



ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
**ΜΗΧΑΝΙΚΩΝ Η/Υ
& ΔΙΚΤΥΩΝ**

ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.

**ΠΜΣ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ
ΔΙΚΤΥΩΝ**

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΕΙ NFC (NEAR
FIELD COMMUNICATION) ΤΕΧΝΟΛΟΓΙΑ ΚΑΙ EMBEDDED
SENSORS ΓΙΑ ΕΞΥΠΝΕΣ ΣΥΣΚΕΥΕΣ ΚΙΝΗΤΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝ
ANDROID.**

Καρανίκας Μιχάλης (Α.Μ.15)

Επιβλέπων Καθηγητής: Δημόπουλος Δημήτριος.

Άρτα, Φεβρουάριος 2019



ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.

ΠΜΣ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ
ΔΙΚΤΥΩΝ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΕΙ NFC
(NEAR FIELD COMMUNICATION) ΤΕΧΝΟΛΟΓΙΑ ΚΑΙ
EMBEDDED SENSORS ΓΙΑ ΕΞΥΠΝΕΣ ΣΥΣΚΕΥΕΣ
ΚΙΝΗΤΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ANDROID

Καρανίκας Μιχάλης (Α.Μ.15)

Επιβλέπων Καθηγητής: Δημόπουλος Δημήτριος

Άρτα, Φεβρουάριος 2019

**APPLICATION DEVELOPMENT USING NEAR FIELD
COMMUNICATION (NFC) TECHNOLOGY AND
EMBEDDED SENSORS ON SMART PHONE ANDROID
DEVICE**

Εγκρίθηκε από τριμελή εξεταστική επιτροπή

Τόπος, Ημερομηνία

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1. Επιβλέπων καθηγητής

Δημήτριος Δημόπουλος,

Τίτλος, Βαθμίδα

2. Μέλος επιτροπής

Όνομα Επίθετο,

Τίτλος, Βαθμίδα

3. Μέλος Επιτροπής

Όνομα Επίθετο,

Τίτλος, Βαθμίδα

Ο/Η Διευθυντής/τρια του ΠΜΣ

Όνομα Επίθετο,

τίτλος, βαθμίδα

Υπογραφή

© Καρανίκας, Μιχάλης, 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.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν αποκλειστικά τον συγγραφέα και δεν αντιπροσωπεύουν τις επίσημες θέσεις του ΤΕΙ Ηπείρου.

Δήλωση μη λογοκλοπής

Δηλώνω υπεύθυνα και γνωρίζοντας τις κυρώσεις του Ν.2121/1993 περί Πνευματικής Ιδιοκτησίας, ότι η παρούσα μεταπτυχιακή εργασία είναι εξ ολοκλήρου αποτέλεσμα δικής μου ερευνητικής εργασίας, δεν αποτελεί προϊόν αντιγραφής ούτε προέρχεται από ανάθεση σε τρίτους. Όλες οι πηγές που χρησιμοποιήθηκαν (κάθε είδους, μορφής και προέλευσης) για τη συγγραφή της περιλαμβάνονται στη βιβλιογραφία.

Καρανίκας Μιχάλης

Υπογραφή

Ευχαριστίες

Θα ήθελα να ευχαριστήσω όλους όσους συνετέλεσαν στην ολοκλήρωση της παρούσας διπλωματικής εργασίας. Ευχαριστώ τον επιβλέποντα καθηγητή μου κ. Δημόπουλο Δημήτριο για την πολύτιμη βοήθεια και καθοδήγησή του κατά την εκπόνηση της εργασίας μου. Ιδιαίτερες ευχαριστίες θέλω να εκφράσω στην οικογένεια μου για την ηθική στήριξή τους και κυρίως στη σύζυγό μου για τη συνεχή συμπαράστασή της.

Περίληψη

Στην παρούσα διπλωματική εργασία πραγματοποιείται ιστορική αναδρομή, αναφέρονται τα βασικά χαρακτηριστικά και αναλύονται οι διαφορετικές εκδόσεις του λογισμικού Android, για την καλύτερη κατανόηση της εξέλιξής του. Ακόμα, η έρευνα καταπιάνεται με τον ορισμό, την ιστορία και την ανάπτυξη του gamification, αναλύει την Οκτάλυση του Yu-kai Chou και παρουσιάζει τα πλεονεκτήματα και τα μειονεκτήματά του gamification.

Επιπροσθέτως, ασχολείται με την τεχνολογία του NFC, τις εφαρμογές της, παρουσιάζει την ιστορία της, αναλύει τον NFC αναγνώστη και παρουσιάζει τα πλεονεκτήματα της τεχνολογίας. Στο τέταρτο και πέμπτο κεφάλαιο, παρουσιάζεται βήμα – βήμα η ανάπτυξη εφαρμογών σε Android λογισμικό και πραγματοποιείται ανάλυση του Android Studio. Τέλος, στα δύο τελευταία κεφάλαια παρουσιάζεται το Qt περιβάλλον και η εφαρμογή Gaming του ερευνητή, που απευθύνεται σε μια ομάδα security του αεροδρομίου των Ιωαννίνων κατά τη διάρκεια της υπηρεσίας τους (περίπολος).

Ανάμεσα στους στόχους της παρούσας έρευνας είναι η κατανόηση του λογισμικού Android και των χαρακτηριστικών του, η παρουσίαση και κατανόηση του gamification και της συμβολής του, η ανάδειξη της τεχνολογίας NFC, των πλεονεκτημάτων της αλλά και η δυνατότητα πολυεπίπεδης εφαρμογής της. Τέλος, στοχεύει στην παρουσίαση της βήμα – βήμα ανάπτυξης μίας εφαρμογής Android, στην ανάδειξη της συμβολής του περιβάλλοντος Qt στην ανάπτυξη αυτή και στην προώθηση της εφαρμογής Gaming. Εδώ, επιβάλλεται να αναφερθεί ότι για τα δευτερογενή δεδομένα που συλλέχθηκαν, χρησιμοποιήθηκε η μέθοδος της βιβλιογραφικής ανασκόπησης.

Η έρευνα κατέληξε στην τεράστια ανάπτυξη, προώθηση και αναγνώριση που κατέχει στη σύγχρονη εποχή το λογισμικό Android και στην καταλληλότητά του για την ανάπτυξη ενός πλήθους εφαρμογών. Επιπλέον, εξήγαγε ως συμπέρασμα την πολύ θετική συμβολή του gamification στο σύνολο των πτυχών της καθημερινότητας των ανθρώπων, προσωπικής και επαγγελματικής και κατέληξε στην ευελιξία της NFC τεχνολογίας και στην δυνατότητα πολυεπίπεδης εφαρμογής της. Τέλος, κατέστη κατανοητή η σύνθεση και δομή των εφαρμογών Android και κατανοήθηκε το θετικό αποτέλεσμα της εφαρμογής Gaming.

Λέξεις κλειδιά : Android, Gamification, NFC, smartphones.

Abstract

In the following paper, a historical review is made, the basic features and the different versions of Android software are mentioned and analyzed, in order to understand better its evolution. Furthermore, the research deals with the definition, history and development of gamification, analyzes the Yu – kai Chou Octalysis, and presents the advantages and disadvantages of gamification.

In addition, the research deals with NFC technology, its applications, presents its history, analyzes the NFC reader and presents the benefits of the technology. In the fourth and fifth chapters, the step by step development of applications in Android software is presented and the Android Studio is analyzed. Finally, in the last two chapters, the Qt environment and the researcher's Gaming application, which is addressed to a security team at Ioannina Airport during their service (patrol), is presented.

Among the goals of the following research are the understanding of the Android software and its features, the presentation and understanding of gamification and its contribution, the highlighting of NFC technology, its advantages and its ability to apply in multiple levels. Finally, the paper aims to present the step by step development of an Android application, to highlight the contribution of the Qt environment in the development process and to promote the Gaming application. Here, it should be noted that for the collected secondary data, the bibliographic review method was used.

The research has led to the immense development, promotion and recognition that Android software possesses in the modern age and its suitability to develop a multitude of applications. In addition, it concluded the very positive contribution of gamification to all aspects of everyday, personal and professional life, and resulted in the flexibility of NFC technology and its multilevel applicability. Finally, the composition and structure of Android application and the positive effect of Gaming were understood.

Key Words: Android, Gamification, NFC, smartphones.

Πίνακας Περιεχομένων

Ευχαριστίες	vii
Περίληψη	viii
Abstract	x
Περιεχόμενα εικόνων – διαγραμμάτων.	xiii
Συντομογραφίες	xv
1. Κεφάλαιο 1 ^ο : Το λειτουργικό πρόγραμμα Android	1
1.1. Ετυμολογία	1
1.2. Ιστορία του Android	1
1.3. Αρχιτεκτονική του Android	3
1.3.1. Πυρήνας Linux (Linux Kernel)	4
1.3.2. Βιβλιοθήκες (Libraries)	4
1.3.3. Χρόνος Εκτέλεσης και η εικονική μηχανή Dalvik (Android Runtime and Dalvik Virtual Machine)	4
1.3.4. Πλαίσιο Εφαρμογής (Application Framework)	5
1.3.5. Εφαρμογές (Applications)	6
1.4. Πωλήσεις	6
1.5. Εκδόσεις και χαρακτηριστικά	7
1.5.1. Παλιότερες εκδόσεις Android	7
1.5.2. Android 6.0 Marshmallow	11
1.5.3. Android 7.1 Nougat	11
1.5.4. Android 8.0 Oreo	12
1.5.5. Android 9.0 Pie	12
2. Κεφάλαιο 2 ^ο : Gamification	14
2.1 Ιστορία gamification	16
2.2. Τεχνικές Ανάπτυξης Gamification.	17
2.3. Οι 8 Βασικές Μονάδες του Gamification(Οκτάλυση)	18
2.4. Χαρακτηριστικά παραδείγματα εφαρμογής του Gamification	21
2.5. Οφέλη – Πλεονεκτήματα- Μειονεκτήματα (Κριτική Gamification)	23
3. Κεφάλαιο 3 ^ο : Η τεχνολογία του NFC	25
3.1. Τι είναι το RFID	25
3.2. NFC ιστορική αναδρομή	26
3.3. Εφαρμογές NFC	27
3.3.1. Πληρωμές	27
3.3.2. Μεταφορές	27

3.3.3.	Υγεία	28
3.3.4.	Επιχειρήσεις	28
3.3.5.	Εκπαίδευση	28
3.4.	NFC Αναγνώστης (NFC Reader).....	29
3.5.	Πλεονεκτήματα NFC	29
4.	Κεφάλαιο 4ο: Ανάπτυξη Εφαρμογών σε Android.....	31
4.1.	Εγκατάσταση Λογισμικού	31
4.2.	Ανάπτυξη Πηγαίου Κώδικα Εφαρμογής	31
4.3.	Αποσφαλμάτωση (Debugging) και Δοκιμαστική Φάση Εφαρμογής	32
4.4.	Τελική έκδοση και δημοσίευση της εφαρμογής στο κοινό.	33
4.5.	Android SDK.	33
5.	Κεφάλαιο 5 ^ο : Android Studio περιβάλλον.....	36
6.	Κεφάλαιο 6 ^ο : Qt περιβάλλον.	37
7.	Κεφάλαιο 7 ^ο : Η εφαρμογή Gaming.	39
7.1.	Προδιαγραφές χρήσης της εφαρμογής.	39
7.2.	Σχεδίαση και ανάλυση της εφαρμογής.	40
7.3.	Υλοποίηση της εφαρμογής	54
7.4.	Παρουσίαση της εφαρμογής.	57
8.	Συμπεράσματα	65
9.	Επίλογος.....	70
	Βιβλιογραφία	71
	ΠΑΡΑΡΤΗΜΑ I	74
	ΠΑΡΑΡΤΗΜΑ I I	75
	ΠΑΡΑΡΤΗΜΑ I I I	79
	ΠΑΡΑΡΤΗΜΑ IV	81
	ΠΑΡΑΡΤΗΜΑ V	82
	ΠΑΡΑΡΤΗΜΑ VI	83
	ΠΑΡΑΡΤΗΜΑ VII	85
	ΠΑΡΑΡΤΗΜΑ VIII.....	86

Περιεχόμενα εικόνων – διαγραμμάτων.

Γράφημα 1. 1: Πωλήσεις Λειτουργικών Συστημάτων για Smartphone 2013.....	6
Εικόνα 1. 1: Λογότυπο Android.....	1
Εικόνα 1. 2: Μέλη της Open Handset Alliance.....	3
Εικόνα 1. 3: Τα βασικά περιεχόμενα του λειτουργικού συστήματος Android.	3
Εικόνα 1. 6: Λογότυπο Android Donut.	8
Εικόνα 1. 5: Λογότυπο Android Cup Cake	8
Εικόνα 1. 8: Λογότυπο Android Fro Yo.....	8
Εικόνα 1. 7: Λογότυπο Android Éclair.	8
Εικόνα 1. 9: Λογότυπο Android Gingerbread.	8
Εικόνα 1. 10: Λογότυπο Android Honeycomb.	9
Εικόνα 1. 11: Λογότυπο Android Ice Cream Sandwich.....	9
Εικόνα 1. 12: Λογότυπο Android Jelly Bean.	9
Εικόνα 1. 13: Λογότυπο Android Kit Kat.	10
Εικόνα 1. 14: Λογότυπο Android Lollipop.	10
Εικόνα 1. 15: Λογότυπο Android Marshmallow.....	11
Εικόνα 1. 16: Λογότυπο Android Nougat.	11
Εικόνα 1. 17: Λογότυπο Android Oreo.	12
Εικόνα 1. 18: Λογότυπο Android Pie.....	13
Εικόνα 2. 1: Διάκριση του gamification ανάμεσα σε παρεμφερείς όρους σχετικούς με το gaming.	14
Εικόνα 2. 2: Ιστορική αναδρομή Gamification.	16
Εικόνα 2. 3: Κύκλος Gartner, Ιούλιος 2013.	17
Εικόνα 2. 4: Η Οκτάλυση του Yu-kai Chou.....	18
Εικόνα 2. 5: Gamification.....	23
Εικόνα 3. 1: RFID tag.....	26
Εικόνα 3. 2: Λογότυπο NFC.....	26
Εικόνα 3. 3: NFC reader εισόδου/εξόδου.....	29
Εικόνα 4. 1: Πρώτο Βήμα – Εγκατάσταση λογισμικού.	31

Εικόνα 4. 2: Δεύτερο βήμα – Ανάπτυξη πηγαίου κώδικα εφαρμογής.....	32
Εικόνα 4. 3: Τρίτο βήμα – Δοκιμή και Debugging της εφαρμογής.	33
Εικόνα 4. 4: Τέταρτο βήμα – Δημοσίευση της εφαρμογής.....	33
Εικόνα 4. 5: NDK και SDK package.....	34
Εικόνα 5. 1: Προεπιλεγμένη διάταξη του Android Studio.....	36
Εικόνα 5. 2: Αρχική εικόνα ελέγχου εφαρμογής.....	36
Εικόνα 6. 1: Λογότυπο Qt development	37
Εικόνα 6. 2: Το περιβάλλον της Qt (Qt Creator).....	38
Εικόνα 7. 1: Περιβάλλον Qt και φάκελοι εφαρμογής.	55
Εικόνα 7. 2: Φάκελος Sources.....	55
Εικόνα 7. 3: Φάκελος Forms.	56
Εικόνα 7. 4: Τόπος αποθήκευσης εικόνων.	56
Εικόνα 7. 5: Μέγεθος εικόνων.	57
Εικόνα 7. 6: Αρχική εικόνα εφαρμογής.	58
Εικόνα 7. 7: Το Menu της εφαρμογής.....	58
Εικόνα 7. 8: Το Profile κάθε Security.	59
Εικόνα 7. 9: Puzzle View Menu.	60
Εικόνα 7. 10: NFC Menu.	61
Εικόνα 7. 11: Αρχική μορφή του Puzzle.....	62
Εικόνα 7. 12: Ολοκληρωμένη εικόνα Puzzle.....	62
Εικόνα 7. 13: Categories Menu.	62
Εικόνα 7. 14: Αρχική μορφή του Gallery.....	64
Εικόνα 7. 15: Ολοκλήρωση της πρώτης εικόνας και αποθήκευση στο Gallery.	64

Συντομογραφίες

API	Application Programming Interface
ART	Android Runtime
SDK	Software Developers Kit
IDE	Integrated Development Enviroment
SVG	Scalable Vector Graphics.
ADT	Android Development Tools
NFC	Near Field Communication
GUI	Graphical User Ineterface
RFID	Radio – Frequency Indentification

1. Κεφάλαιο 1^ο: Το λειτουργικό πρόγραμμα Android

1.1.Ετυμολογία

Το Android είναι ένα λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας (Εξυπνες συσκευές), το οποίο τρέχει τον πυρήνα του λειτουργικού συστήματος Linux· αρχικά, αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance και πρόκειται για ένα προϊόν ελευθέρου λογισμικού.

Η λέξη Android έχει ελληνική ρίζα, καθώς προέρχεται από την λέξη andro, δηλαδή «ανθρώπινη» και τη λέξη eides που σημαίνει «μορφή» (Gargenta, 2011). Επομένως, η έννοια της λέξης android σχετίζεται με το ανδροειδές και συμβολίζεται από ένα ρομπότ



που φέρει ανθρώπινα χαρακτηριστικά (Εικόνα 1.1).

Εικόνα 1. 1: Λογότυπο Android

1.2.Ιστορία του Android

Η εταιρία Android Inc ιδρύθηκε στην Καλιφόρνια των Ηνωμένων Πολιτειών τον Οκτώβριο του 2003 από τον Andy Rubin, τον Rich Miner, τον Nick Sears και τον Chris White. Πρωταρχικός σκοπός τους ήταν να αναπτύξουν το πιο έξυπνο κινητό που θα γνωρίζει ανά πάσα στιγμή τη θέση και τις προτιμήσεις του ιδιοκτήτη του και απώτερος στόχος τους, ήταν να ανταγωνιστούν τα κυρίαρχα λειτουργικά συστήματα που ήδη υπήρχαν στο χώρο των κινητών συσκευών, όπως το Symbian και τα Windows Mobile.

Η Google απέκτησε την Android Inc στις 17 Αυγούστου 2005, κάνοντας την Android Inc εξολοκλήρου θυγατρική της και το σύνολο των κύριων εργαζομένων της τελευταίας, συμπεριλαμβανομένων των Andy Rubin, Rich Miner και Chris White, παρέμειναν στην εταιρεία και μετά την εξαγορά της. Η Android Inc δεν ήταν γνωστή κατά το χρόνο της απόκτησης, αλλά πολλοί υπέθεσαν ότι η Google σχεδίαζε να διεισδύσει στην αγορά της κινητής τηλεφωνίας με αυτή της την κίνηση.

Στις 5 Νοεμβρίου 2007, η Open Handset Alliance, μία κοινοπραξία πολλών εταιρειών που περιλαμβάνουν τις εταιρίες Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, T-Mobile και Texas Instruments αποκαλύφθηκε, με βασικό στόχο της την ανάπτυξη ανοικτών προτύπων για κινητές συσκευές. Την ίδια ημέρα, η Open Handset Alliance παρουσίασε επίσης το πρώτο της προϊόν, το Android, μια κινητή συσκευή πλατφόρμα χτισμένη στον πυρήνα του Linux.

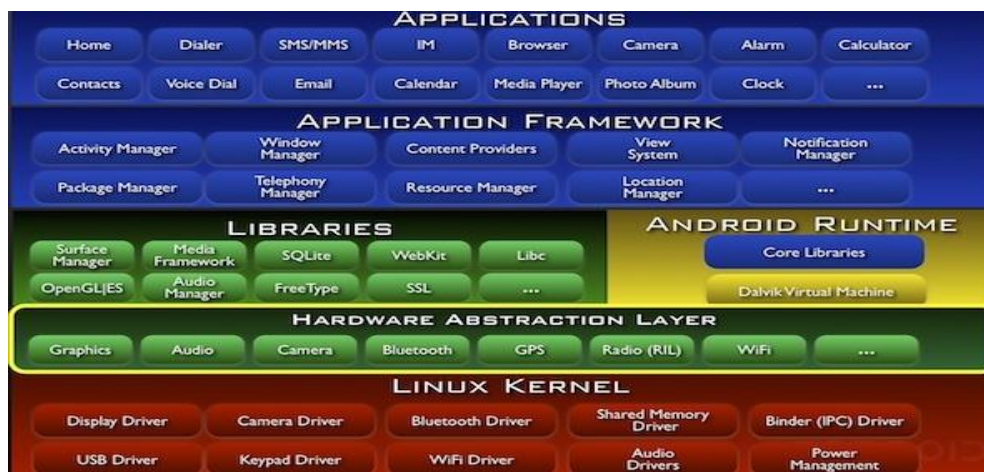
Στις 9 Δεκεμβρίου 2008, 14 νέα μέλη εντάχθηκαν στην Open Handset Alliance, συμπεριλαμβανομένων των εταιριών ARM Holdings, Atheros Communications, ASUSTeK Computer Inc, Garmin Ltd, Huawei Technologies, PacketVideo, Softbank, Sony Ericsson, Toshiba, Vodafone Group, προμηνύοντας τη μεγάλη επιτυχία που αναμενόταν να έχει αυτό το νέο Λειτουργικό Σύστημα. Αρχικά, ο αριθμός των εταιρειών που αποτελούσαν ιδρυτικά μέλη της σύμπραξης Open Handset Alliance ήταν 34, ενώ τώρα αριθμεί 84 μέλη (Εικόνα 1.2).



Εικόνα 1. 2: Μέλη της Open Handset Alliance.

1.3. Αρχιτεκτονική του Android

Το Android δεν είναι απλώς ένα Λειτουργικό Σύστημα. Συνιστά μια στοίβα λογισμικού, η οποία αποτελείται από 4 επίπεδα και 5 ομάδες, τον πυρήνα Linux, τις βιβλιοθήκες, το χρόνο εκτέλεσης και το πλαίσιο εφαρμογής (Εικόνα 1.3) (Lessard &



Εικόνα 1. 3: Τα βασικά περιεχόμενα του λειτουργικού συστήματος Android.

Kessler, 2010).

1.3.1. Πυρήνας Linux (Linux Kernel)

Ο πυρήνας Linux είναι ο πυρήνας του λειτουργικού συστήματος, στο οποίο βασίζεται το Android και βρίσκεται στο χαμηλότερο επίπεδο. Ο συγκεκριμένος πυρήνας υποστηρίζει όλες τις κύριες λειτουργίες του λειτουργικού συστήματος, όπως τη διαχείριση μνήμης και διεργασιών, τις λειτουργίες δικτύου, την ασφάλεια του λειτουργικού συστήματος και ένα σύνολο οδηγών υλικού που χρειάζεται για να τρέξει το σύστημα, όπως π.χ. της οθόνης, της κάμερας, αισθητήρας δαχτυλικών αποτυπωμάτων κ.α.

1.3.2. Βιβλιοθήκες (Libraries)

Στο αμέσως υψηλότερο επίπεδο βρίσκονται οι βιβλιοθήκες, το σύνολο των οποίων είναι γραμμένο σε γλώσσα προγραμματισμού C και C++ και έγινε μεταγλώττισή τους για τη συγκεκριμένη αρχιτεκτονική υλικού που χρησιμοποιείται από τη συσκευή. Οι βιβλιοθήκες δεν είναι εφαρμογές που μπορούν να σταθούν από μόνες τους, αλλά υπάρχουν για να μπορούν να κληθούν από προγράμματα υψηλότερου επιπέδου. Για παράδειγμα, η SQLite βιβλιοθήκη παρέχει υποστήριξη έτσι ώστε μια εφαρμογή να χρησιμοποιεί την αποθήκευση δεδομένων. Επίσης, η Webkit βιβλιοθήκη παρέχει λειτουργίες για τη διαδικτυακή πλοήγηση.

1.3.3. Χρόνος Εκτέλεσης και η εικονική μηχανή Dalvik (Android Runtime and Dalvik Virtual Machine)

Στο ίδιο επίπεδο με τις βιβλιοθήκες βρίσκεται ο χρόνος εκτέλεσης, ένα σύνολο βασικών βιβλιοθηκών που επιτρέπουν στους προγραμματιστές να γράψουν εφαρμογές χρησιμοποιώντας Java. Στη γλώσσα προγραμματισμού Java, υπάρχει η Java Virtual Machine (εικονική μηχανή) στην οποία εκτελείτε ο κώδικας byte code των εφαρμογών. Στο Android υπάρχει μια υλοποίηση της εικονικής μηχανής, η οποία ονομάζεται Dalvik, μέσω της οποίας τρέχουν εφαρμογές του Android, κάθε μία με τη δική της διεργασία και εκτελούνται ταυτόχρονα (Lessard & Kessler, 2010).

Η εικονική μηχανή Dalvik τρέχει αρχεία .dex, τα οποία είναι byte codes που προέρχονται από πηγαίο κώδικα και βιβλιοθήκες της Java, τα οποία ουσιαστικά αποτελούν συμπιεσμένα δεδομένα για εξοικονόμηση χώρου κατά την εκτέλεση. Το Android είναι από τη φύση του multitasking λειτουργικό σύστημα και για αυτό το λόγο επιτρέπει στις εφαρμογές του να τρέχουν σε πολλά νήματα ταυτόχρονα, απασχολώντας πολλές

διαδικασίες εάν αυτό κρίνεται αναγκαίο. Για να γίνει αυτό εφικτό, η μηχανή Dalvik είναι σχεδιασμένη έτσι ώστε να έχει ελάχιστο αντίκτυπο στη χρήση της μνήμης. Χάρη στον λιτό της σχεδιασμό, το σύστημα είναι σε θέση να τρέχει πολλές εικονικές μηχανές ταυτόχρονα.

1.3.4. Πλαίσιο Εφαρμογής (Application Framework)

Στο αμέσως πιο πάνω επίπεδο από τις βιβλιοθήκες και το χρόνο εκτέλεσης Android, βρίσκεται το πλαίσιο εφαρμογής, το οποίο και εκθέτει διάφορες δυνατότητες του Android στους προγραμματιστές των εφαρμογών. Το πλαίσιο εφαρμογής είναι προεγκαταστημένο στο Android, αλλά είναι επεκτάσιμο, αφού ο κάθε κατασκευαστής-προγραμματιστής έχει τη δυνατότητα να το συμπληρώσει με δικά του κομμάτια. Τα σημαντικότερα δομικά στοιχεία είναι τα ακόλουθα (Lessard & Kessler, 2010):

- Διαχειριστής δραστηριοτήτων – Activity Manager: Υπεύθυνο για τον έλεγχο του χρόνου ζωής των εφαρμογών και για τη διατήρηση μιας στοίβας, που επιτρέπει την πλοήγηση του χρήστη σε προηγμένες οθόνες.
- Παροχέας Περιεχομένου – Content Providers: Αυτά τα αντικείμενα περιέχουν δεδομένα που μπορούν να διαμοιραστούν μεταξύ εφαρμογών.
- Διαχειριστής Πόρων - Resource Manager: Οι πόροι είναι οτιδήποτε υπάρχει σε ένα πρόγραμμα και δεν πρόκειται για κώδικα. Για παράδειγμα, μπορεί να είναι κωδικοί χρημάτων, αλφαριθμητικοί χαρακτήρες ή ακόμα και έτοιμα σχεδιαγράμματα οθονών φτιαγμένα σε XML, τα οποία μπορεί το πρόγραμμα να καλεί.
- Διαχειριστής Τοποθεσίας – Location Manager: Χρησιμοποιείται στο πλαίσιο της εφαρμογής για να αναγνωρίζει ανά πάσα στιγμή που βρίσκεται η συσκευή.
- Διαχειριστής Κοινοποιήσεων – Notification Manager: Ιδανικός τρόπος για να ενημερώνεται ο χρήστης για γεγονότα που συμβαίνουν, διακριτικά, χωρίς να διακόπτεται η εργασία του.

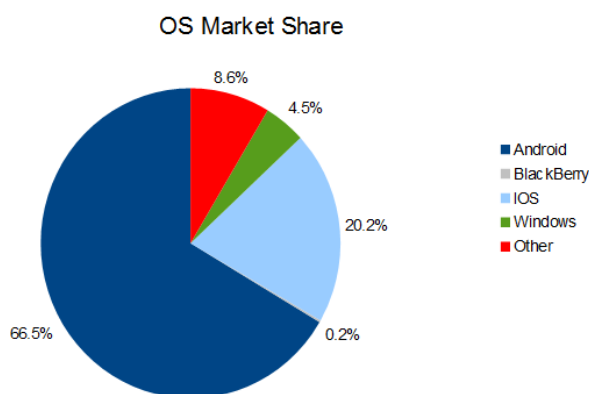
1.3.5. Εφαρμογές (Applications)

Στο πιο υψηλό επίπεδο βρίσκονται οι εφαρμογές που καταλαμβάνουν ολόκληρη την οθόνη και αλληλεπιδρούν με το χρήστη. Οι εφαρμογές αυτές είναι προγράμματα τα οποία βλέπουν οι χρήστες, χωρίς να γνωρίζουν την υπόλοιπη στοιβή του λειτουργικού. Μερικές εφαρμογές είναι προ-εγκατεστημένες με κάθε Android συσκευή, όπως για παράδειγμα επαφές, μουσική, ρολόι κ.α., ενώ τις περισσότερες τις εγκαθιστά ο χρήστης ανάλογα με τις ανάγκες του.

Για τη διευκόλυνση της διαδικασίας εύρεσης και εγκατάστασης εφαρμογών, σε κάθε Android συσκευή υπάρχει διαθέσιμη και προ-εγκαταστημένη εφαρμογή Google Play. Αυτή η υπηρεσία – εφαρμογή επιτρέπει στους χρήστες να κατεβάζουν εφαρμογές δωρεάν αλλά και επί πληρωμή που έχουν δημιουργηθεί από διάφορους προγραμματιστές.

1.4.Πωλήσεις

Το 2013 και μόλις έξι χρόνια μετά την πρώτη επίσημη παρουσίαση της πλατφόρμας Android, προς έκπληξη πολλών, το Android, έχει ξεπεράσει σε πωλήσεις τον κύριο αντίπαλο του iPhone που λειτουργεί με IOS της Mac OS· πλέον το Android OS κυριαρχεί για πρώτη φορά με ποσοστό που φτάνει το 66,5% των Smartphone's παγκοσμίως με το IOS να έρχεται δεύτερο με ποσοστό 20,2%. Όπως φαίνεται και στην



Γράφημα 1. 1: Πωλήσεις Λειτουργικών Συστημάτων για Smartphone 2013.

παρακάτω εικόνα (Γράφημα 1.4) οι πωλήσεις του Android OS ανεβαίνουν με ραγδαίους ρυθμούς.

1.5. Εκδόσεις και χαρακτηριστικά

Το Android κυκλοφορεί σε διάφορες εκδόσεις με ονομασίες που ανοίγουν την όρεξη. Παλαιότερες εκδόσεις Android Cup Cake (1.5), Donut (1.6), Éclair (2.0, 2.1), Ginger Bread (2.3), Fro Yo (2.2), Honeycomb (3.0, 3.1, 3.2), Ice Cream Sandwich (4.0), Jelly Bean (4.1), KitKat (4.4), Lollipop (5.0, 5.1), Marshmallow (6.0) και νέες εκδόσεις Nougat (7.0,7.1), Oreo (8.0, 8.1), Pie (9.0) (Android, 2019). Από την παρθενική έκδοση Android 1.0, η οποία κυκλοφόρησε το Σεπτέμβριο του 2008 μέχρι την αμέσως επόμενη έκδοση 1.1 που παρουσιάστηκε το Φεβρουάριο του 2009, χρειάστηκε ένας χρόνος για να γίνει η έκρηξη των καινοτόμων εκδόσεων και των σημαντικών αλλαγών που επέφεραν ριζικές αλλαγές για τον κάθε χρήστη.

Εδώ, αξίζει να αναφερθεί ότι το πρώτο smartphone που είχε λογισμικό Android ήταν το T-Mobile G1 κατασκευασμένο από την HTC με οθόνης αφής TFT-LCD 3.2 ίντσες, full qwerty πληκτρολόγιο. Είχε πρόσβαση σε Gmail, YouTube, Google maps, calendar, κάμερα 3.2MP (megapixel) με αυτόματη εστίαση και κάρτα μνήμης micro SD.

1.5.1. Παλιότερες εκδόσεις Android.

- **Android σε κινητά τηλέφωνα.**

Πιο συγκεκριμένα, μελετώντας τις εκδόσεις Android CupCake 1.5 (Εικόνα 1.5) και Donut 1.6 (Εικόνα 1.6), επιβάλλεται να αναφερθούν και τα πιο καινοτόμα χαρακτηριστικά της κάθε έκδοσης. Δημοσιευμένες και οι δύο το 2009 και βασισμένες σε πυρήνα Linux, είναι οι εκδόσεις που εισάγουν και τα πιο χαρακτηριστικά γνωρίσματα του λογισμικού.

Από την μία πλευρά, η πρώτη, εισάγει κινούμενες μεταβάσεις οθόνης και υποστήριξη Widget στην αρχική οθόνη, δυνατότητα πρόβλεψης κειμένου και προσαρμοσμένων λέξεων, εναλλαγή κάθετης και οριζόντιας διάταξης, δυνατότητα καταγραφής και αναπαραγωγής βίντεο σε μορφή MPEG-4 και 3GP αλλά και μέσα από τη λειτουργία της κάμερας, μεταμόρφωση βίντεο στο YouTube και φωτογραφιών στο Picasa απευθείας από τη συσκευή αλλά και υποστήριξη προτύπου Bluetooth A2DP και AVRCP και ικανότητα αυτόματης σύνδεσης σε μικροσυσκευή Bluetooth.

Από την άλλη πλευρά, η δεύτερη έκδοση εισάγει την πολλαπλή αφή στην οθόνη, βελτιώνει την αναζήτηση, εισάγοντας την εύρεση σελιδοδεικτών και επαφών από την αρχική οθόνη, υποστηρίζει την ανάλυση οθονών WVGA, τις τεχνολογίες για

CDMA/EVDO, VPNs, text-to-speech και Virtual Private Networks, ανανεώνει την φωνητική αναζήτηση, όπου και συμπεριλαμβάνεται η δυνατότητα κλήσης επαφών, βελτιώνει την υπηρεσία πλοήγησης Google turn-by-turn αλλά και το Android Market.

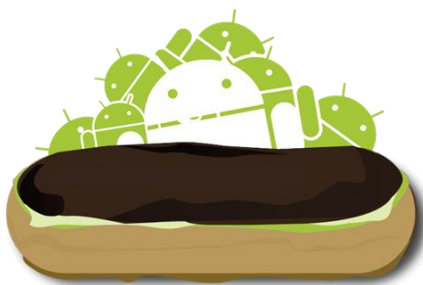


Εικόνα 1. 5: Λογότυπο Android Cup Cake



Εικόνα 1. 4: Λογότυπο Android Donut.

Βασισμένη στον ίδιο πυρήνα Linux της έκδοσης Donut, τον 2.6.29, οι εκδόσεις Éclair 2.0, 2.1 (Εικόνα 1.7), παρουσιάζουν βελτιώσεις στην ταχύτητα, στη διεπιφάνεια χρήστη, στις υπηρεσίες χαρτών της Google, υποστηρίζουν την έκδοση 2.1 του πρωτοκόλλου Bluetooth και όλα τα νέα χαρακτηριστικά του καθώς και το πρότυπο HTML 5. Ακόμα, παρουσιάζουν περισσότερες οθόνες και αναλύσεις. Το 2010, ανεπτυγμένη στον πυρήνα Linux 2.6.32, η έκδοση Fro Yo 2.2 (Εικόνα 1.8), παρουσιάζει βελτιώσεις που σχετίζονται με τον χειρισμό της κάμερας και του flash, με το Bluetooth και το πληκτρολόγιο, που τώρα εμφανίζεται ως πολυγλωσσικό και εισάγει νέα Widgets.



Εικόνα 1. 7: Λογότυπο Android Éclair.



Εικόνα 1. 6: Λογότυπο Android Fro Yo.

Ωστόσο, με ποσοστό που φτάνει το 55%, η Gingerbread του 2010 (Εικόνα 1.9), αποτελεί την πιο διαδεδομένη έκδοση Android. Και αυτό γιατί παρουσιάζει πολλές



Εικόνα 1. 8: Λογότυπο Android Gingerbread.

βελτιώσεις όσον αφορά τις λειτουργίες της κάμερας, όπως είναι η βίντεο-κλήση, η υποστήριξη των μπροστά και πίσω καμερών αλλά και του WebM για την αναπαραγωγή βίντεο. Επιπλέον χαρακτηριστικά της είναι η βελτιωμένη διαχείριση ενέργειας, η επιλογή, αντιγραφή και επικόλληση με ένα άγγιγμα αλλά και η υποστήριξη για πληθώρα αισθητήρων (όπως είναι το βαρόμετρο κ.α.).

- **Android σε tablet και κινητά τηλέφωνα.**

Η επόμενη Android έκδοση, η Honeycomb (Εικόνα 1.10), σχεδιασμένη το 2011 και βασισμένη στον Linux 2.6.36, προοριζόταν μόνο για tablet. Εξαιτίας της ιδιαιτερότητάς της και του μεγάλου της συσκευής εφαρμογής της, τα κύρια χαρακτηριστικά σχετίζονται με τον επανασχεδιασμό του γραφικού περιβάλλοντος, την υποστήριξη για φυσικά πληκτρολόγια, 2D και 3D γραφικά και επιλογή κειμένου αλλά και



τον επανασχεδιασμό του εικονικού πληκτρολογίου.

Εικόνα 1. 9: Λογότυπο Android Honeycomb.

Αντίθετα με την αποκλειστικότητα της Honeycomb, με την έκδοση Ice Cream Sandwich (Εικόνα 1.11), το 2011, επιτυγχάνεται η εφαρμογή του λογισμικού Android και σε tablet και σε κινητά τηλέφωνα. Ανεπτυγμένη στον πυρήνα Linux 3.0.1, εμφανίζει χαρακτηριστικά που αναφέρονται σε βελτιώσεις υποστήριξης των υπηρεσιών κοινωνικής δικτύωσης, δυνατότητας εναλλαγής σε ανοιχτές εφαρμογές και μεγέθους στα Widgets, αναγνώρισης προσώπου για ξεκλείδωμα αλλά και δυνατότητας χειρισμού των δεδομένων δικτύων.

Ως αποτέλεσμα των παραπάνω βελτιώσεων, που απευθύνονται περισσότερο στην εμπειρία του χρήστη, προέκυψε και η έκδοση Android Jelly Bean (Εικόνα 1.12), το 2012, σχεδιασμένη με βάση τον Linux kernel 3.4. Εδώ, υφίστανται βελτιώσεις όπως πρόβλεψη



Εικόνα 1. 10: Λογότυπο Android Ice Cream Sandwich.

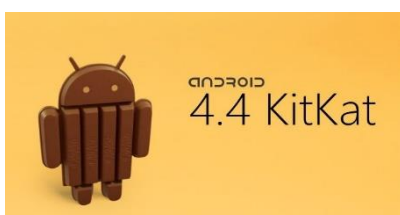


Εικόνα 1. 11: Λογότυπο Android Jelly Bean.

θέσης δαχτύλων, Triple Buffering, εκτεταμένο vsync timing, εργαλείο Systrace, cpu input boost, πολυκάναλος ήχος και υποστήριξη κωδικοποίησης/αποκωδικοποίησης ήχου AAC 5.1 και έξοδος USB Audio. Αυτή η έκδοση παρουσιάζει ενισχυμένη προσιτότητα, αφού εστιάζει στο που τοποθετεί ο χρήστης το χέρι του καθώς εκτελεί εντολές στην οθόνη αφής και αυξημένη υποστήριξη πολλών γλώσσών. Οι χρήστες μπορούν να χρησιμοποιούν εναλλακτικά πληκτρολόγια ή και να δημιουργούν τις δικές τους παραλλαγές και υπάρχουν επιπλέον νέοι τρόποι να εμφανίζεται το User Interface με μεγαλύτερες ειδοποιήσεις, υποστήριξη περισσότερων μορφών περιεχομένου στο Notifications bar, με τις εφαρμογές (apps) να μπορούν να συνοδευτούν από μία έως και τρεις διαφορετικές ειδοποιήσεις. Ακόμα, από την άλλη πλευρά, οι developers μπορούν να χρησιμοποιήσουν δυο κατευθύνσεις γραφής, ώστε να μην υπάρχουν περιορισμοί στις γλώσσες που μπορούν να υποστηρίξουν οι εφαρμογές.

Η έκδοση του 2013, η Kit Kat 4.0 (Εικόνα 1.13), παρουσιάζει κυρίως βελτιώσεις στο λειτουργικό σύστημα και στις επιδόσεις του λογισμικού σε entry-level συσκευές με περιορισμένους πόρους. Οι βελτιώσεις που παρουσιάζονται σχετίζονται με νέες μορφές συνδεσιμότητας και δυνατότητες NFC με Host Card Emulation, με easy animations for Activity Launch και νέους τρόπους μετάβασης, με νέα APIs και με βελτιώσεις στην φωνητική υπηρεσία αναζήτησης της Google.

Στο ίδιο μήκος κύματος κινήθηκε και η έκδοση του 2014, η Lollipop 5.1 (Εικόνα 1.14), με νέες λειτουργίες για χρήστες και χιλιάδες νέα API για προγραμματιστές, επεκτείνοντας το λογισμικό Android ακόμη περισσότερο σε tablet, τηλεοράσεις και αυτοκίνητα. Ως αποτέλεσμα των προηγούμενων, η έκδοση αυτή χαρακτηρίζεται από επανασχεδιασμένο περιβάλλον εργασίας του χρήστη χτισμένο πάνω σε μια σχεδιαστική γλώσσα, τη Material Design, από βελτιωμένη κάμερα και βίντεο και νέες δυνατότητες πολυμέσων, από γραφικά υψηλής απόδοσης με το Khronos Open GL ES 3.1, από βέλτιστη διάρκεια ζωής της μπαταρίας και από βελτιωμένη απόδοση των εφαρμογών με αντικατάσταση του Dalvik με το Android Runtime (ART).



Εικόνα 1. 12: Λογότυπο Android Kit Kat.



Εικόνα 1. 13: Λογότυπο Android Lollipop.

1.5.2. Android 6.0 Marshmallow

Σε σύγκριση με την Lollipop, τον Οκτώβριο του 2015 κυκλοφόρησε η έκδοση Marshmallow 6.0 (Εικόνα 1.15), η οποία και επικεντρώνεται κυρίως στη βελτίωση της συνολικής εμπειρίας του χρήστη. Η επίτευξη της τελευταίας κατέστη δυνατή με την εισαγωγή μιας νέας αρχιτεκτονικής για τα δικαιώματα των εφαρμογών, την υποστήριξη για την αναγνώριση δακτυλικών αποτυπωμάτων και την εισαγωγή USB θύρα τύπου –C (type C). Επιπλέον, διέθετε ένα νέο σύστημα διαχείρισης ενέργειας μείωσης της δραστηριότητας του παρασκηνίου όταν μια εφαρμογή δεν χρησιμοποιείται και εισήχθησαν



Εικόνα 1. 14: Λογότυπο Android Marshmallow

νέα APIs με νέο χαρακτηριστικό “Now on Tap”.

1.5.3. Android 7.1 Nougat.

Τον αμέσως επόμενο χρόνο, βγαίνει στην αγορά η έκδοση 7.1 Nougat (Εικόνα 1.16), με σημαντικές αλλαγές στο λειτουργικό σύστημα και την πλατφόρμα ανάπτυξης. Οι αλλαγές που σημειώθηκαν αναφέρονται στην εμφάνιση multi-window UI, σε περιβάλλον java βασισμένο στο Open JDK, στη βελτίωση γραφικών Vulkan API αλλά και σε



Εικόνα 1. 15: Λογότυπο Android Nougat.

ειδοποιήσεις άμεσης απάντησης.

1.5.4. Android 8.0 Oreo

Το 2017 κυκλοφόρησε η έκδοση 8.0 με ονομασία Oreo και κωδική ονομασία O (Εικόνα 1.17). Η έκδοση αυτή προσφέρει πολλές νέες δυνατότητες για να επεκτείνει ο χρήστης την εφαρμογή και να αναπτυχθεί αποτελεσματικά. Από τις νέες δυνατότητες αξίζει να σημειωθούν η αυτόματη προσαρμογή του Text View, τα χαρακτηριστικά Web



View, η υποστήριξη πολλαπλών εμφανίσεων, τα API νευρωνικών δικτύων, οι γραμματοσειρές με δυνατότητα λήψης, το πλαίσιο αυτόματης συμπλήρωσης, το Wide-gamut color, τα Media features και το face id.

Εικόνα 1. 16: Λογότυπο Android Oreo.

1.5.5. Android 9.0 Pie

Το νέο λογισμικό Android 9.0 έχει την ονομασία Pie (Εικόνα 1.18). Κυκλοφόρησε επίσημα τον Αύγουστο του 2018 μετά από διάφορες δοκιμαστικές εκδόσεις (Beta). Η παραπάνω έκδοση είναι διαθέσιμη για συσκευές Google Pixel και Essential Phone και τα πιο σημαντικά χαρακτηριστικά της είναι τα παρακάτω (Android, 2019· Yi, Jia & Saniie, 2012):

- Το ποσοστό της μπαταρίας εμφανίζεται πάντα στην οθόνη.
- DNS μέσω TLS.
- Ένα κουμπί “Στιγμιότυπο” έχει προστεθεί στις επιλογές ενέργειας.
- Ενημερώσεις ειδοποιήσεων και έξυπνη απάντηση με τυποποιημένα μηνύματα.
- Υποστήριξη HEIF IMAGE COMPRESSION.
- Security enhancements (Android Protected Confirmation).
- Screen Rotation Button.
- Well Being (Χρόνος χρήσης μιας εφαρμογής).

- Βελτιωμένο χαρακτηριστικό προσαρμοστικής φωτεινότητας το οποίο τροποποιεί τη φωτεινότητα της οθόνης με βάση τις προσωπικές προτιμήσεις.



Εικόνα 1. 17: Λογότυπο Android Pie.

2. Κεφάλαιο 2^ο: Gamification

Ο όρος gamification ή αλλιώς «παιχνιδοποίηση», όπως και αποδίδεται στην ελληνική ορολογία, είναι συνυφασμένος με την αξιοποίηση στοιχείων, γνωρισμάτων και τεχνικών σχεδίασης παιχνιδιών σε εφαρμογές που δεν εντάσσονται στα παιχνίδια καθαυτά γι' αυτό και είναι σημαντικό να μην ταυτίζονται ο όρος Gamification με τον όρο Game. Πληθώρα ερευνών έχει αποδείξει πως η ενσωμάτωση μηχανισμών παιχνιδιού έχει θετικό αντίκτυπο στα άτομα. Βασικά χαρακτηριστικά ενός gamification αποτελούν οι πόντοι (points), τους οποίους κερδίζει ο εκάστοτε παίκτης μετά την επιτυχή ολοκλήρωση μιας διαδικασίας, οι πίνακες κατάταξης των παικτών (leaderboards) που αποτελούνται από το σύνολο των παικτών οι οποίοι κάνουν χρήση της εφαρμογής και τέλος, από τα σήματα (badges) που κερδίζει ο κάθε παίκτης αφού περάσει επιτυχώς την αντίστοιχη δοκιμασία που έχει θέσει ο σχεδιαστής για το κάθε επίπεδο της δοκιμασίας (Faiella & Ricciardi, 2015).

Ο όρος gamification δύσκολα διαχωρίζεται από άλλους όρους που έχουν σχέση με το παιχνίδι (Εικόνα 2.1). Παρ' όλα αυτά, ο διαχωρισμός γίνεται εύκολα εάν εξεταστούν ορισμένα βασικά χαρακτηριστικά. Το παιχνίδι ως πράξη (game) όπως παρουσιάζεται και στην επόμενη εικόνα, αλλά και το παιχνίδι ως αντικείμενο (toy) βασίζονται ολοκληρωτικά στην έννοια του παιχνιδιού. Στην περίπτωση του gamification χρησιμοποιούνται μόνο μερικά μέρη και ορισμένα στοιχεία από την έννοια του παιχνιδιού, ωστόσο, το βασικό σύστημα είναι ανεξάρτητο από το παιχνίδι και χρησιμοποιούνται μηχανισμοί παιχνιδιού για να υποβοηθήσουν τη γενικότερη λειτουργία και να επιτευχθούν ευκολότερα οι στόχοι που έχουν τεθεί. Το ίδιο ισχύει και για συστήματα που χρησιμοποιούν ένα σχέδιο παιχνιδιού με μοναδικό στόχο να έχουν μία πιο ευχάριστη εμφάνιση για το χρήστη έτσι ώστε να τον προσελκύσουν ευκολότερα. Ωστόσο, και σε αυτή την περίπτωση χρησιμοποιούνται αποκλειστικά ορισμένα στοιχεία ενός κανονικού παιχνιδιού. Οι διακρίσεις μεταξύ των διαφορετικών όρων παρουσιάζονται ακολούθως (Vinichenko,



Εικόνα 2. 1: Διάκριση του gamification ανάμεσα σε παρεμφερείς όρους σχετικούς με το gaming.

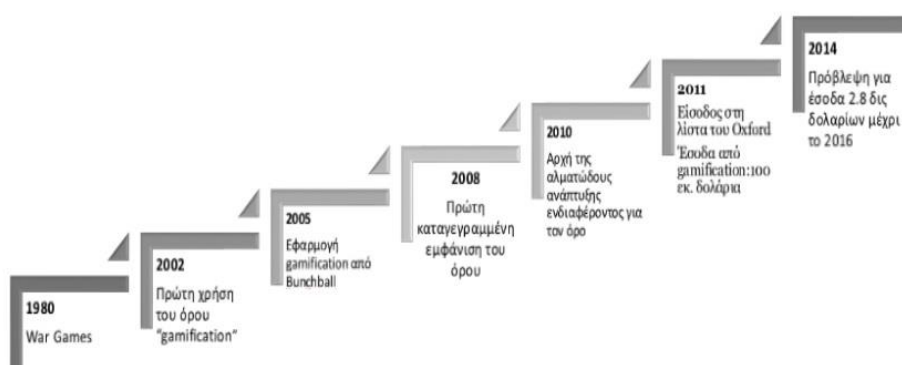
Melnichuk, Kirillov, Makushkin & Melnichul, 2016).

Η φιλοσοφία του gamification επικεντρώνεται στην παροχή κινήτρων που στοχεύουν στη δημιουργία ενός διαδραστικού περιβάλλοντος, το οποίο θα δελεάσει τον εκάστοτε χρήστη στηριζόμενο στις προτιμήσεις του, θα εντείνει το ενδιαφέρον του και θα τον κρατήσει αφοσιωμένο στο στόχο που έχει τεθεί από το δημιουργό της εφαρμογής. Επιπλέον, σημαντικό στοιχείο του gamification είναι η ενεργή εμπλοκή του χρήστη μέσω της διαρκούς επιβράβευσής του, προκειμένου να καλυφθεί μια πιο εσωτερική ανάγκη για αναγνώριση της προσπάθειάς του.

Το gamification δεν έχει στόχο να κάνει πιο ευχάριστες «ανιαρές» διαδικασίες και να τις επενδύσει με στοιχεία παιχνιδιών, αλλά να βοηθήσει στην ευθυγράμμιση των στόχων εταιριών με τους στόχους των πελατών τους. Για παράδειγμα, δεν αποτελεί gamification η επιλογή να χρησιμοποιεί ένας διευθυντής ένα παιχνίδι προκειμένου να αξιολογήσει τους υπαλλήλους του. Η περιγραφή στόχων όμως που επιτρέπουν στους υπαλλήλους την επίτευξη στόχων (π.χ. πωλήσεις), αλλά και τη συνεργασία εσωτερικά (π.χ. συλλογή πόντων από πετυχημένους πωλητές αν προσφέρουν υποστήριξη σε λιγότερο πετυχημένους συναδέλφους) αποτελούν ορθά παραδείγματα gamification στις επιχειρήσεις. Ως καινοτόμο εργαλείο, αξιοποιείται ως επί το πλείστον από πληθώρα εταιρειών και επιχειρήσεων που εδράζονται στο Δυτικό κόσμο, με απώτερο στόχο την εκπαίδευση των υπαλλήλων, τη διαρκή επιμόρφωση, την παροχή ερεθισμάτων και κινήτρων, που θα συμβάλλουν καθοριστικά στην αύξηση της αποδοτικότητας στον εργασιακό χώρο και την επακόλουθη επιβράβευση των εργαζομένων μετά την αποπεράτωση των καθημερινών εργασιακών τους καθηκόντων. Πιο συγκεκριμένα, οι χρήσεις του gamification δε συσχετίζονται αποκλειστικά με το χώρο του marketing και των επιχειρήσεων, αλλά επεκτείνονται στον τομέα της εκπαίδευσης και της μάθησης (Jonshon, Adams & Cummins, 2012), της εκμάθησης ξένων γλωσσών με έναν πιο διασκεδαστικό τρόπο, της συνεχούς επιμόρφωσης των εργαζομένων στα πλαίσια της εταιρικής κατάρτισης και τέλος στον κλάδο της διασκέδασης. Αναφορικά με τον τομέα των επιχειρήσεων και του marketing, συνιστά ένα δημιουργικό εργαλείο που ενισχύει την εταιρική στρατηγική, αυξάνει τα ποσοστά των πωλήσεων, βελτιώνει την αποδοτικότητα των εργαζομένων και την εσωτερική επικοινωνία μεταξύ τους, αυξάνει το αίσθημα της αυτοεκτίμησης και την παραγωγικότητά τους. Όλα τα παραπάνω, επιτυγχάνονται μέσω της επιβράβευσής που δίνεται στο χρήστη για τη νίκη του σε κάθε επίπεδο του παιχνιδιού.

2.1 Ιστορία gamification

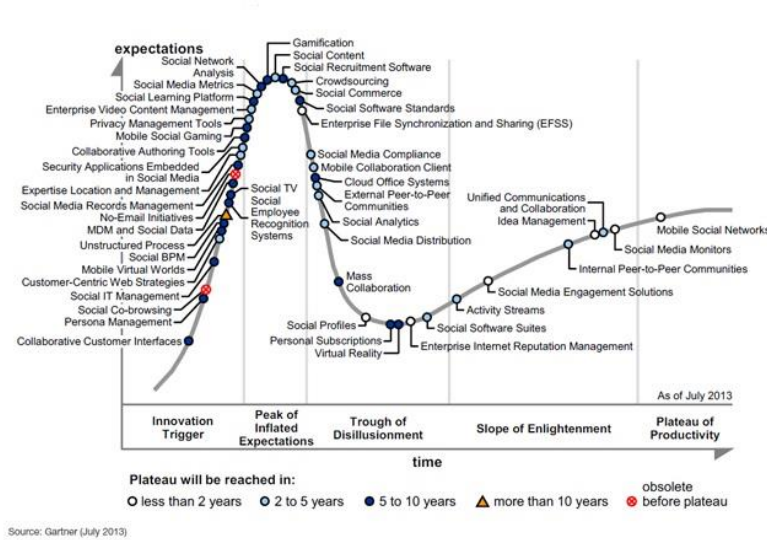
Από τη δεκαετία του 1980 το Hollywood ξεκίνησε να ασχολείται με την έννοια του gamification μέσω της ταινίας War Games. Η έννοια του gamification αρχίζει να απασχολεί αρκετούς μελετητές και να αποτελεί κεντρικό θέμα για βιβλία, πρωτίστως όμως για το ρόλο που διαδραματίζει το gamification αναφορικά με την εκπαίδευση. Το gamification σαν όρος έχει τις ρίζες του στην ψηφιακή βιομηχανία. Η πρώτη καταγεγραμμένη παρουσία του χρονολογείται το 2008, ενώ πιθανολογείται ότι έχει ακόμα βαθύτερες ρίζες, όταν το 2002 ο Nick Pelling το αξιοποίησε ώστε να περιγράψει τον τρόπο με τον οποίο η εργασία του ως σύμβουλος, ήταν εφικτό να γίνει πιο διασκεδαστική. Το 2005, τα πρώτα συστήματα gamification εφαρμόζονται από την αμερικανική εταιρία Bunchball. Η εν λόγω εταιρία παρείχε πλατφόρμα gamification με έτοιμα στοιχεία, όπως πόντους και πίνακες κατάταξης. Το 2010 και μετά έγινε αντιληπτή η δύναμη του καινούριου αυτού εργαλείου, το οποίο και γίνεται φανερό από την αλματώδη ανάπτυξη ενδιαφέροντος για τον συγκεκριμένο όρο στη μηχανή αναζήτησης Google. Το 2011, ο όρος Gamification εντάχθηκε στη λίστα του Oxford για τη λέξη της χρονιάς και τα έσοδα παγκοσμίως από την εφαρμογή του gamification ανέρχονται στα 100 εκατομμύρια δολάρια. Στην παρακάτω εικόνα (Εικόνα 20) παρουσιάζεται συνοπτικά μία ιστορική αναδρομή του όρου από το 1980 μέχρι τις προβλέψεις που έγιναν για αυτόν το 2014 (Εικόνα 2.2) (Kamasheva, Valeev, Yagudin & Maksimova, 2015 · Unger, Schwartz &



Εικόνα 2. 2: Ιστορική αναδρομή Gamification.

Foucher, 2013).

Η βιομηχανία του Gamification γνωρίζει τεράστια ανάπτυξη από το 2014 και έπειτα, ενώ η αξία της μέχρι και σήμερα υπολογίζεται πάνω από 5.5 δισεκατομμύρια δολάρια. Στην παρακάτω εικόνα, δίνεται ο “Κύκλος του Gartner” (Εικόνα 2.3), ο οποίος παρουσιάζει την εξέλιξη των νέων τεχνολογιών. Η συγκεκριμένη εκδοχή του Κύκλου του



Εικόνα 2. 3: Κύκλος Gartner, Ιούλιος 2013.

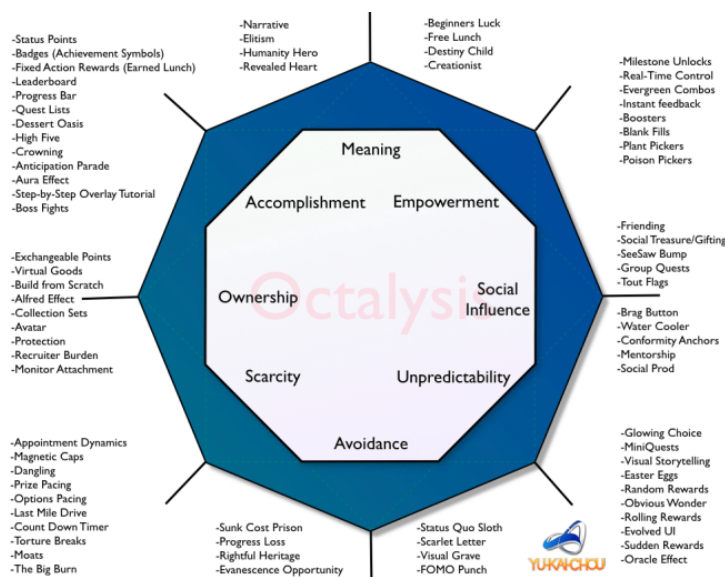
Gartner είναι μια ανανεωμένη εικόνα, όπως διαμορφώθηκε τον Ιούλιο του 2013 και αναφέρει πλέον ότι το gamification παύει να συνιστά έναν όρο πειραματισμού, καθώς αρχίζει πλέον να βρίσκει πρακτικές εφαρμογές με νόημα στις επιχειρήσεις (Xu, Weber & Buhalis, 2014· Hamari, Koivisto & Sarsa, 2014).

2.2. Τεχνικές Ανάπτυξης Gamification.

Το Gamification συνιστά την τέχνη στην οποία συγκεντρώνονται όλα τα διασκεδαστικά και δεσμευτικά στοιχεία που βρίσκονται στα παιχνίδια και εφαρμόζονται σε πραγματικές ή παραγωγικές δραστηριότητες. Η διαδικασία αυτή ονομάζεται κατά τον Yu-kai Chou “Σχεδίαση Εστιασμένη στον Άνθρωπο” σε αντίθεση με τη “Σχεδίαση που είναι Εστιασμένη στις Λειτουργίες”. Πρόκειται για μια διαδικασία σχεδιασμού που βελτιστοποιεί το ανθρώπινο κίνητρο, σε αντίθεση με την καθαρή απόδοση. Τα περισσότερα συστήματα επικεντρώνονται στις λειτουργίες και είναι σχεδιασμένα για γρήγορη εκτέλεση μιας λειτουργίας. Σύμφωνα με τον Yu-kai Chou (Yu-kai, 2019) είναι σαν ένα εργοστάσιο που θεωρεί πως οι εργαζόμενοι του θα κάνουν τις δουλειές τους

επειδή είναι υποχρεωμένοι. Ωστόσο, ο «Σχεδιασμός που είναι Εστιασμένος στον Άνθρωπο» θεωρεί πως ο άνθρωπος που βρίσκεται σε ένα σύστημα έχει συναισθήματα, ανασφάλειες και λόγους για τους οποίους θέλει ή δεν θέλει να κάνει κάποια πράγματα, ως αποτέλεσμα να βελτιώνει τα συναισθήματα, τα κίνητρα και τις δεσμεύσεις του. Ο λόγος που όλο αυτό αποκαλείται Gamification είναι επειδή η βιομηχανία παιχνιδιών ήταν η πρώτη που υλοποίησε τη Σχεδίαση που είναι Εστιασμένη στον Άνθρωπο.

Τα παιχνίδια έχουν ως απώτερο σκοπό να τέρψουν και να ψυχαγωγήσουν τον παίκτη-χρήστη, καθώς με το πέρασμα των ετών μαθαίνουμε μέσα από αυτά πώς να κυριαρχήσουμε στα κίνητρα και την αφοσίωση των παικτών. Έτσι λοιπόν, ο Yu-kai Chou μέσα από ενδελεχή έρευνα και μελέτη προσπάθησε να αναλύσει και να οικοδομήσει στρατηγικές βασισμένες σε διάφορα συστήματα που καθιστούν ένα παιχνίδι διασκεδαστικό, με σκοπό να διαμορφώσει ένα πλήρες πλαίσιο. Το τελικό αυτό πλαίσιο του Gamification ονομάζεται Οκτάλυση (Octalysis), βασίζεται σε ένα σχήμα Οκταγώνου με 8 βασικές μονάδες (8 κατηγορίες) καθεμία από τις οποίες αντιπροσωπεύουν την κάθε πλευρά του εαυτού μας, που μας παρακινεί σε ορισμένες δραστηριότητες μέσα σε ένα



Εικόνα 2. 4: Η Οκτάλυση του Yu-kai Chou.

παιχνίδι (Εικόνα 2.4) (Yu-kai, 2016).

2.3. Οι 8 Βασικές Μονάδες του Gamification(Οκτάλυση)

- **Επική σημασία και κάλεσμα (Epic Meaning and Calling)**

Είναι η βασική κατηγορία όπου ο παίκτης – χρήστης θεωρεί πως κάνει κάτι καλύτερα από ότι πίστευε ή ότι είναι ο εκλεκτός για κάτι που κάνει. Ένα από τα χαρακτηριστικά αυτού του παίκτη είναι ότι αφιερώνει πολύ χρόνο στη διαχείριση ενός φόρουμ, είτε συμβάλει στη δημιουργία πραγμάτων για ολόκληρη την κοινότητα (π.χ. Wikipedia, Open Source). Επίσης, ο παίκτης αυτός θεωρεί πως έχει κάποιου είδους θείο δώρο, σε αντίθεση με άλλους παίκτες, ενώ ουσιαστικά έχει την τύχη του πρωτάρη. Ένα τέτοιο αντιπροσωπευτικό παράδειγμα είναι η εύρεση ενός σπάνιου αντικειμένου στην αρχή του παιχνιδιού.

- **Εξέλιξη και Επίτευξη (Development and Accomplishment)**

Είναι η κατηγορία στην οποία γίνεται εσωτερική προσπάθεια να αναπτυχθούν δεξιότητες, να σημειωθεί πρόοδος και τελικά να ξεπεραστούν οι προκλήσεις. Η έννοια της πρόκλησης είναι πολύ σημαντική, διότι χωρίς αυτή δεν υπάρχει νόημα κατάκτησης ενός τρόπαιου ή ενός εμβλήματος. Σε αυτή τη κατηγορία εμφανίζονται τα περισσότερα αντικείμενα επιβράβευσης όπως τρόπαια, εμβλήματα και βαθμολογικοί πίνακες.

- **Ενδυνάμωση της Δημιουργικότητας και Ανάδραση (Empowerment og Creativity and Feedback)**

Είναι η κατηγορία στην οποία ο χρήστης ασχολείται με μια δημιουργική διαδικασία, όπου πρέπει να υπολογίζει επανειλημμένα τα πράγματα και να δοκιμάζει διαφορετικούς συνδυασμούς. Οι άνθρωποι όχι μόνο χρειάζονται τρόπους να εκφράσουν τη δημιουργικότητά τους, αλλά πρέπει συγχρόνως να μπορούν να δουν τα αποτελέσματά της, να λάβουν σχόλια και να μπορέσουν να ανταποκριθούν ανάλογα. Ένα στοχευμένο παράδειγμα, είναι η ενασχόληση με τα Lego ή τη ζωγραφική που ψυχαγωγούν τους χρήστες σε τέτοιο βαθμό ώστε οι σχεδιαστές των παιχνιδιών δε χρειάζεται να προσθέσουν επιπρόσθετο περιεχόμενο για να καταστήσουν τη δραστηριότητα πιο ελκυστική και ευχάριστη.

- **Ιδιοκτησία και Κατοχή (Ownership and Possession)**

Είναι η κατηγορία στην οποία ο χρήστης παρακινείται επειδή αισθάνεται ότι κατέχει κάτι. Όταν ένας παίκτης νιώσει το αίσθημα της ιδιοκτησίας, θέλει να κάνει αυτά που έχει στην κατοχή του καλύτερα και όσο το δυνατόν περισσότερα. Ακόμη, αν ο παίκτης ξοδεύει πολύ χρόνο για να προσαρμόσει το προφίλ του ή τον εικονικό του χαρακτήρα, αισθάνεται αυτόματα μεγαλύτερη κατοχή σε αυτό. Ουσιαστικά, γίνεται λόγος για τη βασική μονάδα επιθυμίας συσσώρευσης πλούτου, όπου οι παίκτες τείνουν να συλλέγουν πολλαπλά εικονικά αγαθά, νομίσματα και διασκεδάζουν περισσότερο με τη συλλογή κομματιών πάζλ.

- **Κοινωνική Επιρροή και Σχετικότητα(Social Influence and Relatedness)**

Σε αυτή την κατηγορία, ενσωματώνονται όλα τα κοινωνικά στοιχεία που διέπουν τους ανθρώπους συμπεριλαμβανομένων και των παρακάτω: καθοδήγηση, αποδοχή, κοινωνικές αντιδράσεις, συντροφικότητα, καθώς επίσης και ανταγωνισμός και φθόνος. Όταν κάποιος βλέπει έναν φίλο του να είναι καλός σε κάτι ή να έχει μια δεξιότητα, απευθείας θα προσπαθήσει να τον περάσει ή να βρεθεί στο ίδιο επίπεδο με εκείνον. Αυτή η κατηγορία περιλαμβάνει επίσης την τάση που έχει ο άνθρωπος να προσεγγίσει άλλους ανθρώπους, μέρη ή γεγονότα με τα οποία μπορεί να συσχετιστεί. Για παράδειγμα, αν ένα προϊόν προκαλεί την ανάκληση αναμνήσεων από την παιδική ηλικία, αυξάνει τις πιθανότητες για την αγορά του, καθώς μεγαλώνει η αίσθηση της νοσταλγίας. Αυτή η κατηγορία είναι ιδιαίτερα σημαντική, κυρίως διότι πολλές εταιρίες δίνουν πλέον μεγάλη προτεραιότητα στη βελτιστοποίηση των κοινωνικών τους στρατηγικών στο διαδίκτυο.

- **Έλλειψη και Ανυπομονησία (Scarcity and Impatience)**

Η εν λόγω κατηγορία έχει ως κύριο χαρακτηριστικό την επιθυμία για κάτι το οποίο δεν μπορεί να αποκτηθεί εύκολα. Πολλά παιχνίδια φέρουν Μηχανισμούς Συνάντησης (Appointment Dynamics), όπως για παράδειγμα, επιστρέψτε σε δύο ώρες για να παραλάβετε την ανταμοιβή σας. Ένας τέτοιος μηχανισμός παρακινεί τον άνθρωπο να το σκέφτεται όλη μέρα, αφού δεν μπορεί να το αποκτήσει εκείνη τη δεδομένη στιγμή. Αυτή η τεχνική χρησιμοποιήθηκε από το Facebook όταν ξεκίνησε για πρώτη φορά από το Harvard.

- **Έλλειψη Προγνωσιμότητας και Περιέργεια (Unpredictability and Curiosity)**

Σε αυτή τη κατηγορία υπάγεται η έμφυτη περιέργεια του παίκτη που θέλει να μάθει τι συμβαίνει στη συνέχεια. Όταν δεν γνωρίζει τι πρόκειται να συμβεί, ο εγκέφαλος αφοσιώνεται σε αυτό και το σκέφτεται αδιαλείπτως. Πολλοί άνθρωποι καθημερινά παρακολουθούν ταινίες και διαβάζουν μυθιστορήματα εξαιτίας αυτής της περιέργειας. Αυτή η κατηγορία είναι επίσης ο βασικός παράγοντας που προκαλεί τον εθισμό στα τυχερά παιχνίδια. Πολλές εταιρίες επιστρατεύουν αυτή την τακτική για να προσελκύσουν τους χρήστες, μέσα από προγράμματα κληρώσεων και λαχειοφόρων αγορών.

- **Απώλεια και Αποφυγή (Loss and Avoidance)**

Η συγκεκριμένη κατηγορία βασίζεται στην αποφυγή κάποιου αρνητικού συμβάντος. Σε μικρή κλίμακα, θα μπορούσε να αποφευχθεί η απώλεια της προηγούμενης δουλειάς. Σε μεγαλύτερη κλίμακα θα μπορούσε να αποφευχθεί η παραδοχή του παίκτη πως για όλα όσα έκανε μέχρι σε αυτό το σημείο είναι άχρηστα επειδή τώρα παραιτείται και αποχωρεί από το παιχνίδι. Τέλος, αυτή η κατηγορία αναφέρεται και στις ευκαιρίες που χάνονται, καθώς οι άνθρωποι αισθάνονται πως δεν έδρασαν άμεσα και τώρα νιώθουν πως έχασαν για πάντα την ευκαιρία.

2.4. Χαρακτηριστικά παραδείγματα εφαρμογής του Gamification

Το Gamification έχει χρησιμοποιηθεί πλέον σαν μια εφαρμογή στον τομέα του Μάρκετινγκ. Πάνω από το 70% των εταιριών παγκοσμίως σκοπεύουν να χρησιμοποιήσουν το gamification για σκοπούς μάρκετινγκ και για τη διατήρηση του πελατολογίου τους. Μερικά χαρακτηριστικά παραδείγματα εφαρμογής του είναι τα ακόλουθα (Sailer, Mense, Mayr & Mandl, 2017· Korn & Schmidt, 2015).

- Τα yellows από την Τράπεζα Πειραιώς. Πρόκειται για ένα σύστημα επιβράβευσης των αγορών που πραγματοποιούνται σε συνεργαζόμενα καταστήματα. Το πρόγραμμα των yellows βασίζεται στο μηχανισμό των εικονικών νομισμάτων. Οι πόντοι

αυτοί μπορούν να αυξηθούν και να μειωθούν ανάλογα με τη συχνότητα χρήσης που κάνει ο κάθε χρήστης.

- Το Foursquare ενσωμάτωσε με εξαιρετικό τρόπο μηχανισμούς παιχνιδιού. Ειδικότερα, εφάρμοσε τη δυνατότητα συγκέντρωσης πόντων για κάθε check-in που πραγματοποιεί ο χρήστης. Ταυτόχρονα, πρόσθεσε το μηχανισμό ανάδειξης του σε «δήμαρχο περιοχής», όταν αυτός έχει επισκεφτεί ένα σημείο πολλαπλές φορές. Βασικό χαρακτηριστικό του αποτελούν τα παράσημα που δίνει σε κάθε χρήστη μόλις επιτευχθεί ο στόχος.

- Το Facebook και το LinkedIn διαθέτουν εξίσου Gamification strategies. Η ολοκλήρωση του προφίλ ενός χρήστη ως διαδικασία ισοδυναμεί με το χτίσιμο ενός χαρακτήρα role-playing game και την ολοκλήρωση μιας αποστολής quest.

- Η Domino's pizza στο παρελθόν πρόσθεσε στην εφαρμογή της μία δυνατότητα για τους χρήστες που δεν ξέρουν τι να διαλέξουν. Κουνώντας απλά τη συσκευή τους, η ίδια η εφαρμογή τους πρότεινε τα toppings που μπορούσαν να προσθέσουν στη pizza τους.

- Η Vodafone-CU μέσα από την καινούρια της εφαρμογή CU Big Bang κατάφερε και δημιούργησε ένα διαδραστικό gamify game. Κάθε φορά που ο χρήστης ξεμένει από MB στο κινητό του, μπορεί μέσω της εφαρμογής να συλλέγει διάφορα ήδη μετεωριτών και με αυτόν τον τρόπο να συλλέγει πόντους. Όταν συλλέξει τον κατάλληλο αριθμό, του δίνεται η δυνατότητα να εξαργυρώσει τους πόντους με αντίστοιχα MB έτσι ώστε να μην ξεμένει ποτέ από δεδομένα και να σερφάρει απεριόριστα.

- Ο τομέας της εκπαίδευσης αποτελεί αναμφίβολα έναν χώρο όπου το Gamification δύναται να εφαρμοστεί με συναρπαστικά αποτελέσματα. Αντιπροσωπευτικό παράδειγμα αποτελεί το “Quest to learn”, ένα δημόσιο σχολείο στο Manhattan. Το συγκεκριμένο σχολείο ξεκίνησε τη λειτουργία του το 2009, εφαρμόζοντας επιτυχώς τη στρατηγική του Gamification of learning. Τα αποτελέσματα έδειξαν πως οι στρατηγικές gamified είχαν θετικά αποτελέσματα κατά τη μαθησιακή διαδικασία. Οι μαθητές που συμμετείχαν σε αυτά με όρους role-playing ανέπτυξαν την αυτενέργειά τους, ενώ παράλληλα ενίσχυσαν την ικανότητά τους για την επίλυση προβλημάτων. Το συγκεκριμένο μοντέλο κατάφερε να αναπτύξει στους μαθητές ικανότητες και δεξιότητες που θεωρούνται απαραίτητες, για να ανταποκριθούν στις απαιτήσεις του 21^{ου} αιώνα.

Μια καλοσχεδιασμένη Gamification (Εικόνα 2.5) εφαρμογή μπορεί να δημιουργήσει στο χρήστη ένα απόλυτα διασκεδαστικό περιβάλλον και να αποσπάσει την αφοσίωσή του. Δεν είναι τυχαίο ότι εφαρμογές σχεδιασμένες με Gamification-based μηχανισμούς δεν εγκαταλείπονται εύκολα από τους χρήστες τους. Σε κάθε περίπτωση όμως θα πρέπει να γίνει κριτική χρήση, έτσι ώστε το άτομο να διατηρεί τον έλεγχο του



Εικόνα 2. 5: Gamification.

χειρισμού.

2.5. Οφέλη – Πλεονεκτήματα- Μειονεκτήματα (Κριτική Gamification)

Ο όρος gamification έχει δεχτεί αρκετά πυρά κριτικής από την πρώτη του κιόλας εμφάνιση μέχρι και σήμερα. Η χρήση του ως εργαλείου πάνω σε ένα site ή σε μια εφαρμογή σε μορφή παιχνιδιών, διαγωνισμών και προκλήσεων, έχει αποδειχτεί ότι μπορεί να αυξήσει κατακόρυφα τη διαδραστικότητα και την αφοσίωση των χρηστών. Η δυναμική του gamification εμπλέκεται πάνω σε τρεις διαφορετικές ψυχολογίες: στα προγράμματα Loyalty, στο σχεδιασμό παιχνιδιών και στα Behavioral Economics. Σύμφωνα με την εταιρεία Badgeville, το gamification μπορεί να οδηγήσει σε αύξηση κατά 100% στο χρόνο παραμονής των επισκεπτών στο site, σε διπλάσιες εγγραφές από τις υπάρχουσες πλατφόρμες, σε τρεις φορές περισσότερο περιεχόμενο από τους χρήστες και σε πενταπλάσια αύξηση των sharing από τα sites κοινωνικής δικτύωσης.

Όλοι οι επιχειρηματικοί κλάδοι μπορούν να αξιοποιήσουν τα σημαντικά οφέλη των τεχνικών gamification σε τραπεζικά, επενδυτικά και ασφαλιστικά προϊόντα, προϊόντα ευρείας κατανάλωσης, πολυκαταστήματα, εμπορικά κέντρα, multiplex κινηματογράφους, τηλεοπτικά κανάλια και πάροχοι τηλεοπτικού περιεχομένου, προϊόντα τηλεπικοινωνιών, αθλητικό περιεχόμενο, αθλητικές ομάδες, αθλητικοί φορείς και διοργανώσεις, αεροπορικές εταιρίες, υπηρεσίες τουρισμού, φιλοξενίας και εστίασης, συνεδριών και κάθε φορέας που επιθυμεί να κερδίσει το ενδιαφέρον είτε του καταναλωτή είτε του συνεργάτη του. Επιπλέον, η εκπαίδευση, τα πανεπιστήμια, οι σχολές, οι βιβλιοθήκες, ο τύπος

(έντυπος και ψηφιακός), καθώς και οι κυβερνητικοί φορείς συνιστούν φορείς με σημαντική αποδοτικότητα μέσω της αξιοποίησης των τεχνικών gamification (Lucassen & Jansen, 2014).

Η ανάγκη να δημιουργήσεις «gamify» τη ζωή κάθε χρήστη είναι δεδομένη. Χαρακτηριστικό άλλωστε της όλης διαδικασίας είναι να μεταφέρει ο χρήστης τους κανόνες ενός παιχνιδιού σε συνθήκες που λογικά δεν εμπίπτουν, με την έκπληξη να αποτελεί χαρακτηριστικό της φιλοσοφίας χρήσης του.

Παρά την πληθώρα των πλεονεκτημάτων που παρέχει, εντούτοις το gamification έχει δεχθεί και αρνητική κριτική. Δεν είναι λίγοι που υποστηρίζουν πως η συγκεκριμένη στρατηγική μέθοδος οδηγεί στην ανάδυση ακούσιων συμπεριφορών, ενώ η εφαρμογή της στον εργασιακό χώρο αυξάνει τα επίπεδα άγχους στους εργαζόμενους, οι οποίοι κατά την προσπάθειά τους να διακριθούν, αρκετά συχνά οδηγούνται σε σύνδρομο burn-out με επακόλουθα συμπτώματα την εκδήλωση έντονου εκνευρισμού, ανίας, συχνής αφηρημάδας και διάσπασης, έντονης απογοήτευσης και χαμηλής αυτοπεποίθησης.

3. Κεφάλαιο 3^ο: Η τεχνολογία του NFC

Η τεχνολογία NFC αποτελεί μια πρότυπη τεχνολογία ασύρματης συνδεσιμότητας που επιτρέπει τη διασύνδεση δυο συσκευών ή μιας συσκευής NFC με μία NFC ετικέτα (NFC tag) με σκοπό την ανταλλαγή δεδομένων με ένα απλό άγγιγμα. Το NFC (Near Field Communication) αναπτύχθηκε το 2004 από το NFC Forum, ο οποίος είναι ένας μη κερδοσκοπικός οργανισμός, και πιο συγκεκριμένα ως κεντρική ιδέα ξεκίνησε από τις εταιρίες Nokia και Sony (Badra, Serhrouchni & Urien, 2004).

3.1. Τι είναι το RFID

Η τεχνολογία RFID ή διαφορετικά Radio - Frequency Identification είναι μια ασύρματη τεχνολογία που επιτρέπει την ανταλλαγή δεδομένων μεταξύ ενός RFID reader (αναγνώστης) και μιας ηλεκτρονικής RFID tag (ηλεκτρονική ετικέτα) μέσω ραδιοκυμάτων. Τα ηλεκτρομαγνητικά κύματα βοηθούν στη μεταφορά των δεδομένων έτσι ώστε ο reader να μπορεί να διαβάσει ή να γράψει δεδομένα από και προς το RFID tag (Nath, Reynolds & Want, 2006· Syed & Mohammad, 2008).

Όταν ένα RFID tag βρεθεί εντός εμβέλειας με έναν RFID reader, τότε σε αυτό το σημείο ενεργοποιείται το εσωτερικό του κύκλωμα και εκκινεί η μεταφορά των δεδομένων. Ένα RFID tag (Εικόνα 3.1) αποτελείται από 2 μέρη και πιο συγκεκριμένα από ένα ολοκληρωμένο κύκλωμα (IC - Integrated Circuit), το οποίο περιλαμβάνει μνήμη ώστε να αποθηκεύει δεδομένα - πληροφορίες και να συμβάλλει στη διαμόρφωση και αποδιαμόρφωση του ραδιοκύματος (RF Signal), και μια κεραία (antenna). Η συγκεκριμένη κεραία είναι υπεύθυνη για την αποστολή και τη λήψη του σήματος. Επιπλέον, τα RFID Tags διακρίνονται σε δυο (2) κατηγορίες, στις ενεργές ετικέτες και στις παθητικές ετικέτες.

Τα ενεργά RFID tags, από τη μία, διαθέτουν ένα ολοκληρωμένο κύκλωμα, μια κεραία και μια πηγή ενέργειας, η οποία ενεργοποιεί το κύκλωμα ώστε να παραχθεί το σήμα που θα αποσταλεί. Τα ενεργά RFID tags έχουν μεγαλύτερη εμβέλεια και μεγαλύτερη μνήμη σε σχέση με τα παθητικά και το μέγεθός τους είναι σαφώς μεγαλύτερο από αυτό των παθητικών λόγω της ύπαρξης πηγής ενέργειας. Τα παθητικά RFID tags αποτελούνται όπως και στα ενεργά tags από ένα ολοκληρωμένο σύστημα και μια κεραία. Η έλλειψη πηγής ενέργειας είναι ο λόγος που λειτουργεί παθητικά. Το παθητικό tag ενεργοποιείται

όταν δεχτεί το ραδιοκύμα ώστε να στείλει και αυτό με τη σειρά του την απάντηση μέσω



Εικόνα 3. 1: RFID tag.

κεραίας. Λόγω της έλλειψης της πηγής ενέργειας είναι μικρά σε μέγεθος και η τιμή τους στην αγορά είναι πολύ χαμηλή. Η διάρκεια ζωής τους είναι μεγάλη.

3.2. NFC ιστορική αναδρομή

Το Near Field Communication (NFC) (Εικόνα 3.2) είναι μια καινούρια τεχνολογία βασισμένη στο RFID, η οποία υποστηρίζει την ασύρματη διασύνδεση μεταξύ συσκευών σε απόσταση μικρότερη από 10 εκατοστά. Η συγκεκριμένη τεχνολογία αναπτύχθηκε με σκοπό να επιλύσει υπάρχοντα και μελλοντικά προβλήματα στο χώρο των επικοινωνιών



Εικόνα 3. 2: Λογότυπο NFC.

και στη μεταφορά των δεδομένων (Badra & Badra, 2016· Want, 2011).

Το Near Field Communication (NFC) καθόρισε τις τεχνικές προδιαγραφές και την αρχιτεκτονική που πρέπει να εφαρμόζονται σε κάθε συσκευή η οποία είναι συμβατή με το NFC. Σκοπός του NFC Forum είναι η δημιουργία εφαρμογών και λύσεων βασισμένων στη τεχνολογία NFC. Το 2006 το NFC Forum προχώρησε στην επίσημη ανακοίνωση σχετικά με την αρχιτεκτονική της τεχνολογίας NFC για τις συσκευές και τις «έξυπνες» ετικέτες (NFC Tag), οι οποίες χρησιμοποιούν το NFC προσελκύοντας έτσι οργανισμούς, όπως ο GSMA (GSM Association), EMVCo και Smart Card Alliance. Το Forum υπογράφοντας συμφωνίες με τους παραπάνω οργανισμούς κατόρθωσε να κατοχυρώσει τις απαιτούμενες

τεχνικές προδιαγραφές για τη ορθή και ασφαλή χρήση του NFC. Επιπλέον την ίδια χρονιά, η Nokia κυκλοφόρησε στην αγορά το πρώτο κινητό τηλέφωνο που υποστήριζε NFC, το μοντέλο Nokia 6131. Ακολούθως το 2009, το NFC Forum ανακοίνωσε τα πρότυπα για την Peer – to – Peer επικοινωνία, τη μεταφορά επαφών, url και την εκκίνηση Bluetooth επικοινωνίας. Μετά το πέρας ενός έτους, το 2010 η Samsung λάνσαρε το πρώτο κινητό τηλέφωνο NFC με λειτουργικό Android που ήταν το μοντέλο Nexus S.

3.3. Εφαρμογές NFC

Η τεχνολογία NFC δύναται να εφαρμοστεί και να αξιοποιηθεί ποικιλοτρόπως σε διάφορους και πολυάριθμους τομείς, όπως είναι η υγεία, η εκπαίδευση και η κάλυψη καθημερινών αναγκών, παρέχοντας στον εκάστοτε άνθρωπο - χρήστη ένα ευρύ φάσμα δυνατοτήτων και διευκολύνσεων στο άμεσο μέλλον. Ωστόσο, στην Ελλάδα δεν έχουν αξιοποιηθεί ακόμα οι δυνατότητες και τα πλεονεκτήματα που παρέχει το NFC (Nasution, Husni & Wuryandari, 2012· Vedat, Busra & Kerem, 2013).

3.3.1. Πληρωμές

Το NFC διαθέτει όλες τις προδιαγραφές ώστε να αποτελέσει αναπόσπαστο κομμάτι στις πληρωμές. Κάθε κάτοχος κινητού τηλεφώνου NFC μπορεί με ένα άγγιγμα να δώσει το προβλεπόμενο αντίτιμο για τις υπηρεσίες που του προσφέρονται. Ακόμη, το NFC έχει τη δυνατότητα να εγγυηθεί για την εξοικονόμηση χρόνου, διότι ο χρόνος που απαιτείται για την ολοκλήρωση μιας συναλλαγής είναι ελάχιστος, της τάξης του δευτερολέπτου. Παράλληλα, εγγυάται τη μέγιστη δυνατή ασφάλεια, καθώς η συναλλαγή πραγματοποιείται γρήγορα και από κοντινή απόσταση.

3.3.2. Μεταφορές

Η εφαρμογή του NFC στα Μέσα Μαζικής Μεταφοράς αποτελεί μία καινοτομία που προσφέρει πολλά προνόμια στον επιβάτη. Πιο συγκεκριμένα, μπορεί να υποστηρίξει την αγορά εισιτηρίων ή καρτών μακράς διάρκειας με ένα απλό άγγιγμα στα σημεία πώλησής τους. Ο έλεγχος επιβίβασης του κάθε επιβάτη πραγματοποιείται μέσω του κινητού του τηλεφώνου σε τρένα, λεωφορεία και αεροπλάνα. Επίσης, μέσα στα Μέσα Μαζικής Μεταφοράς ο επιβάτης θα μπορεί με ένα άγγιγμα στα «έξυπνα» πόστερ που είναι

αναρτημένα, να κατεβάζει αυτόματα πληροφορίες, όπως δρομολόγια και χάρτες, καθώς επίσης θα μπορεί να ενημερώνεται αυτόματα για ενδεχόμενες προσφορές εισιτηρίων και καινούριων εκπτώσεων.

3.3.3. Υγεία

Το NFC σε νοσοκομειακές μονάδες χρησιμοποιείται για την εύκολη και γρήγορη πρόσβαση στο ιστορικό των ασθενών προσφέροντας την απαραίτητη ασφάλεια ως προς το απόρρητο των δεδομένων. Κάθε ασθενής διαθέτει το δικό του NFC Tag στο οποίο αποθηκεύεται το ιατρικό ιστορικό του, καθώς και οι επισκέψεις του σε νοσοκομειακές μονάδες. Οι γιατροί μέσω της τεχνολογίας NFC θα έχουν τη δυνατότητα να καταγράψουν όλες τις ενέργειες που θα πραγματοποιηθούν σχετικά με τη νοσηλεία του εκάστοτε ασθενή, την πορεία της υγείας τους και την άμεση ενημέρωση του ιστορικού. Με αυτόν τον τρόπο, θα εξαλείφονται πιθανές καθυστερήσεις και ενδεχόμενα λάθη.

3.3.4. Επιχειρήσεις

Οι επιχειρήσεις μέσω της τεχνολογίας NFC μπορούν να δημιουργήσουν ένα σύστημα ελέγχου και εποπτείας των εργαζομένων. Κάθε υπάλληλος θα δηλώνει την ώρα προέλευσης και αποχώρησής του από την επιχείριση στην οποία απασχολείται με ένα απλό σκανάρισμα της έξυπνης κάρτας του στον reader εισόδου/εξόδου. Έτσι, θα υπάρχει πλήρης εποπτεία των ωρών εργασίας του κάθε υπαλλήλου. Με αυτόν τον τρόπο, ο εργοδότης θα έχει τη δυνατότητα μέσω της έξυπνης κάρτας, να ορίσει τα μέρη στα οποία τα μέλη του προσωπικού θα έχουν ελεύθερη πρόσβαση.

3.3.5. Εκπαίδευση

Στον τομέα της εκπαίδευσης, χάρη στην τεχνολογία του NFC είναι εφικτός ο αυτοματισμός πολλών διαδικασιών που πραγματοποιούνται σε ένα σχολικό ή ακαδημαϊκό συγκρότημα. Κάθε μαθητής/φοιτητής θα διαθέτει δική του NFC ταυτότητα την οποία θα χρησιμοποιεί ώστε να δηλώσει την παρουσία του στους χώρους του συγκροτήματος (τάξεις, αμφιθέτρα, βιβλιοθήκη, γυμναστήριο, κ.α), αλλά και προσιτά φέρνοντάς την σε επαφή με έναν NFC αναγνώστη. Με αυτόν τον τρόπο, δίνεται η δυνατότητα στους

καθηγητές να ελέγχουν κατά κάποιο τρόπο (και να νιώθουν ασφάλεια) τους χώρους που κινούνται οι μαθητές/φοιτητές.

3.4. NFC Αναγνώστης (NFC Reader)

Ένας NFC Reader (NFC αναγνώστης) (Εικόνα 3.3) είναι μια ενεργή συσκευή NFC η οποία μπορεί να είναι εσωτερική ή εξωτερική. Ένας εσωτερικός αναγνώστης μπορεί να ενσωματωθεί σε ένα κινητό τηλέφωνο NFC, ώστε να εκτελεί διαδικασίες ανάγνωσης/εγγραφής από και προς το Tag. Ένα κινητό τηλέφωνο με εσωτερικό αναγνώστη είναι πάντα ενεργό ώστε να παράγει το δικό του μαγνητικό πεδίο το οποίο είναι απαραίτητο για τη διασύνδεση μεταξύ ενός Tag ή ενός δεύτερου τηλεφώνου. Τέλος υπάρχει και ο εξωτερικός NFC reader που χρησιμοποιείται για ανάγνωση/εγγραφή καρτών

ACR1255U-J1



Bluetooth NFC Reader

Εικόνα 3. 3: NFC reader εισόδου/εξόδου.

όπως επίσης για την πραγματοποίηση χρηματικών συναλλαγών (Hongwei, 2013).

3.5. Πλεονεκτήματα NFC

Το NFC έχει το πλεονέκτημα ότι είναι εύκολο και απλό στη χρήση του και προορίζεται για όλες τις ηλικίες. Δεν χρειάζεται τίποτα περισσότερο από το να ακουμπήσει ο χρήστης το κινητό του τηλέφωνο σε ένα tag ή σε ένα αναγνώστη. Χάρη στην αυτοματοποιημένη διαδικασία στην οποία έχει προγραμματιστεί η κάθε εφαρμογή, ανάλογα με την κατάσταση, δρα μόνη της. Για παράδειγμα στα σενάρια πληρωμών δεν χρειάζεται να έχει μαζί του κάποιος χρήστης όλες του τις κάρτες παρά μόνο το κινητό του τηλέφωνο. Όλοι θα μπορούν με ένα άγγιγμα σε κάποιο NFC Tag να λάβουν αυτόματα χρήσιμες πληροφορίες για αυτό που απλά κοιτάζουν χωρίς περεταίρω διαδικασία (Kai, 2007).

Το NFC προσφέρει το απαιτούμενο επίπεδο ασφάλειας στη μεταφορά των δεδομένων διότι κάνει χρήση ασφαλούς καναλιού μεταφοράς. Όλες οι συναλλαγές

γίνονται σε απόσταση 5 εκατοστών. Είναι σχεδόν απίθανο κάποιος να παρέμβει και να υποκλέψει δεδομένα . Το NFC διασφαλίζει το απόρρητο των πληροφοριών, όπως καταστάσεις υγείας και κωδικούς καρτών. Στην περίπτωση για παράδειγμα του ιατρικού ιστορικού τα δεδομένα χάρις στη λειτουργία έξυπνης κάρτας του κινητού τηλεφώνου δεν μπορούν να χαθούν. Στην δεύτερη περίπτωση με τις πιστωτικές κάρτες , εάν χαθεί μια πιστωτική κάρτα μπορεί εύκολα κάποιος να τη χρησιμοποιήσει και να κάνει αγορές , ενώ εάν είναι όλες οι κάρτες στο NFC κινητό τηλέφωνο ακόμα και να χαθεί το κινητό θα πρέπει να είναι γνωστός ο κωδικός πρόσβασης της συσκευής.

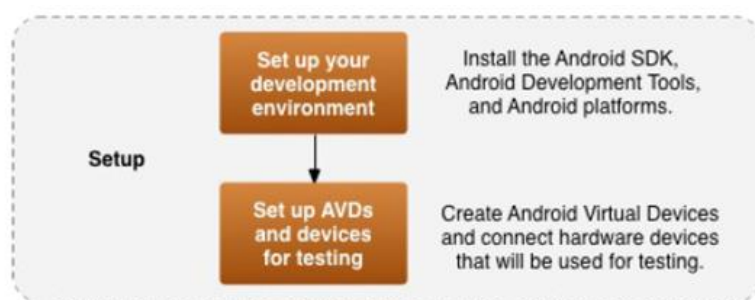
Τέλος , το NFC μπορεί να αξιοποιηθεί σε διάφορους τομείς όπως για παράδειγμα στην υγεία , στα Μέσα Μαζικής Μεταφοράς, εκπαίδευση, σε αθλητικούς χώρους κ.α. μέσα στο κινητό του κάθε χρήστη και απλώς με ένα άγγιγμα να παίρνει το ελεύθερο για τις υπηρεσίες που θέλει να χρησιμοποιήσει.

4. Κεφάλαιο 4ο: Ανάπτυξη Εφαρμογών σε Android

Η ανάπτυξη εφαρμογών στο Android είναι μια σύνθετη και χρονοβόρα διαδικασία η οποία αποτελείται από 4 βασικά στάδια.

4.1.Εγκατάσταση Λογισμικού

Στο πρώτο στάδιο της ανάπτυξης, ο προγραμματιστής καλείται να δημιουργήσει το περιβάλλον εργασίας στο οποίο θα γίνει ο σχεδιασμός, η ανάπτυξη, ο έλεγχος και η λειτουργία των εφαρμογών (Εικόνα 4.1). Ο χρήστης στο στάδιο αυτό κάνει εγκατάσταση το περιβάλλον ανάπτυξης (IDE) και τα εργαλεία του Android SDK που επιθυμεί (Studio,



Εικόνα 4. 1: Πρώτο Βήμα – Εγκατάσταση λογισμικού.

2019).

Εν συνεχεία, ο προγραμματιστής δημιουργεί έναν αριθμό από εικονικές συσκευές στη διαχείριση εικονικών συσκευών (AVD) έτσι ώστε να μπορέσει να δοκιμάσει τις εφαρμογές του σε διαφορετικές συσκευές Android.

4.2. Ανάπτυξη Πηγαίου Κώδικα Εφαρμογής

Πρόκειται για μια χρονοβόρα και πολύπλοκη διαδικασία. Σε αυτό το στάδιο ο προγραμματιστής πρέπει να αποφασίσει για τις δυνατότητες και το περιεχόμενο που θα περιλαμβάνει η εφαρμογή, να εντοπίσει ποιες από αυτές τις δυνατότητες είναι εφικτές και ποιες χρήζουν περαιτέρω έρευνας για να προστεθούν στο μέλλον, να σχεδιάσει το layout με γνώμονα τη λειτουργικότητα και να αποφύγει υπερβολές στο σχεδιασμό.

Η διαδικασία ξεκινάει με ένα νέο Project το οποίο θα περιέχει τον πηγαίο κώδικα, τις εικόνες, τα κείμενα και οποιαδήποτε εργαλεία χρειάζονται έτσι ώστε να δομηθεί σωστά μια εφαρμογή (Εικόνα 4.2). Σε κάθε Project, ο προγραμματιστής φροντίζει ώστε το υλικό

του να είναι τακτοποιημένο και ο κώδικας ευανάγνωστος με σκοπό να ακολουθήσει τη διαδικασία του Debugging.



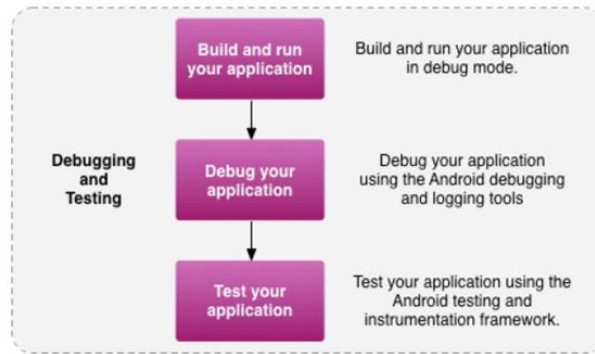
Εικόνα 4. 2: Δεύτερο βήμα – Ανάπτυξη πηγαίου κώδικα εφαρμογής.

4.3. Αποσφαλμάτωση (Debugging) και Δοκιμαστική Φάση Εφαρμογής

Μια εξίσου κρίσιμη διαδικασία είναι η αποσφαλμάτωση. Το πρώτο στάδιο σχετίζεται με το αρχικό χτίσιμο της εφαρμογής και τη λειτουργία της σε debug mode (Εικόνα 4.3). Για να γίνει το compile της εφαρμογής βασική προϋπόθεση είναι ο κώδικας να μην έχει κανένα συντακτικό λάθος, διαφορετικά ειδοποιούν τον προγραμματιστή να τα διορθώσει. Εφόσον έχει γίνει η διαδικασία του compile, η εφαρμογή μπορεί να δοκιμαστεί είτε σε εικονική συσκευή μέσω του AVD Manager, είτε απευθείας σε φυσική συσκευή μέσω ADB push εντολής (Studio, 2019).

Στο δεύτερο στάδιο, ο προγραμματιστής καλείται να αντιμετωπίσει τα λειτουργικά προβλήματα της εφαρμογής του. Κατά τη λειτουργία της συσκευής εντοπίζει τα τυχόν προβλήματα που υπάρχουν και ύστερα διορθώνει τα κομμάτια του κώδικα που δημιουργούν σφάλματα. Επιπλέον, διορθώνει τα αισθητικά σφάλματα (interface, menu) της εφαρμογής.

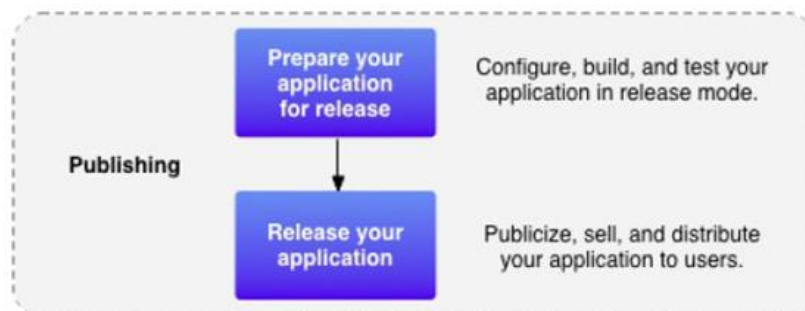
Στο τρίτο στάδιο, ο προγραμματιστής αφού έχει ολοκληρώσει τη φάση της αποσφαλμάτωσης (debugging) επιστρέφει στο πρώτο βήμα, δηλαδή στο compile και τη δοκιμή της εφαρμογής σε εικονική ή φυσική συσκευή, ώστε να διαπιστώσει τα αποτελέσματα του δεύτερου βήματος, της αποσφαλμάτωσης. Ένα προαιρετικό στάδιο είναι η “Δημόσια δοκιμαστική φάση” της εφαρμογής. Σε αυτή τη φάση, εθελοντές προσφέρονται να δοκιμάσουν τις λειτουργίες της εφαρμογής στις συσκευές τους και να αναφέρουν τυχόν προβλήματα, παρατηρήσεις, προτάσεις και διάφορα σχόλια προκειμένου να είναι πλήρως λειτουργική και ευχάριστη στο χρήστη η εφαρμογή.



Εικόνα 4. 3: Τρίτο βήμα – Δοκιμή και Debugging της εφαρμογής.

4.4. Τελική έκδοση και δημοσίευση της εφαρμογής στο κοινό.

Στο τέταρτο και τελευταίο στάδιο της εφαρμογής, ο προγραμματιστής είναι έτοιμος να εκτελέσει το τελικό compile της εφαρμογής σε κανονική λειτουργία και όχι σε debug (Εικόνα 4.4). Στη συνέχεια, ακολουθεί η φάση του developer. Αφού πρώτα ο χρήστης δημιουργήσει κάποιο λογαριασμό, είτε στο Google play είτε σε κάποιο εναλλακτικό market όπως το marketplace της Amazon και το AppStore της Huawei, έπειτα είναι σε θέση να δημοσιεύσει την εφαρμογή. Μπορεί κατά τη δημοσίευση της εφαρμογής να ορίσει τιμή πώλησης ή να την διαθέσει δωρεάν. Επιπλέον, ένα άλλο μέσο δημοσίευσης της εφαρμογής είναι και η προσωπική ή εταιρική ιστοσελίδα του δημιουργού. Το μειονέκτημα σε αυτή την περίπτωση είναι η έλλειψη ελέγχου για τις αναβαθμίσεις της εφαρμογής από



Εικόνα 4. 4: Τέταρτο βήμα – Δημοσίευση της εφαρμογής.

έναν αυτόματο μηχανισμό ελέγχου και λήψης, όπως είναι τα διάφορα marketplaces.

4.5. Android SDK.

Το Android SDK αποτελεί μια συλλογή εργαλείων και βιβλιοθηκών που καθιστούν εφικτή την ανάπτυξη εφαρμογών στο Android. Το λογισμικό ανάπτυξης

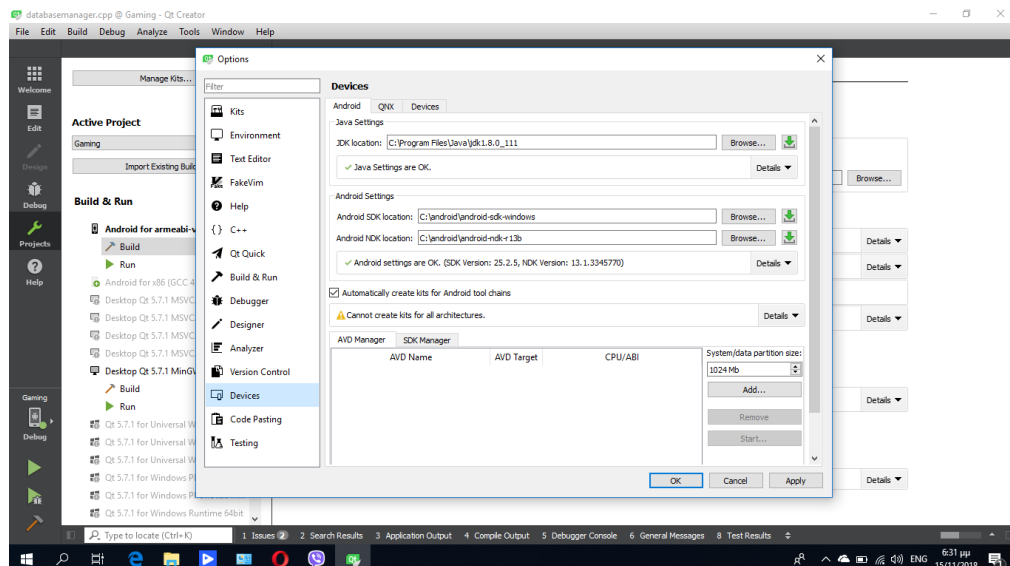
περιλαμβάνει μια μεγάλη λίστα με εργαλεία ανάπτυξης. Σε αυτά περιλαμβάνονται (Studio, 2019):

- Εργαλεία Debugging των εφαρμογών
- Βιβλιοθήκες
- Εξομοιωτής συσκευών (Android Virtual Machines)
- Documentation
- Δείγματα Κώδικα
- Tutorials

Το SDK υποστηρίζει πολλά δημοφιλή λειτουργικά συστήματα συμπεριλαμβανομένων όλων των σύγχρονων διανομών Linux, το MAC OS 11.1.2 και μεταγενέστερα, τα Windows XP και τις μεταγενέστερες εκδόσεις.

Το λογισμικό ανάπτυξης αποτελείται από πακέτα τα οποία βρίσκονται αποθηκευμένα σε ένα επίσημο repo της Google, και ο προγραμματιστής μπορεί να κατεβάσει πέραν των βασικών πακέτων και άλλα τα οποία υποστηρίζουν παλαιότερες εκδόσεις του Android, ή άλλες συσκευές εκτός κινητών συσκευών (π.χ. Google TV Addon).

Όσον αφορά την υποστήριξη παλαιότερων εκδόσεων του Android, το SDK κάνει εφικτή την υποστήριξη σε αυτές δίνοντας στον προγραμματιστή την δυνατότητα να στοχεύσει αυτός σε ποια APIs θα απευθύνεται η εφαρμογή του. Αυτό είναι αναγκαίο λόγω του ότι πολλοί χρήστες έχουν παλαιότερες λειτουργικές συσκευές οι οποίες κυκλοφόρησαν με παλαιότερες εκδόσεις του Android (π.χ. 1.6 ή 2.1) και ο κατασκευαστής της συσκευής δεν έχει ή δεν πρόκειται να βγάλει αναβάθμιση για την

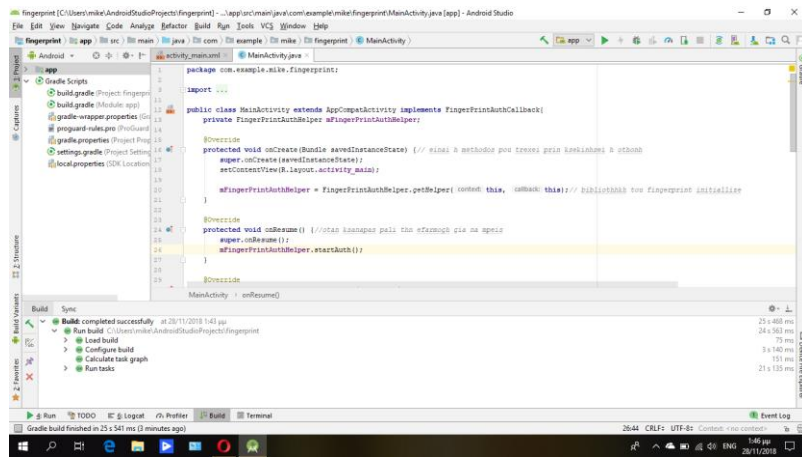


Εικόνα 4. 5: NDK και SDK package.

συσκευή τους. Το πρόβλημα αυτό είναι γνωστό ως διάσπαση του Android (Android Fragmentation).

5. Κεφάλαιο 5^ο: Android Studio περιβάλλον

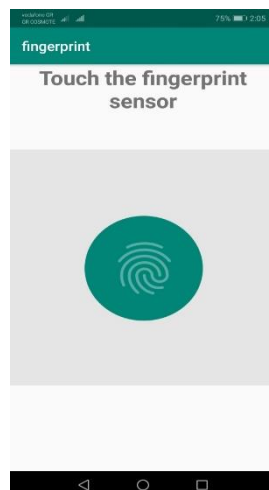
Για την κατασκευή της εφαρμογής του δαχτυλικού αποτυπώματος (fingerprint) χρειάστηκε το προγραμματιστικό περιβάλλον του Android Studio (Εικόνα 5.1). Το Android Studio είναι ένα ολοκληρωμένο προγραμματιστικό περιβάλλον (IDE) για την ανάπτυξη εφαρμογών στην πλατφόρμα Android. Είναι ένα ελεύθερο λογισμικό και είναι διαθέσιμο για Windows, Mac OS και Linux. Αντικατέστησε τα εργαλεία Eclipse Android Development Tools (ADT) ως το κύριο IDE της Google για ανάπτυξη εφαρμογών Android



Εικόνα 5. 1: Προεπιλεγμένη διάταξη του Android Studio.

(Studio, 2019).

Η εφαρμογή ελέγχει ουσιαστικά την είσοδο του κάθε χρήστη με βάση το δαχτυλικό του αποτύπωμα. Με αυτό τον τρόπο ο κάθε χρήστης που θα μπαίνει στην εφαρμογή θα είναι μοναδικός. Απαραίτητη προϋπόθεση είναι ότι όλοι οι χρήστες πρέπει να χρησιμοποιούν συσκευές Android με έκδοση 6.0 Marshmallow και πάνω. Η εικόνα



Εικόνα 5. 2: Αρχική εικόνα ελέγχου εφαρμογής.

(Εικόνα 5.2) που ακολουθεί παρουσιάζει την αρχική εικόνα της εφαρμογής του δαχτυλικού αποτυπώματος.

6. Κεφάλαιο 6^ο: Qt περιβάλλον.

Το 1991, ο Havaard Nord και ο Eirik Chambe-Eng άρχισαν τη δημιουργία του QT (προφέρεται “cute”), το οποίο αρχικά υποστηριζόταν μόνο σε Windows και Linux, ενώ το 2001 δημιουργείται το Qt 3.0, το οποίο υποστηρίζει και Mac OS X, όπου η έκδοση δεν προσφερόταν δωρεάν, ενώ το 2003 δημιουργείται το Qt 3.2 το οποίο υποστήριζε πλέον δωρεάν έκδοση για τα Mac OS X. Το 2008 η Nokia εξαγόρασε την εταιρία των Nord και Chambe- Eng και την μετονόμασε σε Qt development Frameworks. Από το 2008 μέχρι σήμερα χρησιμοποιείται η κύρια πλατφόρμα ανάπτυξης λογισμικού για όλες τις συσκευές τους. Το Qt είναι ένα ελεύθερο λογισμικό για τη δημιουργία γραφικών διεπαφών χρήστη (graphical user interfaces) και εφαρμογών· δεν είναι γλώσσα προγραμματισμού από μόνο του αλλά ένα πλαίσιο (framework) γραμμένο στη γλώσσα προγραμματισμού C++.



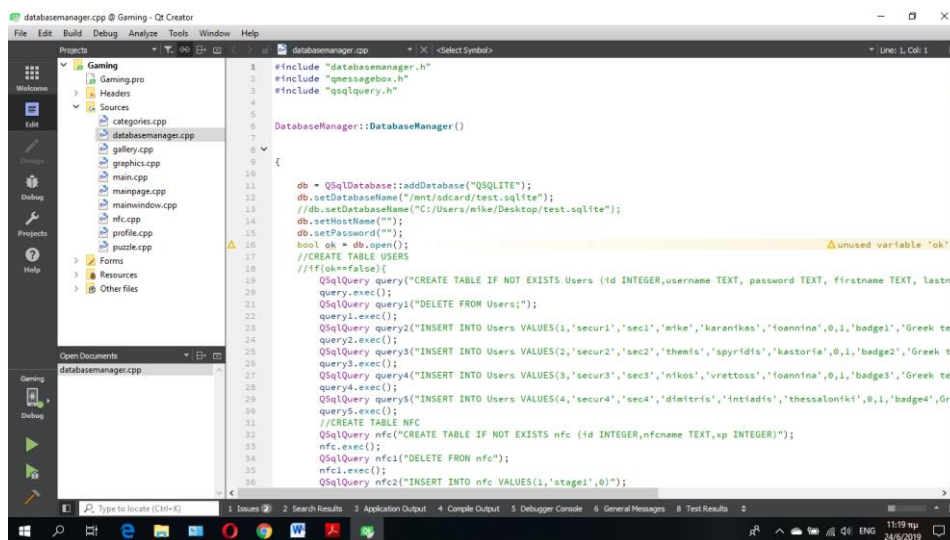
Εικόνα 6. 1: Λογότυπο Qt development .

(Nicholas, 2018).

Το Qt αποτελεί βασικό εργαλείο της ανάπτυξης εφαρμογών, καθώς απλοποιεί τη δημιουργία γραφικών εφαρμογών και προσφέρει πολλαπλά οφέλη. Το πρώτο από αυτά είναι οι προσφερόμενες κλάσεις. Η βάση του Qt αποτελείται από περισσότερες από 400 κλάσεις για containers, δημιουργία και σχεδίαση GUI, βάσεις δεδομένων, δικτύωση, υποστήριξη XML και πολυγλωσσικότητας, OpenGL κ.α. Το δεύτερο πλεονέκτημα οφείλεται στο Qt Designer του περιβάλλοντος. Το τρίτο αποτελεί μία εφαρμογή οπτικής δημιουργίας GUI, με τη χρήση του ποντικιού, που επιτρέπει την γρήγορη ανάπτυξη γραφικών περιβαλλόντων, που είναι ίδια σε όλες τις πλατφόρμες. Ένα ακόμα όφελος εντοπίζεται στο Qt Linguist, το οποίο και αποτελεί ένα σύνολο εργαλείων που διευκολύνει την υποστήριξη πολλών γλωσσών, με αποτέλεσμα την εύκολη μετάφραση των μηνυμάτων

και των μενού κάθε εφαρμογής Qt. Κλείνοντας, ένα ακόμα πολύ σημαντικό όφελος προκύπτει από την Qt Assistant· μία εφαρμογή που είναι σε θέση να ενσωματώνεται με αλλαγές μέσα στις εφαρμογές Qt. Εδώ, επιβάλλεται να σημειωθεί ότι το Qt περιβάλλον είναι η μοναδική compiler εφαρμογή που είναι σε θέση να μετατρέψει την γλώσσα προγραμματισμού C++ απευθείας σε Android, προσφέροντας μία πληθώρα γραφικών.

Συνεχίζοντας, το Qt Creator, αποτελεί ένα ενσωματωμένο περιβάλλον ανάπτυξης C++, JavaScript και QML, μέρος του SDK για το πλαίσιο ανάπτυξης εφαρμογών Qt GUI. Ο επεξεργαστής κώδικα στο Qt Creator υποστηρίζει την επισήμανση σύνταξης για διάφορες γλώσσες, είναι σε θέση να αναλύσει τον κώδικα σε γλώσσες C++ και QML και ενσωματώνει τον Qt Designer για το σχεδιασμό και την κατασκευή GUI από τα Widgets Qt. Περιλαμβάνει ένα οπτικό εργαλείο εντοπισμού σφαλμάτων (visual debugger) και μια ενσωματωμένη διάταξη GUI (GUI Layout) και σχεδιαστή μορφών (forms designer) (Εικόνα 6.2). Το Qt Creator χρησιμοποιεί τον μεταγλωτιστή C++ από τη συλλογή του GNU Compiler σε Linux και FreeBSD. Στα windows μπορεί να χρησιμοποιήσει το MinGW ή το MSVC με την προεπιλεγμένη εγκατάσταση και μπορεί επιπλέον να χρησιμοποιήσει τον Microsoft Debugger Console όταν μεταγλωτιστεί από τον πηγαίο



Εικόνα 6. 2: Το περιβάλλον της Qt (Qt Creator).

κώδικα· εδώ, υποστηρίζεται και το Clang.

Στην παρούσα εργασία το περιβάλλον Qt χρησιμοποιήθηκε για τη δημιουργία της εφαρμογής Gaming, που θα αναλυθεί στο επόμενο κεφάλαιο.

7. Κεφάλαιο 7^ο: Η εφαρμογή Gaming.

Η ιδέα ανάπτυξης της εφαρμογής προήλθε από την χρήση των NFC tags στο αεροδρόμιο Ελευθέριος Βενιζέλος· σε κάθε βάρδια που ο κάθε security περνούσε και χτυπούσε με τη συσκευή του στο NFC reader (κάνοντας ουσιαστικά check in- check out) επιβεβαίωνε ότι πραγματοποιήθηκε έλεγχος, σημειώνοντας την ημερομηνία και την ώρα τσεκαρίσματος και έδινε αναφορά. Θέτοντας ως βάση την διαθεσιμότητα των συγκεκριμένων tags και στο αεροδρόμιο των Ιωαννίνων, αλλά και τα πολλαπλά οφέλη του gamification, αναπτύχθηκε η παρακάτω εφαρμογή, η οποία όχι μόνο θα παρείχε διασκέδαση και ευχάριστο περιβάλλον στους χρήστες, που περιπολούν 24 ώρες το 24ωρο αλλά και βοηθά στην βελτίωση της επαγγελματικής συμπεριφοράς και στην αύξηση της παραγωγικότητας.

Σε γενικές γραμμές για να ξεκινήσει η ανάπτυξη κάποιας εφαρμογής, είναι επιτακτική ανάγκη να πραγματοποιηθεί ανάλυση των απαιτήσεων του χρήστη, σχεδίαση και τέλος υλοποίηση της εφαρμογής. Κλείνοντας, υπάρχει το στάδιο του ελέγχου και της διανομής της εφαρμογής.

7.1. Προδιαγραφές χρήσης της εφαρμογής.

Η παραπάνω gamification εφαρμογή βασίζεται εξ ολοκλήρου στο λειτουργικό πρόγραμμα Android και για την εκτέλεσή της κρίνεται απαραίτητο κάθε συσκευή να έχει διαθέσιμη προς χρήση τη λειτουργία του NFC, ενώ είναι προαιρετική η λειτουργία του Fingerprint (δαχτυλικού αποτυπώματος) για περαιτέρω ασφάλεια, αφού υφίσταται μόνο σε εκδόσεις που έπονται του Marshmallow 6.0. Η ανάπτυξη της εφαρμογής σε λογισμικό Android οφείλεται στην τεράστια απήχισή του, στο μεγαλύτερο ποσοστό των χρηστών smartphones αλλά και στο γεγονός ότι, ως ελεύθερο λογισμικό, το Android, παρέχει μία πληθώρα δυνατοτήτων, τόσο όσον αφορά τον προγραμματισμό και την ανάπτυξη μίας εφαρμογής όσο και από την πλευρά χρήσης της. Ακόμα, το προαναφερθέν λογισμικό αποτελεί ένα λογισμικό πολλαπλών λειτουργιών (multi – tasking), γεγονός που επιτρέπει την ταυτόχρονη λειτουργία περισσότερων από μία εφαρμογών, χωρίς την μείωση της απόδοσης τόσο της εκάστοτε συσκευής όσο και της ίδιας της εφαρμογής. Τέλος, αφού το Android αποτελεί μία πλατφόρμα ανοιχτού κώδικα, παρέχει στον εκάστοτε προγραμματιστή τη δυνατότητα επέκτασης και τροποποίησης, ώστε να είναι δυνατή η

προσαρμογή στις τελευταίες εξελίξεις και η υιοθέτηση νέων τεχνολογιών και διαφαλίζει την συνεχόμενη πρόοδό του.

Όπως αναφέρθηκε και παραπάνω, η gamification εφαρμογή που δημιουργήθηκε απευθύνεται κατ' αποκλειστικότητα σε μια ομάδα security του αεροδρομίου των Ιωαννίνων και βασίζεται στο ήδη υπάρχον σύστημά (database) τους. Οι συναρτήσεις που χρησιμοποιήθηκαν στο πρόγραμμα είναι συναρτήσεις για το NFC, fingerprint, database και sql query.

Από τα παραπάνω καθίσταται σαφές ότι για την χρήση της εφαρμογής, επιβάλλεται, ο χρήστης, να είναι security στο αεροδρόμιο των Ιωαννίνων, να είναι εξοικειωμένος και η συσκευή του να υποστηρίζει την τεχνολογία NFC για να είναι σε θέση να κάνει χρήση της εφαρμογής και να πάρει τα εκάστοτε βραβεία – puzzles, να έχει κάνει ξανά χρήση των smartphones και, στην περίπτωση χρήσης του δκτυλικού αποτυπώματος να διατίθεται έκδοση Android 6.0 και πάνω.

7.2. Σχεδίαση και ανάλυση της εφαρμογής.

Στο πρώτο στάδιο σχεδιασμού της εφαρμογής δημιουργήθηκε ο πρώτος έλεγχος του χρήστη με δακτυλικό αποτύπωμα, ο οποίος βασίζεται σε γλώσσα προγραμματισμού Java και υλοποιήθηκε στο πρόγραμμα Android Studio, καθώς η Qt Creator δεν παρέχει βιβλιοθήκη και υποστήριξη για το δακτυλικό αποτύπωμα.

```
package com.example.mike.fingerprint;

import android.content.Intent;
import android.hardware.fingerprint.FingerprintManager;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;

import com.multidots.fingerprintauth.FingerPrintAuthCallback;
import com.multidots.fingerprintauth.FingerPrintAuthHelper;

public class MainActivity extends AppCompatActivity implements
FingerPrintAuthCallback{
    private FingerPrintAuthHelper mFingerPrintAuthHelper;

    @Override
    protected void onCreate(Bundle savedInstanceState) { (μέθοδος πριν την
εκκίνηση της οθόνης).
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mFingerPrintAuthHelper = FingerPrintAuthHelper.getHelper(this,
this); (initializing την βιβλιοθήκη του fingerprint)
```

```

    }

    @Override
    protected void onResume () (επιστροφή στην εφαρμογή και επανείσοδος)
        super.onResume ();
        mFingerprintAuthHelper.startAuth ();
    }

    @Override
    protected void onPause () { (μέθοδος αναμονής στο background)
        super.onPause ();
        mFingerprintAuthHelper.stopAuth ();
    }
    (συναρτήσεις fingerprint)
    @Override
    public void onNoFingerprintHardwareFound () {

    }

    @Override
    public void onNoFingerprintRegistered () {

    }
    (ανυπαρξία fingerprint πριν από την έκδοση marshmallow)
    @Override
    public void onBelowMarshmallow () {

    }
    (εισαγωγή fingerprint και authorization)
    @Override
    public void onAuthSuccess (FingerprintManager.CryptoObject
    cryptoObject) {
        Intent launchIntent =
        getPackageManager ().getLaunchIntentForPackage ("org.qtproject.example"); //
        (εκκίνηση δεύτερης εφαρμογής και εκκίνηση νέας διεργασίας)
        if (launchIntent != null) {
            startActivity (launchIntent);
        }
    }
    (εισαγωγή fingerprint αλλά μη αναγνώριση)
    @Override
    public void onAuthFailed (int errorCode, String errorMessage) {

    }
}

```

Στο δεύτερο στάδιο δημιουργήθηκε μία τοπική βάση δεδομένων (database) η οποία στηρίζεται στην υπάρχουσα database των security και περιέχει την ταυτότητα (id) το κάθε χρήστη· αυτό σημαίνει ότι περιέχει το username, το password, το firstname, το lastname, το location, το experience, το level, τα badges και το category. Η συγκεκριμένη βάση δεδομένων δημιουργήθηκε πάνω σε QSqlQuery.

```

#include "databasemanager.h"
#include "qmessagebox.h"
#include "qsqlquery.h"

```



```
DatabaseManager::DatabaseManager ()
```

```
{  
  
    db = QSqlDatabase::addDatabase("QSQLITE");  
    db.setDatabaseName("/mnt/sdcard/test.sqlite");  
    //db.setDatabaseName("C:/Users/mike/Desktop/test.sqlite");  
    db.setHostName("");  
    db.setPassword("");  
    bool ok = db.open();  
    //CREATE TABLE USERS  
    //if(ok==false){  
        QSqlQuery query("CREATE TABLE IF NOT EXISTS Users (id  
INTEGER,username TEXT, password TEXT, firstname TEXT, lastname TEXT,  
location TEXT, xp INTEGER, level INTEGER,badges TEXT,category TEXT)");  
        query.exec();  
        QSqlQuery query1("DELETE FROM Users;");  
        query1.exec();  
        QSqlQuery query2("INSERT INTO Users  
VALUES(1,'secur1','sec1','mike','karanikas','ioannina',0,1,'badge1','Gree  
k television presenters')");  
        query2.exec();  
        QSqlQuery query3("INSERT INTO Users  
VALUES(2,'secur2','sec2','themis','spyridis','kastoria',0,1,'badge2','Gre  
ek television presenters')");  
        query3.exec();  
        QSqlQuery query4("INSERT INTO Users  
VALUES(3,'secur3','sec3','nikos','vrettoss','ioannina',0,1,'badge3','Gree  
k television presenters')");  
        query4.exec();  
        QSqlQuery query5("INSERT INTO Users  
VALUES(4,'secur4','sec4','dimitris','intiadis','thessaloniki',0,1,'badge4  
,Greek television presenters')");  
        query5.exec();  
        //CREATE TABLE NFC  
        QSqlQuery nfc("CREATE TABLE IF NOT EXISTS nfc (id INTEGER,nfcname  
TEXT,xp INTEGER)");  
        nfc.exec();  
        QSqlQuery nfc1("DELETE FROM nfc");  
        nfc1.exec();  
        QSqlQuery nfc2("INSERT INTO nfc VALUES(1,'stage1',0)");  
        nfc2.exec();  
        QSqlQuery nfc3("INSERT INTO nfc VALUES(2,'stage2',0)");  
        nfc3.exec();  
        QSqlQuery nfc4("INSERT INTO nfc VALUES(3,'stage3',0)");  
        nfc4.exec();  
        QSqlQuery nfc5("INSERT INTO nfc VALUES(4,'stage4',0)");  
        nfc5.exec();  
        //CREATE TABLE PUZZLE  
        QSqlQuery puzzle("CREATE TABLE IF NOT EXISTS puzzle (id  
INTEGER,puzzlename TEXT,pieces INTEGER)");  
        puzzle.exec();  
        QSqlQuery puzzle1("DELETE FROM puzzle");  
        puzzle1.exec();  
        QSqlQuery puzzle2("INSERT INTO puzzle VALUES(1,'puz1',5)");  
        puzzle2.exec();  
        QSqlQuery puzzle3("INSERT INTO puzzle VALUES(2,'puz2',5)");  
        puzzle3.exec();  
        QSqlQuery puzzle4("INSERT INTO puzzle VALUES(3,'puz3',5)");  
    }  
}
```

```

        puzzle4.exec();
        QSqlQuery puzzle5("INSERT INTO puzzle VALUES(4,'puz4',5)");
        puzzle5.exec();
        //CREATE IMAGE WOMEN TABLE
        QSqlQuery image("CREATE TABLE IF NOT EXISTS image(id INTEGER,path
TEXT,category TEXT)");
        image.exec();
        QSqlQuery image1("DELETE FROM image");
        image1.exec();
        QSqlQuery image2("INSERT INTO image VALUES(1,':/Foreign
presenter/beyonce.jpg','Foreign presenters')");
        image2.exec();
        QSqlQuery image3("INSERT INTO image VALUES(2,':/Foreign
presenter/jlo.jpg','Foreign presenters')");
        image3.exec();
        QSqlQuery image4("INSERT INTO image VALUES(3,':/Foreign
presenter/eva mentes.jpg','Foreign presenters')");
        image4.exec();
        QSqlQuery image5("INSERT INTO image VALUES(4,':/Foreign
presenter/lagertha.jpg','Foreign presenters')");
        image5.exec();
        QSqlQuery image6("INSERT INTO image VALUES(5,':/Foreign
presenter/madona.jpg','Foreign presenters')");
        image6.exec();
        QSqlQuery image7("INSERT INTO image VALUES(6,':/Foreign
presenter/shakira.jpg','Foreign presenters')");
        image7.exec();
        QSqlQuery image8("INSERT INTO image VALUES(1,':/Greek television
presenters/elena Papari(500x500).jpg','Greek television presenters')");
        image8.exec();
        QSqlQuery image9("INSERT INTO image VALUES(2,':/Greek television
presenters/jenny-balatsinou(500x500).jpg','Greek television
presenters')");
        image9.exec();
        QSqlQuery image10("INSERT INTO image VALUES(3,':/Greek television
presenters/korinthiou-homepage-image-500x500.jpg','Greek television
presenters')");
        image10.exec();
        QSqlQuery image11("INSERT INTO image VALUES(4,':/Greek television
presenters/nomikou(500x500).jpg','Greek television presenters')");
        image11.exec();
        QSqlQuery image12("INSERT INTO image VALUES(5,':/Greek television
presenters/pinelopi-anastasopoulou-homepage-image-500x500.jpg','Greek
television presenters')");
        image12.exec();
        QSqlQuery image13("INSERT INTO image VALUES(6,':/Greek television
presenters/zeta-makripoulia-homepage-image-500x500.jpg','Greek television
presenters')");
        image13.exec();
        QSqlQuery image14("INSERT INTO image VALUES(7,':/Greek television
presenters/μενεγακη(500x500).jpg','Greek television presenters')");
        image14.exec();

    }

//}

int DatabaseManager::check_login(QString username, QString password){
    int check = -1;
    QSqlQuery query(QString("SELECT id,username,password FROM Users;"));

```

```

        qDebug() << "error====" << query.lastError().text();
        while (query.next()) {

if ((username==query.value(1).toString()) && (password==query.value(2).toString())) {
            check=query.value(0).toInt();

        }

    }
    return check;
}

QStringList DatabaseManager::get_info(int id) {
    QStringList info;
    QSqlQuery query(QString("SELECT username, xp, level,badges,category
FROM Users WHERE id=%1").arg(id));
    while (query.next()) {
        info.append(query.value(0).toString());
        info.append(query.value(1).toString());
        info.append(query.value(2).toString());
        info.append(query.value(3).toString());
        info.append(query.value(4).toString());

    }
    return info;
}

//update xp
void DatabaseManager::change_xp(int xp,int id)
{
    int x=0;

    QSqlQuery query(QString("SELECT xp FROM Users WHERE id=%1").arg(id));
    while (query.next()) {
        x= query.value(0).toInt();
    }
    qDebug() << "before xp=====" << x;
    x += xp;
    qDebug() << "after xp=====" << x;
    //check update xp
    QSqlQuery query2(QString("UPDATE Users SET xp=%1 WHERE
id=%2").arg(x).arg(id));
    query2.exec();

}

//update level
void DatabaseManager::change_level(int level,int id)
{
    int y=0;
    QSqlQuery query(QString("SELECT level FROM Users WHERE
id=%1").arg(id));
    while (query.next()) {
        y=query.value(0).toInt();
    }
    qDebug() << "before level=====" << y;
    y += level;
    qDebug() << "after level=====" << y;
    //check update level

```

```

QSqlQuery query3(QString("UPDATE Users SET level=%1 WHERE
id=%2").arg(y).arg(id));
query3.exec();

}
(επιστροφή στην τιμή του xp)
int DatabaseManager::get_xp(int id){
    int xp=0;
    QSqlQuery query(QString("SELECT xp FROM Users WHERE id=%1").arg(id));
    while(query.next()){
        xp= query.value(0).toInt();
    }
    return xp;
}
(επιστροφή στην τιμή του level)
int DatabaseManager::get_level(int id){
    int level=0;
    QSqlQuery query(QString("SELECT level FROM Users WHERE
id=%1").arg(id));
    while(query.next()){
        level=query.value(0).toInt();
    }
    return level;
}
(επιστροφή στο κομμάτι puzzle)
QString DatabaseManager::get_puzzle_name(int id){
    QString puzzlename="";
    QSqlQuery query(QString("SELECT puzzlename FROM puzzle WHERE
id=%1").arg(id));
    while(query.next()){
        puzzlename=query.value(0).toString();
    }
    return puzzlename;
}

int DatabaseManager::set_xp(int xp,int id){
    int z=0;
    QSqlQuery query(QString("SELECT xp FROM Users WHERE id=%1").arg(id));
    while(query.next()){
        z= query.value(0).toInt();
    }
    z =0;
    QSqlQuery query2(QString("UPDATE Users SET xp=%1 WHERE
id=%2").arg(z).arg(id));
    query2.exec();

    return z;
}
QString DatabaseManager::get_image(int level,QString category){
    QString path;
    QSqlQuery query(QString("SELECT path FROM image WHERE id=%1 AND
category='%2'").arg(level).arg(category));
    while(query.next()){
        path=query.value(0).toString();
    }
    return path;
}
//update category
bool DatabaseManager::set_category(int id,QString category){

```

```

        QString command = QString("UPDATE Users SET category='%1' WHERE
id=%2").arg(category).arg(id);
        QSqlQuery query;
        return query.exec(command);
    }

    QString DatabaseManager::get_category(int id){
        QString category = "";
        QString command = QString("SELECT category FROM Users WHERE
id=%1").arg(id);
        QSqlQuery query(command);
        while(query.next()){
            category=query.value(0).toString();
        }
        return category;
    }

```

Εν συνεχεία δημιουργήθηκε το NFC της εφαρμογής, βασισμένο σε έτοιμη βιβλιοθήκη nfc.h με ιδιότητα την αναγνώριση του εκάστοτε NFC της συσκευής. Εδώ, είναι σημαντικό να σημειωθεί ότι χρησιμοποιήθηκαν και άλλες βιβλιοθήκες: αυτές είναι ui_nfc.h, qmessagebox.h, qndefmessage.h, qndefrecord.h.

```

#include "nfc.h"
#include "ui_nfc.h"
#include "qmessagebox.h"
#include "qndefmessage.h"
#include "qndefrecord.h"

nfc::nfc(int id, puzzle *userPuzzle, QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::nfc)
{
    ui->setupUi(this);
    this->id = id;
    this->userPuzzle = userPuzzle;
    nfcManager=new QNearFieldManager(this);
    nfcManager->
>setTargetAccessModes(QNearFieldManager::NdefReadTargetAccess);
    //Get notified when the tag gets out of range

    connect(nfcManager, SIGNAL(targetLost(QNearFieldTarget*)), this, SLOT(target
Lost(QNearFieldTarget*)));

    connect(nfcManager, SIGNAL(targetDetected(QNearFieldTarget*)), this, SLOT(ta
rgetDetected(QNearFieldTarget*)));
    nfcManager->startTargetDetection();
}

void nfc::targetDetected(QNearFieldTarget *target)
{
    const bool hasNdefMessage = target->hasNdefMessage();
    if(hasNdefMessage){

        connect(target, SIGNAL(ndefMessageRead(QNdefMessage)), this, SLOT(ndefMessag
eRead(QNdefMessage)));
        target->readNdefMessages();
    }
}

```

```

        qDebug() << "uri target detected!";

    }

    //cache target to member variable for writing
    cachedTarget=target;
}

void nfc::targetLost(QNearFieldTarget *target)
{
    cachedTarget =NULL;
    target->deleteLater();
}

// read nfc message for nfc
void nfc::ndefMessageRead(const QNdefMessage &message)
{
    foreach(const QNdefRecord &record, message){
        if (record.isRecordType<QNdefNfcTextRecord>()){
            QNdefNfcTextRecord textRecord(record);
            ui->nfc_label->setText(textRecord.text());
            //metatroph tou int se qstring
            //QString string_id = QString("%1").arg(id);
            db->change_xp(50,id);
            int new_xp = db->get_xp(id);
            int new_level=db->get_level(id);
            qDebug() << "new puzzle=====" << userPuzzle->return_puzzle();
            qDebug() << "new xp=====" << new_xp;
            qDebug() << "new level=====" << new_level;

                if(new_xp%200==0){

QMessageBox::critical(this, tr("Congratulation!!!"), tr(" Collect Piece of
Puzzle "));

                qDebug() << "Piece Collect";
                userPuzzle->increasePieces();
            }
            if (new_xp ==1000){
                new_xp = db->set_xp((new_xp - 1000),id);
                qDebug() << "poso xp" << new_xp;
            }
            if(new_xp%1000==0){

QMessageBox::critical(this, tr("Congratulation!!!"), tr("Level up"));
            qDebug() << "Level up";
            db->change_level(1,id);
            userPuzzle->increaseId();
        } //if (new_xp > 1000){
            // new_xp = db->get_xp(id);
            //new_level=db->get_level(id);
            //new_xp = db->set_xp((new_xp-1000),id);
            //qDebug() << "mphke edw" << new_xp;
            //}

        }

    }

}

void nfc::nfcWriteTag(const QString &nfcTagText){
    //the device currently has a target in reach

```

```

        if (cachedTarget){
            //convert text to a uri, adding missing http://if necessary
            QUrl convertedUrl=QUrl::fromUserInput(nfcTagText);
            //check if it is a valid url
            if (convertedUrl.isValid() && nfcTagText.contains('.')){
                QNdefMessage message;
                //create a uri ndef record
                QNdefNfcUriRecord uriRecord;
                uriRecord.setUri(convertedUrl);
                message.append(uriRecord);
                qDebug() << "Writting URI";
                //write ndef message to the tag
                cachedTarget->writeNdefMessages(QList<QNdefMessage>() << message);
            }else{
                qDebug() << "no valid uri";
            }
        }else{
            qDebug() << "no cached target available";
        }
    }
}

nfc::~~nfc()
{
    delete ui;
}

```

Επιπροσθέτως, δημιουργήθηκε η καρτέλα puzzle (Screenshot από το puzzle) και στη συνέχεια δημιουργήθηκε η καρτέλα profile του εκάστοτε χρήστη· εδώ, συμπεριλαμβάνονται το id, το username, το level, το xp, τα badges, το category και το share your profile in facebook.

```

#include "puzzle.h"
#include "ui_puzzle.h"

puzzle::puzzle(DatabaseManager *db, int id, QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::puzzle)
{
    ui->setupUi(this);
    this->id=id;
    this->puzzlename=db->get_puzzle_name(id);
}

QString puzzle::return_puzzle(){
    return puzzlename;
}

void puzzle::set_xp_and_level(int level, int xp){
    int remainxp=xp-((level-1)*1000);
    if(xp<200){
        this->pieces=0;
    }else if(xp<400){
        this->pieces=1;
    }else if(xp<600){
        this->pieces=2;
    }
}

```

```

    }else if(xp<800){
        this->pieces=3;
    }else if(xp<1000){
        this->pieces=4;
    }

    ui->puzzlename->setText (puzzlename);
    ui->puzzlename->setReadOnly(true);
    ui->pieces->setText (QString("%1").arg(pieces));
    ui->pieces->setReadOnly(true);
}

void puzzle::increasePieces () {
    pieces++;
}
void puzzle::increaseId () {
    pieces=0;
    //id puzzle=level user
    id++;
    this->puzzlename=db->get_puzzle_name(id);
}
puzzle::~puzzle ()
{
    delete ui;
}

```

Ακολουθεί ο κώδικας δημιουργίας profile

```

#include "profile.h"
#include "ui_profile.h"

profile::profile(int id,DatabaseManager *db,QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::profile)
{
    ui->setupUi(this);
    this->id=id;
    info=db->get_info(id);
    ui->id->setText (QString("%1").arg(id));
    ui->id->setReadOnly(true);
    ui->username->setText (info[0]);
    ui->username->setReadOnly(true);
    ui->level->setText (info[2]);
    ui->level->setReadOnly(true);
    ui->xp->setText (info[1]);
    ui->xp->setReadOnly(true);
    ui->badges->setText (info[3]);
    ui->badges->setReadOnly(true);
    ui->Category->setText (info[4]);
    ui->Category->setReadOnly(true);
}

profile::~profile ()
{
    delete ui;
}

```



```

void profile::on_pushButton_clicked()
{

    QString link="http://www.facebook.com";
    QDesktopServices::openUrl(QUrl(link));
}

```

Επίσης, δημιουργήθηκε η καρτέλα των categories, όπου δίδεται η δυνατότητα στο χρήστη να επιλέξει την επιθυμητή κατηγορία, η οποία, σε κάθε περίπτωση, προσφέρει μία ποικιλία από εικόνες. Η εφαρμογή προσφέρει 4 κατηγορίες με 4 φωτογραφίες στην κάθε κατηγορία.

```

#include "categories.h"
#include "ui_categories.h"

Categories::Categories(int id, DatabaseManager *db, int level, QWidget
*parent) :
    QMainWindow(parent),
    ui(new Ui::Categories)
{

    ui->setupUi(this);
    this->db=db;
    this->id=id;
    current_category = "";
    view = new QListView();
    // Create model
    model = new QStringListModel(this);
    // Make data
    QStringList List;
    List << "Greek television presenters" << "Foreign presenters" <<
"Global security staff " << "Landscapes";
    QStandardItemModel *itemmodel = new QStandardItemModel;
    for (int i = 0; i < List.count(); i++)
    {

        QStandardItem *item = new QStandardItem();
        item->setText(List[i]);
        item->setSizeHint(QSize(item->sizeHint().width(), 100));
        itemmodel->setItem(i, item);
        view->setModel(itemmodel);

    }

    // Populate our model
    model->setStringList(List);
    ui->listView->setModel(itemmodel);
    ui->listView->setEditTriggers(QAbstractItemView::NoEditTriggers);
    connect(ui->listView, SIGNAL(clicked(QModelIndex)), this, SLOT(get_selected()));
    connect(ui->Save, SIGNAL(clicked(bool)), this, SLOT(save_category()));
}

```

```

Categories::~Categories()
{
    delete ui;
}

void Categories::on_Save_clicked()
{
    QMessageBox::information(this, tr("Checkable list
model"), tr("Required components were saved."), QMessageBox::Ok);
    qDebug() << "saved====" << db->set_category(id, ui->listView-
>selectionModel()->selectedIndexes().at(0).data().toString());
}

void Categories::save_category() {

}

void Categories::get_selected() {

    qDebug() << "select category====" << ui->listView->selectionModel()-
>selectedIndexes().at(0).row();
    current_category = ui->listView->selectionModel()-
>selectedIndexes().at(0).data().toString();
    qDebug() << "current category====" << current_category;
    QString path = db->get_image(level, category);
    QImage *img=new QImage();
    img->load(path);}
}

```

Ακόμα, δημιουργήθηκε καρτέλα gallery, η οποία και περιλαμβάνει το σύνολο των ολοκληρωμένων εικόνων που έχει ξεκλειδώσει ο κάθε χρήστης.

```

#include "gallery.h"
#include "ui_gallery.h"

gallery::gallery( DatabaseManager *db, int id, int level, int xp, QWidget
*parent) :
    QMainWindow(parent),
    ui(new Ui::gallery)
{
    ui->setupUi(this);

    if (level>1){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek television
presenters/elena Papari(500x500).jpg")));
    }
    if (level>2){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou(500x500).jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari(500x500).jpg")));
    }
    if (level>3){

```

```

        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/zeta-makripoulia-homepage-image-500x500.jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari (500x500).jpg")));
        ui->pic3->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou (500x500).jpg")));
    }
    if (level>4){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/zeta-makripoulia-homepage-image-500x500.jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari (500x500).jpg")));
        ui->pic3->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou (500x500).jpg")));
        ui->pic4->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/korinthiou-homepage-image-500x500.jpg")));
    }
    if (level>5){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/nomikou (500x500).jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/zeta-makripoulia-homepage-image-500x500.jpg")));
        ui->pic3->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari (500x500).jpg")));
        ui->pic4->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou (500x500).jpg")));
        ui->pic5->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/korinthiou-homepage-image-500x500.jpg")));
    }
    if (level>6){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/nomikou (500x500).jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/zeta-makripoulia-homepage-image-500x500.jpg")));
        ui->pic3->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari (500x500).jpg")));
        ui->pic4->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou (500x500).jpg")));
        ui->pic5->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/korinthiou-homepage-image-500x500.jpg")));
        ui->pic6->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/pinelopi-anastasopoulou-homepage-image-
500x500.jpg")));
    }
}

gallery::~gallery()
{
    delete ui;
}

```

Επόμενη καρτέλα που δημιουργήθηκε είναι η καρτέλα graphics, όπου και εμφανίζεται κομμάτι – κομμάτι η κάθε φωτογραφία.

```

#include "graphics.h"
#include "ui_graphics.h"

graphics::graphics(int id, DatabaseManager *db,int pieces,int
level,QWidget *parent) :

```

```

    QMainWindow(parent),
    ui(new Ui::graphics)
{
    ui->setupUi(this);
    QPixmap pix;
    this->pieces=pieces;
    width = pix.width()/10;
    height = pix.height()/10;
    square_x=0;
    square_y=0;
    pscene=new QGraphicsScene(this);
    ui->graphicsView-
>setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
    ui->graphicsView-
>setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
    qDebug()<<"level===="<<level;
    category = db->get_category(id);
    qDebug()<<"category mikele====="<<category;
    change_image(level,category);
    set_category(id,category);
    pscene->addItem(item);
    ui->graphicsView->setScene(pscene);
    //pscene->setSceneRect(0,0,500,500);
    if(pieces==0){
        pscene-
>addRect(250,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(0,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(250,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(0,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        qDebug()<<"pieces 0";
    }else if(pieces==1){
        pscene-
>addRect(0,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(250,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(0,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        qDebug()<<"pieces 1";
    }else if(pieces==2){
        pscene-
>addRect(250,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(0,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        qDebug()<<"pieces 2" ;
    }else if (pieces==3){
        pscene-
>addRect(0,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        qDebug()<<"pieces 3";
    }

    qDebug()<<pix.width();
    qDebug()<<pix.height();

}

```

```

void graphics::change_image(int level,QString category){
    QString path = db->get_image(level,category);
    QImage *img=new QImage();
    img->load(path);
    qDebug()<<"path here"<<path;
    item = pscene->addPixmap(QPixmap::fromImage(QImage(*img)));
    pscene->addItem(item);
    ui->graphicsView->setScene(pscene);
}

void graphics::set_category(int id,QString category){
    qDebug()<<"set category"<<category;
    category=db->set_category(id,category);
}

graphics::~graphics()
{
    delete ui;
}

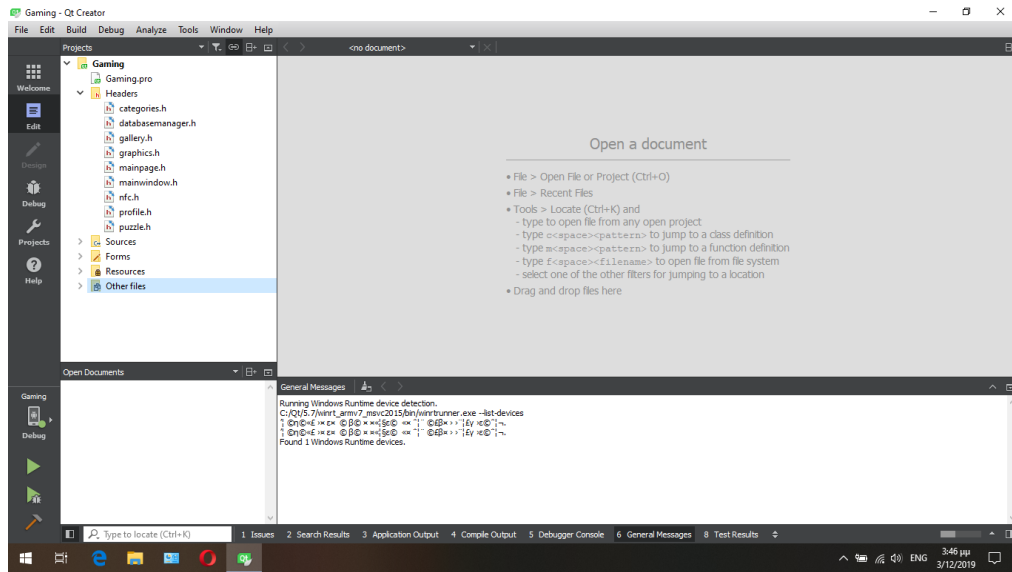
```

Στο τελευταίο στάδιο δημιουργήθηκε η καρτέλα exit η οποία και κλείνει την εφαρμογή.

Έχοντας αναλύσει όλα τα παραπάνω, στο σημείο αυτό θα αναλυθεί και ο τρόπος λειτουργίας της εφαρμογής. Οι security, κατά τη διάρκεια της υπηρεσίας τους στο χώρο του αεροδρομίου (περίπολος), όταν ο χρήστης - security πλησιάζει σε ένα από τα NFC - tags θα συλλέγει πόντους (xp) και αφού συγκεντρώσει τους πρώτους 200 πόντους, ξεκλειδώνει ένα κομμάτι puzzle (badges). Κάθε 1000 xp, ο χρήστης ανεβαίνει ένα επίπεδο (level) κατακτώντας την εκάστοτε θεματική ενότητα του επιπέδου και κάθε puzzle θα παρουσιάζει μια εικόνα ανάλογα με τη θεματολογία που έχει οριστεί από το χρήστη. Στο τέλος, αφού ολοκληρωθεί το puzzle, δίδεται η δυνατότητα στο χρήστη να ανατρέχει στο menu gallery ώστε να βλέπει τις φωτογραφίες που έχει ξεκλειδώσει αθροιστικά.

7.3. Υλοποίηση της εφαρμογής

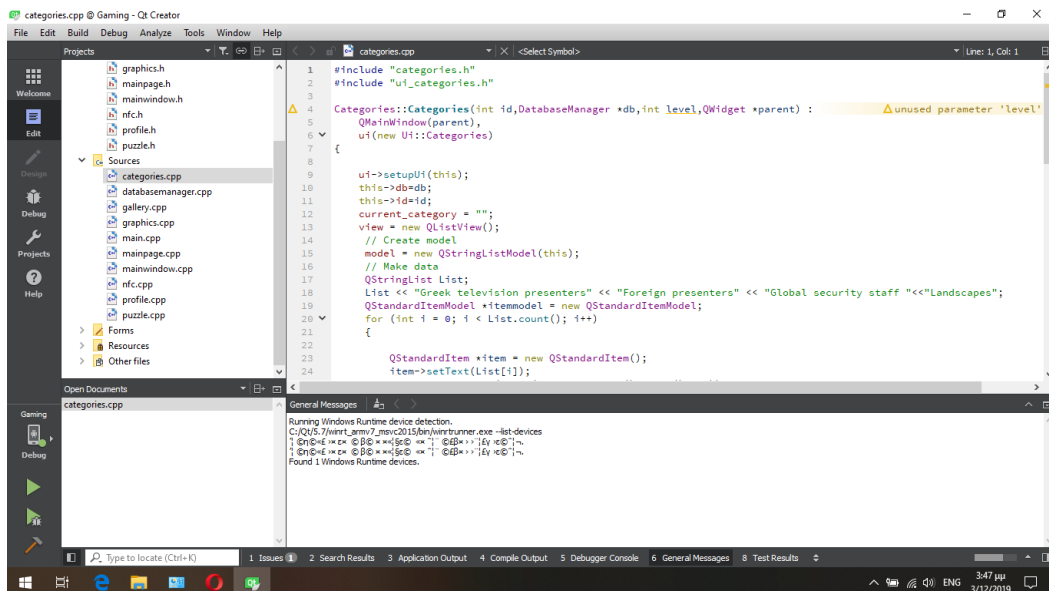
Στο υποκεφάλαιο αυτό θα γίνει αναφορά στη διαδικασία υλοποίησης της



Εικόνα 7. 1: Περιβάλλον Qt και φάκελοι εφαρμογής.

εφαρμογής, η οποία και πραγματοποιήθηκε με την χρήση του Qt Creator. Αρχικά, όπως φαίνεται και στην παρακάτω εικόνα (Εικόνα 7.1) εκινήθηκε η εφαρμογή Qt και δημιουργήθηκαν οι φάκελοι για την τμηματοποίηση του συνόλου των περιεχομένων της εφαρμογής. Πιο συγκεκριμένα, ο φάκελος με την ονομασία Headers εμπεριέχει το σύνολο των μεταβλητών και των τιμών και των μεταβλητών που χρησιμοποιήθηκαν στην εφαρμογή· αποτελεί, ουσιαστικά τον φάκελο αρχικοποίησης της εφαρμογής.

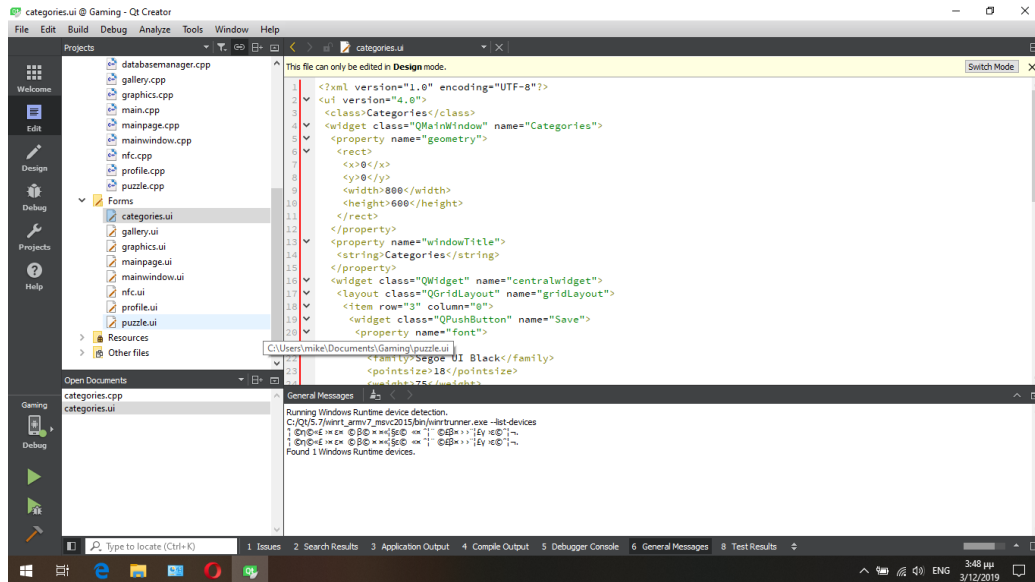
Στη συνέχεια, όπως φαίνεται και στο στιγμιότυπο παρακάτω (Εικόνα 7.2) δημιουργήθηκε φάκελος, ο οποίος εμπεριέχει τον πηγαίο κώδικα της κάθε καρτέλας –



Εικόνα 7. 2: Φάκελος Sources.

κατηγορίας· ο φάκελος αυτός ονομάζεται Sources.

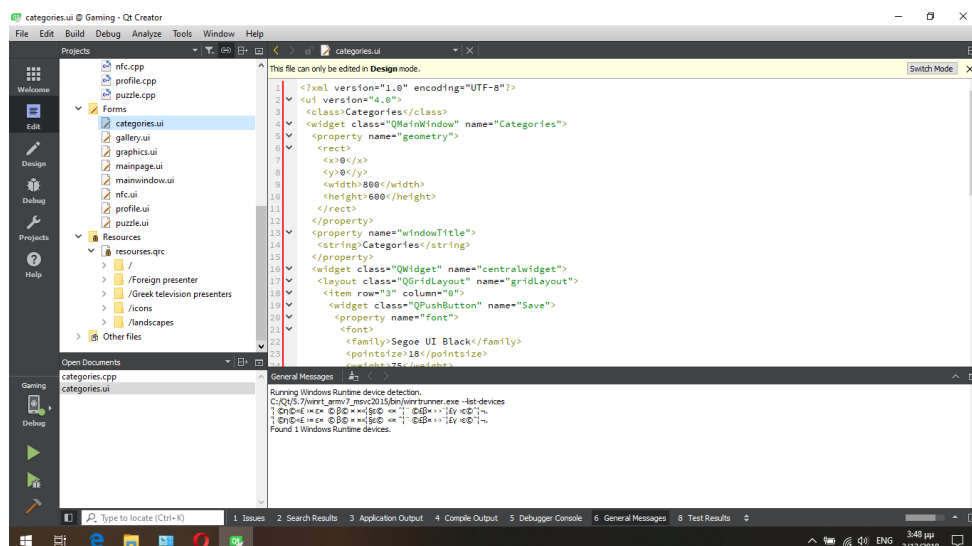
Επιπλέον, δημιουργήθηκε φάκελος με την ονομασία Forms (Εικόνα 7.3), ο οποίος και εμπεριέχει το σύνολο των πινάκων, των εικόνων, των φωτογραφιών κ.α. της



Εικόνα 7. 3: Φάκελος Forms.

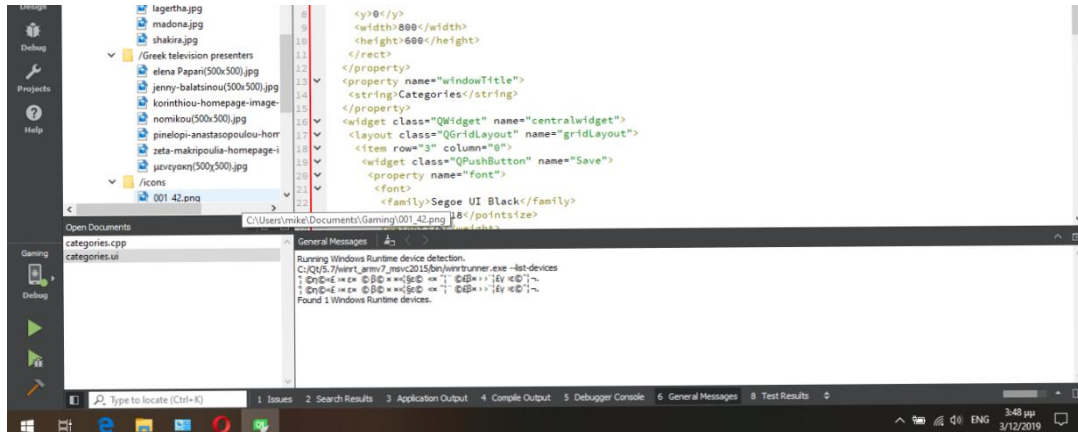
εφαρμογής· αποτελεί ουσιαστικά τον σχεδιασμό της εφαρμογής.

Ακόμα δημιουργήθηκε φάκελος, ο οποίος εμπεριέχει τον χώρο αποθήκευσης των φωτογραφιών που ξεκλειδώνει ο εκάστοτε χρήστης (Εικόνα 7.4). Μέσα στον φάκελο αυτόν υφίστανται και υποφάκελοι που εμπεριέχουν κατηγορίες εικόνων (τοπία, Έλληνες παρουσιαστές κ.α.). Εδώ, αξίζει να σημειωθεί, ότι οι φωτογραφίες που δίδονται ως



Εικόνα 7. 4: Τόπος αποθήκευσης εικόνων.

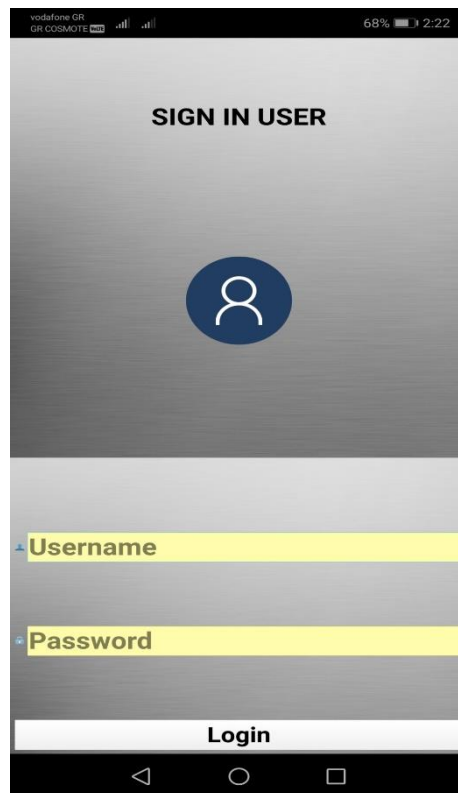
βραβεία στους χρήστες είναι μορφής jpeg και διαστάσεων 4x4 ή 500x500 (Εικόνα 7.5).



Εικόνα 7. 5: Μέγεθος εικόνων.

7.4. Παρουσίαση της εφαρμογής.

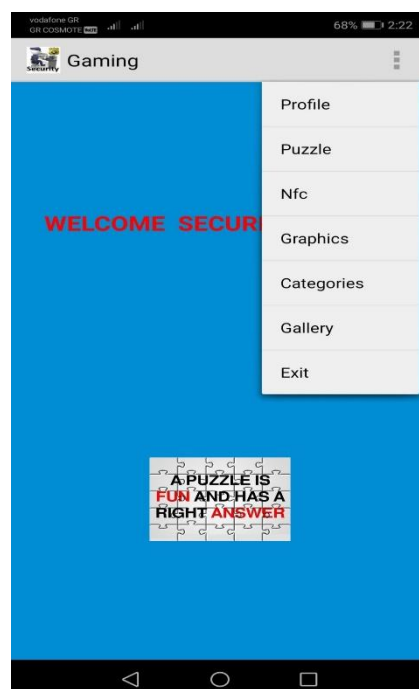
Ο χρήστης εκκινεί την εφαρμογή gaming και εν συνεχεία του παρουσιάζεται η



Εικόνα 7. 6: Αρχική εικόνα εφαρμογής.

παρακάτω αρχική εικόνα (Εικόνα 7.6).

Στην επόμενη εικόνα (Εικόνα 7.7), παρουσιάζεται στην επάνω δεξιά πλευρά το



Εικόνα 7. 7: Το Menu της εφαρμογής.

κεντρικό Menu.

Ξεκινώντας με την πρώτη επιλογή, εμφανίζεται το Profile του κάθε χρήστη

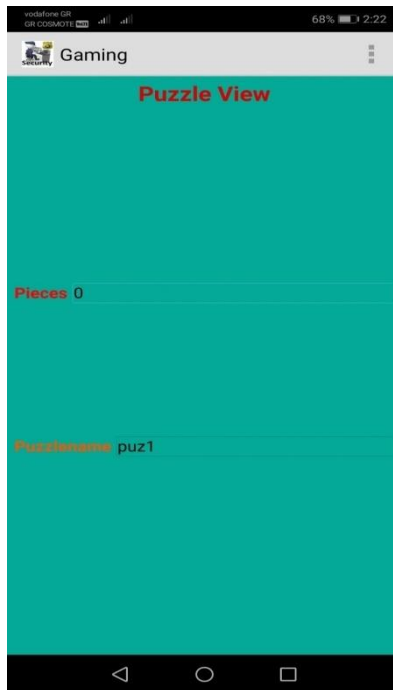


Εικόνα 7. 8: Το Profile κάθε Security.

(Εικόνα 7.8).

Αρχικά, εμφανίζεται το ID του κάθε χρήστη, το Username, το επίπεδο (level) στο οποίο βρίσκεται τη δεδομένη στιγμή, πόσους πόντους (xp) έχει συλλέξει, το βραβείο (badge) που παίρνει και σε ποια κατηγορία των φωτογραφιών βρίσκεται. Επιπλέον, μπορεί με ένα πάτημα του κουμπιού (tap) να κάνει share στο facebook, ώστε να μοιράζεται με τους επιπλέον χρήστες τα βραβεία που δέχεται κάθε φορά.

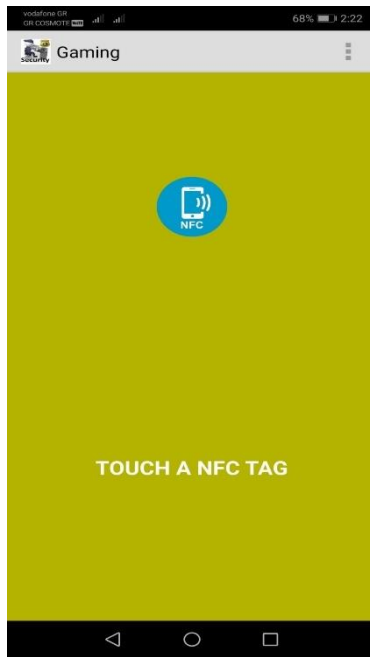
Στην λίστα που ακολουθεί, υπάρχει το Puzzle View (Εικόνα 7.9). Εδώ, ο χρήστης βλέπει κάθε φορά πόσα κομμάτια του puzzle έχει ξεκλειδώσει και το όνομα που φέρει το



Εικόνα 7. 9: Puzzle View Menu.

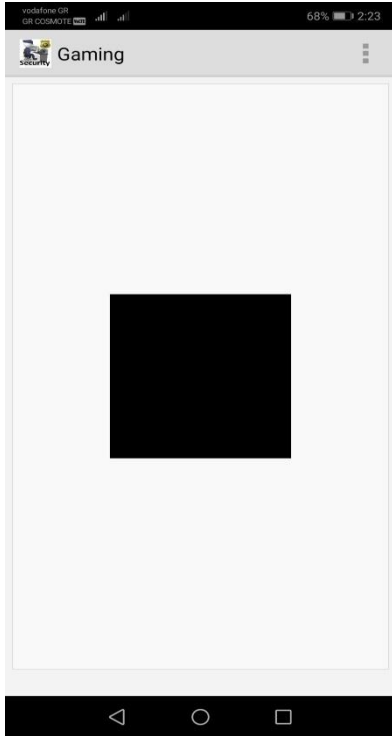
κάθε Puzzle.

Στη συνέχεια, ακολουθεί το Menu του NFC (Εικόνα 7.10), όπου ο χρήστης θα χτυπάει κάθε φορά τα NFC tags του αεροδρομίου στη συσκευή του.

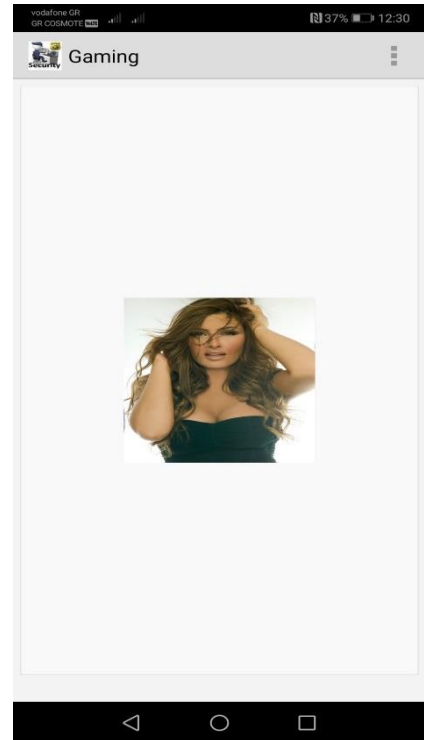


Εικόνα 7. 10: NFC Menu.

Ακολούθως, υπάρχει η επιλογή Graphics. Στην παρακάτω εικόνα (Εικόνα 7.11), παρουσιάζεται η πρώτη μορφή της εικόνας ενός puzzle. Η δεύτερη εικόνα (Εικόνα 7.12) αποτελεί ένα αντιπροσωπευτικό παράδειγμα μιας ολοκληρωμένης εικόνας ενός puzzle



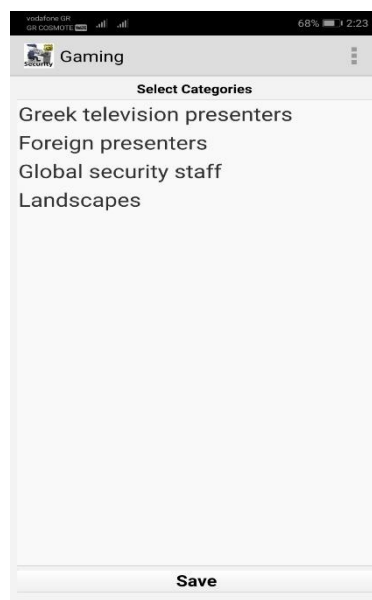
Εικόνα 7. 12: Αρχική μορφή του Puzzle.



Εικόνα 7. 11: Ολοκληρωμένη εικόνα Puzzle.

μετά τη συλλογή των πρώτων 1000 πόντων (xp).

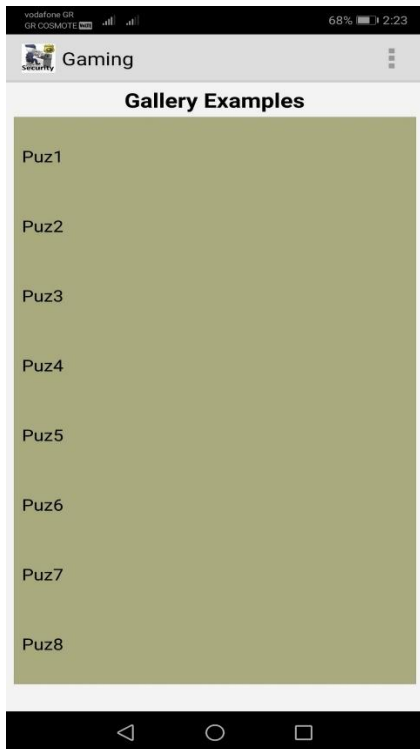
Η επόμενη επιλογή που εμφανίζεται είναι το Categories, όπου ο χρήστης κάθε φορά μπορεί να επιλέγει μια θεματολογία (κατηγορία) και να εμφανίζονται αντίστοιχα οι



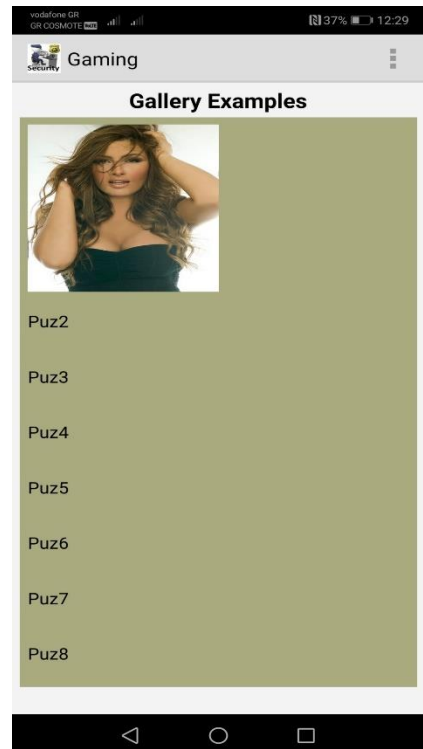
Εικόνα 7. 13: Categories Menu.

κατάλληλες φωτογραφίες, όπως παρουσιάζεται στην ακόλουθη εικόνα (Εικόνα 7.13).

Την πρωτελευταία επιλογή αποτελεί το Gallery (Εικόνα 7.14). Εδώ εμφανίζονται όλες οι τελικές εικόνες (Εικόνα 7.15), τις οποίες έχει καταφέρει να αποκαλύψει ο χρήστης συγκεντρώνοντας όλα τα κομμάτια που απαιτούνται, ώστε να τις βλέπει κάθε φορά που το επιθυμεί. Τέλος, υπάρχει η επιλογή Exit η οποία τερματίζει την εφαρμογή του Gamification.



Εικόνα 7. 15: Αρχική μορφή του Gallery.



Εικόνα 7. 14: Ολοκλήρωση της πρώτης εικόνας και αποθήκευση στο Gallery.

8. Συμπεράσματα

Η δημιουργία του λογισμικού Android αποτελεί την κίνηση αντίδρασης και συμμετοχής στον ανταγωνισμό απέναντι στα υφιστάμενα λογισμικά της δεκαετίας του 2000· το Symbian και το Windows Mobile. Η επιτυχημένη πορεία της κίνησης επιβεβαιώνεται από τη δημιουργία και από την αυξανόμενη επέκταση της Open Handset Alliance· τον πιο ισχυρό αντίπαλο για τις υπόλοιπες εταιρίες του κλάδου. Ακόμα, απόδειξη της επιτυχίας αποτελεί και η ραγδαία αύξηση των πωλήσεων των εκδόσεων του λογισμικού Android, ιδιαίτερα μετά το 2013 και πιο συγκεκριμένα, του Android OS έναντι του IOS, επιβεβαιώνοντας την άριστη λειτουργία και δομή, την πολύ μεγάλη ικανοποίηση των καταναλωτών και την συνεχόμενη βελτίωση και ανάπτυξη των παροχών, των λειτουργιών και του περιβάλλοντος του Android.

Μελετώντας το λογισμικό Android είναι φανερό ότι στην αρχιτεκτονική του, τον πρωταγωνιστικό ρόλο διαδραματίζουν 3 στοιχεία. Το πρώτο είναι ο πυρήνας στον οποίο βασίζεται, ο Linux, που αποτελεί την κινητήρια δύναμη για τις πιο βασικές λειτουργίες του συστήματος (διαχείριση εργασιών, μνήμης κ.α.). Το δεύτερο κομβικής σημασίας στοιχείο είναι η χρήση των βιβλιοθηκών, που υφίστανται για να χρησιμοποιούνται από προγράμματα υψηλότερου επιπέδου και η μεταγλώττισή τους από τις γλώσσες προγραμματισμού C και C++ είναι βαρύνουσας σημασίας αφού χρησιμοποιούνται για παροχή υποστήριξης στην αποθήκευση δεδομένων, στη διαδικτυακή πλοήγηση κ.α. Συνεχίζοντας, στο ίδιο μήκος κύματος βρίσκεται και το τρίτο στοιχείο που είναι ο χρόνος εκτέλεσης ως ένα σύνολο βιβλιοθηκών, κατάλληλων για προγραμματισμό σε Java, με την εικονική μηχανή Java Virtual Machine και στην περίπτωση του Android με τη Dalvik. Η σημασία της Dalvik ενισχύεται τόσο από την multitasking φύση του Android όσο και από το ελάχιστο αντίκτυπο που αυτή παρουσιάζει στη μνήμη της συσκευής, προσφέροντας καλύτερη και ταχύτερη λειτουργία και άρα καλύτερη εμπειρία στο χρήστη.

Η αύξηση των πωλήσεων smartphones με λογισμικό Android ξεκινώντας, το 2008, από την ένταξη νέων μελών στην Open Handset Alliance, με έμφαση την αύξηση που σημειώθηκε μετά το 2013, δικαιολογεί και τις πολλές εκδόσεις του λογισμικού. Αυτές αναδεικνύουν και την επιθυμία των αρμοδίων για άμεση και έγκαιρη αντιμετώπιση των προβλημάτων που ανέκυπταν στην αρχιτεκτονική του λογισμικού, για βελτίωση της εμπειρίας των χρηστών και του περιβάλλοντος χρήσης, για μεγαλύτερη χρηστικότητα και καλύτερη απόδοση του Android με λιγότερες απαιτήσεις στον πυρήνα και τη μνήμη και για παροχή πληθώρας δυνατοτήτων στους χρήστες και στους προγραμματιστές.

Έτσι οι εκδόσεις του έτους 2009-2010, μετατόπισαν το βάρος σε λειτουργίες της οθόνης και της κάμερας, του Bluetooth και των Widgets. Ωστόσο, οι εκδόσεις των επόμενων χρόνων, 2011-2014, προσαρμόζονται στις νέες εξελίξεις και βελτιώνονται ώστε να μπορούν να λειτουργήσουν σε tablets, τηλεοράσεις κ.α. Έτσι, κατά το έτος 2011-2012 δίδεται έμφαση στην προσιότητα στον καταναλωτή και το 2013-2014 το βάρος μετατοπίζεται στη βελτίωση του λειτουργικού συστήματος, αντικαθιστώντας τη Dalvik με το ART, για καλύτερη απόδοση των εφαρμογών. Τέλος, από το 2015-2018 επιχειρείται η πολυεπίπεδη ανάπτυξη των εκδόσεων του Android, με προσπάθεια βελτίωσης τόσο του λειτουργικού συστήματος όσο και της συνολικής εμπειρίας του χρήστη.

Στο επίπεδο του gamification των εφαρμογών, τώρα, αξίζει να σημειωθεί ότι με την εφαρμογή των στοιχείων, γνωρισμάτων και τεχνικών σχεδίασης επιδιώκεται η πιο ευχάριστη παρουσίαση, η επίτευξη στόχων, η αύξηση της παραγωγικότητας, η μείωση του άγχους και του στρες, η πολύπλευρη παροχή κινήτρων, η περαιτέρω επιμόρφωση και εκπαίδευση κ.α., σε καθημερινό προσωπικό και εργασιακό επίπεδο. Όλα αυτά έχουν ως αποτέλεσμα την βελτίωση, σε ανθρώπινο επίπεδο, πρώτον της προσωπικής και εργασιακής πραγματικότητας και δεύτερον, σε επίπεδο επιχειρήσεων, την αύξηση των πωλήσεων των εφαρμογών. Κατά κύριο λόγο, όμως, αποτελεί μία πρακτική που είναι στραμμένη και έχει αναπτυχθεί με κέντρο τον ίδιο τον άνθρωπο και όχι τις λειτουργίες, με αποτέλεσμα να επιτυγχάνεται η βελτίωση των συναισθηματικών κινήτρων και των δευσμέσεων και να αυξάνεται η αποτελεσματικότητα των ανθρώπων σε όλα τα επίπεδα.

Έχοντας ως βάση τον άνθρωπο, αναπτύχθηκε το πλαίσιο της Οκτάλυσης, η οποία αποπειράται να αποδόσει την πολυεπίπεδη επιρροή του gamification. Το αντίκτυπο του τελευταίου εντοπίζεται τόσο στην πνευματική, προσωπική και κοινωνική ανάπτυξη του χρήστη όσο και στον τομέα της διαφημιστικής δραστηριότητας και προσέλκυσης χρηστών των επιχειρήσεων. Όσον αφορά την προσωπική και πνευματική ανάπτυξη επιβάλλεται να σημειωθεί ότι στο πρώτο, το δεύτερο, το τρίτο, το τέταρτο και το έκτο στάδιο επιτυγχάνεται η ενίσχυση της αυτοπεποίθησης του παίκτη – χρήστη, τονώνεται το αίσθημα της πρόκλησης, που αποτελεί και το βασικό στοιχείο προόδου και ανάπτυξης, επανατροφοδοτείται η φαντασία με ταυτόχρονη αύξηση της ελκυστικότητας της εφαρμογής και της ταχύτητας ανταπόκρισης του χρήστη, αναπτύσσεται η ανάγκη προστασίας των ήδη κερτημένων αγαθών και ανάπτυξής τους και τέλος παρέχεται παρακίνηση και ολόημερη ενασχόληση με την εκάστοτε εφαρμογή, με αποτέλεσμα την

αύξηση της υπομονής και την αποδοχή των ορίων του χρήστη αλλά και της εφαρμογής, αντίστοιχα.

Στο επίπεδο της κοινωνικής ανάπτυξης του ατόμου, τα αποτελέσματα του gamification παρατηρούνται στο πέμπτο στάδιο της Οκτάλυσης, όπου συμπεριλαμβάνονται η συντροφικότητα, η αποδοχή του άλλου, η ενίσχυση της κοινωνικότητας κ.α., στοιχεία που άπτονται της ανθρώπινης ανάγκης για επικοινωνία. Στην περίπτωση της διαφημιστικής καμπάνιας των εταιρειών, στο έβδομο στάδιο περιέχει και ένα διαφημιστικό τρικ για καλύτερο marketing των εφαρμογών, με τη δημιουργία και ανάπτυξη της περιέργειας των χρηστών, που απορρέει από την έλλειψη προγνωσιμότητας της εφαρμογής. Εκτός, όμως, από την πληθώρα των πλεονεκτημάτων του gamification, στο τελευταίο στάδιο της Οκτάλυσης, υφίσταται και το μόνο "αρνητικό" στοιχείο, καθώς από την απώλεια εργασίας που έχει ήδη γίνει ή από την πιθανότητα παραίτησης και αποχώρησης από την εφαρμογή, είναι δυνατόν να πληγεί τόσο η αυτοπεποίθηση του χρήστη όσο και η φήμη της εφαρμογής αλλά και της εταιρίας, εξαιτίας της απογοήτευσης του πρώτου.

Στα παραπάνω πλεονεκτήματα του gamification είναι δυνατόν να προστεθούν και κάποια οφέλη των επιχειρήσεων. Το πρώτο από αυτά πηγάζει από τη διαδραστικότητα και την αφοσίωση που παρουσιάζουν οι εφαρμογές, αυξάνοντας την αναγνωρισιμότητα μίας ιστοσελίδας, εφαρμογής κ.α. και άρα τις πωλήσεις των επιχειρήσεων. Επιπλέον, αξίζει να σημειωθεί η πολυεπίπεδη επιρροή των προαναφερθέντων εφαρμογών σε όλο το φάσμα της εμπορικής ζωής, τράπεζες, πολυκαταστήματα, κινηματογράφος κ.α., καθώς προσφέρουν ευκολία στο marketing, ευελιξία και ταχύτητα αλλά και προσιτότητα στον χρήστη.

Όσον αφορά τις τεχνολογίες NFC και RFID, επιβάλλεται να αναφερθεί ότι αποτελούν τεχνολογίες ασύρματης σύνδεσης συσκευών και ανταλλαγής δεδομένων. Η διαφορά τους έγκειται στο γεγονός ότι η πρώτη απαιτεί την συμμετοχή μίας NFC συσκευής με ένα NFC tag, ενώ στην δεύτερη περίπτωση απαιτείται ένας RFID reader και ένα RFID tag, με το τελευταίο να διαχωρίζεται σε 2 κατηγορίες· τα ενεργητικά tags και τα παθητικά. Η ανωτερότητα των ενεργητικών tags οφείλεται στην μεγαλύτερη εμβέλεια και μνήμη που προσφέρουν, εξαιτίας της ύπαρξης πηγής ενέργειας. Η ύπαρξη πηγής ενέργειας, ωστόσο, έχει και αρνητική όψη καθώς αυξάνει το μέγεθος του ενεργητικού tag.

Η χρήση των προαναφερθέντων τεχνολογιών γνωρίζει μεγάλη άνθιση σε πολλά επίπεδα διεθνώς, κάτι που δεν συμβαίνει και στον ελλαδικό χώρο. Η μη αξιοποίηση των

συγκεκριμένων τεχνολογιών σε όλο το φάσμα της καθημερινότητας των χρηστών, οι οποίες μπορούν να χρησιμοποιηθούν σε τραπεζικές συναλλαγές, στην υγεία, στην εκπαίδευση κ.α., ενδεχομένως να οφείλεται στα μεγάλα ποσοστά γραφειοκρατίας του ελληνικού χώρου, γεγονός που καθιστά πάρα πολύ δύσκολη έως και ακατόρθωτη την ψηφιοποίηση των απαραίτητων εγγράφων και στοιχείων και στην απροθυμία των αρμοδίων να υλοποιήσουν αλλαγές στο σύνολο του συστήματος διασύνδεσης. Επιπλέον, οι Έλληνες παρουσιάζουν μία φοβική συμπεριφορά απέναντι σε τέτοιου είδους τεχνολογίες και σε κάθε τι καινοτόμο, η οποία πιθανότατα να οφείλεται είτε στην άγνοια είτε στην λανθασμένη χρήση είτε στις προκαταλήψεις των προηγούμενων ετών, για τον τεχνολογικό τομέα. Όμως, μία τέτοια συμπεριφορά δεν μπορεί να δικαιολογηθεί, αφού οι προαναφερθείσες τεχνολογίες προσφέρουν απόλυτη ασφάλεια στις συναλλαγές, εξαιτίας της μικρής απόστασης των συσκευών, των readers και των tags και της χρήσης ασφαλούς καναλιού μεταφοράς, καθιστώντας σχεδόν αδύνατη την υποκλοπή στοιχείων.

Περνώντας στον τομέα της ανάπτυξης εφαρμογών σε λογισμικό Android, είναι επιτακτική ανάγκη να αναφερθεί ότι τα βασικότερα σημεία της είναι αρχικά η ανάπτυξη πηγαίου κώδικα και δευτερευόντως η απασφαλμάτωση. Και αυτό γιατί, στην πρώτη περίπτωση συντάσσονται όλα τα απαραίτητα στοιχεία για τη δημιουργία και το τρέξιμο μίας εφαρμογής ενώ στην δεύτερη περίπτωση διορθώνονται στα λάθη του κώδικα ώστε να λειτουργήσει σωστά η εφαρμογή χωρίς σφάλματα σε λειτουργία debug· πραγματοποιείται δηλαδή ξανά compile της εφαρμογής. Αρωγό στην ανάπτυξη μίας εφαρμογής σε Android, αποτελεί το Android SDK, εξαιτίας των βιβλιοθηκών, των δειγμάτων κώδικα και το debugging που προσφέρει. Ακόμα ένας βοηθός στην σύνθεση και ανάπτυξη μίας Android εφαρμογής είναι και το Android Studio περιβάλλον.

Φτάνοντας στο περιβάλλον Qt επιβάλλεται να αναφερθεί ότι πρόκειται για ένα σύνολο βιβλιοθηκών, που αποτελούνται από ένα μεγάλο πλήθος κλάσεων και συναρτήσεων. Οι βιβλιοθήκες του περιβάλλοντος Qt, ένα ένα σύνολο από υποκατηγορίες προγραμματιστικών εργαλείων και μηχανισμών που ευθύνονται για τις συνδέσεις με συστήματα διαχείρισης βάσεων δεδομένων, για την ανάπτυξη περιβάλλοντος διεπαφής μίας εφαρμογής, για την ανάπτυξη εφαρμογών και για τον προγραμματισμό εφαρμογών δικτύου, για την ενσωμάτωση τρισδιάστατων γραφικών, για την χρήση μηχανισμών και για τη δημιουργία και απεικόνιση γραφικών. Επιπλέον, είναι το μοναδικό compile εργαλείο που μπορεί να μεταγλωττίσει απευθείας τη γλώσσα προγραμματισμού C++ σε android. Ο Qt Creator Συνεχίζοντας, ο Qt Creator, ως μέρος του SDK για το πλαίσιο

ανάπτυξης εφαρμογών Qt GUI, είναι αυτός που υποστηρίζει την σύνταξη κώδικα για διάφορες γλώσσες και αναλύσει τον κώδικα σε γλώσσες C++ και QML, άρα αποτελεί το κύριο εργαλείο δημιουργίας ίας εφαρμογής στο SDK.

Η εφαρμογή που δημιουργήθηκε στην παρούσα εργασία αποτελεί μία εφαρμογή με στοιχεία gamification. Εξάπτεται η φαντασία και η περιέργεια για το τι πρόκειται να ακολουθήσει, προσφέρει παρακίνηση για ενασχόληση καθ' όλη τη διάρκεια της περιπόλου και τονώνει την προσπάθεια του χρήστη για περαιτέρω απόκτηση τροφαίων. Επιπλέον, συμβάλλει στην μείωση του άγχους των εργαζομένων και στην αύξηση της παραγωγικότητάς τους. Η χρήση της τεχνολογίας NFC στην εφαρμογή επιχειρεί την εξοικείωση των Ελλήνων χρηστών με την νέα και σχεδόν άγνωστη τεχνολογία και την προώθησή της για χρήση σε ένα ευρύ φάσμα συναλλαγών και δραστηριοτήτων.

9. Επίλογος

Η παραπάνω εργασία επέτυχε την παρουσίαση των βασικότερων στοιχείων, της ιστορίας, της αρχιτεκτονικής και των εκδόσεων του λογισμικού Android, πάνω στο οποίο αναπτύχθηκε και η εφαρμογή Gaming. Επιπλέον, πραγματοποίησε μία ιστορική αναδρομή στον όρο του gamification, παρουσίασε τον τρόπο ανάπτυξής του, τα πλεονεκτήματα και τα μειονεκτήματά του και την έννοια της Οκτάλυσης. Ακόμα, απέδωσε με τον καλύτερο δυνατό τρόπο την ιστορία, τους τομείς εφαρμογής και τα πλεονεκτήματα της NFC τεχνολογίας, τον ορισμό του NFC reader, ανέλυσε τον τρόπο ανάπτυξης εφαρμογών σε περιβάλλον Android και ασχολήθηκε με τη χρήση του Qt περιβάλλοντος και περιέγραψε την εφαρμογή Gaming του εκπονητή.

Κατά την εκπόνηση της εργασίας αποκτήθηκαν γνώσεις προγραμματισμού στο λειτουργικό σύστημα Android και στην NFC τεχνολογία. Η ενασχόληση με την εφαρμογή που δημιουργήθηκε παρείχε υποστήριξη στην αξιολόγησή της και συνετέλεσε στην κατανόηση της λειτουργίας αρκετών αισθητηρίων των smartphones. Η ανάπτυξη της συγκεκριμένης Android εφαρμογής είναι μία απαιτητική διαδικασία, που επιβάλλει την άριστη γνώση της γλώσσας προγραμματισμού C++ αλλά και μεγάλη εμπειρία στον προγραμματισμό γενικότερα.

Βιβλιογραφία

- [1] Gargenta, M. (2011). *Building Applications for the Android Market*. USA: O'Reilly.
- [2] Lessard J. & Kessler, G. (2010). "Android forensics: Simplifying cell phone examinations". vol 4, p.p 1-12. Retrieved in 22/11 from: <https://ro.ecu.edu.au/ecuworks/6479>
- [3] Android. (2019, November 22). *What is android?*. Retrieved in 22/11 from Android: <https://www.android.com/the-platform/>
- [4] Yi, W.-J., Jia W. D. & Saniie, J. (2012). "Mobile Sensor Data Collector using Android Smartphone". *Proceedings of the 2012 IEEE 55th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Boise, ID, USA. 5-8 August; vol 5, pp. 5-8.
- [5] Gandhewar, N. & Sheikh, R. (2010). "Google Android: An emerging software platform for mobile devices". *Int.J.Computer Science*, vol 1, pp. 12-17.
- [6] Faiella, F. & Ricciardi, M. (2015). "Gamification and learning: a review of issues and research". *Journal of e-learning and Knowledge Society*, vol.11, pp. 13-21.
- [7] Vinichenko, M. V., Melnichuk, A. V., Kirillov, A. V., Makushkin, S. A. & Melnichul, Y. A. (2016). Modern views on the Gamification of Business, *Journal of Internet Banking and Commerce*, vol 21, pp. 1 – 13.
- [8] Kamasheva, A. V., Valeev, E. R., Yagudin, R. Kh. & Maksimova, K. R. (2015). Usage of Gamification Theory for Increase Motivation of Employees, *Mediterranean Journal of Social Sciences*, vol 6, pp. 77 – 80.
- [9] Unger, K., Schwartz, D. & Foucher, J. (2013). Increasing Employee Productivity through Gamification and Blended Learning. In T. Bastiaens & G. Marks (Eds.), *Proceedings of E-Learn 2013--World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education* (p.p. 2538-2545). Las Vegas, NV, USA: Association for the Advancement of Computing in Education (AACE). Retrieved July 7, 2019 from <https://www.learntechlib.org/primary/p/115272/>
- [10] Xu, F., Weber, J. & Buhalis, D. (2014). Gamification in Tourism. In *Information and Communication Technologies in Tourism 2014*, pp. 525-537.
- [11] Hamari, J., Koivisto, J. & Sarsa, H. (2014). Does Gamification Work? – A Literature Review of Empirical Studies on Gamification. In *proceedings of the 47th Hawaii International Conference On System Sciences*, Hawaii, USA, January 6-9.
- [12] Yu –kai C. (2019, November 22). Gamification & Behavioral Design. User and Player Types in Gamifird Systems. *WordPress*. Retrieved in 22/11 from : <https://yukaichou.com/gamification-study/user-types-gamified-systems/>
- [13] Yu-kai Chou. (2016). *Actionable Gamification: Beyond Points, Badges, and Leaderboards*. Canada: Leanpub.
- [14] Sailer, M., Mense, S. U., Mayr, S. K. & Mandl, H. (2017). How gamification motivates: An experimental study of the effects of specific game design elements on psychological need satisfaction. *Computer in Human Behavior*, vol 69, pp. 371-380.

- [15] Korn, O. & Schmidt, A. (2015). Gamification of business processes: Re-designing work in production and service industry. *Procedia Manufacturing*, vol 3, pp. 3424-3431.
- [16] Lucassen, G. & Janseu, S. (2014). Gamification in Consumer Marketing-Future or Fallacy?. *Procedia-Social and Behavioral Sciences*, vol 148, pp. 194-202.
- [17] Faiella, F. & Ricciardi, M. (2015). Gamification and learning: A review of issues and research, *Journal of e-Learning and Knowledge Society*, vol 11, pp. 13-21.
- [18] Anusha, R., Gokul, K., Unni, K., & Sethuraman, R. (2015). Near Field Communication (NFC) Technology. *A Survey*, vol. 4, pp. 133-144.
- [19] Badra, M., Serhrouchni, A. & Urien, P. (2004). A lightweight identity authentication protocol for wireless networks. *Computer Communication*, vol 27, pp. 1738-1745.
- [20] Badra, M. & Badra, R. (2016). A lightweight security protocol for NFC-based mobile payments. *Computer Science*, vol 83, pp. 705-711.
- [21] Want, R. (2011). Near Field Communication. *IEEE Pervasive Computing*, vol 10, pp. 4-7.
- [22] Nasution, S.M., Husni, E.M. and Wuryandari, A.I. (2012). *Prototype of train ticketing application using Near Field Communication (NFC) technology on Android device*. International Conference on System Engineering and Technology (ICSET), Bandung, pp. 1-6.
- [23] Vedat, C. Busra, O. & Kerem, O. (2013). A Survey on Ner Field Communication (NFC). *Technology Wireless Personal Communications*, vol 71, pp. 2259 – 2294.
- [24] Hongwei, Du. (2013). NFC Technology: Today and Tomorrow. *International Journal of Future Computer ND Communication*, vol 2, p.p 351 – 354.
- [25] Kai, G. (2007). Easy handling and security make NFC a success. *Card Technology Today*, vol 19, p.p 12 – 13.
- [26] Nath, B., Reynolds, F. & Want, R. (2006). RFID Technology and Applications. *IEEE Pervasive Computing*, vol 5, p.p 22 – 24.
- [27] Syed, A. & Mohammad, I. (2008). *RFID Handkbook Applications, Technology, Security and Privacy*. USA: Taylor & Francis Group CRC Press.
- [28] Mandeep, K., Manjeet, S., Neeraj, M. & Parvinder, S. S. (2011). RFID Technology Principles, Advantages, Limitations & Its Applications. *International Journal of Computer and Electrical Engineering*, vol 3, p.p. 151 – 157.
- [29] Nicholas, S. (2018). *Learn Qt 5: Build modern, responsive cross- platform desktop applications with Qt, C++, and QML*. Birmicham: Packt
- [30] Studio, (2019, November 22). Android Studio. Retrieved in 22/11 from: <http://developer.android.com/guide/developing/index.html>
- [31] Qt Platform: <https://www.qt.io/>
- [32] Introduction to Android : <http://developer.android.com/guide/index.html>.

- [33] Android API: <http://developer.android.com/reference/packages.html>
- [34] Introduction to QT: <https://wiki.qt.io/Main>
- [35] NFC Forum : <https://nfc-forum.org/>
- [36] XDA-Developers Forums: <http://forum.xda-developers.com/>
- [37] Android Development: <https://stackoverflow.com>
- [38] Android NDK tool: <https://developer.android.com/ndk>
- [39] Gamification Examples : <https://yukaichou.com/gamification-examples/gamification-stats-figures/>

ΠΑΡΑΡΤΗΜΑ Ι

Ακολουθεί ο κώδικας που βασίζεται η εφαρμογή για τον έλεγχο του δαχτυλικού αποτυπώματος και σχολιάζονται τα σημαντικότερα σημεία :

```
package com.example.mike.fingerprint;

import android.content.Intent;
import android.hardware.fingerprint.FingerprintManager;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;

import com.multidots.fingerprintauth.FingerPrintAuthCallback;
import com.multidots.fingerprintauth.FingerPrintAuthHelper;

public class MainActivity extends AppCompatActivity implements
FingerPrintAuthCallback{
    private FingerPrintAuthHelper mFingerPrintAuthHelper;

    @Override
    protected void onCreate(Bundle savedInstanceState) {// einai h
methodos pou trexei prin kseinhsei h othonh
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mFingerPrintAuthHelper = FingerPrintAuthHelper.getHelper(this,
this);// bibliothhkh tou fingerprint initiallize
    }

    @Override
    protected void onResume() {//otan ksanapas pali thn efarmogh gia na
mpeis
        super.onResume();
        mFingerPrintAuthHelper.startAuth();
    }

    @Override
    protected void onPause() {//methodos gia na menei sto background
        super.onPause();
        mFingerPrintAuthHelper.stopAuth();
    }
    // synarthseis tou fingerprint
    @Override
    public void onNoFingerprintHardwareFound() {

    }

    @Override
    public void onNoFingerprintRegistered() {

    }
    // den yparxei to fingerprint prin apo ekdosh marshmallow
    @Override
    public void onBelowMarshmallow() {

    }
    // exei mpei to fingerprint kai ginetai authorization
    @Override
```

```

    public void onAuthSuccess(FingerprintManager.CryptoObject
cryptoObject) {
        Intent launchIntent =
getPackageManager().getLaunchIntentForPackage("org.qtproject.example");//
ksekinaw thn allh efarmogh kai ksekinaei kainoyria diergasia
        if (launchIntent != null) {
            startActivity(launchIntent);
        }
    }
// exei mpei to fingerprint kai den anagnwrizetai
@Override
    public void onAuthFailed(int errorCode, String errorMessage) {
    }
}

```

ΠΑΡΑΡΤΗΜΑ Ι Ι

Ακολουθεί ο κώδικας της βάσης δεδομένων (database) των security.

```

#include "databasemanager.h"
#include "qmessagebox.h"
#include "qsqlquery.h"

```

```

DatabaseManager::DatabaseManager()

```

```

{
    db = QSqlDatabase::addDatabase("QSQLITE");
    db.setDatabaseName("/mnt/sdcard/test.sqlite");
    //db.setDatabaseName("C:/Users/mike/Desktop/test.sqlite");
    db.setHostName("");
    db.setPassword("");
    bool ok = db.open();
    //CREATE TABLE USERS
    //if(ok==false){
        QSqlQuery query("CREATE TABLE IF NOT EXISTS Users (id
INTEGER,username TEXT, password TEXT, firstname TEXT, lastname TEXT,
location TEXT, xp INTEGER, level INTEGER,badges TEXT,category TEXT)");
        query.exec();
        QSqlQuery query1("DELETE FROM Users;");
        query1.exec();
        QSqlQuery query2("INSERT INTO Users
VALUES(1,'secur1','sec1','mike','karanikas','ioannina',0,1,'badge1','Gree
k television presenters')");
        query2.exec();
        QSqlQuery query3("INSERT INTO Users
VALUES(2,'secur2','sec2','themis','spyridis','kastoria',0,1,'badge2','Gre
ek television presenters')");
        query3.exec();
        QSqlQuery query4("INSERT INTO Users
VALUES(3,'secur3','sec3','nikos','vrettoss','ioannina',0,1,'badge3','Gree
k television presenters')");
        query4.exec();
        QSqlQuery query5("INSERT INTO Users
VALUES(4,'secur4','sec4','dimitris','intiadis','thessaloniki',0,1,'badge4
','Greek television presenters')");
    }
}

```

```

query5.exec();
//CREATE TABLE NFC
TEXT, xp QSqlQuery nfc("CREATE TABLE IF NOT EXISTS nfc (id INTEGER,nfcname
INTEGER)");
nfc.exec();
QSqlQuery nfc1("DELETE FROM nfc");
nfc1.exec();
QSqlQuery nfc2("INSERT INTO nfc VALUES(1,'stage1',0)");
nfc2.exec();
QSqlQuery nfc3("INSERT INTO nfc VALUES(2,'stage2',0)");
nfc3.exec();
QSqlQuery nfc4("INSERT INTO nfc VALUES(3,'stage3',0)");
nfc4.exec();
QSqlQuery nfc5("INSERT INTO nfc VALUES(4,'stage4',0)");
nfc5.exec();
//CREATE TABLE PUZZLE
INTEGER, puzzlename TEXT, pieces INTEGER");
puzzle.exec();
QSqlQuery puzzle1("DELETE FROM puzzle");
puzzle1.exec();
QSqlQuery puzzle2("INSERT INTO puzzle VALUES(1,'puz1',5)");
puzzle2.exec();
QSqlQuery puzzle3("INSERT INTO puzzle VALUES(2,'puz2',5)");
puzzle3.exec();
QSqlQuery puzzle4("INSERT INTO puzzle VALUES(3,'puz3',5)");
puzzle4.exec();
QSqlQuery puzzle5("INSERT INTO puzzle VALUES(4,'puz4',5)");
puzzle5.exec();
//CREATE IMAGE WOMEN TABLE
TEXT, category TEXT");
image.exec();
QSqlQuery image1("DELETE FROM image");
image1.exec();
QSqlQuery image2("INSERT INTO image VALUES(1,':/Foreign
presenter/beyonce.jpg','Foreign presenters')");
image2.exec();
QSqlQuery image3("INSERT INTO image VALUES(2,':/Foreign
presenter/jlo.jpg','Foreign presenters')");
image3.exec();
QSqlQuery image4("INSERT INTO image VALUES(3,':/Foreign
presenter/eva mentes.jpg','Foreign presenters')");
image4.exec();
QSqlQuery image5("INSERT INTO image VALUES(4,':/Foreign
presenter/lagertha.jpg','Foreign presenters')");
image5.exec();
QSqlQuery image6("INSERT INTO image VALUES(5,':/Foreign
presenter/madona.jpg','Foreign presenters')");
image6.exec();
QSqlQuery image7("INSERT INTO image VALUES(6,':/Foreign
presenter/shakira.jpg','Foreign presenters')");
image7.exec();
QSqlQuery image8("INSERT INTO image VALUES(1,':/Greek television
presenters/elena Papari(500x500).jpg','Greek television presenters')");
image8.exec();
QSqlQuery image9("INSERT INTO image VALUES(2,':/Greek television
presenters/jenny-balatsinou(500x500).jpg','Greek television
presenters')");
image9.exec();

```

```

        QSqlQuery image10("INSERT INTO image VALUES(3,':/Greek television
presenters/korinthiou-homepage-image-500x500.jpg','Greek television
presenters')");
        image10.exec();
        QSqlQuery image11("INSERT INTO image VALUES(4,':/Greek television
presenters/nomikou(500x500).jpg','Greek television presenters')");
        image11.exec();
        QSqlQuery image12("INSERT INTO image VALUES(5,':/Greek television
presenters/pinelopi-anastasopoulou-homepage-image-500x500.jpg','Greek
television presenters')");
        image12.exec();
        QSqlQuery image13("INSERT INTO image VALUES(6,':/Greek television
presenters/zeta-makripoulia-homepage-image-500x500.jpg','Greek television
presenters')");
        image13.exec();
        QSqlQuery image14("INSERT INTO image VALUES(7,':/Greek television
presenters/μενεγακη(500x500).jpg','Greek television presenters')");
        image14.exec();

    }

//}

int DatabaseManager::check_login(QString username, QString password){
    int check = -1;
    QSqlQuery query(QString("SELECT id,username,password FROM Users;"));
    qDebug() << "error====" << query.lastError().text();
    while(query.next()){

if((username==query.value(1).toString()) && (password==query.value(2).toStr
ing())){
        check=query.value(0).toInt();

    }

    }
    return check;
}

QStringList DatabaseManager::get_info(int id){
    QStringList info;
    QSqlQuery query(QString("SELECT username, xp, level,badges,category
FROM Users WHERE id=%1").arg(id));
    while(query.next()){
        info.append(query.value(0).toString());
        info.append(query.value(1).toString());
        info.append(query.value(2).toString());
        info.append(query.value(3).toString());
        info.append(query.value(4).toString());

    }
    return info;
}

//update xp
void DatabaseManager::change_xp(int xp,int id)
{
    int x=0;

```

```

        QSqlQuery query(QString("SELECT xp FROM Users WHERE id=%1").arg(id));
        while(query.next()){
            x= query.value(0).toInt();
        }
        qDebug()<<"before xp====="<<x;
        x += xp;
        qDebug()<<"after xp====="<<x;
        //check update xp
        QSqlQuery query2(QString("UPDATE Users SET xp=%1 WHERE
id=%2").arg(x).arg(id));
        query2.exec();

    }
    //update level
    void DatabaseManager::change_level(int level,int id)
    {
        int y=0;
        QSqlQuery query(QString("SELECT level FROM Users WHERE
id=%1").arg(id));
        while(query.next()){
            y=query.value(0).toInt();
        }
        qDebug()<<"before level====="<<y;
        y += level;
        qDebug()<<"after level====="<<y;
        //check update level
        QSqlQuery query3(QString("UPDATE Users SET level=%1 WHERE
id=%2").arg(y).arg(id));
        query3.exec();

    }
    //epistrefw thn timh tou xp
    int DatabaseManager::get_xp(int id){
        int xp=0;
        QSqlQuery query(QString("SELECT xp FROM Users WHERE id=%1").arg(id));
        while(query.next()){
            xp= query.value(0).toInt();
        }
        return xp;
    }
    //epistrefw thn timh tou level
    int DatabaseManager::get_level(int id){
        int level=0;
        QSqlQuery query(QString("SELECT level FROM Users WHERE
id=%1").arg(id));
        while(query.next()){
            level=query.value(0).toInt();
        }
        return level;
    }
    //epistrefw to kommati puzzle
    QString DatabaseManager::get_puzzle_name(int id){
        QString puzzlename="";
        QSqlQuery query(QString("SELECT puzzlename FROM puzzle WHERE
id=%1").arg(id));
        while(query.next()){
            puzzlename=query.value(0).toString();
        }
        return puzzlename;
    }

```

```

}

int DatabaseManager::set_xp(int xp,int id){
    int z=0;
    QSqlQuery query(QString("SELECT xp FROM Users WHERE id=%1").arg(id));
    while(query.next()){
        z= query.value(0).toInt();
    }
    z =0;
    QSqlQuery query2(QString("UPDATE Users SET xp=%1 WHERE
id=%2").arg(z).arg(id));
    query2.exec();

    return z;
}

QString DatabaseManager::get_image(int level,QString category){
    QString path;
    QSqlQuery query(QString("SELECT path FROM image WHERE id=%1 AND
category='%2'").arg(level).arg(category));
    while(query.next()){
        path=query.value(0).toString();
    }
    return path;
}
//update category
bool DatabaseManager::set_category(int id,QString category){
    QString command = QString("UPDATE Users SET category='%1' WHERE
id=%2").arg(category).arg(id);
    QSqlQuery query;
    return query.exec(command);
}

QString DatabaseManager::get_category(int id){
    QString category = "";
    QString command = QString("SELECT category FROM Users WHERE
id=%1").arg(id);
    QSqlQuery query(command);
    while(query.next()){
        category=query.value(0).toString();
    }
    return category;
}

```

ΠΑΡΑΡΤΗΜΑ Ι Ι Ι

Ακολουθεί ο κώδικας ενεργοποίησης του nfc.

```

#include "nfc.h"
#include "ui_nfc.h"
#include "qmessagebox.h"
#include "qndefmessage.h"
#include "qndefrecord.h"

nfc::nfc(int id,puzzle *userPuzzle,QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::nfc)
{
    ui->setupUi(this);
    this->id = id;
}

```

```

        this->userPuzzle = userPuzzle;
        nfcManager=new QNearFieldManager(this);
        nfcManager-
>setTargetAccessModes(QNearFieldManager::NdefReadTargetAccess);
        //Get notified when the tag gets out of range

connect(nfcManager,SIGNAL(targetLost(QNearFieldTarget*)),this,SLOT(target
Lost(QNearFieldTarget*)));

connect(nfcManager,SIGNAL(targetDetected(QNearFieldTarget*)),this,SLOT(ta
rgetDetected(QNearFieldTarget*)));
        nfcManager->startTargetDetection();
}
void nfc::targetDetected( QNearFieldTarget *target)
{

        const bool hasNdefMessage = target->hasNdefMessage();
        if(hasNdefMessage){

connect(target,SIGNAL(ndefMessageRead(QNdefMessage)),this,SLOT(ndefMessag
eRead(QNdefMessage)));
        target->readNdefMessages();
        qDebug()<<"uri target detected!";

        }

        //cache target to member variable for writing
        cachedTarget=target;
        }

void nfc::targetLost(QNearFieldTarget *target)
{
        cachedTarget =NULL;
        target->deleteLater();
}
// read nfc message for nfc
void nfc::ndefMessageRead(const QNdefMessage &message)
{
        foreach(const QNdefRecord &record, message){
        if (record.isRecordType<QNdefNfcTextRecord>()){
                QNdefNfcTextRecord textRecord(record);
                ui->nfc_label->setText(textRecord.text());
                //metatroph tou int se qstring
                //QString string_id = QString("%1").arg(id);
                db->change_xp(50,id);
                int new_xp = db->get_xp(id);
                int new_level=db->get_level(id);
                qDebug()<<"new puzzle====="<<userPuzzle->return_puzzle();
                qDebug()<<"new xp====="<<new_xp;
                qDebug()<<"new level====="<<new_level;

                if(new_xp%200==0){

QMessageBox::critical(this,tr("Congratulation!!!!"),tr(" Collect Piece of
Puzzle "));
                qDebug()<<"Piece Collect";
                userPuzzle->increasePieces();
                }
                if (new_xp ==1000){

```

```

        new_xp = db->set_xp((new_xp - 1000), id);
        qDebug() << "poso xp" << new_xp;
    }
    if(new_xp%1000==0){
QMessageBox::critical(this, tr("Congratulation!!!"), tr("Level up"));
        qDebug() << "Level up";
        db->change_level(1, id);
        userPuzzle->increaseId();
    } //if (new_xp > 1000){
        // new_xp = db->get_xp(id);
        //new_level=db->get_level(id);
        //new_xp = db->set_xp((new_xp-1000), id);
        //qDebug() << "mphke edw" << new_xp;
        //}

    }

    }

}

void nfc::nfcWriteTag(const QString &nfcTagText){
    //the device currently has a target in reach
    if (cachedTarget){
        //convert text to a uri, adding missing http://if necessary
        QUrl convertedUrl=QUrl::fromUserInput(nfcTagText);
        //check if it is a valid url
        if (convertedUrl.isValid() && nfcTagText.contains('.')){
            QNdefMessage message;
            //create a uri ndef record
            QNdefNfcUriRecord uriRecord;
            uriRecord.setUri(convertedUrl);
            message.append(uriRecord);
            qDebug() << "Writting URI";
            //write ndef message to the tag
            cachedTarget->
            >writeNdefMessages(QList<QNdefMessage>() << message);

        }else{
            qDebug() << "no valid uri";
        }
    }else{
        qDebug() << "no cached target available";
    }
}

nfc::~nfc()
{
    delete ui;
}

```

ΠΑΡΑΡΤΗΜΑ IV

Ακολουθεί ο κώδικας δημιουργίας puzzle


```

#include "puzzle.h"
#include "ui_puzzle.h"

puzzle::puzzle(DatabaseManager *db,int id,QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::puzzle)
{
    ui->setupUi(this);
    this->id=id;
    this->puzzlename=db->get_puzzle_name(id);
}

QString puzzle::return_puzzle(){
    return puzzlename;
}

void puzzle::set_xp_and_level(int level, int xp){
    int remainxp=xp-((level-1)*1000);
    if(xp<200){
        this->pieces=0;
    }else if(xp<400){
        this->pieces=1;
    }else if(xp<600){
        this->pieces=2;
    }else if(xp<800){
        this->pieces=3;
    }else if(xp<1000){
        this->pieces=4;
    }

    ui->puzzlename->setText(puzzlename);
    ui->puzzlename->setReadOnly(true);
    ui->pieces->setText(QString("%1").arg(pieces));
    ui->pieces->setReadOnly(true);
}

void puzzle::increasePieces(){
    pieces++;
}

void puzzle::increaseId(){
    pieces=0;
    //id puzzle=level user
    id++;
    this->puzzlename=db->get_puzzle_name(id);
}

puzzle::~puzzle()
{
    delete ui;
}

```

ΠΑΡΑΡΤΗΜΑ V

Ακολουθεί ο κώδικας δημιουργίας profile

```

#include "profile.h"
#include "ui_profile.h"

profile::profile(int id, DatabaseManager *db, QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::profile)
{
    ui->setupUi(this);
    this->id=id;
    info=db->get_info(id);
    ui->id->setText(QString("%1").arg(id));
    ui->id->setReadOnly(true);
    ui->username->setText(info[0]);
    ui->username->setReadOnly(true);
    ui->level->setText(info[2]);
    ui->level->setReadOnly(true);
    ui->xp->setText(info[1]);
    ui->xp->setReadOnly(true);
    ui->badges->setText(info[3]);
    ui->badges->setReadOnly(true);
    ui->Category->setText(info[4]);
    ui->Category->setReadOnly(true);
}

profile::~profile()
{
    delete ui;
}

void profile::on_pushButton_clicked()
{
    QString link="http://www.facebook.com";
    QDesktopServices::openUrl(QUrl(link));
}

```

ΠΑΡΑΡΤΗΜΑ VI

Ακολουθεί ο κώδικας δημιουργίας gallery φωτογραφιών

```

#include "gallery.h"
#include "ui_gallery.h"

gallery::gallery( DatabaseManager *db, int id, int level, int xp, QWidget
*parent) :
    QMainWindow(parent),
    ui(new Ui::gallery)
{
    ui->setupUi(this);

    if (level>1){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek television
presenters/elena Papari(500x500).jpg")));
    }
    if (level>2){

```

```

        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou(500x500).jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari(500x500).jpg")));
    }
    if (level>3){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/zeta-makripoulia-homepage-image-500x500.jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari(500x500).jpg")));
        ui->pic3->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou(500x500).jpg")));
    }
    if (level>4){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/zeta-makripoulia-homepage-image-500x500.jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari(500x500).jpg")));
        ui->pic3->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou(500x500).jpg")));
        ui->pic4->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/korinthiou-homepage-image-500x500.jpg")));
    }
    if (level>5){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/nomikou(500x500).jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/zeta-makripoulia-homepage-image-500x500.jpg")));
        ui->pic3->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari(500x500).jpg")));
        ui->pic4->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou(500x500).jpg")));
        ui->pic5->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/korinthiou-homepage-image-500x500.jpg")));
    }
    if (level>6){
        ui->pic1->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/nomikou(500x500).jpg")));
        ui->pic2->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/zeta-makripoulia-homepage-image-500x500.jpg")));
        ui->pic3->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/elena Papari(500x500).jpg")));
        ui->pic4->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/jenny-balatsinou(500x500).jpg")));
        ui->pic5->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/korinthiou-homepage-image-500x500.jpg")));
        ui->pic6->setPixmap(QPixmap::fromImage(QImage(":/Greek
television presenters/pinelopi-anastasopoulou-homepage-image-
500x500.jpg")));
    }
}

gallery::~gallery()
{
    delete ui;
}

```

ΠΑΡΑΡΤΗΜΑ VII

Ακολουθεί ο κώδικας εμφάνισης εικόνας puzzle.

```
#include "graphics.h"
#include "ui_graphics.h"

graphics::graphics(int id, DatabaseManager *db,int pieces,int
level,QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::graphics)
{
    ui->setupUi(this);
    QPixmap pix;
    this->pieces=pieces;
    width = pix.width()/10;
    height = pix.height()/10;
    square_x=0;
    square_y=0;
    pscene=new QGraphicsScene(this);
    ui->graphicsView-
>setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
    ui->graphicsView-
>setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
    qDebug() <<"level====" <<level;
    category = db->get_category(id);
    qDebug() <<"category mikele=====" <<category;
    change_image(level,category);
    set_category(id,category);
    pscene->addItem(item);
    ui->graphicsView->setScene(pscene);
    //pscene->setSceneRect(0,0,500,500);
    if(pieces==0){
        pscene-
>addRect(250,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(0,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(250,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(0,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        qDebug() <<"pieces 0";
    }else if(pieces==1){
        pscene-
>addRect(0,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(250,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(0,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        qDebug() <<"pieces 1";
    }else if(pieces==2){
        pscene-
>addRect(250,0,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        pscene-
>addRect(0,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        qDebug() <<"pieces 2" ;
    }else if (pieces==3){
```

```

        pscene-
>addRect (0,250,250,250,QPen(Qt::NoPen),QBrush(QColor(Qt::black)));
        qDebug()<<"pieces 3";
    }

    qDebug()<<pix.width();
    qDebug()<<pix.height();

}

void graphics::change_image(int level,QString category){
    QString path = db->get_image(level,category);
    QImage *img=new QImage();
    img->load(path);
    qDebug()<<"path here"<<path;
    item = pscene->addPixmap(QPixmap::fromImage(QImage(*img)));
    pscene->addItem(item);
    ui->graphicsView->setScene(pscene);
}

void graphics::set_category(int id,QString category){
    qDebug()<<"set category"<<category;
    category=db->set_category(id,category);
}

graphics::~graphics()
{
    delete ui;
}

```

ΠΑΡΑΡΤΗΜΑ VIII

Ακολουθεί ο κώδικας διάφορων κατηγοριών από εικόνες.

```

#include "categories.h"
#include "ui_categories.h"

Categories::Categories(int id,DatabaseManager *db,int level,QWidget
*parent) :
    QMainWindow(parent),
    ui(new Ui::Categories)
{

    ui->setupUi(this);
    this->db=db;
    this->id=id;
    current_category = "";
    view = new QListView();
    // Create model
    model = new QStringListModel(this);
}

```

```

    // Make data
    QStringList List;
    List << "Greek television presenters" << "Foreign presenters" <<
    "Global security staff " << "Landscapes";
    QStandardItemModel *itemmodel = new QStandardItemModel;
    for (int i = 0; i < List.count(); i++)
    {

        QStandardItem *item = new QStandardItem();
        item->setText(List[i]);
        item->setSizeHint(QSize(item->sizeHint().width(),100));
        itemmodel->setItem(i, item);
        view->setModel(itemmodel);

    }

    // Populate our model
    model->setStringList(List);
    ui->listView->setModel(itemmodel);
    ui->listView->setEditTriggers(QAbstractItemView::NoEditTriggers);
    connect(ui-
>listView, SIGNAL(clicked(QModelIndex)), this, SLOT(get_selected()));
    connect(ui->Save, SIGNAL(clicked(bool)), this, SLOT(save_category()));
}

Categories::~Categories()
{
    delete ui;
}

void Categories::on_Save_clicked()
{
    QMessageBox::information(this, tr("Checkable list
model"), tr("Required components were saved."), QMessageBox::Ok);
    qDebug() << "saved====" << db->set_category(id, ui->listView-
>selectionModel()->selectedIndexes().at(0).data().toString());
}

void Categories::save_category() {

}

void Categories::get_selected() {

    qDebug() << "select category====" << ui->listView->selectionModel()-
>selectedIndexes().at(0).row();
    current_category = ui->listView->selectionModel()-
>selectedIndexes().at(0).data().toString();
    qDebug() << "current category====" << current_category;
    QString path = db->get_image(level, category);
    QImage *img=new QImage();
    img->load(path);}

```