



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Ανάπτυξη εφαρμογής γνώσεων για Android – Kn-owl-edge Development of a quiz application for Android – Kn-owl-edge
Όνοματεπώνυμο Φοιτητή	ΓΕΩΡΓΙΟΣ ΘΕΟΧΑΡΗΣ
Πατρώνυμο	ΔΗΜΗΤΡΙΟΣ
Αριθμός Μητρώου	ΜΠΠΛ/ 12012
Επιβλέπων	ΜΑΡΙΑ ΒΙΡΒΟΥ, ΚΑΘΗΓΗΤΡΙΑ

Ημερομηνία Παράδοσης **Νοέμβριος 2015**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Μαρία Βίρβου
Καθηγήτρια

Γεώργιος Α. Τσιχριντζής
Καθηγητής

Ευθύμιος Αλέπης
Επίκουρος Καθηγητής

Tell me and I will forget. Show me and I may remember. Involve me and I will understand.

Confucius

Μαθαίνεις περισσότερο σε μια ώρα παιχνιδιού, παρά σε ένα χρόνο συζήτησης.

Πλάτωνας

Play is the highest form of research.

Albert Einstein

Ευχαριστίες:

Για την περάτωση της παρούσας μεταπτυχιακής διατριβής, συνέβαλλαν ορισμένα άτομα των οποίων η προσπάθεια και η υποστήριξη ήταν αξιοσημείωτη.

Επομένως με την ευκαιρία αυτή, θα ήθελα να ευχαριστήσω θερμά:

- Την κα. Βίββου Μαρία, για την επίβλεψη της εργασίας, αλλά και για την ενεργώ συμβολή της στην επιτυχημένη διεξαγωγή του μεταπτυχιακού προγράμματος με τίτλο «Πληροφορική».
- Τον κ. Παπαδημητρίου Σπύρο για τον συμβουλευτικό του ρόλο και την ουσιαστική καθοδήγηση κατά την εκπόνηση της εργασίας.
- Τον κ. Λεοντή Αριστείδη, πτυχιούχο Βιολογίας και μεταπτυχιακό στην «Πρόληψη λοιμώξεων» για τη συνεργασία και την επιμέλεια των ερωτήσεων αλλά και της ύλης την οποία καλύπτει η εφαρμογή.

Επιπλέον θα ήθελα να ευχαριστήσω την οικογένειά μου για τη δυνατότητα που μου παρείχαν να συμμετάσχω στο μεταπτυχιακό πρόγραμμα του Πανεπιστημίου Πειραιώς αλλά και την στήριξή τους κατά τη διάρκειά του.

Τέλος, θα ήθελα να ευχαριστήσω την Βολυράκη Αθηνά για την βοήθεια στην συγγραφική επιμέλεια του γραπτού κειμένου.

Συνοπτική Περίληψη

Η παρούσα μεταπτυχιακή διατριβή έχει ως στόχο να αναδείξει τις δυνατότητες που παρουσιάζονται από τη συνεχή εξέλιξη της τεχνολογίας και να συνδέσει την επιστήμη της Πληροφορικής με τον κλάδο της εκπαίδευσης και της ψυχαγωγίας. Με την άνοδο των «έξυπνων» κινητών και την εμφάνιση ενός κικεώνα εφαρμογών στα αντίστοιχα «ηλεκτρονικά» τους καταστήματα, δίνεται η ευκαιρία για την ανέλιξη ενός νέου είδους εκπαιδευτικών εφαρμογών που σκοπεύει όχι μόνο στην εκπαίδευση του χρήστη, αλλά και στην ψυχαγωγία του. Από αυτή τη συγχώνευση προκύπτει η ιδέα του Kn-owl-edge. Το Kn-owl-edge είναι ένα είδος εκπαιδευτικού παιχνιδιού, το οποίο προσφέρει στο χρήστη, κυρίως τους μαθητές που δίνουν ή προετοιμάζονται για να δώσουν πανελλήνιες, ένα οικείο περιβάλλον, χρησιμοποιώντας τεχνικές τις οποίες δανείζεται από το μέσο των ηλεκτρονικών παιχνιδιών. συνδυάζοντας την εμπειρία αυτή με την επανάληψη της ύλης.

Abstract

The current master dissertation aims to showcase the possibilities that arise from the continuous technological progress and tries to combine the field of Computer Science with that of education and entertainment. With the advent of smartphones and the appearance of a constellation of mobile applications on the respective mobile stores, a new opportunity arises, that of a new kind of app that targets not only at educating the user, but also to entertain him on an equal level while doing so. From that convergence, the idea of Kn-owl-edge has been given birth. K-owl-edge is an educational game that offers to the players, mainly students that are preparing for the Greek National Exams, the feeling of a familiar environment that borrows techniques from the medium of computer games and incorporate it with ways that enables them to focus on their syllabus.

Πίνακας περιεχομένων

Κεφάλαιο 1 - Εισαγωγή	8
1.1 Ιστορική αναδρομή	8
1.2 Μορφές πληροφορικής που βοηθούν στην εκπαίδευση	9
1.3 Ηλεκτρονικά παιχνίδια	10
1.4 Η συμβολή του παιχνιδιού στην εκπαίδευση	10
1.5 Παραδείγματα ηλεκτρονικών εκπαιδευτικών εφαρμογών	12
1.5.1 Ένα παιχνίδι με άρωμα αρχαίας Ελλάδας	12
1.5.2 Quizdom: Ένα παιχνίδι γνώσεων	13
1.5.3 Ελληνικές εκπαιδευτικές εφαρμογές	14
Κεφάλαιο 2 - Ανάλυση της Εφαρμογής Kn-owl-edge	14
2.1 Kn-owl-edge	15
2.2 Άξονας Εκπαίδευσης	18
2.3 Άξονας Ψυχαγωγίας	19
Κεφάλαιο 3 - Τεχνολογίες που χρησιμοποιήθηκαν	20
3.1 Λειτουργικό Σύστημα Android	21
3.2 Πλατφόρμα Ανάπτυξης Android Studio	22
3.3 Αρχιτεκτονικό Πρότυπο Προγραμματισμού Android	23
3.3.1 Πρότυπο Αρχιτεκτονικής MVC	23
3.3.2 Αρχιτεκτονική Προγραμματισμού Android	24
3.4 Back End	28
3.5 Υλοποίηση Kn-owl-edge	32
3.5.1 Αρχιτεκτονική Εφαρμογής Kn-owl-edge	32
3.5.2 Βιβλιοθήκες, Plugins	44
Κεφάλαιο 4 – Επίλογος	46
4.1 Συμπεράσματα	46
4.2 Προτάσεις για μελλοντική ανάπτυξη της εφαρμογής Kn-owl-edge	47
Βιβλιογραφία	48
Ιστογραφία	50

Πίνακας εικόνων

Εικόνα 1 Serious Game Ιατρική Προσομοίωση	11
Εικόνα 2 Minecraft Αρχαία Ελλάδα	12
Εικόνα 3 Quizdom	13
Εικόνα 4 Λογότυπο Εφαρμογής Kn-owl-edge.....	15
Εικόνα 5 Διαδικασία Εγγραφής μέσω email.....	16
Εικόνα 6 Επιλογή Κεφαλαίων	17
Εικόνα 7 Ενότητες και Αστέρια.....	17
Εικόνα 8 Παράδειγμα Ερώτησης.....	18
Εικόνα 9 Τεχνική Gamification	20
Εικόνα 10 Λογότυπο Λειτουργικού Android	21
Εικόνα 11 Αρχιτεκτονική Λειτουργικού Android.....	22
Εικόνα 12 Πλατφόρμα Ανάπτυξης Android Studio.....	22
Εικόνα 13 Πρότυπο Αρχιτεκτονικής MVC	24
Εικόνα 14 Main Activity Kn-owl-edge	25
Εικόνα 15 Κύκλος Ζωής Activity.....	26
Εικόνα 16 Relative Layout.....	26
Εικόνα 17 Horizontal Linear Layout	27
Εικόνα 18 Cloud Service Parse.....	28
Εικόνα 19 Cloud Platform Parse	28
Εικόνα 20 Chapter Pointer	29
Εικόνα 21 Δεδομένα Pointer.....	29
Εικόνα 22 Ημερήσιες Εγκαταστάσεις Kn-owl-edge.....	32
Εικόνα 23 Questions Exporter Form	45
Εικόνα 24 Json Format.....	46

Κεφάλαιο 1 - Εισαγωγή

Στη σύγχρονη εποχή όπου η τεχνολογία έχει εισχωρήσει για τα καλά στη ζωή μας οι νέοι σε όλο τον κόσμο χρησιμοποιούν καθημερινά διάφορες ηλεκτρονικές συσκευές όπως Η/Υ, κινητά, tablets, laptops κ.α. οι οποίες απλοποιούν τη ζωή και τις συνήθειες τους. Τα κινητά τηλέφωνα (smartphones) και τα tablets, είναι πλέον ευρέως διαδεδομένα και προσφέρουν ποικίλους τρόπους ψυχαγωγίας, ενημέρωσης, αλλά και εκπαίδευσης.

Στην παρούσα μεταπτυχιακή διατριβή θα ασχοληθούμε με την ανασκόπηση της συμβολής των ηλεκτρονικών εφαρμογών που διατίθενται για κινητά τηλέφωνα στην εκπαίδευση, αλλά και την προσέγγιση τους μέσω ενός διαφορετικού τομέα, αυτού των ηλεκτρονικών παιχνιδιών.

Θα αναλύσουμε τους τρόπους με τους οποίους συμβάλλουν τα κινητά τηλέφωνα στην εκπαίδευση, θα παρουσιάσουμε παραδείγματα εκπαιδευτικών εφαρμογών και παιχνιδιών, αλλά και θα δημιουργήσουμε μια εφαρμογή με σκοπό την εκπλήρωση των παραπάνω στόχων.

1.1 Ιστορική αναδρομή

Η χρήση των τεχνολογιών στην εκπαίδευση χωρίζεται χρονικά στις ακόλουθες φάσεις.

Η Πρώτη Φάση χρονολογικά τοποθετείται πριν το 1970. Τότε για πρώτη φορά έκαναν την εμφάνισή τους στο χώρο της εκπαίδευσης τα Μέσα (media) και οι τεχνολογίες όπως ο κινηματογράφος, το ραδιόφωνο και η εκπαιδευτική τηλεόραση. Σταδιακά κατασκευάστηκαν προγραμματισμένα βιβλία και χρησιμοποιήθηκαν οι διδακτικές μηχανές.

Η Δεύτερη Φάση που σχετίζεται με την πιο εντατική πληροφορική προσέγγιση της εκπαίδευσης, ξεκινά το 1970 και φτάνει μέχρι το 1980. Οι άνθρωποι στρέφουν το ενδιαφέρον τους στη Μάθηση του Προγραμματισμού και ασχολούνται εκτεταμένα με τους υπολογιστές και τις γλώσσες προγραμματισμού (γλώσσα Logo & Basic). Αυτό συμβαίνει στο πλαίσιο της προσπάθειάς τους να κάνουν τη διδασκαλία πιο ελκυστική με τη χρήση του υπολογιστή (Δι.Β.Υ. - Computer Assisted Instruction, CAI / Μα.Β.Υ. - Computer Assisted Learning, CAL) ως διδακτικό εργαλείο και εργαλείο μάθησης. Η χρήση αυτή αρχικά βρίσκεται σε πειραματικό στάδιο με την πιλοτική εφαρμογή στη Δευτεροβάθμια Εκπαίδευση και κυρίως στο Λύκειο. Η πλειονότητα των εκπαιδευτικών προγραμμάτων κατά την περίοδο εκείνη είναι προγράμματα εξάσκησης και πρακτικής εφαρμογής (drill and practice) και πολύ λίγα είναι αυτά που αφορούν εναλλακτικές εφαρμογές (όπως προγράμματα προσομοιώσεων και έμπειρα διδακτικά συστήματα).

Η Τρίτη Φάση (1980 -1990) χαρακτηρίζεται από μια τάση γενικευμένης εισαγωγής. Η Πληροφορική γίνεται μέσο και αντικείμενο διδασκαλίας και παρατηρείται σημαντική παραγωγή εκπαιδευτικού λογισμικού διαφόρων τύπων. Η αρχή έγινε στην Αγγλία με το πρόγραμμα «Υπολογιστές στα Σχολεία» (Micros in Schools) και στη συνέχεια στη Γαλλία που είχε τον τίτλο «Πληροφορική Για Όλους» (Informatique Pour Tous - IPT). Παρόμοια προγράμματα εισαγωγής των τεχνολογιών στην Εκπαίδευση γίνονται στις ΗΠΑ καθώς και σε άλλες αναπτυγμένες χώρες του κόσμου. Πιο συγκεκριμένα, στις ΗΠΑ η ενέργεια αυτή ενισχύθηκε κατά πολύ από ερευνητές και εκπαιδευτικούς αλλά και από τη βιομηχανική αγορά. Σταδιακά η εισαγωγή της πληροφορικής εφαρμόζεται σε όλες τις εκπαιδευτικές βαθμίδες και αυτό ενισχύεται ιδιαίτερα από την εξάπλωση των προσωπικών υπολογιστών, την μείωση των τιμών τους καθώς και λόγω της επερχόμενης πληροφορικής επανάστασης.

Εν κατακλείδι, μετά το 1990 φτάνουμε στην Τέταρτη Φάση. Σε αυτή τη περίοδο τα τεχνολογικά ερεθίσματα είναι πολλά. Όλοι και κυρίως οι νέοι χρησιμοποιούν το Διαδίκτυο, τους υπολογιστές, τα πολυμέσα και υπερμέσα. Χρησιμοποιούνται πολλά συνεργατικά περιβάλλοντα μάθησης, προσομοιώσεις και την εμφάνισή της κάνει και η εικονική πραγματικότητα. Έκτοτε η τεχνολογία έχει διεισδύσει σε μεγάλο βαθμό στην καθημερινότητά μας. Όποιος δε γνωρίζει να χρησιμοποιεί τις τεχνολογίες, θεωρείται «αναλφάβητος».

1.2 Μορφές πληροφορικής που βοηθούν στην εκπαίδευση

Οι άνθρωποι και οι κοινωνίες εξελίσσονται στο πέρασμα του χρόνου. Αναπόφευκτα λοιπόν συμβαίνουν αλλαγές και στην εκπαίδευση, η οποία έχοντας αποκτήσει μαθητοκεντρικό χαρακτήρα, πρέπει να διαμορφώνεται σύμφωνα με τα ενδιαφέροντα των μαθητών. Στο πλαίσιο της συνεχούς προσπάθειας για τη δημιουργία ενός αποτελεσματικότερου και ελκυστικότερου περιβάλλοντος μάθησης και διδασκαλίας, τα τελευταία χρόνια οι εκπαιδευτικοί χρησιμοποιούν στις διάφορες σχολικές μονάδες πλήθος τεχνολογικών εφαρμογών. Πολλά επιμορφωτικά προγράμματα γίνονται προκειμένου οι εκπαιδευτικοί να είναι όσο το δυνατόν καλύτερα ενημερωμένοι και εξοικειωμένοι με τα τεχνολογικά μέσα. Ένα χαρακτηριστικό παράδειγμα είναι το πρόγραμμα «Επιμόρφωση Εκπαιδευτικών για την αξιοποίηση και εφαρμογή των ΤΠΕ στην σχολική τάξη».

Ένα τέτοιου είδους τεχνολογικό μέσο είναι ο διαδραστικός πίνακας, ο οποίος σε συνδυασμό με τις κατάλληλες παιδαγωγικές προϋποθέσεις και την αλληλεπίδραση εκπαιδευτικών και μαθητών, μπορεί να οδηγήσει στο επιθυμητό αποτέλεσμα και στην επίτευξη μαθησιακών στόχων. Ουσιαστικά, με ένα μόνο άγγιγμα δίνει τη δυνατότητα αλληλεπίδρασης με αντικείμενα που εμφανίζονται στην επιφάνειά του όπως για παράδειγμα να γράψει με ειδικό μαρκαδόρο, να περιστρέψει ή να μετακινήσει αντικείμενα με το χέρι. Μερικά από τα πλεονεκτήματα που προσφέρει είναι ότι μπορούν να συνδυαστούν με εκπαιδευτικό λογισμικό για την κατασκευή πλάνων μαθημάτων, εντυπωσιακών παρουσιάσεων και σημειώσεων, εποπτικών ασκήσεων και παιχνιδιών. Με αυτόν τον ευρηματικό τρόπο ο εκπαιδευτικός βάζει τις βάσεις για έναν διαφορετικό τρόπο παρακολούθησης ενός μαθήματος ο οποίος τείνει να είναι πιο φιλικός προς τον μαθητευόμενο.

Η χρήση συσκευών κινητής τεχνολογίας (smartphones, tablets, laptops κ.α.) στη διδασκαλία και τη μάθηση αναμένεται ότι θα αυξηθεί ραγδαία στα επόμενα χρόνια και θα ελαττωθεί αισθητά τα σημερινά σχολικά βιβλία. Στην Ελλάδα ήδη έχουν ξεκινήσει προγράμματα την τελευταία πενταετία, όπως το πρόγραμμα «ένας φορητός υπολογιστής ανά μαθητή» (στο γυμνάσιο) ή το πρόγραμμα «κινητά εργαστήρια Η/Υ» για τα δημοτικά σχολεία τύπου ΕΑΕΠ. Επιπρόσθετα, τα τελευταία χρόνια αρκετοί ερευνητές έχουν επικεντρωθεί στο ερευνητικό κομμάτι αυτής της διαδικασίας αναζητώντας καινοτόμες εφαρμογές που θα συμβάλλουν στην βελτίωση της εκπαιδευτικής εμπειρίας και σε επιμόρφωση υψηλού επιπέδου.

Το εκπαιδευτικό λογισμικό είναι το λογισμικό που εκτελείται σε υπολογιστή, το οποίο περιλαμβάνει διδακτικούς στόχους και προσδοκά να επιφέρει συγκεκριμένα διδακτικά και μαθησιακά αποτελέσματα. Λειτουργεί υποστηρικτικά παρέχοντας βοήθεια στο μαθητή ώστε να αποκτήσει το μέγιστο δυνατό επίπεδο γνώσης. Το εκπαιδευτικό λογισμικό διαχωρίζεται με κριτήριο τη χρήση του στη μαθησιακή διαδικασία, ως εξής: Λογισμικό εξάσκησης (Drill and Practice). Λογισμικό παρουσίασης - φροντιστηρίου (Tutorial). Λογισμικό προσομοίωσης (Simulation). Λογισμικό επίλυση προβλήματος (Problem solving). Περιβάλλοντα Εικονικής Πραγματικότητας (Virtual Reality).

Η αξιοποίηση των εκπαιδευτικών παιχνιδιών στη μάθηση αποτελεί ένα ακόμα εργαλείο τόσο για μαθητές όσο και για εκπαιδευτικούς. Τα ψηφιακά παιχνίδια είναι η δημοφιλέστερη μορφή τεχνολογίας στην παιδική και νεανική διασκέδαση. Ταυτόχρονα, μπορούν να προσφέρουν θετικά αποτελέσματα όσο αφορά την κινητοποίηση των εκπαιδευόμενων και την αποτελεσματικότητα της εκπαιδευτικής διαδικασίας. Η μάθηση που βασίζεται στα ψηφιακά παιχνίδια αποτελεί ένα ιδιαίτερο ερευνητικό πεδίο μέσα στο ευρύτερο πλαίσιο της εκπαίδευσης με ΤΠΕ και προσελκύει το έντονο ενδιαφέρον της εκπαιδευτικής και επιστημονικής κοινότητας με συχνά αντικρουόμενες απόψεις.

Τέλος, η χρήση του διαδικτύου στην εκπαίδευση, μπορεί να συμβάλλει καθοριστικά στη διδασκαλία των μαθητών από τις πρώτες τάξεις του δημοτικού έως και τις τελευταίες του λυκείου. Ενδεικτικό παράδειγμα, είναι η σωστή κατηγοριοποίηση του εκπαιδευτικού υλικού με τεχνολογίες file sharing ή άλλες σχετικές διαδικτυακές τεχνολογίες. Επίσης διαδικτυακά εργαλεία θα μπορούσαν να συμβάλλουν στην όσο το δυνατόν καλύτερη αξιολόγηση της εκπαιδευτικής διαδικασίας γενικότερα.

Όλες οι παραπάνω διαφορετικές εκδοχές των μέσων πληροφορικής μπορούν να βελτιώσουν σημαντικά την ποιότητα της εκπαίδευσης, να μεγιστοποιήσουν το ενδιαφέρον των μαθητών και να τους κινητοποιήσουν σημαντικά στον δύσκολο δρόμο που έχουν να διανύσουν

στην κατάκτηση της γνώσης. Οργανώνοντας τον καταγισμό των πληροφοριών που δέχονται οι μαθητές σε όλα τα χρόνια της φοίτησής τους, οι εκπαιδευτικοί βελτιώνουν σημαντικά την ίδια την δουλειά τους.

1.3 Ηλεκτρονικά παιχνίδια

Το μέσο των ηλεκτρονικών παιχνιδιών έκανε την εμφάνισή του στις αρχές της δεκαετίας του '80, προκαλώντας έκρηξη στον τομέα της ψυχαγωγίας. Κατά την διάρκεια της δεκαετίας του '90 τα ηλεκτρονικά παιχνίδια αποτέλεσαν τμήμα της καθημερινότητας μικρών και μεγάλων.

Ιδιαίτερα σημαντικοί λόγοι στους οποίους οφείλεται η ταχύτατη εξάπλωση των ηλεκτρονικών παιχνιδιών είναι αφενός το **κέρδος** στο οποίο στοχεύουν οι αντίστοιχες βιομηχανίες. Ανέκαθεν, ευάλωτες κοινωνικές ομάδες όπως αυτές των παιδιών και των εφήβων, αποτελούσαν «στόχο» για τις βιομηχανίες των ΜΜΕ, της μόδας, της μουσικής κ.λπ. Οι νέοι άνθρωποι θεωρούνται «κινητήρια δύναμη» για οποιοδήποτε νέο προϊόν στην αγορά, μιας και η αγοραστική τους δύναμη είναι αξιοσημείωτη (Fromme, 2003). Αφετέρου είναι η εξοικείωση των νεότερων ηλικιακά ομάδων με την νέα τεχνολογία. Η ραγδαία εξάπλωση των ηλεκτρονικών παιχνιδιών συνέβαλε ώστε σε σύντομο χρονικό διάστημα να αποτελέσουν σημείο αναφοράς όχι μόνο σαν μέσο ψυχαγωγίας, αλλά και σαν πεδίο ερευνητικού ενδιαφέροντος (Cianvarro, 2006).

Σχετική μελέτη των Beentjes και Johannes (2001) η οποία έλαβε χώρα σε 11 ευρωπαϊκά κράτη κατά τα έτη 1997-98, σε παιδιά μεταξύ 6 και 16 ετών εξέτασε τον αριθμό των λεπτών που αφιερώνεται ανά ημέρα σε τρία διαφορετικά αλληλεπιδραστικά μέσα: το διαδίκτυο, τον ηλεκτρονικό υπολογιστή (για εργασίες εκτός παιχνιδιών) και τα ηλεκτρονικά παιχνίδια. Τα αποτελέσματα έδειξαν ότι αφιέρωναν 32 λεπτά ανά ημέρα παίζοντας ηλεκτρονικά παιχνίδια, 17 λεπτά χρησιμοποιώντας διάφορες εφαρμογές στον ηλεκτρονικό υπολογιστή (εκτός παιχνιδιών) και 5 λεπτά για πλοήγηση στο διαδίκτυο Παρόλο που υπήρχε διαφοροποίηση των αποτελεσμάτων ανά χώρα, είναι αξιοσημείωτο ότι ως επί των πλείστων, ο χρόνος αφορούσε την ενασχόληση με τα ηλεκτρονικά παιχνίδια.

Μια ακόμα έρευνα που επιβεβαιώνει την υψηλή απήχηση των ηλεκτρονικών παιχνιδιών πραγματοποιήθηκε από τους Feierabend και Klingler το 2001, σε δείγμα 740 παιδιών μεταξύ 6 και 13 ετών. Αντίστοιχα, διαπιστώθηκε ότι η ενασχόληση των νέων με τα ηλεκτρονικά παιχνίδια αποτελεί την πιο σημαντική μεταξύ όλων των δραστηριοτήτων που λαμβάνουν χώρα σε ένα ηλεκτρονικό υπολογιστή (Fromme, 2003).

Εύλογα λοιπόν, ειδικοί σε όλο τον κόσμο συνηγορούν στο ότι τα ηλεκτρονικά παιχνίδια θα μπορούσαν να διευκολύνουν αλλά και να εμπλουτίσουν την εκπαιδευτική διαδικασία προσφέροντας γνώση με διασκεδαστικό τρόπο. Βασιζόμενες στην ίδια λογική η οποία έχει ως στόχο την διαφοροποίηση από τη συμβατική εκπαίδευση έρευνες σε ηλεκτρονικά παιχνίδια κατέδειξαν ότι χαρακτηριστικά όπως η φαντασία, ο έλεγχος, η πρόκληση, η περιέργεια και ο ανταγωνισμός προκαλούν εσωτερική υποκίνηση (Malone, 1981; Cordova and Lepper, 1996). Επισημαίνεται ότι, η αποτελεσματικότητα της εσωτερικής υποκίνησης βασίζεται στο γεγονός ότι ο εκπαιδευόμενος συμμετέχει σε μαθησιακές δραστηριότητες χωρίς να απαιτεί οποιασδήποτε μορφής ανταμοιβή. Ο Prensky (2002) αναφέρει ότι μάθηση και διασκέδαση δεν είναι δύο ασυμβίβαστες έννοιες, επίσης οι Lepper & Cordova (1992) συμφωνούν ότι η μάθηση που είναι διασκεδαστική μπορεί να είναι και αποτελεσματική. Επιπλέον, ο Boyle (1997) επισημαίνει ότι τα παιχνίδια είναι δυνατόν να προσδώσουν ένα ελκυστικό και ευχάριστο χαρακτήρα στην μάθηση, προσφέροντας μια ισχυρή «φόρμα» για τον σχεδιασμό αποτελεσματικών εκπαιδευτικών περιβαλλόντων (Virvou, Katsionis & Manos, 2005).

Με κύριο γνώμονα την επίτευξη θετικών μαθησιακών αποτελεσμάτων, σε συνδυασμό με ενδιαφέρουσες εκπαιδευτικές μεθόδους, οι ηλεκτρονικές εφαρμογές (βασιζόμενες στη λογική του «παιχνιδιού») είναι το επόμενο βήμα τόσο για μαθητές όσο και για εκπαιδευτικούς.

1.4 Η συμβολή του παιχνιδιού στην εκπαίδευση

Για όλους τους εμπλεκόμενους η χρήση του παιχνιδιού στην εκπαιδευτική διαδικασία είναι ένα αμφιλεγόμενο ζήτημα, κατά πόσον δηλαδή θα μπορούσε να είναι ένα αποτελεσματικό μέσο μάθησης. Από τη μια είναι σίγουρα μια διαφορετική εκδοχή από αυτή της κλασικής παράδοσης

του μαθήματος, στον αντίποδα όμως μπορεί να προκαλέσει αποδιοργάνωση τόσο στο μαθητή όσο και στον εκπαιδευτικό. Ένα παιχνίδι για να αποτελέσει επιτυχημένο εργαλείο μάθησης θα πρέπει να συμβάλλει στην απόκτηση γνώσεων, να έχει συγκεκριμένους εκπαιδευτικούς στόχους και να είναι διασκεδαστικό. Επίσης η ενεργή συμμετοχή των μαθητών σε αυτό είναι καθοριστική καθώς έχει περισσότερες πιθανότητες να επιφέρει ένα θετικό αποτέλεσμα.

Επιπλέον, ο ρόλος του δασκάλου είναι πολύ σημαντικός αφού θα χρειαστεί να συντονίσει όσο το δυνατόν καλύτερα την διεξαγωγή του ώστε να γίνει αποδεκτό από τους μαθητές δηλαδή να μην το απορρίψουν ή να το υποτιμήσουν και παράλληλα να κινητοποιηθούν ώστε να επιφέρει το αναμενόμενο αποτέλεσμα. Είναι κάτι το οποίο έγκειται στην οπτική του εκάστοτε εκπαιδευτικού το πώς θα το προσαρμόσει, αν και εφόσον το επιθυμεί φυσικά στο μάθημα του ώστε να είναι εποικοδομητικό και ευχάριστο.

Όσο αφορά τα ψηφιακά παιχνίδια με βάση το περιεχόμενο τους αλλά και τις δεξιότητες που επιδιώκουν να ενισχύσουν στους μαθητούμενους κατηγοριοποιούνται διαφορετικά. Ευρέως γνωστά εκπαιδευτικά παιχνίδια είναι «Το θαυμαστό ταξίδι των Ζουμπίνις» (“Logical Journey of the Zoombinis”) και πιο πρόσφατο το “Itacante”. Επίσης σημαντικό είναι να αναφέρουμε και το ψηφιακό εκπαιδευτικό παιχνίδι που δημιουργήθηκε από το Μουσείο Μακεδονικού Αγώνα σε συνεργασία με το Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης με τίτλο «Ο Μακεδονικός Αγώνας στη λίμνη των Γιαννισών».

Πλέον πρόσφατη υπό - κατηγορία είναι αυτή των «σοβαρών παιχνιδιών» (“serious games” – «παιχνίδια για σοβαρό σκοπό»). Οι στόχοι τους είναι πιο σύνθετοι καθώς είναι απαραίτητο να διατηρηθεί ο έλεγχος της διασκέδασης που οδηγεί στην συμμετοχή και τα εκπαιδευτικά στοιχεία που οδηγούν στην εκπαιδευτική εμπειρία και τη μάθηση (Gunter, Kenny and Vick, 2006). Ο Zyda (2005) το περιγράφει ως ένα ψυχικό διαγωνισμό, ο οποίος με τρεις βασικούς βοηθούς τη διασκέδαση, τον υπολογιστή και συγκεκριμένους κανόνες στοχεύει να αναπτύξει τομείς όπως της εταιρικής κατάρτισης, της εκπαίδευσης, της υγείας, της δημόσιας τάξης και τους στρατηγικούς στόχους επικοινωνίας. Επίσης, μέσα στο ευρύτερο πεδίο χρήσης τους τα ψηφιακά παιχνίδια με σοβαρούς σκοπούς, όπως είναι η διδασκαλία ή η κατάρτιση, έχουν πρωτεύων στόχο την εκπαίδευση. Τα σοβαρά παιχνίδια επιδιώκουν την ένωση των ψηφιακών παιχνιδιών με τον ακαδημαϊκό κόσμο και τον κόσμο της προσομοίωσης, προκειμένου να «μεταφέρουν» τον παίκτη σε ένα ασφαλές και διασκεδαστικό περιβάλλον μάθησης (Rankin & Vargas, 2008).

Σύμφωνα με τους Mansour και El-Said (2008), τα σοβαρά παιχνίδια μπορούν να προσφέρουν πολλά πλεονεκτήματα καθώς και να μετατραπούν σε ιδιαίτερα αποτελεσματικά μέσα διδασκαλίας. Μερικά από αυτά τα πλεονεκτήματα είναι η ανάπτυξη διαφόρων δεξιοτήτων, όπως η στρατηγική σκέψη, ο σχεδιασμός, η επικοινωνία, η συνεργασία, οι ομαδικές αποφάσεις και οι δεξιότητες διαπραγμάτευσης, η ενίσχυση της αποκτηθείσας γνώσης και της διάρκειάς της, η προσαρμογή της μαθησιακής εμπειρίας, σύμφωνα με το προφίλ του κάθε μαθητή, το στυλ και το βαθμό μάθησης, η διευκόλυνση της μάθησης, ώστε να διαδραματιστεί σε ένα πλαίσιο που να έχει νόημα για το παιδί και η στήριξη της δημιουργίας ομάδων (<http://www.digitalgames.edu.gr>).



Εικόνα 1 Serious Game Ιατρική Προσομοίωση

Ανάπτυξη εφαρμογής γνώσεων για Android – Kn-owl-edge

1.5 Παραδείγματα ηλεκτρονικών εκπαιδευτικών εφαρμογών

1.5.1 Ένα παιχνίδι με άρωμα αρχαίας Ελλάδας

Πρόκειται για το ηλεκτρονικό παιχνίδι Minecraft το οποίο βασίζεται στην ψηφιακή εκδοχή του πασίγνωστου και κλασικού πλέον παιχνιδιού Lego. Στόχος του, η κατασκευή κόσμων και η εξερεύνηση. Δημιουργός του εν λόγω παιχνιδιού είναι ο Σουηδός προγραμματιστής Markus Persson (2009) και από το 2013 έχει πουλήσει πάνω από 15 εκατ. αντίτυπα στον υπολογιστή και 54 εκατ. σε όλες τις πλατφόρμες. Πριν από ένα χρόνο αγοράστηκε από την αμερικανική εταιρεία λογισμικού Microsoft για 2,5 δισεκατομμύρια δολάρια. Το 2011, άνοιξε ο Rainbow Minecraft Server, από την ελληνική κοινότητα Minecraft ο οποίος αναγνωρίστηκε ως ένας από τους καλύτερους server παγκοσμίως. Το πρώτο Minecraft βασισμένο σε ιστορικά γεγονότα με αναφορά στην Αρχαία Ελλάδα, είναι σε διαδικασία υλοποίησης από την ελληνική ομάδα RFS (Rainbow Family Servers) με βασικό του στόχο να αποτελέσει εκπαιδευτικό υλικό.



Εικόνα 2 Minecraft Αρχαία Ελλάδα

Μέσα από αυτό, ο παίκτης θα έχει τη δυνατότητα να περιηγηθεί σε ελληνικές πόλεις (Σπάρτη, Αθήνα, Πέλλα, Ολυμπία) και να γνωρίσει ψηφιακά τόπους με ιδιαίτερη αρχαιολογική σημασία. Οι δημιουργοί του αναφέρουν ότι θα υπάρχουν αναφορές και από την Ελληνική Μυθολογία, όπως μυστικά περάσματα στον κόσμο του Άδη ή ο Προκρούστης στην Ελευσίνα. Επιπλέον στόχος τους είναι η αναπαράσταση ιστορικών μαχών (300 – Θερμοπύλεις / ναυμαχία της Σαλαμίνας), στις οποίες ο παίκτης θα συμμετέχει ενεργά. Το ελληνικό project δεν είναι ακόμη διαθέσιμο στο κοινό. Άλλες πηγές μεταφέρουν πως αναμένεται να παρουσιαστεί η 3D εκδοχή του στο υπουργείο πολιτισμού.

Αντίστοιχη της ελληνικής ιδέας το 2011, μια διεθνής ομάδα εκπαιδευτικών και προγραμματιστών, η TeacherGaming, δημιούργησε το πρόγραμμα MinecraftEDU με σκοπό να ερευνήσει κατά πόσο τα ψηφιακά παιχνίδια θα μπορούσαν να λειτουργήσουν ως εργαλείο διδασκαλίας στις σχολικές αίθουσες. Ο ιρλανδός εκπαιδευτικός Τζόελ Μιλς πρότεινε το Minecraft στην εκπαίδευση, εκπροσωπώντας το Πανεπιστήμιο Χαλ της Αγγλίας με ένα σύντομο πρόγραμμα σπουδών που θα φιλοξενήσει το 2015 η εκπαιδευτική ψηφιακή πλατφόρμα ακαδημαϊκών μαθημάτων του δικτύου Canvas. Ο κ. Μιλς έχει ασχοληθεί επισταμένα με τον τρόπο που το μάθημα της Ιστορίας μπορεί να ενταχθεί στο Minecraft και να λειτουργήσει ως ένα πρωτότυπο εργαλείο για τους εκπαιδευτικούς. Για παράδειγμα, η εκδοχή «Ο άνθρωπος των ανθρωπιστικών σπουδών» είναι ένα παιχνίδι ρόλων μέσω του οποίου μπορεί κανείς να περιηγηθεί στα αρχαία ερείπια της Ρώμης, τον πύργο του Άιφελ ακόμα και να εξερευνήσει τις πυραμίδες. Σύμφωνα με τον Μιλς, πρόκειται για έναν ιδιαίτερο τρόπο να συστήσει κανείς στους μαθητές του διαφορετικούς πολιτισμούς.

Η συνεισφορά του Minecraft δεν περιορίζεται μόνο στο πεδίο των ανθρωπιστικών μαθημάτων. Το παιχνίδι δίνει το ερέθισμα στους μαθητές να ασχοληθούν με διαφορετικές

επιστήμες, όπως η Φυσική, ο Προγραμματισμός και η Αρχιτεκτονική. Η δυνατότητα που παρέχει στους μαθητές όσο αφορά την κατασκευή κτιρίων τα οποία είναι προσχεδιασμένα από αρχιτέκτονες και να τους δώσουν μια απτή μορφή, με τη χρήση εκτυπωτών 3D, είναι εντυπωσιακή.

Το παιχνίδι αποτελεί σημαντικό εργαλείο στα χέρια των εκπαιδευτικών, αφού σύμφωνα με τον κ Μιλς οι εκπαιδευόμενοι χωρίς να το αντιληφθούν έρχονται σε επαφή με αντικείμενα τα οποία άλλες συνθήκες δεν θα τους ενδιέφεραν.

Σύμφωνα με έρευνα του S.R.I. International, μιας ομάδας ειδικών σε θέματα τεχνολογίας της Silicon Valley, τα παιχνίδια δύνανται να αναπτύξουν τη γνωστική μάθηση κατά 12%, αυξάνοντας την ικανότητα επίλυσης προβλημάτων καθώς και τις λειτουργίες της μνήμης.

Ο Ερικ Κλόπφερ, καθηγητής στο Ινστιτούτο Τεχνολογίας της Μασαχουσέτης, εξηγεί πως ειδικά το Minecraft ενισχύει τις κατασκευαστικές δεξιότητες των μαθητών και ενεργοποιεί την αντίληψη του χώρου, ενώ σύμφωνα με τη διδάκτορα του Τμήματος Επικοινωνίας, Μέσων και Πολιτισμού του Παντείου Πανεπιστημίου κ. Λήδα Τσενέ, αυξάνει «το ομαδικό πνεύμα, την κριτική σκέψη και την δημιουργικότητα». Η ίδια συμπληρώνει, ότι η καλλιέργεια των συγκεκριμένων δεξιοτήτων είναι ο λόγος που «στις ΗΠΑ τα παιχνίδια έχουν προσαρμοστεί στο ισχύον εκπαιδευτικό πρόγραμμα». Σύμφωνα με έρευνα του Παντείου, παρόλο που οι έλληνες εκπαιδευτικοί κρατούν μια επιφυλακτική στάση για το ρόλο του παιχνιδιού, αυτό αρχίζει σιγά σιγά να εντάσσεται στην εκπαιδευτική διαδικασία, με αποτελεσματικό τρόπο. Σαφώς όλα τα παραπάνω θα αρχίσουν να αποκτούν περισσότερη βάση μέσω των διαφόρων δεξιοτήτων που θα καλλιεργούνται στους εμπλεκόμενους όπως ηγετικό πνεύμα, ομαδικότητα, ευρηματικότητα και στρατηγική σκέψη.

Εν κατακλείδι, ο κ Μιλς εξηγεί πως δεν φοβάται καθόλου τα παιδιά που παίζουν παιχνίδια στον υπολογιστή, τονίζει όμως πως οι γονείς τους θα πρέπει να είναι σίγουροι πως θα είναι προστατευμένα μέσα σε αυτό το περιβάλλον.

*Συμπληρωματικά να αναφέρουμε πως το παιχνίδι Minecraft είναι πλέον εφαρμογή για κινητές συσκευές android εκτός από την έκδοσή του για υπολογιστές.

1.5.2 Quizdom: Ένα παιχνίδι γνώσεων

Ένα ιδιαίτερα δημοφιλές παιχνίδι γνώσεων είναι το Quizdom το οποίο εμφανίστηκε ως mobile εφαρμογή στις 31 Ιουλίου του 2014. Δημιουργοί και εμπνευστές της, είναι οι Τριαντάφυλλος Ξυλούρης και Παντελής Μαυρομάτης. Μετά από σχεδόν ένα χρόνο κυκλοφορίας το Quizdom έχει ξεπεράσει κάθε προσδοκία για τα ελληνικά δεδομένα. Συγκεκριμένα αριθμεί πάνω από 1.000.000 χρήστες, περισσότερες από 20.000 ερωτήσεις διαφορετικής θεματολογίας ενώ παράλληλα, η εμπειρία που προσφέρει στο χρήστη είναι τόσο ψυχαγωγική όσο και εκπαιδευτική.



Εικόνα 3 Quizdom

Η εν λόγω εφαρμογή βασίζεται σε ένα quiz γνώσεων το οποίο χωρίζεται σε 9 κατηγορίες:

Ανάπτυξη εφαρμογής γνώσεων για Android – Kn-owl-edge

- Επιστήμη
- Σώμα και Υγεία
- Θρησκεία
- Business
- 20^{ος} αιώνας
- Πολιτική
- Παιχνίδια
- Φαγητό
- Ζώα

Από αυτές ο κάθε χρήστης καλείται να επιλέξει 2 κατηγορίες από αυτές. Καινοτόμο στοιχείο είναι το γεγονός ότι κάθε παίκτης έρχεται αντιμέτωπος είτε με τυχαίους αντιπάλους (άλλους χρήστες της εφαρμογής) είτε μπορεί να επιλέξει μέσα από τη λίστα φίλων του στο κοινωνικό δίκτυο Facebook. Επιπρόσθετα, η επιλογή του αντιπάλου μπορεί να γίνει και βάση ονόματος χρήστη (username). Η δυνατότητα ανταγωνισμού που προσφέρει η εφαρμογή, προσδίδει ένα κοινωνικό χαρακτήρα στο παιχνίδι και συμπληρώνεται με την ύπαρξη chat.

Η «μάχη» γίνεται σε 4 κατηγορίες (2 για κάθε παίκτη) και σε ασύγχρονο χρόνο. Αρχικά, ο πρώτος παίκτης έχει 30 δευτερόλεπτα για να απαντήσει σε κάθε ερώτηση από τις 4 συνολικές της κάθε κατηγορίας. Μόλις ολοκληρώσει με τις απαντήσεις του ξεκινάει ο γύρος του αντιπάλου του ο οποίος σε περίπτωση που δεν είναι ενεργός εκείνη τη στιγμή πρέπει να απαντήσει σε διάστημα 12 ωρών, ώστε να μην ακυρωθεί το παιχνίδι.

Ένα ακόμα ανταγωνιστικό στοιχείο του παιχνιδιού είναι η ύπαρξη των επιπέδων και στατιστικών στοιχείων. Με κάθε νίκη του εκάστοτε παίκτη συγκεντρώνεται ένας αριθμός πόντων. Όταν ο χρήστης συμπληρώσει ένα προκαθορισμένο σύνολο πόντων αυξάνει κατά ένα το επίπεδό του. Συνολικά τα επίπεδα του παιχνιδιού είναι 80 και καθένα από αυτά συνοδεύεται με την αναβάθμιση της εικόνας προφίλ του συμμετέχοντα. Το χαρακτηριστικό αυτό κινητοποιεί το χρήστη μέσω του ανταγωνισμού να προχωρήσει στα επόμενα επίπεδα, να φτάσει ψηλότερα σε πόντους και να ανακαλύψει πως θα εμφανίζεται η εικόνα του. Η εξέλιξη της εικόνας παρουσιάζεται από μία απλή αντρική η γυναικεία μορφή (ανάλογα με το φύλο του συμμετέχοντα) στα πρώτα επίπεδα μέχρι μια πιο εμπλουτισμένη αντίστοιχη μορφή στα επόμενα.

Συμπεραίνουμε λοιπόν, ότι ο συνδυασμός ανταγωνιστικού παιχνιδιού και επιβράβευσης στοχεύει σε μια διαδικασία μάθησης με διασκεδαστικό τρόπο και απλουστευμένες τεχνικές. Ενισχύοντας είτε θετικά το χρήστη σε μια σωστή απάντηση είτε αρνητικά σε μια λανθασμένη, με τη χρήση ήχων, δημιουργείται ένα στοιχείο εθισμού και έτσι ο χρήστης συνεχίζει να παίζει. Το ενδιαφέρον που δημιουργεί στο κοινό του το εν λόγω παιχνίδι του προσδίδει επιπλέον επιμορφωτικό χαρακτήρα, μέσω των γνώσεων που προσφέρει.

1.5.3 Ελληνικές εκπαιδευτικές εφαρμογές

Αρχικά, στο χώρο της ελληνικής εκπαίδευσης οι διαθέσιμες σχετικές εφαρμογές είναι περιορισμένες. Παρόλα αυτά, θα συναντήσουμε για μαθητές της Γ' Λυκείου μια σειρά εφαρμογών διαφορετικής μορφής από την εταιρία Υβριδική Εκπαίδευση. Η σειρά αυτή αποτελείται από διάφορα quiz με θέματα θεωρίας σωστό – λάθος περασμένων Πανελληνίων εξετάσεων. Άλλες εφαρμογές προσφέρουν υπολογισμό των μορίων, παλαιότερα θέματα εξετάσεων ή την εξεταστέα ύλη των μαθημάτων.

Συγκριτικά με το εξωτερικό διαπιστώνουμε ότι η ποικιλία εφαρμογών με τη μορφή quiz καθιστά αναγκαία την εμφάνιση μιας πρωτότυπης και καινοτόμου ιδέας που θα αποτελέσει ένα ουσιαστικό εργαλείο γνώσης για τους Έλληνες μαθητές και εκπαιδευτικούς.

Κεφάλαιο 2 - Ανάλυση της Εφαρμογής Kn-owl-edge

Λαμβάνοντας υπόψη τα στοιχεία που παρουσιάστηκαν στο προηγούμενο κεφάλαιο, καθώς και το πρόσφορο έδαφος στο χώρο της εκπαίδευσης για νέους τρόπους επιμόρφωσης, αναδύεται η ευκαιρία για τη δημιουργία μιας πρωτότυπης εφαρμογής, για την κάλυψη αυτού του κενού. Ο στόχος της παρούσας μεταπτυχιακής διατριβής είναι η δημιουργία και διαμόρφωση μιας Ανάπτυξη εφαρμογής γνώσεων για Android – Kn-owl-edge

εκπαιδευτικής εφαρμογής η οποία θα είναι διαθέσιμη σε κινητά και tablets με λειτουργικό android. Μια εφαρμογή η οποία θα στηρίζεται σε δυο βασικούς άξονες, τον άξονα της εκπαίδευσης, αλλά και της ψυχαγωγίας μέσω του παιχνιδιού.

Στο κεφάλαιο που ακολουθεί, θα γίνει περαιτέρω ανάλυση και παρουσίαση των βασικών αυτών αξόνων, αλλά και περιγραφή της εφαρμογής, της ροής και του περιεχομένου της.

2.1 Kn-owl-edge

Η εν λόγω εφαρμογή, αποτελεί μια υβριδική μίξη του δημοφιλούς τύπου παιχνιδιού γνώσεων με φροντιστηριακού επιπέδου βοηθό πανελληνίων. Ονομάζεται Kn-owl-edge και αυτή τη στιγμή παρέχει τη δυνατότητα επιλογής του μαθήματος της Βιολογίας Γενικής Παιδείας. Απευθύνεται σε μαθητές της Γ' Λυκείου οι οποίοι προετοιμάζονται για τη συμμετοχή τους στις Πανελλαδικές εξετάσεις. Στον πυρήνα της η εφαρμογή έχει τη μορφή ενός παιχνιδιού γνώσεων όπως το ευρέως γνωστό trivial pursuit όπου ανέκαθεν συνδύαζε την ψυχαγωγία με τη μάθηση, με την ειδοποιό διαφορά της στοχευμένης θεματολογίας ως προς τα μαθήματα των πανελληνίων. Παράλληλα η απόδοσή της γίνεται μέσω της πλατφόρμας των κινητών τηλεφώνων επικεντρώνοντας στη σχέση της τεχνολογίας και του μαθητή.

Αρχικά το όνομα της εφαρμογής έχει διπλή σημασία, καθώς η ερμηνεία της αγγλικής λέξης knowledge είναι η γνώση ενώ αντίστοιχα, υποκρύπτει το λογοπαίγνιο της κουκουβάγιας (owl) γνωστή από την Αρχαία Ελλάδα ως σύμβολο της Σοφίας, της Γνώσης και της Μάθησης.



Εικόνα 4 Λογότυπο Εφαρμογής Kn-owl-edge

Κάθε μάθημα αποτελείται από τα κεφάλαια και τις ενότητες τα οποία καλύπτουν την υπάρχουσα ύλη των Πανελληνίων. Στόχος της εφαρμογής είναι η μάθηση αλλά και η επανάληψη των ήδη «αποκτημένων» γνώσεων. Συνεπώς, φιλοδοξεί να αποτελέσει ένα ιδιαίτερα βοηθητικό μέσο μάθησης προετοιμάζοντάς με ψυχαγωγικό τρόπο τους χρήστες στο δύσκολο δρόμο τους προς τις εξετάσεις.

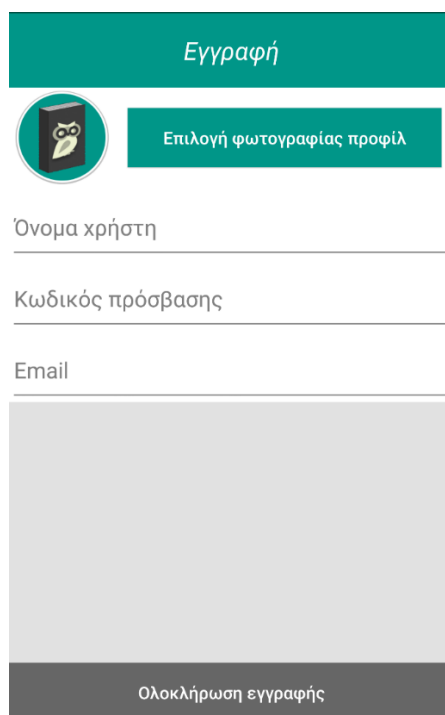
Για τις ανάγκες της παρούσας μεταπτυχιακής διατριβής χρησιμοποιήθηκε το μάθημα της Βιολογίας Γενικής Παιδείας λόγω της συνεργασίας με τον αντίστοιχο καθηγητή. Επιπλέον είναι σημαντικό να διευκρινιστεί ότι κάθε μάθημα έχει τις δικές του ιδιαιτερότητες ως προς την ένταξή του σε μια τέτοια μορφή παιχνιδιού ώστε να αποδοθεί αποτελεσματικά και να είναι βοηθητικό για το μαθητή. Επομένως δημιουργήθηκε η πλατφόρμα με στόχο την μελλοντική ένταξη όλων των μαθημάτων.

Αρχικά, ο χρήστης-μαθητής δημιουργεί έναν λογαριασμό ώστε να του προσφερθεί μια εξατομικευμένη εμπειρία. Για τη δημιουργία λογαριασμού χρησιμοποιείται η κλασική φόρμα συμπλήρωσης του email, του ονόματος χρήστη και του κωδικού πρόσβασης. Εναλλακτικά, δίνεται και η δυνατότητα εγγραφής μέσω κοινωνικού δικτύου και συγκεκριμένα του Facebook το οποίο είναι ιδιαίτερα δημοφιλές στα παιδιά αυτών των ηλικιών.

Αυτή η οικεία εμπειρία της εγγραφής του χρήστη συνοδεύεται από την επιλογή φωτογραφίας για το προφίλ του μαθητή είτε μέσα από τις ήδη αποθηκευμένες στο κινητό του,

είτε από την λήψη μιας νέας κατά την εγγραφή του, πράγμα σύνηθες σε διάφορες δημοφιλείς εφαρμογές ή μέσα κοινωνικής δικτύωσης (selfies¹). Μέσα από αυτή τη διαδικασία ο χρήστης ήδη εισέρχεται σε ένα γνώριμο περιβάλλον χωρίς να προκαταβάλλεται ότι πρόκειται για μια εφαρμογή αυστηρά εκπαιδευτικού περιεχομένου.

Στην παρακάτω εικόνα, παρουσιάζεται η μέθοδος εγγραφής χωρίς κοινωνικό δίκτυο, με την επιλογή της φωτογραφίας προφίλ.

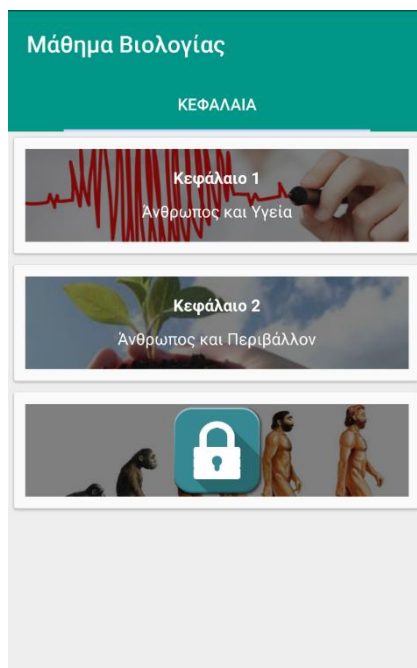


Εικόνα 5 Διαδικασία Εγγραφής μέσω email

Αφού ο χρήστης ολοκληρώσει την εγγραφή του, μπορεί να επιλέξει το μάθημα που τον ενδιαφέρει το οποίο όπως προαναφέραμε, για τα πλαίσια της διατριβής, είναι η Βιολογία Γενικής Παιδείας. Στη συνέχεια, του παρουσιάζονται τα κεφάλαια του μαθήματος με μια ιδιαιτερότητα, όλα εκτός από το πρώτο, είναι κλειδωμένα. Όπως σε πολλά παιχνίδια, ο χρήστης έρχεται αντιμέτωπος με κλειδωμένο περιεχόμενο, το οποίο θα γίνει προσβάσιμο μόνο εφόσον ολοκληρώσει επιτυχώς τις αντίστοιχες προκλήσεις. Το γεγονός αυτό προτρέπει το μαθητή να αντιληφθεί την εκπαιδευτική διαδικασία ως ένα σύνολο στόχων / αποστολών που πρέπει να φέρει εις πέρας επιτυχώς, όπως ακριβώς θα γινόταν σε ένα παιχνίδι. Είναι σύνηθες στα ηλεκτρονικά παιχνίδια, ο χρήστης να «ξεκλειδώνει» το περιεχόμενο έπειτα από την ολοκλήρωση μιας αποστολής / πρόκλησης.

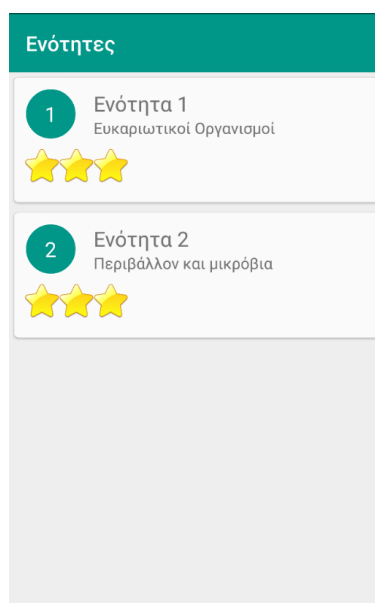
Στην εικόνα που ακολουθεί παρατηρούμε τα δύο πρώτα κεφάλαια του μαθήματος της Βιολογίας να είναι προσβάσιμα, ενώ το τρίτο παραμένει κλειδωμένο.

¹ Selfies: Όρος που προέκυψε από το μέσο του διαδικτύου. Η λέξη προέρχεται από τα Self Portraits το οποίο σημαίνει πορτραίτα του εαυτού μας. Με τον όρο Selfies σήμερα εννοούμε την ενέργεια φωτογράφισης του εαυτού μας συνήθως μέσω κινητών τηλεφώνων. Ο όρος Selfie έγινε και επίσημη λέξη από το Oxford English Dictionary.



Εικόνα 6 Επιλογή Κεφαλαίων

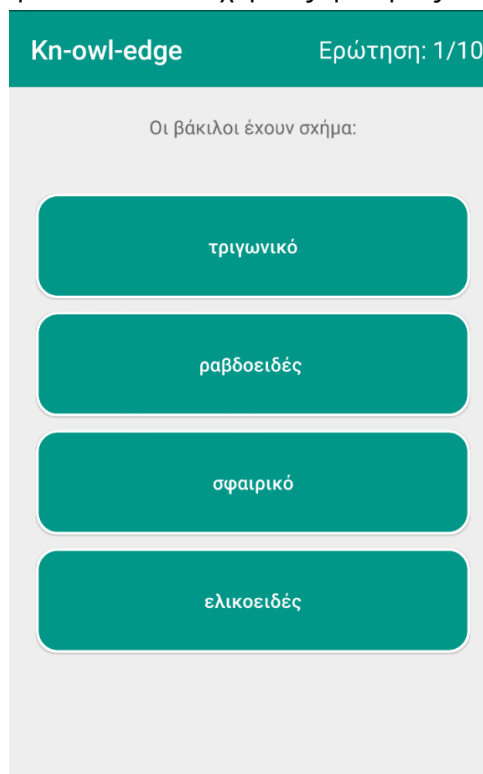
Αυτό το μοτίβο της γραμμικής εξερεύνησης και εκπλήρωσης στόχων παρατηρείται και στις αντίστοιχες ενότητες κάθε κεφαλαίου. Συγκεκριμένα ο χρήστης πρέπει να απαντήσει σωστά σε 10 ερωτήσεις ανά ενότητα. Σύμφωνα με το ποσοστό των σωστά απαντημένων ερωτήσεων, επιβραβεύεται και με τον αντίστοιχο αριθμό χρυσών αστεριών. Εάν ο χρήστης απαντήσει σωστά σε πέντε ή έξι ερωτήσεις θα επιβραβευτεί με ένα χρυσό αστέρι, επτά έως εννιά με δύο χρυσά αστέρια, ενώ τέλος αν απαντήσει και στις δέκα σωστά, επιβραβεύεται με τον μέγιστο αριθμό χρυσών αστεριών που είναι τα τρία. Με την απόκτηση ενός τουλάχιστον χρυσού αστεριού του δίνεται η δυνατότητα να επισκεφτεί την αμέσως επόμενη ενότητα. Για να «ξεκλειδώσει» όμως το επόμενο κεφάλαιο πρέπει να επαναλάβει τις ενότητες έως ότου πάρει σε όλες τον μέγιστο αριθμό αστεριών.



Εικόνα 7 Ενότητες και Αστέρια

Με αυτόν τον τρόπο ο μαθητής καταφέρνει να έχει όλη την εξεταστέα ύλη του μαθήματος δομημένη ανά ενότητα και ανά κεφάλαιο στο κινητό του, έχοντας πρόσβαση οποιαδήποτε στιγμή. Με τη γνωστή μέθοδο από τα παιχνίδια, της επανάληψης του περιεχομένου, ο χρήστης «αναγκάζεται» να επαναλάβει τις ερωτήσεις προκειμένου να ολοκληρώσει επιτυχώς τα κεφάλαια, διαδικασία η οποία συμβάλλει στην μάθηση της ύλης. Επίσης, η διαδικασία επιβράβευσης με αστέρια ενισχύει θετικά το μαθητή, δίνοντας του κίνητρο για την υποβολή περισσότερης προσπάθειας, με στόχο την απόκτηση του μέγιστου βαθμού αστεριών.

Ο τύπος των ερωτήσεων είναι της μορφής πολλαπλής επιλογής. Στον χρήστη εμφανίζεται μια ερώτηση με τέσσερις πιθανές απαντήσεις. Διαθέτει όσο χρόνο θέλει για να επιλέξει μία από αυτές. Ο μη χρονικός περιορισμός αποσκοπεί στο να δοθεί η δυνατότητα στο μαθητή να ανατρέξει στο αντίστοιχο κεφάλαιο του σχολικού του βιβλίου, με σκοπό την αναζήτηση της λύσης. Σκοπός της εφαρμογής είναι η ενίσχυση της διδασκαλίας του μαθήματος και όχι η αντικατάσταση του σχολικού βιβλίου από αυτή. Μόλις ο χρήστης κάνει την επιλογή του, το σύστημα τον ενημερώνει με πράσινο χρώμα εάν απάντησε σωστά ενώ με κόκκινο εάν απάντησε λάθος. Για να κριθεί έγκυρη η βαθμολογία και να γίνει δυνατή η απόκτηση χρυσών αστεριών, πρέπει να απαντήσει σε δέκα συνεχόμενες ερωτήσεις.



Εικόνα 8 Παράδειγμα Ερώτησης

2.2 Άξονας Εκπαίδευσης

Η εφαρμογή έχει ως βασική προϋπόθεση την επιμόρφωση του μαθητή. Για αυτό το λόγο οι ερωτήσεις και το περιεχόμενο της, έχουν συγκροτηθεί έχοντας ως θεματολογία τις πανελλήνιες εξετάσεις. Πολλά παιχνίδια τύπου Quiz ή εφαρμογές έχουν εκπαιδευτικό χαρακτήρα και αυτό διότι προσφέρουν στο χρήστη γνώσεις που μπορεί να μην κατείχε. Η παρούσα εφαρμογή όμως θέτει την εκπαίδευση του χρήστη, τη μάθηση, ως τον κεντρικό πυλώνα της, διατηρώντας όμως στην πορεία τα υπόλοιπα στοιχεία που εμφανίζονται σε αντίστοιχες εφαρμογές. Το εφαλτήριο λοιπόν για την εφαρμογή είναι ο άξονας της εκπαίδευσης.

Για το μάθημα της Βιολογίας, χρησιμοποιείται αναθεωρημένη ύλη για το σχολικό έτος 2015 – 2016, από τον καθηγητή Αριστείδη Λεοντή. Συγκεκριμένα, πραγματοποιήθηκε κατανομή της

ύλης κάθε κεφαλαίου, σε δέκα ενότητες. Κάθε ενότητα, αντίστοιχα, περιλαμβάνει δέκα ερωτήσεις. Σκοπός της κατανομής ήταν:

- Ο χωρισμός σε μικρά θεματικά τμήματα της ύλης, με στόχο ο μαθητής να παίρνει όση περισσότερη πληροφορία γίνεται για τις αντίστοιχες σελίδες.
- Κάθε ενότητα, να μπορεί να χρησιμοποιηθεί αυτόνομα, αλλά ταυτόχρονα να ενσωματώνεται και στο ευρύτερο θέμα του κεφαλαίου στο οποίο ανήκει.
- Κάθε ερώτηση να είναι διαφορετική δίχως να υπάρχει επανάληψη, εκτός εάν ενισχύει την εκπαιδευτική λειτουργία. Για παράδειγμα, στο πρώτο κεφάλαιο, οι μαθητές, μαθαίνουν για τους μύκητες και τους ευκαρυωτικούς οργανισμούς, μέσω πολλών διαφορετικών ερωτήσεων με μικρές διαφορές που όμως οδηγούν το μαθητή στην ουσιαστική αντίληψη του κεφαλαίου.
- Κάθε ερώτηση να είναι της μορφής ερώτησης πολλαπλής επιλογής, με μία σωστή απάντηση και τρεις λανθασμένες. Ανάμεσα στις λανθασμένες, υπάρχουν απαντήσεις που οι διαφορές ως προς τη σωστή αιτιολόγηση είναι ελάχιστες, με στόχο να παρουσιάσουν στο μαθητή παγίδες που μπορεί να εμφανιστούν σε εξετάσεις επιπέδου πανελληνίων.

Τα τρία κεφάλαια στα οποία χωρίστηκε η ύλη είναι:

1. Άνθρωπος και Υγεία
2. Άνθρωπος και Περιβάλλον
3. Εξέλιξη

Μερικά παραδείγματα ερωτήσεων:

- ❖ Φαγοκύτταρα είναι: α) τα ουδετερόφιλα, β) τα αιμοπετάλια, γ) τα Β-λεμφοκύτταρα, δ) τα πλασματοκύτταρα.
- ❖ Τα πρωτόζωα είναι: α) πολυκύτταροι ευκαρυωτικοί οργανισμοί, β) μονοκύτταροι ευκαρυωτικοί οργανισμοί, γ) μονοκύτταροι προκαρυωτικοί οργανισμοί, δ) ακυτταρικές μορφές ζωής.
- ❖ Τα βακτήρια διαθέτουν: α) έλυτρο, β) πυρήνα, γ) κυτταρικό τοίχωμα, δ) όλα όσα αναφέρονται.
- ❖ Θεμελιώδη γενίκευση της Βιολογίας αποτελεί: α) η θεωρία του Λαμάρκ, β) η θεωρία της εξέλιξης, γ) η κυτταρική θεωρία, δ) όλα όσα αναφέρονται.
- ❖ Πληθυσμός είναι το σύνολο των ατόμων: α) που ζουν σε μια περιοχή, β) που μπορούν να αναπαραχθούν μεταξύ τους, γ) που μπορούν να αναπαραχθούν μεταξύ τους επειδή ζουν στην ίδια περιοχή, δ) του ίδιου είδους που ζουν σε διαφορετικές περιοχές.

2.3 Άξονας Ψυχαγωγίας

Ο δεύτερος άξονας πάνω στον οποίο εδραιώθηκε η ανάπτυξη και ο σχεδιασμός της εφαρμογής, είναι ο άξονας της ψυχαγωγίας. Ένας από τους κυριότερους στόχους ήταν η παρουσίαση στο χρήστη, περιεχομένου, μέσω του οποίου θα επιτυγχάνονταν όχι μόνο η επιμόρφωση του, αλλά και η ψυχαγωγία του. Για να γίνει αυτό εφικτό, χρησιμοποιήθηκε η τεχνική του “gamefication”.

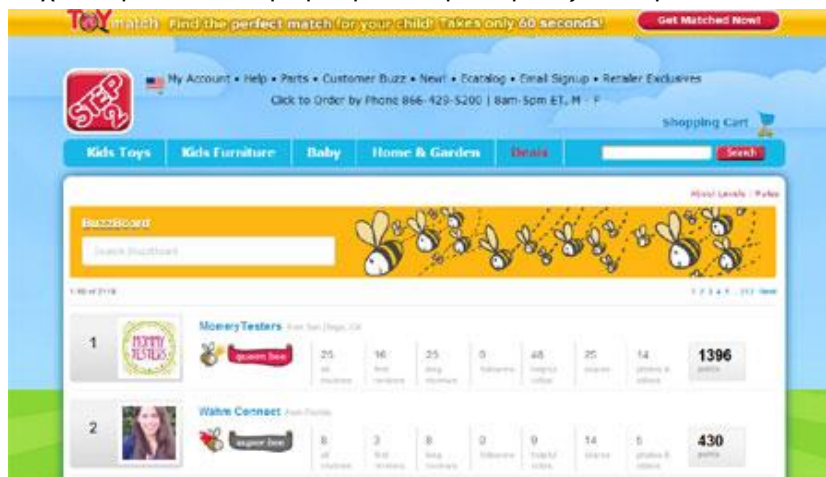
Με τον όρο gamefication εννοείται η ενίσχυση μιας εφαρμογής με στοιχεία από το μέσο των παιχνιδιών και την αφομοίωση χαρακτηριστικών που υπάρχουν σε αυτά, ώστε να αποδοθεί ένας πιο ψυχαγωγικός ή και εθιστικός χαρακτήρας σε αυτή. Για να αναλυθεί αυτός ο όρος περαιτέρω και να γίνει απόλυτα κατανοητός, είναι καλύτερη η χρήση παραδειγμάτων.

Πολλές φορές, κατά τη διάρκεια περιήγησης σε κάποιον ιστότοπο ή κατά τη χρήση μιας εφαρμογής, μπορεί κάποιος να παρατηρήσει διάφορα χαρακτηριστικά ή στοιχεία, τα οποία είναι «δανεισμένα» από το μέσο των βιντεοπαιχνιδιών. Ένα από αυτά είναι η ύπαρξη βραβείων, τα οποία κερδίζει ο χρήστης εκπληρώνοντας συγκεκριμένες ενέργειες. Τα βραβεία αυτά, ονομάζονται achievements (επιτεύγματα), ή badges (διακριτικά / σήματα). Ένα παράδειγμα αυτής της τεχνικής είναι ο ιστότοπος της εταιρείας Treehouse².

² Τον ιστότοπο της Treehouse, μπορεί κάποιος να τον επισκεφτεί στη διεύθυνση <https://teamtreehouse.com/>. Η εταιρεία Treehouse είναι μια διαδικτυακή σχολή μάθησης Ανάπτυξη εφαρμογής γνώσεων για Android – Kn-owl-edge

Η εταιρεία Treehouse έχει οργανώσει τα μαθήματα τις σε διάφορες κατευθύνσεις, υπό την ονομασία tracks. Κάθε track, χωρίζεται σε αντίστοιχες υποενότητες, όπου η ολοκλήρωση κάθε μίας βασίζεται στη συλλογή συγκεκριμένου συνόλου πόντων και την κατάκτηση του αντίστοιχου achievement. Η διαδικασία αυτή προσφέρει μια αίσθηση ανταγωνισμού στο μαθητή και καλλιεργεί το αίσθημα της προόδου. Αυτό έχει ως άμεση συνέπεια, ο χρήστης να εθίζεται στο διάβασμα και να θέλει να μάθει περισσότερα για το track το οποίο έχει επιλέξει.

Ένα άλλο παράδειγμα της τεχνικής του gamification είναι ο ιστότοπος της εταιρείας Step2³. Η εταιρεία Step2 έχει δημιουργήσει ένα σύστημα συλλογής πόντων και απόκτησης διαφόρων επιπέδων, όπως αυτών που αναφέρθηκαν στην εφαρμογή Quizdom. Οι γονείς, αναλυτικότερα, μπορούν να γράψουν σχόλια και περιλήψεις για τα προϊόντα που αγόρασαν από την εταιρεία και να τα μοιραστούν για να συλλέξουν τους αντίστοιχους πόντους. Η τεχνική αυτή, στην περίπτωση της εταιρείας Step2, φιλοδοξεί στο να κάνει διαφήμιση του ονόματός της, μέσα από τον σχολιασμό και το διαμοιρασμό απόψεων μεταξύ των γονιών.



Εικόνα 9 Τεχνική Gamefication

Στην εφαρμογή Kn-owl-edge γίνεται χρήση των τεχνικών του όρου gamification με σκοπό, την ψυχολογική ενίσχυση του μαθητή, κάνοντας να αντιληφθεί τη διαδικασία της μάθησης, ως μια ψυχαγωγική διαδικασία. Η τεχνική που χρησιμοποιείται είναι της επιβράβευσης μέσω των αστεριών, σύμφωνα με τις σωστές απαντήσεις ανά ενότητα.

Η διαδικασία αυτή πρωτοεμφανίστηκε στο παιχνίδι Angry Birds της εταιρείας Rovio. Στο παιχνίδι αυτό ο χρήστης μετά από κάθε επιτυχής ολοκλήρωση μιας αποστολής, βαθμολογούνταν από ένα έως τρία χρυσά αστερία. Στο συγκεκριμένο παιχνίδι, ο μηχανισμός αυτός αφορούσε καθαρά στο να εθίσει το χρήστη να επαναλαμβάνει τις πίστες με σκοπό να επιτύχει καλύτερη βαθμολογία. Στην εφαρμογή Kn-owl-edge επιτυγχάνεται και η ψυχολογική τόνωση του μαθητή όταν παίρνει καλή βαθμολογία, αλλά και η επανάληψη μαθαίνοντας έτσι καλύτερα την ύλη.

Κεφάλαιο 3 - Τεχνολογίες που χρησιμοποιήθηκαν

Στο κεφάλαιο που ακολουθεί θα γίνει ανασκόπηση των τεχνολογιών που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής Kn-owl-edge, από το λειτουργικό που επιλέχθηκε μέχρι τις υπηρεσίες του back end και τα cloud services. Επίσης, θα αναλυθούν οι λόγοι για τους οποίους έγινε η κάθε επιλογή, αλλά και τα πλεονεκτήματα τα οποία επιφέρουν.

προγραμματισμού, ανάπτυξης εφαρμογών και δεξιοτήτων που μπορούν να ενισχύσουν τις επαγγελματικές δεξιότητες των χρηστών τους.

³ Τον ιστότοπο της Step2 μπορεί κάποιος να τον επισκεφτεί στη διεύθυνση <http://www.step2.com/>. Η εταιρεία Step2 ειδικεύεται σε παιδικά προϊόντα.

Ανάπτυξη εφαρμογής γνώσεων για Android – Kn-owl-edge

3.1 Λειτουργικό Σύστημα Android



Εικόνα 10 Λογότυπο Λειτουργικού Android

Για τη δημιουργία της εφαρμογής Kn-owl-edge αρχικά, έπρεπε να γίνει η επιλογή πλατφόρμας στην οποία θα πραγματοποιηθεί η ανάπτυξη της. Κριτήρια για αυτή την επιλογή ήταν:

1. Να στοχεύει στα «έξυπνα» κινητά τηλέφωνα.
2. Να είναι ευρέως διαδεδομένη.
3. Η πρόσβαση σε αυτήν από χρήστες να γίνεται δωρεάν.
4. Η ανάπτυξη εφαρμογών για αυτή την πλατφόρμα να είναι δωρεάν.

Βάση των παραπάνω επιλέχθηκε το λειτουργικό σύστημα Android που δημιουργήθηκε από την εταιρεία Google και παρουσιάστηκε για πρώτη φορά τον Νοέμβριο του 2007 και διατίθεται δωρεάν. Η πρώτη επίσημη έκδοση του λειτουργικού ήταν η έκδοση 1.5 με τίτλο Cupcake στις 30 Απριλίου 2009. Ακολούθησαν οι εκδόσεις:

- 1.6 Donut 15 Σεπτεμβρίου 2009
- 2.0, 2.1 Éclair 26 Οκτωβρίου 2009
- 2.2 Froyo 20 Μαΐου 2010
- 2.3 Gingerbread 6 Δεκεμβρίου 2010
- 3.x Honeycomb 22 Φεβρουαρίου 2011
- 4.0.x Ice Cream Sandwich 19 Οκτωβρίου 2011
- 4.x Jelly Bean 10 Ιουλίου 2012
- 4.4 KitKat 31 Οκτωβρίου 2013
- 5.0 Lollipop 3 Νοεμβρίου 2014
- 6.0 Marshmallow αναμένεται

Σήμερα το Android αποτελεί το πιο δημοφιλές λογισμικό στον κόσμο, ενώ παράλληλα οι συσκευές που ενσωματώνουν το λειτουργικό αυτό μετρούν περισσότερες πωλήσεις από όλες τις ανταγωνιστικές συσκευές μαζί (Windows, iOS).

Το λειτουργικό Android συνοδεύει ένα μεγάλο φάσμα κινητών τηλεφώνων δίνοντας τη δυνατότητα στους χρήστες του, να έχουν ένα «έξυπνο» κινητό, έναν δηλαδή μικρό υπολογιστή στο χέρι τους. Οι δυνατότητες που τους δίνει το Android είναι αμέτρητες. Από την αναμενόμενη τηλεφωνική επικοινωνία, τη δυνατότητα για λήψη φωτογραφιών, την αναπαραγωγή πολυμεσικού περιεχομένου, την περιήγηση στο διαδίκτυο, αλλά και την χρήση οργάνων όπως το Παγκόσμιο Σύστημα Στιγματοθέτησης ή Θεσιθεσίας, γνωστό σε όλους ως GPS (Global Positioning System).

Το κυριότερο στοιχείο όμως του λειτουργικού, είναι το Google Play Store, το εικονικό δηλαδή κατάστημα της Google, στο οποίο μπορεί να περιηγηθεί ο χρήστης και να διαλέξει ανάμεσα σε μια πληθώρα εφαρμογών για το λειτουργικό, σύμφωνα με τα ενδιαφέροντα, αλλά και τις ανάγκες του. Σε αυτό το ηλεκτρονικό κατάστημα μπορεί κανείς να ανακαλύψει όλες τις εφαρμογές οι οποίες έχουν υλοποιηθεί για το λειτουργικό. Μέσω του Play Store θα είναι προσβάσιμη και η εφαρμογή Kn-owl-edge.

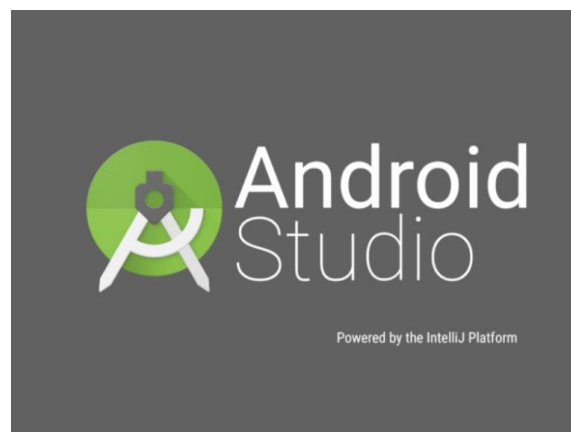
Το Android έχει ως βάση του το λειτουργικό Linux, για τις άμεσες ανάγκες του, όπως η διαχείριση της μνήμης και η διαχείριση των διεργασιών του και αποτελεί τον πυρήνα του. Σε δεύτερο επίπεδο, συναντάμε τις βιβλιοθήκες του λειτουργικού, οι οποίες είναι γραμμένες σε C / C++. Στο επόμενο επίπεδο υπάρχει το εικονικό μηχανήμα Dalvik το οποίο είναι υπεύθυνο για τη μετάφραση του κώδικα σε γλώσσα που αντιλαμβάνεται το λειτουργικό. Τα δύο τελευταία επίπεδα που συνθέτουν το λειτουργικό Android είναι το Application Framework Layer στο οποίο «τρέχει» ο κώδικας που δίνεται από την Google και τέλος το Application Layer στο οποίο υπάρχουν οι εφαρμογές που συνοδεύουν τη συσκευή.

Η γλώσσα με την οποία γίνεται η ανάπτυξη για τις εφαρμογές Android είναι η Java, ενώ οι πλατφόρμες μέσω των οποίων αυτό επιτυγχάνεται, είναι ποικίλες. Συγκεκριμένα, μπορεί κάποιος να χρησιμοποιήσει το Android Studio, το Eclipse, ή ακόμα και το Xamarin, όπως και πολλές άλλες.



Εικόνα 11 Αρχιτεκτονική Λειτουργικού Android

3.2 Πλατφόρμα Ανάπτυξης Android Studio



Εικόνα 12 Πλατφόρμα Ανάπτυξης Android Studio

Η πλατφόρμα Android Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment / IDE), η οποία έχει δημιουργηθεί με σκοπό να βοηθάει τον προγραμματιστή στην ανάπτυξη λογισμικού. Ένα IDE συνήθως περιλαμβάνει επεξεργαστή πηγαίου κώδικα, κάποιον compiler (μεταγλωττιστή) και γραφικό περιβάλλον στο οποίο φαίνεται η εξέλιξη του προγράμματος που αναπτύσσεται. Από τα πιο γνωστά ολοκληρωμένα περιβάλλοντα ανάπτυξης στο χώρο της πληροφορικής είναι το Visual Studio της Microsoft.

Το Android Studio είναι βασισμένο στο IntelliJ IDEA της εταιρείας JetBrains και έχει σχεδιαστεί ειδικά για την ανάπτυξη εφαρμογών Android από την ίδια την Google και αντικατέστησε το μέχρι πρότινος περιβάλλον Eclipse Android Development Tools, το οποίο κατείχε τον ρόλο αυτό. Η Google ανακοίνωσε το Android Studio τον Μάιο του 2013 και ήταν έτοιμο για χρήση από το κοινό, το Δεκέμβρη του 2014.

Ένα από τα σημαντικότερα στοιχεία που διαθέτει το Android Studio είναι ο WYSIWYG (what you see is what you get) Editor που διαθέτει, δηλαδή ένα σύστημα στο οποίο εμφανίζεται το περιεχόμενο που δημιουργεί ο προγραμματιστής για την εφαρμογή του. Ονομάζεται WYSIWYG γιατί προσφέρει ακριβώς αυτή τη δυνατότητα, δηλαδή, ό,τι εμφανίζεται στην οθόνη του προγραμματιστή, θα εμφανιστεί και στο τελικό προϊόν.

Μερικά ακόμα λειτουργικά στοιχεία που προσφέρει η πλατφόρμα είναι:

- Το σύστημα gradle (ένα σύστημα της Google για να προσθέτει εύκολα ο προγραμματιστής εξωτερικές βιβλιοθήκες στην εφαρμογή του. Το μόνο που έχει να κάνει για παράδειγμα για να προσθέσει την βιβλιοθήκη Picasso, είναι να ανοίξει το αρχείο gradle του project του και να προσθέσει τη γραμμή «compile com.squareup.picasso:picasso:2.5.2»).
- Έτοιμα πρότυπα κώδικα για διάφορες λειτουργίες εφαρμογής που επαναλαμβάνονται.
- Εργαλεία που μετράνε την απόδοση της εφαρμογής, τη χρηστικότητα, αλλά και την συμβατότητα ανά έκδοση του λειτουργικού.
- Έξυπνες προτάσεις για την ταχύτερη ολοκλήρωση του κώδικα που έχει αρχίσει να γράφει ο προγραμματιστής έχοντας ως αποτέλεσμα την αύξηση του ρυθμού παραγωγής.
- Σύστημα μετάφρασης της εφαρμογής σε διάφορες γλώσσες με πολύ απλό και γρήγορο τρόπο.
- Βοηθάει μέσω γραφικού περιβάλλοντος στη δημιουργία εφαρμογών που στοχεύουν σε πολλές και διαφορετικές αναλύσεις και διαστάσεις οθονών.
- Προσομοιωτή κινητών για να γίνουν δοκιμές στα χιλιάδες διαφορετικά μεγέθη συσκευών.

Τα χαρακτηριστικά που πρέπει να διαθέτει ο υπολογιστής στον οποίο σκοπεύει να εγκαταστήσει ο προγραμματιστής την πλατφόρμα Android Studio είναι:

Για λειτουργικό Windows:

1. Windows 8/7/Vista (32bit/64bit)
2. 2 GB Ram ελάχιστο όριο, προτεινόμενο μέγεθος 4 GB
3. 400 MB ελεύθερο χώρο στον σκληρό δίσκο
4. 1 GB περίπου ελεύθερο χώρο για το Android SDK και για τον προσομοιωτή
5. 1280 x 800 ελάχιστη ανάλυση οθόνης
6. Εγκατεστημένο το Java Development Kit (JDK) 7

Για λειτουργικό Mac OS X:

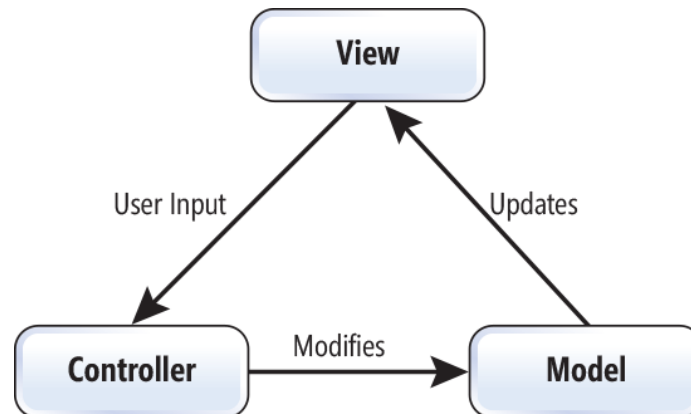
1. Mac OS X: 10.8.5 ή μεγαλύτερο μέχρι 10.9 (Mavericks)
2. 2 GB Ram ελάχιστο όριο, προτεινόμενο μέγεθος 4 GB
3. 400 MB ελεύθερο χώρο στον σκληρό δίσκο
4. 1 GB περίπου ελεύθερο χώρο για το Android SDK και για τον προσομοιωτή
5. 1280 x 800 ελάχιστη ανάλυση οθόνης
6. Εγκατεστημένο το Java Development Kit (JDK) 7

Ομοίως και για λειτουργικό Linux με τη διαφορά ότι χρειάζεται την έκδοση GNOME ή KDE.

3.3 Αρχιτεκτονικό Πρότυπο Προγραμματισμού Android

3.3.1 Πρότυπο Αρχιτεκτονικής MVC

Ο τρόπος με τον οποίο γίνεται ο προγραμματισμός στο λειτουργικό Android είναι μια διασκευή του πρότυπου αρχιτεκτονικής MVC (Model View Controller). Το πρότυπο MVC χωρίζει τον τρόπο με τον οποίο προγραμματίζει κάποιος σε τρία νοητά επίπεδα.



Εικόνα 13 Πρότυπο Αρχιτεκτονικής MVC

Το πρώτο επίπεδο είναι το ονομαζόμενο Model, το οποίο αφορά τις κλάσεις ουσιαστικά, που απαρτίζουν τα δεδομένα της εφαρμογής και είναι υπεύθυνο για τη διαχείριση τους, τη λογική αλλά και τους κανόνες που διέπουν την εφαρμογή.

Το δεύτερο επίπεδο είναι το επίπεδο που ορίζει τον κώδικα των Views, δηλαδή το τι παρουσιάζεται στο χρήστη. Είναι η γραφική αναπαράσταση, η οποία είναι υπεύθυνη να μεταδίδει στο χρήστη την πληροφορία που αποθηκεύεται στο πρώτο επίπεδο.

Τέλος, το τρίτο επίπεδο είναι αυτό του Controller, σε ελεύθερη μετάφραση ελεγκτή, ο οποίος είναι υπεύθυνος για τη διαχείριση και την επικοινωνία των δύο προαναφερθέντων επιπέδων. Ο Controller μπορεί να επηρεάζει την κατάσταση του Model, καθώς και να στέλνει εντολές στα αντίστοιχα Views, όταν αυτά πρέπει να ενημερωθούν.

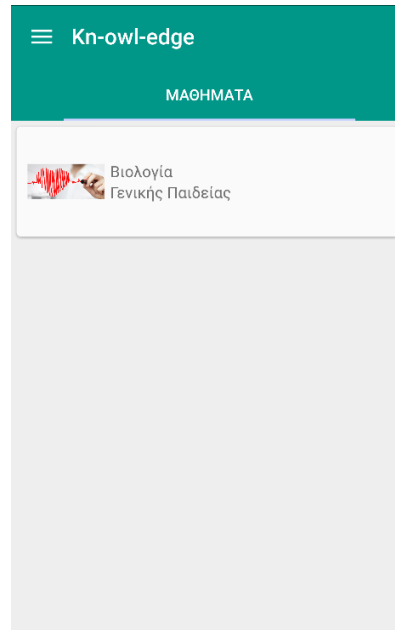
Το πρότυπο αρχιτεκτονικής MVC δημιουργήθηκε για να ξεχωρίσει την αναπαράσταση της πληροφορίας από τη διάδραση του χρήστη (2009, Reenskaug, Corlien). Έχει ως στόχο τον διαχωρισμό του κώδικα σε λογικά κομμάτια, όπου καθένα από αυτά είναι υπεύθυνο για τη δική του δουλειά. Σύμφωνα με το πρότυπο MVC, οποιαδήποτε στιγμή θελήσει ο προγραμματιστής να αλλάξει εντελώς ένα επίπεδο, μπορεί να το πραγματοποιήσει, χωρίς την ανάγκη να αλλάξει και τα υπόλοιπα επίπεδα. Αυτός ο διαχωρισμός των λειτουργιών δίνει τη δυνατότητα στον προγραμματιστή να έχει μεγαλύτερη ελευθερία κινήσεων.

3.3.2 Αρχιτεκτονική Προγραμματισμού Android

Όπως αναφέρθηκε προηγουμένως το Android διαθέτει μια παραλλαγή του πρότυπου αρχιτεκτονικής MVC. Συγκεκριμένα, στο ρόλο των Views υπάρχουν τα Layouts, ενώ τον ρόλο του Controller αναλαμβάνουν τα Activities. Το επίπεδο του Model είναι συνήθως ίδιο σε όλες τις εφαρμογές, εφόσον αναπαριστά όπως αναφέρθηκε, τα δεδομένα της εκάστοτε εφαρμογής.

Τα Activities είναι το σημαντικότερο αντικείμενο και ο βασικός πυλώνας πάνω στον οποίο σχεδιάζεται και αναπτύσσεται μια εφαρμογή Android. Ένα Activity είναι υπεύθυνο για μια οθόνη που θα παρουσιαστεί στο χρήστη με σκοπό να προσφέρει μια συγκεκριμένη λειτουργία, με την οποία μπορούν να αλληλοεπιδράσουν οι χρήστες. Γίνεται άμεσα κατανοητό ότι μια εφαρμογή αποτελείται από πολλά διαφορετικά Activities, τα οποία είναι σχετικά συνδεδεμένα μεταξύ τους, δημιουργώντας μια ροή κατεύθυνσης. Συνήθως κάποιο αρχικό Activity αποτελεί και το κύριο σημείο εισόδου, καθώς και την κύρια οθόνη της εφαρμογής.

Για να γίνει εμφανές το τι είναι ένα Activity, ας δούμε στην παρακάτω εικόνα, το κύριο Activity της εφαρμογής Kn-owl-edge, το οποίο είναι η οθόνη στην οποία επιλέγουμε μάθημα.



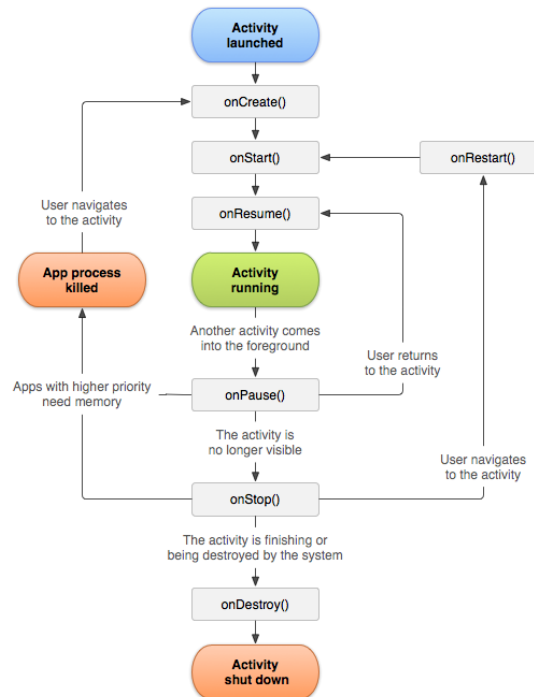
Εικόνα 14 Main Activity Kn-owl-edge

Στην παραπάνω εικόνα παρατηρούμε το κύριο Activity του Kn-owl-edge. Από αυτό το Activity ξεκινάει ουσιαστικά η ροή της εφαρμογής και καθορίζονται οι επιλογές του χρήστη. Όπως αναφέρθηκε ένα Activity είναι υπεύθυνο για μια λειτουργία. Στη συγκεκριμένη περίπτωση, η λειτουργία είναι η παρουσίαση των μαθημάτων στο χρήστη και η αναμονή επιλογής ενός από αυτά. Μόλις επιλεγθεί ένα μάθημα, δημιουργείται μέσω κώδικα ένα δεύτερο Activity στο οποίο και μετακινούμαστε. Το πρώτο Activity μπαίνει σε μία στοιβιά, η οποία αποτελεί το ιστορικό και στην οποία μπορούμε να περιηγηθούμε πατώντας το πλήκτρο πίσω.

Ένα Activity έχει τον δικό του, αντίστοιχο, κύκλο ζωής, όπως μια εφαρμογή. Κάθε στάδιο σε αυτόν τον κύκλο, εκπροσωπείται και από τις αντίστοιχες μεθόδους στον κώδικα.

Τα βασικά στάδια και μέθοδοι είναι:

- **onCreate:** Το αρχικό στάδιο. Σε αυτό το στάδιο δημιουργείται το Activity.
- **onStart:** Ένα ενδιάμεσο στάδιο. Χρονικά συμβαίνει όταν το Activity ετοιμάζεται να γίνει ορατό στο χρήστη.
- **onResume:** Το κύριο στάδιο ενός Activity. Όταν γίνεται πλήρως ορατό στο χρήστη και καταλαμβάνει την οθόνη της συσκευής μας, βρισκόμαστε σε αυτό το στάδιο.
- **onPause:** Ένα ακόμα ενδιάμεσο στάδιο. Αυτό το στάδιο είναι παρόμοιο με το δεύτερο στάδιο, με τη διαφορά ό,τι πλέον το Activity ετοιμάζεται να γίνει αόρατο, διότι ένα άλλο ετοιμάζεται να πάρει τη θέση του.
- **onStop:** Όταν πλέον ένα Activity δεν είναι πλέον ορατό στο χρήστη και ένα άλλο έχει πάρει τη θέση του, είμαστε επισήμως στο στάδιο του onStop.
- **onDestroy:** Τέλος, όταν ένα Activity καταστραφεί ολοσχερώς, φτάνουμε στο τελικό στάδιο του κύκλου ζωής του.

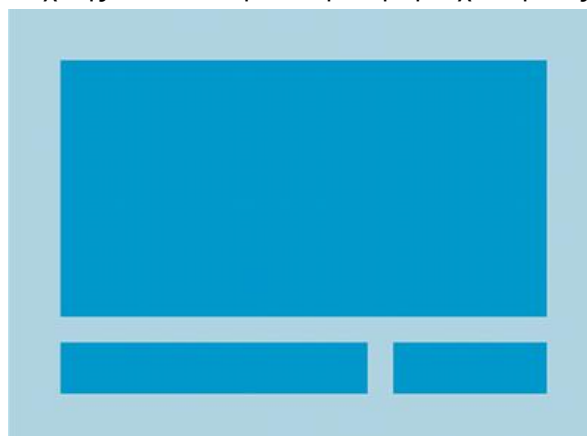


Εικόνα 15 Κύκλος Ζωής Activity

Μελετώντας τον κύκλο ζωής ενός Activity καταλήγουμε στο συμπέρασμα ό,τι στην δομή του κώδικα που γράφει ένας προγραμματιστής, όταν αναπτύσσει μια εφαρμογή για Android, πρέπει να ακολουθεί κάποιους κανόνες. Σύνηθες κανόνας είναι, η αρχικοποίηση των Views και των δεδομένων στη μέθοδο onCreate και η απελευθέρωση της μνήμης στην μέθοδο onDestroy. Τα Views στην περίπτωση του Android, αντικαθίστανται από Layouts τα οποία περιέχονται σε αρχεία xml.

Ένα αρχείο xml περιέχει από ένα ή περισσότερα Layouts. Με τον όρο layout εννοείται μια δομή σύμφωνα με την οποία εμφανίζονται και διατάσσονται τα οπτικά στοιχεία στην οθόνη του κινητού. Τα βασικά Layouts είναι δύο, το Relative (σχετικό) και το Linear (γραμμικό).

Στο Relative Layout η τοποθέτηση των οπτικών στοιχείων μπορεί να γίνει σύμφωνα με τα υπόλοιπα οπτικά στοιχεία που είναι παιδιά του ίδιου Layout ή σύμφωνα με τον γονέα. Δίνεται δηλαδή η δυνατότητα στοίχισης των αντικειμένων βάση την σχετική τους θέση στην οθόνη.



Εικόνα 16 Relative Layout

Στο Linear Layout, όπως αντικατοπτρίζει και το όνομα του, η στοίχιση γίνεται γραμμικά. Υπάρχουν δύο τύποι Linear Layouts, τα οριζόντια και τα κάθετα. Με αυτόν τον τρόπο κάθε παιδί του layout παίρνει γραμμικά τη θέση που του αντιστοιχεί βάσει προτεραιότητας είτε οριζόντια είτε κάθετα.



Εικόνα 17 Horizontal Linear Layout

Ένα παράδειγμα Layout σε xml μορφή είναι το παρακάτω:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <TextView android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello, I am a TextView" />
    <Button android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello, I am a Button" />
</LinearLayout>
```

Στο συγκεκριμένο παράδειγμα παρατηρούμε πως τα οπτικά στοιχεία TextView (στοιχείο για παράθεση κειμένου) και Button (κουμπί) είναι παιδιά ενός κάθετου Linear Layout. Αν επικεντρωθούμε στα χαρακτηριστικά του Layout θα δούμε ότι είναι γονέας των στοιχείων και δεν είναι παιδί κανενός άλλου Layout και ότι το μήκος και ύψος του είναι δηλωμένα ως fill parent. Στην περίπτωση που κάποιο Layout δεν έχει γονέα στον οποίο ανήκει, γονέας γίνεται αυτόματα ολόκληρη η οθόνη. Άρα στη συγκεκριμένη περίπτωση το Layout γεμίζει όλη την οθόνη.

Ορισμένες φορές όμως που θέλουμε να δημιουργήσουμε πιο σύνθετες γραφικές αναπαραστάσεις μπορούμε να συνδυάσουμε τους τύπους των Layout και να δημιουργήσουμε περίπλοκες συσχετίσεις. Πρέπει όμως να έχουμε υπόψιν ότι πολύπλοκα Layouts μπορεί να μην είναι αποδοτικά ως προς τη σχέση τους με τη μνήμη του κινητού, ειδικά αν ανανεώνονται συχνά.

3.4 Back End

Έως τώρα αναλύθηκε ο τρόπος με τον οποίο λειτουργεί η εφαρμογή ως προς το μέρος του πελάτη (client side). Στόχος της εφαρμογής όμως ήταν η ενημέρωση της να είναι δυναμική ώστε να ενημερώνεται αυτόματα, δίχως την ανάγκη για την προσθήκη νέας έκδοσης στο Play Store. Για να επιτευχθεί αυτό έπρεπε να χρησιμοποιηθεί ένα σύστημα back end, το οποίο θα είναι υπεύθυνο για την αποθήκευση δεδομένων, την αποστολή τους στην εφαρμογή και την ανανέωση τους.

Για έναν προγραμματιστή η δημιουργία ενός back end συστήματος είναι μια μεγάλη πρόκληση, καθώς αποτελεί την ραχοκοκαλιά του συστήματος του. Η κύρια προσέγγιση του θέματος είναι η επιλογή ενός έτοιμου συστήματος το οποίο θα παρέχει ευχέρεια κινήσεων και επιλογών στο δημιουργό, αλλά παράλληλα θα διαθέτει και την κατάλληλη αξιοπιστία. Παραδείγματα τέτοιων συστημάτων είναι το Java Spring Framework, Play Framework κ.α.

Για τις ανάγκες της παρούσας εφαρμογής έγινε επιλογή του συστήματος Parse.com το οποίο πρόσφατα αγοράστηκε και από το Facebook και λειτουργεί στο Cloud.



Εικόνα 18 Cloud Service Parse

Μέσω του Parse.com, μπορεί κανείς κάνοντας έναν λογαριασμό, να έχει πρόσβαση σε μία πλατφόρμα ικανή για οτιδήποτε μπορεί να έχει σχεδιάσει να υλοποιηθεί. Ένα από τα κύρια πλεονεκτήματα της πλατφόρμας Parse είναι το κόστος της, το οποίο είναι δωρεάν μέχρι ένα συγκεκριμένο επίπεδο. Για παράδειγμα διατίθενται 20 GB χώρου για βάση δεδομένων, δωρεάν notifications για 1.000.000 μοναδικούς χρήστες κ.α. Για έξτρα ανάγκες μπορεί ο οποιοσδήποτε να δεχτεί τις ανάλογες χρεώσεις και να αυξήσει τους αριθμούς αυτούς.

Η πλατφόρμα Parse φαίνεται παρακάτω:

objectId	Title	createdAt	updatedAt	ACL	Description	Image
L68YxHbK3q	Βιολογία	Sep 07, 2015, 07:56	Oct 05, 2015, 16:44	Public Read and Write	Γενικής Παιδείας	J0ZM06rI10

Εικόνα 19 Cloud Platform Parse

Όπως παρατηρούμε στην εικόνα δίνονται διάφορες δυνατότητες όπως η παρακολούθηση αναλυτικών στοιχείων, η δημιουργία αποθηκευτικού χώρου, η δημιουργία δεδομένων κ.α.

Στην εφαρμογή έχουν δημιουργηθεί και αποθηκευτεί στην πλατφόρμα οι εξής κλάσεις:

- Book
- Chapter
- Section
- Question
- User
- ChapterProgress
- SectionProgress
- Image

Αναλυτικά στην κάθε κλάση αποθηκεύονται τα εξής δεδομένα με την παρακάτω μορφή:

1. Στην κλάση Book αποθηκεύονται τα μαθήματα με τον τίτλο, την περιγραφή και έναν δείκτη στον πίνακα με τις εικόνες.
2. Στην κλάση Chapter αποθηκεύονται τα κεφάλαια του κάθε μαθήματος με τον τίτλο τους, την περιγραφή τους, τον δείκτη στην εικόνα και την σειρά εμφάνισης, καθώς και αν είναι κλειδωμένο ως περιεχόμενο την πρώτη φορά που θα το συναντήσει ο χρήστης.
3. Στην κλάση Section αποθηκεύονται οι ενότητες με τον ίδιο τρόπο και πληροφορία με τα κεφάλαια.
4. Στην κλάση Question αποθηκεύονται οι ερωτήσεις με το αντίστοιχο σετ απαντήσεων αλλά και το ποια από όλες είναι η σωστή.
5. Στις κλάσεις ChapterProgress και SectionProgress αποθηκεύεται αντίστοιχα η εξέλιξη του κάθε χρήστη στα κεφάλαια κ στις ενότητες.
6. Τέλος στις κλάσεις Image και User, αποθηκεύονται οι εικόνες που χρησιμοποιούνται από την εφαρμογή και οι χρήστες.

Ένα πολύ χρήσιμο εργαλείο που παρέχει η Parse είναι και η μορφή δεδομένων Pointer. Με αυτή τη μορφή δίνεται η δυνατότητα δημιουργίας συσχετίσεων των δεδομένων με γρήγορο και εύκολο τρόπο, αλλά και η παρακολούθηση και η εύρεση αυτών των συσχετίσεων άμεσα και απλά. Στις παρακάτω εικόνες παρατηρούμε τη μορφή Pointer όταν αυτή εμπεριέχεται σε κάποια κλάση και έπειτα πως μπορούμε άμεσα να μεταφερθούμε στα δεδομένα στα οποία δείχνει.

objectId	User	createdAt	updatedAt	ACL	Chapter
Ing17tHDYR	YfaeM0dVvKA	Nov 03, 2015, 11:49	Nov 03, 2015, 11:50	Public Read and Write	BP6uSjclxK
JSbQCQD01r	YfaeM0dVvKA	Nov 03, 2015, 11:49	Nov 03, 2015, 11:49	Public Read and Write	ufQuhTJ7FC
Nducw0QH4Y	YfaeM0dVvKA	Nov 03, 2015, 11:49	Nov 03, 2015, 11:49	Public Read and Write	k9DdmJ8W42

Εικόνα 20 Chapter Pointer

objectId	Title	createdAt	updatedAt	ACL	Description	Image
ufQuhTJ7FC	Κεφάλαιο 1	Sep 07, 2015, 09:23	Nov 01, 2015, 14:16	Public Read and Write	Άνθρωπος και Υγε...	J0ZMO6pIi0

Εικόνα 21 Δεδομένα Pointer

Με τη χρήση των Pointers τα οφέλη δεν σταματάνε μόνο στην εύκολη δημιουργία συσχετίσεων. Αντιθέτως, η σημαντικότερη απόρροια της χρήσης τους ήταν η εξοικονόμηση χώρου. Για να γίνει αντιληπτό αυτό, φτάνει να αναφέρουμε ότι πολλές εικόνες που επαναλαμβάνονται, δημιουργούνται μόνο μία φορά στην πλατφόρμα της Parse και έπειτα με τη χρήση των δεικτών επαναχρησιμοποιούνται συνεχώς χωρίς να καταλαμβάνουν έξτρα αποθηκευτικό χώρο.

Ένα άλλο βοήθημα που παρέχει η υπηρεσία Parse είναι το `api` που διαθέτει στους χρήστες τις. Με τον όρο `api` (application programming interface) εννοείται η διεπαφή με τις προγραμματιστικές εντολές που δίνονται στον προγραμματιστή για την εκτέλεση βασικών ενεργειών. Μέσω του `api` μπορεί κάποιος να δημιουργήσει κλάσεις όπως αυτές που προαναφέραμε δυναμικά από τον κώδικα, να αποθηκεύσει αλλά και να αναζητήσει πληροφορία.

Για το λειτουργικό του android η πλατφόρμα Parse, παρέχει την κλάση `ParseObject`. Κάνοντας `extend` το `ParseObject` μπορεί ο προγραμματιστής να δημιουργήσει το μοντέλο του ως ένα «περιτύλιγμα» (`wrapper`). Έπειτα μπορεί να καλέσει τα δεδομένα του από το Cloud αρκεί να έχει σύνδεση στο Internet. Για παράδειγμα για την αποθήκευση ενός αντικειμένου μπορεί να καλέσει τη μέθοδο `saveInBackground`, η οποία θα λειτουργήσει σε κάποιο δευτερεύον νήμα ώστε να μην εμποδίζεται και το `user interface` και να μειώνεται η εμπειρία του χρήστη καθώς χειρίζεται ή πλοηγείται στην εφαρμογή μας.

```
package george.theocharis.knowledge.models;

import com.parse.ParseClassName;
import com.parse.ParseObject;

@ParseClassName("Book")
public class Book extends ParseObject {

    public Book() {
        // empty constructor needed by Parse
    }

    public String getId() {
        return getObjectId();
    }

    public String getTitle() {
        return getString("Title");
    }

    public String getDescription() {
        return getString("Description");
    }

    public String getImage() {
        return getParseObject("Image").getParseFile("Value").getUrl();
    }
}
```

Στο παραπάνω απόσπασμα κώδικα παρατίθεται η κλάση `Book` από την εφαρμογή `Kn-owl-edge`. Όπως αναφέρθηκε η κλάση κάνει `extend` το `ParseObject` και παρατηρούμε στις μεθόδους `get` ότι ουσιαστικά αποτελούν ένα περιτύλιγμα των μεθόδων της `Parse`. Αξίζει να σημειωθεί ότι η πλατφόρμα `Parse` λειτουργεί κυρίως με το σύστημα `key – value`. Δηλαδή κάθε πεδίο έχει μια τιμή κλειδί όπως το όνομα του και μέσω του κλειδιού βρίσκουμε την αντίστοιχη τιμή του. Για παράδειγμα `"Title" : "Βιολογία"`.

Μέσω του `Parse api` δίνεται και η δυνατότητα των `queries` (ερωτημάτων) στη βάση της `Parse`. Για να πραγματοποιήσουμε ένα ερώτημα, αρχικά διαλέγουμε σε ποιο αντικείμενο αναφερόμαστε, δηλαδή ποιον πίνακα από τη βάση μας. Έπειτα μπορούμε να χρησιμοποιήσουμε μία ή περισσότερες από τις βοηθητικές μεθόδους όπως η `whereEqualTo`, `whereGreaterThan` κ.α. Αυτές οι μέθοδοι αποσκοπούν στο να προσομοιώσουν τη συνθήκη `where` σε ένα αντίστοιχο `sql` ερώτημα. Για παράδειγμα για να αναζητήσουμε την εξέλιξη που έχει στα κεφάλαια του μαθήματος, ο χρήστης μπορούμε να γράψουμε:

```
ParseQuery query = ParseQuery.getQuery("ChapterProgress")
    .fromLocalDatastore()
    .fromPin(context.getString(R.string.TABLE_SYNC))
    .whereEqualTo("User", ParseUser.getCurrentUser())
    .whereEqualTo("Chapter", chapter);
```

Τα αποτελέσματα ενός τέτοιου ερωτήματος μας γίνονται διαθέσιμα μέσω των μηχανισμών callbacks του api. Κάθε ερώτημα έχει ένα callback, όπως getCallback, findCallback για τις αντίστοιχες μεθόδους. Στο callback λοιπόν, παρέχεται το αντικείμενο της αναζήτησης ή η λίστα των αντικειμένων αν περισσότερα από ένα αντικείμενα ανταποκρίνονται στο ερώτημα μας, ή σε περίπτωση που κάτι έγινε λάθος ή δεν είχαμε αποτέλεσμα το αντίστοιχο exception. Ο μηχανισμός των callbacks είναι πολύ ισχυρός γιατί μας δίνει τη δυνατότητα να δουλέψουμε σειριακά με αλυσιδωτά ερωτήματα που το καθένα ενεργοποιείτε όταν τελειώνει το προηγούμενο, να χειραγωγήσουμε τα τυχόν λάθη και να τα διορθώσουμε και τέλος, να αποκτήσουμε πρόσβαση στις απαντήσεις των ερωτημάτων μας.

Για να γίνει αυτό εφικτό η πλατφόρμα Parse βασίζεται στις δυνατότητες που προσφέρει η βιβλιοθήκη Bolts. Μέσω αυτής της βιβλιοθήκης προσομοιώνεται μια μορφή εντολών που παρέχει η τεχνολογία .Net, αλλά όχι η Java, αυτή των Tasks. Με τη βοήθεια των Tasks μπορεί ο προγραμματιστής να σχεδιάσει και να υλοποιήσει όποια διαδικασία είναι χρονοβόρα να πραγματοποιείται σε κάποιο background thread. Αυτή η τακτική είναι σύνηθες να ακολουθείται στην πλατφόρμα του Android, ώστε να μένει το κύριο thread πάντα ελεύθερο καθώς είναι υπεύθυνο για το UI που εμφανίζεται στο χρήστη.

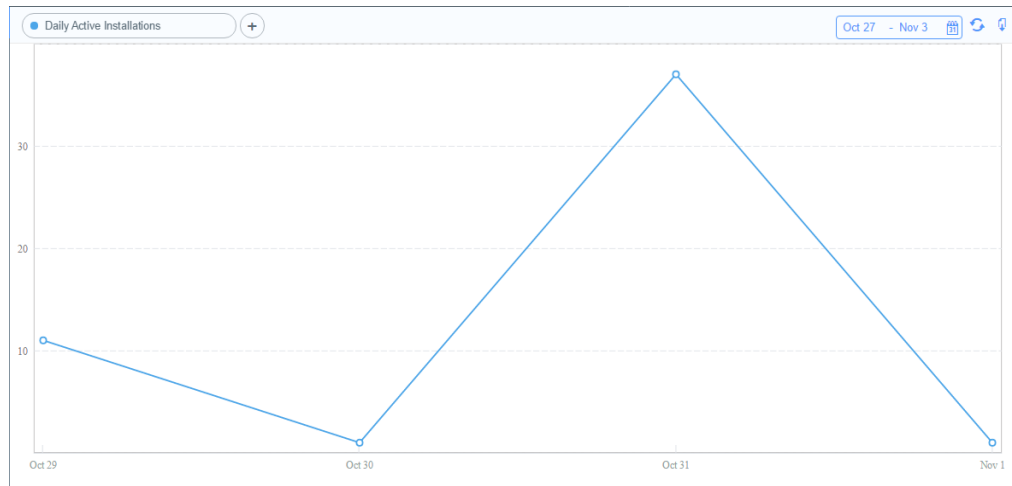
Οι διαφορές τους με τα Async Tasks που παρέχει η Java είναι:

- Κατανάλωση λιγότερων πόρων από το σύστημα.
- Δίνεται η δυνατότητα ολοκλήρωσης πολλών tasks αλυσιδωτά.
- Μπορούμε να οργανώσουμε πότε θα εκτελεστεί το κάθε task, χωρίς να πρέπει να μπλεχτούμε με τη διαχείριση threads(νημάτων).

Επίσης, ένα από τα χαρακτηριστικά των μοντέρνων εφαρμογών είναι η χρήση των notifications. Τα notifications είναι μικρές επισημάνσεις, σαν διαφημίσεις ή υπενθυμίσεις που στέλνονται στο κινητό του χρήστη, εάν έχει την αντίστοιχη εφαρμογή στο κινητό του και έχει δεχτεί τους όρους. Η υπηρεσία της Parse, δίνει τη δυνατότητα στον προγραμματιστή να στείλει δωρεάν μέχρι και σε ένα εκατομμύριο μοναδικούς χρήστες τα δικά του παραμετροποιημένα μηνύματα. Παρέχεται η ευκαιρία για επιλογή του συνόλου στο οποίο θέλει να σταλθεί το μήνυμα, για παράδειγμα σε μια ομάδα για testing ή σε κάποια νηρ μέλη, αλλά και ο χρόνος στον οποίο θέλει να παραδοθεί το μήνυμα.

Τελευταίο, αλλά όχι ασήμαντο χαρακτηριστικό της Parse, είναι το local datastore ή ο τοπικός αποθηκευτικός χώρος που προσφέρει. Με την Parse δεν χρειάζεται κάποιος να κάνει χρήση της βάσης sqlite που είναι η καθιερωμένη για το λειτουργικό σύστημα του android. Μπορεί να κάνει χρήση του μέσου τοπικής αποθήκευσης της Parse, τα λεγόμενα pins. Με τα pins (ουσιαστικά οι αντίστοιχοι πίνακες της sql) μπορεί ο προγραμματιστής να αποθηκεύσει στο κινητό του χρήστη αντικείμενα μόνα τους ή ομαδοποιημένα δίνοντας ένα όνομα στο αντίστοιχο pin. Η μόνη διαφορά στην αναζήτηση αντικειμένων έπειτα, είναι η χρήση της μεθόδου fromLocalDatastore και fromPin στη δημιουργία ενός ερωτήματος.

Τα χαρακτηριστικά που αναφέρθηκαν προηγουμένων ανήκουν όλα στον πυρήνα της υπηρεσίας, δηλαδή στο πακέτο Core που προσφέρει η εταιρεία. Εκτός από το πακέτο Core, παρέχονται και άλλα πακέτα όπως τα Analytics για την παρακολούθηση στατιστικών στοιχείων τα οποία θα δώσουν στον προγραμματιστή την κατάλληλη πληροφορία για το τι αλλαγές πρέπει να κάνει για να αυξήσει το δημογραφικό του κοινό. Στην εικόνα που ακολουθεί μπορούμε για παράδειγμα να δούμε τις ημερήσιες εγκαταστάσεις της εφαρμογής.



Εικόνα 22 Ημερήσιες Εγκαταστάσεις Kn-owl-edge

Έχοντας αναλύσει τα βασικά πλεονεκτήματα της cloud πλατφόρμας Parse, γίνονται εύκολα αντιληπτοί οι λόγοι για την επιλογή της. Αποτελεί ένα σημαντικό σύμμαχο στην ανάπτυξη μιας σωστά δομημένης εφαρμογής και προσθέτει δυνατότητες και επιλογές που θα κόστιζαν τόσο σε χρόνο υλοποίησης, αλλά και σε οικονομικό επίπεδο.

Έως τώρα αναλύθηκαν τα βασικά χαρακτηριστικά των εργαλείων και τεχνολογιών που χρησιμοποιήθηκαν, αλλά και οι λόγοι για την επιλογή του καθενός. Στην ενότητα που ακολουθεί θα αναλυθεί ο προγραμματιστικός σκελετός και η υλοποίηση της εφαρμογής που αναπτύχθηκε στα πλαίσια της μεταπτυχιακής διατριβής.

3.5 Υλοποίηση Kn-owl-edge

3.5.1 Αρχιτεκτονική Εφαρμογής Kn-owl-edge

Πριν ξεκινήσουμε την περιγραφή της ανάλυσης της εφαρμογής παραθέτουμε μερικά στοιχεία σε αριθμούς. Για τη δημιουργία της χρησιμοποιήθηκαν:

- 27 αρχεία Java
- 42 αρχεία xml
- 14 εικόνες
- 7 βιβλιοθήκες
- 3130 γραμμές κώδικα
- 1 πρόγραμμα .NET

Τα Activities της εφαρμογής είναι:

- IntroActivity
- LoginCredentialsActivity
- RegisterActivity
- MainActivity
- ChaptersActivity
- SectionsActivity
- QuizActivity

Ξεκινώντας με το IntroActivity, η εφαρμογή κάνει χρήση των πρώτων της fragments. Με τον όρο fragments εννοείται ένα αντικείμενο του λειτουργικού android, το οποίο ορίζεται σαν ένα μικρότερο Activity και έχει το δικό του κύκλο ζωής, αλλά καθ' όλη τη διάρκεια ζωής του είναι προσκολλημένο σε κάποιο activity γονέα. Η χρήση τους βοηθάει στη δημιουργία δυναμικού περιεχομένου το οποίο μπορεί να επαναχρησιμοποιηθεί ή ακόμα και στη δημιουργία διαφορετικών layout για να καλύψουν τα διάφορα μεγέθη οθονών ανά συσκευή.

Στο IntroActivity παρουσιάζεται στο χρήστη το ερώτημα «Νέος Παίκτης» ή «Έχω ήδη λογαριασμό». Τα δύο πλήκτρα αποτελούν ένα fragment το οποίο αλλάζει σύμφωνα με την επιλογή του χρήστη και αντικαθίσταται από το LoginFragment ή το RegisterFragment. Ασχέτως της επιλογής του χρήστη του δίνεται άμεσα η δυνατότητα να συνδεθεί μέσω του Facebook. Σημειωτέον ότι για την ενσωμάτωση του Facebook στην εφαρμογή έχει χρησιμοποιηθεί το Facebook Sdk σε συνδυασμό με το Parse api για την αποθήκευση του νέου χρήστη.

Τα αμέσως επόμενα activities είναι τα LoginCredentialsActivity ή RegisterActivity. Στο πρώτο, ο χρήστης μεταφέρεται στην οθόνη εισαγωγής των στοιχείων του και επιχειρεί να συνδεθεί στην εφαρμογή. Η εφαρμογή κάνει τα αντίστοιχα ερωτήματα στη βάση για να βεβαιωθεί ότι υπάρχει χρήστης με αυτά τα στοιχεία και αν υπάρχει τον μεταφέρει στο επόμενο activity. Σε περίπτωση λάθους του αναφέρει να ξανά προσπαθήσει. Στο δεύτερο Activity, ο χρήστης μπορεί να πραγματοποιήσει την εγγραφή του ως νέος παίκτης. Η ολοκλήρωση της εγγραφής γίνεται εφικτή αν και μόνο αν, έχουν συμπληρωθεί όλα τα πεδία σωστά και το email του είναι της σωστής μορφής.

Η επιλογή εικόνας είναι προαιρετική και σε περίπτωση μη επιλογής, παραμένει το λογότυπο της εφαρμογής. Εάν επιθυμεί όμως ο χρήστης την αλλαγή της, μπορεί να διαλέξει μία από το άλμπουμ φωτογραφιών του ή να φωτογραφηθεί επιτόπου. Το αποτέλεσμα γυρνάει στο activity, επεξεργάζεται το αρχείο bitmap και αποθηκεύεται στην Parse.

Στο MainActivity της εφαρμογής παρουσιάζονται τα μαθήματα στο χρήστη (βλ. εικόνα 12). Για την παρουσίαση αυτή έχει χρησιμοποιηθεί το εργαλείο RecyclerView. Το συγκεκριμένο view του Android, ουσιαστικά αποτελεί μια λίστα η οποία μπορεί να δει ο χρήστης σύροντας το δάχτυλο του προς τα πάνω ή κάτω.

Προγραμματιστικά το RecyclerView χωρίζεται σε δύο κλάσεις. Η πρώτη κλάση είναι ο adapter ο οποίος είναι υπεύθυνος για τα δεδομένα που υπάρχουν στο view. Στο συγκεκριμένο Activity ο adapter δέχεται μια λίστα από αντικείμενα Book. Ο adapter εμφανίζει τόσες γραμμές στο view όσο το μέγεθος της λίστας, το οποίο είναι εμφανές από τη μέθοδο size. Το σημαντικότερο όμως στοιχείο του adapter είναι η μέθοδος onBindViewHolder. Στην onBindViewHolder ο adapter συνδέει τα δεδομένα του αντίστοιχου Book με την αντίστοιχη θέση στο View. Με αυτόν τον τρόπο η πρώτη γραμμή θα έχει πάντα τα στοιχεία του πρώτου βιβλίου από την αντίστοιχη λίστα. Η δεύτερη κλάση που επιτρέπει στο RecyclerView να λειτουργήσει είναι η ViewHolder. Σε αυτήν την κλάση γίνεται αρχικοποίηση των views που επαναλαμβάνονται σε κάθε γραμμή της λίστας, αλλά και η διαχείριση των τυχών επιλογών του παίκτη.

Με τη χρήση του RecyclerView έχουν δημιουργηθεί και τα ChaptersActivity και SectionsActivity, τα οποία είναι υπεύθυνα για την παρουσίαση των κεφαλαίων και των ενοτήτων αντίστοιχα. Για την παρουσίαση των views ανά γραμμή των λιστών, χρησιμοποιήθηκε το λεγόμενο CardView το οποίο είναι ένα view προσωμοίωσης κάρτας. Χαρακτηριστικό παράδειγμα CardView είναι το view της κάθε ενότητας στο SectionsActivity. Συγκεκριμένα το xml που ορίζει το CardView μιας ενότητας είναι το εξής:

```
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:card_view="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <android.support.v7.widget.CardView
        android:id="@+id/card_view"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_margin="5dp"
        card_view:cardCornerRadius="4dp"
        card_view:contentPadding="10dp">

        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent">
```

```
<LinearLayout
    android:baselineAligned="false"
    android:id="@+id/sectionDetails"
    android:weightSum="1"
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <FrameLayout
        android:layout_weight=".2"
        android:layout_width="0dp"
        android:layout_height="50dp">

        <ImageView
            android:id="@+id/lockedSection"
            android:layout_width="50dp"
            android:layout_height="wrap_content"
            android:adjustViewBounds="false"
            android:cropToPadding="false"
            android:src="@drawable/lockpad"
            android:visibility="gone" />

    <FrameLayout
        android:id="@+id/sectionNumberContainer"
        android:layout_width="50dp"
        android:layout_height="wrap_content"
        android:layout_gravity="center|start"
        android:visibility="visible">

        <ImageView
            android:id="@+id/sectionNumberDrawable"
            android:layout_width="50dp"
            android:layout_height="wrap_content"
            android:src="@drawable/section_circle" />

        <TextView
            android:id="@+id/sectionNumber"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:textColor="@color/white"
            android:textSize="20sp" />
    </FrameLayout>
</FrameLayout>

<LinearLayout
    android:layout_weight=".8"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:orientation="vertical">

    <TextView
        android:id="@+id/sectionTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="start"
        android:text="Biology">
```

```

        android:textSize="20sp" />

        <TextView
            android:id="@+id/sectionDescription"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="start"
            android:text="Biology Description"
            android:textSize="15sp" />
    </LinearLayout>
</LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:weightSum="1"
        android:orientation="horizontal"
        android:layout_below="@+id/sectionDetails">

        <ImageView
            android:id="@+id/first_star"
            android:layout_width="0dp"
            android:layout_height="50dp"
            android:layout_weight=".1"
            android:src="@drawable/star_empty"/>

        <ImageView
            android:id="@+id/second_star"
            android:layout_width="0dp"
            android:layout_height="50dp"
            android:layout_weight=".1"
            android:src="@drawable/star_empty"/>

        <ImageView
            android:id="@+id/third_star"
            android:layout_width="0dp"
            android:layout_height="50dp"
            android:layout_weight=".1"
            android:src="@drawable/star_empty"/>
    </LinearLayout>

</RelativeLayout>

</android.support.v7.widget.CardView>
</LinearLayout>

```

Το παραπάνω CardView αποτελείται από ένα εξωτερικό Linear Layout και κάνει χρήση συνδυασμό διαφόρων τύπων layouts στο εσωτερικό του για να αναπαραστήσει είτε τις κλειδωμένες ενότητες είτε τις ξεκλειδωτες.

Τα κύρια στοιχεία του είναι τα TextView για προβολή κειμένου όπως ο τίτλος μιας ενότητας και τα ImageView για την προβολή εικόνων όπως τα αστέρια. Να σημειωθεί ότι γίνεται χρήση της βιβλιοθήκης android.support.v7 για να μπορεί να λειτουργήσει το CardView σε κινητά με παλαιότερες εκδόσεις από την έκδοση 5 Lollipop.

Στο QuizActivity γίνεται χρήση ξανά ενός fragment. Το fragment αυτό, με την ονομασία StoryQuizFragment, είναι υπεύθυνο για την παρουσίαση των ερωτήσεων, το ανακάτεμα των

απαντήσεων και τη διατήρηση σκορ. Ένα από τα σημαντικότερα αλγοριθμικά στοιχεία της εφαρμογής βρίσκεται σε αυτό το activity.

Μετά το πέρας των δέκα ερωτήσεων ακολουθεί ένας αλγόριθμος όπου κάνει αναζήτηση στην εξέλιξη του χρήστη στην παρούσα ενότητα, κοιτάει τη βαθμολογία του σε αστέρια και τη συγκρίνει με την τωρινή βαθμολογία. Εάν η τωρινή είναι μεγαλύτερη την αποθηκεύει στη βάση του. Εάν ο χρήστης έχει πάρει τουλάχιστον ένα αστέρι, αναζητάει στη βάση την επόμενη ενότητα, αν υπάρχει. Σε περίπτωση ύπαρξης παρατηρεί αν η ενότητα είναι κλειδωμένη και την ξεκλειδώνει. Εφόσον δεν υπάρχει η ενότητα κοιτάει για το επόμενο κεφάλαιο αν και μόνο αν η βαθμολογία όλων των ενοτήτων είναι 3 χρυσά αστέρια. Με το πέρας του αλγόριθμου, το activity τερματίζει και επιστρέφουμε στις ενότητες όπου ανανεώνεται η βαθμολογία.

```
private void quizDone() {

getFragmentManager().beginTransaction().remove(quizFragment).commit();

quizScore.setVisibility(View.VISIBLE);

quizScore.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        ParseQuery sectionQuery =
ParseQuery.getQuery("Section").whereEqualTo("objectId", sectionId);

        try {
            Log.d("Overview", "Step 1: find the section
progress");

            ParseObject parseObject =
ParseQuery.getQuery("SectionProgress")
                .fromLocalDatastore()
                .fromPin("section_progress")
                .whereEqualTo("User",
ParseUser.getCurrentUser())
                .whereMatchesQuery("Section", sectionQuery)
                .getFirst();

            if (parseObject.getInt("Stars") < stars) {

                Log.d("Overview", "Step 2: Stars: " +
parseObject.getInt("Stars") + " are lower than current score " +
stars);

                switch (stars) {
                    case 1:
                        parseObject.put("Stars", 1);
                        break;
                    case 2:
                        parseObject.put("Stars", 2);
                        break;
                    case 3:
                        parseObject.put("Stars", 3);
                        break;
                    default:
                        parseObject.put("Stars", 0);
                        break;
                }
            }
        }
    }
}
```

```

        parseObject.pinInBackground();
        parseObject.saveEventually();

        int nextSection = getIntent().getIntExtra("Order",
0) + 1;

        int lastSection =
ParseQuery.getQuery("Section").fromLocalDatastore().fromPin(getString(
R.string.TABLE_SECTIONS)).addDescendingOrder("Order").getFirst().getIn
t("Order");

        Log.d("OverView", "Step 3: nextSection: " +
nextSection + ", lastSection: " + lastSection);

        if (nextSection <= lastSection) {

            ParseQuery<com.parse.ParseObject> chapterQuery
= new ParseQuery<>("Chapter");
            chapterQuery.whereEqualTo("objectId",
chapterId);

            sectionQuery =
ParseQuery.getQuery("Section").whereMatchesQuery("ParentChapter",
chapterQuery)
                .whereEqualTo("Order", nextSection);

            parseObject =
ParseQuery.getQuery("SectionProgress")
                .fromLocalDatastore()
                .fromPin("section_progress")
                .whereEqualTo("User",
ParseUser.getCurrentUser())
                .whereMatchesQuery("Section",
sectionQuery)
                .getFirst();

            if (parseObject != null) {
                Log.d("OverView", "Step 4: There is a next
Section");

                parseObject.put("Locked", false);
                parseObject.pinInBackground();
                parseObject.saveEventually();
            }

            ParseQuery.getQuery("SectionProgress")
                .fromLocalDatastore()
                .fromPin("section_progress")
                .whereEqualTo("User",
ParseUser.getCurrentUser())
                .findInBackground(new
FindCallback<ParseObject>() {
                    @Override
                    public void done(List<ParseObject>
list, ParseException e) {
                        boolean notAllStarsCompleted =
false;

```

```

        for (ParseObject obj : list) {
            if (obj.getInt("Stars") <
3)
                notAllStarsCompleted =
true;
        }
        Log.d("OverView", "Step 5: All
sections 3starred: " + !notAllStarsCompleted);
        if (!notAllStarsCompleted) {
            ParseObject parseObject =
null;
            try {
                parseObject =
ParseQuery.getQuery("ChapterProgress")
.fromLocalDatastore()
.fromPin(getString(R.string.TABLE_SYNC))
.whereEqualTo("User", ParseUser.getCurrentUser())
.whereEqualTo("Locked", true)
.orderByAscending("Order")
                .getFirst();
            } catch (ParseException
e1) {
                e1.printStackTrace();
            }
            if (parseObject != null) {
                Log.d("Size", "Chapter
Locked Found");
                parseObject.put("Locked", false);
                parseObject.pinInBackground();
                parseObject.saveInBackground();
            } else {
                Log.d("Size", "Chapter
Locked Not Found");
            }
        }
    });
} else {
    ParseQuery.getQuery("SectionProgress")
        .fromLocalDatastore()
        .fromPin("section_progress")
        .whereEqualTo("User",
ParseUser.getCurrentUser())
        .findInBackground(new
FindCallback<ParseObject>() {
            @Override
            public void done(List<ParseObject>
list, ParseException e) {

```

```

false;
boolean notAllStarsCompleted =
3)
true;
next section only chapter");
Log.d("OverView", "Step 4: no
Log.d("OverView", "Step 5: All
sections 3starred: " + !notAllStarsCompleted);
if (!notAllStarsCompleted) {
ParseObject parseObject =
null;
try {
parseObject =
ParseQuery.getQuery("ChapterProgress")
.fromLocalDatastore()
.fromPin(getString(R.string.TABLE_SYNC))
.whereEqualTo("User", ParseUser.getCurrentUser())
.whereEqualTo("Locked", true)
.orderByAscending("Order")
.getFirst();
if (parseObject !=
Log.d("Size",
"Chapter Locked Found");
parseObject.put("Locked", false);
parseObject.pinInBackground();
parseObject.saveInBackground();
} else {
Log.d("Size",
"Chapter Locked Not Found");
}
} catch (ParseException
e1) {
e1.printStackTrace();
}
}
});
}
} catch (ParseException e) {
e.printStackTrace();
}
}

```



```

        finish();
    }
});

finalScore.setText("Απάντησες σωστά σε " + score + " ερωτήσεις!");

switch (score)
{
    case 5:
    case 6:

Picasso.with(this).load(R.drawable.star_fill).into(firstStar);

Picasso.with(this).load(R.drawable.star_empty).into(secondStar);

Picasso.with(this).load(R.drawable.star_empty).into(thirdStar);
        stars = 1;
        break;
    case 7:
    case 8:
    case 9:
        stars = 2;

Picasso.with(this).load(R.drawable.star_fill).into(firstStar);

Picasso.with(this).load(R.drawable.star_fill).into(secondStar);

Picasso.with(this).load(R.drawable.star_empty).into(thirdStar);
        break;
    case 10:
        stars = 3;

Picasso.with(this).load(R.drawable.star_fill).into(firstStar);

Picasso.with(this).load(R.drawable.star_fill).into(secondStar);

Picasso.with(this).load(R.drawable.star_fill).into(thirdStar);
        break;
    default:
        stars = 0;

Picasso.with(this).load(R.drawable.star_empty).into(firstStar);

Picasso.with(this).load(R.drawable.star_empty).into(secondStar);

Picasso.with(this).load(R.drawable.star_empty).into(thirdStar);
        break;
    }
}
}

```

Τα κουμπιά των απαντήσεων στο QuizActivity έχουν δημιουργηθεί μέσω xml προσφέροντας μεγαλύτερη απόδοση στη μνήμη της εφαρμογής, αλλά και στη χωρητικότητα της. Ο κώδικας για τα κουμπιά αποτελείται από 4 xml αρχεία, ένα για κάθε στάδιο τους (normal, pressed, correct_answer, wrong_answer). Το παρακάτω xml παρουσιάζει το normal στάδιο τους:

```

<?xml version="1.0" encoding="utf-8"?>
<shape

```

```

xmlns:android="http://schemas.android.com/apk/res/android"
android:shape="rectangle">

<solid
    android:color="@color/primaryColor"/>
<corners
    android:radius="15dp"/>
<padding
    android:left="10dp"
    android:top="10dp"
    android:right="10dp"
    android:bottom="10dp" />
    <stroke android:width="2dp" android:color="@color/white" />
</shape>

```

Ενώ αντίστοιχα το Pressed στάδιο τους:

```

<?xml version="1.0" encoding="utf-8"?>
<shape
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle">

    <solid
        android:color="@color/primaryColorDark"/>
    <corners
        android:radius="15dp"/>
    <padding
        android:left="10dp"
        android:top="10dp"
        android:right="10dp"
        android:bottom="10dp" />
        <stroke android:width="2dp" android:color="@color/white" />
</shape>

```

Ολοκληρώνοντας να επισημανθεί ότι για τη δημιουργία της εφαρμογής έχει γίνει χρήση του φακέλου values του android. Κάνοντας χρήση αυτού του φακέλου, δίνεται η δυνατότητα απομόνωσης τιμών όπως όλα τα Strings, Integers, Colors βοηθώντας έτσι στην μετάφραση ή στην αλλαγή της εμφάνισης τροποποιώντας μόνο το εκάστοτε αρχείο. Για παράδειγμα το xml αποθήκευσης της χρωματικής παλέτας της εφαρμογής είναι:

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="primaryColor">#009688</color>
    <color name="primaryColorDark">#004D40</color>
    <color name="correctAnswer">#8FF7A7</color>
    <color name="wrongAnswer">#F8333C</color>
    <color name="pressedAnswer">#004F47</color>
    <color name="accentColor">#C6D4FF</color>
    <color name="white">#FFFFFF</color>
    <color name="gray">#e1e1e1</color>
    <color name="transparentTint">#88101010</color>
</resources>

```

Η χρήση του κάθε χρώματος γίνεται με τον απλό τρόπο @color/ ακολουθώντας το αντίστοιχο χρώμα που θέλουμε όπως για παράδειγμα @color/white. Ένα άλλο παράδειγμα είναι το xml αρχείο Strings όπου υπάρχουν αποθηκευμένα όλα τα λεκτικά της εφαρμογής, κάτι που επιτρέπει την εύκολη μετάφραση του αρχείου.

Ένα άκρως σημαντικό στοιχείο της υλοποίησης είναι και το αρχείο Android Manifest, τύπου xml, στο οποίο δηλώνονται τα activities, αλλά και τυχόν απαιτήσεις τα εφαρμογής τις οποίες πρέπει να αποδεχτεί ο χρήστης. Ακολουθεί το αρχείο:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="george.theocharis.knowledge" >

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.WAKE_LOCK" />
    <uses-permission android:name="android.permission.VIBRATE" />
    <uses-permission
android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
    <uses-permission android:name="android.permission.GET_ACCOUNTS" />
    <uses-permission
android:name="com.google.android.c2dm.permission.RECEIVE" />

    <permission
        android:name="com.parse.starter.permission.C2D_MESSAGE"
        android:protectionLevel="signature" />

    <uses-permission
android:name="george.theocharis.bookofknowledge.permission.C2D_MESSAGE"
    />
    <uses-permission android:name="android.permission.CAMERA" />

    <uses-feature
        android:name="android.hardware.camera"
        android:required="false" />
    <uses-feature android:name="android.hardware.camera.autofocus" />

    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

    <application
        android:name="george.theocharis.knowledge.ParseApplication"
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <meta-data
            android:name="com.facebook.sdk.ApplicationId"
            android:value="@string/facebook_app_id" />
        <activity
            android:name="com.facebook.FacebookActivity"

android:configChanges="keyboard|keyboardHidden|screenLayout|screenSize
|orientation"
            android:label="@string/app_name"/>
        <activity
android:name="george.theocharis.knowledge.activities.MainActivity"
            android:label="@string/app_name" >
        </activity>
        <activity
android:name="george.theocharis.knowledge.activities.ModesActivity"
```

```

        android:label="@string/title_activity_story_mode" >
    </activity>
</activity>

android:name="george.theocharis.knowledge.activities.SectionsActivity"
    android:label="@string/title_activity_sections"

android:parentActivityName="george.theocharis.knowledge.activities.ModesActivity" >
    <meta-data
        android:name="android.support.PARENT_ACTIVITY"
        android:value="george.theocharis.knowledge.activities.ModesActivity"
    />
</activity>
</activity>
android:name="george.theocharis.knowledge.activities.QuizActivity" >
</activity>

<service android:name="com.parse.PushService" />

<receiver android:name="com.parse.ParseBroadcastReceiver" >
    <intent-filter>
        <action
            android:name="android.intent.action.BOOT_COMPLETED" />
        <action
            android:name="android.intent.action.USER_PRESENT" />
    </intent-filter>
</receiver>
<receiver
    android:name="com.parse.ParsePushBroadcastReceiver"
    android:exported="false" >
    <intent-filter>
        <action android:name="com.parse.push.intent.RECEIVE"
        />
        <action android:name="com.parse.push.intent.DELETE" />
        <action android:name="com.parse.push.intent.OPEN" />
    </intent-filter>
</receiver>
<receiver
    android:name="com.parse.GcmBroadcastReceiver"
    android:permission="com.google.android.c2dm.permission.SEND" >
    <intent-filter>
        <action
            android:name="com.google.android.c2dm.intent.RECEIVE" />
        <action
            android:name="com.google.android.c2dm.intent.REGISTRATION" />
        <category
            android:name="george.theocharis.bookofknowledge" />
    </intent-filter>
</receiver>

<activity
    android:name="george.theocharis.knowledge.activities.IntroActivity" >
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

```

```

        <category
android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
<activity
android:name="george.theocharis.knowledge.activities.RegisterActivity"
    android:label="@string/title_activity_register" >
    </activity>
<activity
android:name="george.theocharis.knowledge.activities.LoginCredentialsA
ctivity"
    android:label="@string/title_activity_login_credentials" >
    </activity>
</application>
</manifest>

```

Τέλος, να επισημανθεί η χρήση μιας κλάσης Utilities όπου παρέχει διάφορες μεθόδους που μπορεί να είναι χρήσιμες καθ' όλη την ανάπτυξη της εφαρμογή όπως η `isConnectedToInternet`, η οποία επιστρέφει εάν υπάρχει σύνδεση στο internet.

```

public static boolean isConnectedToInternet(Context context)
{
    ConnectivityManager cm =
    (ConnectivityManager) context.getSystemService(Context.CONNECTIVITY_SER
VICE);

    NetworkInfo activeNetwork = cm.getActiveNetworkInfo();
    boolean isConnected = activeNetwork != null &&
        activeNetwork.isConnectedOrConnecting();

    return isConnected;
}

```

3.5.2 Βιβλιοθήκες, Plugins

Ένα από τα πλεονεκτήματα της επιλογής του λειτουργικού Android ως πλατφόρμα ανάπτυξης της εφαρμογής ήταν και η πληθώρα βιβλιοθηκών ανοιχτού κώδικα που υπάρχουν, αλλά και Plugins (προσθέτων).

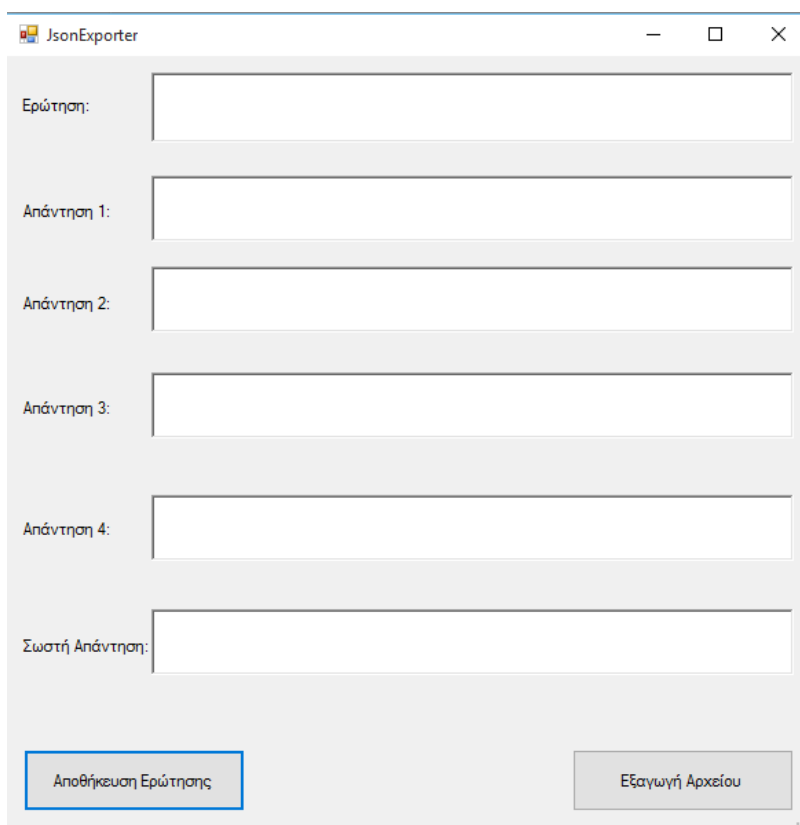
Με τον όρο βιβλιοθήκη εννοείται συνήθως μια προσθήκη κώδικα στο βασικό πρόγραμμα, ο οποίος διατίθεται δωρεάν ή επί αντιτίμου και παρέχει ένα σύνολο μεθόδων το οποίο μπορεί να χρησιμοποιηθεί σε μεγαλύτερα projects, δίνοντας πρόσβαση σε λειτουργίες που πιθανώς επαναλαμβάνονται. Για παράδειγμα μία από τις πιο γνωστές βιβλιοθήκες σε όλες τις γλώσσες προγραμματισμού είναι η Math, η οποία παρέχει όλες τις μεθόδους αριθμητικών πράξεων.

Στην εφαρμογή Kn-owl-edge χρησιμοποιήθηκαν οι ακόλουθες βιβλιοθήκες και Plugins:

- ✚ Android.Support Library. Βιβλιοθήκη η οποία είναι υπεύθυνη για τη μετατροπή νέων στοιχείων του Android που εμφανίστηκαν σε νεότερες εκδόσεις, σε συμβατή μορφή με παλαιότερες. Χωρίς αυτή τη βιβλιοθήκη δεν θα ήταν εφικτό να χρησιμοποιηθούν στοιχεία όπως το RecyclerView ή το CardView.
- ✚ De.hdodenhof.CircleImageView. Βιβλιοθήκη η οποία παρέχει μια μετατροπή του παραδοσιακού ImageView και του προσδίδει κυκλικό σχήμα.

- ✚ Lzyzsd.CircleProgress. Βιβλιοθήκη η οποία παρέχει χρονομετρητές σε διάφορες μορφές όπως κυκλική, donut κ.α.
- ✚ Bolts Framework. Παρέχει τη δυνατότητα χρήσης tasks τα οποία βοηθούν στη δημιουργία ασύγχρονου κώδικα. Είναι απαραίτητη βιβλιοθήκη για να λειτουργήσει σωστά η πλατφόρμα της Parse.
- ✚ Picasso Library. Βιβλιοθήκη που παρέχει τη δυνατότητα ασύγχρονης φόρτωσης εικόνων στα αντίστοιχα views, είτε δίνοντας κάποια εικόνα είτε δίνοντας ακόμα και σύνδεσμο σε κάποια ιστοσελίδα στο διαδίκτυο.
- ✚ Facebook SDK. Μέσω του πρόσθετου που διατίθεται δωρεάν από το Facebook.com μπορεί κάποιος να συνδέσει την εφαρμογή του με το διάσημο κοινωνικό δίκτυο.
- ✚ Parse. Η πλατφόρμα Parse διαθέτει τη δική της βιβλιοθήκη όπου παρέχει το api για την επικοινωνία με το cloud.

Τέλος, για τις ανάγκες της εφαρμογής Kn-owl-edge, δημιουργήθηκε συμπληρωματικά, ένα πρόγραμμα σε τεχνολογία windows forms⁴, ώστε να μπορεί ο καθηγητής να υποβάλλει τις ερωτήσεις στο σύστημα της Parse, μόνος του, δίχως τη βοήθεια του προγραμματιστή. Συγκεκριμένα, το πρόγραμμα αποτελείται από μία φόρμα, η οποία περιέχει textboxes για να εισάγονται οι ερωτήσεις καθώς και οι απαντήσεις κάθε ενότητας. Ακολουθούν 2 κουμπιά(buttons) για την τοπική αποθήκευση των ερωτήσεων αρχικά και σε δεύτερο χρόνο για την εξαγωγή τους σε αρχείο json. Ακολουθεί σχετική εικόνα του προγράμματος:



Εικόνα 23 Questions Exporter Form

Η μορφή με την οποία γίνεται το export είναι η μορφή json, πλέον δημοφιλής για τη μεταφορά δεδομένων μέσω web services σε κινητά τηλέφωνα. Η δομή ενός αρχείου όπως των ερωτήσεων είναι αυτή που ακολουθεί:

⁴ Τα windows forms γράφονται σε γλώσσα προγραμματισμού C# και αποτελούν μια από τις πιο δημοφιλείς τεχνολογίες της Microsoft.

```

{
  "results":
  [
    {
      "Title": "Ποιό από τα παρακάτω δεν είναι δευτερογενές λεμφικό όργανο;",
      "Answer1": "σπλήνας",
      "Answer2": "λεμφαδένες",
      "Answer3": "ήπαρ",
      "Answer4": "όλα όσα αναφέρονται είναι",
      "CorrectAnswer": "ήπαρ"
    },
    {
      "Title": "Δευτερογενές λεμφικό όργανο είναι ο:",
      "Answer1": "σπλήνας",
      "Answer2": "θύμος αδένας",
      "Answer3": "μυελός των οστών",
      "Answer4": "πνεύμονας",
      "CorrectAnswer": "σπλήνας"
    }
  ]
}

```

Εικόνα 24 Json Format

Κεφάλαιο 4 – Επίλογος

Σε αυτό το κεφάλαιο θα αναλύσουμε τα συμπεράσματα όπως αυτά διαμορφώθηκαν με το πέρας της μεταπτυχιακής διατριβής αλλά και τις προτάσεις μας για μελλοντικές προσθήκες και βελτιώσεις με βασικό στόχο την ανάπτυξη της εφαρμογής.

4.1 Συμπεράσματα

Μέσα από την συστηματική έρευνα τόσο στο διαδίκτυο αλλά και σε σχετική βιβλιογραφία προκύπτουν ορισμένα συμπεράσματα τα οποία είναι:

- Η τεχνολογία έχει πλέον φτάσει σε τέτοιο επίπεδο όπου είναι ικανή να βελτιώσει σημαντικά και να αναβαθμίσει το επίπεδο της εκπαίδευσης.
- Οι έφηβοι είναι πια πολύ εξοικειωμένοι με ηλεκτρονικές συσκευές, όπως Η/Υ και smartphones επομένως νιώθουν ικανοί να τις χρησιμοποιούν σε πολλούς και διαφορετικούς τομείς της καθημερινότητάς τους, μεταξύ αυτών και για την επιμόρφωσή τους.
- Οι εκπαιδευτικοί έχουν αντιληφθεί την ανάγκη για μια πιο ποιοτική εμπειρία μάθησης και ψάχνουν νέους τρόπους διδασκαλίας με σύμμαχό τους την τεχνολογία.
- Στον ελληνικό χώρο υπάρχει πρόσφορο έδαφος για τη δημιουργία πλήθους εκπαιδευτικών εφαρμογών που μπορούν να αποτελέσουν ευρηματικά μέσα μάθησης.
- Μέχρι στιγμής δεν υπάρχει κάποια εφαρμογή που να καλύπτει το κενό αυτό και παράλληλα να ψυχαγωγεί το χρήστη, πόσο μάλλον όταν είναι νεαρής ηλικίας.
- Οι εφαρμογές με τη «λογική» του παιχνιδιού αποτελούν τη βάση για μια πιο ενδιαφέρουσα εκδοχή του διαβάσματος.
- Ένα κομμάτι γνώσεων είναι ο απλούστερος τρόπος κινητοποίησης των νεαρών παικτών οι οποίοι εκπαιδεύονται με ευφάνταστο τρόπο και προετοιμάζονται για τη συμμετοχή τους σε εξετάσεις.
- Μέσα από αυτήν την διαδικασία οι ίδιοι αποβάλλουν το στρες και παράλληλα νιώθουν αυτοπεποίθηση.
- Οι νέες εφαρμογές αποκτούν άλλο νόημα μέσω της εκπαίδευσης.
- Οι γονείς αισθάνονται ότι θα παιδιά τους έχουν μια πιο ολοκληρωμένη εκπαίδευση και ότι οι κόποι τους επιφέρουν κάποιο αποτέλεσμα.

- Επίσης σε ένα περιβάλλον που έχει ως θεμελιώδη στόχο να προσφέρει γνώση ο γονιός νιώθει ότι το παιδί του είναι προστατευμένο από τους συχνούς κινδύνους του διαδικτύου.
- Οι εκπρόσωποι του κλάδου της πληροφορικής που έχουν αντιληφθεί την ανάγκη αυτή γίνονται πιο ευρηματικοί και ανταγωνιστικοί ώστε το έργο τους να γίνει γνωστό και να προσελκύσει το ενδιαφέρον των χρηστών.
- Ακόμα και δημιουργοί καταξιωμένων ηλεκτρονικών παιχνιδιών με εκατομμύρια πωλήσεις στο ενεργητικό τους έχουν αντιληφθεί τη σπουδαιότητα της σωστής εκπαίδευσης.

4.2 Προτάσεις για μελλοντική ανάπτυξη της εφαρμογής Kn-owl-edge

Η εφαρμογή Kn-owl-edge είναι σε αυτό το σημείο μια πλήρης εφαρμογή με περιεχόμενο, το οποίο εκμεταλλεύεται όλες τις μοντέρνες τάσεις και τεχνικές της πληροφορικής και της τεχνολογίας, προσφέροντας μάθηση και ψυχαγωγία. Τα επόμενα βήματα που μπορούν να γίνουν στο μέλλον είναι πολλά και εξίσου σημαντικά.

Ακολουθούν κάποιες προτάσεις, οι οποίες μπορούν να εξελίξουν περαιτέρω μια εφαρμογή τέτοιου είδους.

Προτάσεις:

- Προσθήκη όλων των μαθημάτων των Πανελληνίων. Η προσθήκη αυτή θα μετέτρεπε την εφαρμογή σε έναν πλήρες βοηθό για το μαθητή και τις εξετάσεις του.
- Περαιτέρω εξειδίκευση στις επιλογές, όπως επιλογή κατεύθυνσης (Τεχνολογική, Θεωρητική και Θετική).
- Προσθήκη παλιών λυμένων θεμάτων για εύκολη πρόσβαση από τους μαθητές.
- Δημιουργία επιπέδων με διάφορους τίτλους, όπως μαθητής πρώτου τριμήνου, όταν δεν έχει ξεκλειδώσει νέες ενότητες ή τελειόφοιτος όταν τα ξεκλειδώσει όλα.
- Προσθήκη διαφορετικών τύπων προβλημάτων, εκτός από πολλαπλής επιλογής, για να προστεθούν μαθήματα όπως η έκθεση.

Μέσα από αυτές τις προσθήκες, οι οποίες θα γίνουν μελλοντικά, η εφαρμογή Kn-owl-edge θα αποτελέσει έναν ολοκληρωμένο βοηθό πανελληνίων, ο οποίος θα έχει στόχο να αποδείξει ότι η εκπαιδευτική διαδικασία και ο τομέας της Πληροφορικής είναι πλέον αναπόσπαστες όψεις του ίδιου νομίσματος. Επιπλέον, θα έχει καταφέρει να αναδείξει, αλλά και να συμπληρώσει, το κενό το οποίο υπάρχει αυτή τη στιγμή στις επιλογές τέτοιου είδους εφαρμογών.

Το συμπέρασμα της διατριβής είναι ότι εκπαίδευση και ψυχαγωγία δεν είναι πια, όροι ασύνδετοι, αλλά μέσω της τεχνολογίας και του ευφυούς συνδυασμού τους, μπορούν πλέον να αποτελέσουν μια χείρα βοήθειας στους εκπαιδευτικούς, αλλά κυρίως στους μαθητές και να αποτελέσουν το εφάθηριο για μια νέα μορφή εκπαίδευσης. Μιας μορφής εκπαίδευσης, μέσω της συμμετοχής και του παιχνιδιού, τακτική που υμνείται ανά τους αιώνες, από τον Πλάτωνα ως τον Αϊνστάιν.

Βιβλιογραφία

Calvert, S.L., & Tan, S. (1994), Impact of virtual reality on young adults' physiological arousal and aggressive thoughts: Interaction versus observation. Special Issue: Effects of interactive entertainment technologies on development.

Cassell, J & Jenkins, H. (1998), From Barbie to Mortal Kombat: Gender and Computer Games. Cambridge, MA: MIT Press.

Amory, A., Naicker, K., Vincent, J. & Claudia, A. (1998), Computer Games as a Learning Resource. Proceedings of ED-MEDIA, EDTELECOM 98, World Conference on Education Multimedia and Educational Telecommunications, Vol. 1, 50-55.

Appleman, R. & Goldsworthy, R. 1999. The Juncture of Games & Instructional Design: Can Fun be Learning?". Houston, TX.

Aldrich, Clark. 2003. Simulations and the Future of Learning: An Innovation (and Perhaps Revolutionary) Approach to E-Learning (Business). John Wiley & Sons

Virvou, M., Katsionis, G., & Manos, K. (2005), Combining Software Games with Education: Evaluation of its Educational Effectiveness. Educational Technology & Society, Journal of International Forum of Educational Technology & Society and IEEE Learning Technology Task Force, April 2005, Vol. 8, No 2.

Eckel, Bruce. 2006. Thinking in Java. Prentice Hall.

Bloch, Joshua. 2008. Effective Java, Second Edition. Addison-Wesley.

Αναγνώστου, Κώστας. 2009. Βιντεοπαιχνίδια: Βιομηχανία και Ανάπτυξη. Αθήνα: Κλειδάριθμος.

Δρ. Μπαρμπάτσης, Κωνσταντίνος, Δρ. Οικονόμου, Δάφνη, Παπαμαγκανά, Ιωάννα και Ζώζας, Ιωάννης. 2010. Ηλεκτρονικά Παιχνίδια ως Εκπαιδευτικά Εργαλεία. 2^ο Πανελλήνιο Εκπαιδευτικό Συνέδριο Ημαθίας.

Spaniel, William. 2011. Game Theory 101: The Complete Textbook. CreateSpace Independent Publishing Platform.

Kapp, M. Karl. 2012. The Gamefication of Learning and Instruction: Game – Based Methods and Strategies for Training and Education. John Wiley & Sons.

McGonigal, Jane. 2012. Reality is Broken: Why Games Make Us Better and How They Can Change the World. Vintage.

Annuzzi Jr., Joseph, Darcey, Lauren and Conder, Shane. 2014. Advanced Android Application Development. Addison-Wesley.

Pegrum, Mark. 2014. Mobile Learning (New Language Learning and Teaching Environments). Palgrave Macmillan.

Gerber, Adam. 2015. Learn Android Studio: Build Android Apps Quickly and Effectively. Apress.

Chou, Yu-kai. 2015. Actionable Gamification: Beyond Point, Badges and Leaderboards. CreateSpace Independent Publishing Platform.

Middleton, Andrew. 2015. Smart Learning: Teaching and learning with smartphones and tablets. Sheffield Hallam University.

McQuiggan, Scott et al. 2015. Mobile Learning: A Handbook for Developers, Educators, and Learners. John Wiley & Sons.

Ιστογραφία

<http://i-teacher.gr>

<http://www.smartedu.gr/interactive-whiteboards-smartboard-tm-smarttech>

<http://www.usq.edu.au/course/material/EDU5472/content/mod08.htm>

<http://www.digitalgames.edu.gr/epsilonkappapialphaiotadeltaepsilonupsilontaiotakappa940-pialphaiotachinu943deltaiotaalpha.html>

<http://www.kathimerini.gr/784679/gallery/tecnologia/games/h-arxaia-ellada-sto-minecraft>

<http://developer.android.com/develop/index.html>

<https://developers.facebook.com/>

<https://parse.com/docs/android/guide>

<https://parse.com/docs/android/api/>