ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Ανάπτυξη Εφαρμογής Επαυξημένης Πραγματικότητας για τη Διαχείριση Πωλήσεων Προϊόντων

Τσιολάκης Γεώργιος Δαναός

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

Κολομβάτσος Κωνσταντίνος
Επίκουρος Καθηγητής

Λαμία ..... έτος 2024

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Ανάπτυξη Εφαρμογής Επαυξημένης Πραγματικότητας για τη Διαχείριση Πωλήσεων Προϊόντων

Τσιολάκης Γεώργιος Δαναός

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

Κολομβάτσος Κωνσταντίνος
Επίκουρος Καθηγητής

Λαμία ..... έτος 2024

UNIVERSITY OF THESSALY

SCHOOL OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE & TELECOMMUNICATIONS

Development of an Augmented Reality Application for Product Sales Management

Tsiolakis Georgios Danaos

FINAL THESIS

ADVISOR

Kolomvatsos Konstantinos
Assistant Professor

Lamia ..... year 2024

«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις [(1)], που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

*1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάσθηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.*

*2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.*

3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια

4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ημερομηνία: ……/…../2024

Ο Δηλών

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση
του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων
σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»

ΠΕΡΙΛΗΨΗ

Στόχος της πτυχιακής εργασίας είναι η εύρεση καινοτόμων μεθόδων προβολής πληροφοριών των προϊόντων χρησιμοποιώντας τεχνολογίες επαυξημένης πραγματικότητας, συμβάλλοντας στην διαδικασία αγοράς τους. Το σύστημα που αναπτύχθηκε αποτελείτε από την εφαρμογή κινητού η οποία προβάλει τοποθετημένες τις εικονικές ετικέτες πάνω στο QR Code του προϊόντος, και την ιστοσελίδα η οποία έχει αποθηκευμένες όλες τις εγγραφές που χρειάζεται η εφαρμογή. Με την εφαρμογή αυτού του συστήματος μπορούν να επωφεληθούν μαζί οι εργαζόμενοι και οι καταναλωτές: οι εργαζόμενοι μέσω των αποτελεσματικών εργαλείων διαχείρισης προϊόντων, και οι καταναλωτές μέσω των επιπρόσθετων πληροφοριών που προσφέρει και των ρυθμίσεων πρόσβασης της εφαρμογής. Η ανάπτυξη και η εφαρμογή αυτού του συστήματος είναι εφικτή με την αύξηση της πρόσβασης στην επαυξημένη πραγματικότητα χάρις στην τεχνολογική εξέλιξη, καθώς και στην ανάπτυξη ανοιχτής πρόσβασης βιβλιοθηκών και εργαλείων για την υλοποίηση των εφαρμογών. Η πρόοδος αυτής της τεχνολογικής εξέλιξης και ανάπτυξης λογισμικών αναλύεται, μαζί με τα διάφορα συστήματα επαυξημένης πραγματικότητας που δημιουργήθηκαν, κάποια από τα οποία έχουν παρόμοιο στόχο με το προτεινόμενο σύστημα.

ABSTRACT

This thesis aims to explore innovative methods of displaying product information by utilizing augmented reality technologies, contributing to the purchasing process. The developed system consists of a mobile application that displays the virtual labels placed on the QR Code of the product, and the website that has stored all the entries that the application requires. With the adaptation of this system in real world scenarios both the employees and consumers may benefit from it: Employees through the efficient product management tools, and consumers through the additional information it offers and the app's accessibility settings. The development and implementation of the system was made possible with the increased access in augmented reality result of the technological progress, while also with the release of open access libraries and software aimed at helping the application development process. The course of this technological evolution and software development is analyzed, along with the various augmented reality systems that have been created, some of which have a similar goal to the proposed system.

Table of Contents

ΠΕΡΙΛΗΨΗ I
ABSTRACT II

ΚΕΦΑΛΑΙΟ 1 ΕΙΣΑΓΩΓΗ 2

1.1 ΚΙΝΗΤΡΟ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ 2
1.2 ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΙΣΤΟΡΙΑ ΕΠΑΥΞΗΜΕΝΗΣ ΠΡΑΓΜΑΤΙΚΟΤΗΤΑΣ 3
1.2.1 ΙΣΤΟΡΙΑ 3
1.2.2 ΜΟΡΦΕΣ 6
1.2.3 ΕΦΑΡΜΟΓΕΣ 8
1.2.4 ΑΠΑΡΑΙΤΗΤΟΣ ΕΞΟΠΛΙΣΜΟΣ 10
1.3 ΔΥΣΚΟΛΙΕΣ 11

ΚΕΦΑΛΑΙΟ 2 ΤΕΧΝΟΛΟΓΙΕΣ ΑΝΑΠΤΥΞΗΣ ΕΦΑΡΜΟΓΩΝ ΕΠΑΥΞΗΜΕΝΗΣ ΚΑΙ ΕΙΚΟΝΙΚΗΣ ΠΡΑΓΜΑΤΙΚΟΤΗΤΑΣ 12

2.1 SOFTWARE DEVELOPMENT KIT ΤΗΣ GOOGLE 12
2.1.1 ΠΕΡΙΓΡΑΦΗ 12
2.1.2 ΤΕΧΝΟΛΟΓΙΕΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ 12
2.2 UNITY 14
2.2.1 ΠΕΡΙΓΡΑΦΗ 14
2.2.2 ΒΑΣΙΚΕΣ ΈΝΝΟΙΕΣ 14
2.2.3 ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΩΝ 14
2.3 AR FOUNDATION 15
2.3.1 ΠΕΡΙΓΡΑΦΗ 15
2.3.2 COMPONENTS 15

ΚΕΦΑΛΑΙΟ 3 ΣΥΣΤΗΜΑΤΑ ΕΠΑΥΞΗΜΕΝΗΣ ΠΡΑΓΜΑΤΙΚΟΤΗΤΑΣ 16

3.1 ΕΞΕΛΙΞΗ ΣΥΣΤΗΜΑΤΩΝ ΓΙΑ ΤΗΝ ΔΙΑΦΗΜΙΣΗ ΚΑΙ ΤΗΝ ΑΓΟΡΑ ΠΡΟΪΟΝΤΩΝ 16
3.2 ΣΥΣΤΗΜΑΤΑ ΑΓΟΡΑΣ ΠΡΟΪΟΝΤΩΝ 17
3.2.1 IKEA PLACE 17
3.2.2 IKEA KREATIV 18
3.3. ΣΥΣΤΗΜΑΤΑ ΔΙΑΦΗΜΙΣΗΣ ΠΡΟΪΟΝΤΩΝ 19
3.3.1 CATALOG-POWERED SHOPPING LENS ΤΟΥ SNAPCHAT 19
3.3.2 LEGO AR-STUDIO 19
3.3.3 NIKE 3D BILLBOARD 20

ΚΕΦΑΛΑΙΟ 4 ΠΡΟΤΕΙΝΟΜΕΝΟ ΣΥΣΤΗΜΑ 20

4.1 ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΡΟΪΟΝΤΩΝ 20
4.1.1 ΠΕΡΙΓΡΑΦΗ 20
4.1.2 ΤΕΧΝΟΛΟΓΙΕΣ 20
4.1.3 ΕΠΙΛΟΓΕΣ ΔΙΑΧΕΙΡΙΣΤΗ 21
4.1.4 SOURCE CODE SNIPPETS 21

4.2 ΕΦΑΡΜΟΓΗ ΠΡΟΒΟΛΗΣ ΠΡΟΪΟΝΤΩΝ 27
4.2.1 ΠΕΡΙΓΡΑΦΗ 27
4.2.2 ΤΕΧΝΟΛΟΓΙΕΣ 27
4.2.3 ΕΠΙΛΟΓΕΣ ΧΡΗΣΤΗ 28
4.2.4 SOURCE CODE SNIPPETS 29

ΚΕΦΑΛΑΙΟ 5 ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ 32

5.1 WORKFLOW 32
5.2 ΣΤΙΓΜΙΟΤΥΠΑ ΕΠΕΞΕΡΓΑΣΙΑΣ ΠΡΟΪΟΝΤΩΝ 34
5.3 ΣΤΙΓΜΙΟΤΥΠΑ ΕΚΤΕΛΕΣΗΣ ΕΦΑΡΜΟΓΗΣ 35

ΚΕΦΑΛΑΙΟ 6 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΠΡΟΕΚΤΑΣΕΙΣ 39

ΒΙΒΛΙΟΓΡΑΦΙΑ 40

ΚΕΦΑΛΑΙΟ 1 Εισαγωγή

1.1 Κίνητρο Πτυχιακής Εργασίας

Σκοπός της πτυχιακής εργασίας είναι η έρευνα καινοτόμων εναλλακτικών μέσων πώλησης προϊόντων, ενσωματώνοντας τεχνολογίες επαυξημένης πραγματικότητας με στόχο την διευκόλυνση του προσωπικού καθώς και των πελατών.

Διευκόλυνση των Εργαζομένων

Χάρις την ιστοσελίδα που έχει δημιουργηθεί για την διαχείριση των προϊόντων του καταστήματος, οι εργαζόμενοι έχουν την δυνατότητα άμεσης επεξεργασίας των πληροφοριών για κάθε προϊόν εξ αποστάσεως δίχως την φυσική παρέμβαση. Έτσι άτομα με κινητικές δυσκολίες έχουν την δυνατότητα να είναι εξίσου παραγωγικά με τους υπολοίπους.

Επιπλέον η ιστοσελίδα έχει κατασκευαστεί με σκοπό την εύκολη και απλή διαχείριση των δεδομένων, με στόχο την γρήγορη εξοικείωση χρήσης από τους εργαζόμενους καθώς και τις χαμηλές απαιτήσεις τεχνογνωσίας.

Τέλος υπάρχει η δυνατότητα επέκτασης των λειτουργιών και δυνατοτήτων της ιστοσελίδας, προσθέτοντας πεδίο αναζήτησης για γρήγορη εύρεση προϊόντων, επιλογές προσβασιμότητας όπως μεγαλύτερη γραμματοσειρά κειμένου, υψηλή αντίθεση χρωμάτων, μεγαλύτερα πλαίσια πεδίων και άλλα, καθώς και τρόπους πρόσβασης στην διαθεσιμότητα και τις πληροφορίες των προϊόντων από πιθανούς πελάτες μέσω του διαδικτύου.

Διευκόλυνση των Πελατών

Από την πλευρά των πελατών, με τις εικονικές ετικέτες που προβάλλονται μέσω της εφαρμογής επαυξημένης πραγματικότητας μπορούν να αναγνωρίσουν με ευκρίνεια τις πληροφορίες που χρειάζονται για την αγορά του προϊόντος, όπως την τιμή του. Αυτό είναι δυνατό με τα καθαρά και έντονα γράμματα σε λευκό φόντο, σε συνδυασμό με την γραμματοσειρά που χρησιμοποιείται στο κείμενο της ετικέτας. Επιπλέον οι χρήστες μπορούν να θέσουν το μέγεθος της γραμματοσειράς, διευκολύνοντας την ανάγνωση.

Υπάρχει δυνατότητα ακόμα της επιλογής του νομίσματος που χρησιμοποιείται και προβάλλεται στις ετικέτες, προσφέροντας μία αμεσότητα στην σύγκριση τιμών σε περίπτωση που οι χρήστες είναι συνηθισμένοι με ένα διαφορετικό νόμισμα, διευκολύνοντας ιδιαιτέρως επισκέπτες από άλλες χώρες ειδικά όταν συνδυάζεται με τις διάφορες γλώσσες που μπορούν να επιλεχθούν.

Με την εφαρμογή τρόπων πρόσβασης μέσα από το διαδίκτυο στην διαθεσιμότητα και τις πληροφορίες των προϊόντων των καταστημάτων, οι πελάτες έχουν την δυνατότητα σύγκρισης των τιμών με άλλα σημεία πώλησης, καθώς και αν θα βρουν τα προϊόντα που επιθυμούν στο κατάστημα.

1.2 Περιγραφή και Ιστορία Επαυξημένης Πραγματικότητας

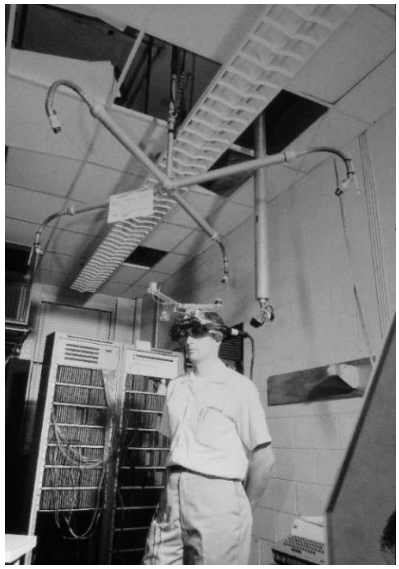
1.2.1 Ιστορία

Η επαυξημένη πραγματικότητα πρωτοεμφανίστηκε την δεκαετία του 1950 από τον κινηματογραφιστή Morton Heilig, όταν σκέφτηκε πως με την χρήση παραπάνω ερεθισμάτων, εκτός του οπτικού και του ακουστικού, μπορεί να επιτυγχάνει μία πιο ζωντανή εμπειρία θέασης. Το 1955 παρουσίασε σε δημοσίευση με τίτλο "El Cine Del Futuro" (Κινηματογράφος του Μέλλοντος) ένα πρωτότυπο σύστημα με όνομα Sensorama, το οποίο κατασκεύασε και πατένταρε το 1962.

Έπειτα, το 1968 ο Ivan Sutherland εφηύρε ένα σύστημα επαυξημένης πραγματικότητας για την προβολή τρισδιάστατων γεωμετρικών σχημάτων στον χώρο. Η συσκευή προβολής των γεωμετρικών σχημάτων προσαρμοζόταν συνδεδεμένη με καλώδια στο κεφάλι του/ης χρήστη για την αναγνώριση των κινήσεων του μεταβάλλοντας ταυτόχρονα την όψη των αντικειμένων.

Η οθόνη που χρησιμοποιήθηκε στην συσκευή ήταν παρόμοια με αυτές που χρησιμοποιούσαν σε πολεμικά ελικόπτερα οι χειριστές, επιτρέποντας τον/ην χρήστη να επικοινωνεί με το περιβάλλον λόγο της διαφάνειας της.

Εικόνα i. Sensorama – Παρακολούθηση ταινίας μικρού μήκους

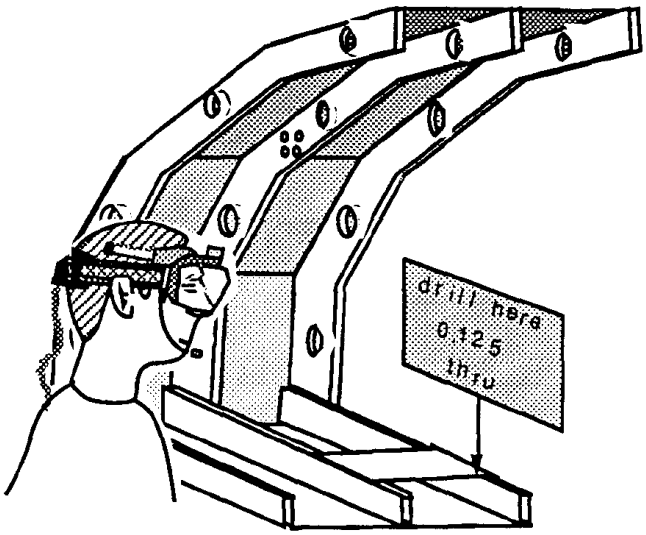


Εικόνα ii. Head-Mounted Display – Προβολή γεωμετρικών σχημάτων

Το 1975 ο Myron Krueger δημιουργεί το Videoplace, ένα εξοπλισμένο δωμάτιο που επιτρέπει για πρώτη φορά τους χρήστες να αλληλοεπιδράσουν με εικονικά αντικείμενα, τον ίδιο εξοπλισμό χρησιμοποίησε στην δημιουργία του Small Planet, μία εφαρμογή η οποία επιτρέπει στους χρήστες να «πετάξουν» σε έναν εικονικό τρισδιάστατο χώρο, με την κίνηση των χεριών τους.

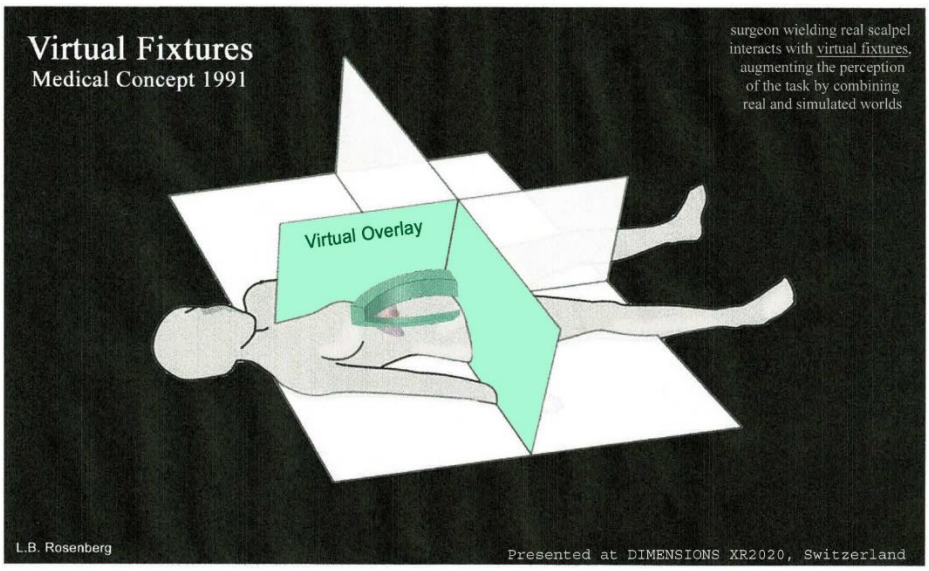
Εικόνα iii. Small Planet – Αλληλεπίδραση με τον χρήστη

Όμως η φράση Augmented Reality δεν είχε επινοηθεί μέχρι το 1992, όπου ο Thomas Caudell και ο David Mizell δημιούργησαν ένα AR σύστημα “Heads-up, see-through, head-mounted display (HUDSET)” [Thomas P. Caudell και David W. Mizell, 1992, σελ. 659], το οποίο βοηθούσε την κατασκευή αεροσκαφών της Boeing. Στην ίδια δημοσίευση όπου αναφέρεται το HUDSET, ανέδειξαν τα πλεονεκτήματα και τις απαιτήσεις του συστήματος που ανέπτυξαν, σε σχέση με την εικονική πραγματικότητα (Virtual Reality – VR), θέματα όπως την μειωμένη υπολογιστική ισχύ που χρειάζεται το AR, αφού σε σχέση με το VR, ο υπολογιστής δεν χρειάζεται να σχεδιάσει γραφικά για κάθε pixel της οθόνης, αλλά μόνο να προβάλει το μέγιστο 50 γραμμές στο οπτικό πεδίο του χρήστη. Επιπλέον κάποια AR συστήματα μπορεί να είναι πιο απαιτητικά από το VR, αφού το AR χρειάζεται μία ακριβή αντίληψη του περιβάλλοντος.



Εικόνα iv. AR HUDSET – Προβολή οδηγιών κατασκευής

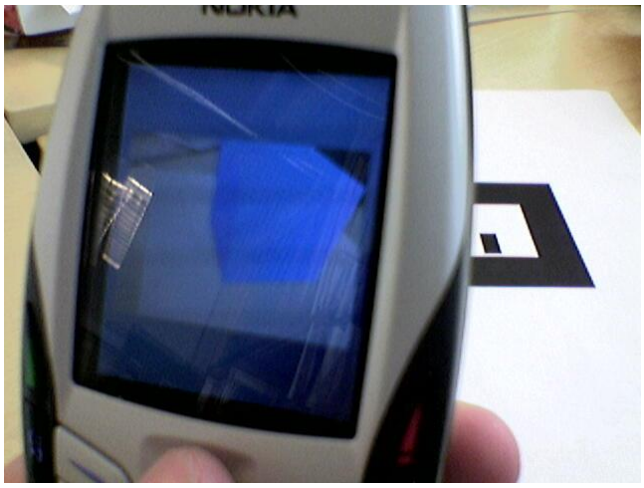
Το 1992 ο Louis Rosenberg αναπτύσσει ένα από τα πρώτα πλήρως λειτουργικά AR συστήματα, με ονομασία, Virtual Fixtures, για την καθοδήγηση κινήσεων του/ης χείριστή/τριας, προβάλλοντας οπτικούς οδηγούς και πληροφορίες. “Tools and fixtures in the real world (e.g., a ruler guiding a pencil) enhance human performance by guiding manual operations, providing localizing references, reducing mental workload, and increasing precision.” [Louis B. Rosenberg, 1992, σελ. 3]



Εικόνα v. Virtual Fixtures – Προβολή οδηγιών χειρουργικής τομής

Το 1999 δημιουργείται το ARToolKit από τον Hirokazu Kato του Nara Institute of Science and Technology, όπου και κυκλοφορεί από το Πανεπιστήμιο της Washington, HIT Lab. Το 2001 ιδρύεται η εταιρεία ARToolworks για την παροχή εμπορικών αδειών του ARToolKit, ενώ η έκδοση v1.0 του ανοιχτού κώδικα κοινοποιείται μέσω του HIT Lab. Το ARToolKit ήταν ένα από τα πρώτα Κιτς Ανάπτυξης Λογισμικού (Software Development Kits – SDK) επαυξημένης πραγματικότητας για κινητά τηλέφωνα.

Από τους πρώτους που αξιοποιούν την βιβλιοθήκη του ARToolKit είναι οι Anders Henrysson, Mark Billinghurst και Mark Ollila στο λειτουργικό σύστημα Symbian, το οποίο χρησιμοποιείται συνήθως σε κινητά Nokia. Έπειτα χρησιμοποιείται για πρώτη φορά σε λειτουργικό iOS το έτος 2008, και τέλος στο λειτουργικό Android το 2010.



Εικόνα vi. ARToolKit – Προβολή εικονικού κύβου

Το 2016 κυκλοφορεί το ARCore από την Google, ένα SDK για κινητές συσκευές σχεδιασμένο για την ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας. Το ARCore δεν ήταν το μόνο AR SDK της Google, πριν αυτό κυκλοφόρησε το Tango τον Ιούνιο του 2014. Το 2018 όμως σταματάει η υποστήριξη του Tango για να δοθεί περισσότερη προσοχή στην ανάπτυξη αυτού του νέου λογισμικού.

Ο λόγος της παύσης υποστήριξης του Tango ήταν οι τεχνολογικές απαιτήσεις. Τα κινητά δεν είχαν τους απαραίτητους αισθητήρες με αποτέλεσμα να υποστηριζόταν σε έναν μικρό αριθμό συσκευών, σε αντίθεση με το ARCore το οποίο υποστηρίζεται από πολλές Android συσκευές καθώς και iOS.

Το 2017 κυκλοφορεί το ARKit, λογισμικό της Apple με παρόμοιες λειτουργίες με το ARCore της Google, το οποίο επιτρέπει την δημιουργία AR εφαρμογών για iPhone και iPad συσκευές με προϋπόθεση η έκδοση iOS να είναι 11 και άνω.

1.2.2 Μορφές

Υπάρχουν πολλές μορφές επαυξημένης πραγματικότητας (Augmented Reality – AR) με την κάθε μία να έχει ξεχωριστές απαιτήσεις για την λειτουργία της, κάποιες από τις κύριες μορφές είναι:

1. Βασισμένη σε Δείκτες – Marker Based AR

Το Marker Based AR βασίζεται συνήθως σε Quick Response – QR Codes, εικόνες και αντικείμενα για τον εντοπισμό της επιφάνειας όπου θα προβληθεί το αντικείμενο ή οι πληροφορίες. Μπορεί να χρησιμοποιηθεί για την προβολή της αρχιτεκτονικής μηχανών, πληροφοριών/μοντέλων προϊόντων και αυτοκινήτων, ενημερωτικά βίντεο και άλλα.

2. Βασισμένη στην Τοποθεσία – Location Based AR

Το Location Based AR αξιοποιώντας την τοποθεσία και κατεύθυνση της συσκευής μπορεί να προβάλει οδηγίες για την άφιξη στον προορισμό του/ης χρήστη, βλέπουμε όμως την χρήση του και σε ηλεκτρονικά παιχνίδια, όπως το Pokémon GO, εφαρμογή επαυξημένης πραγματικότητας για Android και iOS κινητά η οποία λειτουργεί με βάσει την τοποθεσία του/ης παίκτη/τριας χρησιμοποιεί όμως και το superimposition το οποίο αναφέρεται παρακάτω.

3. Βασισμένη σε Φυσική Προβολή – Projection AR

Το Projection AR, αλλιώς Spatial Augmented Reality είναι η προβολή του περιεχομένου της επαυξημένης εφαρμογής πάνω στο φυσικό αντικείμενο, σε αντίθεση με τις υπόλοιπες μεθόδους οι οποίες απαιτούν ο/η χρήστης να έχει στην κατοχή του/ης μία οθόνη ή ένα head-mounted display ως το μέσο προβολής των πληροφοριών. Τα μειονεκτήματα του SAR είναι ότι για να λειτουργήσει αποτελεσματικά ο προτζέκτορας στην προβολή των πληροφοριών, η επιφάνεια είναι απαραίτητο να έχει τις κατάλληλες ιδιότητες για να μην αλλοιώνει ή να αντανακλά το φως που προβάλλεται πάνω της. Ένας ακόμα περιορισμός είναι η τοποθεσία που μπορεί να χρησιμοποιηθεί, σε εξωτερικούς χώρους κάτω από το φως του Ηλίου ένας κοινός projector δεν μπορεί να εκπέμπει όσο φως χρειάζεται για να είναι ευδιάκριτες οι πληροφορίες πάνω στην επιφάνεια του αντικειμένου.

4. Βασισμένη στην Υπέρθεση – Superimposition AR

Υπέρθεση είναι η προβολή ενός εικονικού αντικειμένου/πληροφοριών στον χώρο, κυρίως αξιοποιείται στα πλαίσια του ηλεκτρονικού εμπορίου. Κάποιες από τις εταιρείες που το έχουν χρησιμοποιήσει είναι το IKEA, Adidas, Nike, Farfetch, Prada, Piaget και άλλες. Το 2019 για παράδειγμα κυκλοφόρησε μία εφαρμογή για το κινητό από την Adidas η οποία επέτρεπε στους χρήστες να «δοκιμάσουν» διάφορα ζευγάρια υποδημάτων στρέφοντας την κάμερα του κινητού στο πέλμα τους.

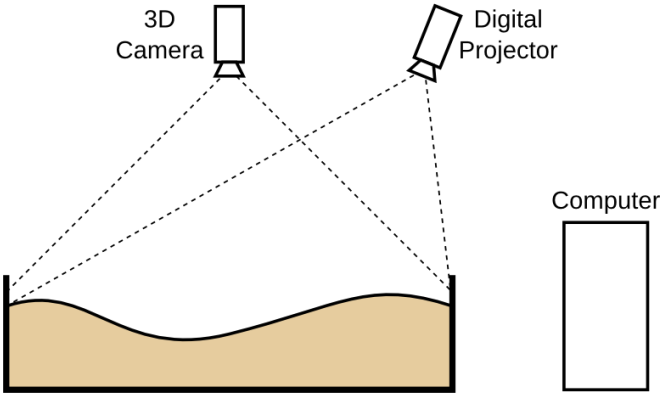
1.2.3 Εφαρμογές

Εφαρμογή Επαυξημένης Πραγματικότητας στην Εκπαίδευση:

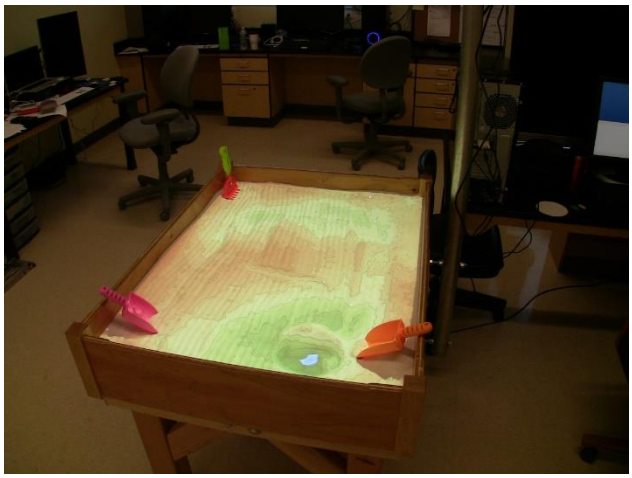
Augmented Reality Sandbox

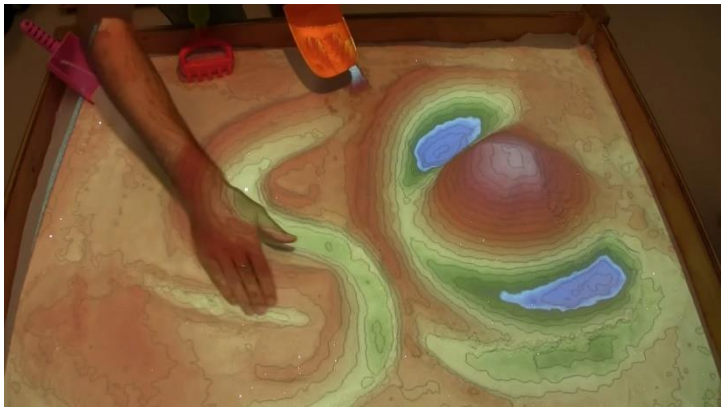
Το Augmented Reality Sandbox είναι ένα σύστημα από τους UC Davis και W.M Keck, το οποίο έχει σκοπό την εκμάθηση γεωγραφικών, γεωλογικών και υδρολογικών εννοιών μέσω του διαδραστικού τοπογραφικού χάρτη που προβάλλεται πάνω στην άμμο.

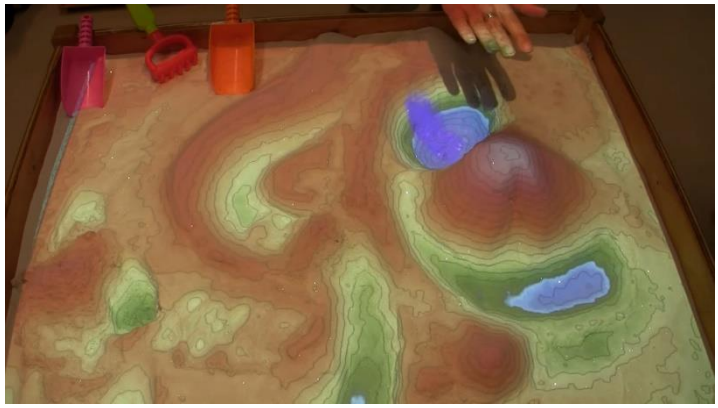
Αποτελείται από έναν υπολογιστή, μία Microsoft Kinect 3D κάμερα, έναν projector και ένα κουτί με άμμο. Ενώ για να λειτουργήσει η εφαρμογή ο projector με το Kinect τοποθετούνται πάνω από την λεκάνη με την άμμο.



Οι χρήστες κρατώντας το χέρι τους πάνω από την άμμο έχουν την δυνατότητα να προσθέσουν νερό και να δημιουργήσουν ποτάμια, λίμνες και θάλασσες, αφού η εφαρμογή επιτρέπει την προσομοίωση του. Καθώς μεταβάλλεται το ανάγλυφο της άμμου το χρώμα του χάρτη αλλάζει σε πραγματικό χρόνο, σχεδιάζοντας τις ισοϋψείς και το νερό αντιδρά ανάλογα.



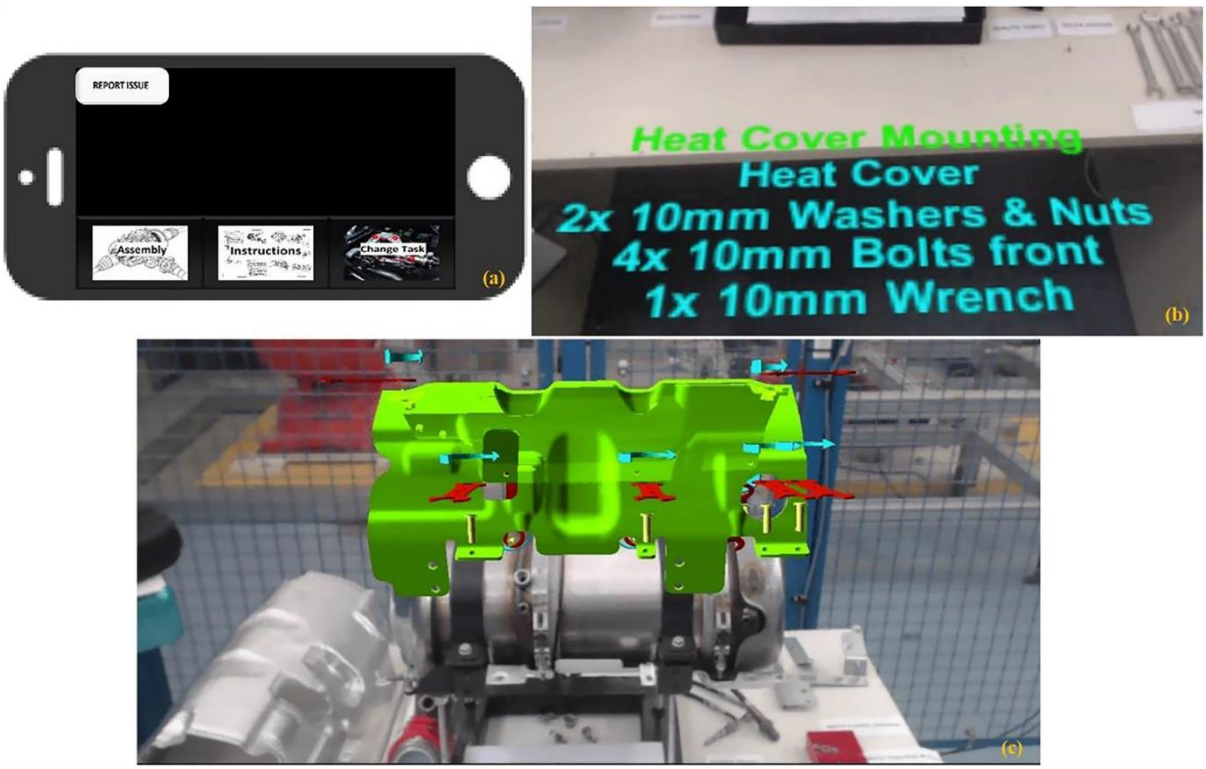




Εφαρμογή Επαυξημένης Πραγματικότητας στην Βιομηχανία:

Μία εφαρμογή που αναπτύχθηκε από τους Δημήτριος Μούρτζης, Βασίλειος Ζογόπουλος και Φωτεινή Ξάνθη με σκοπό την συναρμολόγηση προϊόντων σε γραμμή βιομηχανίας με μεταβαλλόμενα κομμάτια και βήματα συναρμολόγησης. Οι χρήστες χρησιμοποιούν αποκλειστικά το κινητό τους για την εφαρμογή, ή χρησιμοποιούν το head mounted display συνδεδεμένο με το κινητό το οποίο είναι σταθερά προσαρμοσμένο στον καρπό του χεριού τους.

Στόχος της εφαρμογής είναι η κάλυψη της ζήτησης προϊόντων τα οποία μπορεί να αλλάξουν με κάθε παραγγελία. Σε αυτές τις μεταβολές οι εργαζόμενοι μπορούν να προσαρμόζονται πιο αποτελεσματικά μέσω των οδηγιών όπου θα παραλαμβάνουν από τα συστήματα βάσεων δεδομένων, και θα προβάλλονται πάνω στην επιφάνεια εργασίας τους.

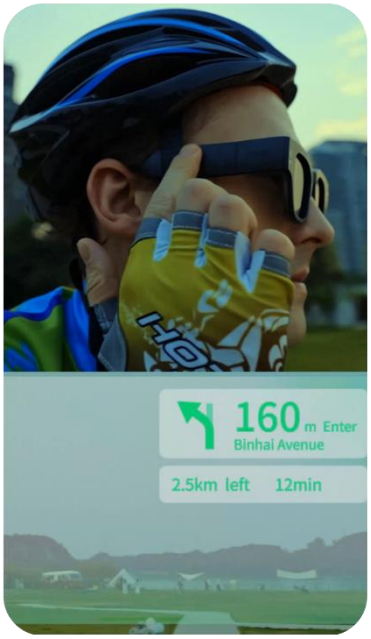


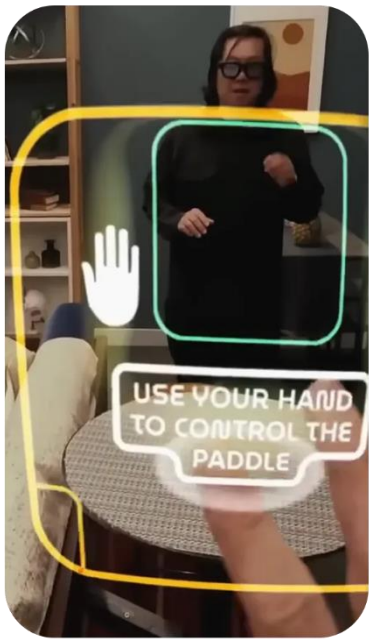
1.2.4 Απαραίτητος Εξοπλισμός

Είναι εμφανές σε αυτό το σημείο ότι για την υλοποίηση εφαρμογών που αξιοποιούν τις τεχνολογίες της επαυξημένης πραγματικότητας, σε ένα μεγάλο βαθμό, είναι απαραίτητη προϋπόθεση η χρήση κάμερας για την καταγραφή και αναγνώριση του περιβάλλοντος με την οθόνη να αποτελεί το μέσο προβολής των πληροφοριών που παράγει η εφαρμογή.

Με την ραγδαία τεχνολογική εξέλιξη των κινητών τηλεφώνων, και την παροχή βιβλιοθηκών και εργαλείων για την διευκόλυνση ανάπτυξης εφαρμογών επαυξημένης πραγματικότητας, παρατηρούμε μία τάση κυριαρχίας των κινητών τηλεφώνων ως την κύρια πλατφόρμα υλοποίησης εφαρμογών. Έτσι, σε συνδυασμό με το σχετικά χαμηλό κόστος των κινητών, ο αριθμός των ατόμων που έχουν πρόσβαση στις τεχνολογίες επαυξημένης πραγματικότητας καθώς και στις εμπειρίες που προσφέρει, είναι αρκετά αυξημένος σε σχέση με παλαιότερα.

Για εμπειρίες πιο συναρπάστηκες και ζωντανές όμως μπορούν να χρησιμοποιηθούν γυαλιά επαυξημένης πραγματικότητας, συσκευές οι οποίες έχουν αισθητήρες και διακριτικές οθόνες ενσωματωμένες στους φακούς συγκεκριμένα για αυτό τον σκοπό. Έχουν δημιουργηθεί διαφορετικών ειδών συσκευές, από γυαλιά που έχουν σχεδιαστεί αποκλειστικά για την προβολή της οθόνης του κινητού με σύνδεση καλωδίου ως μέσο ψυχαγωγίας (π.χ. G330 Smart Glasses – Xrany Yuan Ni), σε μοντέλα γυαλιών με ενσωματωμένο επεξεργαστή, λειτουργικό σύστημα συνήθως Android και επιφάνεια αφής στον σκελετό της συσκευής για την πλοήγηση των χρηστών στις εφαρμογές που προσφέρει (π.χ. INMO Air – INMO), έως και προηγμένα μοντέλα τα οποία ανιχνεύουν τις κινήσεις των χεριών (π.χ. Orion – Meta).





1.3 Δυσκολίες

Στα πλαίσια της υλοποίησης της εφαρμογής στο χώρο ενός καταστήματος

- Προϋπόθεση είναι πως οι πελάτες έχουν πρόσβαση σε συσκευές που υποστηρίζουν τις τεχνολογίες της επαυξημένης πραγματικότητας

- Υπάρχει το ενδεχόμενο της δυσκολίας χρήσης της εφαρμογής από ηλικιωμένα άτομα, λόγο της ελλιπής εμπειρίας της τεχνολογίας των σύγχρονων κινητών τηλεφώνων

- Τα μικρά καταστήματα μπορεί να μην επωφελούνται, αφού το κόστος της εφαρμογής αυτού του μέσου ενδέχεται να είναι υψηλότερο από τις μη εικονικές μεθόδους, λόγο της πρόσληψης των τεχνικών για την εγκατάσταση του συστήματος και την λειτουργία του διακομιστή

Στα πλαίσια της εκπόνησης της πτυχιακής

Αδυναμία αγοράς γυαλιών επαυξημένης πραγματικότητας εξοπλισμένο με τις απαραίτητες δυνατότητες για την λειτουργία της εφαρμογής σε αυτά, λόγω υψηλού κόστους των συσκευών οι οποίες επιτρέπουν την ανάπτυξη εφαρμογών σε αυτές, με αποτέλεσμα την υλοποίηση της εφαρμογής σχεδιασμένη για κινητά τηλέφωνα.

Για την αποθήκευση, διαχείριση, επεξεργασία και ανάκτηση των πληροφοριών των προϊόντων θα χρειαστεί μία διεύθυνση του διαδικτύου η οποία θα εμπεριέχει τις πληροφορίες και την διεπαφή του διαχειριστή, έναν διακομιστή δηλαδή. Υπάρχουν υπηρεσίες στο διαδίκτυο οι οποίες προσφέρουν διακομιστές δίχως χρέωση αλλά με περιορισμούς, το οποίο είχε ως αποτέλεσμα την υλοποίηση της επικοινωνίας μεταξύ εφαρμογή και διακομιστή με έναν μη βέλτιστο τρόπο. Ο εναλλακτικός και πιο αποδοτικός τρόπος περιγράφεται στο κεφάλαιο 5.

ΚΕΦΑΛΑΙΟ 2 Τεχνολογίες Ανάπτυξης Εφαρμογών Επαυξημένης και Εικονικής Πραγματικότητας

2.1 Software Development Kit της Google

2.1.1 Περιγραφή

Το ARCore ή αλλιώς Google Play Services for AR είναι το SDK της Google με το οποίο μας επιτρέπει να έχουμε την δυνατότητα ανάπτυξης εφαρμογών επαυξημένης πραγματικότητας, οι οποίες προορίζονται για κινητά τηλέφωνα.

Σχεδιάστηκε για Android και iOS συσκευές, και υποστηρίζεται σε ένα μεγάλο αριθμό μοντέλων συσκευών που αναφέρονται στην ιστοσελίδα https://developers.google.com/ar/devices. Χρησιμοποιώντας αυτά τα δεδομένα και λαμβάνοντας υπόψη το μερίδιο της αγοράς (market share) των συσκευών μπορούν οι υπεύθυνοι πωλήσεων να γνωρίζουν αν αξίζει να αναπτύξουν εφαρμογές αξιοποιώντας το SDK της Google, ή αν θα ήταν καλύτερο να χρησιμοποιήσουν το αντίστοιχο API της Apple με ονομασία ARKit.

2.1.2 Τεχνολογίες που Χρησιμοποιούνται

1. Motion Tracking

Καθώς το κινητό μεταφέρεται στον χώρο, το ARCore χρησιμοποιεί το simultaneous localization and mapping ή SLAM για την αναγνώριση θέσης και φοράς της συσκευής.

Το ARCore ακόμα αναγνωρίζει με την κάμερα σημεία ενδιαφέροντος (feature points) στο περιβάλλον και τα χρησιμοποιεί για τον υπολογισμό της μεταβολής της τοποθεσίας.

Οι πληροφορίες της κάμερας συνδυάζονται με τις μετρήσεις του IMU της συσκευής (Μονάδα Μέτρησης Αδράνειας) για να γίνει η εκτίμηση της στάσης (θέση και προσανατολισμός) της κάμερας σε σχέση με το περιβάλλον κατά το χρόνο εκτέλεσης.

Έτσι, καθώς ο/η χρήστης κινείται στο χώρο, το αντικείμενο παραμένει στην θέση που του έχει οριστεί με αποτέλεσμα, ό,τι εικονικό αντικείμενο προβληθεί στο περιβάλλον να δίνει τη ψευδαίσθηση ότι είναι μέρος του.

2. Environmental Understanding

Για την αναγνώριση επίπεδων επιφανειών στον χώρο, όπως τραπέζια και τοίχοι, ώστε να είναι διαθέσιμες για χρήση από την εφαρμογή ως γεωμετρικά επίπεδα "geometric planes" για την τοποθέτηση αντικειμένων πάνω στις επιφάνειες.

Αναγνωρίζει αυτές τις επίπεδες επιφάνειες εντοπίζοντας ομάδες σημείων ενδιαφέροντος τα οποία φαίνονται να είναι στην ίδια ευθεία. Μεγάλες επιφάνειες που δεν έχουν σημεία ενδιαφέροντος όπως ένας λευκός τοίχος μπορεί να μην ανιχνευτεί με ακρίβεια.

3. Depth understanding

Με την χρήση RGB-D σε συσκευές που το υποστηρίζουν υπάρχει δυνατότητα αναγνώρισης βάθους στον χώρο με αποτέλεσμα εικονικά αντικείμενα να αλληλοεπιδρούν και να συγκρούονται με ακρίβεια με το περιβάλλον. Επίσης μπορούν να προβάλλονται μπροστά ή πίσω από φυσικά αντικείμενα.

4. Light Estimation

Αντλώντας πληροφορίες για το φωτισμό του χώρου επιτρέπει να αποκτούν τα εικονικά αντικείμενα "φυσικές" φωτοσκιάσεις.

5. User Interaction

Χρησιμοποιώντας το hit-testing για την αναγνώριση τον συντεταγμένων του περιβάλλοντος αφού επιλέξει ένα σημείο της οθόνης ο/η χρήστης, εκπέμπεται μία ακτίνα στο περιβάλλον από την κάμερα επιστρέφοντας τα γεωμετρικά επίπεδα ή τα σημεία ενδιαφέροντος που τέμνει η ακτίνα, καθώς και τον προσανατολισμό των αντικειμένων την διεύθυνση της τομής των επιπέδων.

Έτσι ο/η χρήστης έχει την δυνατότητα να επιλέγει και να αλληλοεπιδρά με αντικείμενα στον χώρο.

6. Oriented Points

Η φορά των σημείων επιτρέπει την τοποθέτηση αντικειμένων σε κεκλιμένες επιφάνειες. Όταν εκτελείται το hit-testing το οποίο επιστρέφει ένα σημείο ενδιαφέροντος, το ARCore κοιτάει κοντινά σημεία ενδιαφέροντος και χρησιμοποιεί την θέση τους για να υπολογίσει την κλίση της επιφάνειας. Έπειτα το ARCore επιστρέφει μία στάση εκλαμβάνοντας την κλίση της επιφάνειας.

7. Anchors and Trackables

Οι στάσεις μπορούν να αλλάξουν καθώς το ARCore βελτιώνει την αντίληψη της θέσης του, καθώς και του περιβάλλοντος. Όταν ο/η χρήστης επιθυμεί να τοποθετήσει ένα εικονικό αντικείμενο, πρέπει να θέσει και ένα anchor ώστε το ARCore να παρακολουθεί την θέση του αντικειμένου στον χρόνο εκτέλεσης. Η δημιουργία του anchor πραγματοποιείται συνήθως βάσει της στάσης που επέστρεψε το hit test.

Το γεγονός πως οι στάσεις μπορούν να αλλάξουν με την πάροδο του χρόνου δείχνει ότι το ARCore μπορεί να ανανεώνει τις θέσεις των αντικειμένων του περιβάλλοντος, όπως γεωμετρικές επιφάνειες και σημεία ενδιαφέροντος. Οι επιφάνειες και τα σημεία είναι ειδικού τύπου αντικείμενα που αποκαλούνται trackables. Αυτά είναι αντικείμενα τα οποία τα παρακολουθεί το ARCore με την πάροδο του χρόνου. Έτσι μπορούν να τοποθετηθούν εικονικά αντικείμενα με anchors πάνω στα trackables για να εξασφαλιστεί πως η σχέση μεταξύ τους θα παραμείνει σταθερή ακόμα και καθώς κινείται η συσκευή. Για παράδειγμα αν τοποθετηθεί μία εικονική φιγούρα πάνω σε ένα γραφείο και το ARCore αργότερα αλλάξει την στάση του γεωμετρικού επιπέδου που σχετίζεται με το γραφείο, η φιγούρα θα συνεχίσει να φαίνεται πως είναι τοποθετημένη πάνω στο γραφείο.

8. Augmented Images

Προσφέρει την δυνατότητα οι εφαρμογές να ενεργούν μόλις αναγνωρίσουν συγκεκριμένες εικόνες. Για παράδειγμα μόλις ο χρήστης στρέψει την κάμερα σε μία αφίσα ταινίας, ένας χαρακτήρας της μπορεί να ξεπροβάλει από αυτήν και να πάρει μέρος σε ένα σκηνικό της ταινίας, πάνω στην αφίσα της αλλά και πάνω σε διαφημιστικό λεωφορείου εν κινήσει.

2.2 Unity

2.2.1 Περιγραφή

Το Unity είναι ένα πλαίσιο εργασίας (framework) το οποίο δίνει την δυνατότητα σε προγραμματιστές/τριες να δημιουργήσουν εφαρμογές και παιχνίδια για μία πληθώρα πλατφόρμων (Windows, Linux, macOS, Android, iOS, WebGL κ.α.) που κυκλοφόρησε για πρώτη φορά το 2005.

Το λογισμικό του Unity έχει γραφτεί στην γλώσσα C++ έτσι ώστε να έχει την απαραίτητη αποδοτικότητα για την ομαλή και γρήγορη προβολή των κινήσεων των αντικειμένων καθώς και την προσομοίωση φυσικών δυνάμεων, για παράδειγμα την βαρύτητα. Από την μεριά του χρήστη του Unity η γλώσσα προγραμματισμού που χρησιμοποιείται είναι η C#, αλλά υποστηρίζεται και το framework .NET της Microsoft.

2.2.2 Βασικές Έννοιες

Scene: Το περιβάλλον στο οποίο χρησιμοποιούμε τα αντικείμενα που μας προσφέρει το Unity (Components, 3D Μοντέλα, Χαρακτήρες, Διεπαφή Χρήστη κ.α.)

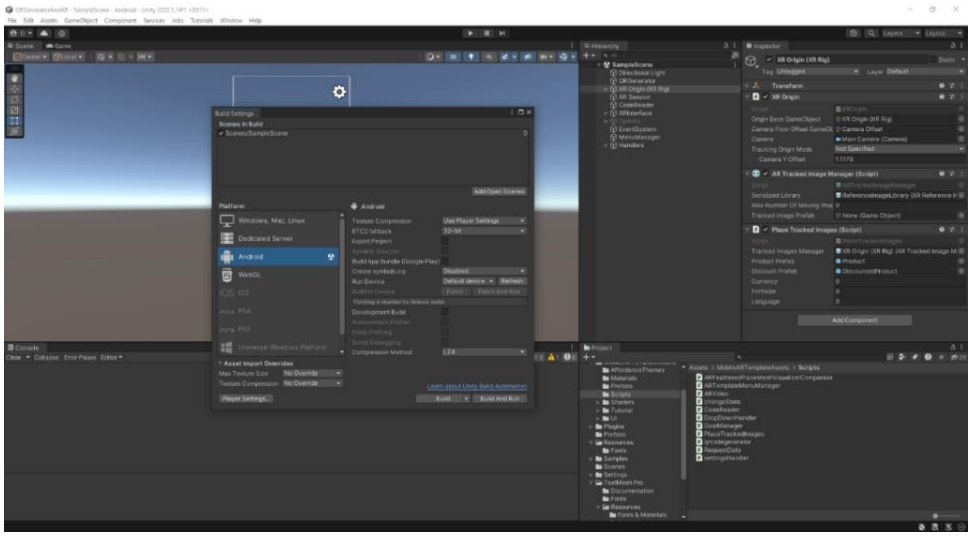
GameObject: Το GameObject είναι η κύρια κλάση όλων τον οντοτήτων στο Scene. Τα GameObjects μπορούν να οριστούν ως τα δομικά στοιχεία για κάθε scene της εφαρμογής μας, και λειτουργούν ως containers για τα components του Unity.

Components: Είναι η κύρια κλάση για οτιδήποτε είναι συνδεδεμένο σε ένα GameObject. Υπάρχει μία πληθώρα από components στο Unity, το κάθε ένα με ξεχωριστές λειτουργίες και ιδιότητες.

2.2.3 Δημιουργία Εφαρμογών

Η διαδικασία μετατροπής της εφαρμογής που λειτουργεί μέσα στο περιβάλλον του Unity, σε ένα ανεξάρτητο εκτελέσιμο αρχείο είναι αρκετά απλή και γρήγορη. Αφού έχουν τοποθετηθεί όλα τα απαραίτητα components, έχουν δημιουργηθεί τα αρχεία που εμπεριέχουν τον κώδικα για την επικοινωνία και την λειτουργικότητα των GameObjects στο scene, ο/η προγραμματιστής/στρια επιλέγει τα απαραίτητα XR plugins της πλατφόρμας (Linux, Windows, Mac, WebGL) που προορίζεται η εφαρμογή, καθώς και το SDK που θα χρησιμοποιεί η συσκευή, το Google ARCore για παράδειγμα αν προορίζεται για Android κινητά.

Έπειτα μπορούν να οριστούν τα πεδία στις ρυθμίσεις όπως το όνομα της εφαρμογής και του πακέτου (com.example.myapplication), τον προσανατολισμό που θα χρησιμοποιεί η εφαρμογή (οριζόντια ή κάθετα), το εικονίδιο της, την τεχνολογία διεπαφής για τα γραφικά (OpenGLES3 ή Vulkan για παράδειγμα) και άλλα. Τέλος αφού έχουν οριστεί όλα τα επιθυμητά πεδία γίνεται οι επιλογή των scene που θα χρησιμοποιηθούν στην εφαρμογή και δημιουργείται το εκτελέσιμο αρχείο.



2.3 AR Foundation

2.3.1 Περιγραφή

Το AR Foundation επιτρέπει την δημιουργία εφαρμογών επαυξημένης πραγματικότητας με το Unity για διάφορες συσκευές. Στο project που έχει δημιουργηθεί, μπορούν να οριστούν ποιες θα είναι οι λειτουργίες AR που θα χρησιμοποιηθούν, ενεργοποιώντας τες με την προσθήκη των αντίστοιχων manager components στο scene.

2.3.2 Components

AR Session

Το AR Session είναι το κύριο τμήμα μίας AR εφαρμογής στο Unity, ο ρόλος του είναι να ελέγχει τον κύκλο ζωής των λειτουργιών επαυξημένης πραγματικότητας ενεργοποιώντας και απενεργοποιώντας τες.

Το Session αναφέρεται σε ένα instance του AR. Οι λειτουργίες, όπως η αναγνώριση επίπεδων επιφανειών, μπορούν να ενεργοποιηθούν και να απενεργοποιηθούν ανεξάρτητα. Το session ελέγχει όλα τα AR χαρακτηριστικά / σημεία ενδιαφέροντος, έτσι αν απενεργοποιηθεί το AR Session, αντίστοιχα η εφαρμογή παύει να τα παρακολουθεί στο περιβάλλον. Σε περίπτωση που ενεργοποιηθεί αργότερα, η συσκευή θα προσπαθήσει να ανακαλέσει και να διατηρήσει χαρακτηριστικά / σημεία ενδιαφέροντος που είχε εντοπίσει.

Managers

Στο AR Foundation, ο manager είναι ένα component που προσφέρει την διεπαφή για τον προγραμματισμό AR λειτουργιών της εφαρμογής. Για παράδειγμα το Plane Manager ενεργοποιεί την λειτουργία του εντοπισμού επίπεδων επιφανειών στην εφαρμογή και επιτρέπει την δυνατότητα αλληλεπίδρασης αυτών των εντοπισμένων επιφανειών μέσω του κώδικα εφαρμογής.

Πολλές από αυτές τις AR τεχνολογίες που προσφέρει το AR Foundation, για να λειτουργήσουν χρησιμοποιούν υποσυστήματα τα οποία προσφέρουν ένα layer of abstraction μεταξύ του κώδικα της εφαρμογής και του SDK της επιλεγμένης πλατφόρμας, όπως το ARCore της Google για Android και ARKit της Apple για iOS, αφού ο τρόπος υλοποίησης κάθε AR λειτουργίας δεν είναι ο ίδιος για κάθε πλατφόρμα. Το πλεονέκτημα του layer of abstraction είναι πως ο κώδικας παραμένει ο ίδιος ανεξαρτήτως την πλατφόρμα.

ΚΕΦΑΛΑΙΟ 3 Συστήματα Επαυξημένης Πραγματικότητας

3.1 Εξέλιξη Συστημάτων για την Διαφήμιση και την Αγορά Προϊόντων

Όταν οι τεχνολογίες επαυξημένης πραγματικότητας εφαρμόστηκαν αρχικά στην αγορά με σκοπό την διαφήμιση προϊόντων, οι μέθοδοι που είχαν οι επιχειρήσεις για την προβολή των προϊόντων τους μέσω της επαυξημένης πραγματικότητας ήταν αρκετά περιορισμένες, καθώς προϋπόθεση για να λειτουργήσουν ήταν να διαθέτουν και οι καταναλωτές τον απαραίτητο εξοπλισμό.

Μία από τις εταιρείες που εκμεταλλεύτηκαν αυτή την νέα -για την εποχή της- τεχνολογία ήταν η αυτοκινητιστική εταιρία BMW το 2008. Με την προσθήκη ενός QR code σε διάφορα αυτοκινητιστικά περιοδικά οι αναγνώστες μπορούσαν να μεταβούν στην ιστοσελίδα της εταιρείας και να στρέψουν την σελίδα με το QR code προς την κάμερα τους. Αυτό είχε ως αποτέλεσμα να προβληθεί στην οθόνη τους ένα τρισδιάστατο μοντέλο του αυτοκινήτου της εταιρίας, το οποίο ήταν τοποθετημένο στην επιφάνεια της σελίδας του περιοδικού.

Με την συνεχή αύξηση του αριθμού των ατόμων που έχουν πρόσβαση σε συσκευές που υποστηρίζουν τις τεχνολογίες επαυξημένης πραγματικότητας, αυξάνονται τα συστήματα και οι εφαρμογές διαφήμισης και διευκόλυνσης των αγορών προϊόντων από τους καταναλωτές. Εφαρμογές όπως το Snapchat δίνουν την δυνατότητα σε μάρκες ρούχων, εταιρείες υποδημάτων, οπτικών, επίπλων κ.α. να διαφημίσουν τα προϊόντα τους αξιοποιώντας τις δυνατότητες που παρέχουν οι εφαρμογές, αναδεικνύοντας το μειωμένο κόστος διαφήμισης που μπορούν να επιτύχουν οι επιχειρήσεις και την αυξημένη επιστροφή της επένδυσης (Return Of Investment) σε σχέση με τυπικές μεθόδους διαφήμισης.

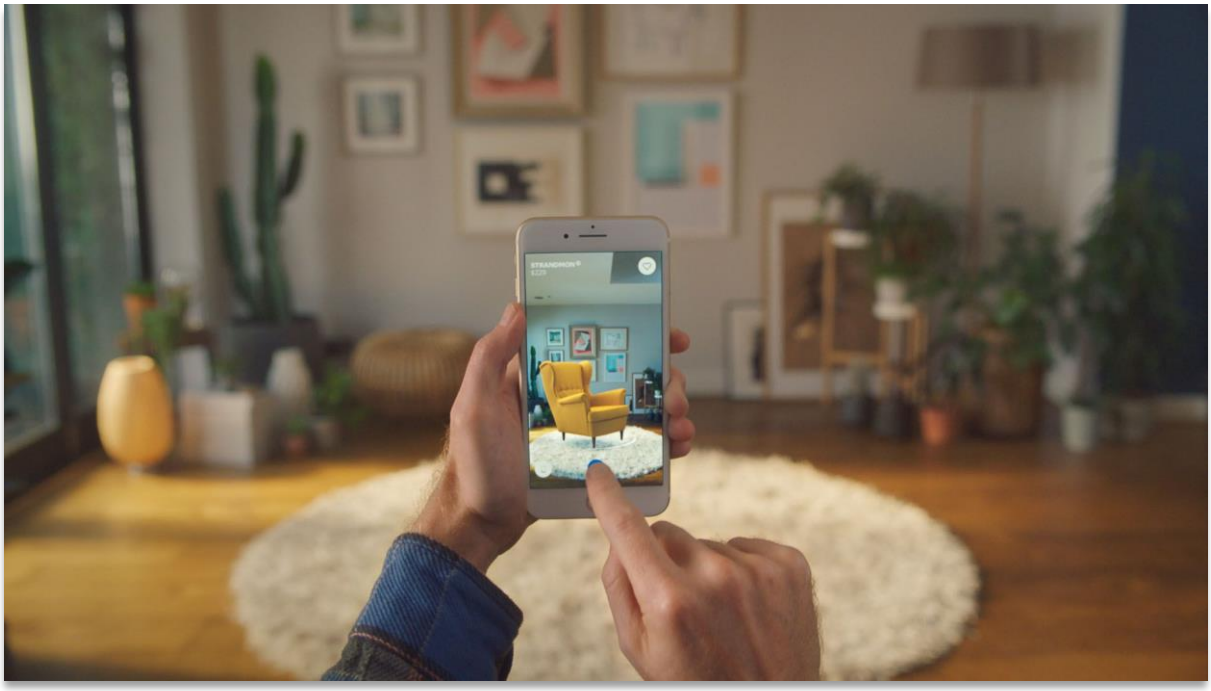
3.2 Συστήματα Αγοράς Προϊόντων

3.2.1 IKEA Place

Το IKEA Place είναι μία εφαρμογή επαυξημένης πραγματικότητας του IKEA για iOS κινητά με στόχο την προβολή τρισδιάστατων προϊόντων στον χώρο σε πραγματικό μέγεθος.

Ο χρήστης αφού στρέψει την κάμερα της συσκευής προς το δάπεδο, η εφαρμογή αναγνωρίζει την διάταξη του δωματίου και τον προσανατολισμό της. Στην συνέχεια ο χρήστης έχει την δυνατότητα να επιλέξει ένα αντικείμενο από την λίστα προϊόντων του IKEA και να το τοποθετήσει στο σημείο του χώρου που επιθυμεί. Χρησιμοποιώντας τις διαστάσεις του δωματίου τα προϊόντα που τοποθετούνται έχουν 0.98:1 αναλογία ή 98% ακρίβεια με το πραγματικό μέγεθος του αντικειμένου.

Έτσι υπάρχει διευκόλυνση στις αποφάσεις των πελατών για τα έπιπλα που ταιριάζουν στον χώρο, διότι προσφέρει έναν άμεσο και ακριβές τρόπο προβολής των επίπλων.

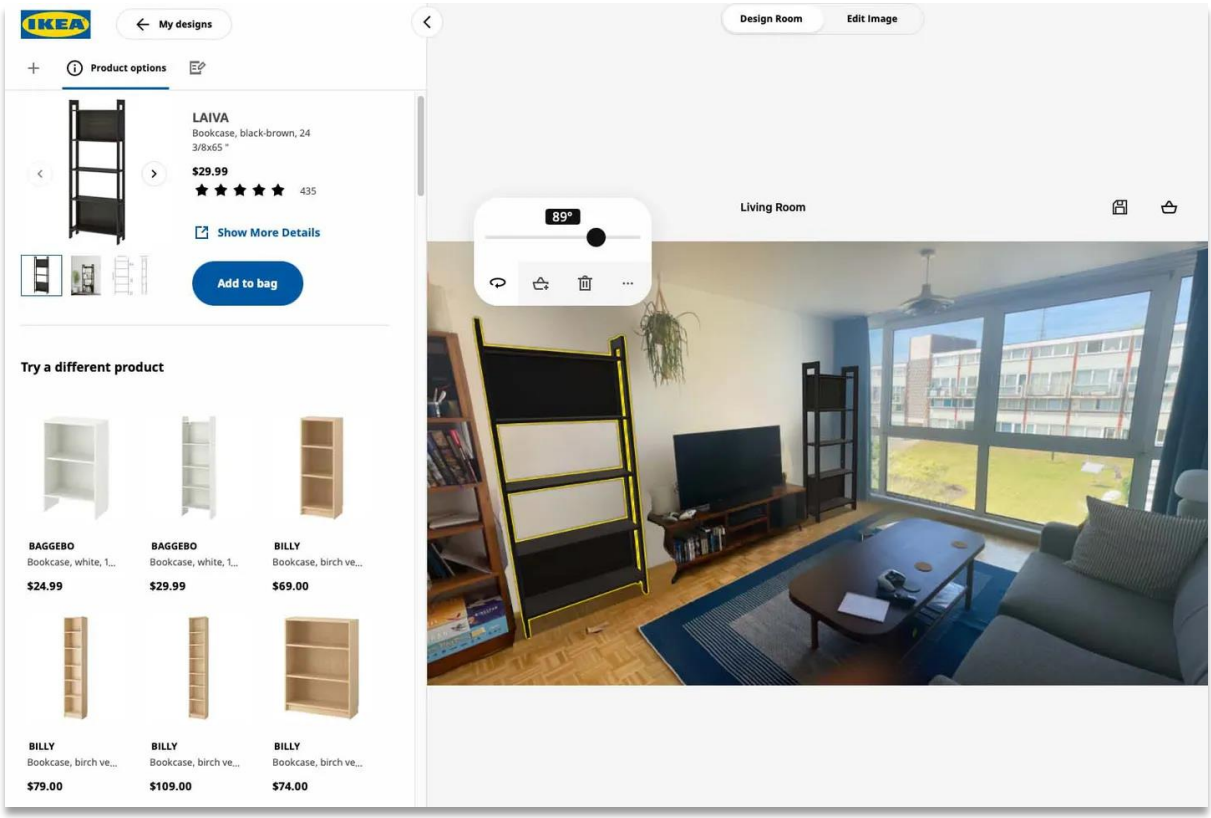


3.2.2 IKEA Kreativ

Το IKEA Kreativ πηγαίνει ένα επίπεδο παραπάνω με την βοήθεια της τεχνητής νοημοσύνης σε σχέση με το IKEA Place, αφού ο σκοπός του είναι η ολοκληρωμένη επίπλωση ενός δωματίου, έναντι της τοποθέτησης ενός μοναδικού αντικειμένου στο δωμάτιο.

Ο χρήστης αφού βγάλει μία σειρά φωτογραφιών του δωματίου, δημιουργείται μία ρέπλικα του του χώρου με την οποία ο χρήστης μπορεί να αφαιρέσει έπιπλα και αντικείμενα, αλλά και να προσθέσει έπιπλα από τον κατάλογο του IKEA.

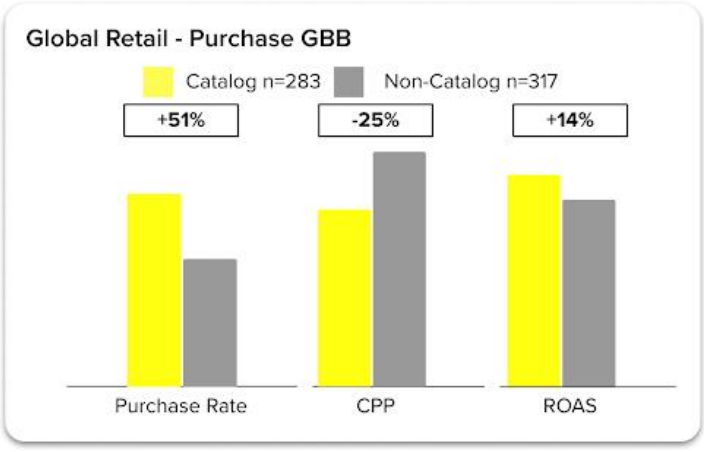
Αυτό επιτρέπει τον σχεδιασμό του χώρου σε εικονικό περιβάλλον διευκολύνοντας τους πελάτες να αποφασίζουν για τις αγορές τους με μεγαλύτερη βεβαιότητα.



3.3. Συστήματα Διαφήμισης Προϊόντων

3.3.1 Catalog-Powered Shopping Lens του Snapchat

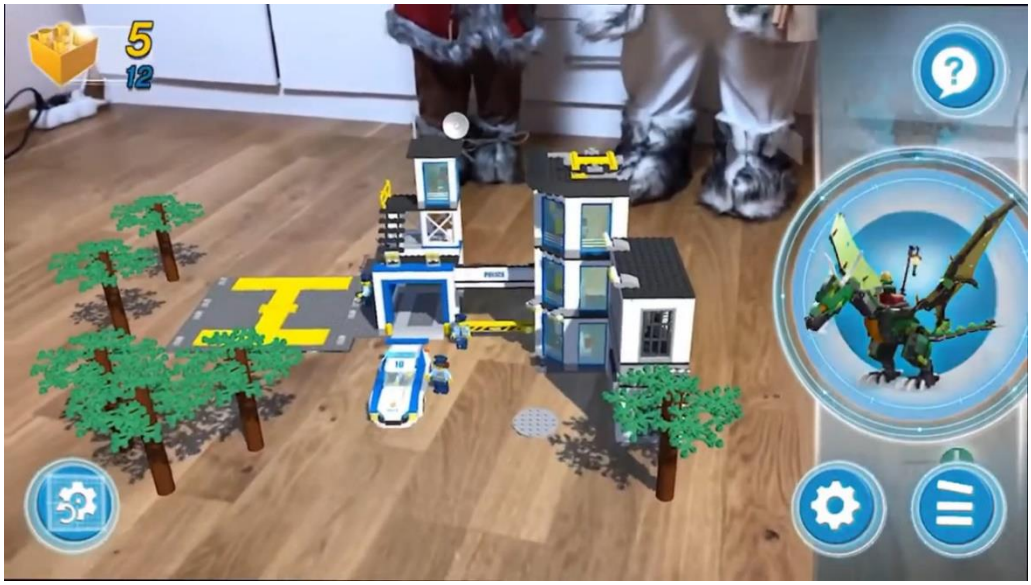
Με την εισαγωγή του “Catalog-Powered Shopping Lens” του Snapchat τον Ιανουάριο του 2022, εταιρείες και πωλητές/τριες έχουν την δυνατότητα προβολής των προϊόντων τους αξιοποιώντας τις δυνατότητες επαυξημένης πραγματικότητας του Snapchat. Ο αριθμός των μηνιαίων χρηστών της εφαρμογής καθώς και το χαμηλό κόστος διαφήμισής των προϊόντων στην πλατφόρμα έχει ως αποτέλεσμα να είναι ένας αποτελεσματικός και οικονομικός τρόπος παρουσίασης των προϊόντων.



Snap Inc. internal data. January 1 - June 30, 2022, Comparison of Catalog and Non-Catalog Shopping Lenses with at least 1 attributed purchase and over 100 impressions served

3.3.2 LEGO AR-Studio

Τον Δεκέμβριο του 2017 εταιρεία παιχνιδιών ανακοίνωσε την εφαρμογή επαυξημένης πραγματικότητας για iOS συσκευές, LEGO AR-Studio για την προβολή των προϊόντων της. Με την εφαρμογή ο/η χρήστης έχει την δυνατότητα να τοποθετήσει sets στον χώρο καθώς και να κατευθύνει κάποια από αυτά.



3.3.3 Nike 3D Billboard

Ένας πρωτότυπος τρόπος προβολής προϊόντων όπου δίνει την ψευδαίσθηση πως είναι μέρος του περιβάλλοντος είναι με την βοήθεια μεγάλων διαφημιστικών οθονών τοποθετημένες πάνω σε κτήρια. Εταιρεία υποδημάτων το 2023 στην Ιαπωνία αξιοποίησε αυτές τις οθόνες με τέτοιο τρόπο που τα ζευγάρια παπουτσιών «ξεπρόβαλαν» έξω από την οθόνη.

ΚΕΦΑΛΑΙΟ 4 Προτεινόμενο Σύστημα

4.1 Σύστημα Διαχείρισης Προϊόντων

4.1.1 Περιγραφή

Το προτεινόμενο σύστημα αποτελείται από δύο μέρη, το σύστημα διαχείρισης βάσης δεδομένων και την εφαρμογή του κινητού που λειτουργεί ως την διεπαφή του πελάτη με αυτή την βάση.

Η διαχείριση των δεδομένων πραγματοποιείται σε μία single page ιστοσελίδα, η οποία εμπεριέχει όλες τις εγγραφές σε έναν πίνακα με κάθε στήλη να έχει τα χαρακτηριστικά όπως το ID, Product, Price, Price per Kilo και % Discount. Επιλέγοντας μία από αυτές τις εγγραφές ο/η διαχειριστής/στρια έχει την δυνατότητα να κάνει μετατροπές σε αυτές και να τις ενημερώνει, καθώς και να τις διαγράφει αντίστοιχα όπως και να προσθέτει νέες εγγραφές.

4.1.2 Τεχνολογίες

Ο διακομιστής έχει γραφτεί στην γλώσσα προγραμματισμού JavaScript αξιοποιώντας το Node.js. Μέσω αυτού είναι δυνατή η ανάπτυξη κώδικα για το back-end και την απάντηση των requests που κάνει η εφαρμογή επαυξημένης πραγματικότητας, καθώς και ο/η διαχειριστής/στρια κατά την διάρκεια επεξεργασίας των εγγραφών.

Οι εγγραφές είναι αποθηκευμένες σε JSON αρχείο με μορφή:
{"id":1, "product":"Example", "price":1.99, "pricekilo":3.99, "discount":10}

4.1.3 Επιλογές Διαχειριστή

Οι επιλογές του/ης διαχειριστή/στριας είναι οι εξής:

- Προσθήκη εγγραφών προϊόντων

Αφού συμπληρωθούν τα πεδία αναγνωριστικού, ονόματος, τιμής, τιμής κιλού και προσφοράς προϊόντος στο σύστημα διαχείρισης, τότε μπορεί να γίνει η άμεση προσθήκη του στο σύστημα για αναγνώριση από την εφαρμογή του κινητού.

- Επεξεργασία προϋπαρχουσών εγγραφών κάνοντας μετατροπές στα πεδία τους

Αντίστοιχα η διαδικασία επεξεργασίας προϊόντων είναι παρόμοια με αυτή της προσθήκης, με προϋπόθεση το ID της εγγραφής που τροποποιείτε να είναι έγκυρο, δηλαδή να υπάρχει στο σύνολο των εγγραφών. Μόλις γίνει η επιλογή update τροποποιείται το JSON αρχείο με της αλλαγές που ορίστηκαν για την συγκεκριμένη εγγραφή.

- Διαγραφή εγγραφών προϊόντων

Για την διαγραφή εγγραφών θα χρειαστεί μόνο το πεδίο ID του προϊόντος, με την ίδια προϋπόθεση της επεξεργασίας των εγγραφών, την εγκυρότητα του ID. Αφού γινεί η επιλογή της διαγραφής γίνονται οι μετατροπές του αρχείου JSON για την αφαίρεση της.

4.1.4 Source Code Snippets

Back-end Request Handling:

Για την ανάκτηση όλων των δεδομένων από τα αποθηκευμένα προϊόντα, γίνεται ανάγνωση του αρχείου JSON όπου είναι αποθηκευμένες οι πληροφορίες και επιστρέφεται ως JSON μορφή.

```
app.get("/api/Database.json", (req, res) => {
  fs.readFile("Database.json", (err, jsondata) => {
    let products = [];
    if (!err) products = JSON.parse(jsondata);
    res.status(200).json(products);
  });
});
```

Ορισμός συνάρτησης που καλείται αναλόγως τον τύπο του request σε κάθε route για την ανάκτηση και την επεξεργασία των εγγραφών.

```
app.get("/api/Database.json/:id", (req, res) => getEntry(req, res));
app.post("/api/Database.json", (req, res) => addEntry(req, res));
app.put("/api/Database.json/:id", (req, res) => updateEntry(req, res));
app.delete("/api/Database.json/:id", (req, res) => removeEntry(req, res));
```

Η συνάρτηση getEntry επιτρέπει την ανάκτηση των πληροφοριών μία μοναδικής εγγραφής, σε αντίθεση με την ανάκτηση ολόκληρου του JSON αρχείου.

```
function getEntry(req, res) {
```

```
  const id = parseInt(req.params.id);
  fs.readFile(
    "Database.json",
    function (err, data) {
      let products = [];
      if (!err) products = JSON.parse(data);
      res.status(200).json(products.filter((p) => p.id === id));
    },
  );
}
```

Η addEntry συνάρτηση προσθέτει μία εγγραφή στα προϊόντα με βάσει τα πεδία που έθεσαν οι χρήστες, εκτελώντας ένα POST request.

```
function addEntry(req, res) {
  const { id, product, price, pricekilo, discount } = req.body;
  const newProduct = { id: parseInt(id), product, price: parseFloat(price),
pricekilo: parseFloat(pricekilo), discount: parseInt(discount)};
  fs.readFile(
    "Database.json",
    function (err, data) {
      let product = [];
      if (!err) product = JSON.parse(data);
      product.push(newProduct);
      fs.writeFile(
        "Database.json",
        JSON.stringify(product),
        function (err) {
          if (err) {
            res.status(200).json(`Error adding id: ${id}`);
          } else {
            res.status(200).json(`Entry added with id: ${id}`);
          }
        },
      );
    },
  );
}
```

Το updateEntry αντίστοιχα ενημερώνει τα αποθηκευμένα δεδομένα αφού οι χρήστες τροποποιήσουν μία εγγραφή, εφόσον εντοπιστεί η εγγραφή με το ID που ορίστηκε.

```
function updateEntry(req, res) {
  const { id, product, price, pricekilo, discount } = req.body;
  const updatedProduct = { id: parseInt(id), product, price:
parseFloat(price), pricekilo: parseFloat(pricekilo), discount:
parseInt(discount) };
  fs.readFile(
    "Database.json",
```

```
    function (err, data) {
      let product = [];
      if (!err) product = JSON.parse(data);
      const anIndex = product.findIndex((p) => p.id === updatedProduct.id);
      if (anIndex < 0) {
        res.status(200).json(`Cannot find entry with ID: ${id}`);
        return;
      }
      product[anIndex] = updatedProduct;
      fs.writeFile(
        "Database.json",
        JSON.stringify(product),
        function (err) {
          if (err) {
            res.status(200).json(`Error updating entry with ID: ${id}`);
          } else {
            res.status(200).json(`Updated entry with ID: ${id}`);
          }
        },
      );
    },
  );
}
```

Για την διαγραφή εγγραφών του αρχείου χρησιμοποιείται η removeEntry συνάρτηση, όπου ομοίως με την updateEntry απαραίτητο στοιχείο είναι ένα έγκυρο ID.

```
function removeEntry(req, res) {
  const id = parseInt(req.body.id);
  fs.readFile(
    "Database.json",
    function (err, data) {
      let product = [];
      if (!err) product = JSON.parse(data);
      console.log(id)
      const anIndex = product.findIndex((p) => p.id === id);
      if (anIndex < 0) {
        res.status(200).json(`Cannot find entry with ID: ${id}`);
        return;
      }
      product.splice(anIndex, 1);
      fs.writeFile(
        "Database.json",
        JSON.stringify(product),
        function (err) {
          if (err) {
            res.status(200).json(`Error deleting entry with id: ${id}`);
          } else {
            res.status(200).json(`Deleted entry with id: ${id}`);
```

```
            }
          },
        );
      },
    );
}
```

Front-end:

Κάθε συνάρτηση έχει τον εξής σκοπό:

1. **loadPages**, **displayTable** και **updatePagination**:
   Προβολή των εγγραφών στον πίνακα χωρισμένες σε σελίδες, με 26 εγγραφές ανά σελίδα.

2. **selectEntry**:
   Συμπλήρωση των πεδίων με τα στοιχεία της εγγραφής αμέσως μετά της επιλογής της από τον πίνακα για επεξεργασία/διαγραφή.

3. **getFormData**
   Ανάγνωση των πληροφοριών που έχουν τεθεί στα πεδία του προϊόντος και επιστροφή σε μορφή JSON.

4. **addEntry, updateEntry** και **removeEntry**:
   Με το κάλεσμα της συνάρτησης getFormData ορίζεται το πεδίο body των δεδομένων που θα μεταφερθούν με POST, PUT και DELETE request αντίστοιχα.

```
const rowsPerPage = 26;
let currentPage = 1;

function loadPages() {
    fetch("/api/Database.json")
    .then((res) => res.json())
    .then((json) => displayTable(currentPage, json))
}

function displayTable(page, data) {

    const DatabaseContainer = document.getElementById("DatabaseContainer");
    DatabaseContainer.addEventListener("click", selectEntry);

    console.log(data)
    const table = document.getElementById("myTable");
    const startIndex = (page - 1) * rowsPerPage;
    const endIndex = startIndex + rowsPerPage;
    const slicedData = data.slice(startIndex, endIndex);
    console.log(slicedData)
    table.innerHTML = `
    <tr>
        <th>ID</th>
        <th>Product</th>
```

```
        <th>Price</th>
        <th>Price Per Kilo</th>
        <th>% Discount</th>
    </tr>
    `;

    slicedData.forEach(item => {
        const row = table.insertRow();
        const idCell = row.insertCell(0);
        const productCell = row.insertCell(1);
        const priceCell = row.insertCell(2);
        const pricePerKiloCell = row.insertCell(3);
        const discountCell = row.insertCell(4);
        idCell.innerHTML = item.id;
        productCell.innerHTML = item.product;
        priceCell.innerHTML = item.price;
        pricePerKiloCell.innerHTML = item.pricekilo;
        discountCell.innerHTML = item.discount;
    });

    updatePagination(page, data);
}

function updatePagination(currentPage, data) {
    const pageCount = Math.ceil(data.length / rowsPerPage);
    const paginationContainer = document.getElementById("pagination");
    paginationContainer.innerHTML = "";

    for (let i = 1; i <= pageCount; i++) {
        const pageLink = document.createElement("a");
        pageLink.href = "#";
        pageLink.innerText = i;
        pageLink.onclick = function () {
            displayTable(i, data);
        };
        if (i === currentPage) {
            pageLink.style.fontWeight = "bold";
        }
        paginationContainer.appendChild(pageLink);
        paginationContainer.appendChild(document.createTextNode(" "));
    }
}

function selectEntry(event) {
    if (!event.target.outerHTML.startsWith("<td>")) return;
    let aRow = event.target.parentNode;
    let inputs=document.querySelectorAll("#linkForm input");
    for (let i=0; i < inputs.length; i++){
```

```
        inputs[i].value = aRow.children[i].innerText;
    }
    let rows = [...aRow.parentNode.children];
    rows.forEach((r => r.classList.remove("selected")));
    aRow.classList.add("selected");
}

function getFormData() {
    let entry = {
        id: document.querySelector("#linkForm [name='id']").value,
        product: document.querySelector("#linkForm [name='product']").value,
        price: document.querySelector("#linkForm [name='price']").value,
        pricekilo: document.querySelector("#linkForm
[name='pricekilo']").value,
        discount: document.querySelector("#linkForm [name='discount']").value
    };
    return JSON.stringify(entry);
}

function addEntry() {
    let bodyData = getFormData();
    fetch(`/api/Database.json`, {
        headers: {
            'Accept': 'application/json',
            'Content-Type': 'application/json'
        },
        method: "POST",
        body: bodyData
    })
        .then((res) => res.json())
        .then((json) => {
            alert(json);
            loadPages();
        })
        .catch((err) => alert("error:", err));
}

function updateEntry() {
    let bodyData = getFormData();
    fetch(`/api/Database.json/`+bodyData.id, {
        headers: {
            'Accept': 'application/json',
            'Content-Type': 'application/json'
        },
        method: "PUT",
        body: bodyData
    })
        .then((res) => res.json())
```

```
        .then((json) => {
            alert(json);
            loadPages();
        })
        .catch((err) => alert("error:", err));
}

function removeEntry() {
    let bodyData = getFormData();
    fetch(`/api/Database.json/`+bodyData.id, {
        headers: {
            'Accept': 'application/json',
            'Content-Type': 'application/json'
        },
        method: "DELETE",
        body: bodyData
    })
        .then((res) => res.json())
        .then((json) => {
            alert(json);
            loadPages();
        })
        .catch((err) => alert("error:", err));
}
```

4.2 Εφαρμογή Προβολής Προϊόντων

4.2.1 Περιγραφή

Η εφαρμογή που έχει δημιουργηθεί στο Unity, προορίζεται για κινητά Android και δρα ως μέσο διεπαφής του χρήστη με τα ψηφιακά καταχωρημένα δεδομένα των προϊόντων του καταστήματος. Έχει σκοπό την προβολή εικονικών ετικετών τοποθετημένες πάνω στην επιφάνεια του αναγνωριστικού της εγγραφής του προϊόντος. Αυτό το αναγνωριστικό είναι ένας QR κωδικός όπου αντιστοιχεί με το ID του προϊόντος.

Έτσι για παράδειγμα αν το αναγνωριστικό είναι η τιμή 1 και αναγνωριστεί ο QR κωδικός με μία τυπική εφαρμογή θα προβληθεί ο αριθμός 1, με την χρήση της εφαρμογής επαυξημένης πραγματικότητας όμως θα γίνει η ανάκτηση και η προβολή των πληροφοριών της εγγραφής με το αντίστοιχο αναγνωριστικό.

4.2.2 Τεχνολογίες

Η γλώσσα προγραμματισμού που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής είναι η C#, με συνδυασμό κάποιων βιβλιοθηκών και namespaces ονομαστικά:

- ZXing
- UnityEngine.Networking
- UnityEngine.XR.ARFoundation
- UnityEngine.XR.ARCore

- UnityEngine.XR.ARSubsystems

➢ ZXing

Το ZXing ευθύνεται για την αποκωδικοποίηση των QR Codes καθώς και την παραγωγή τους σε μορφή εικόνας PNG.

➢ UnityEngine.Networking

Με το UnityEngine.Networking είναι δυνατό να επικοινωνεί η εφαρμογή με τον διακομιστή για την ανάκτηση των πληροφοριών των προϊόντων.

➢ UnityEngine.XR.ARFoundation

Το UnityEngine.XR.ARFoundation εμπεριέχει κλάσεις για την υποστήριξη των λειτουργιών επαυξημένης πραγματικότητας, όπως το ARSession, ARTrackedImage, ARTrackedImageManager και άλλα.

➢ UnityEngine.XR.ARCore

Το UnityEngine.XR.ARCore λειτουργεί ως την διεπαφή του ARCore SDK της Google με το Unity, με αποτέλεσμα να είναι δυνατή η υποστήριξη των τεχνολογιών του SDK.

➢ UnityEngine.XR.ARSubsystems

Αντίστοιχο ρόλο έχει το UnityEngine.XR.ARSubsystems, όπου βρίσκονται κλάσεις σαν την MutableRuntimeReferenceImageLibrary όπου ευθύνεται για την διαχείριση και την αποθήκευση των εικόνων που αναγνωρίζονται στο περιβάλλον κατά την ώρα εκτέλεσης.

4.2.3 Επιλογές Χρήστη

Στο μενού ρυθμίσεων της εφαρμογής παρουσιάζονται οι εξής επιλογές για τον/ην χρήστη:

- Το νόμισμα που χρησιμοποιείται:
  - Ευρώ
  - Αμερικάνικο Δολάριο
  - Βρετανική Λύρα
  - Γιαπωνέζικο Γιεν
  - Τούρκικη Λύρα
- Μέγεθος Γραμματοσειράς:
  - Μεγάλο
  - Μεσαίο
  - Μικρό
- Γλώσσα:
  - Ελληνικά
  - Αγγλικά
  - Γιαπωνέζικα
  - Τούρκικα

4.2.4 Source Code Snippets

Προσθήκη των εγγραφών του GET response της ιστοσελίδας στο dictionary DataBase.

```
foreach (var entry in entries)
{
      productJSON = ProductInfo.CreateFromJSON("{" + entry + "}");
      DataBase.Add(productJSON.id, productJSON);
}
```

```
[System.Serializable]
public class ProductInfo
{

    public int id;
    public string product;
    public float price;
    public float pricekilo;
    public int discount;

    public static ProductInfo CreateFromJSON(string json)
    {
        return JsonUtility.FromJson<ProductInfo>(json);
    }
}
```

Ανάκτηση πληροφορίων προϊόντος σε μορφή λίστας από το dictionary όπου εμπεριέχει όλες τις εγγραφές. Σε περίπτωση που δεν βρεθεί το κλειδί στο dictionary, ως κλειδί λειτουργεί το ID των προϊόντων σε αυτή την περίπτωση διότι είναι μοναδικό, τότε επιστρέφει στα περιεχόμενα της λίστας ότι δεν είναι έγκυρη εγγραφή.

```
public string[] GetProductEntry(int productId)
    {
        List<string> output = new List<string>();
        if (DataBase.ContainsKey(productId))
        {

            var entry = DataBase[productId];
            output.Add(entry.id.ToString());
            output.Add(entry.product);
            output.Add(entry.price.ToString());
            output.Add(entry.pricekilo.ToString());
            output.Add(entry.discount.ToString());

            return output.ToArray();
        }
        else
        {
            for (int i = 0; i < 5; i++) {
```

```
                output.Add("Invalid Entry");
            }
            return output.ToArray();
        }
    }
```

Ο παρακάτω κώδικας ευθύνεται για την δημιουργία των εικονικών ετικετών. Μόλις αναγνωριστεί μία εικόνα η οποία έχει προστεθεί στο ARTrackedImageManager γίνεται έλεγχος αν έχει δημιουργηθεί προηγουμένως βάσει το όνομα της ψηφιακής εικόνας του QR Code. Η εικόνα με ονομασία "Unused" λειτουργεί ως placeholder, διότι το TrackedImageManager πρέπει να έχει τουλάχιστον μία εικόνα για να αναγνωρίσει, και δεν αξιοποιείται στην εφαρμογή.

Αφού αναγνωριστεί μία εικόνα δημιουργείτε η εικονική ετικέτα του προϊόντος, σε περίπτωση που είναι σε προσφορά τότε έχει διαφορετική όψη, με κάθε περιγραφή της ετικέτας να βασίζεται στις επιλογές των ρυθμίσεων που έχουν τεθεί.

Έπειτα αναγνωρίζεται κάθε πεδίο κειμένου της εικονικής ετικέτας ώστε να οριστεί κάθε ένα από αυτά για την αναγραφή των πληροφοριών του προϊόντος. Τα πεδία που έχουν να κάνουν με το κόστος του προϊόντος προκύπτουν με τον υπολογισμό της αξίας στο νόμισμα του ευρώ, πολλαπλασιάζοντας το με την ποσοστιαία διαφορά του επιλεγμένου νομίσματος. (παράδειγμα: αντικείμενο αξίας: 3.45 € αν το επιλεγμένο νόμισμα είναι το Αμερικάνικο δολάριο με αξία 1.11$ = 1€ τότε θα υπολογιστεί ο πολλαπλασιασμός 3.45 * 1.11 = 3.83 $ όπου θα είναι και η αναγραφόμενη τιμή στην ετικέτα. Σε περίπτωση που το προϊόν είναι σε προσφορά εκτελείται ένας επιπλέον υπολογισμός αφαίρεσης για την τελική τιμή του αντικειμένου.

Τέλος αφού δημιουργηθεί η εικονική ετικέτα προστίθεται στο dictionary με όλες τις δημιουργημένες ετικέτες, με αποτέλεσμα να επιτυγχάνονται δύο στόχοι, έλεγχος αν έχει ήδη δημιουργηθεί η εικονική ετικέτα, και δυνατότητα αλλαγής των πεδίων δημιουργημένων ετικετών με την μετατροπή των ρυθμίσεων.

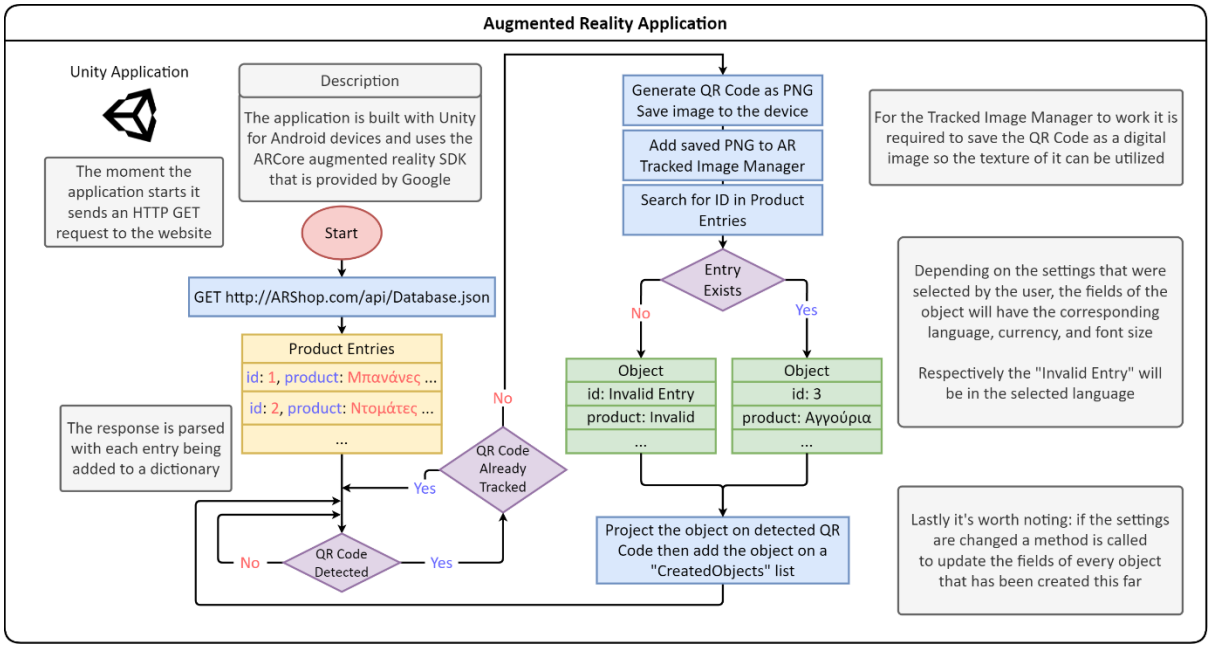
```
private void OnTrackedImagesChanged(ARTrackedImagesChangedEventArgs eventArgs)
{

        foreach (var trackedImage in eventArgs.added) {

            var imageName = trackedImage.referenceImage.name;

            if (!_instantiatedPrefabs.ContainsKey(imageName)) {

                currency = settings.getCurrencySelection();
                fontsize = settings.getFontSizeSelection();
                language = settings.getLanguageSelection();

                if (imageName != "Unused") {

                    productInformation =
requestHandler.GetProductEntry(int.Parse(imageName));

                    if (productInformation[0] != "Invalid Entry" &&
int.Parse(productInformation[4]) != 0) {
```

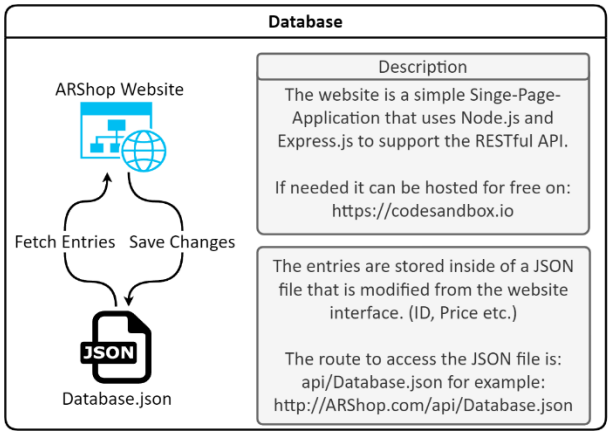
```
                        chosenPrefab = Instantiate(DiscountPrefab,
trackedImage.transform);
                    }
                    else {
                        chosenPrefab = Instantiate(ProductPrefab,
trackedImage.transform);
                    }

                    prefabTextComponents =
chosenPrefab.GetComponentsInChildren<TMP_Text>();

                    var productName = prefabTextComponents[0];
                    var productPrice = prefabTextComponents[1];
                    var productPriceSign = prefabTextComponents[2];
                    var productPerKiloText = prefabTextComponents[3];
                    var productPricePerKilo = prefabTextComponents[4];
                    var productPricePerKiloSign = prefabTextComponents[5];

                    if (productInformation[0] != "Invalid Entry") {
                        productName.text = productInformation[1];

                        productPrice.text = string.Format("{0:.00}",
float.Parse(productInformation[2]) * currencyConversion[currency]);
                        productPricePerKilo.text = string.Format("{0:.00}",
float.Parse(productInformation[3]) * currencyConversion[currency]);

                        if (int.Parse(productInformation[4]) != 0 ) {
                            prefabTextComponents[6].text =
discountMessage[language] + productInformation[4] + "%";
                            discount = 1 - (int.Parse(productInformation[4]) /
100F) ;

                            prefabTextComponents[7].text =
string.Format("{0:.00}", discount * float.Parse(productInformation[2]) *
currencyConversion[currency]);
                            prefabTextComponents[8].text =
string.Format("{0:.00}", discount * float.Parse(productInformation[3]) *
currencyConversion[currency]);
                        }

                    } else {
                        productName.text = errorMessage[language];
                        productPrice.text = errorMessage[language];
                        productPricePerKilo.text = errorMessage[language];
                    }

                    productName.fontSize = fontSize[fontsize];
                    productPriceSign.text = currencySign[currency];
```

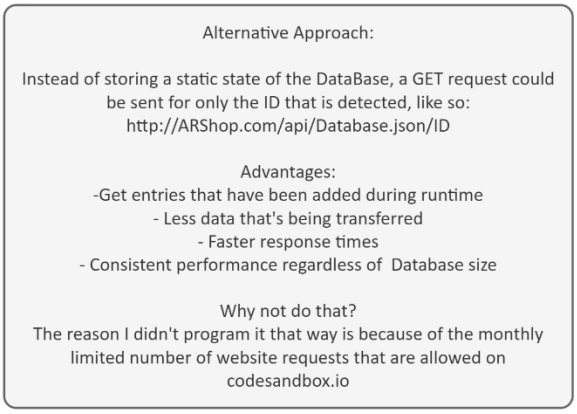
```
                    productPerKiloText.text = ppkMessage[language];
                    productPricePerKiloSign.text =
currencySign[currency];

                }
                else {
                    chosenPrefab = Instantiate(ProductPrefab,
trackedImage.transform);
                }

                _instantiatedPrefabs[imageName] = chosenPrefab;

            }
        }

    }
```

ΚΕΦΑΛΑΙΟ 5 Μελέτη Περίπτωσης

5.1 Workflow







Αμέσως αφού ξεκινήσει η εκτέλεση της εφαρμογής, πραγματοποιείται ένα GET request στην βάση δεδομένων του καταστήματος (για παράδειγμα http://arshop.com/api/Database.json), για την ανάκτηση όλων των προϊόντων από την ιστοσελίδα σε JSON μορφή. Έτσι κάθε εγγραφή αυτού του JSON αρχείου προστίθεται σε ένα dictionary για μελλοντική χρήση.

Εναλλακτικά θα μπορούσε να γίνει η ανάκτηση μόνο του προϊόντος που έχει ανιχνευθεί. Ακολουθώντας αυτή την προσέγγιση τα πλεονεκτήματα θα ήταν:

- Ανάκτηση εγγραφών οι οποίες προστέθηκαν εν ώρα εκτέλεσης της εφαρμογής
- Μεταφορά λιγότερων δεδομένων
- Γρηγορότερη ανάκτηση των δεδομένων
- Σταθερές ταχύτητες ανεξαρτήτως του μεγέθους της βάσης δεδομένων, επιτρέποντας την κλιμάκωση της

Ο λόγος που δεν έγινε αυτή η υλοποίηση είναι πως η ιστοσελίδα που χρησιμοποιήθηκε είχε όριο των GET requests που μπορούσαν να πραγματοποιηθούν.

Μετά την αποθήκευση των δεδομένων του διακομιστή στο dictionary, ο/η χρήστης έχει την δυνατότητα να στρέψει την κάμερα σε QR κωδικούς. Η εφαρμογή μόλις αναγνωρίσει και

αποκωδικοποιήσει έναν από τους κωδικούς, ελέγχει αν εμπεριέχεται στο MutableRuntimeReferenceImageLibrary.

Σε περίπτωση που δεν βρεθεί στην βιβλιοθήκη εικόνων τότε δημιουργείται το αντίστοιχο QR code που αποκωδικοποιήθηκε σε μορφή ψηφιακής εικόνας. Έπειτα προστίθεται στην βιβλιοθήκη για να μπορεί να αναγνωριστεί και να προβληθούν αντικείμενα πάνω στην επιφάνεια των QR codes.

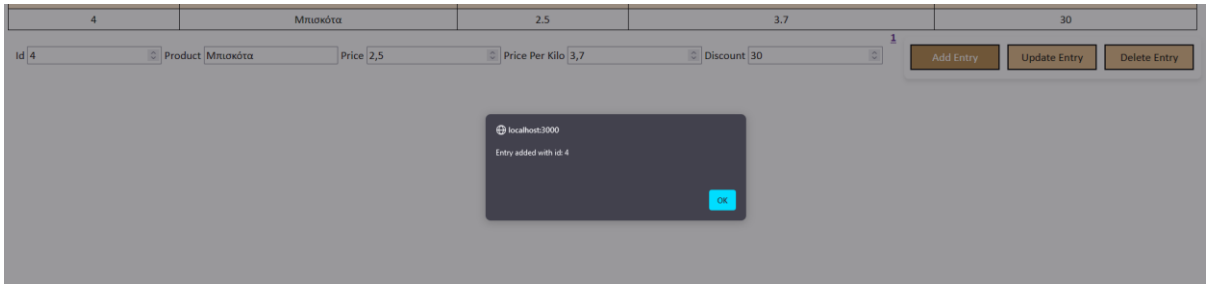
Μόλις προστεθεί μία QR Code εικόνα στο MutableRuntimeReferenceImageLibrary γίνεται η έυρεση του ID στο dictionary, δημιουργείται ένα τρισδιάστατο αντικείμενο που ανατίθεται στην συγκεκριμένη εικόνα και προστίθενται οι πληροφορίες σε αυτό. Έτσι είναι δυνατό να υπάρχουν πολλαπλά αντικείμενα τα οποία προβάλλονται ταυτοχρόνως.

Κάθε αντικείμενο που έχει δημιουργηθεί προστίθεται σε μία λίστα με σκοπό τη αποτροπή αντιστοίχισης πολλαπλών όμοιων αντικειμένων σε έναν κωδικό QR, καθώς και την δυνατότητα αλλαγής των πεδίων των δημιουργημένων αντικειμένων σε περίπτωση που ο/η χρήστης αλλάξει τις ρυθμίσεις της εφαρμογής.

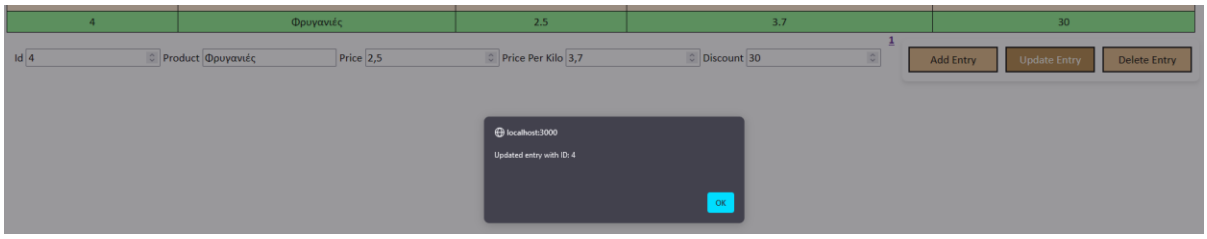
5.2 Στιγμιότυπα Επεξεργασίας Προϊόντων

Ιστοσελίδα επεξεργασίας εγγραφών:

Προσθήκη εγγραφής:



Τροποποίηση εγγραφής:



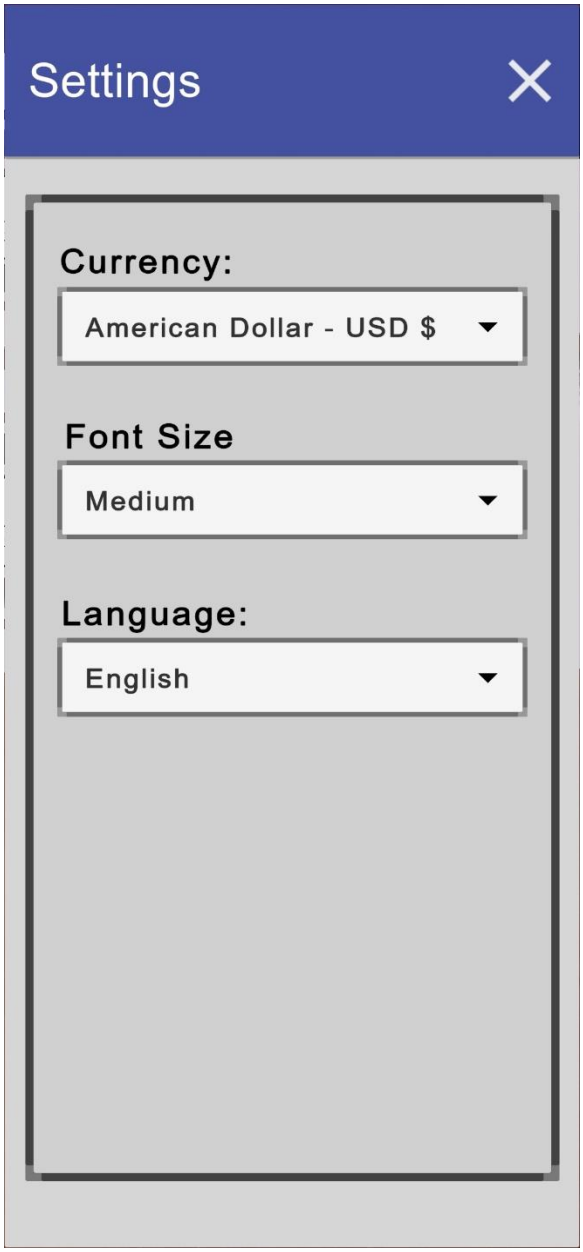
5.3 Στιγμιότυπα Εκτέλεσης Εφαρμογής

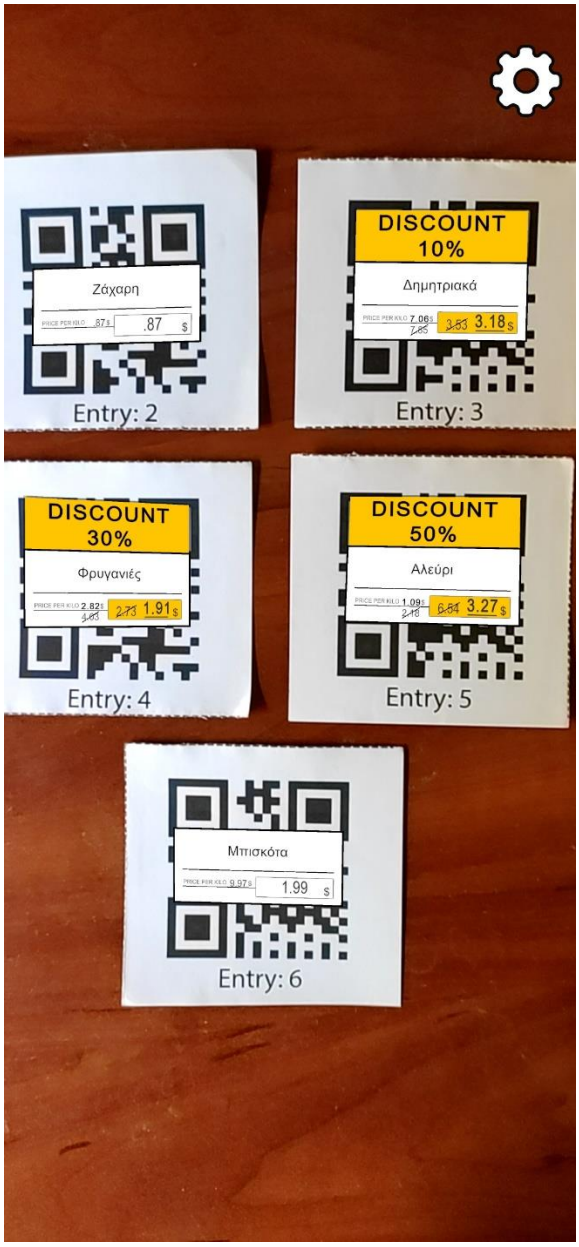
Ζάχαρη
.80 €
Entry: 2
ΠΡΟΣΦΟΡΆ
10%
Δημητριακά
2.92€
Entry: 3
Μη Έγκυρη Εγγραφή
Entry: 4
ΠΡΟΣΦΟΡΆ
50%
Αλεύρι
3.00€
Entry: 5
Μπισκότα
1.83 €
Entry: 6

Ζάχαρη
.80 €
Entry: 2
ΠΡΟΣΦΟΡ
10%
Δημητριακά
2.92€
Entry: 3
ΠΡΟΣΦΟΡΆ
50%
Αλεύρι
3.00€
Entry: 5

Settings
Currency:
Euro - EUR €
Font Size
Medium
Language:
English
Greek - Ελληνικά
English
Japanese - 日本語
Turkish - Türkçe

Settings
Currency:
Euro - EUR €
Euro - EUR €
American Dollar - USD $
British Pound - GBP £
Japanese Yen - JPY ¥
Turkish Lira - TRY ₺


Settings
Currency:
American Dollar - USD $
Font Size
Medium
Language:
English


Ζάχαρη
.87 $
Entry: 2
DISCOUNT 10%
Δημητριακά
3.18 $
Entry: 3
DISCOUNT 30%
Φρυγανιές
1.91 $
Entry: 4
DISCOUNT 50%
Αλεύρι
3.27 $
Entry: 5
Μπισκότα
1.99 $
Entry: 6

ΚΕΦΑΛΑΙΟ 6 Συμπεράσματα και Μελλοντικές Προεκτάσεις

Συμπεράσματα

Με το πέρας της πτυχιακής εργασίας μπορούμε να συμπεράνουμε πως είναι δυνατή η αξιοποίηση της τεχνολογίας επαυξημένης πραγματικότητας σε φυσικό χώρο πωλήσεων προϊόντων με εικονικές ετικέτες. Οι χρήστες της εφαρμογής μπορούν να αντλήσουν πληροφορίες σε πιο ικανοποιητικό βαθμό από αυτόν που προσφέρει μία τυπική ετικέτα, χάρις στις επιπλέον δυνατότητες που προσφέρει η εφαρμογή.

Θετικά:

1. Εύκολος τρόπος αλλαγής γραφικών των ετικετών
2. Άμεση επεξεργασία και ενημέρωση των προϊόντων
3. Δυνατότητα συμπυκνωμένων πληροφοριών για κάθε εγγραφή
4. Εύκολη επέκταση των λειτουργιών

Αρνητικά:

1. Πιθανότητα να διαφύγει της προσοχής του πελάτη ένα προϊόν σε προσφορά αν δεν στρέψει τη συσκευή του για να δημιουργηθεί η εικονική ετικέτα
2. Απαίτηση καλής ποιότητας κάμερας για την αποκωδικοποίηση των QR κωδικών σε απόσταση

Μελλοντικές Προεκτάσεις

Κάποιες από τις μελλοντικές προεκτάσεις που μπορούν να εφαρμοστούν στα συστήματα είναι:

1. Access Control και Login Credentials

Σε περίπτωση που η ιστοσελίδα χρησιμοποιείται σε μία αλυσίδα καταστημάτων, με την βοήθεια του access control ο κάθε λογαριασμός αντιστοιχεί σε ένα συγκεκριμένο κατάστημα, μη επιτρέποντας την ανεπιθύμητη επεξεργασία εγγραφών.

2. Επιλογή «Πληροφορίες» στις Εικονικές Ετικέτες

Για την προβολή επιπρόσθετων πληροφοριών, όπως π.χ. συστατικά, είναι δυνατή η πρόσθεση επιλογής «Πληροφορίες» σε κάθε εικονική ετικέτα, έτσι, μόλις ο χρήστης την επιλέξει να ανοίγει ένα παράθυρο στην εφαρμογή το οποίο εμπεριέχει ό,τι έχει οριστεί από το προσωπικό. Για τον ορισμό αυτών των πληροφοριών θα χρειαστεί και η πρόσθεση του αντίστοιχου πεδίου στην ιστοσελίδα.

3. Βελτιστοποίηση Εφαρμογής

Ένα αποτέλεσμα που έχει ο εντοπισμός δύο αντικειμένων ταυτόχρονα με την κάμερα, QR κωδικών μέσω του ZXing καθώς και τις αποθηκευμένες εικόνες του MutableRuntimeReferenceImageLibrary, είναι η ανάπτυξη υψηλών θερμοκρασιών της κάμερας. Με την μείωση των καρέ που επεξεργάζεται η εφαρμογή ανά δευτερόλεπτο είναι δυνατό να μειωθεί η θερμοκρασία που αναπτύσσει.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Carmigniani, J. (2011). Augmented Reality: An Overview. *Handbook of augmented reality/Springer*.

2. Heilig, M. L. (1992). El cine del futuro: the cinema of the future. *Presence Teleoperators Virtual Environ.*, *1*(3), 279-294.

3. Henrysson, A., Billinghurst, M., & Ollila, M. (2005, October). Face to face collaborative AR on mobile phones. In *Fourth ieee and acm international symposium on mixed and augmented reality (ismar'05)* (pp. 80-89). IEEE.

4. Nowacki, P., & Woda, M. (2020). Capabilities of arcore and arkit platforms for ar/vr applications. In *Engineering in Dependability of Computer Systems and Networks: Proceedings of the Fourteenth International Conference on Dependability of Computer Systems DepCoS-RELCOMEX, July 1–5, 2019, Brunów, Poland* (pp. 358-370). Springer International Publishing.

5. Rosenberg, L. B. (1992). The use of virtual fixtures as perceptual overlays to enhance operator performance in remote environments. *Air force material command*, 1-42.

6. Edwards-Stewart, A., Hoyt, T., & Reger, G. (2016). Classifying different types of augmented reality technology. *Annual review of cybertherapy and telemedicine*, *14*, 199-202.

7. Mourtzis, D., Zogopoulos, V., & Xanthi, F. (2019). Augmented reality application to support the assembly of highly customized products and to adapt to production re-scheduling. *The International Journal of Advanced Manufacturing Technology*, *105*, 3899-3910.

8. https://web.cs.ucdavis.edu/~okreylos/ResDev/SARndbox/

9. https://en.wikipedia.org/wiki/ARToolKit

10. https://www.artoolworks.com/corporate/about-us.html

11. https://developers.google.com/ar/develop/fundamentals

12. https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@6.0/manual/index.html

13. https://www.lego.com/en-us/aboutus/news/2019/october/lego-ar-studio

14. https://www.ikea.com/global/en/newsroom/innovation/ikea-launches-ikea-place-a-new-app-that-allows-people-to-virtually-place-furniture-in-their-home-170912/

15. https://www.ikea.com/gb/en/home-design/

16. https://forbusiness.snapchat.com/blog/catalog-powered-shopping-lenses

17. https://forbusiness.snapchat.com/blog/introducing-catalog-powered-ar-shopping-lenses