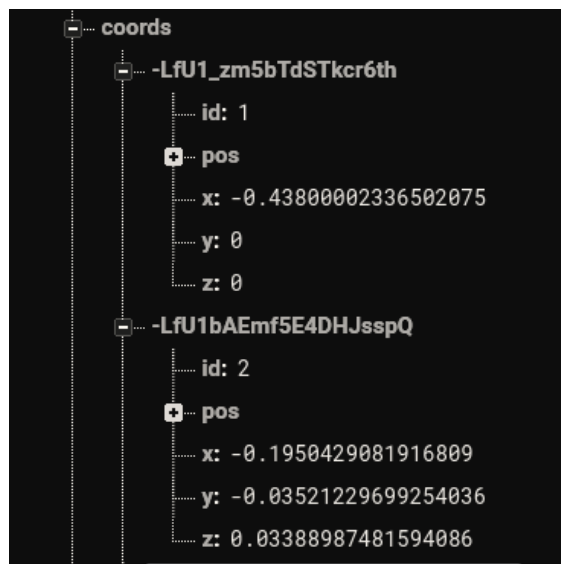
Αρα με το Firebase μπορούμε να προσθέσουμε τον κατάλληλο κώδικα στο location.cs και να στέλνουμε κάθε στίγμα που αφήνουμε.

Το Firebase, μας λύνει το πρόβλημα της μεταφοράς των δεδομένων από το ένα μέρος της εφαρμογής στο άλλο. Ωστόσο, θα επιβαρύνουμε την εφαρμογή με πολλά ασχρηστά call κάθε φορά που θα θέλαμε να κάνουμε μία επεξεργασία στα δεδομένα, πράγμα που θα κα-

```
// Get the root reference location of the database.
DatabaseReference reference = FirebaseDatabase.DefaultInstance.RootReference;
pos = this.transform.position;
Debug.Log("numberOfNodes"+DBInit.numOfNodes);
Node node = new Node(pos, ++DBInit.numOfNodes);
string json = JsonUtility.ToJson(node);
string key = reference.Push().Key;
reference.Child("coords").Child(key).SetRawJsonValueAsync(json);
```

**Σχήμα 6.5:** Ο κώδικας που παίρνει τις συντεταγμένες του prefab που τοποθετήθηκε και τις στέλνει στο Firebase

θυστερούσε αρκετά την εφαρμογή. Έτσι, θα ήταν χρήσιμο να γίνει χρήση και μίας βάσης δεδομένων σε τοπικό επίπεδο ώστε να ελαχιστοποιήσουμε τα call προς το Firebase. Για αυτή τη δουλειά χρησιμοποιήθηκε η SQLite.



**Σχήμα 6.6:** Αποθήκευση δεδομένων σε Firebase που αποτελεί No-SQL βάση.

Όπως, προδίδει το όνομα της πρόκειται για μία Structured Query Language, που σημαίνει ότι τα δεδομένα είναι αποθηκευμένα σε πίνακες. Μια no-SQL θα ήταν αρκετά πιο απαιτητική σε πόρους και θα επιβάρυνε την εφαρμογή.

Μιλώντας για πόρους, οι βάσεις δεδομένων είχαν σχεδιαστεί για υπολογιστές και servers για να αποθηκεύονται μεγάλες ποσότητες δεδομένων. Βέβαια, στην περίπτωση μίας εφαρμογής για κινητά δεν έχουμε στη διάθεση μας τόσους πόρους, αλλά, ούτε και μας χρειάζεται. Για αυτό είναι χρήσιμη η SQLite, που προσφέρει τις βασικές δυνατότητες, χωρίς τις πιο απαιτητικές, για την εξασφάλιση πόρων. Είναι ελαφριά στο τρέξιμο, έχει λιγότερους τύπους δεδομένων και δεν επιτρέπει την ταυτόχρονη σύνδεση πολλαπλών χρηστών.

Η SQLite είναι ιδιαίτερα διαδεδομένη στους developers για εφαρμογές κινητών και συνίσταται στο εγχειρίδιο του Android. Ωστόσο, δεν υπήρχε επίσημη έκδοση για Unity και χρειάστηκε να χρησιμοποιηθεί μία βιβλιοθήκη ανοιχτού κώδικα που προσέφερε τα κατάλληλα dll assembly αρχεία για την ενσωμάτωση της[SQLi].

## Δημιουργία της πλοήγησης

Σε αυτή τη φάση της εφαρμογής η λίστα με όλα τα πιθανά σημεία είναι έτοιμη. Το πρώτο πράγμα που πρέπει να ελεγχθεί είναι αν έχουν φορτωθεί τα σημεία από το Firebase και υπάρχουν στην τοπική βάση δεδομένων, αλλιώς, πρέπει να γίνει το κατάλληλο request για να τα κατεβάσουμε. Αυτά αποθηκεύονται στη συνέχεια σε κατάλληλο πίνακα. Αυτή η διαδικασία γίνεται ασύγχρονα λόγω υλοποίησης Firebase και δε συμβαίνει στο main thread οπότε δεν έχουμε να φοβόμαστε για τυχόν lags της εφαρμογής και μείωση της εμπειρίας του χρήστη. Ωστόσο, η ασύγχρονη μεταφορά των δεδομένων μας δημιούργησε ένα διαφορετικού είδους πρόβλημα. Αρχικά κανένα thread εκτός από main δεν έχει πρόσβαση στο API της unity οπότε υποκειμέθα σε αρκετούς περιορισμούς από άποψη δυνατοτήτων. Όμως το πιο βασικό πρόβλημα μας είναι ότι κάθε παράληψη στοιχείου συμβαίνει σε διαφορετικό thread με αποτέλεσμα να μην γνωρίζουμε αν έχουν παραληφθεί όλα τα σημεία ή το πότε θα συμβεί αυτό. Αυτό δημιουργεί την ανάγκη για να χρησιμοποιηθεί κάποιου είδους μεταβλητή καταμέτρησης σε συνδυασμό με κάποιου είδους σημαφόρο για να αντιμετωπίσουμε τα ταυτόχρονα writes στη μεταβλητή. Η λύση που μας αντιπετοπίζει και τα δύο προβλήματα είναι μία βιβλιοθήκη η οποία μας βοηθάει να στέλνουμε κώδικα προς εκτέλεση στο main thread. Με αυτόν τον τρόπο έχουμε και πρόσβαση στο api και γλιτώνουμε τις περίπλοκες λύσεις με σημαφόρους για την ορθή καταμέτρηση των λήψεων.

Μέχρι να ολοκληρωθεί η διαδικασία που περιγράψαμε ο χρήστης δεν έχει τη δυνατότητα να χρησιμοποιήσει τη δυνατότητα της περιήγησης. Όμως, στη συνέχεια η εφαρμογή μπαίνει στην κανονική κατάσταση. Αρχικά, παίρνουμε τη θέση του χρήστη στον κόσμο και ψάχνουμε να βρούμε το πιο κοντινό μαρκαρισμένο σημείο. Με βάση το σημείο αυτό πρέπει να δημιουργήσουμε μία διαδρομή για τον χρήστη. Για να καταφέρουμε να βρούμε το κοντινότερο μονοπάτι χρησιμοποιήσαμε τον αλγόριθμο A star.

## Αναζήτηση A star: Ελαχιστοποίηση του ολικού εκτιμώμενου κόστους λύσης

Ο A star είναι ένας ευριστικός αλγόριθμος χρησιμοποιούμενος κυρίως σε γράφους με βάρη[AStar]. Ξεκινώντας από έναν αρχικό κόμβο του γράφου, στοχεύει στο να βρει ένα μονοπάτι προς ένα κόμβο βρίσκοντας το μονοπάτι με λιγότερο κόστος (μικρότερη διαδρομή, συνολικό βάρος). Αυτό επιτυγχάνεται κρατώντας ένα δέντρο από μονοπάτια από τον αρχικό κόμβο και επεκτείνονται κατά μία ακμή τη φορά μέχρι να εκπληρωθεί το κριτήριο τερματισμού.

Σε κάθε επανάληψη του κύριου βρόχου, ο αλγόριθμος πρέπει να αποφασίσει ποια μονοπάτια θα επεκτείνει. Αυτό το κάνει σύμφωνα με το κόστος του ήδη σχηματισμένου μονοπατιού και μία εκτίμηση του κόστους που θα χρειαστεί η επέκταση του εξεταζόμενου μονοπατιού όλη μέχρι τον τελικό κόμβο. Πιο συγκεκριμένα, προσπαθεί να μειώσει το

$$f(x) = g(n) + h(n)$$

όπου με  $n$  αναφερόμαστε στον επόμενο κόμβο του μονοπατιού,  $g(n)$  είναι το κόστος του μονοπατιού από τον αρχικό κόμβο στο  $n$  και  $h(n)$  είναι η ευριστική συνάρτηση που υπολογίζει το κόστος του φθηνότερου μονοπατιού από το  $n$  στον τελικό κόμβο. Ο αλγόριθμος τερματίζει όταν το μονοπάτι που εξετάζεται για επέκταση είναι αυτό τον αρχικό κόμβο στον τελικό ή δεν υπάρχουν διαθέσιμα μονοπάτια προς επέκταση. Η ευριστική δεν είναι καθολική, πρέπει να οριστεί με βάση το εκάστοτε πρόβλημα. Αν η ευριστική δεν υπερεκτιμάει το πραγματικό κόστος για τον στόχο, τότε είναι βέβαιο ότι θα επιστραφεί το λιγότερο κοστοβόρο μονοπάτι.



Τυπικές υλοποιήσεις χρησιμοποιούν priority queue για να εκτελέσουν την επαναλαμβανόμενη επιλογή των λιγότερων κοστοβόρων μονοπατιών. Αυτό το priority queue ονομάζεται *open set*. Σε κάθε κομμάτι του αλγορίθμου, ο κόμβος με το μικρότερο  $f(x)$  εξάγεται από το queue, τα  $f$  και  $g$  των γειτόνων του ενημερώνονται αντίστοιχα, και στη συνέχεια αυτοί εισάγονται στο queue. Ο αλγόριθμος συνεχίζει μέχρι ο τελικός κόμβος να έχει μικρότερο  $f$  από κάθε άλλο κόμβο στο queue ή μέχρι αυτό να είναι άδειο. Το  $f$  είναι η τιμή του συντομότερου μονοπατιού, αφού το  $h$  είναι μηδέν.

Ο αλγόριθμος ως εδώ βρίσκει το κόστος του συντομότερου μονοπατιού. Για να βρούμε το ίδιο το μονοπάτι, ο αλγόριθμος χρειάζεται μία προσαρμογή ώστε κάθε κόμβος να αποθηκεύει τον προηγούμενο του. Έτσι, με τον τερματισμό του αλγορίθμου ξεκινώντας από τον τελικό κόμβο έχουμε όλη τη διαδρομή μέχρι την αρχή.

Για παράδειγμα, όταν ψάχνουμε για συντομότερο μονοπάτι πάνω σε έναν χάρτη, το  $h(x)$  μπορεί να είναι η απόσταση σε ευθεία γραμμή μεταξύ δύο κόμβων, αφού αποτελεί την μικρότερη δυνατή απόσταση.



## Κεφάλαιο 7

# Η εφαρμογή

## 7.1 Εισαγωγή

Πρόκειται για μία εφαρμογή επαυξημένης πραγματικότητας που σκοπό έχει να διευκολύνει τους εργαζομένους των αποθηκευτικών χώρων στη διαδικασία του picking, μέσω ενός απλού και εύχρηστου γραφικού περιβάλλοντος. Η εφαρμογή αναπτύχθηκε κατά τη διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας.

Μέσω της εφαρμογής ο χρήστης έχει τη δυνατότητα να εκτελέσει οποιαδήποτε ενεργεία είναι απαραίτητη για τη διαδικασία του picking, που στην παρούσα φάση γίνεται με χαρτί και μολύβι και με τη χρήση scanner χειρός. Το κύριο πλεονέκτημα της εφαρμογής είναι ότι καθόλη τη διάρκεια της διαδικασίας ο χρήστης έχει και τα δύο χέρια αποδεσμευμένα για να μπορεί να ασχοληθεί με το picking ενόσω όλες οι απαραίτητες πληροφορίες είναι στο οπτικό πεδίο μέσω των γυαλιών. Όλες οι κινήσεις του χρήστη που απαιτούσαν χαρτί και μολύβι τώρα γίνονται πιο απλά με το πάτημα κουμπιών ή με φωνητικές εντολές.


Η εφαρμογή παρέχει οποιαδήποτε πληροφορία είναι απαραίτητη για τον χρήστη, όπως θέση των αντικειμένων, σχετική θέση του χρήστη με αυτή, ποσότητα για συλλογή κα. Επίσης, τα barcode scanners χειρός δεν είναι πλέον απαραίτητα αφού η διαδικασία σκαναρίσματος μπορεί να υλοποιηθεί από την κάμερα των γυαλιών.

## 7.2 Περιγραφή της διαδικασίας picking

Απαραίτητη προϋπόθεση για τη δημιουργία μίας εφαρμογής εύχρηστης, η οποία ανταποκρίνεται στις ανάγκες του χρήστη, είναι η κατανόηση και η αναλυτική καταγραφή της διαδικασίας του picking, όλων των διακλαδώσεων της ροής, όπως και σφαλμάτων που μπορεί να τη διακόψουν.












Με τον όρο picking περιγράφουμε τη συλλογή αντικειμένων και αγαθών από χώρους και ράφια μίας αποθήκης σύμφωνα με τις προκύπτουσες ανάγκες των πελατών. Δηλαδή, ο εργαζόμενος πρέπει να παραλάβει αντικείμενα από καθορισμένες θέσεις μίας αποθήκης με συγκεκριμένη σειρά και στη συνέχεια πρέπει να παραδώσει τα συλλεγμένα αγαθά στην επόμενη θέση τους μέσα στην αποθήκη, υπηρετώντας έτσι τη σχεδιασμένη διαδικασία ροής υλικών εντός του κέντρου διανομής.

Τυπικά, η διαδικασία ξεκινά με μία λίστα συλλογής, η οποία καθορίζει τη θέση κάθε είδους προς συλλογή, την ποσότητα από κάθε είδος και την ακολουθία σύμφωνα με την οποία θα ληφθούν τα προϊόντα. Στην ουσία, ένας εργαζόμενος συλλέγει τα αντικείμενα από το απόθεμα, σύμφωνα με τη λίστα συλλογής (picking list) και τα μεταφέρει στο επόμενο βήμα της διαδικασίας, που είναι συνήθως η προετοιμασία της παραγγελίας και η φόρτωση προς αποστολή.

<b>Pick list</b>	
Clothing Company	
<b>Pick Order</b>	12
<b>Print date</b>	2018-11-23
<b>Picked by</b>	aje

OrderId	OrderNr	Bin
23452	001	1
23453	002	2
23454	003	3
23455	004	4
23456	005	5

**ongoing...**  
warehouse management systems

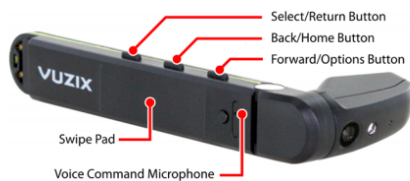
Location	Barcode	Article No	Article name	Quantity	Bin
B-11-04		Computer	Computer	2	1
B-11-04		Computer	Computer	1	2
B-11-04		Computer	Computer	2	3
B-11-04		Computer	Computer	2	5
F01-04-01		Computer Mouse	Computer Mouse	1	1
F01-04-01		Computer Mouse	Computer Mouse	2	3
F01-04-01		Computer Mouse	Computer Mouse	1	4
F01-04-01		Computer Mouse	Computer Mouse	1	5
F03-04-01		Keyboard	Keyboard	1	1
F03-04-01		Keyboard	Keyboard	1	4
F03-04-01		Keyboard	Keyboard	1	5

**Σχήμα 7.1:** Μία λίστα συλλογής - Picking List

Η συλλογή παραγγελιών χωρίζεται σε τέσσερα χρονικά στάδια α) base time - ο χρόνος μέχρι τη λήψη της επόμενης πληροφορίας συλλογής, β) dead time - ο χρόνος που απαιτείται μέχρι ο εργαζόμενος να αντιληφθεί, να επεξεργαστεί και να κατανοήσει την πληροφορία που έλαβε ώστε να είναι έτοιμος να δράσει, γ) way time - ο χρόνος κατά τον οποίο ο εργαζόμενος κινείται μέχρι να φθάσει στην ακριβή θέση τους αντικειμένου και δ) picking time - ο χρόνος μέχρι την πραγματική συλλογή του αντικειμένου.

Κατά το picking time στις αποθήκες που χρησιμοποιούν barcodes και scanners είναι απαραίτητο το σκανάρισμα για την ενημέρωση των αποθεμάτων στη βάση δεδομένων, όπως, και για έλεγχο ότι συλλέχθη το σωστό αντικείμενο. Στην αρχή σκανάρεται ο κωδικός που αφορά το είδος του προϊόντος για τον έλεγχο. Όσον αφορά το σκανάρισμα για την επαλήθευση της ποσότητας διακρίνουμε τρεις διαφορετικές περιπτώσεις. Στην πρώτη και πιο απλή περίπτωση αρκεί μόνο ο έλεγχος για τον τύπο του προϊόντος οπότε δε χρειάζεται επιπλέον σκανάρισμα. Στη δεύτερη, απαιτείται το σειριακό σκανάρισμα του κάθε πακέτου ξεχωριστά. Στην τρίτη και τελευταία περίπτωση ο συλλέκτης σκανάρει τα πακέτα σε δυάδες.

Πέρα από την κανονική ροή που περιγράφηκε υπάρχουν και κάποιες περιπτώσεις που αυτή διακόπτεται ή αλλάζει. Ο συλλέκτης για παράδειγμα μπορεί να αδυνατεί να ολοκληρώσει την παραγγελία, οπότε θα την αφήσει και θα ασχοληθεί με κάποια άλλη με αποτέλεσμα να πρέπει κάποιος άλλος να αναλάβει την ημιτελή παραγγελία και να γνωρίζει τι έχει συ και τι όχι. Επίσης, μπορεί να χρειαστεί να συλλέξει με διαφορετική σειρά τα πακέτα ή να συλλέξει από κάποιον προϊόν λιγότερα αποθέματα και να πρέπει να ενημερώσει τη βάση για αυτό.



(a) Τα τρία κουμπιά του Vuzix M300.



(b) Εκτός από τα τρία κουμπιά υπάρχει και ένα Pad.

## 7.3 Παρουσίαση της εφαρμογής

### 7.3.1 Σκανάρισμα του barcode της παραγγελίας

Η πρώτη οθόνη που παρουσιάζεται στον χρήστη τον προτρέπει πατήσει το κατάλληλο κουμπί ή να δώσει τη κατάλληλη φωνητική εντολή για μπει στη λειτουργία σκαναρίσματος. Κατά αυτή τη λειτουργία η εφαρμογή περιμένει να σκαναριστεί ένα barcode το οποίο αντιστοιχίζεται στον κωδικό μίας παραγγελίας, ώστε να μπορέσει να φορτώσει από τη βάση δεδομένων την αντίστοιχη λίστα συλλογής. Ο χρήστης χρησιμοποιώντας την φωνητική εντολή *start scan* πατώντας το μεσαίο κουμπί των γυαλιών μεταβαίνει στην οθόνη σκαναρίσματος.



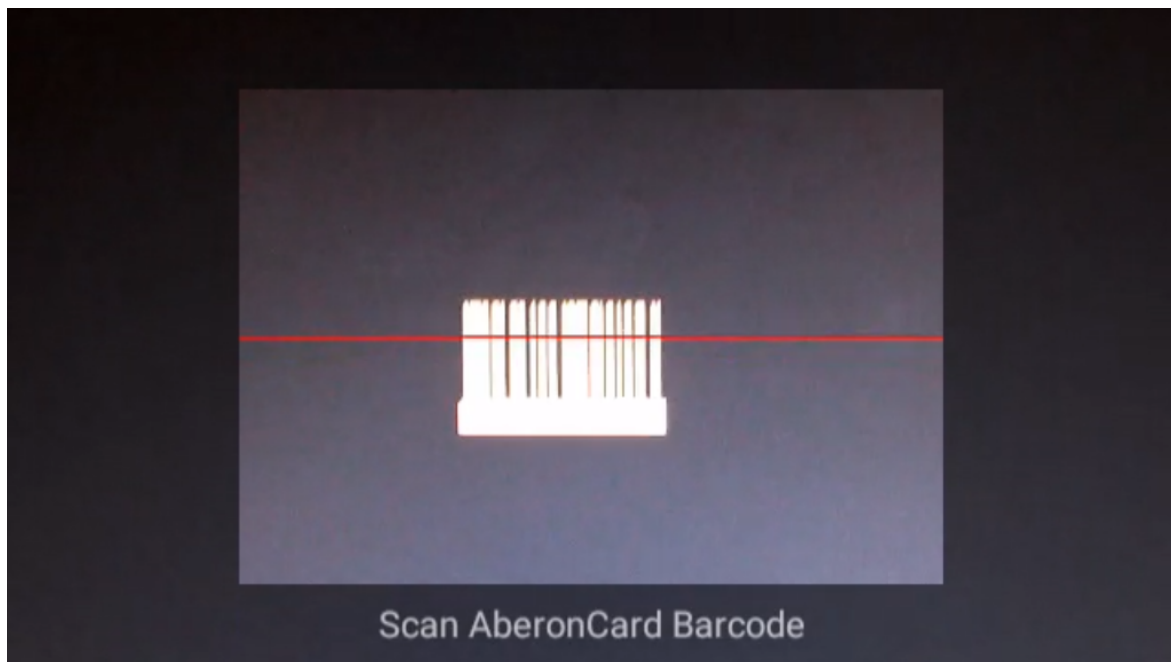
Σχήμα 7.2: Η πρώτη οθόνη της εφαρμογής

Κάπου εδώ φαίνεται χρήσιμο να δούμε τους τρόπους που μπορεί ο χρήστης να δώσει είσοδο στο γυαλί μέσω αφής.

Κάθε ένα από τα τρία κουμπιά που βρίσκονται στα γυαλιά έχει δύο τρόπους χρήσης. Ο πρώτος είναι το απλό πάτημα και ο δεύτερος είναι το παρατεταμένο πάτημα (long press). Όσον αφορά το Pad, αυτό έχει τέσσερις κατευθύνσεις, όπως, και και το διπλό πάτημα. Για να καταφέρουμε να χρησιμοποιήσουμε αυτούς τους τρόπους εισόδους χρειάστηκε να χρησιμοποιήσουμε το Event class της Unity και να διαβάσουμε την παράμετρο KeyCode. Έτσι αντιστοιχίσαμε κάθε πιθανή είσοδο σε ένα event για να μπορέσουμε να τη χρησιμοποιήσουμε ανάλογα με τις ανάγκες μας. Έτσι προέκυψε ο εξής πίνακας.

front key press	<i>RightArrow</i>
front key long press	<i>Menu</i>
middle key press	<i>LeftArrow</i>
rear key press	<i>Return</i>
rear key long press	<i>Escape</i>
swipe front	<i>RightArrow</i>
swipe rear	<i>LeftArrow</i>
swipe up	<i>UpArrow</i>
swipe down	<i>DownArrow</i>
pad double tap	<i>JoystickButton0</i>

Στην εικόνα που ακολουθεί φαίνεται η οθόνη του σκαναρίσματος του barcode της παραγγελίας.



**Σχήμα 7.4:** Σκανάρισμα του barcode της παραγγελίας

Για να καταφέρουμε να ενσωματώσουμε τη λειτουργία του σκαναρίσματος των barcode μέσω της κάμερας ήταν απαραίτητο να χρησιμοποιηθεί κάποια σχετική βιβλιοθήκη. Λόγω της αυξημένης χρήσης των smartphones τα τελευταία χρόνια υπάρχουν αρκετές βιβλιοθήκες που μπορούν να μας βοηθήσουν στο στόχο μας, αφού το σκανάρισμα των barcodes βρίσκει πολλές εφαρμογές σε καθημερινή βάση.

Πριν την τελική επιλογή, δοκιμάσαμε αρκετές βιβλιοθήκες μερικές από τις οποίες είναι το ZXing, ZBar, Scandit κα. Κάθε μία είχε τα δικά της πλεονεκτήματα και μειονεκτήματα. Ωστόσο, καμία δεν έφτανε στο ρυθμό ανταποκρισης που είχε η βιβλιοθήκη για barcodes από την ίδια τη Vuzix. Το οποίο είναι ένα εύλογο αποτέλεσμα εφόσον πρόκειται για μία βιβλιοθήκη βασισμένη πάνω στο hardware του συσκευριμένου γυαλιού και μπορεί να το εκμεταλευντεί πλήρως.

Ωστόσο, με αυτή την απόφαση δημιουργήθηκε η εξής δυσκολία. Η Vuzix δεν έχει κυκλοφορήσει api για τις βιβλιοθήκες της για κάποια άλλη γλώσσα πέρα από τη Java. Εφόσον, για την ανάπτυξη της εφαρμογής χρησιμοποιήσαμε τη Unity η γλώσσα ανάπτυξης είναι η C#, έτσι γίνεται φανερό ότι πρέπει να βρούμε ένα τρόπο να ενσωματώσουμε τη βιβλιοθήκη της Vuzix.

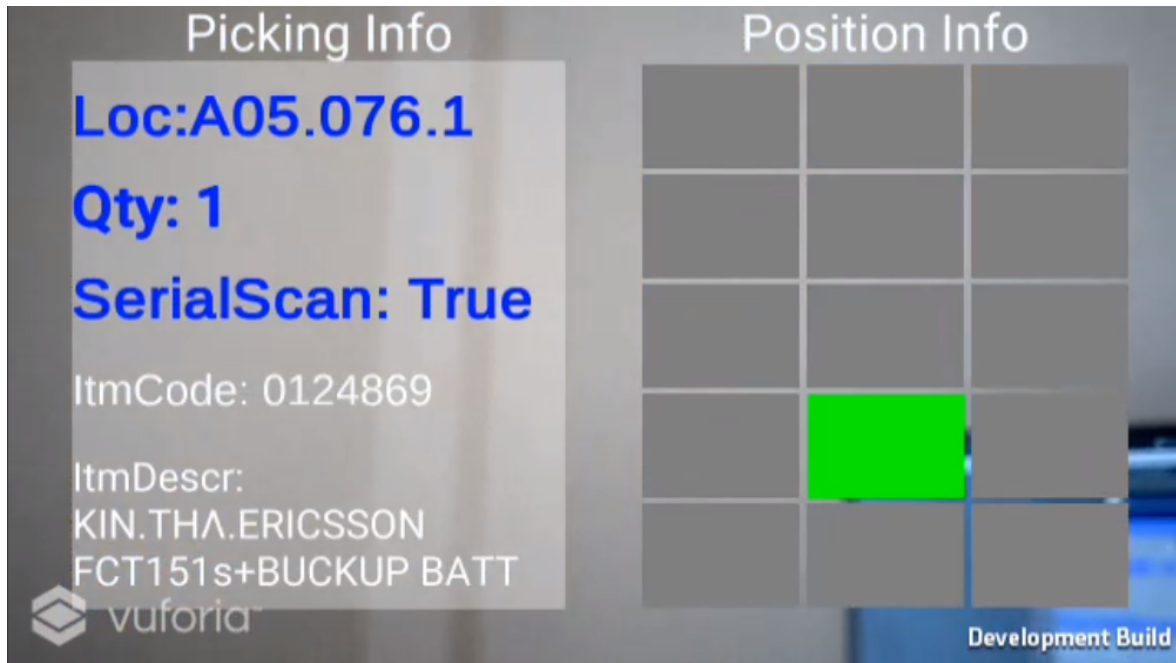
Τη λύση μας τη δίνει το api της Unity. Μέσω του AndroidJavaClass μπορούμε να έχουμε πρόσβαση στις κλάσεις βιβλιοθηκών που έχουμε εισάγει στη Unity και είναι εξόλοκληρου γραμμένες σε Java. Άρα αυτό που έχουμε να κάνουμε είναι να γράψουμε μία βιβλιοθήκη σε Java με το Android Studio χρησιμοποιώντας την βιβλιοθήκη barcode της Vuzix η οποία να λειτουργεί σύμφωνα με τις ανάγκες της εφαρμογής μας και να την εισάγουμε στη Unity στο φάκελο Plugins/Android ώστε να μπορεί να εντοπιστεί από την πλατφόρμα.

```
jclass = new AndroidJavaClass("com.unity3d.player.UnityPlayer");  
activity = jclass.GetStatic<AndroidJavaObject>("currentActivity");  
barcodeScannerSingle = new AndroidJavaClass("com.vuzix.sample.barcode_scan.MainActivity");
```

**Σχήμα 7.5:** Αρχικοποίηση μίας κλάσης γραμμένης σε Java μέσω Unity api

## Οδηγίες για τη συλλογή των πακέτων

Μετά την ολοκλήρωση του σκαναρίσματος γίνεται το κατάλληλο request στο api και φορτώνεται στην εφαρμογή η λίστα με τα πακέτα προς συλλογή. Η επόμενη οθόνη της εφαρμογής αφορά τις λεπτομέρειες που χρειάζεται ο χρήστης για να ολοκληρώσει τη συλλογή αυτή.



**Σχήμα 7.6:** Οι πληροφορίες που απεικονίζονται στον χρήστη μέχρι να φτάσει στη θέση του πακέτου που πρέπει να συλλέξει

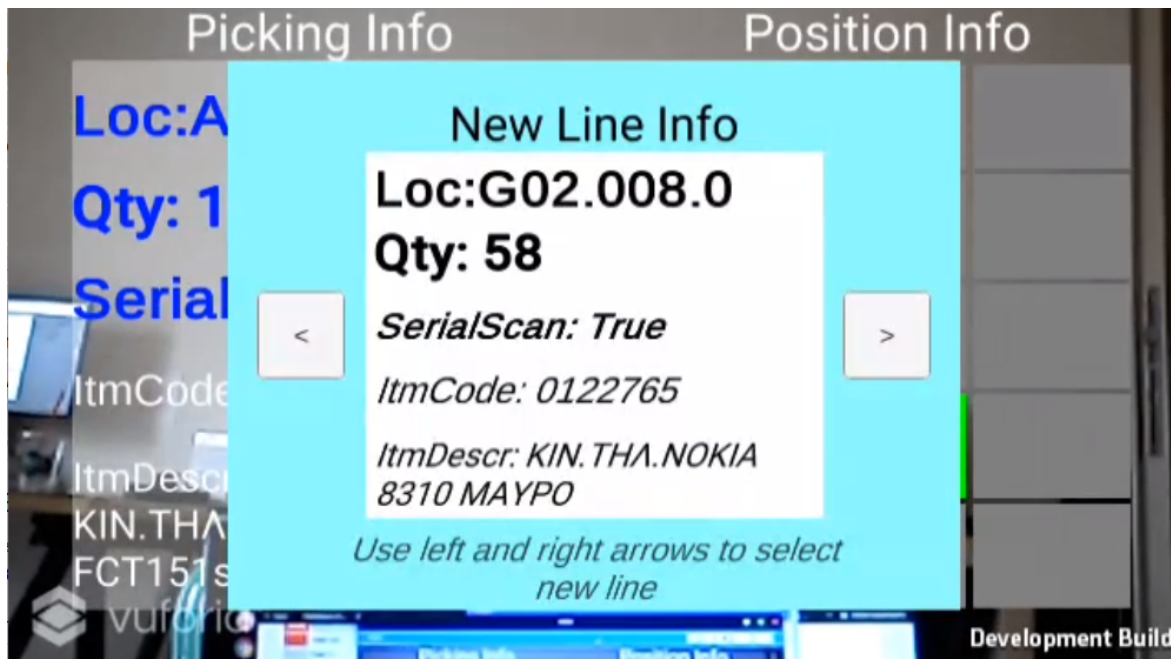
Όπως φαίνεται στην εικόνα η οθόνη χωρίζεται σε δύο μέρη. Στο αριστερό μισό εμφανίζονται οι πληροφορίες που αφορούν τη συλλογή. Πάνω εμφανίζεται η ακριβή θέση του πακέτου στην αποθήκη, δηλαδή, σε ποιο διάδρομο βρίσκεται, σε τι βάθος στο διάδρομο αυτό, όπως και το επίπεδο στο οποίο βρίσκεται στο ράφι. Η επόμενη πληροφορία αφορά την ποσότητα του προϊόντος που πρέπει να συλλέξει. Το πεδίο SerialScan που ακολουθεί είναι αυτό που κρίνει την ροή της εφαρμογής από δω και πέρα, είναι αυτό που ενημερώνει τον χρήστη για το εάν απαιτείται σειριακό σκανάρισμα σε ένα ένα στα κομμάτια που έχει να συλλέξει ή όχι. Κάτω από το SerialScan εμφανίζεται ο κωδικός του προϊόντος που είναι προς συλλογή. Τέλος, εμφανίζεται μία μικρή περιγραφή του προϊόντος η οποία βοηθάει το χρήστη στη συλλογή, αφού του δίνει μία ιδέα για το είδος του προϊόντος και τον κατευθύνει.

Όσον αφορά, το δεξί μισό της οθόνης αυτό που εμφανίζεται είναι μία αναπαράσταση του φαντώματος που θα συναντήσει ο χρήστης μόλις φτάσει στη θέση που πρέπει. Η πληροφορία που λαμβάνουμε από το api είναι ένας πίνακας από ακεραίους, που κάθε στοιχείο αντιπροσωπεύει ένα επίπεδο στο φάντωμα και η τιμή του ακεραίου τις στήλες ανά επίπεδο. Με αυτά τα δεδομένα δημιουργούμε την αναπαράσταση, ώστε καθώς κατευθύνεται ο χρήστης προς τον προορισμό να ξέρει πως θα είναι το φάντωμα και να ξέρει που να ψάξει σε αυτό.

### 7.3.2 Αλλαγή πακέτου προς συλλογή

Υπάρχουν αρκετοί λόγοι για τους οποίους ένας χρήστης μπορεί να χρειαστεί να αλλάξει τη σειρά με την οποία θα ολοκληρώσει την παραγγελία. Για παράδειγμα, μπορεί όταν φτά-

σει σε ένα πακέτο αυτό να είναι χτυπημένο οπότε να μη μπορεί να χρησιμοποιηθεί για την ολοκλήρωση της παραγγελίας, ή μπορεί να μην υπάρχει απόθεμα τη δεδομένη στιγμή και να περιμένει ανατροφοδότηση, οπότε μπορεί να συνεχίσει την παραγγελία μέχρι να ολοκληρωθεί και να μην είναι ανενεργός ωστόσο. Μία συνηθισμένη περίπτωση είναι να μην χρησιμοποιείται κάποιος αλγόριθμος σύντομου μονοπατιού με τις θέσεις των πακέτων στο Backend του αρι που χρησιμοποιείται, με αποτέλεσμα ο ίδιος ο χρήστης να πρέπει να ταξινομήσει την συλλογή κατάλληλα για μείωση τη μετακίνηση του μέσα στην αποθήκη. Για αυτούς τους λόγους, έχει υλοποιηθεί η επόμενη οθόνη που επιτρέπει στον χρήστη να διατρέχει όλη την λίστα με τα πακέτα που έχει να συλλέξει για την παραγγελία και να διαλέξει με ποιο από αυτά θα ασχοληθεί τη δεδομένη στιγμή, εφόσον πατήσει παρατεταμένα το μπροστινό κουμπί.



**Σχήμα 7.7:** Ο χρήστης μπορεί να δει όλα τα πακέτα προς συλλογή της παραγγελίας και μπορεί να διαλέξει ποίο θέλει να συλλέξει πρώτα.

### 7.3.3 Αλλαγή παραγγελίας

Εκτός από την αλλαγή πακέτου προς συλλογή, υπάρχουν αρκετοί λόγοι που μπορεί να οδηγήσουν τον χρήστη να αλλάξει ολόκληρη την παραγγελία με την οποία ασχολείται. Ένας από αυτούς είναι η αδυναμία συλλογής παραπάνω τους ενός πακέτου και για να είναι πιο παραγωγικός ο χρόνος του να αρχίσει την επόμενη παραγγελία. Οι επείγουσες παραγγελίες είναι ακόμα ένας λόγος που θα μπορούσε να οδηγήσει τον χρήστη να αλλάξει την παραγγελία με την οποία ασχολείται. Η δυνατότητα αυτή δίνεται στο χρήστη ενόσω είναι στην οθόνη με τις πληροφορίες πατώντας παρατεταμένα το πίσω κουμπί. Οπότε και εμφανίζεται η ακόλουθη οθόνη.

Για επιβεβαίωση ο χρήστης πρέπει να πατήσει παρατεταμένα το πίσω κουμπί.

### 7.3.4 Χάρτης της αποθήκης

Όπως είδαμε στο αντίστοιχο κεφάλαιο, αναπτύξαμε ένα σύστημα πλοήγησης εσωτερικού χώρου το οποίο βασίζεται εξόλοκληρου στους αισθητήρες της συσκευής, όπως και στους αλ-





**Σχήμα 7.8:** Ο χρήστης μπορεί να σταματήσει την τρέχουσα παραγγελία και να σκανάρει το barcode κάποιας άλλης για να την ξεκινήσει

γορίθμους slam της Vuforia με το extended tracking. Ωστόσο, το hardware των Vuzix M300, όντας τα ίδια μία μικρή σχετικά κινητή συσκευή δε μπορεί να διαθέσει ικανή υπολογιστική ισχύ για μία τέτοια λειτουργία. Επίσης, το λειτουργικό τους σύστημα είναι Android 6.0, μία έκδοση η οποία δε μπορεί να υποστηρίξει το ARCore, το οποίο χρησιμοποιεί Vuforia στις Android συσκευές. Έτσι, η Vuforia στηρίζεται σε εναλλακτικούς αλγορίθμους για αυτές τις λειτουργίες [Vufo], με συνέπεια να μην έχουμε το επιθυμητό αποτέλεσμα.

Έτσι, χρειάστηκε να υλοποιήσουμε μία διαφορετική λειτουργία για την περιήγηση στην αποθήκη, λιγότερο απαιτητική σε πόρους. Το κομμάτι που είναι ιδιαίτερα αποδοτική η Vuforia είναι αυτό της αναγνώρισης markers και της προβολής των έξτρα εικονικών στοιχείων στον πραγματικό κόστος όσο ο marker μένει στο οπτικό πεδίο της κάμερας. Η λύση μας στηρίζεται σε αυτό και στη γνώση του σχεδίου της αποθήκης. Ουσιαστικά, τοποθετούμε markers σε συγκεκριμένα μέρη στην αποθήκη και φτιάχνουμε μία λίστα αντιστοίχισης marker και θέσης. Καθώς ο χρήστης βρίσκεται στο δρόμο προς τη συλλογή του πακέτου και βλέπει τις πληροφορίες στην οθόνη του μπορεί να στρέψει το βλέμμα του σε οποιονδήποτε marker που έχουμε τοποθετήσει και να δει τη σχετική θέση του με το πακέτο μέσα στην αποθήκη. Η οθόνη αυτή φαίνεται στην παρακάτω εικόνα, όπου βλέπουμε τον χρήστη στην είσοδο ενός διαδρόμου, αναπαριστώμενο από ένα ανθρωπάκι και με ένα πράσινο βέλος το διάδρομο του πακέτου, το βάθος σε αυτόν, όπως και την πλευρά.

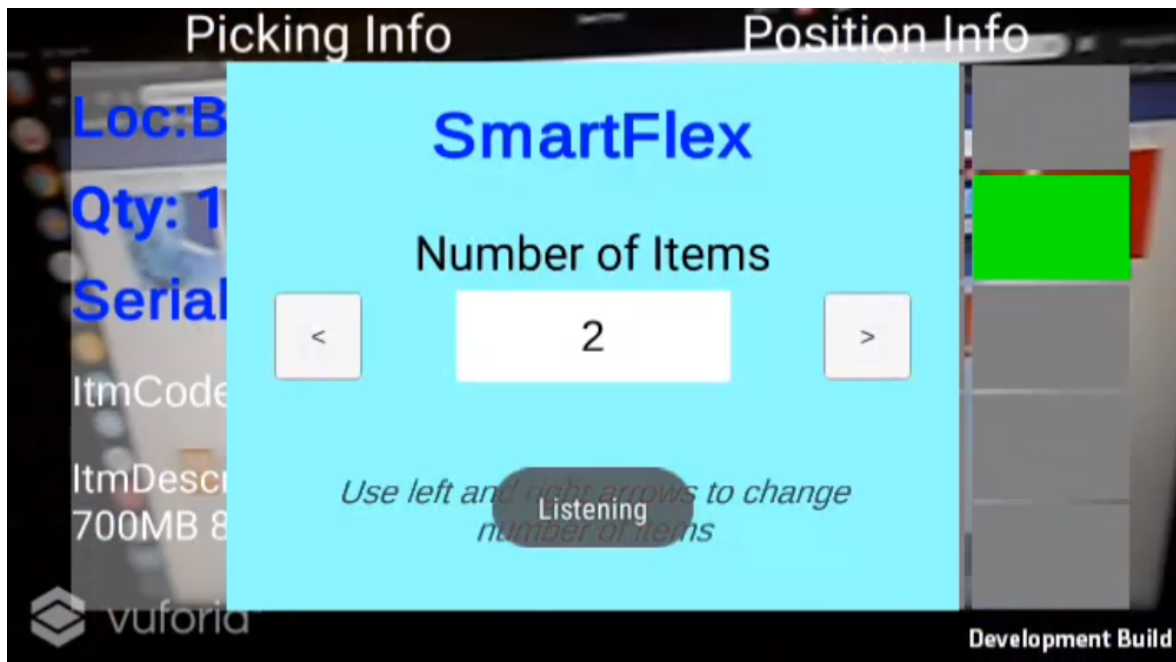


**Σχήμα 7.9:** Ο χάρτης της αποθήκης στον οποίο φαίνεται η σχετική θέση του χρήστη με το πακέτο προς συλλογή.

### 7.3.5 Αλλαγή ποσότητας

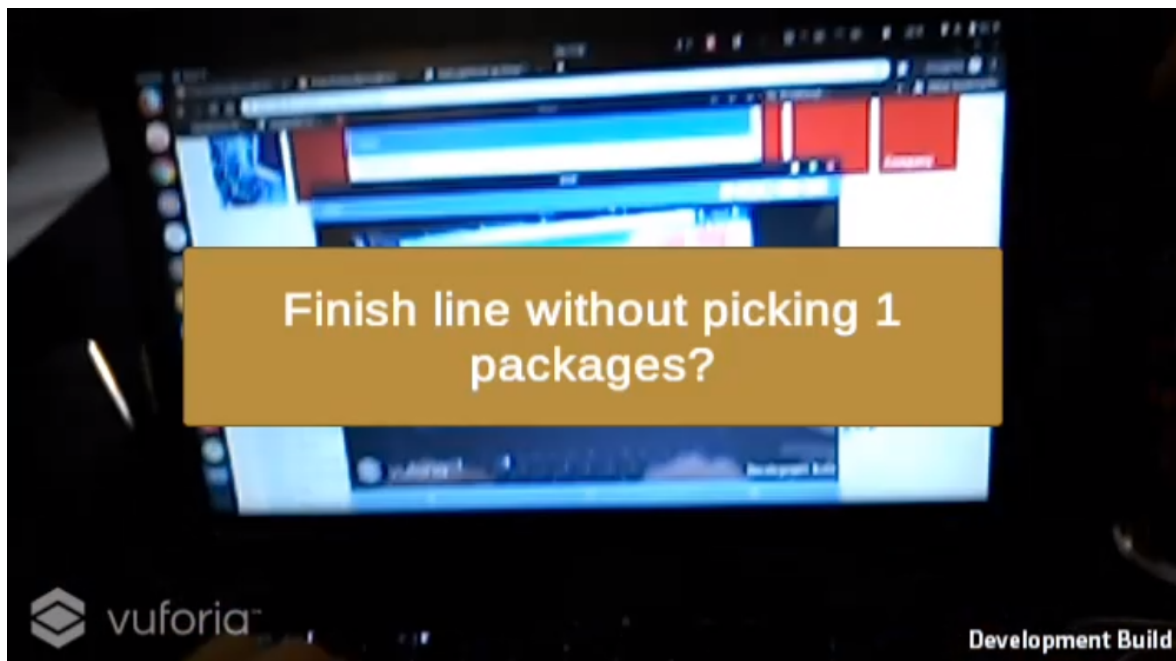
Μία χρήσιμη ακόμα δυνατότητα που μπορούμε να δώσουμε στον χρήστη για την ολοκλήρωση της παραγγελίας είναι η αλλαγή της ποσότητας του πακέτου προς συλλογή. Αυτή η δυνατότητα φαίνεται χρήσιμη, για παράδειγμα, όταν η ποσότητα στα ράφια δεν φτάνει για την παραγγελία και η τροφοδοσία του πακέτου δε προβλέπεται να γίνει στο εγγύς μέλλον. Η παραγγελία δε μπορεί να παραμείνει ενεργή, οπότε πρέπει να δίνουμε τη δυνατότητα στο χρήστη να ολοκληρώσει τη συλλογή του πακέτου με μειωμένο απόθεμα με κατάλληλη επισήμανση στο σύστημα.

Όταν το πακέτο προς συλλογή δεν απαιτεί σειριακό σκανάρισμα τότε με ένα πάτημα στο rad των γυαλιών εμφανίζεται η παρακάτω οθόνη που ζητά στον χρήστη να επιλέξει την ποσότητα που θα συλλέξει.



**Σχήμα 7.10:** Η οθόνη που επιλέγει ο χρήστης την ποσότητα που θα συλλέξει.

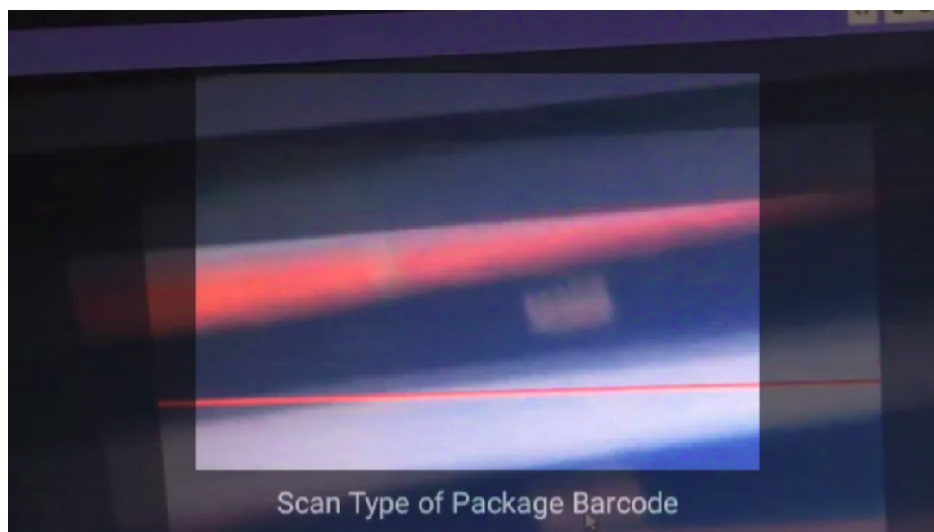
Όταν ωστόσο, πρόκειται για σειριακό σκανάρισμα ο χρήστης μπορεί απλώς να διακόψει τη διαδικασία σκαναρίσματος των πακέτων μόλις έχει σκανάρει και το τελευταίο πακέτο που έχει συλλέξει, πατώντας παρατεταμένα το πίσω κουμπί. Στη συνέχεια εμφανίζεται κατάλληλο μήνυμα, όπως, φαίνεται στην εικόνα 7.11, για επιβεβαίωση.



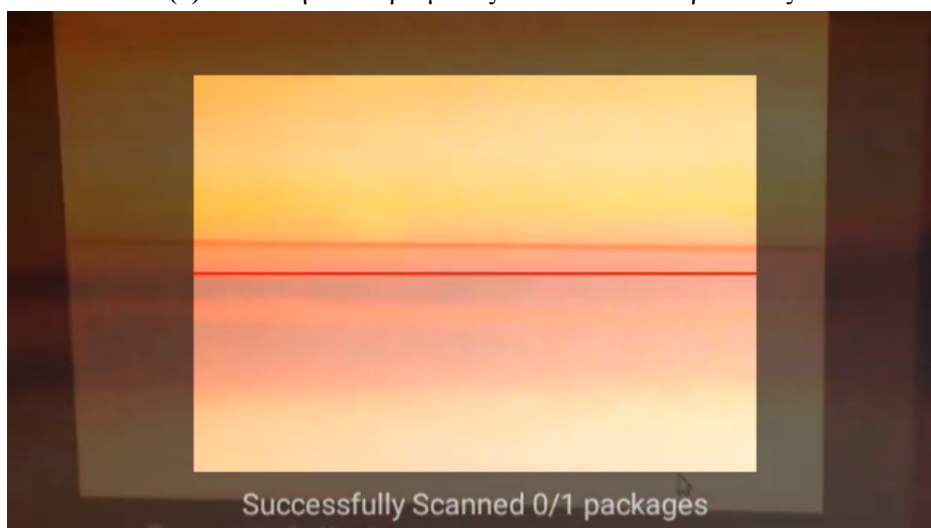
**Σχήμα 7.11:** Η ερώτηση για επιβεβαίωση διακοπής της διαδικασίας σκαναρίσματος με λιγότερα πακέτα συλλεγμένα.

### 7.3.6 Σκανάρισμα των κωδικών των προϊόντων

Μόλις φτάσει στο επιθυμητό σημείο ο χρήστης μπορεί να αρχίσει την συλλογή των πακέτων. Για τη συλλογή τους χρειάζεται το σκανάρισμα των barcodes τους. Όπως, έχουμε αναφέρει νωρίτερα διακρίνουμε δύο διαφορετικές περιπτώσεις για τη συλλογή πακέτων, τη σειριακή και τη μη σειριακή. Κατά τη σειριακή, αρχικά πρέπει ο χρήστης να σκανάρει το barcode που αντιστοιχεί στον κωδικό του προϊόντος, ο οποίος αναγράφεται στις πληροφορίες που βλέπει μέχρι να φτάσει στο πακέτο. Αν ο κωδικός αυτός είναι λανθασμένος, τότε η εφαρμογή τον προτρέπει να ξανασκανάρει ένα άλλο barcode. Μόλις σκανάρει με επιτυχία το σωστό barcode, τότε ξεκινάει η διαδικασία του σειριακού σκαναρίσματος κατά την οποία ο χρήστης πρέπει να σκανάρει ένα barcode για κάθε πακέτο που συλλέγει. Μία λίστα με τα έγκυρα barcodes έχουμε παραλάβει από το api.

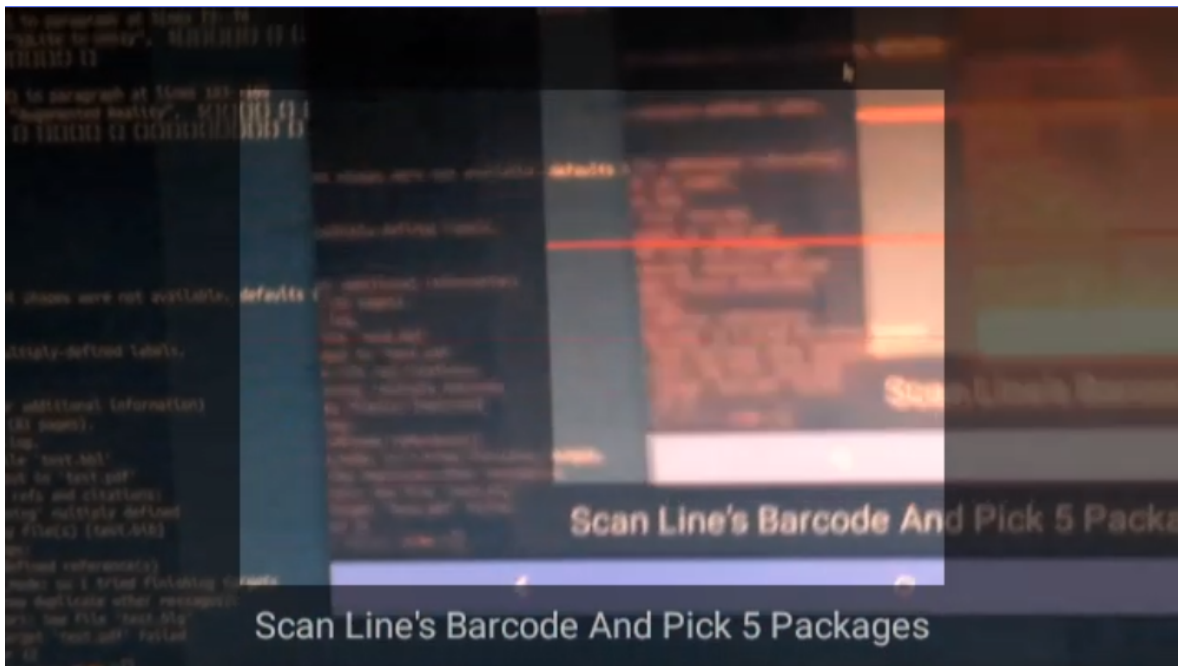


(a) Η οθόνη σκαναρίσματος του τύπου του προϊόντος.



(b) Το σειριακό σκανάρισμα των προϊόντων.

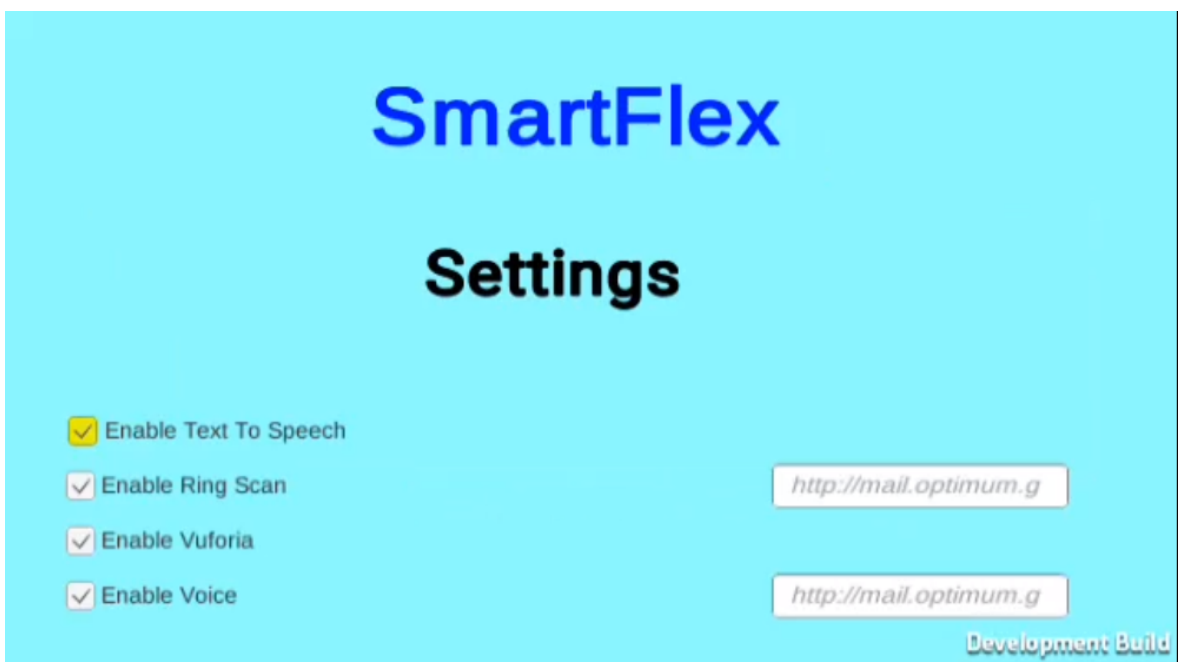
Όσον αφορά τη μη σειριακή περίπτωση είναι αρκετό ο χρήστης να σκανάρει έναν έγκυρος κωδικό και απλώς να συλλέξει τον αριθμό των πακέτων.



**Σχήμα 7.13:** Όταν δεν απαιτείται σειριακό σκανάρισμα, αρκεί το σκανάρισμα ενός μονάχα έγκυρου barcode.

### 7.3.7 Ρυθμίσεις της εφαρμογής

Στην εικόνα 7.14 βλέπουμε όλες τις πιθανές ρυθμίσεις που μπορεί να επεξεργαστεί ο χρήστης. Το κίτρινο μαρκάρισμα υποδηλώνει σε ποια από αυτές έχει εστιάσει τη δεδομένη στιγμή.



**Σχήμα 7.14:** Οι ρυθμίσεις της εφαρμογής.

Οι τέσσερις από αυτές είναι δυαδικές, δηλαδή, απαιτούν ναι ή όχι, ενώ οι άλλες δύο απαιτούν κείμενο.

## Text To Speech

Με αυτή την επιλογή ο χρήστης μπορεί να επιλέξει εάν οι οδηγίες που του δίνονται στην οθόνη, θα του δίνονται και ηχητικά από το ηχείο της συσκευής. Με αυτόν τον τρόπο, μπορεί να ακούει τις οδηγίες χωρίς να χρειαστεί να εστιάσει στην οθόνη των γυαλιών.

## Ring Scanner

Το Ring Scanner είναι μία συσκευή ειδικά κατασκευασμένη για να σκανάρει barcode. Επειδή, αρκετές φορές είναι πιο πρακτικό να χρησιμοποιείται το χέρι από το μάτι κρίθηκε απαραίτητο να δώσουμε τη δυνατότητα στο χρήστη να μπορεί να συνδέσει με bluetooth ένα ring scanner και να μη χρησιμοποιεί μονάχα το Barcode scanner της κάμερας.



Σχήμα 7.15: Μία συσκευή ring scanner.

## Vuforia

Επειδή οι υπάλληλοι των αποθηκών είναι εξοικειωμένοι αρκετά με την αποθήκη και την τοποθεσία των προϊόντων σε αυτή, ο χάρτης των αποθηκών είναι περιττός για αυτούς. Επίσης, μιας και η Vuforia επεξεργάζεται κάθε frame για ανίχνευση marker μπορεί να αυξήσει την κατανάλωση μπαταρίας, οπότε αν δεν χρησιμοποιείται θα ήταν χρήσιμο να μη λειτουργεί. Έτσι, δίνουμε τη δυνατότητα στο χρήστη να επιλέξει αν θέλει αυτή τη δυνατότητα ή όχι.

## Voice Commands

Αυτή η επιλογή αφορά τη δυνατότητα να μπορεί ο χρήστης να επικοινωνήσει με την εφαρμογή και με φωνητικές εντολές πέρα από τα κουμπιά.

## Get και Post Request

Αυτά είναι τα δύο πεδία που απαιτούν κείμενο και αφορούν τη διεύθυνση που πρέπει να επικοινωνεί η εφαρμογή για την απόκτηση των λιστών συλλογής και την διεύθυνση στην οποία πρέπει να σταλθεί η ενημέρωση για την εξέλιξη της παραγγελίας.

### 7.3.8 Φωνητικές Εντολές

Στην προηγούμενη ενότητα αναλύσαμε τις ρυθμίσεις τις οποίες ο χρήστης μπορεί να επεξεργαστεί. Μία από αυτές αφορά τις φωνητικές εντολές (Voice Commands). Σκοπός της ενότητας αυτής είναι να μελετήσουμε τις εντολές αυτές.

Όταν κάποιος εργάζεται σε μία αποθήκη τα χέρια του είναι συνέχεια απασχολημένα, ή συλλέγοντας πακέτα ή σύροντας τα καρότσια που έχουν φορτώσει τα πακέτα. Εύκολα, γίνεται αντιληπτό ότι κάθε ενέργεια που απαιτεί ενασχόληση με τα χέρια καθυστερεί τους εργαζομένους στη διαδικασία συλλογής. Ένα βασικό πλεονέκτημα μίας καινοτόμας λύσης στο χώρο της συλλογής πακέτων θα ήταν η παροχή υποστήριξης ως προς τους εργαζομένους χωρίς να τους κρατά τα χέρια απασχολημένα. Αυτό το πλεονέκτημα έρχονται να το δώσουν οι φωνητικές εντολές.

Για την υλοποίηση των φωνητικών εντολών χρησιμοποιήσαμε μία βιβλιοθήκη η οποία παρέχεται από τη Vuzix. Όπως και στη λειτουργία σκαναρίσματος barcode, δημιουργήσαμε ένα Android Plugin και το ενσωματώσαμε στη Unity για να μπορούμε να εκμεταλευτούμε τις δυνατότητες της. Οι φωνητικές εντολές είναι διαθέσιμες σε όλη τη διάρκεια της εφαρμογής.

Ας δούμε τις φωνητικές εντολές που είναι διαθέσιμες στον χρήστη.

#### Exit Action

Με αυτή τη φωνητική εντολή ο χρήστης μπορεί να σταματήσει την τρέχουσα παραγγελία και να επιστρέψει στην αρχική οθόνη και να ξεκινήσει μία καινούργια.

#### Change Line

Όπως φαίνεται και από την ίδια την εντολή, ο χρήστης τώρα μπορεί να αλλάξει τρέχουσα γραμμή της παραγγελίας και από την οθόνη επιλογής να διαλέξει με ποιο πακέτο θέλει να ασχοληθεί.

#### Change Size

Όταν ο χρήστης βρίσκεται σε πακέτο που δεν απαιτεί σειριακό σκανάρισμα μπορεί να χρησιμοποιήσει αυτή την εντολή και να αλλάξει την ποσότητα την οποία θα συλλέξει.

#### Cancel Action

Τη συγκεκριμένη φωνητική εντολή μπορεί να τη χρησιμοποιήσει οποιαδήποτε στιγμή ο χρήστης για να επιστρέψει στην κανονική ροή της εφαρμογής. Για παράδειγμα, όταν αλλάζει την ποσότητα των πακέτων μπορεί να επιστρέψει στην οθόνη πληροφοριών.

## **Skip Scan**

Η συγκεκριμένη φωνητική εντολή είναι διαθέσιμη όταν η εφαρμογή βρίσκεται σε διαδικασία σκαναρίσματος. Με την διατύπωση της ο χρήστης μπορεί να σταματήσει τη διαδικασία σκαναρίσματος στην παρούσα φάση.



## Κεφάλαιο 8

### Σύνοψη και μελλοντικές επεκτάσεις

Η παρούσα διπλωματική εργασία επιχείρησε να προσεγγίσει τις ανάγκες που υπάρχουν στη διαδικασία του picking και να βελτιώσει την παραγωγικότητα των εργαζομένων χρησιμοποιώντας AR τεχνολογίες. Για να το πετύχουμε αυτό κληθήκαμε να επιλέξουμε ένα ζευγάρι smart glasses. Αυτά που ικανοποιούσαν τις ανάγκες μας και είναι τα πλέον διαδεδομένα για logistics είναι τα Vuzix M300. Πρόκειται για μία συσκευή Android, οπότε και η εφαρμογή μας στοχεύει το συγκεκριμένο λειτουργικό. Επειδή η Unity έχει γίνει ιδιαίτερα γνωστή τα τελευταία χρόνια, οι περισσότερες γνωστές βιβλιοθήκες επαυξημένης πραγματικότητας προσφέρουν τη δυνατότητα ενσωμάτωσης στο περιβάλλον της Unity. Η Vuforia όντας η μόνη διαθέσιμη βιβλιοθήκη επαυξημένης πραγματικότητας η οποία υποστήριζε τα M300 προσφέρει και αυτή δυνατότητα ενσωμάτωσης στη Unity. Λαμβάνοντας υπόψιν το Android Studio για να μπορούμε να χρησιμοποιήσουμε τις βιβλιοθήκες για barcodes και φωνητικές εντολές έχουμε όλα τα εργαλεία μας. Όσον αφορά την ανάπτυξη της εφαρμογής, μετά από συζητήσεις, έρευνα στο διαδίκτυο και επισκέψεις σε αποθήκες προσπαθήσαμε να καταλάβουμε όλες τις διαδικασίες και τις ανάγκες που λαμβάνουν χώρα ως τώρα στις αποθήκες. Έτσι αναπτύξαμε μία εφαρμογή φιλική προς το χρήστη την οποία μέσα από τις ρυθμίσεις μπορεί να τη φέρει στα μέτρα του σύμφωνα με τις ανάγκες του και να διευκολύνει την εργασία του στην αποθήκη.

Μελλοντικές επεκτάσεις θα μπορούσαν να περιλαμβάνουν την δημιουργία μίας οθόνης στην αρχή της εφαρμογής για να μπορεί ο χρήστης να κάνει login σκανάροντας κάποιο barcode, ώστε η εφαρμογή να είναι προσωποποιημένη. Ένα επόμενο βήμα θα ήταν η αναβάθμιση της εφαρμογής στο επόμενο μοντέλο γυαλιών ώστε να μπορεί να ενσωματωθεί και η δυνατότητα για την πλοήγηση εσωτερικού χώρου. Αυτή η δυνατότητα θα φαινόταν ιδιαίτερα χρήσιμη σε καινούργιους εργαζομένους οι οποίοι δεν είναι εξοικειωμένοι με την αποθήκη. Μία ακόμα δυνατότητα που θα φαινόταν χρήσιμη θα ήταν να ενσωματώσουμε και άλλες γλώσσες τόσο στη διαδικασία Text to Speech, που διαβάζονται οι οδηγίες στο χρήστη τόσο και στις φωνητικές εντολές που μπορεί να δώσει ο χρήστης.



## Βιβλιογραφία

- [ASta] “Πληροφορημένη αναζήτηση και εξερεύνηση”, [http://ai.uom.gr/aima/Samples/AIMA\\_chapter04.pdf](http://ai.uom.gr/aima/Samples/AIMA_chapter04.pdf).
- [Auks92] S. Roth και D. Blatner Aukstakalnis, *Silicon Mirage: The Art and Science of Virtual Reality*, 1992.
- [Azum97] Ronald T. Azuma, “A Survey of Augmented Reality”, *Presence: Teleoper. Virtual Environ.*, pp. 355–385, 1997.
- [Ch15] Sravan Ch, Shivanku Mahna and Nirbhay Kashyap, “Article: Optical Character Recognition on Handheld Devices”, *International Journal of Computer Applications*, vol. 115, no. 22, pp. 10–13, April 2015. Full text available.
- [DHLL] “AUGMENTED REALITY IN LOGISTICS”, [https://www.dhl.com/content/dam/downloads/g0/about\\_us/logistics\\_insights/csi\\_augmented\\_reality\\_report\\_290414.pdf](https://www.dhl.com/content/dam/downloads/g0/about_us/logistics_insights/csi_augmented_reality_report_290414.pdf).
- [Fein97] Steven Feiner, Blair Macintyre, Tobias Höllerer and Anthony Webster, “A Touring Machine: Prototyping 3D Mobile Augmented Reality Systems for Exploring the Urban Environment”, *Personal Technologies*, pp. 74–81, 1997.
- [Fire] “Firebase”, <https://firebase.google.com>.
- [hitl] “HITLab Projects: Shared Space. Human Interface Technology Laboratory”.
- [Inca] PTC Inc, “Natural Features and Image Ratings”, <https://library.vuforia.com/articles/Solution/Optimizing-Target-Detection-and-Tracking-Stability#Natural-Features-and-Image-Ratings>.
- [Incb] PTC Inc, “Object Recognition”, <https://library.vuforia.com/articles/Training/Object-Recognition>.
- [Incc] PTC Inc, “Text Recognition”, <https://library.vuforia.com/articles/Training/Text-Recognition-Guide>.
- [Kish94] P. Milgram και F. Kishino, “A Taxonomy of Mixed Reality Visual Displays”, *IEICE Transactions on Information and Systems*, 1994.
- [Pry098] Homer L. Pryor, III Thomas A. Furness and III Erik Viirre, “The Virtual Retinal Display: A new Display Technology using Scanned Laser Light”, *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, pp. 1570–1574, 1998.

- [Rask99] Ramesh Raskar, Greg Welch and Wei-Chao Chen, “Table-Top Spatially-Augmented Reality: Bringing Physical Models to Life with Projected Imagery”, pp. 64–, 1999.
- [SQLi] “SQLite in Unity”, <https://medium.com/@rizasif92/sqlite-and-unity-how-to-do-it-right-31991712190>.
- [Suth65] Ivan E. Sutherland, “The Ultimate Display”, *Proceedings of the IFIP Congress*, pp. 506–508, 1965.
- [Suth68] Ivan E. Sutherland, “A Head-Mounted Three-Dimensional Display”, pp. 757–764, 1968.
- [Vall] Jim Vallino, “Introduction to Augmented Reality”, <http://masters.donntu.org/2008/fvti/todoraki/library/7.html>.
- [Vufo] “Vuforia Fusion”, <https://library.vuforia.com/articles/Training/vuforia-fusion-article.html>.
- [Wiki] “Augemented Reality”, [http://en.wikipedia.org/wiki/Augmented\\_reality](http://en.wikipedia.org/wiki/Augmented_reality).