



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Πρόγραμμα Μεταπτυχιακών Σπουδών

Τμήμα Ψηφιακών Συστημάτων

«Διδακτική της Τεχνολογίας & Ψηφιακά Συστήματα»

Κατεύθυνση: «Ηλεκτρονική Μάθηση»

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**«Δημιουργία Tutorial για το Εργαλείο Οπτικού
Προγραμματισμού MIT App Inventor & Ανάπτυξη
Πρότυπης Εφαρμογής»**

Βλάσση Ελένη (ME11003)

Επιβλέπων: Ρετάλης Συμεών,

Επίκουρος καθηγητής Πανεπιστημίου Πειραιά

Πειραιάς, Νοέμβριος 2015

Στον Νίκο μου...

Περίληψη

Το MIT App Inventor είναι ένα δωρεάν οπτικό περιβάλλον προγραμματισμού που μας δίνει την δυνατότητα να δημιουργούμε εφαρμογές για κινητά τηλέφωνα τα οποία χρησιμοποιούν android λογισμικό. Η λογική του εργαλείου βασίζεται στη WYSIWYG (συντομογραφία της φράσης "What You See Is What You Get" (Αυτό που βλέπεις είναι αυτό που θα εμφανιστεί)) λύση της Google, που επιτρέπει τη δημιουργία εφαρμογών για το δημοφιλές λειτουργικό της Android, μέσω απλού drag'n'drop. Έτσι, θεωρητικά, ακόμη και ένας αρχάριος με μηδενικές γνώσεις προγραμματισμού χρήστης, μπορεί μέσω του App Inventor μέσα σε λιγότερο από μια ώρα να έχει δημιουργήσει τη δική του android εφαρμογή.

Η παρούσα εργασία ως σκοπό έχει να δημιουργήσει ένα λεπτομερές εγχειρίδιο χρήσης του app inventor. Για το λόγο αυτό παρουσιάζεται αναλυτικά το τι είναι το εργαλείο αυτό, πού μπορεί να βρει εφαρμογή, το έργο της ομάδας του, οι συνεργάτες, καθώς και οδηγίες φορμαρίσματος και επεξήγηση της κάθε εντολής και στοιχείου που παρέχεται από το εργαλείο και μπορεί ο χρήστης να χρησιμοποιήσει για να φτιάξει το δικό του application. Επιπλέον, για την καλύτερη κατανόηση του εργαλείου, αναπτύχθηκε μια πρότυπη εφαρμογή μέσω της οποίας ο χρήστης μπορεί να δει τη θεωρία να μετατρέπεται σε πράξη.

Abstract

MIT app inventor is a free, visual programming environment that gives us the opportunity to create applications for mobile phones, which use android software. The logic of the tool is based on WYSIWYG (abbreviation for "What You See Is What You Get" (What you see is what you will receive)) solution of Google, allowing to create applications for popular Android operating system, via simple drag 'n' drop. So theoretically, even a novice user with zero programming knowledge can, through the App Inventor in less than an hour, create his own android application.

This paper is intended to create a detailed user manual for app inventor. Therefore, it is presented in detail what the tool is, in which fields it can be applied, the work of its team, the partners as well as set up instructions and an explanation of each command and component provided by the environment that users can use to make their own application. In addition, to better understand the tool, a standard application is developed through which the user can see theory be turned into practice.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον Επίκουρο Καθηγητή κo Συμεών Ρετάλη για την επίβλεψη και τη βοήθεια που μου παρείχε για την ολοκλήρωση της διπλωματικής μου, καθώς και την υπομονή και την κατανόησή που έδειξε όλο αυτό το διάστημα.

Ευχαριστώ, επίσης, τους διδάσκοντες που καθ' όλη την διάρκεια της μεταπτυχιακής μου φοίτησης μου προσέφεραν πολύτιμες γνώσεις και έμπρακτο ενδιαφέρον.

Ακόμη, ένα ιδιαίτερο ευχαριστώ οφείλω στους φίλους μου που δεν σταμάτησαν στιγμή να με εμπυχώνουν και να πιστεύουν σε εμένα.

Τέλος, εκφράζω την ευγνωμοσύνη μου στους γονείς μου για την υποστήριξη και βοήθειά τους σε όλη τη διάρκεια των μεταπτυχιακών σπουδών μου. Χωρίς αυτούς δεν θα τα είχα καταφέρει.

Περιεχόμενα

Περίληψη	iii
Abstract	iv
Ευχαριστίες	v
Κεφάλαιο 1	1
1.1 Τι είναι το app inventor	1
1.2 Σκοπός της διπλωματικής εργασίας	2
1.3 Αναγκαιότητα του app inventor tutorial	4
1.4 Χρησιμότητα του app inventor tutorial	4
1.5 Δομή του tutorial	5
1.5.1 Σε ποιόν απευθύνεται το tutorial	6
1.5.2 Εκπαιδευτική φιλοσοφία του tutorial	7
1.6 Μάθηση μέσω μελέτης περίπτωσης	9
1.6.1 Βήματα μιας μελέτης περίπτωσης	10
Κεφάλαιο 2	22
2.1 Λογισμικά εφαρμογών για κινητές συσκευές	22
2.2 Εργαλεία (γλώσσες) οπτικού προγραμματισμού	23
2.2.1 Python	23
2.2.2 Scratch	25
2.2.3 Visual Basic	26
2.2.4 Kodu	29
2.2.5 Easy Logo	30
2.2.6 Turtle Art	31
2.2.7 StarLogo & StarLogo TNG	33
2.2.8 BYOB	35
2.2.9 Alice	36
2.2.10 Game Maker	38
2.2.11 Microworld pro	39
Κεφάλαιο 3	41
3.1 Ανάλυση του app inventor	41
3.1.1 Το έργο της ομάδας του app inventor	42
3.1.2 Οι συνεργάτες	43
3.1.3 Οδηγίες φορμαρίσματος	44

3.1.4 Ανάλυση της οθόνης σχεδιασμού (designer)	62
3.1.5 Ανάλυση της οθόνης των blocks (blocks editor).....	186
3.1.6 Πακετάρισμα και μοίρασμα εφαρμογών	222
Κεφάλαιο 4	227
4.1 Σχεδιασμός εφαρμογής.....	227
4.2 Αρχιτεκτονική εφαρμογής.....	228
4.3 Πλεονεκτήματα αρχιτεκτονικής 3ων επιπέδων.....	231
4.4 Σχεδίαση βάσης δεδομένων (database).....	232
4.5 Επίπεδο λειτουργικότητας.....	240
4.5.1 Περιπτώσεις Χρήσης	240
4.5.2 Κλάσεις Αντικειμένων	241
4.5.3 Σχεδίαση Διεπαφών	245
4.6 Υλοποίηση web server	247
4.7 Υλοποίηση εφαρμογής με app inventor	248
Κεφάλαιο 5	265
5.1 Αξιολόγηση εφαρμογής	265
5.1.1 Ευρήματα ερωτηματολογίου	267
5.2 Πλεονεκτήματα και μειονεκτήματα app inventor	277
5.3 Μελλοντικές επεκτάσεις εφαρμογής.....	279
5.4 Συμπεράσματα.....	280
Βιβλιογραφικές αναφορές	282
Παράρτημα.....	287
Ερωτηματολόγιο.....	287

Περιεχόμενα Πινάκων

Πίνακας 1: Εκπαιδευτικός στόχος μελέτης περίπτωσης.....	12
Πίνακας 2: Σχέση συγκεκριμένης μελέτης με άλλες	12
Πίνακας 3: Παρουσίαση εκπαιδευτικού υλικού που συλλέχθηκε	13
Πίνακας 4: Ερωτήσεις σε περίπτωση που η μελέτη είναι βασισμένη σε στοιχεία που συλλέχθηκαν απευθείας από τον οργανισμό	14
Πίνακας 5: Χρήση μελέτης περίπτωσης	16
Πίνακας 6: Σημεία στα οποία δίνει έμφαση η μελέτη περίπτωσης	17
Πίνακας 7: Η δομή του θέματος της μελέτης	18
Πίνακας 8: Αναμενόμενα μαθησιακά αποτελέσματα	19
Πίνακας 9: Σχέσεις οντοτήτων	236
Πίνακας 10: Πεδία του πίνακα users	237
Πίνακας 11: Πεδία του πίνακα categories	238
Πίνακας 12: Πεδία του πίνακα webpage	238
Πίνακας 13: Πεδία του πίνακα users categories	239
Πίνακας 14: Πεδία του πίνακα webpage categories	239
Πίνακας 15: Ερωτήσεις ερωτηματολογίου	269

Περιεχόμενα Σχημάτων

Σχήμα 1: Βαθμοτή αρχιτεκτονική τεχνοτροπία τριών επιπέδων	229
Σχήμα 2: Υψηλότερο επίπεδο- Client.....	230
Σχήμα 3: Ενδιάμεσο επίπεδο- Application server	231
Σχήμα 4: Κατώτερο επίπεδο- Database server	231
Σχήμα 5: Σχεδίαση βάσης δεδομένων	233
Σχήμα 6: Διάγραμμα οντοτήτων- συσχετίσεων	237
Σχήμα 7: Σχισιακό μοντέλο.....	240
Σχήμα 8: Διάγραμμα use- case	242
Σχήμα 9: Διάγραμμα κλάσεων.....	245
Σχήμα 10: Ερώτηση 6/ Ενασχόληση με προγραμματισμό	270
Σχήμα 11: Ερώτηση 7/ Ενασχόληση με παρόμοιο εργαλείο όπως το app inventor ..	271
Σχήμα 12: Ερώτηση 8/ Πώς βρίσκετε το εργαλείο app inventor.....	272
Σχήμα 13: Ερώτηση 13/ Ανάγκη για περαιτέρω ανάλυση του εργαλείου.....	273
Σχήμα 14: Ερώτηση 14/ Ανάπτυξη της γενικότερης κατανόησης του προγραμματισμού	274
Σχήμα 15: Ερώτηση 15/ Θα ξαναχρησιμοποιήσετε το app inventor?	275
Σχήμα 16: Ερώτηση 9/ Δημιουργία εφαρμογής με app inventor	276
Σχήμα 17: Ερώτηση 10/ Πόσο βοήθησε το εγχειρίδιο στην κατανόηση του app inventor	276
Σχήμα 18: Ερώτηση 12/ Σε τι θέση βρίσκομαι μετά την ολοκλήρωση του εγχειριδίου	277
Σχήμα 19: Ερώτηση 11/ Συμβολή του πρότυπου παραδείγματος στην κατανόηση του app inventor	278

Κεφάλαιο 1

1.1 Τι είναι το App Inventor



Εικόνα 1- App Inventor

Το App Inventor για Android αποτελεί ένα νέο, δωρεάν οπτικό περιβάλλον προγραμματισμού με πλακίδια (blocks), για τη δημιουργία εφαρμογών για κινητά τηλέφωνα με Λειτουργικό Σύστημα Android. Το App Inventor δίνει την ευκαιρία σε όποιον ενδιαφέρεται αλλά δεν έχει τις απαραίτητες γνώσεις, να φτιάξει μια εφαρμογή για Android. Είναι ουσιαστικά το μοναδικό δωρεάν εργαλείο για εύκολη ανάπτυξη εφαρμογών.

Αναπτύχθηκε στα εργαστήρια της Google από μια ομάδα με επικεφαλή τον καθηγητή του MIT Hal Abelson (Abelson, 2009). Το App Inventor χρησιμοποιείται και δοκιμάζεται ήδη ως πλατφόρμα διδασκαλίας και εισαγωγής στον Προγραμματισμό, τόσο στην Τριτοβάθμια, όσο και στη σχολική εκπαίδευση. Το περιβάλλον του App Inventor έχει πολλές ομοιότητες με το περιβάλλον του Scratch (Το Scratch είναι ένα περιβάλλον προγραμματισμού σχεδιασμένο για εκπαίδευση και ψυχαγωγία) και του Alice (είναι ένα πρωτοποριακό 3D περιβάλλον προγραμματισμού), με τη διαφορά ότι οι εφαρμογές που δημιουργούνται τρέχουν σε έξυπνα τηλέφωνα (smart phones).

Το App Inventor είναι μια WYSIWYG (συντομογραφία της φράσης "What You See Is What You Get" (Αυτό βλέπεις είναι αυτό που θα εμφανιστεί)) λύση της Google, που επιτρέπει τη δημιουργία εφαρμογών για το δημοφιλές λειτουργικό της Android, μέσω απλού drag'n'drop. Θεωρητικά, ακόμη και ένας αρχάριος με μηδενικές γνώσεις προγραμματισμού, μπορεί μέσω του App Inventor μέσα σε λιγότερο από μια ώρα να έχει δημιουργήσει τη δική του εφαρμογή για το Android. Πρακτικά, δεν χρειάζεται καν να έχει κάποιος ένα smartphone ή tablet με αυτό το λειτουργικό σύστημα, αφού

μπορεί να βασιστεί σε μια "εξομοιωμένη" εκδοχή του για να δει τους καρπούς της προσπάθειάς του.

Όπως ήδη αναφέρθηκε, για τη χρήση της εφαρμογής δεν είναι απαραίτητες οι γνώσεις προγραμματισμού, μιας και αντί για τη συγγραφή κώδικα ο δημιουργός αποφασίζει για την εμφάνιση και τη συμπεριφορά της εφαρμογής μέσα από ένα καλαίσθητο παραθυρικό περιβάλλον. Μάλιστα, οι δημιουργοί της πρόσθεσαν διάφορα blocks εντολών, τα οποία «ενεργοποιούνται» πατώντας το αντίστοιχο πλήκτρο και εκτελούν προκαθορισμένες λειτουργίες.

Για παράδειγμα, υπάρχουν blocks επανάληψης εντολών τύπου for, do-while κτλ, blocks αποθήκευσης πληροφοριών, καθώς και blocks κώδικα για την εκτέλεση λειτουργιών κάτω από συγκεκριμένες συνθήκες. Επίσης, είναι διαθέσιμα ακόμη και blocks για επικοινωνία με τοποθεσίες κοινωνικής δικτύωσης, όπως π.χ. το Twitter.

Άλλες ενδιαφέρουσες πτυχές του App Inventor είναι η συνεργασία με τους δέκτες GPS, ώστε να «χτιστούν» εφαρμογές που θα χρησιμοποιούν πληροφορίες εντοπισμού θέσης. Ακόμη υπάρχουν αρκετές άλλες εφαρμογές, ίσως πιο πρακτικής φύσεως, οι οποίες εκμεταλλεύονται ή υποβοηθούν τις «καθημερινές» λειτουργίες ενός κινητού τηλεφώνου.

Καταλήγοντας, να σημειώσουμε πως η εφαρμογή App Inventor για Android κάνει χρήση των Java βιβλιοθηκών Open Blocks για την δημιουργία των εικονικών blocks, ενώ ο μεταφραστής της γλώσσας των visual blocks (η οποία ενσωματώθηκε και ρυθμίστηκε ειδικά για το Android) χρησιμοποιεί το Kawa Language Framework.

Το «App Inventor» (Εφευρέτης Εφαρμογών) απεικονίζει τις επιμέρους λειτουργίες του τηλεφώνου ως χρωματιστά τουβλάκια, τα οποία οι χρήστες μπορούν να σέρνουν και να συνδέουν πάνω στην οθόνη του υπολογιστή τους. Οι μαθητευόμενοι προγραμματιστές μπορούν έτσι να δημιουργούν παζλ και άλλα παιχνίδια ή προγράμματα που αξιοποιούν ταυτόχρονα πολλές επιμέρους δυνατότητες του Android.

1.2 Σκοπός Διπλωματικής

Σκοπός της παρούσας διπλωματικής εργασίας είναι η δημιουργία ενός εκπαιδευτικού υλικού που στόχο θα έχει να βοηθήσει στη κατανόηση της λειτουργίας του περιβάλλοντος οπτικού προγραμματισμού app inventor αλλά και στην εκμάθηση της δημιουργίας εφαρμογών για κινητές συσκευές μέσω του προγράμματος αυτού.

Πιο αναλυτικά, σκοπός του tutorial είναι:

1. η κατανόηση της χρησιμότητας και της ευκολίας που παρέχει το app inventor σε άτομα τα οποία δεν είναι εξοικειωμένα με την σύνταξη γλωσσών προγραμματισμού,
2. η παρουσίαση στον εκπαιδευόμενο της δυνατότητα χρήσης των εφαρμογών που αναπτύσσονται μέσω του εργαλείου σε διάφορους τομείς,
3. η συμβολή στην βαθύτερη κατανόηση από πλευράς του χρήστη των νέων τεχνολογιών και των τομέων εφαρμογής τους,
4. η παρακίνηση του εκπαιδευόμενου να ξεκινήσει τον πειραματισμό με το εργαλείο και να μπορέσει να δημιουργήσει τις δικές του εφαρμογές.

Θεμέλιος λίθος του εκπαιδευτικού υλικού είναι η δημιουργία μιας πρότυπης εφαρμογής μέσα από την ανάλυση και εξήγηση της οποίας, στοχεύεται η καλύτερη παρουσίαση του εργαλείου.

Η εφαρμογή που αναπτύχθηκε προσφέρει στους χρήστες τη δυνατότητα να επιλέξουν μέσα από μια ευρεία γκάμα κατηγοριών που τους παρέχονται, βασισμένες στα ενδιαφέροντα που μπορεί να έχει ο καθένας από αυτούς. Έτσι, επιλέγοντας τις κατηγορίες οι οποίες τον ενδιαφέρουν, ο χρήστης θα μπορεί κάθε φορά που ανοίγει την εφαρμογή να ενημερώνεται σχετικά με αυτές μέσω διαφόρων πηγών όπως βίντεο, εικόνων, άρθρων κ.λ.π. Ακόμη, υπάρχει η δυνατότητα αποθήκευσης των σελίδων που επιθυμούνται ώστε να μπορεί να επισκεφτεί αυτές οποιαδήποτε χρονική στιγμή θελήσει.

Για την λειτουργία της εφαρμογής εκτός από τον σχεδιασμό αυτής με το εργαλείο app inventor, χρειάστηκε η δημιουργία μιας βάσης δεδομένων όπου αποθηκεύονται τα περιεχόμενα και γίνεται η διαχείριση αυτών από τον διαχειριστή.

1.3 Αναγκαιότητα του app inventor tutorial

Η ανάγκη για δημιουργία του συγκεκριμένου tutorial για το app inventor δημιουργήθηκε, αρχικά, καθώς δεν υπήρχε αντίστοιχο ολοκληρωμένο εκπαιδευτικό περιεχόμενο στην ελληνική γλώσσα.

Η επίσημη ηλεκτρονική σελίδα του app inventor (<http://appinventor.mit.edu/explore/>), παραθέτει το περιεχόμενο μόνο στην αγγλική γλώσσα, και οι πληροφορίες που διατίθενται στο υπόλοιπο διαδίκτυο είναι σκόρπιες, ελλείψεις και συγκεχυμένες. Ως εκ τούτου, κρίθηκε ότι ένα ολοκληρωμένο εγχειρίδιο του app inventor στην ελληνική γλώσσα θα ήταν χρήσιμο βοήθημα για όσους Έλληνες θέλουν να μάθουν σχετικά με αυτό, αλλά και πώς να το χρησιμοποιήσουν για να μπορούν να καλύψουν προσωπικές και επαγγελματικές τους ανάγκες.

Η ανάλυση των εργαλείων, της δομής και των δυνατοτήτων του app inventor δίνουν τη δυνατότητα στον χρήστη να κατανοήσει το εργαλείο και τις αρχές που το διέπουν, να μπει στη λογική του προγράμματος και να μπορέσει να το κάνει κτήμα του.

Εν συνεχεία, με την παράθεση και ανάλυση της αναπτυχθείσας εφαρμογής πάμε το tutorial ένα βήμα παραπέρα στοχεύοντας στην σφαιρική και πιο ολοκληρωμένη κατανόηση του εργαλείου και στην πλέον, εκ μέρους και του χρήστη, δυνατότητα ανάπτυξης δικής του εφαρμογής μέσω του app inventor.

1.4 Χρησιμότητα του app inventor tutorial

Ο εκάστοτε εκπαιδευόμενος που θα έχει ολοκληρώσει το συγκεκριμένο εγχειρίδιο για το app inventor, θα είναι σε θέση να:

- γνωρίζει τις βασικές αρχές και λειτουργίες που διέπουν αυτό,
- συζητάει και να αναλύει ό,τι αφορά το app inventor,
- γνωρίζει παρεμφερή προγράμματα δημιουργίας εφαρμογών για κινητές συσκευές,
- γνωρίζει τις αρχές που διέπουν τη διαδικασία ανάπτυξης μιας εφαρμογής και να κρίνει σωστά τις ήδη υπάρχουσες εφαρμογές που έχουν δημιουργηθεί με το συγκεκριμένο εργαλείο,

- δημιουργεί ολοκληρωμένες εφαρμογές για κινητά τηλέφωνα κάνοντας σωστή χρήση του app inventor.

1.5 Δομή του tutorial

Η παρούσα διπλωματική εργασία έχει την μορφή εγχειριδίου (tutorial) και είναι χωρισμένη σε πέντε κεφάλαια, όπως παρουσιάζονται παρακάτω.

Κεφάλαιο 1^ο: Στο πρώτο κεφάλαιο, γίνεται μια παρουσίαση του τι είναι το εργαλείο οπτικού προγραμματισμού app inventor, έτσι ώστε να μπορέσει ο χρήστης να έχει μια πρώτη εικόνα για το αντικείμενο το οποίο πραγματεύεται η εργασία. Αναφέρεται, ακόμη, η αναγκαιότητα και η χρησιμότητα της δημιουργίας του συγκεκριμένου tutorial, καθώς και η εκπαιδευτική φιλοσοφία αυτού και η ανάλυση του τρόπου μάθησης που χρησιμοποιήθηκε.

Κεφάλαιο 2^ο: Στο δεύτερο κεφάλαιο, γίνεται μια μικρή αναφορά στα λογισμικά εφαρμογών για κινητές συσκευές και στη συνέχεια παρουσιάζονται εργαλεία προγραμματισμού παρεμφερή με αυτό του app inventor.

Κεφάλαιο 3^ο: Στο τρίτο κεφάλαιο, γίνεται ανάλυση του εργαλείου app inventor. Αρχικά, παρατίθενται οδηγίες φορμαρίσματος για την σωστή χρήση αυτού και γνωστοποίηση των δυνατοτήτων που προσφέρει. Εν συνεχεία, γίνεται εκτενής παρουσίαση τόσο της οθόνης σχεδιασμού (design), όσο και της οθόνης των blocks (blocks editor). Στην οθόνη σχεδιασμού της εφαρμογής παρατίθενται αναλυτικά όλα τα παράθυρα που την απαρτίζουν και γίνεται επεξήγηση των στοιχείων (components), των ιδιοτήτων (properties), των γεγονότων (events) και των μεθόδων (methods) που αυτά μπορεί να έχουν. Αντίστοιχα, στην οθόνη του blocks editor, γίνεται ανάλυση των εντολών που υπάρχουν έτοιμες προς χρήση. Ακόμη, αναφέρονται οι τομείς που μπορούν να βρουν χρήση οι εφαρμογές που δημιουργούνται από το AI (app inventor), καθώς και οι σκοποί τους οποίους μια τέτοια εφαρμογή μπορεί να εκπληρώσει. Τέλος, παρουσιάζεται το πώς γίνεται το πακετάρισμα των ολοκληρωμένων εφαρμογών και το μοίρασμα αυτών.

Κεφάλαιο 4^ο: Στο τέταρτο κεφάλαιο, περνάμε στην ανάλυση της εφαρμογής που δημιουργήθηκε έτσι ώστε ο χρήστης του tutorial να μπορέσει να κατανοήσει εις βάθος, μέσω ενός απτού παραδείγματος, τις δυνατότητες του AI. Αναλύεται, λοιπόν,

ο γενικός σχεδιασμός της αναπτυχθείσας εφαρμογής, η αρχιτεκτονική και η λειτουργικότητά της, ο γραφικός της σχεδιασμός, η ανάπτυξη του web server που χρειάστηκε να δημιουργηθεί, και τέλος η εφαρμογή ολοκληρωμένη.

Κεφάλαιο 5^ο: Στο κεφάλαιο πέμπτο, που είναι και το τελευταίο κεφάλαιο της παρούσας διπλωματικής, γίνεται η τελική αξιολόγηση της εφαρμογής μέσω ερωτηματολογίου, αναφέρονται τα πλεονεκτήματα και τα μειονεκτήματα που παρατηρήθηκαν και προέκυψαν μέσα από την χρήση του app inventor, καθώς και οι επεκτάσεις που μπορεί να έχει η αναπτυχθείσα εφαρμογή στο μέλλον. Τέλος, παρατίθενται τα συμπεράσματα που προέκυψαν από την όλη έρευνα και διαδικασία δημιουργίας του application.

1.5.1 Σε ποιόν απευθύνεται το tutorial

Το App Inventor μπορεί να χρησιμοποιηθεί από μαθητές Γυμνασίου και Λυκείου για την δημιουργία εφαρμογών, σε κινητά με λειτουργικό σύστημα Android, σε ένα περιβάλλον προγραμματισμού παρόμοιο με του Scratch και της StarLogo TNG, μέχρι επαγγελματίες προγραμματιστές, εκπαιδευτικούς, φοιτητές και απλούς χρήστες. Το ελεύθερο και ανοικτό αυτό περιβάλλον οπτικού προγραμματισμού είναι ιδανικό για χρήστες με ελάχιστη ή και καθόλου σχετική εμπειρία.

Πιστεύεται ότι η πλατφόρμα του App Inventor, ως κατάλληλα σχεδιασμένη για αρχάριους στον προγραμματισμό χρήστες και βελτιστοποιημένη για εκπαιδευτική χρήση, δίνει κίνητρα στους χρήστες, καθώς τους επιτρέπει να δημιουργούν εύκολα εφαρμογές, άμεσα αξιοποιήσιμες από τους ίδιους αλλά και από τον περίγυρό τους, για συσκευές που αγαπάνε και χρησιμοποιούν σε καθημερινή βάση.

Όπως χαρακτηριστικά αναφέρει η Google, «...για πολλούς ανθρώπους, η κινητή τους συσκευή και η πρόσβαση που επιτυγχάνεται στο Διαδίκτυο μέσω αυτής, έχει διάφορους περιορισμούς. Το App Inventor για Android δίνει σε όλους, ανεξαρτήτως προγραμματιστικών ικανοτήτων, την ευκαιρία να ελέγχουν και να μετασχηματίζουν την εμπειρία της επικοινωνίας τους. Παρατηρούμε μάλιστα πως οι άνθρωποι αισθάνονται υπερήφανοι με το να γίνονται δημιουργοί της κινητής τεχνολογίας και όχι απλοί χρήστες της».

Με το App Inventor, οι χρήστες μπορούν να δημιουργήσουν πρακτικά όποια εφαρμογή θέλουν: από ένα απλό παιχνίδι έως εφαρμογές που θα χρησιμοποιούν τους αισθητήρες της συσκευής (εγγύτητας, προσανατολισμού κ.ά.). Επιπλέον, είναι δυνατή και η δημιουργία εφαρμογών εκμάθησης, όπως η δημιουργία ενός κουίζ ερωτήσεων ή η εκμετάλλευση της τεχνολογίας text-to-speech για εκφώνηση διαφόρων εντολών από τη συσκευή.

Δίνεται έτσι η ευκαιρία στους νέους προγραμματιστές ταυτόχρονα με την διαδικασία ανάπτυξης των εφαρμογών τους να εξοικειωθούν και με έννοιες του αντικειμενοστραφούς και οδηγούμενου από γεγονότα προγραμματισμού. Έννοιες όπως της ακολουθίας, των δομών επανάληψης και ελέγχου ροής, της διαδικασίας και συνάρτησης, παραμέτρου, μεταβλητής, αριθμητικών, συγκριτικών και λογικών τελεστών, και λιστών. Είναι επίσης μερικές από αυτές που θα συναντήσουν με ευδιάκριτο τρόπο οι νέοι στον προγραμματισμό χρήστες του App Inventor σε ένα περιβάλλον ίσως με μεγαλύτερο βαθμό κινητοποίησης από ότι συνήθως.

1.5.2 Εκπαιδευτική Φιλοσοφία του tutorial

Η μέθοδος της μελέτης των περιπτώσεων (case studies) είναι μία από τις αποτελεσματικότερες και πλέον αποδεκτές παγκοσμίως μεθόδους εκπαίδευσης, που καλύπτει το κενό που υπάρχει πάντοτε ανάμεσα στην πράξη και στη θεωρία. Ουσιαστικά, συμπληρώνει και ολοκληρώνει τη θεωρία, με συγκεκριμένα παραδείγματα που παρουσιάστηκαν στον επαγγελματικό χώρο ή που αντιμετώπισαν επιχειρήσεις και οργανισμοί κατά τη λειτουργία τους.

Η μέθοδος της μελέτης των περιπτώσεων πρωτοεμφανίστηκε στο Harvard Law School από τον Christopher Columbus Langdell και σήμερα είναι η βασικότερη μέθοδος διδασκαλίας στις Πανεπιστημιακές Νομικές Σχολές των ΗΠΑ. Βασίζεται στο ότι οι φοιτητές διδάσκονται το αμερικανικό δίκαιο με πραγματικά δεδομένα, μελετώντας πρωτογενές υλικό δικαστικών αποφάσεων, αντί να μελετούν απλά και μόνο θεωρητικές έννοιες και αφηρημένες περιλήψεις νόμων και κανόνων. Η μέθοδος της μελέτης των περιπτώσεων αποσκοπεί στο να αναπτύξει στους εκπαιδευόμενους την ικανότητα της κριτικής σκέψης, με σκοπό να επιλύουν προβλήματα υψηλού βαθμού πολυπλοκότητας, αφού προηγουμένως αναλύσουν και αξιολογήσουν όλα τα

δεδομένα. Τους βοηθάει να αποκτήσουν τις πρώτες εμπειρίες τους και τους προετοιμάζει στο να λαμβάνουν αποφάσεις, όταν βρεθούν στον επαγγελματικό χώρο μετά το πέρας των σπουδών τους.

Με την επανάληψη της μεθόδου αποκτούν σταδιακά την ικανότητα για ορθολογική λήψη αποφάσεων. Οι περιπτώσεις για μελέτη παρουσιάζουν συγκεκριμένες καταστάσεις, πραγματικές ή φανταστικές, που συνήθως έχουν δημιουργηθεί ειδικά για εκπαιδευτικές ανάγκες. Παρουσιάζουν επίσης και γεγονότα που έχουν συμβεί στον επαγγελματικό χώρο, σε επιχειρήσεις και οργανισμούς, και αναφέρονται σε συγκεκριμένα προβλήματα που εντοπίστηκαν κατά τη λειτουργία τους. Συνήθως οι περιπτώσεις αφορούν σε καθημερινά προβλήματα διαχείρισης, διοίκησης, ανάπτυξης, ανταγωνισμού, κοινωνικής παθογένειας, ανθρώπινου δυναμικού, κ.λπ. Οι περιπτώσεις για μελέτη κατηγοριοποιούνται σε απλές, σε σύνθετες και σε υψηλού βαθμού πολυπλοκότητας, ανάλογα με τη δυσκολία που παρουσιάζουν, τα προβλήματα που εμπεριέχουν, τη θεωρητική γνώση που απαιτείται για την ανάλυσή τους και τη συνθετότητα των παραγόντων που πρέπει να ληφθούν υπόψη για να δοθούν οι απαντήσεις στα ερωτήματα που τίθενται.

Με τη μέθοδο της μελέτης των περιπτώσεων βελτιώνονται οι δυνατότητες αλλά και οι ικανότητες αντίληψης, διάγνωσης, ανάλυσης και σύνθεσης προβλημάτων, επικοινωνίας, επιλογής μεταξύ εναλλακτικών λύσεων και λήψης αποφάσεων των εκπαιδευομένων. Οι φοιτητές αποκτούν εμπειρίες. Μελετούν αναζητώντας ενδείξεις και αποδείξεις. Αυτοαξιολογούνται και αυτοβελτιώνονται. Αντιλαμβάνονται ότι οι απόψεις και οι θέσεις τους πρέπει να αιτιολογούνται και να τεκμηριώνονται. Μαθαίνουν να ακούνε και να σέβονται την άποψη και τη γνώμη των άλλων, να τη συγκρίνουν με τη δική τους, να δέχονται τα ουσιαστικά και να απορρίπτουν τα περιττά, να αντιλαμβάνονται την πραγματική διαφορά μεταξύ υποκειμενικών και αντικειμενικών απόψεων και τελικά να συναποφασίζουν ορθολογικά για τη βέλτιστη λύση. Αυτά τους βοηθούν στο να αποκτήσουν κριτική σκέψη. Επιπρόσθετα, δίνεται η ευκαιρία στους εκπαιδευόμενους, να μελετήσουν πρακτικά τις άτυπες οργανώσεις, που δεν διδάσκονται αναλυτικά όσο οι θεωρητικές τυπικές. Μαθαίνουν να διερευνούν και να διαπιστώνουν τις πραγματικές αιτίες που δημιούργησαν το πρόβλημα και να μη βλέπουν μόνο την αφορμή και την έκβασή του. Εξοικειώνονται με την πολυπλοκότητα των προβλημάτων της καθημερινότητας και της πραγματικότητας του επαγγελματικού χώρου.

Όταν η μελέτη περίπτωσης πρόκειται να χρησιμοποιηθεί ως διδακτικό εργαλείο μπορεί και να ξεφεύγει από την αντικειμενική, πλήρη και ακριβή ανάλυση της πραγματικότητας που είναι το ζητούμενο μιας ερευνητικής μελέτης περίπτωσης.

Για την ανάλυση, ανάπτυξη και επίλυση των περιπτώσεων απαιτείται βασική γνώση της θεωρίας που αντιστοιχεί στο συγκεκριμένο πρόβλημα. Η γνώση αυτή συνήθως προέρχεται από τη διδασκαλία ενός συνόλου μαθημάτων που έχουν σχέση με την ενότητα διδασκαλίας. Ο βαθμός δυσκολίας κάθε περίπτωσης, (απλή, σύνθετη ή υψηλού βαθμού πολυπλοκότητας) προσδιορίζει συνήθως και το επίπεδο της γνώσης που απαιτείται για την επίλυσή της. Η μέθοδος της μελέτης των περιπτώσεων χρησιμοποιείται κυρίως για εκπαιδευτικούς σκοπούς. Χρησιμοποιείται όμως και για λόγους έρευνας. Σε χώρες του εξωτερικού, όπου υπάρχει καλύτερη διασύνδεση των επιχειρήσεων και οργανισμών με τα εκπαιδευτικά ιδρύματα, συχνά, όταν εταιρείες θέλουν να διαπιστώσουν πώς θα αντιμετώπιζαν ένα συγκεκριμένο θέμα οι καταναλωτές-πελάτες τους ή ακόμη και το προσωπικό τους, περιγράφουν το πρόβλημα σε μορφή περίπτωσης και το αναθέτουν σε συγκεκριμένα τμήματα εκπαιδευτικών ιδρυμάτων, όπου δημιουργούνται ομάδες φοιτητών που το μελετούν και οι εταιρείες συλλέγουν, επεξεργάζονται και αξιοποιούν τις απαντήσεις τους. Τα πλεονεκτήματα από την εφαρμογή της μεθόδου μελέτης των περιπτώσεων είναι πολλά: Το βασικότερο είναι η κάλυψη του χάσματος που υπάρχει ανάμεσα στη θεωρία και στην πράξη. Εφόσον η μέθοδος χρησιμοποιηθεί σωστά, διαπιστώνονται τα κενά που έχει αφήσει η διδασκαλία συγκεκριμένων μαθημάτων ή συγκεκριμένων κεφαλαίων από κάποια μαθήματα. Διαπιστώνεται, δηλαδή, αν έχει εμπεδωθεί σωστά και καλά όλη η θεωρία.

1.6 Μάθηση μέσω Μελέτη Περίπτωσης

Ο λόγος της διάδοσής των μελετών περιπτώσεων είναι απλός: πέρα από τη συνήθη διδασκαλία βασικών επιστημονικών αρχών και θεωριών, πολλές φορές χρειάζονται τρόποι εκπαίδευσης των σπουδαστών σε σενάρια που αναδεικνύουν την πρακτική αφομοίωση της γνώσης. Οι μελέτες περίπτωσης περιγράφουν καταστάσεις στις οποίες τα εμπλεκόμενα πρόσωπα πρέπει να αντιμετωπίσουν κάποιο πολύπλοκο πρόβλημα ή να δραστηριοποιηθούν για την επίλυση κάποιου πολύπλευρου διλήμματος. Αποτελούν λοιπόν εργαλείο εκπαίδευσης στην επίλυση προβλημάτων,

όπου οι σπουδαστές τοποθετούνται 'μέσα' στο πρόβλημα και καλούνται να συνθέσουν υπάρχουσες γνώσεις και δεξιότητες για να προτείνουν πιθανές λύσεις, με ιδιαίτερη έμφαση στη λήψη αποφάσεων.

Με βάση τα παραπάνω, ο σκοπός μιας μελέτης περίπτωσης είναι διπλός:

- Να βοηθήσει τους εκπαιδευτές να εμπλουτίσουν τα εκπαιδευτικά εργαλεία τους με ένα νέο τρόπο διάδρασης με τους εκπαιδευόμενους. Η πρόκληση για τους εκπαιδευτές είναι αφ' ενός να καθορίσουν την πολυπλοκότητα και ευρύτητα της μελέτης και αφ' ετέρου να κρατήσουν το ενδιαφέρον των εκπαιδευόμενων και να τους κατευθύνουν ανάλογα.
- Να βοηθήσει τους εκπαιδευόμενους να αφομοιώσουν ευκολότερα γνωστικά αντικείμενα, συνδυάζοντάς τα με την ανάπτυξη: α) κριτικής σκέψης, και β) δεξιοτήτων δημιουργικής ανάπτυξης λύσεων.

Συνολικά, μία μελέτη περίπτωσης μπορεί να λειτουργήσει ως πρακτικό εργαστήριο εφαρμογής γνώσεων σε πραγματικά προβλήματα. Βοηθά έτσι τον εκπαιδευόμενο να αποκτήσει διάφορες δεξιότητες:

- Αναγνώριση και ανάλυση προβλημάτων.
- Επεξεργασία και σύνθεση δεδομένων.
- Δημιουργία υποθέσεων εργασίας.
- Ανοχή σε μη ξεκάθαρες καταστάσεις και αντοχή στην πολυπλοκότητά τους.
- Κριτική σκέψη και σύνθεση απόψεων.
- Επιχειρηματολογία και λήψη αποφάσεων.
- Επικοινωνία απόψεων και ιδεών.

Στην συνέχεια, παρατίθενται τα βήματα μιας μελέτης περίπτωσης, καθώς και ανάλυση των βημάτων που ακολουθήθηκαν για την εκπόνηση του tutorial.

1.6.1 Βήματα μιας μελέτης περίπτωσης

Εν γένει, υπάρχουν τέσσερα στάδια συγγραφής μιας μελέτης περίπτωσης:

1. σχεδιασμός,
2. οργάνωση υλικού,
3. υλοποίηση/συγγραφή, και

4. αξιολόγηση/επικαιροποίηση.

1. Σχεδιασμός

Στο στάδιο αυτό καθορίζονται οι βασικές παράμετροι συγγραφής της μελέτης περίπτωσης, όπως: εκπαιδευτικοί στόχοι, προφίλ εκπαιδευόμενων στους οποίους απευθύνεται, και εκπαιδευτικό υλικό που πρέπει να συλλεχθεί.

Οι **εκπαιδευτικοί στόχοι** ορίζουν τι αναμένεται να γνωρίζουν οι εκπαιδευόμενοι και τι να μπορούν να κάνουν, να νιώθουν ή να αξιολογούν μετά το πέρας της διδασκαλίας μιας μελέτης περίπτωσης. Οι στόχοι αυτοί τελικά καθορίζουν το περιεχόμενο της μελέτης, καθώς και τις έννοιες, τους κανόνες και τις αρχές που θα περιλαμβάνει.

Η επιτυχής υλοποίηση των στόχων βασίζεται άμεσα στην κατανόηση του **προφίλ των εκπαιδευόμενων** στους οποίους απευθύνεται η μελέτη. Ο προφανής λόγος είναι ότι αυτό βοηθά τον συγγραφέα να ορίσει το εκπαιδευτικό σημείο εκκίνησης (γνώσεις, δεξιότητες) και την απόσταση που θα πρέπει να καλύψει ο εκάστοτε εκπαιδευόμενος για την επίτευξη των εκπαιδευτικών στόχων. Η απόσταση αυτή θα καθορίσει τι είδους πληροφορίες και εκπαιδευτικό υλικό θα πρέπει να συμπεριλάβει ο συγγραφέας, καθώς και το επίπεδο της γλώσσας που πρέπει να χρησιμοποιηθεί και των λεπτομερειών που πρέπει να συμπεριληφθούν.

Επικεντρώνοντας στο **εκπαιδευτικό υλικό** που πρέπει να συλλεχθεί, είναι σημαντικό να οριστούν όσο το δυνατόν νωρίτερα οι πηγές από τις οποίες θα αναζητηθεί και θα αντληθεί. Οι πηγές αυτές μπορεί να είναι προσωπικές εμπειρίες του συγγραφέα ή πηγές δευτερογενούς έρευνας, όπως άρθρα και γενικές δημοσιεύσεις. Αρκετές φορές, μία μελέτη περίπτωσης βασίζεται σε πραγματικούς οργανισμούς και πραγματικά πρόσωπα, οπότε το σχετικό υλικό μπορεί να συλλεχθεί μέσω συνεντεύξεων με τα ίδια τα πρόσωπα. Αυτό αυξάνει τον ρεαλισμό της μελέτης και φέρνει πιο κοντά τους εκπαιδευόμενους στις πραγματικές παραμέτρους του προβλήματος που θα κληθούν να επιλύσουν.

Στην πράξη, ο συγγραφέας μιας μελέτης περίπτωσης την σχεδιάζει βασιζόμενος σε εργαλεία τύπου checklist τα οποία περιγράφουν τα στοιχεία σχεδιασμού που προαναφέραμε. Οι πίνακες που ακολουθούν είναι παραδείγματα τέτοιων εργαλείων:

Πίνακας 1- Εκπαιδευτικός Στόχος Μελέτης Περίπτωσης

1. Ποιος είναι ο εκπαιδευτικός στόχος της μελέτης περίπτωσης;	
Έρευνα	Στόχος είναι να διερευνηθεί η σχέση αιτίας-αιτιατού σε μία κατάσταση αυξημένης πολυπλοκότητας.
Συζήτηση	Στόχος είναι να χρησιμοποιηθεί η μελέτη ως βάση συζήτησης για την εύρεση ή αξιολόγηση λύσεων ενός προβλήματος.
Παράδειγμα κατάστασης	Στόχος είναι να δοθεί ένα αναλυτικό παράδειγμα μιας κατάστασης/φαινομένου, το οποίο συμπληρώνει κι εμπλουτίζει το θεωρητικό μέρος ενός μαθήματος.
Παράδειγμα διαδικασίας	Στόχος είναι να δοθεί ένα αναλυτικό παράδειγμα μιας διαδικασίας (με ποιοτικά ή/και ποσοτικά χαρακτηριστικά).
Άλλος στόχος	(παρακαλώ περιγράψτε)

Ο εκπαιδευτικός στόχος της συγκεκριμένης μελέτης περίπτωσης είναι να παρουσιαστεί το εργαλείο app inventor τόσο σε θεωρητικό πλαίσιο, όσο και σε επίπεδο εφαρμογής παρέχοντας ένα αναλυτικό παράδειγμα εφαρμογής σχεδιασμένης με το εργαλείο αυτό, έτσι ώστε να εμπλουτιστεί και να συμπληρωθεί το θεωρητικό υπόβαθρο του tutorial.

Πίνακας 2- Σχέση συγκεκριμένης μελέτης με άλλες

2. Ποια είναι η σχέση της μελέτης αυτής με άλλες μελέτες περιπτώσεων;	
Ανεξάρτητη μελέτη	Η μελέτη αυτή μπορεί να παρουσιαστεί, αναλυθεί και συζητηθεί χωρίς αναφορά σε άλλες μελέτες περίπτωσης. Διαπραγματεύεται αυτοτελώς μια κατάσταση/ φαινόμενο κι έχει νόημα ως

	ανεξάρτητο αντικείμενο.
Συγκριτική μελέτη	Η μελέτη αυτή θα χρησιμοποιηθεί μαζί με άλλες μελέτες περίπτωσης, και θα λειτουργήσει στην βάση συγκριτικής ανάλυσης.
Μέρος μιας σειράς μελετών	Η μελέτη αυτή είναι μέρος μιας σειράς μελετών που συνολικά πραγματεύονται και καλύπτουν μία κατάσταση/φαινόμενο.

Η συγκεκριμένη μελέτη περίπτωσης δεν έχει σχέση με άλλες μελέτες περίπτωσης, αποτελεί ανεξάρτητη μελέτη. Μπορεί να παρουσιαστεί, να αναλυθεί και να συζητηθεί χωρίς αναφορά σε άλλες μελέτες. Διαπραγματεύεται αυτοτελώς μια κατάσταση κι έχει νόημα ως ανεξάρτητο κείμενο.

Πίνακας 3- Παρουσίαση εκπαιδευτικού υλικού που συλλέχθηκε

3. Πώς θα παρουσιαστεί το εκπαιδευτικό υλικό που θα συλλεχθεί;	
Ως πραγματική κατάσταση	Η μελέτη περίπτωσης θα παρουσιάσει τα πραγματικά γεγονότα της κατάστασης/φαινομένου που πραγματεύεται. Τα στοιχεία συνήθως συλλέγονται απ' ευθείας από τον οργανισμό στον οποίον θα αναφέρεται η μελέτη, και απαιτούν σχετική άδεια δημοσιοποίησης.
Ως πραγματική κατάσταση, ελαφρά τροποποιημένη	Η μελέτη θα αναπτυχθεί ως μελέτη 'πραγματικής κατάστασης'. Στη συνέχεια, θα γίνουν περιορισμένες αλλαγές σε στοιχεία της μελέτης (όνομα οργανισμού, πρόσωπα, ποσοτικά στοιχεία).
Ως πραγματική κατάσταση, εκτενώς	Η μελέτη θα αναπτυχθεί ως μελέτη 'πραγματικής κατάστασης'. Στη

τροποποιημένη	συνέχεια, θα γίνουν εκτεταμένες αλλαγές σε σημαντικά στοιχεία της μελέτης (π.χ. είδος επιχείρησης).
Ως πραγματική κατάσταση, με βάση δημοσιευμένα γεγονότα	Η μελέτη περίπτωσης θα παρουσιάσει τα πραγματικά γεγονότα της κατάστασης/φαινομένου που πραγματεύεται. Τα στοιχεία συλλέγονται από δημόσιες πηγές (π.χ. Διαδίκτυο, εφημερίδες) κι επομένως δεν απαιτούν ειδική άδεια για τη δημοσιοποίησή τους.
Ως φανταστικό σενάριο	Η μελέτη περίπτωσης βασίζεται σε φανταστικά γεγονότα, πρόσωπα και καταστάσεις. Τα στοιχεία που παρουσιάζονται βασίζονται στις εμπειρίες του συγγραφέα, στη δημιουργική του φαντασία και σε δημόσιες πηγές (π.χ. διαδίκτυο, εφημερίδες).

Η συγκεκριμένη μελέτη, παρουσιάζει πραγματικά γεγονότα της κατάστασης που πραγματεύεται. Τα στοιχεία συλλέχθηκαν από δημόσιες πηγές και επομένως δεν απαιτούν ειδική άδεια για την δημοσιοποίησή τους.

Πίνακας 4- Ερωτήσεις σε περίπτωση που η μελέτη είναι βασισμένη σε στοιχεία που συλλέχθηκαν απευθείας από τον οργανισμό

3.1 Αν η μελέτη θα βασιστεί σε στοιχεία που θα συλλεχθούν απ' ευθείας από τον οργανισμό στον οποίο αναφέρεται η μελέτη, τότε ο συγγραφέας θα πρέπει να προετοιμάσει ερωτήσεις για μία ή περισσότερες συνεντεύξεις που θα καλύψουν τα εξής θέματα:	
	A) Ποιος είναι ο οικονομικός κλάδος/οργανισμός/ λειτουργική μονάδα στην οποία θα αναφέρεται η

Πλαίσιο της μελέτης	μελέτη B) Ποιο είναι το χρονικό/ιστορικά πλαίσιο στο οποίο διαδραματίζονται τα γεγονότα (π.χ. η οικονομική κρίση του 2009).
Πρόσωπα	Ποιο είναι το προφίλ του προσώπου που θα είναι ο πρωταγωνιστής της μελέτης περίπτωσης.
Καταστάσεις	Ποια προβλήματα/θέματα είναι ενδιαφέροντα/δύσκολα/ άλυτα στον υπό εξέταση οργανισμό.
Πιθανά θέματα/ προβλήματα	Για κάθε ένα θέμα/πρόβλημα: Τι κάνει το πρόβλημα ενδιαφέρον/ξεχωριστό/ πολύπλοκο Τι είδους αποφάσεις πρέπει να ληφθούν Ποιες προηγούμενες αποφάσεις οδήγησαν/ βοήθησαν στη δημιουργία του προβλήματος Πώς μπορεί ο συγγραφέας να βοηθήσει, μέσω της μελέτης περίπτωσης Ποιοι άλλοι σχετίζονται με το πρόβλημα; Μπορούν να μιλήσουν γι αυτό το θέμα;
Δημοσιοποίηση μελέτης	Γενικά: Όταν θα ολοκληρωθεί η μελέτη περίπτωσης, θα ήθελε ο συνεντευξιαζόμενος να προστεθεί ως δεύτερος συγγραφέας; Υπάρχουν σημεία της μελέτης περίπτωσης που θα ήθελε ο συνεντευξιαζόμενος να τροποποιήσει; Ποιος θα δώσει την τελική έγκριση για δημοσιοποίηση;
Άλλο	(παρακαλώ περιγράψτε)

Τα στοιχεία συλλέχθηκαν από ένα ευρύ φάσμα κοινού εντός και εκτός οργανισμού.

Πίνακας 5- Χρήση Μελέτης Περίπτωσης

4. Πώς θα χρησιμοποιηθεί η μελέτη περίπτωσης;	
Εντός του Πανεπιστημίου	Η μελέτη θα χρησιμοποιηθεί ως μέσο διδασκαλίας ή εξάσκησης των φοιτητών.
Εκτός του Πανεπιστημίου, σε περιορισμένη έκταση	Η μελέτη θα χρησιμοποιηθεί εκτός του Πανεπιστημίου, π.χ. ως μέσο ενημέρωσης ή ως working paper.
Εκτός του Πανεπιστημίου, με μορφή δημοσίευσης	Η μελέτη θα χρησιμοποιηθεί σε επιστημονική έκδοση και θα λαμβάνει υπ' όψη της σχετικά κριτήρια ανάπτυξης ως ερευνητικό πόνημα.

Στην παρούσα φάση μας η μελέτη περίπτωσης θα χρησιμοποιηθεί στα πλαίσια του Πανεπιστημίου ως βάση ανάπτυξης ενός εργαλείου στα πλαίσια μιας διπλωματικής εργασίας.

2. Οργάνωση Υλικού

Στο στάδιο αυτό οργανώνονται οι ιδέες και το υλικό που έχει συλλεχθεί στην προηγούμενη φάση και καθορίζονται ο τύπος της μελέτης και το σενάριο που θα αναδείξει τους εκπαιδευτικούς στόχους.

Ο **τύπος της μελέτης** σχετίζεται άμεσα με την επιλογή της θεματολογίας της μελέτης και με το τελικό αποτέλεσμα/εκπαιδευτικό στόχο που θέλει να πετύχει ο συγγραφέας. Για παράδειγμα, ο τύπος μιας μελέτης περίπτωσης ορίζεται ως 'λήψη απόφασης', αν το θέμα της μελέτης επικεντρώνεται σε μία απόφαση που πρέπει να ληφθεί ή στα γεγονότα που οδήγησαν σε μια παλαιότερη απόφαση. Αυτού του τύπου οι μελέτες είναι κατάλληλες όταν ο εκπαιδευτικός στόχος είναι 'Συζήτηση'.

Το **σενάριο της μελέτης** παρουσιάζει τη φύση και την πολυπλοκότητα της κατάστασης/φαινομένου που διαπραγματεύεται η μελέτη περίπτωσης. Η επιτυχία του έγκειται στην ανάδειξη πολλαπλών οπτικών γωνιών της κατάστασης, στην καλλιέργεια κριτικής σκέψης των εκπαιδευόμενων και στη σύνθεση από μεριάς τους γνώσεων, δεξιοτήτων και ικανοτήτων. Γι' αυτό, το σενάριο μιας μελέτης εμπεριέχει

συχνά αβεβαιότητες ή αλληλοσυγκρουόμενες απόψεις/εκτιμήσεις/αποφάσεις, βάσει των οποίων ο εκπαιδευόμενος πρέπει να τοποθετηθεί ή να χρησιμοποιήσει σαν βάση επίλυσης ενός προβλήματος. Για παράδειγμα, μια μελέτη περίπτωσης που αναφέρεται σε νέες διδακτικές πρακτικές σε σχολεία μπορεί να καταγράφει αντιθέσεις και αντιπαλότητες για τα θετικά και αρνητικά αποτελέσματα που προκαλεί μια νέα εκπαιδευτική πολιτική.

Γενικά, η οργάνωση και παρουσίαση μιας μελέτης γίνεται με χρονολογική σειρά, όπως συμβαίνουν τα γεγονότα που σχετίζονται με το θέμα. Μερικές φορές, μπορεί να είναι πιο αποτελεσματική η παρουσίαση με αντίστροφη χρονολογική σειρά. Δηλαδή, να περιγράφεται στην αρχή της μελέτης η κατάληξή της, και στη συνέχεια να αναλύονται τα γεγονότα ως αναλαμπές του παρελθόντος. Σε κάθε περίπτωση, οι βασικοί στόχοι οργάνωσης του υλικού της μελέτης είναι: α) να υπάρχουν ξεκάθαρες μεταβάσεις μεταξύ των γεγονότων, β) να διατηρείται το ενδιαφέρον αναγνώστων/εκπαιδευόμενων, και γ) να κατευθύνονται οι αναγνώστες στους στόχους που έχει θέσει ο συγγραφέας της μελέτης.

Στην πράξη, ο συγγραφέας μιας μελέτης περίπτωσης οργανώνει το υλικό που έχει συλλέξει βασιζόμενος σε εργαλεία τύπου checklist, όπως αυτά που ακολουθούν:

Πίνακας 6- Σημεία στα οποία δίνει έμφαση η μελέτη περίπτωσης

1. Σε τι δίνει έμφαση η μελέτη περίπτωσης;	
Λήψη αποφάσεων	Έμφαση δίνεται στις αποφάσεις που πρέπει να ληφθούν κατά τη διδασκαλία της μελέτης ή στις καταστάσεις που οδήγησαν σε μια ειλημμένη απόφαση. Αυτού του τύπου οι μελέτες είναι καταλληλότερες όταν ο εκπαιδευτικός στόχος είναι 'Συζήτηση'.
Απόκτηση νέων γνώσεων	Έμφαση δίνεται στην καλύτερη κατανόηση: α) του πλαισίου που περιγράφει η μελέτη, β) των γνώσεων και δεξιοτήτων που απαιτούνται κατά τη διαδικασία λήψης μιας συγκεκριμένης απόφασης. Αυτού του

	τύπου οι μελέτες είναι καταλληλότερες όταν ο εκπαιδευτικός στόχος είναι «Έρευνα» ή «Παράδειγμα διαδικασίας».
Παρουσίαση παραδειγμάτων	Έμφαση δίνεται στην παραδειγματική διαχείριση ενός προβλήματος ή μιας κατάστασης. Αυτού του τύπου οι μελέτες είναι καταλληλότερες όταν ο εκπαιδευτικός στόχος είναι «Παράδειγμα κατάστασης».
Συνδυασμός των παραπάνω	(παρακαλώ περιγράψτε)

Στην συγκεκριμένη μελέτη περίπτωσης, δίνεται έμφαση τόσο στην απόκτηση νέων γνώσεων και δεξιοτήτων, όσο και στην παρουσίαση ενός απτού παραδείγματος για την καλύτερη αφομοίωση των πληροφοριών από τον εκπαιδευόμενο.

Πίνακας 7- Η δομή του θέματος της μελέτης

2. Ποια θα είναι η δομή του θέματος της μελέτης;	
Ένα σύνολο	Το κείμενο της μελέτης θα διαπραγματεύεται συνολικά το θέμα που έχει επιλέξει ο συγγραφέας.
Σπονδυλωτή	Το κείμενο της μελέτης μπορεί να χωριστεί σε επεισόδια, τα οποία θα παρουσιάζονται στους εκπαιδευόμενους ανάλογα με την πορεία της διδασκαλίας της μελέτης. Αυτή η δομή επιτρέπει: α) την κατεύθυνση της διδασκαλίας ανάλογα με την πορεία της συζήτησης, β) την βαθμιαία ανάλυση του προβλήματος και των επιμέρους υποθέσεων/συνθηκών/αποτελεσμάτων, πριν τη μετάβαση στο επόμενο επεισόδιο.

Η δομή του θέματος της συγκεκριμένης μελέτης είναι σπονδυλωτή, καθώς το κείμενο της μελέτης είναι χωρισμένο σε κεφάλαια και εισάγει κλιμακωτά και σταδιακά τον εκπαιδευόμενο στο περιεχόμενο του.

Πίνακας 8- Αναμενόμενα μαθησιακά αποτελέσματα

3. Ποια είναι τα αναμενόμενα μαθησιακά αποτελέσματα;	
Απόκτηση γνώσεων	Γνωριμία με (νέα) γνωστικά αντικείμενα και ικανότητα εφαρμογής τους σε πραγματικά σενάρια. Τα δεδομένα της μελέτης είναι από μόνα τους σημαντική πηγή γνώσης για τους εκπαιδευόμενους, δρώντας ως συμπληρωματικό υλικό των κλασικών διαλέξεων.
Ανάπτυξη πρωτοβουλιών στον σχεδιασμό λύσεων	Εμβάθυνση στη διερεύνηση διαφόρων επιλογών και κατευθύνσεων που οδηγούν: α) στην δημιουργία συμπαγών λύσεων, β) στην παρουσίαση νέων ιδεών/κατευθύνσεων που ξεπερνούν το πλαίσιο της συγκεκριμένης μελέτης περίπτωσης.
Αναλυτική σκέψη	Έμφαση στην λεπτομερή ανάλυση επιλογών (π.χ. καταγραφή δυνατών σημείων και αδυναμιών), καθώς και στη δημιουργία και διερεύνηση υποθέσεων εργασίας που μπορεί να βασίζονται ή να ξεπερνούν τα δεδομένα της συγκεκριμένης μελέτης περίπτωσης.
Άλλο	(παρακαλώ περιγράψτε)

Τα αναμενόμενα μαθησιακά αποτελέσματα της συγκεκριμένης μελέτης περίπτωσης, είναι η ανάπτυξη πρωτοβουλιών στον σχεδιασμό λύσεων καθώς και η αναλυτική σκέψη. Κάθε εκπαιδευόμενος με το πέρας της μελέτης θα είναι σε θέση να δημιουργεί συμπαγείς λύσεις, και να δημιουργεί και να διευρύνει υποθέσεις που ξεπερνούν το πλαίσιο της συγκεκριμένης μελέτης περίπτωσης.

3. Υλοποίηση/ Συγγραφή

Στο στάδιο αυτό, ο σχεδιασμός και η οργάνωση του υλικού που συλλέχθηκε κατευθύνουν τη δημιουργία του κειμένου της μελέτης περίπτωσης. Προφανώς, θα υπάρξουν αρκετές παραλλαγές του κειμένου πριν την τελική του μορφή, καθώς θα πρέπει να συγκεραστούν εκπαιδευτικοί στόχοι, αναμενόμενα μαθησιακά αποτελέσματα, κλπ. Όπως για κάθε συγγραφέα, τα σημαντικότερα θέματα και για το συγγραφέα της μελέτης περίπτωσης είναι:

- α) να αντιλαμβάνεται τις ανάγκες και το επίπεδο του αναγνωστικού κοινού/εκπαιδευόμενων,
- β) να τους παρέχει τη σωστή ποσότητα και ποιότητα πληροφοριών για να καταλάβουν τα θέματα που θίγει, και
- γ) να τους κάνει να ταυτιστούν με τα θέματα, τα πρόσωπα και τις καταστάσεις που περιγράφει.

Συνήθης τακτική είναι ο συγγραφέας της μελέτης να δημιουργεί το πρόγραμμα της μελέτης καθώς και μερικές εισαγωγικές παραγράφους, με βάση τους εκπαιδευτικούς στόχους της μελέτης:

- Έρευνα: καθορισμός της σημασίας του ερευνητικού θέματος.
- Συζήτηση: καθορισμός του πλαισίου του προς εξέταση προβλήματος, καθώς και τυχόν εναλλακτικών λύσεων.
- Παράδειγμα κατάστασης/παράδειγμα διαδικασίας: περιληπτική περιγραφή της κατάστασης/φαινομένου/διαδικασίας.

4. Αξιολόγηση/ Επικαιροποίηση

Η ολοκλήρωση της συγγραφής της μελέτης περίπτωσης δεν σημαίνει αυτομάτως ότι είναι έτοιμη για άμεση χρήση, ειδικά αν η χρονική διάρκεια ολοκλήρωσής της είναι μεγάλη. Μία μελέτη η οποία διαπραγματεύεται ξεπερασμένα γεγονότα/ιδέες/προσεγγίσεις, βασίζεται σε ξεπερασμένα δεδομένα ή απευθύνεται σε εκπαιδευόμενους το προφίλ των οποίων έχει αλλάξει, δεν έχει πολλά περιθώρια επιτυχίας και χρήζει επικαιροποίησης.

Γι' αυτό, η συγγραφή μιας μελέτης περίπτωσης ακολουθείται από αξιολόγηση, είτε από τον ίδιο το συγγραφέα είτε από κάποιον στον οποίο ο συγγραφέας αναθέτει την αξιολόγηση. Σκοπός της διαδικασίας αυτής είναι να επιβεβαιώσει ότι οι προηγούμενες τρεις φάσεις ανάπτυξης μιας μελέτης αποτυπώνονται στο τελικό κείμενο, στο βαθμό που θέλει ο συγγραφέας. Σχετικές ερωτήσεις που μπορούν να κατευθύνουν αυτήν την διαδικασία είναι ενδεικτικά οι εξής:

- Η μελέτη περίπτωσης υπηρετεί τους εκπαιδευτικούς στόχους που έθεσε ο συγγραφέας;
- Είναι ικανοποιητικός ο βαθμός πολυπλοκότητάς της;
- Είναι ικανοποιητικός ο βαθμός ρεαλισμού της;
- Το κείμενο έλκει το ενδιαφέρον του αναγνώστη, δηλ. έχει ροή σεναρίου, περιγράφεται το ευρύτερο πλαίσιο του προβλήματος/κατάστασης/φαινομένου, αποτυπώνονται σωστά τα πρόσωπα που πρωταγωνιστούν, υπάρχει λογική μετάβαση μεταξύ των γεγονότων;
- Το περιεχόμενο της μελέτης είναι ακριβές και κατάλληλο για το θέμα που πραγματεύεται;
- Αν χρησιμοποιήθηκαν δημόσιες πηγές για τη συλλογή δεδομένων, είναι αξιόπιστες;

Είναι σημαντικό να επισημάνουμε ότι βασικοί αξιολογητές μιας μελέτης περίπτωσης είναι οι ίδιοι οι εκπαιδευόμενοι, η γνώμη των οποίων για την χρησιμότητα της μελέτης πρέπει να αναζητάται διαρκώς. Τα σχόλιά τους συχνά είναι εξαιρετικά πολύτιμα για την επικαιροποίηση μιας μελέτης και για την διατήρηση/βελτίωση της εκπαιδευτικής της αξίας.

Κεφάλαιο 2

2.1 Λογισμικά εφαρμογών για κινητές συσκευές

Η σημερινή εποχή χαρακτηρίζεται ως ο Χρυσός Αιώνας της ανάπτυξης λογισμικού. Το διαδίκτυο έχει δημιουργήσει εξαιρετικές ευκαιρίες ανάπτυξης λογισμικού, ενώ η άνθηση των δικτύων, των φορητών Η/Υ και των έξυπνων συσκευών, πέρα από τον συμβατικό προσωπικό Η/Υ, έχει ανοίξει πολύ μμεγάλους ορίζοντες εφαρμογής νέων προγραμματιστικών τεχνολογιών. Σημαντικό ρόλο στην ανάπτυξη του λογισμικού έπαιξε η κοινότητα του ανοιχτού λογισμικού η οποία έδωσε ελεύθερα προγράμματα υψηλής ποιότητας σε όλο το φάσμα των προγραμματιστικών αναγκών, από λειτουργικά συστήματα μέχρι προγράμματα για κινητά τηλέφωνα και PDA.

Παράλληλα, μεγάλες εικονικές κοινότητες μάθησης προσφέρουν δωρεάν υποστήριξη σε κάθε προγραμματιστική ανάγκη, από την πιο απλή έως την πιο επαγγελματική. Ο μόνος περιορισμός πλέον είναι η φαντασία και η διάθεση για μάθηση και κατάκτηση των σύγχρονων προγραμματιστικών εργαλείων. Οι εποχές όπου ο προγραμματισμός και η ανάπτυξη λογισμικού ήταν αποκλειστικά και μόνο στα χέρια των επαγγελματιών, οι οποίοι και διέθεταν το σύνολο σχεδόν της τεχνογνωσίας, έχουν πλέον περάσει. Σήμερα, ο καθένας μπορεί, όχι μόνο να ασχοληθεί, αλλά και να κατακτήσει σημαντικές τεχνολογίες τις οποίες μπορεί να χρησιμοποιήσει για προσωπική ή επαγγελματική χρήση. Η εξομάλυνση της καμπύλης μάθησης των νέων προγραμματιστικών περιβαλλόντων κατέβασε το κατώφλι εισόδου, δημιουργώντας μια νέα γενιά προγραμματιστών οι οποίοι δεν προέρχονται από τις παραδοσιακές κατηγορίες των ανθρώπων της πληροφορικής. Άνθρωποι προερχόμενοι από διάφορους επαγγελματικούς χώρους μεταφέρουν την αγάπη τους για τις νέες τεχνολογίες από το επίπεδο της χρήσης στο επίπεδο της ανάπτυξης.

Το κυρίαρχο μοντέλο της δεκαετίας του '90, ο μοναχικός χρήστης έτοιμου κλειστού λογισμικού, έχει μετασχηματιστεί στον ενεργό πολίτη-κάτοικο του διαδικτύου, ο οποίος διαχειρίζεται αλλά και αναπτύσσει την προσωπική του τεχνολογική υποδομή. Έτσι, ο απλός χρήστης της τεχνολογίας τείνει να μετεξελιχτεί σε άτομο οργανικά συνδεδεμένο με αυτήν, το οποίο θα μπορεί όχι μόνο να την κατανοεί και να την

αξιοποιεί, αλλά να την τροποποιεί και να την προσαρμόζει στις ανάγκες του.

2.2 Εργαλεία (γλώσσες) οπτικού προγραμματισμού

Στην επιστήμη υπολογιστών, οπτική γλώσσα προγραμματισμού (visual programming language, συντ. VPL) είναι μια γλώσσα προγραμματισμού που επιτρέπει στο χρήστη τη δημιουργία προγραμμάτων μέσα από το γραφικό χειρισμό προγραμματιστικών στοιχείων (αντί κειμένου). Μια VPL επιτρέπει τον προγραμματισμό με οπτικές εκφράσεις που είναι η τοποθέτηση στο χώρο κειμένου και γραφικών συμβόλων, είτε σαν σύνταξη, είτε σαν βοηθητική σύνταξη. Για παράδειγμα, υπάρχουν πολλές οπτικές γλώσσες, που είναι γνωστές σαν γλώσσες ροής δεδομένων (dataflow) ή διαγραμματικός προγραμματισμός (diagrammatic programming), που βασίζονται στην ιδέα "τετράγωνα και βέλη" ("boxes and arrows"), όπου τα τετράγωνα (ή άλλου τύπου αντικείμενα) της οθόνης θεωρούνται οντότητες που συνδέονται από βέλη, γραμμές ή ακμές, που αναπαριστούν σχέσεις μεταξύ τους.

Οι οπτικές γλώσσες προγραμματισμού μπορούν να κατηγοριοποιηθούν, ανάλογα με τον τύπο και την έκταση της χρήσης των οπτικών εκφράσεων, σε γλώσσες βασισμένες στα εικονίδια (icon-based), γλώσσες βασισμένες στις φόρμες (form-based) και σε γλώσσες διαγραμμάτων (diagram languages). Περιβάλλοντα οπτικού προγραμματισμού παρέχουν τότε στοιχεία εικονιδίων ή γραφικών που μπορούν να χρησιμοποιηθούν διαλογικά από το χρήστη σύμφωνα με κάποια χωρική γραμματική (spatial grammar), για την κατασκευή ενός προγράμματος.

Γίνεται προσπάθεια να ενσωματωθεί ο οπτικός προγραμματισμός στις γλώσσες προγραμματισμού ροής δεδομένων (dataflow programming), ώστε να είναι πιο εύκολη και άμεση η αποσφαλμάτωσή τους ή η αυτόματη παραγωγή προγραμμάτων και τεκμηρίωσης (visual paradigm). Οι γλώσσες ροής δεδομένων επιτρέπουν επίσης αυτόματο παραλληλισμό (automatic parallelization), που αποτελεί σημαντικό και ανοιχτό πρόβλημα στον προγραμματισμό ηλεκτρονικών υπολογιστών.

2.2.1 Python

Η Python είναι μια διερμηνευόμενη (interpreted), αλληλεπιδραστική (interactive) και προσανατολισμένη σε αντικείμενα (object-oriented), υψηλού επιπέδου γλώσσα

προγραμματισμού, που αναπτύχθηκε από τον Ολλανδό Guido van Rossum το 1990. Ο κύριος στόχος της είναι η αναγνωσιμότητα του κώδικά της. Η ευκολία χρήσης της και το συντακτικό της επιτρέπει στους προγραμματιστές να εκφράσουν έννοιες σε λιγότερες γραμμές κώδικα από ότι θα ήταν δυνατόν σε γλώσσες όπως η C++ ή η Java. Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα αρκετές συνηθισμένες εργασίες και για την ταχύτητα εκμάθησής της.



Ο πηγαίος κώδικας (source code) της Python διατίθεται δωρεάν και η διαχείρισή της γίνεται από τον μη κερδοσκοπικό οργανισμό Python Software Foundation. Η επίσημη ιστοσελίδα της γλώσσας είναι η <http://www.python.org>. Η ανάπτυξη της Python ξεκίνησε το 1990 στο CWI του Άμστερνταμ και συνεχίζεται στο CNRI του Reston.

Οι διερμηνευτές της Python είναι διαθέσιμοι για εγκατάσταση σε πολλά λειτουργικά συστήματα, επιτρέποντας στην Python την εκτέλεση κώδικα σε ευρεία γκάμα συστημάτων. Χρησιμοποιώντας εργαλεία τρίτων, όπως το Py2exe ή το Pyinstaller, ο κώδικας της Python μπορεί να πακεταριστεί σε αυτόνομα εκτελέσιμα προγράμματα για μερικά από τα πιο δημοφιλή λειτουργικά συστήματα, επιτρέποντας τη διανομή του βασισμένου σε Python λογισμικού για χρήση σε αυτά τα περιβάλλοντα χωρίς να απαιτείται εγκατάσταση του διερμηνευτή της Python.

Η Python μπορεί να επεκταθεί προσθέτοντάς της καινούργια modules που να είναι γραμμένα σε μια γλώσσα που μεταγλωττίζεται, όπως είναι η C ή η C++. Αυτά τα modules επέκτασης μπορούν να ορίσουν νέες συναρτήσεις (functions) και μεταβλητές (variables) καθώς επίσης και καινούργιους τύπους δεδομένων (object types).

2.2.2 Scratch

Το Scratch είναι μια διερμηνευόμενη δυναμική οπτική γλώσσα προγραμματισμού βασισμένη και υλοποιημένη σε Squeak. Όντας δυναμική, επιτρέπει αλλαγές του κώδικα ακόμη και κατά τη διάρκεια της εκτέλεσης των προγραμμάτων. Έχει ως στόχο τη διδασκαλία εννοιών προγραμματισμού σε παιδιά και εφήβους καθώς και την δημιουργία παιχνιδιών, βίντεο και μουσικής. Μπορεί να μεταφορτωθεί δωρεάν και χρησιμοποιείται σε μια ευρεία ποικιλία δράσεων εντός και εκτός σχολείου ανά τον κόσμο.

Το όνομα Scratch παραπέμπει στην τεχνική του scratching στα παλιά πικάπ, και αναφέρεται τόσο στη γλώσσα όσο και στην υλοποίηση της. Η ομοιότητα προς το scratching στη μουσική είναι η εύκολη επαναχρησιμοποίηση κομματιών: στο Scratch όλα τα αλληλεπιδραστικά αντικείμενα, γραφικά και ήχοι μπορούν εύκολα να εισαχθούν σε ένα νέο πρόγραμμα και να συνδυαστούν με νέους τρόπους. Έτσι οι αρχάριοι μπορούν να λάβουν γρήγορα αποτελέσματα και αποκτούν κίνητρο για να προσπαθήσουν περαιτέρω.

Ο ιστότοπος της κοινότητας του Scratch δείχνει ταχεία ανάπτυξη: από το Μάρτιο 2007 ως τον Ιούνιο 2008, άνω των 119,000 λογαριασμών έχουν καταχωρηθεί, με άνω των 148,000 έργων να έχουν φορτωθεί. Η ταχέως αυξανόμενη διεθνής κοινότητα έχει μεταφράσει το Scratch σε διάφορες γλώσσες, με περισσότερες υπό εξέλιξη.

Η δημοτικότητα του Scratch στην εκπαίδευση οφείλεται στην ευκολία με την οποία μπορούν να δημιουργηθούν προγράμματα: οι εντολές και οι δομές δεδομένων είναι απλές και είναι τουλάχιστον μερικά γραμμένες στην καθομιλουμένη, και η δομή του προγράμματος μπορεί να σχεδιαστεί όπως ένα παζλ, με αποσπώμενα κομμάτια κώδικα που μπορούν να μετακινηθούν και να προσαρμοστούν μαζί.

Το Scratch έχει αναπτυχθεί από μια μικρή ομάδα ερευνητών στο Lifelong Kindergarten Group στο MIT Media Lab. Η δικτυακή κοινότητα του Scratch έλαβε μια τιμητική μνεία στην κατηγορία Digital Communities στο Prix Ars Electronica 2008.

Το Scratch χρησιμοποιείται παγκοσμίως σε διάφορα σχολεία και εκπαιδευτικούς οργανισμούς. Ο δικτυακός τόπος του Scratch έχει αναπτύξει μια κοινότητα από αρχόμενους προγραμματιστές, μαθητές, δασκάλους και ερασιτέχνες, που

αλληλοπαρακινούνται να αναπτύξουν την δημιουργικότητα τους και τις προγραμματιστικές δεξιότητες τους. Ένα από τα φόρα στον ιστότοπο του Scratch είναι αφιερωμένο σε συζητήσεις μεταξύ εκπαιδευτικών.

Το σλόγκαν του Scratch είναι "Φαντάσου · Φτιάξε · Μοιράσου". Η έμφαση στο μοίρασμα είναι σημαντικό μέρος της παιδαγωγικής για το Scratch: τα προγράμματα δεν θεωρούνται μαύρα κουτιά, αλλά αντικείμενα που μπορούν να αναμιχθούν για τη δημιουργία νέων έργων. Ο μόνος τρόπος να γίνει ένα πρόγραμμα διαθέσιμο για χρήση είναι να δοθεί ο πηγαίος κώδικας του.

2.2.3 Visual Basic

Η γλώσσα προγραμματισμού Visual Basic (VB) αποτελεί την πρώτη και πλέον διαδεδομένη υλοποίηση αντικειμενοστραφούς γλώσσας προγραμματισμού σε περιβάλλον MS-Windows. Η σύνταξή της βασίζεται στην Quick Basic που αποτελούσε την πλέον προωθημένη υλοποίηση της Basic για το λειτουργικό σύστημα MS-Dos. Η VB είναι μια εξαιρετικά πλούσια γλώσσα που περιέχει κάποια χαρακτηριστικά από τη γλώσσα C, την μόνη γλώσσα υψηλού επιπέδου για προγραμματισμό συστημάτων. Κύρια διαφοροποίησή της από τις άλλες γλώσσες είναι το ολοκληρωμένο περιβάλλον αναπτύξεως λογισμικού που την συνοδεύει και επιτρέπει την γραφική (οπτική, visual) σύνθεση των διαφόρων στοιχείων που αποτελούν το μέσο επικοινωνίας με το χρήστη του τελικού προγράμματος, δηλαδή τα παράθυρα, μενού, κουμπιά κλπ.

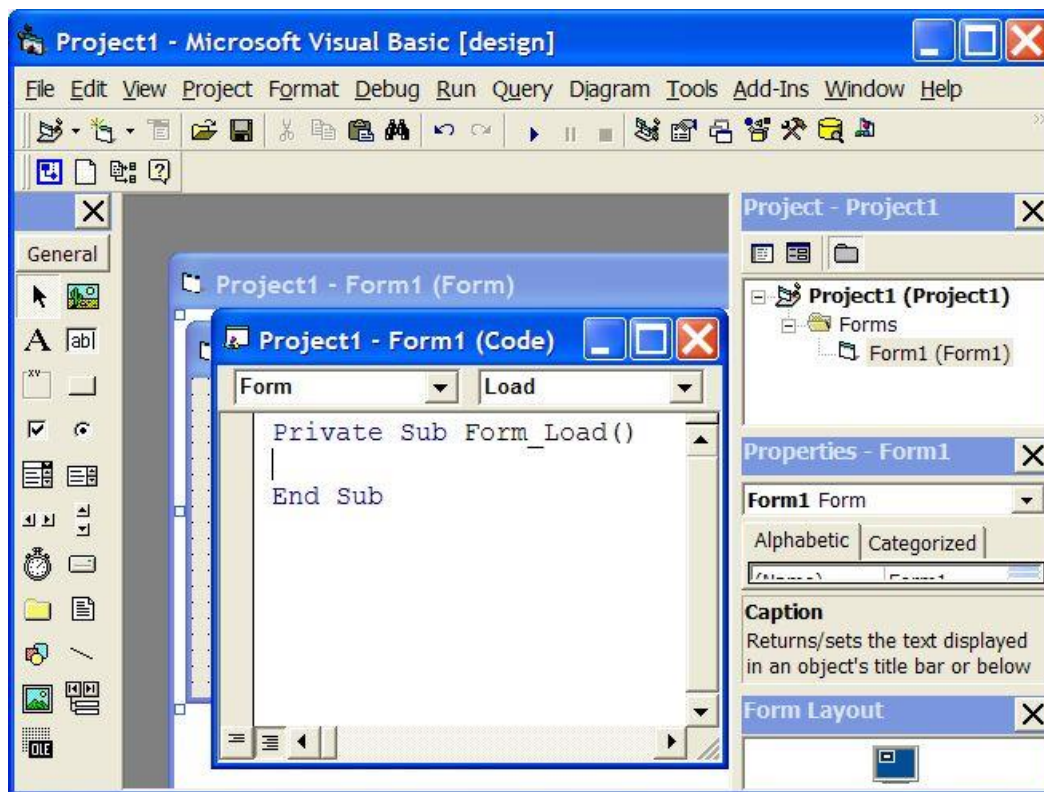


Εικόνα 2- Visual Basic

Είναι γλώσσα υψηλού επιπέδου. Πρωτοπαρουσιάστηκε το 1991 και από τότε έχει εξελιχθεί, αναπτυχθεί και διαδοθεί ευρύτατα για προγραμματισμό κάθε είδους εφαρμογών και οπωσδήποτε επιστημονικών και τεχνικών εφαρμογών, σε ηλεκτρονικούς υπολογιστές τύπου PC με λειτουργικό σύστημα MS- Windows. Με τον όρο γλώσσα υψηλού επιπέδου εννοούμε μια συμβολική γλώσσα προγραμματισμού με στοιχεία γλωσσικά και αριθμητικά συνηθισμένα σε επιστήμονες και τεχνικούς, που επιτρέπει την σύνταξη προγραμμάτων χωρίς καμία απαίτηση για γνώση της εσωτερικής δομής και λειτουργίας του υπολογιστή.

Το όνομά της προέρχεται από το Visual (οπτική) Basic (Beginners All Purpose Symbolic Instruction Code), δηλαδή είναι γλώσσα που επιτρέπει την ανάπτυξη κώδικα για κάθε χρήση από ανθρώπους που δεν είναι καθαρά επαγγελματίες προγραμματιστές. Είναι πολύ καλή για την έκφραση μαθηματικών τύπων και στον χειρισμό συμβολοσειρών.

Η γλώσσα VB έγινε το κατ' εξοχήν εργαλείο προγραμματισμού σε πολλά πανεπιστήμια, ερευνητικά κέντρα και γραφεία μελετών. Αποτελεί ιδιοκτησία της εταιρίας Microsoft που την αναπτύσσει και λόγω της διαδόσεώς της αποτελεί ένα τυποποιημένο εργαλείο ώστε τα προγράμματα να είναι μεταφέρσιμα από υπολογιστή σε υπολογιστή, όμως σε PC με λειτουργικό σύστημα MS-Windows.



Εικόνα 3- Visual Basic

Ακόμη, υπάρχει η Visual Basic for Applications ή VBA και η Embedded Visual Basic ή EBD.

Η VBA ξεκίνησε σαν αντικαταστάτης των «μακροεντολών» στο πακέτο MS- Office και σήμερα διαδίδεται με ταχύτητα σαν τυποποιημένο εργαλείο συντάξεως πρόσθετων εντολών και διαδικασιών (scripts) σε μεγάλα πακέτα λογισμικού όπως είναι το AutoCAD, Geomedia, Arc Info κλπ.

Η EBD απευθύνεται στον προγραμματισμό των υπολογιστών «παλάμης» (PDA), καθώς και τον προγραμματισμό ειδικών συστημάτων υπολογιστών σε αυτοκίνητα και διάφορες οικιακές συσκευές.

Όπως κάθε άλλη γλώσσα προγραμματισμού, η VB περιλαμβάνει εντολές για την επεξεργασία δεδομένων. Επιπλέον, περιλαμβάνει και «αντικείμενα». Πρόκειται για στοιχεία του περιβάλλοντος των Windows όπως παράθυρα, μενού, κουμπιά κλπ.

Οι εντολές περιλαμβάνουν δηλώσεις και εκτελέσιμες εντολές. Οι δηλώσεις προσδιορίζουν διάφορες παραμέτρους και χαρακτηριστικά του προγράμματος. Οι εκτελέσιμες εντολές εφαρμόζουν λογικές ή μαθηματικές πράξεις στα δεδομένα με την

βοήθεια τελεστών και ειδικών συναρτήσεων ή ρυθμίζουν την ροή του όλου προγράμματος και την εισαγωγή δεδομένων και εξαγωγή των αποτελεσμάτων.

2.2.4 Kodu

Το Kodu είναι ένα πρόγραμμα με το οποίο τα παιδιά δημιουργούν παιχνίδια για το PC και το Xbox μέσω μιας απλής οπτικής γλώσσας προγραμματισμού. Το Kodu μπορεί επίσης να χρησιμοποιηθεί για να διδάξει την επίλυση προβλημάτων, αφήγησης, καθώς και τον προγραμματισμό. Οποιοσδήποτε μπορεί να χρησιμοποιήσει το Kodu και να δημιουργήσει ένα παιχνίδι, είτε μικρά παιδιά, είτε ενήλικες χωρίς προηγούμενη εμπειρία σχεδιασμού ή προγραμματισμού.



Εικόνα 4- Kodu

* Το Kodu είναι μια γλώσσα οπτικού προγραμματισμού φτιαγμένη ειδικά για τη δημιουργία παιχνιδιών. Έχει σχεδιαστεί έτσι ώστε να είναι προσιτή σε παιδιά ηλικίας από 6 μόλις ετών και ευχάριστη για όλους.

* Καθώς έχει αναπτυχθεί από τη Microsoft Research, το Kodu «τρέχει» σε προσωπικούς υπολογιστές και Xbox επιτρέποντας τον γρήγορο σχεδιασμό.

Το Kodu διατίθεται δωρεάν.

* Το περιβάλλον εργασίας (User Interface) του Kodu είναι εύχρηστο και οπτικά ελκυστικό ειδικά σε μαθητές δημοτικού.

* Το συντακτικό του κώδικα είναι μεν απλό και εύληπτο, αλλά αφήνει μεγάλα περιθώρια για υπολογιστική σκέψη και για τις θεμελιώδεις αρχές του προγραμματισμού.

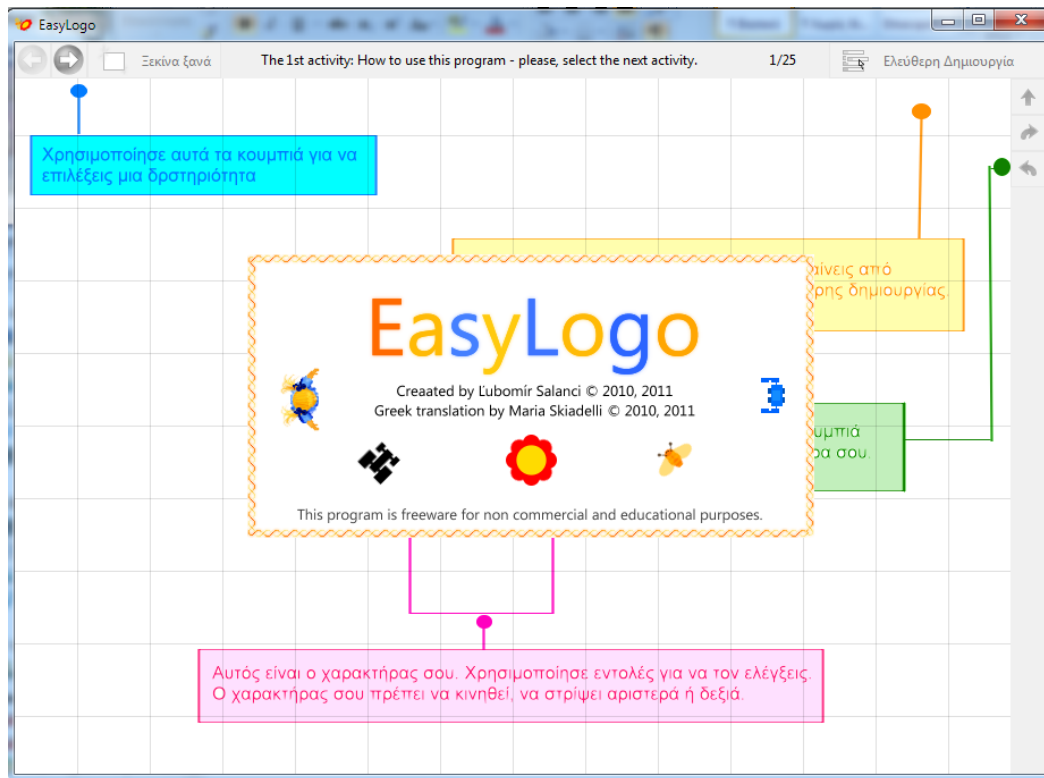
* Το Kodu ενσωματώνεται εύκολα στο αναλυτικό πρόγραμμα, καθώς συνδέεται στενά με τις αριθμητικές και γλωσσικές γνώσεις, αλλά και με τις φυσικές επιστήμες, τα μαθηματικά και τη γεωγραφία.

* Προσφέρεται άφθονη υποστήριξη με πόρους και διαδικτυακές κοινότητες, μεταξύ των οποίων τα: www.kodugamelab.com www.pil-network.com www.planetkodu.com

* Το Kodu αποτελεί μια εξαιρετική εισαγωγή στην υπολογιστική. Επίσης, είναι η αρχή μιας νέας Ευρωπαϊκής περιπέτειας με το Kodu Kup Europe.

2.2.5 Easy Logo

Το EasyLogo, αποτελεί ένα περιβάλλον οπτικού προγραμματισμού που στοχεύει στην εισαγωγική διδασκαλία των αρχών προγραμματισμού σε μαθητές δημοτικού σχολείου. Έχει αναπτυχθεί από το Department of Informatics Education, Comenius University of Bratislava και διατίθεται δωρεάν για εκπαιδευτικούς σκοπούς στην διεύθυνση: <http://edi.fmph.uniba.sk/~salanci/EasyLogo/>



Εικόνα 5- EasyLogo

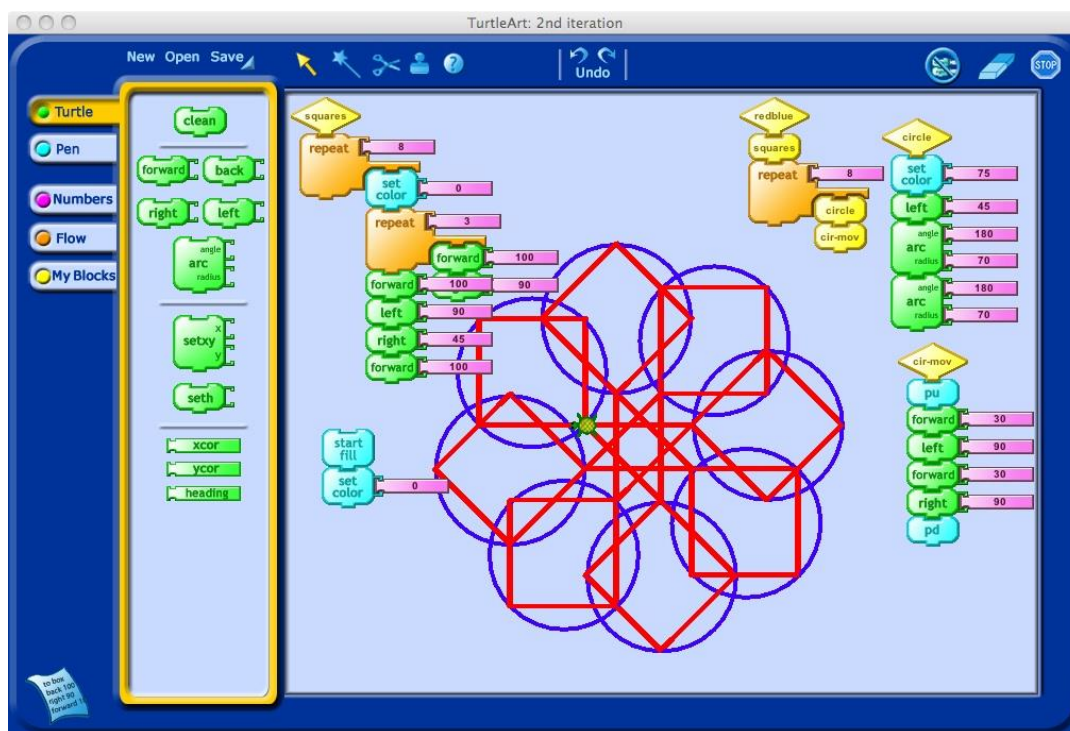
Στο περιβάλλον του EasyLogo:

- οι εντολές ταυτοποιούνται με οπτικά σύμβολα και λέξεις,
- είναι «Αυτονόητος» ο τρόπος δημιουργίας- ορισμού διαδικασιών,
- δεν χρησιμοποιούνται τις διαδικασίες με αναδρομικό τρόπο,
- το πρόγραμμα «τρέχει» συνεχώς παρέχοντας έτσι στον προγραμματιστή άμεση ανατροφοδότηση,
- λειτουργεί μόνο με τιμές και δεν διαθέτει μεταβλητές,
- δεν έχει εντολή αποφάσεων.

2.2.6 TurtleArt

Το TurtleArt (<http://turtleart.org/>) δίνει την δυνατότητα στον χρήστη να φτιάχνει εικόνες με τον υπολογιστή του. Η χελώνα ακολουθεί μια ακολουθία εντολών. Ο χρήστης είναι αυτός που καθορίζει την ακολουθία με τη ένωση κομματιών παζλ όπως

στα μπλοκς (blocks). Τα blocks μπορούν να πουν στη χελώνα να σχεδιάσει γραμμές και τόξα, να σχεδιάσει σε διάφορα χρώματα, να μεταβεί σε μια συγκεκριμένη θέση στην οθόνη, κ.λπ. Υπάρχουν επίσης blocks που επιτρέπουν να επαναληφθεί ή να ονομαστεί μια αλληλουχία. Άλλα blocks εκτελούν λογικές πράξεις.



Εικόνα 6- TurtleArt

Η ακολουθία των blocks ως πρόγραμμα που περιγράφει μια εικόνα. Αυτό το είδος του προγραμματισμού είναι εμπνευσμένο από τη γλώσσα προγραμματισμού LOGO. Είχε σχεδιαστεί για να είναι αρκετά εύκολο για τα παιδιά αλλά και αρκετά ισχυρό για ανθρώπους όλων των ηλικιών. Το TurtleArt εστιάζει στο να φτιάχνει εικόνες, ενώ επιτρέπει στον χρήστη να εξερευνήσει την γεωμετρία και τον προγραμματισμό.

Στο Turtle Art:

- δίνεται βάση στην (πρωτογενής) εντολή σχεδίασης τόξων,
- διατίθενται μόνο δύο μεταβλητές,
- οι διαδικασίες δεν έχουν μηχανισμούς περάσματος παραμέτρων,
- είναι περιορισμένο στη γεωμετρία της χελώνας, χωρίς πολυμεσικές δυνατότητες.

2.2.7 StarLogo & StarLogo TNG

Η StarLogo είναι μια ειδική έκδοση της γλώσσας προγραμματισμού Logo με τα γνωστά γραφικά της χελώνας. Ωστόσο, στη StarLogo παρέχεται η δυνατότητα ελέγχου χιλιάδων χελωνών παράλληλα. Επιπλέον, υπάρχει δυνατότητα αλληλεπίδρασης των χελωνών με το περιβάλλον τους, το οποίο αποτελείται από πλακίδια (patches). Για παράδειγμα, οι χελώνες μπορούν να ανιχνεύσουν συγκεκριμένες ιδιότητες των «πλακιδίων» πάνω στα οποία βρίσκονται καθώς περνούν πάνω από συγκεκριμένα πλακίδια με κάποιο ιδιαίτερο χαρακτηριστικό.

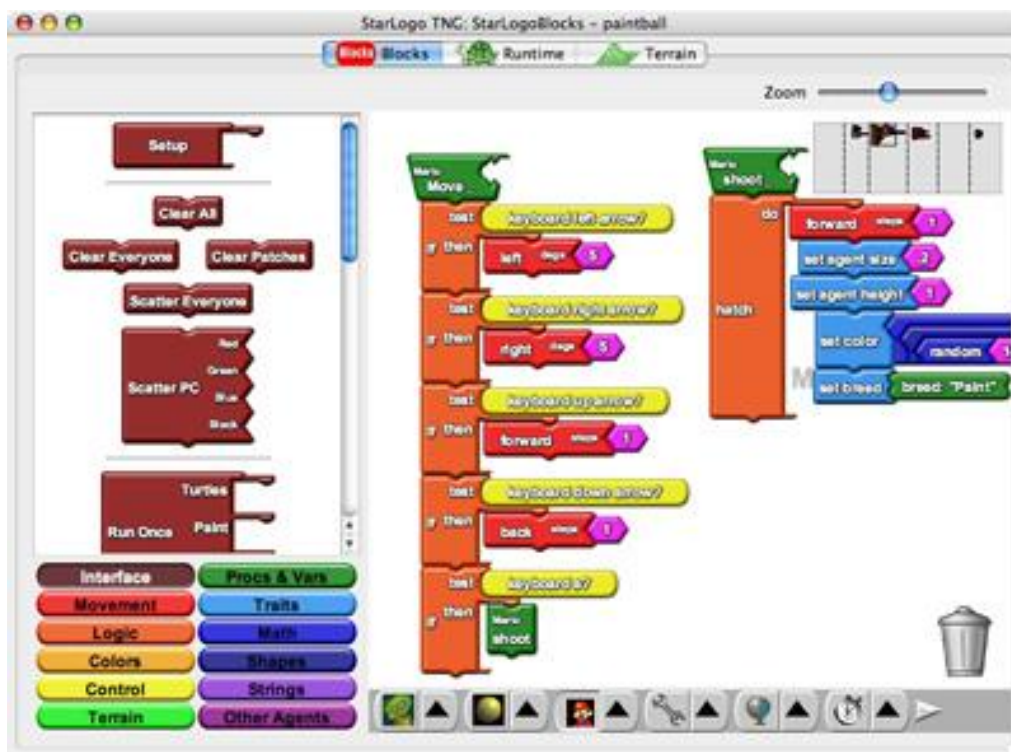
Η StarLogo, αναπτύχθηκε αρχικά από τον Mitchel Resnick στο Media Laboratory του MIT, αποτελεί στην ουσία έναν προγραμματιστικό μικρόκοσμο δημιουργίας προσομοιώσεων αποκεντρωμένων συστημάτων (decentralized systems) – συστημάτων που λειτουργούν χωρίς κεντρικό έλεγχο. Έχει ως στόχο τη στήριξη μαθητών και ερευνητών χωρίς προηγούμενη προγραμματιστική εμπειρία στη μοντελοποίηση και κατανόηση της συμπεριφοράς αποκεντρωμένων συστημάτων (κυκλοφοριακή συμφόρηση, σμήνη πουλιών, αποικία μυρμηγκιών κ.λπ.), αλλά και συστημάτων από πολλά άλλα επιστημονικά πεδία, όπως Μαθηματικά, Φυσική, Βιολογία και Πληροφορική.



Εικόνα 7- StarLogo

Στο δικτυακό τόπο της StarLogo <http://education.mit.edu/starlogo> υπάρχει διαθέσιμο πλούσιο υλικό.

Υπάρχουν και άλλες εκδόσεις της StarLogo, με ορισμένες τροποποιήσεις ως προς την αρχική StarLogo του MIT. Η StarLogo εισήγαγε έναν τελείως καινοτόμο τρόπο προγραμματισμού πολύ κατάλληλο για τη μελέτη αποκεντρωμένων συστημάτων και οδήγησε στη δημιουργία της StarLogo TNG (=StarLogo The Next Generation) ενός επίσης πρωτοποριακού συστήματος προγραμματισμού που καθιέρωσε (μαζί με τη Lego και τα συστήματα Lego Mindstorms) τα σύγχρονα περιβάλλοντα οπτικού προγραμματισμού, τα οποία σήμερα χρησιμοποιούνται σχεδόν αποκλειστικά στα εκπαιδευτικά προγραμματιστικά περιβάλλοντα (όπως το SCRATCH, το Game Maker κ.ά.)



Εικόνα 8- StarLogo TNG

Η StarLogo TNG διαθέτει:

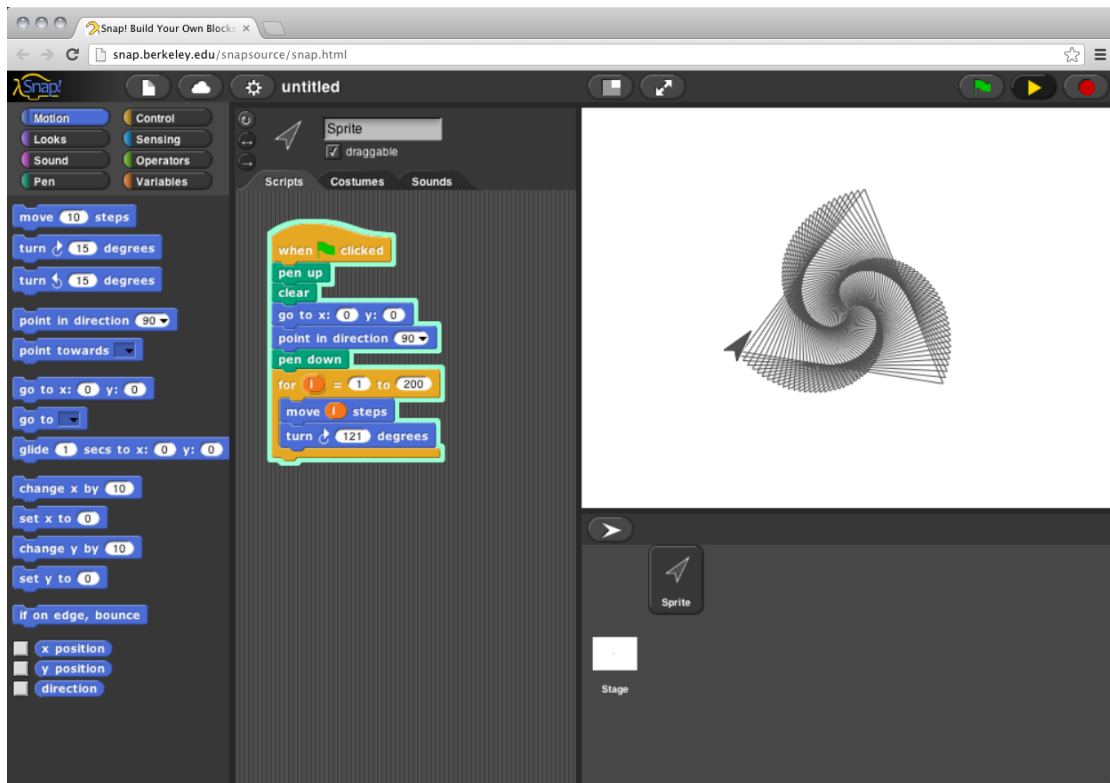
- μόνο εντολή επανάληψης για προκαθορισμένο πλήθος επαναλήψεων,

- διαδικασίες οι οποίες έχουν μηχανισμούς περάσματος παραμέτρων,
- τρισδιάστατο terrain, προσομοιώσεις φυσικών συστημάτων & μοντελοποίηση, «αυθεντικά» παιχνίδια,
- διαχείριση (ιδιόρρυθμη) πολυμεσικών στοιχείων.

2.2.8 BYOB

Το BYOB είναι ένα περιβάλλον προγραμματισμού, που στηρίζεται στη γλώσσα LOGO (<https://snap.berkeley.edu/>). Αποτελεί προέκταση του Scratch. Η νέα δυνατότητα που προσφέρει είναι η υποστήριξη διαδικασιών (δηλαδή εντολών που κατασκευάζει ο ίδιος ο προγραμματιστής και δεν προϋπάρχουν στη γλώσσα). Από εκεί προέρχεται και το όνομα Build Your Own Blocks(BYOB) που σε ελεύθερη μετάφραση σημαίνει “φτιάξε τις δικές σου εντολές”.

Σύμφωνα με τον κατασκευαστή του, το BYOB(Scratch) είναι “Ένα εργαλείο για να κατασκευάσεις τις δικές σου διαδραστικές ιστορίες, παιχνίδια, μουσική ή τέχνη”. Προσφέρει πάρα πολλές δυνατότητες, που σε συνδυασμό με την ευκολία χρήσης, το καθιστούν ένα ιδιαίτερα ελκυστικό περιβάλλον εργασίας, για μαθητές που έρχονται για πρώτη φορά σε επαφή με το κόσμο του προγραμματισμού.



Εικόνα 9- BYOB

Η BYOB διαθέτει:

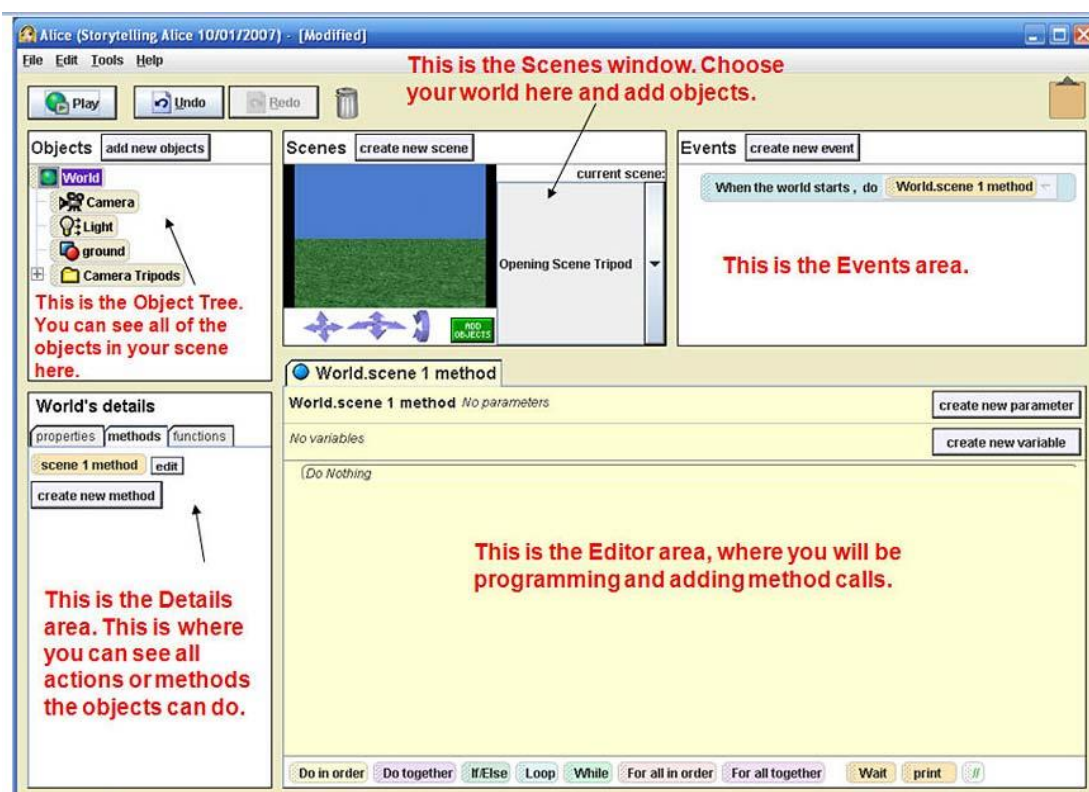
- διαχείριση (κλασική) πολυμεσικών στοιχείων,
- διαδικασίες με ισχυρούς μηχανισμούς περάσματος παραμέτρων,
- ποικιλία εντολών επανάληψης,
- δυνατότητα χρήσης των διαδικασιών έτσι ώστε ο προγραμματιστής να μπορεί να κατασκευάσει τις δικές του εντολές.

2.2.9 Alice

Το Alice (<http://www.alice.org/index.php>) δημιουργήθηκε ουσιαστικά το 1999 από το Πανεπιστήμιο Carnegie Mellon με κύριους ερευνητές τους Wanda Dann, Stephen Cooper και Randy Pausch. Το Alice είναι ένα προγραμματιστικό περιβάλλον που χρησιμοποιείται ευρέως για εκπαιδευτικό σκοπό. Είναι ένα 3D περιβάλλον για δημιουργία εικονικών κόσμων που εμπεριέχει δυναμικές κινήσεις των χαρακτήρων και αλληλεπίδραση τους με το χρήστη. Αυτό είναι και το κύριο πλεονέκτημα του

Alice. Χρησιμοποιεί 3D γραφικά ώστε να προσελκύσει τους μαθητές. Είναι πολύ πιο εύκολο και ευχάριστο για παράδειγμα να χρησιμοποιήσει κανείς μια κλάση Ανθρώπου να της δώσει χαρακτηριστικά όπως φύλλο, χρώμα δέρματος, ύψος, χρώμα μαλλιών και ματιών και άλλα όταν βλέπει ένα χαρακτήρα σαν αντικείμενο στον κόσμο του προγράμματος που δημιουργεί, παρά να γράψει αμέτρητες γραμμές κώδικα χωρίς να έχει άμεσο αποτέλεσμα στην οθόνη του υπολογιστή.

Αναλυτικότερα στο προγραμματιστικό περιβάλλον Alice χρησιμοποιείται ένας συντάκτης μέσα στον οποίο ο χρήστης απλά επιλέγει, σύρει και αφήνει τα αντικείμενα και τις μεθόδους τους. Με αυτό τον τρόπο αποφεύγονται τα συντακτικά λάθη. Μπορεί να συμβούν λογικά λάθη αλλά εφόσον ο χρήστης μπορεί να εκτελέσει το πρόγραμμα, θα εντοπίσει οπτικά το λάθος και θα το διορθώσει εύκολα και κατόπιν δοκιμών. Έτσι προσαρμόζεται εύκολα ο κώδικας και εντοπίζονται τα αποτελέσματα των αλλαγών.



Εικόνα 10- Alice

2.2.10 Game Maker

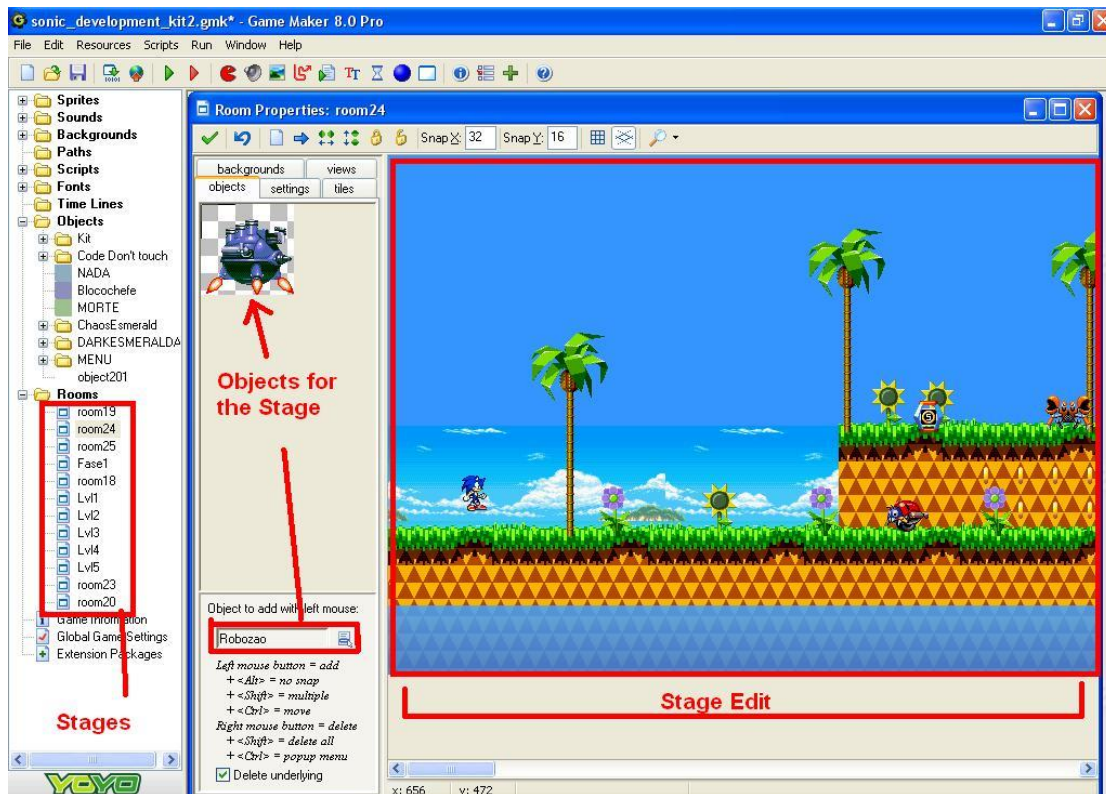
Το GameMaker είναι σημαντικό εκπαιδευτικό εργαλείο, αφού μπορεί να διδάξει έως και σε παιδιά πρωτοβάθμιας εκπαίδευσης, τις βασικές έννοιες του προγραμματισμού και προπαντός κάνοντάς αυτό διασκεδαστικά, δίνοντάς τους έτσι ένα ισχυρό κίνητρο. Επίσης, χρησιμοποιεί και αναπτύσσει ικανότητες όπως η λογική, η δημιουργικότητα, η φαντασία, η πρωτοτυπία, η καλλιτεχνία και άλλες δημιουργικές και νοητικές ικανότητες. Δημιουργήθηκε από τον καθηγητή στο πανεπιστήμιο της Ουτρέχτης Δρ. Μαρκ Όβερμαρς.

Επιπλέον, όταν το παιδί/μαθητής αποκτήσει όλες τις ικανότητες και τη βασική εμπειρία από το σχολείο (για παράδειγμα, στην Ελλάδα, από τη πρώτη του επαφή στον προγραμματισμό στην Γ' Λυκείου, μέσω του μαθήματος "Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον"), αυτό το πρόγραμμα, μπορεί να χρησιμοποιηθεί για περαιτέρω εξάσκηση στον τομέα του προγραμματισμού, μέσω δημιουργικών και συνάμα ψυχαγωγικών, όπως και καλλιτεχνικών ενασχολήσεων με το GameMaker.

Αυτό το πρόγραμμα καθιστά το νέο ικανό και περισσότερο προετοιμασμένο ως άτομο στο μέλλον, με τις γλώσσες και τα περιβάλλοντα προγραμματισμού που χρησιμοποιούνται ευρέως (ιδιαίτερα με τον αντικειμενοστρεφή προγραμματισμό). Όπως προαναφέρθηκε, προσθέτει και άλλες γνώσεις στον δημιουργικό και καλλιτεχνικό τομέα και τέλος στις έννοιες της τέχνης που αφορά στο σχεδιασμό ηλεκτρονικών παιχνιδιών στον υπολογιστή.

Το Game Maker προσφέρει απεριόριστες επιλογές και εναπόκειται στο χρήστη του τι μπορεί να δημιουργήσει.

Υπάρχει προ εγκατεστημένη συλλογή βιβλιοθηκών (Libraries) η οποία μπορεί να επεκταθεί - είτε βρίσκοντας στο Διαδίκτυο έτοιμες είτε δημιουργώντας τις δικές του με το κατάλληλο εργαλείο - με τη μορφή "Drag & Drop" στοιχείων, καθώς και γράφοντας τα δικά του σενάρια (scripts), ο χρήστης έχει πολλές επιλογές. Το σύστημα "Drag & Drop" είναι αυτό που συστήνεται για παιδιά πρωτοβάθμιας εκπαίδευσης.



Εικόνα 11- GameMaker

Επίσης, και πιο σημαντικά, το GameMaker διαθέτει μία ισχυρή γλώσσα προγραμματισμού (GameMaker Language ή GML) στην οποία μπορεί να εξασκηθεί οποιοσδήποτε διαθέτει τις βασικές γνώσεις προγραμματισμού, ακολουθώντας την προγραμματιστική σύνταξη την οποία έχει μάθει ή στην οποία αρέσκεται.

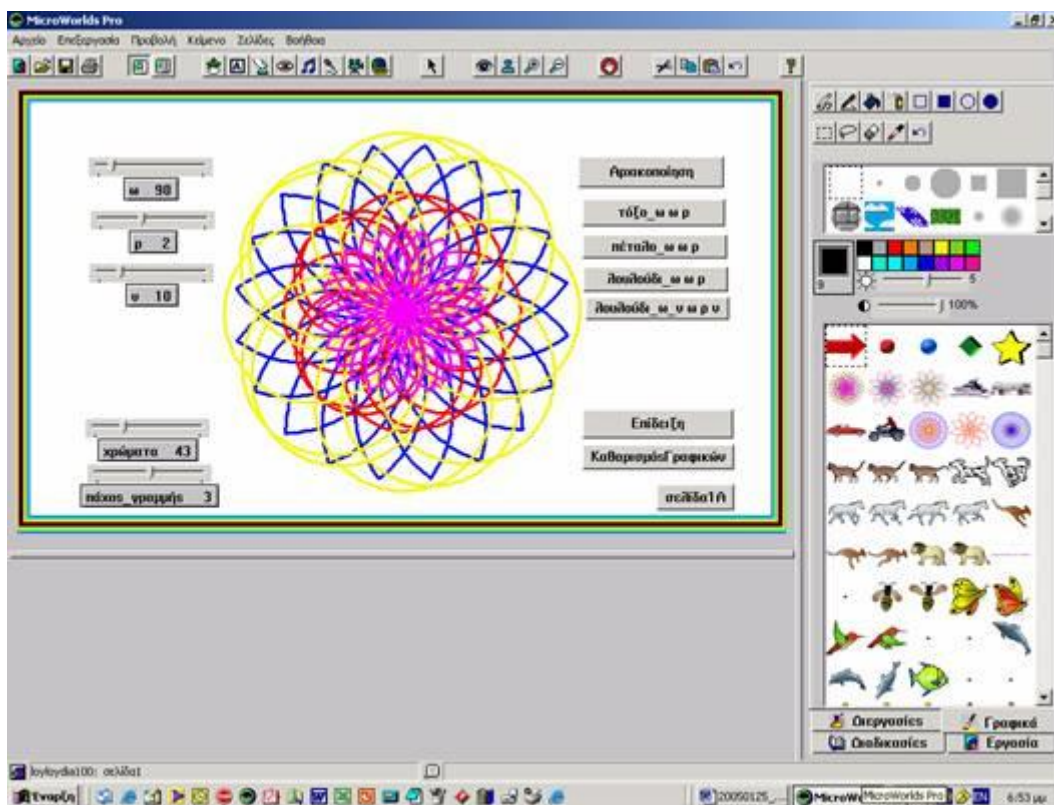
Το GameMaker χρησιμοποιείται ήδη σε πολλά μεγάλα σχολεία ανά τον κόσμο και συνεχώς επεκτείνεται σε περισσότερα.

2.2.11 Microworlds Pro

Το Microworlds Pro (καναδέζικη έκδοση της LCSi, 1999 και εξελληνισμένη έκδοση, 2001) είναι ένα πολυμεσικό περιβάλλον το οποίο διαθέτει ως γλώσσα προγραμματισμού τη γνωστή Logo και έχει σχεδιαστεί σύμφωνα με τις παιδαγωγικές αντιλήψεις του δημιουργού της S. Papert, συνεργάτη του J. Piaget.

Το περιβάλλον αυτό θεωρείται κατάλληλο τόσο για την εξοικείωση των μαθητών με έννοιες και τεχνικές σχετικές με τις Τεχνολογίες της Πληροφορίας και της Επικοινωνίας (ΤΠΕ) όσο και για τη διδασκαλία όλων των μαθημάτων του

Προγράμματος Σπουδών της Πρωτοβάθμιας και πολλών γνωστικών αντικειμένων της Δευτεροβάθμιας εκπαίδευσης.



Εικόνα 12- Microworlds Pro

Το λογισμικό Microworlds Pro έχει τις παρακάτω δυνατότητες:

- Με τη βοήθεια της γλώσσας Logo παρέχεται η δυνατότητα δημιουργίας κινουμένων σχεδίων, προσομοιώσεων, μοντελοποιήσεων, εκπαιδευτικών παιχνιδιών, ασκήσεων και επίλυσης προβλημάτων.
- Μπορεί να χρησιμοποιηθεί από τα παιδιά για τη δημιουργία εργασιών με διαδραστικό και πολυμεσικό χαρακτήρα, καθώς περιέχει όλα τα απαραίτητα εργαλεία (εικόνες, ήχοι, κινούμενα σχέδια, βίντεο, γεωμετρικά σχήματα, κουμπιά κλπ).
- Περιλαμβάνει εργαλεία για απλή επεξεργασία κειμένου, επεξεργασίας εικόνας και σύνθεση ήχων.

- Επιτρέπει τη δημοσίευση των εργασιών σε μορφή html, επομένως και την ανάρτησή τους στο διαδίκτυο (π.χ. στη σχολική ιστοσελίδα)
- Συμβατότητα του περιβάλλοντος με τα γνωστά προγράμματα των Windows (όπως ο έλεγχος της ορθογραφίας) και η άμεση συνεργασία μέσω προγραμματισμού του λογιστικού φύλλου Excel.

Κεφάλαιο 3

3.1 Ανάλυση του app inventor

Το MIT app inventor είναι ένα εργαλείο προγραμματισμού βασισμένο στα blocks, το οποίο επιτρέπει σε όλους, ακόμη και στους αρχάριους χρήστες, να ξεκινήσουν να προγραμματίζουν και να κατασκευάζουν πλήρως λειτουργικές εφαρμογές για android συσκευές. Οι νεοφερμένοι στο app inventor μπορούν να δημιουργήσουν και να «τρέξουν» την πρώτη τους εφαρμογή σε μια ώρα ή και λιγότερο, καθώς και πιο σύνθετες εφαρμογές σε πολύ λιγότερο χρόνο από ότι με τις γλώσσες προγραμματισμού στις οποίες συνθέτεις κώδικα.



Εικόνα 13- What is app inventor

Αρχικά, αναπτύχθηκε από τον καθηγητή Hal Abelson και μια ομάδα από το google education, ενώ ο Hal ήταν σε εκπαιδευτική άδεια στη google, το app inventor λειτουργούσε ως μια υπηρεσία του web διοικούμενη από το προσωπικό του MIT Center για το Mobile Learning – μια συνεργασία του εργαστηρίου Πληροφορικής τεχνητής νοημοσύνης του MIT (CSAIL) και του MIT Media Lab.

Το MIT app inventor υποστηρίζει μια παγκόσμια κοινότητα περίπου 3 εκατομμυρίων χρηστών που εκπροσωπούν 195 χώρες από όλο τον κόσμο. Πάνω από 100 χιλιάδες χρήστες κατασκευάζουν πάνω από 7 εκατομμύρια εφαρμογές android, εβδομαδιαίως.

Όντας ένα εργαλείο ανοικτού κώδικα το οποίο επιδιώκει να καταστήσει τόσο τον προγραμματισμό όσο και την δημιουργία εφαρμογών προσιτά σε ένα ευρύ φάσμα κοινού, το MIT app inventor έχει αρπάξει την προσοχή από:

- Επίσημους και ανεπίσημους εκπαιδευτικούς που χρησιμοποιούν το app inventor για να εισάγουν στον προγραμματισμό φοιτητές πληροφορικής, μέλη συλλόγων επιστημών και όσων παρακολουθούν προγράμματα μετά το σχολείο και καλοκαιρινά κάμπινγκ. Επίσης, πολλοί εκπαιδευτικοί έχουν αρχίσει να χρησιμοποιούν το app inventor ώστε να αναπτύσσουν δικές τους εφαρμογές για την υποστήριξη των διδακτικών τους στόχων.
- Κυβέρνηση, δημόσιους εργαζόμενους και εθελοντές, οι οποίοι εκμεταλλεύτηκαν τη δύναμη του MIT App Inventor για την ανάπτυξη προσαρμοσμένων, συχνά υπερ-τοπικών εφαρμογών για την αντιμετώπιση φυσικών καταστροφών και τις ανάγκες της κοινότητας. Σχεδιαστές και διευθυντές διαχείρισης προϊόντων που έχουν διακρίνει τις δυνατότητες του MIT App Inventor στην υποστήριξη της διαδικασίας επαναληπτικής σχεδίασης μέσω ταχείας προτυποποίησης, ελέγχου και επανάληψης.
- Ερευνητές που χρησιμοποιούν MIT App Inventor για τη δημιουργία προσαρμοσμένων εφαρμογών για να ενισχύσουν την συλλογή και ανάλυση των δεδομένων για την υποστήριξη της έρευνας τους σε ένα ευρύ φάσμα τομέων από ιατρική μέχρι κοινωνική.
- Χομπίστες και επιχειρηματίες που έχουν μια ιδέα η οποία θέλουν να μετατραπεί γρήγορα σε εφαρμογή χωρίς το κόστος ή την σκληρή εκμάθηση που απαιτούν οι πιο παραδοσιακοί τρόποι ανάπτυξης εφαρμογών.

3.1.1 Το έργο της ομάδας του app inventor

Το έργο της ομάδας του MIT App Inventor οδηγείται από πέντε πρωταρχικούς στόχους:

- Τη διατήρηση και ενίσχυση του εργαλείου. Έχει πραγματοποιηθεί μια μακροπρόθεσμη δέσμευση για τη διατήρηση και βελτίωση του MIT App Inventor ως δωρεάν υπηρεσία στους τελικούς χρήστες. Για το σκοπό αυτό, βελτιώνεται συνεχώς το εργαλείο, προσθέτοντας του νέα χαρακτηριστικά, εντοπίζοντας τα τυχόν σφάλματα και αυξάνοντας την απόδοσή του.
- Κτίσιμο ανάπτυξης με επιχειρήσεις. Η Motorola και η Ford σε συνεργασία με δημόσιους φορείς και ιδιωτικές εταιρείες υποστηρίζουν μοναδικές εφαρμογές του εργαλείου με την ανάπτυξη ή την ενίσχυση προσαρμοσμένων χαρακτηριστικών του MIT App Inventor για την αντιμετώπιση των αναγκών των συνεργατών.
- Ενίσχυση των ικανοτήτων. Επιδιώκει να επεκτείνει τη χωρητικότητα της τυπικής και άτυπης εκπαίδευσης των υπολογιστών για τους ενήλικες και τους νέους σε όλο τον κόσμο. Με αυτόν τον τρόπο, έχει εμπλακεί ενεργά στην ανάπτυξη και τη διάδοση πόρων και εκπαιδευτικού υλικού για την υποστήριξη εκείνων που ενδιαφέρονται για τη δημιουργία προγραμμάτων στον τόπο τους.
- Προώθηση της εκπαίδευσης Πληροφορικής. Έχουν δεσμευτεί να εφιστούν την προσοχή στην κατάσταση της επιστήμης των υπολογιστών και της υπολογιστικής σκέψης στην εκπαίδευση. Από πολιτικής σκοπιάς δραστηριοποιούνται με τις τοπικές και εθνικές συζητήσεις σχετικά με τα πρότυπα. Από την άποψη της ευαισθητοποίησης, είναι πρόθυμοι να συμμετάσχουν και να υποστηρίξουν εκστρατείες μεγάλης κλίμακας που στοχεύουν στο να κερδίσουν νέα ακροατήρια.
- Πραγματοποίηση και υποστήριξη της κοινοτικής έρευνας. Προπτυχιακοί και μεταπτυχιακοί φοιτητές του MIT και των συνεργαζόμενων φορέων δραστηριοποιούνται ενεργά στη διεξαγωγή και τη δημοσίευση της έρευνας παράλληλα με την ανάπτυξη, τον έλεγχο και την αξιολόγηση της χρήσης του MIT App Inventor σε όλο τον κόσμο.

3.1.2 Οι Συνεργάτες

Υπάρχει συνεργασία με άλλους οργανισμούς εντός και εκτός του MIT που μοιράζονται το όραμά τους, συμπεριλαμβανομένων:

- Scheller Teacher Education Program
- MIT Media Lab
- MIT Computer Science and Artificial Intelligence Lab (CSAIL)

Το app inventor είναι ένα cloud- based εργαλείο το οποίο σημαίνει ότι ο χρήστης μπορεί να δημιουργήσει εφαρμογές κατευθείαν στον φυλλομετρητή.

3.1.3 Οδηγίες Φορμαρίσματος

Ο χρήστης μπορεί να ρυθμίσει το App Inventor και να αρχίσει την οικοδόμηση εφαρμογής μέσα σε λίγα λεπτά. Τα παράθυρα του design και του block editor τρέχουν αποκλειστικά στο πρόγραμμα περιήγησης (γνωστό και ως cloud). Για να μπορεί ο χρήστης να δει την εφαρμογή του σε μια συσκευή, την ώρα που την δημιουργεί (Live Testing), θα πρέπει να ακολουθήσει ένα από τα παρακάτω βήματα.

Εάν ο χρήστης χρησιμοποιεί μια συσκευή Android και έχει ασύρματη σύνδεση στο internet, μπορεί να ξεκινήσει τη δημιουργία εφαρμογών χωρίς τη λήψη οποιουδήποτε λογισμικού στον υπολογιστή του. Θα χρειαστεί να εγκαταστήσει μόνο το App Inventor Companion App για τη συσκευή. Συνιστάται η πρώτη από τις παρακάτω επιλογές.

Εάν ο χρήστης δεν έχει συσκευή Android, θα χρειαστεί να εγκαταστήσει το λογισμικό στον υπολογιστή του, ώστε να μπορεί να χρησιμοποιήσετε τον Android προσομοιωτή που εμφανίζονται στην οθόνη. Η επιλογή δύο εκ των παρακάτω.

Εάν ο χρήστης δεν έχει ασύρματη σύνδεση στο internet, θα χρειαστεί να εγκαταστήσει το λογισμικό στον υπολογιστή του, ώστε να μπορεί να συνδεθεί με την Android συσκευή μέσω USB. Επιλογή τρία εκ των παρακάτω.

Note: Η επιλογή σύνδεσης μέσω USB μπορεί να είναι δύσκολη, ειδικά για τα Windows. Για το λόγο αυτό συνιστάται η χρήση της ως τελευταία λύση.

1^η Επιλογή (προτεινόμενη)

Σύνδεση τηλεφώνου ή ταμπλέτας μέσω wi-fi

Ο χρήστης μπορεί να χρησιμοποιήσει το App Inventor χωρίς να κατεβάσει οτιδήποτε στον υπολογιστή του. Η ανάπτυξη των εφαρμογών γίνεται μέσω της ιστοσελίδας: ai2.appinventor.mit.edu. Για να γίνεται άμεσος έλεγχος αυτών των οποίων σχεδιάζονται στην Android συσκευή, πρέπει να γίνει εγκατάσταση του MIT App Inventor Companion app στο Android κινητό τηλέφωνο ή στο tablet. Μόλις εγκατασταθεί το Companion, μπορεί ο χρήστης να ανοίξει το App Inventor on the web, το Companion στη συσκευή του, και να δοκιμάσει τις εφαρμογές του, ταυτόχρονα με τη στιγμή που τις δημιουργεί.



**Build your project on
your computer**



**Test it in real-time on
your device**

Εικόνα 14- Σύνδεση τηλεφώνου ή ταμπλέτας μέσω wi-fi

Τα παρακάτω βήματα περιγράφουν τη διαδικασία:

Βήμα 1^ο : Λήψη και εγκατάσταση του Companion MIT AI2 App στο τηλέφωνό

Ο χρήστης πρέπει να ανοίξει το QR code scanner της συσκευής του και να σαρώσει τον κωδικό QR που παρατίθεται στην εικόνα 1 για να κατεβάσει το Companion App από το Play Store. Εάν δεν μπορεί να χρησιμοποιήσει το Play Store, μπορεί να χρησιμοποιήσει τον QR κωδικό που παρατίθεται στην εικόνα 15 για να κατεβάσει το Companion App απευθείας στο τηλέφωνό.

Play Store (προτεινόμενη: αυτόματες ενημερώσεις)



Εικόνα 15- QR code scanner

Σκανάρισμα αυτού του κώδικα ή επιλογή του παρακάτω σύνδεσμου για κατέβασμα από το Play Store
<https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3>

APK Αρχείο (απαιτούνται χειροκίνητες ενημερώσεις)



Εικόνα 16- Κώδικας

Σκανάρισμα αυτού του κώδικα ή επιλογή του παρακάτω σύνδεσμου
<http://ai2.appinventor.mit.edu/#4552323555655680> για κατέβασμα απευθείας.

Note: Τα QR code scanners παρέχονται από το Play Store (π.χ. ZXing).

Μετά τη λήψη, ο χρήστης ακολουθεί τις οδηγίες έτσι ώστε να εγκαταστήσει το Companion app στη συσκευή του. Η εγκατάσταση του Companion γίνεται μια φορά

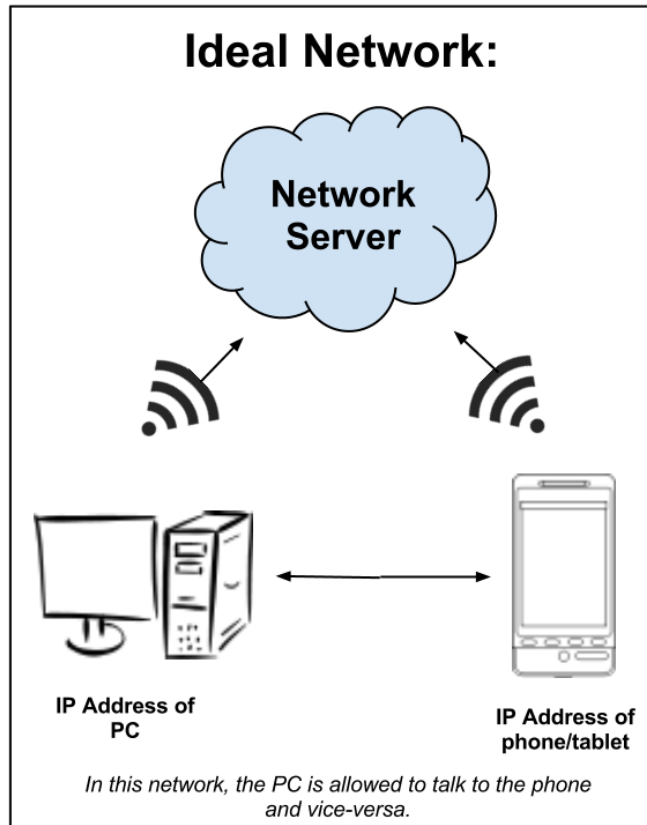
και στη συνέχεια μένει στο τηλέφωνο ή το tablet για κάθε φορά που θα χρησιμοποιείται το App Inventor.

Note 1: Εάν ο χρήστης δεν μπορεί να χρησιμοποιήσει το QR code, μπορεί να εγκαταστήσει το MIT AI2 Companion στο τηλέφωνο ή το tablet του χρησιμοποιώντας το πρόγραμμα περιήγησης στο Web στη συσκευή του για να μεταβεί στο Google Play Store και να αναζητήσει το MIT AI2 Companion.

Note 2: Εάν ο χρήστης δεν επιλέξει να μεταβεί στο Play Store μπορεί να φορτώσει άμεσα το app (aka "side load), θα πρέπει όμως να ενεργοποιήσει την επιλογή στις ρυθμίσεις της συσκευής του ώστε να επιτρέπονται οι εγκαταστάσεις εφαρμογών από «άγνωστες πηγές».

Βήμα 2ο: Σύνδεση υπολογιστή και συσκευής με το ίδιο δίκτυο wi-fi

Το App Inventor θα δείξει αυτόματα στον χρήστη την εφαρμογή την οποία δημιουργεί, αλλά μόνο αν ο υπολογιστής (τρέχοντας το App Inventor) και η Android συσκευή του (τρέχοντας το Companion) είναι συνδεδεμένα στο ίδιο δίκτυο WiFi.

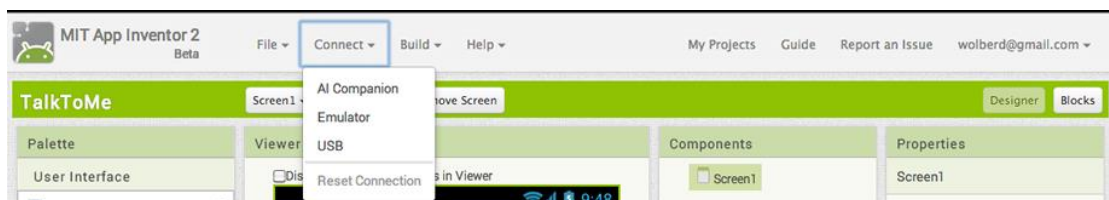


Εικόνα 17- Σύνδεση υπολογιστή και συσκευής με το ίδιο δίκτυο wi-fi

Βήμα 3: Άνοιγμα του App Inventor και σύνδεση με συσκευή

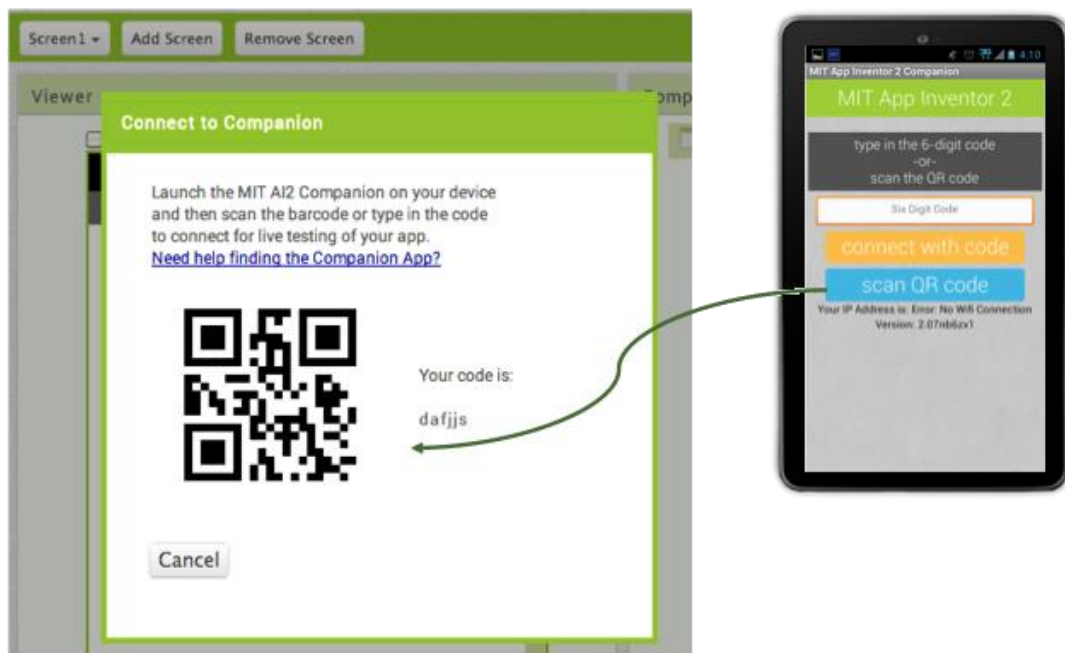
Ο χρήστης μεταβαίνει στο App Inventor και είτε ανοίγει ένα υπάρχον project (είτε δημιουργεί ένα νέο: Project > Start New Project και δίνει στο project ένα όνομα).

Στη συνέχεια επιλέγει "Connect" και "AI Companion" από το μενού στην κορυφή του AI2 browser.



Εικόνα 18- Άνοιγμα του app inventor

Ένα παράθυρο διαλόγου με ένα QR code θα εμφανιστεί στην οθόνη του υπολογιστή. Στη συσκευή, ο χρήστης εκκινεί την εφαρμογή Companion MIT App ακριβώς όπως θα έκανε με οποιαδήποτε εφαρμογή. Στη συνέχεια, επιλέγει το κουμπί "Scan QR code" για το Companion, και σαρώνει τον κώδικα στο παράθυρο του app inventor.



Εικόνα 19- Σύνδεση του app inventor με συσκευή

Μέσα σε λίγα δευτερόλεπτα, θα πρέπει ο χρήστης να δει την εφαρμογή που «χτίζει» στη συσκευή του. Αυτή θα ανανεώνεται καθώς γίνονται αλλαγές στα παράθυρα του σχεδιασμού και των μπλοκ, ένα χαρακτηριστικό που ονομάζεται "live testing".

Εάν ο χρήστης αντιμετωπίσει προβλήματα με τον προς σάρωση κώδικα QR ή η συσκευή του δεν διαθέτει σαρωτή, μπορεί να πληκτρολογήσει τον κωδικό που εμφανίζεται στον υπολογιστή στην περιοχή κειμένου, στην Android συσκευή του ακριβώς όπως φαίνεται. Ο κώδικας είναι ακριβώς κάτω από όπου η οθόνη του υπολογιστή δείχνει "Your code is" και αποτελείται από έξι χαρακτήρες. Πληκτρολογεί τους έξι χαρακτήρες και επιλέγει το πορτοκαλί "Connect with code". Ο χρήστης δεν πρέπει να πληκτρολογήσει Enter ή επιστροφή, μόνο τους έξι χαρακτήρες και στη συνέχεια το πορτοκαλί κουμπί.

Αντιμετώπιση προβλημάτων:

Εάν η εφαρμογή δεν εμφανίζεται στη συσκευή, τα πιο πιθανά προβλήματα είναι:

- Μπορεί η έκδοση του App Inventor Companion App να είναι παλιά. Ο χρήστης πρέπει να σιγουρευτεί ότι έχει το πιο πρόσφατο App Inventor 2.
- Η συσκευή να μην μπορεί να συνδεθεί με wi-fi. Ο χρήστης πρέπει να βεβαιωθεί ότι βλέπει μια διεύθυνση IP στο κάτω μέρος της οθόνης AICompanion App στο τηλέφωνο ή το tablet του.
- Η συσκευή δεν μπορεί να συνδεθεί με το ίδιο δίκτυο Wi-Fi με τον υπολογιστή. Ο χρήστης πρέπει να βεβαιωθεί ότι και οι δύο συσκευές συνδέονται με το ίδιο όνομα δικτύου Wi-Fi.
- Το σχολείο ή ο οργανισμός μπορεί να έχει τα πρωτόκολλα δικτύου σε θέση τέτοια ώστε η ασύρματη σύνδεση δεν επιτρέπεται. Σε αυτή την περίπτωση, μπορεί ακόμα να χρησιμοποιηθεί το App Inventor με τον προσομοιωτή ή μπορεί να χρησιμοποιηθεί ένα καλώδιο USB για να συνδεθεί η συσκευή.

2η Επιλογή (Όταν δεν υπάρχει android συσκευή, γίνεται χρήση του προσομοιωτή)

Εάν ο χρήστης δεν διαθέτει τηλέφωνο Android ή tablet, μπορεί ακόμη να δημιουργήσει εφαρμογές με το App Inventor. Το App Inventor παρέχει έναν Android emulator (προσομοιωτή), ο οποίος λειτουργεί ακριβώς όπως μια android συσκευή, αλλά εμφανίζεται στην οθόνη του υπολογιστή. Έτσι, ο χρήστης μπορεί να δοκιμάσει τις εφαρμογές του σε έναν προσομοιωτή και να εξακολουθεί να διανείμει τις εφαρμογές του σε άλλους, ακόμη και μέσω του Play Store.



Build your project on your computer Test it in real-time on your computer with the onscreen emulator

Εικόνα 20- Χρήση προσομοιωτή

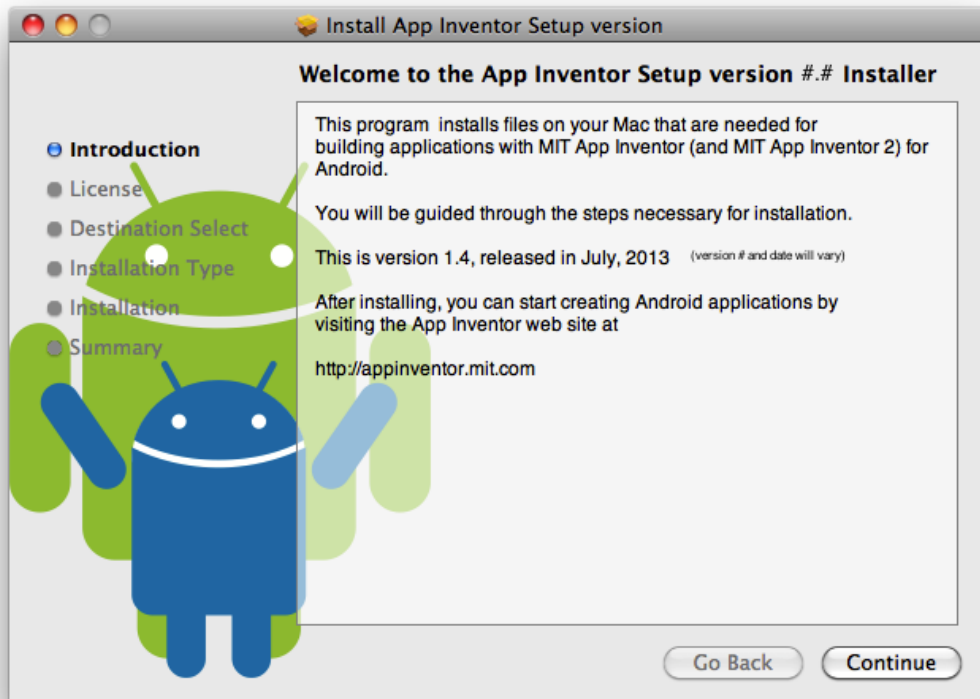
Για να χρησιμοποιήσει ο χρήστης τον εξομοιωτή, θα πρέπει πρώτα να εγκαταστήσει κάποιο λογισμικό στον υπολογιστή του (αυτό δεν είναι απαραίτητο όταν γίνεται χρήση του Wi-Fi). Στη συνέχεια παρατίθενται οι οδηγίες για το λειτουργικό σύστημα.

Βήμα 1ο: Εγκατάσταση του λογισμικού για το app inventor

- **Οδηγίες εγκατάστασης για Mac OS X**

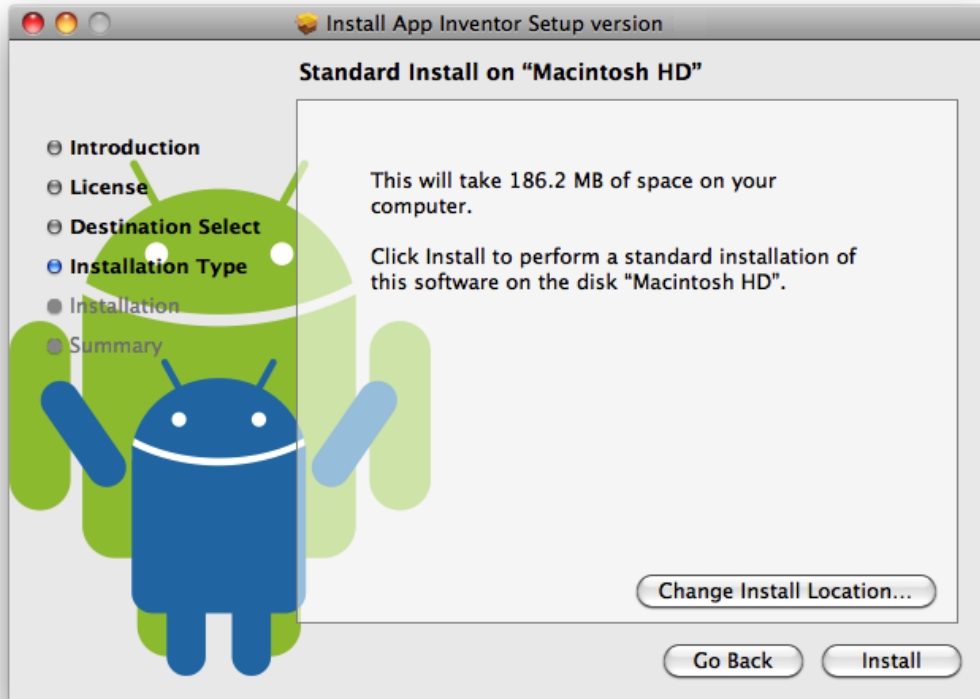
Για να διαθέτει ο χρήστης το Android emulator για Mac, πρέπει αρχικά να κατεβάσει και να εγκαταστήσει το πακέτο εγκατάστασης.

1. Κατέβασμα του προγράμματος εγκατάστασης.
2. Διπλό κλικ στο αρχείο για να ξεκινήσει το πρόγραμμα εγκατάστασης. (Μπορεί να χρειαστεί να εξεταστεί ο φάκελος λήψεων του browser. Το αρχείο ονομάζεται AppInventor_Setup_v_X.X.dmg (όπου το XX είναι ο αριθμός έκδοσης).
3. Επιλογή συνέχειας.



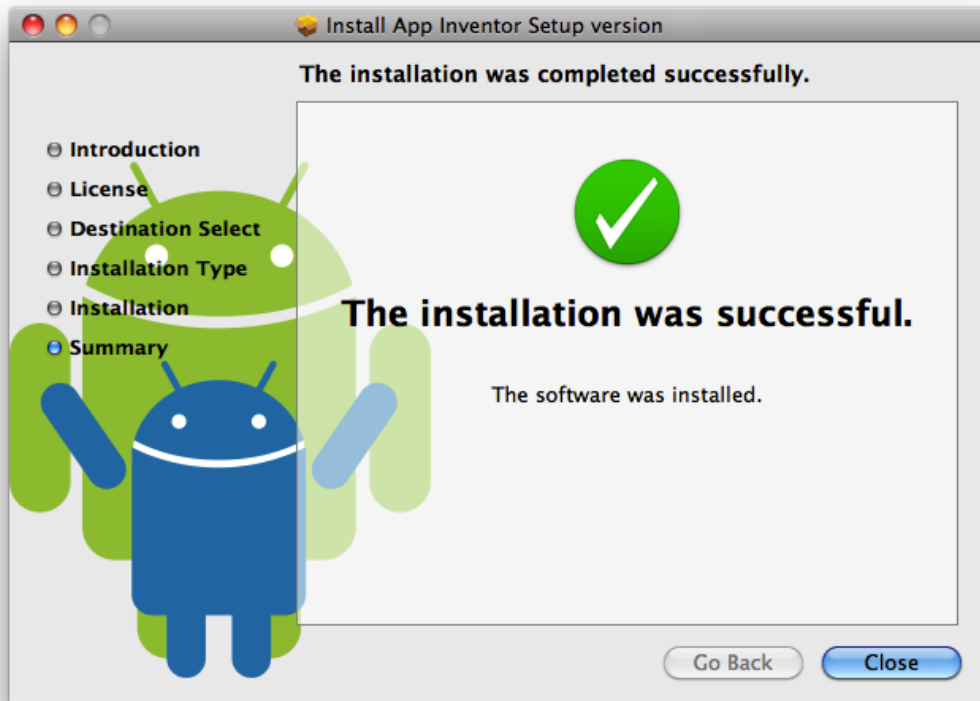
Εικόνα 21- Εγκατάσταση του λογισμικού app inventor

4. Διάβασμα και αποδοχή των όρων του λογισμικού.
5. Στην οθόνη του standard install, επιλογή install χωρίς να γίνει αλλαγή στην τοποθεσία εγκατάστασης.



Εικόνα 22- Εγκατάσταση λογισμικού για Macintosh

6. Εάν ζητηθεί, πρέπει να εισαχθεί το password έτσι ώστε να επιβεβαιωθεί ότι ο χρήστης θέλει πραγματικά να γίνει εγκατάσταση του λογισμικού. Επιλογή του ok.
7. Το πρόγραμμα εγκατάστασης επιβεβαιώνει ότι το πακέτο εγκατάστασης του app inventor έχει εγκατασταθεί με επιτυχία.



Εικόνα 23- Ολοκλήρωση εγκατάστασης

8. Εάν ο χρήστης κάνει ενημέρωση μιας παλαιότερης μορφής του λογισμικού, πρέπει να αποσυνδεθεί και να συνδεθεί ξανά πριν συνεχίσει να χρησιμοποιεί το app inventor.

• Οδηγίες εγκατάστασης για Windows

Η εγκατάσταση του λογισμικού σε περιβάλλον Windows, αποτελείται από δύο μέρη:

1. Εγκατάσταση του πακέτου εγκατάστασης λογισμικού του app inventor. Αυτό το βήμα είναι ίδιο για όλες τις συσκευές android, καθώς και για τα Windows XP, Vista και 7.
2. Εάν ο χρήστης επιθυμεί να συνδέσει τη συσκευή του με καλώδιο USB, τότε πρέπει να εγκαταστήσει στο κινητό του τους οδηγούς των Windows για Android κινητά.

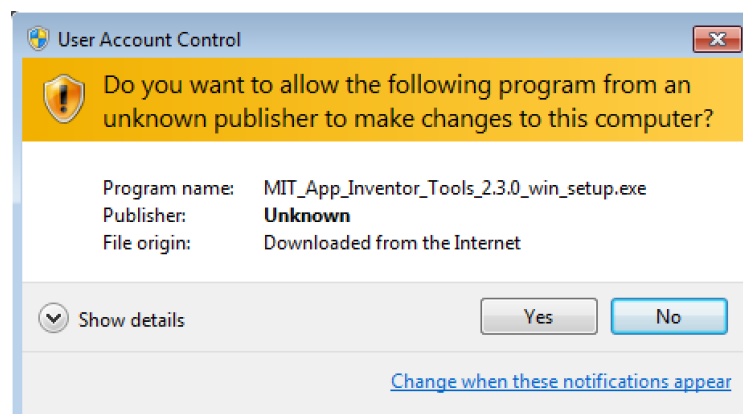
Note: Το App Inventor 2 δεν λειτουργεί με τον Internet Explorer. Για τους χρήστες των Windows, συνιστάτε η χρήση είτε του Chrome, είτε του Firefox ως browser για χρήση του App Inventor.

-Εγκατάσταση του πακέτου λογισμικού App Inventor

Ο χρήστης θα πρέπει να πραγματοποιήσει την εγκατάσταση από έναν λογαριασμό ο οποίος να έχει δικαιώματα διαχειριστή. Η εγκατάσταση μέσω ενός λογαριασμού στον οποίο ο χρήστης δεν έχει διαχειριστικά δικαιώματα, είναι προσωρινά μη διαθέσιμη.

Εάν έχει γίνει εγκατάσταση προηγούμενης έκδοσης του App Inventor 2, θα πρέπει να γίνει απεγκατάσταση και στη συνέχεια εγκατάσταση της πιο πρόσφατης έκδοσης.

1. «Κατέβασμα του προγράμματος εγκατάστασης».
2. Τοποθέτηση του αρχείου MIT_Appinventor_Tools_2.3.0 (~ 80 MB) στο φάκελο λήψεων (downloads) ή στην επιφάνεια εργασίας (desktop). Η θέση της λήψης στον υπολογιστή εξαρτάται από το πώς έχει διαμορφωθεί ο περιηγητής.
3. Άνοιγμα του αρχείου.
4. Κλικ στα βήματα της εγκατάστασης. Δεν χρειάζεται να γίνει αλλαγή στην θέση εγκατάστασης, αλλά πρέπει να καταγραφεί, επειδή μπορεί να χρειαστεί να ελεγχθούν οι οδηγοί αργότερα. Οι οδηγίες θα διαφέρουν ανάλογα με την έκδοση των Windows και με το αν ο χρήστης έχει συνδεθεί ως διαχειριστής ή όχι.
5. Μπορεί να ζητηθεί εάν ο χρήστης επιθυμεί να επιτρέψει ένα πρόγραμμα από έναν άγνωστο εκδότη να προβεί σε αλλαγές σε αυτόν τον υπολογιστή. Κλικ στο κουμπί «Ναι».



Εικόνα 24- Εγκατάσταση λογισμικού app inventor

-Τοποθεσία εγκατάστασης του λογισμικού

Στις περισσότερες περιπτώσεις, το App Inventor είναι σε θέση να επιλέξει μόνο του την τοποθεσία εγκατάστασης του λογισμικού. Αλλά όταν ζητείται η θέση του λογισμικού, η διαδρομή (path) η οποία πρέπει να εισαχθεί είναι: C:\Program Files\Appinventor\commands-for-Appinventor. Εάν χρησιμοποιείται μια μηχανή 64-bit, θα πρέπει να πληκτρολογηθεί Program Files (x86) αντί για Program Files. Επίσης, εάν δεν έχει γίνει εγκατάσταση του λογισμικού ως διαχειριστής, η εγκατάσταση θα γίνει τοπικά και όχι στο C:\Program Files.

• Οδηγίες εγκατάστασης για GNU/Linux

Χρειάζονται προνόμια sudo για να πραγματοποιηθεί η εγκατάσταση.

Σημείωση: Τα προγράμματα εγκατάστασης είναι για 32-bit λογισμικά. Αν ο χρήστης έχει ένα σύστημα 64-bit θα χρειαστεί να εγκαταστήσει τις βιβλιοθήκες του έτσι ώστε να μπορέσει το μηχάνημα να τρέξει 32-bit λογισμικό.

Ένας τρόπος για να γίνει αυτό είναι να εκτελεστεί η εντολή `sudo apt-get install lib32z1`, αλλά αυτό μπορεί να μην λειτουργήσει σε όλες τις εκδόσεις GNU / Linux, και ίσως χρειαστεί να γίνει κάποια περαιτέρω έρευνα για το κάθε σύστημα.

Εάν έχει γίνει παλαιότερα η εγκατάσταση του λογισμικού App Inventor, θα πρέπει να καταργηθούν τα παρακάτω αρχεία πριν από την εγκατάσταση του νέου λογισμικού:

```
sudo rm -rf /usr / google / appinventor
```

```
sudo rm -rf ~ / .appinventor
```

Οδηγίες για συστήματα στα οποία μπορούν να εγκατασταθούν πακέτα Debian.

Οι παρακάτω οδηγίες είναι για συστήματα τα οποία μπορούν να εγκαταστήσουν πακέτα Debian (π.χ. Debian ή Ubuntu):

Σημείωση: Εάν έχει γίνει πρότερη εγκατάσταση του App Inventor Classic, θα πρέπει να αφαιρεθεί, δεδομένου ότι μπορεί να παρεμβαίνει στη νέα εγκατάσταση. Αφαίρεση του πακέτου `sudo apt-get remove appinventor-setup`.

1. «Κατέβασμα» του πακέτου εγκατάστασης Appinventor για Debian. Αυτό είναι ένα αρχείο που ονομάζεται `appinventor2-setup_1.1_all.deb`. Πρόκειται για ένα αρχείο εγκατάστασης του πακέτου του Debian. Η θέση που θα καταλήξει στον υπολογιστή εξαρτάται από το πώς έχει διαμορφωθεί ο περιηγητής. Συνήθως, η αποθήκευση γίνεται στον φάκελο Λήψεις (downloads).
2. Αν το σύστημά μπορεί να εγκαταστήσει τα πακέτα απλώς κάνοντας κλικ στο αρχείο του πακέτου, τότε το κάνουμε με αυτό τον τρόπο.
3. Αν το σύστημά δεν υποστηρίζει εγκατάσταση με ένα κλικ, τότε πρέπει ο χρήστης να μεταβεί εκεί που έχει αποθηκευτεί το αρχείο και να εκτελέσει την παρακάτω εντολή `sudo dpkg --install appinventor2-setup_1.1_all.deb`. Και με τις δυο μεθόδους, ίσως χρειαστεί να διασφαλίσει ότι το deb αρχείο καθώς και η τοποθεσία είναι αναγνώσιμα και εκτελέσιμα. Σε μερικά συστήματα, το `sudo` δεν έχει εξ' ορισμού τα προνόμια να διαβάσει και να εκτελέσει όλα τα αρχεία.
4. Το λογισμικό θα πρέπει να εγκατασταθεί κάτω από το `/usr/google/appinventor`.
5. Μπορεί επίσης να χρειαστεί να ρυθμιστεί το σύστημα για να εντοπιστεί η συσκευή.

-Οδηγίες για άλλα συστήματα GNU / Linux

1. «Κατέβασμα» του προγράμματος εγκατάστασης του tar αρχείου Appinventor. Αυτό είναι ένα αρχείο ονομαζόμενο `appinventor2-setup_1.1.tar.gz`. Πρόκειται για ένα Gzip συμπιεσμένο tar αρχείο.
2. Εγκατάσταση των φακέλων, χρησιμοποιώντας την κατάλληλη μέθοδο ανάλογα με το λειτουργικό σύστημα. Θα χρειαστεί να ελεγχθεί ότι οι εντολές με κατεύθυνση το Appinventor καταλήγουν σε `/usr/google/appinventor`.

-Ξεκινώντας το AiStarter

Το πρόγραμμα aiStarter διαχειρίζεται την επικοινωνία μεταξύ του προγράμματος περιήγησης στο Web και της συσκευής Android. Θα πρέπει να εκτελείται κάθε φορά που ο χρήστης χρησιμοποιεί τον εξομοιωτή ή το USB καλώδιο. Αυτό δεν χρειάζεται

να τρέχει όταν ο χρήστης χρησιμοποιεί την ασύρματη σύνδεση. Κάθε φορά που κάποιος συνδέεται, ώστε να χρησιμοποιήσει το App Inventor με τον εξομοιωτή ή το καλώδιο USB, θα πρέπει να «τρέξει» το aiStarter. Αυτό μπορεί να γίνει με την εντολή

```
/usr/google/appinventor/commands-for-Appinventor/aiStarter &
```

Για λόγους ευκολίας, μπορεί ο χρήστης να ρυθμίσει η εντολή αυτή να «τρέχει» αυτόματα κάθε φορά που κάποιος συνδέεται ή ξεκινά το σύστημα. Ο τρόπος για να το κάνει αυτό εξαρτάται από το GNU / Linux που χρησιμοποιεί.

-Ορισμός Τοποθεσίας Εγκατάστασης

Στις περισσότερες περιπτώσεις, το App Inventor θα πρέπει να είναι σε θέση να ορίσει την τοποθεσία αποθήκευσης του λογισμικού από μόνο του. Εάν δεν το κάνει, η διαδρομή που πρέπει να οριστεί είναι η ακόλουθη:

```
/usr/google/appinventor/commands-for-Appinventor
```

Βήμα 2^ο: Έναρξη aiStarter (για Windows & GNU / Linux μόνο)

Για τη χρήση του emulator ή του καλωδίου USB απαιτείται η χρήση ενός προγράμματος που ονομάζεται aiStarter. Αυτό το πρόγραμμα είναι ο βοηθός που επιτρέπει στον browser να επικοινωνεί με το καλώδιο USB ή τον εξομοιωτή. Το πρόγραμμα aiStarter εγκαθιστάτε όταν εγκατασταθεί το πακέτο εγκατάστασης του App Inventor. Δεν χρειάζεται aiStarter αν χρησιμοποιείται η ασύρματη σύνδεση.

- Σε Mac, το aiStarter θα ξεκινήσει αυτόματα όταν συνδεθεί ο χρήστης στο λογαριασμό του και θα τρέξει αόρατα στο υπόβαθρο.
- Σε Windows, θα υπάρχουν συντομεύσεις για το aiStarter στην επιφάνεια εργασίας, από το μενού Έναρξη, όλα τα προγράμματα στο φάκελο εκκίνησης. Αν ο χρήστης επιλέξει να χρησιμοποιήσει τον εξομοιωτή του App Inventor, θα πρέπει να ξεκινήσει χειροκίνητα το aiStarter στον υπολογιστή όταν πραγματοποιήσει τη σύνδεση. Μπορεί να ξεκινήσει το aiStarter κάνοντας κλικ στο εικονίδιο που υπάρχει στην επιφάνεια εργασίας του υπολογιστή ή χρησιμοποιώντας την είσοδο στο μενού εκκίνησης.



Εικόνα 25 Εικονίδιο aiStarter στα Windows

Για να ξεκινήσει το aiStarter στα Windows, πρέπει να γίνει διπλό κλικ στο εικονίδιο (φαίνεται παραπάνω). Το aiStarter θα έχει ξεκινήσει με επιτυχία όταν εμφανιστεί ένα παράθυρο όπως το παρακάτω:

A screenshot of a Windows terminal window titled "aiStarter". The window has a standard Windows title bar with minimize, maximize, and close buttons. The terminal content is as follows:

```
Platform = Windows
AppInventor tools located here: "C:\Program Files (x86)"
Bottle server starting up (using WSGIRefServer<>)...
Listening on http://127.0.0.1:8004/
Hit Ctrl-C to quit.
```

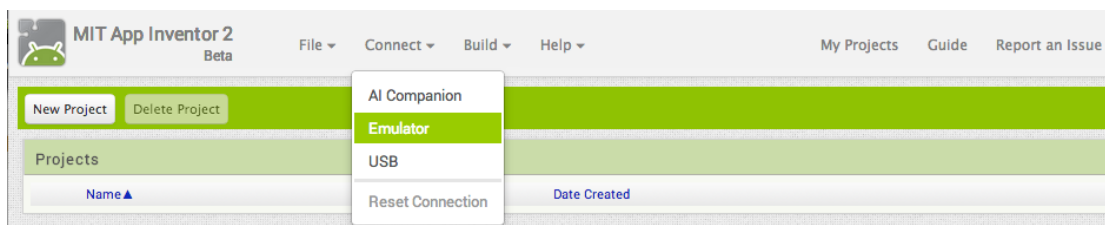
Εικόνα 26- aiStarter

- Για GNU/Linux, το aiStarter θα είναι στο φάκελο /usr/google/appinventor/commands-for-Appinventor και θα πρέπει να γίνει εκκίνηση αυτού χειροκίνητα. Μπορεί, επίσης, να γίνει εκκίνηση από την γραμμή εντολών με την εντολή /usr/google/appinventor/commands-for-appinventor/aiStarter &

Βήμα 3^ο: Άνοιγμα ενός προγράμματος App Inventor και σύνδεση με τον εξομοιωτή

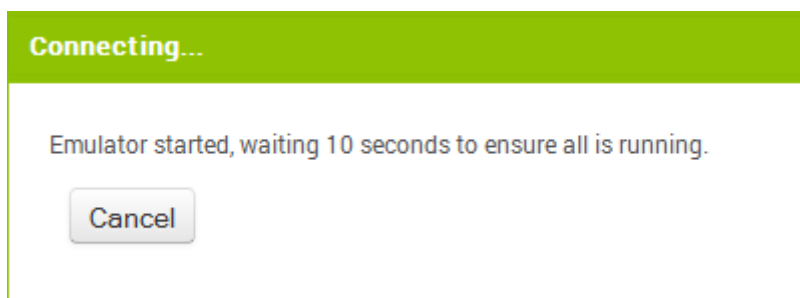
Αρχικά, ο χρήστης πρέπει να μεταβεί στο App Inventor και να ανοίξει ένα έργο (ή να δημιουργήσετε ένα νέο - Project > Start New Project και να δώσει ένα όνομα).

Στη συνέχεια, από το μενού App Inventor (στο cloud-based λογισμικό του App Inventor στο ai2.appinventor.mit.edu), μεταβαίνει στο Connect Menu και επιλέγει την επιλογή του εξομοιωτή (Emulator).



Εικόνα 27- Άνοιγμα του app inventor

Θα εμφανιστεί μια ειδοποίηση η οποία θα ενημερώνει τον χρήστη ότι ο εξομοιωτής είναι συνδεδεμένος. Η εκκίνηση του προσομοιωτή μπορεί να πάρει μερικά λεπτά.



Εικόνα 28- Αναμονή για εκκίνηση του προσομοιωτή

Ο χρήστης μπορεί να δει την ενημέρωση των οθονών καθώς ο προσομοιωτής ξεκινά, όπως δείχνει η παρακάτω εικόνα:

```

aiStarter
127.0.0.1 - - [07/Dec/2013 17:00:53] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:00:54] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:00:55] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:00:56] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:00:57] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:00:58] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:00:59] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:00] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:01] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:02] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:03] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:04] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:05] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:06] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:08] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:08] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:09] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:10] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:11] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:12] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:13] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:14] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:15] "GET /echeck/ HTTP/1.1" 200 38
127.0.0.1 - - [07/Dec/2013 17:01:16] "GET /echeck/ HTTP/1.1" 200 38

```

Εικόνα 29- Ενημέρωση οθονών

Ο εξομοιωτής θα εμφανιστεί αρχικά με μια κενή μαύρη οθόνη (#1). Ο χρήστης πρέπει να περιμένει έως ότου ο εξομοιωτής είναι έτοιμος, με έγχρωμο φόντο στην οθόνη (#2). Ακόμη και μετά την εμφάνιση του φόντου, θα πρέπει να περιμένει έως ότου η προσομοίωση του τηλεφώνου ολοκληρώσει την ετοιμασία της κάρτας SD (#3): θα υπάρχει ανακοίνωση στο πάνω μέρος της οθόνης του τηλεφώνου, όταν η κάρτα βρίσκεται στο στάδιο της προετοιμασίας (#4). Όταν συνδεθεί, ο εξομοιωτής θα ξεκινήσει και να δείξει την εφαρμογή που ο χρήστης έχει ανοίξει στο App Inventor.

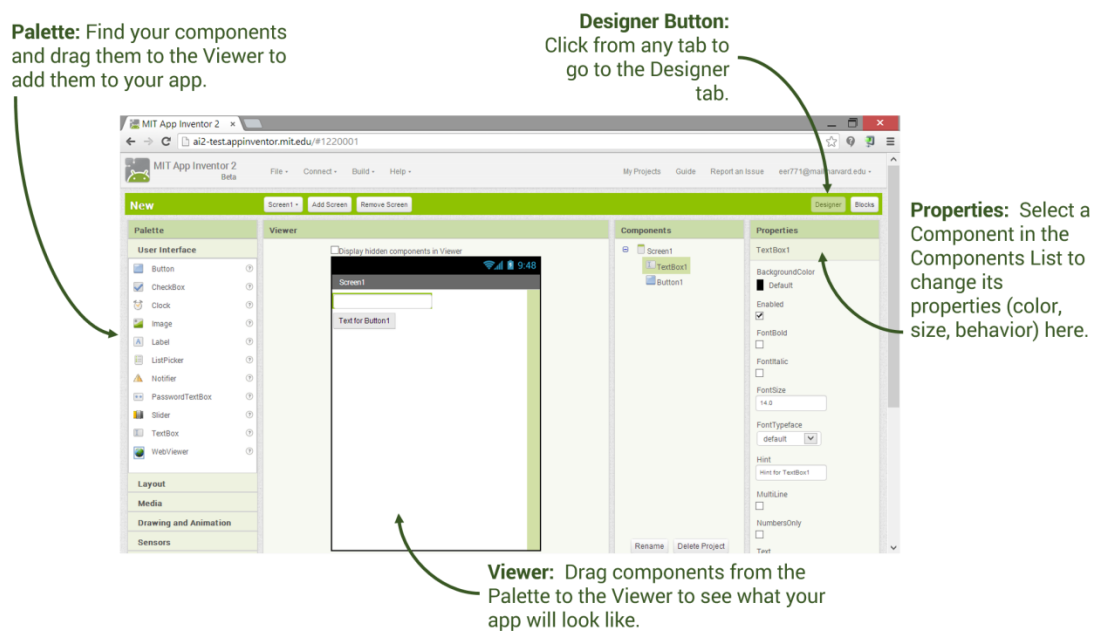


Εικόνα 30- Στάδια εμφάνισης προσομοιωτή έως ότου εμφανιστεί η εφαρμογή

Το app inventor αποτελείται από το designer (περιβάλλον σχεδιασμού) και το block editor (περιβάλλον κώδικα).

3.1.4 Ανάλυση της οθόνης Σχεδιασμού (designer)

Ο σχεδιασμός της οθόνης διεπαφής του χρήστη γίνεται με την χρήση στοιχείων τόσο εντός, όσο και εκτός οθόνης.



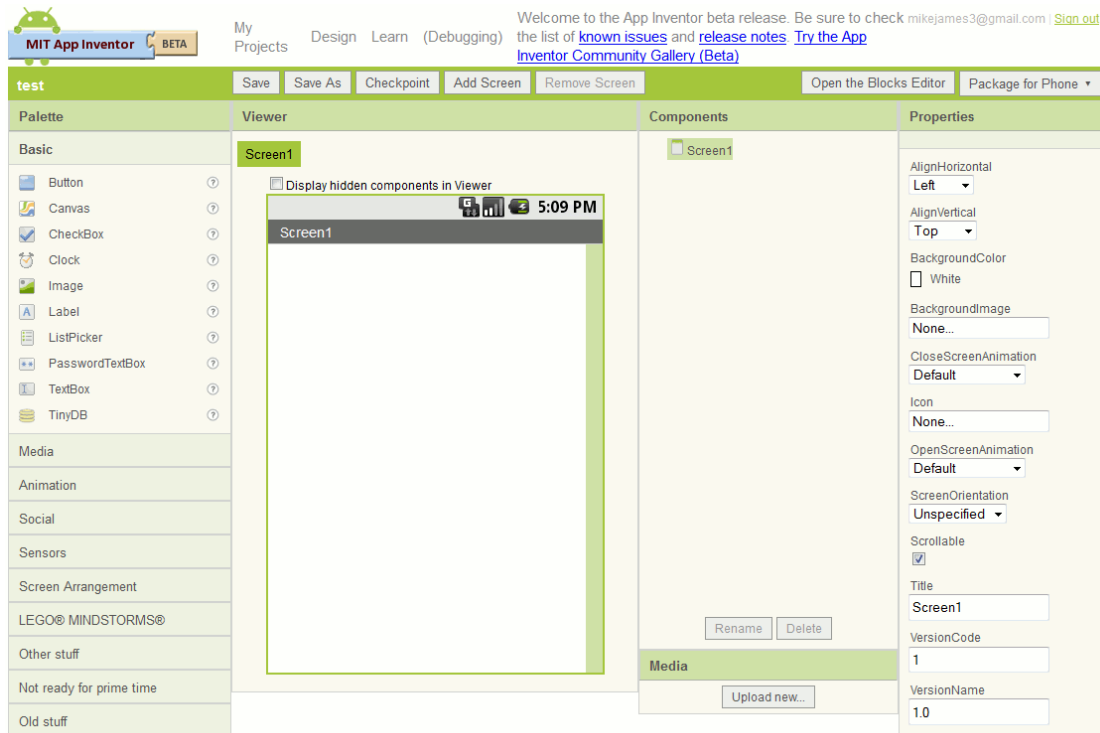
Εικόνα 31- Οθόνη Σχεδιασμού (designer)

Η οθόνη σχεδιασμού χωρίζεται σε τέσσερις περιοχές:

1. την παλέτα (Palette) όπου μπορεί ο χρήστης να επιλέξει τα στοιχεία που θέλει να χρησιμοποιήσει για την κατασκευή της διεπιφάνειας. Η παλέτα διαιρείται σε τμήματα ανάλογα με τον τύπο του στοιχείου (component).
2. την οθόνη του χρήστη (Viewer) που αναπαριστά την οθόνη του τηλεφώνου και χρησιμοποιείται για τη διάταξη των στοιχείων στην διεπιφάνεια του χρήστη (user interface- UI).
3. το παράθυρο των στοιχείων (Components) που παρουσιάζει μια διαφορετική όψη των στοιχείων που έχουν τοποθετηθεί στο παράθυρο του viewer και βοηθάει το

χρήστη να κατανοήσει την διαμόρφωση και τοποθέτηση των στοιχείων που έχουν χρησιμοποιηθεί.

4. το παράθυρο ιδιοτήτων (Properties) που απαριθμεί όλες τις ρυθμίσεις για το επιλεγμένο στοιχείο. Μπορεί να χρησιμοποιηθεί για να προσαρμοστεί κάθε στοιχείο.
5. το παράθυρο πολυμέσων (Media) μέσω του οποίου μπορούμε να εισάγουμε πολυμεσικά στοιχεία όπως εικόνες, βίντεο κλπ.



1. Palette

1.1 User Interface

1.1.1 Button

Το button είναι ένα κουμπί με την ικανότητα να ανιχνεύει τα πατήματα που γίνονται σε αυτό (clicks). Οι πτυχές της εμφάνισής του μπορούν να αλλάξουν, όπως επίσης και το εάν μπορεί να επιλεγεί μπορεί να αλλάξει τόσο από το περιβάλλον του designer όσο και του Block Editor.

Properties

BackgroundColor

Επιστρέφει το χρώμα φόντου του κουμπιού.

Enabled

Εάν έχει ρυθμιστεί, ο χρήστης μπορεί να αξιοποιήσει το πλαίσιο ελέγχου για να προκαλέσει δράση.

FontBold

Αν επιλεγθεί, το κείμενο κουμπιού εμφανίζεται με έντονους χαρακτήρες.

FontItalic

Αν επιλεγθεί, το κείμενο κουμπιού εμφανίζεται με πλάγια γράμματα.

FontSize

Μέγεθος γραμματοσειράς για το κείμενο του κουμπιού.

FontTypeface (μόνο για το designer)

Γραμματοσειρά για το κείμενο του κουμπιού.

Height

Ύψος του κουμπιού (y-size).

Image

Εικόνα για να εμφανιστεί στο κουμπιού.

Shape (μόνο για το designer)

Καθορίζει το σχήμα του κουμπιού (προεπιλογή, στρογγυλεμένα, ορθογώνια, οβάλ). Το σχήμα δεν θα είναι ορατό εάν υπάρχει εικόνα.

ShowFeedback

Καθορίζει αν μια οπτική ανάδραση θα πρέπει να εμφανίζεται για ένα κουμπιού που έχει εικόνα ως φόντο.

Text

Κείμενο το οποίο εμφανίζεται στο κουμπιού.

TextAlignment (μόνο για το designer)

Στοίχιση του κειμένου αριστερά, στο κέντρο ή δεξιά.

TextColor

Χρώμα για το κείμενο του κουμπιού.

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false αν δεν είναι.

Width

Πλάτος κουμπιού (x-size).

Events

Click()

Ο χρήστης επιλέγει και αφήνει το κουμπί.

GotFocus()

Δείχνει τον κέρσορα να κινείται πάνω από το κουμπί οπότε δηλώνει ότι το κουμπί μπορεί να επιλεγεί.

LongClick()

Ο χρήστης κρατάει το κουμπί κάτω.

LostFocus()

Δείχνει τον κέρσορα να έχει μετακινηθεί μακριά από το κουμπί οπότε δηλώνει ότι το κουμπί δεν μπορεί πλέον να επιλεγεί.

Touchdown()

Δηλώνει ότι το πλήκτρο πατήθηκε προς τα κάτω.

TouchUp()

Υποδεικνύει ότι ένα κουμπί έχει απελευθερωθεί.

1.1.2 CheckBox

Το check box μπορεί να ανιχνεύσει τα πατήματα του χρήστη και μπορεί να δώσει μια από τη δυαδική του τιμή ως απάντηση.

Ένα check box εγείρει ένα συμβάν όταν ο χρήστης το επιλέξει. Υπάρχουν πολλές ιδιότητες που επηρεάζουν την εμφάνισή του, οι οποίες μπορούν να ρυθμιστούν στο περιβάλλον είτε του designer είτε του Blocks Editor.

Properties

BackgroundColor

Επιστρέφει το χρώμα φόντου του πλαισίου.

Checked

True εάν το κουτί είναι επιλεγμένο, false στην αντίθετη περίπτωση.

Enabled

Εάν έχει ρυθμιστεί, ο χρήστης μπορεί να αξιοποιήσει το πλαίσιο ελέγχου για να προκαλέσει δράση.

Height

Ύψος πλαισίου (y-size).

Weight

Πλάτος πλαισίου (x-size).

Text

Κείμενο το οποίο θα εμφανίζεται στο check box.

TextColor

Χρώμα για το κείμενο του check box.

Visible

Εάν έχει ρυθμιστεί, το check box είναι ορατό.

Events

Click()

Ο χρήστης επιλέγει και αφήνει το check box.

GotFocus()

Το check box γίνεται το εστιασμένο στοιχείο.

LostFocus()

Το check box παύει να είναι το εστιασμένο στοιχείο.

1.1.3 DatePicker

Το date picker είναι ένα κουμπί που, όταν επιλεγεί, ανοίγει ένα pop-up παράθυρο για να επιτρέψει στο χρήστη να επιλέξει μια ημερομηνία.

Properties

BackgroundColor

Επιστρέφει το χρώμα φόντου του κουμπιού.

Day

Η ημέρα του μήνα που επιλέχθηκε τελευταία χρησιμοποιώντας το DatePicker.

Enabled

Εάν έχει ρυθμιστεί, ο χρήστης μπορεί να επιλέξει το check box για να προκαλέσει δράση.

FontBold

Αν επιλεγθεί, το κείμενο του κουμπιού εμφανίζεται με έντονους χαρακτήρες.

FontItalic

Αν επιλεγθεί, το κείμενο του κουμπιού εμφανίζεται με πλάγια γράμματα.

FontSize

Ορίζει το μέγεθος για το κείμενο του κουμπιού.

FontTypeface (μόνο για το designer)

Οικογένεια γραμματοσειράς για το κείμενο του κουμπιού.

Height

Image

Εικόνα για να εμφανίζεται στο κουμπί.

Month

Ο αριθμός του μήνα που επιλέχθηκε τελευταία φορά χρησιμοποιώντας το DatePicker.

MonthInText

Επιστρέφει το όνομα του μήνα, σε μορφή κειμένου, ο οποίος επιλέχθηκε τελευταία φορά χρησιμοποιώντας το DatePicker.

Shape (μόνο για το designer)

Καθορίζει το σχήμα του κουμπιού (προεπιλογή, στρογγυλεμένα, ορθογώνια, οβάλ). Το σχήμα δεν θα είναι ορατό εάν εμφανίζεται εικόνα.

ShowFeedback

Καθορίζει αν μια οπτική ανάδραση θα πρέπει να εμφανίζεται για ένα κουμπί που έχει εικόνα ως φόντο.

Text

Κείμενο για να εμφανιστεί στο κουμπί.

TextAlignment (μόνο για το designer)

Στοίχιση αριστερά, στο κέντρο ή δεξιά.

TextColor

Χρώμα για το κείμενο του κουμπιού.

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο φαίνεται και false εάν όχι.

Width

Year

Η χρονιά που επιλέχθηκε τελευταία χρησιμοποιώντας το DatePicker.

Events

AfterDateSet()

Γεγονότα που τρέχουν αφού ο χρήστης επιλέξει μια ημερομηνία.

GotFocus()

Δείχνει τον κέρσορα να κινείται πάνω από το κουμπί, δηλώνοντας έτσι ότι είναι δυνατή η επιλογή του.

LostFocus()

Δείχνει τον κέρσορα να έχει μετακινηθεί μακριά από το κουμπί έτσι ώστε να μην είναι πλέον δυνατόν να γίνει η επιλογή του.

Touchdown()

Δηλώνει ότι το πλήκτρο πατήθηκε προς τα κάτω.

TouchUp()

Υποδεικνύει ότι ένα κουμπί έχει απελευθερωθεί.

Methods

LaunchPicker()

Εκκινεί το αναδυόμενο παράθυρο για την επιλογή της ημερομηνίας.

SetDateToDisplay(number year, number month, number day)

Επιτρέπει στο χρήστη να ορίσει την ημερομηνία που θα εμφανίζεται όταν ανοίγει το παράθυρο του date picker. Οι έγκυρες τιμές για το πεδίο μήνα είναι 1-12 και 1-31 για το πεδίο ημέρας.

1.1.4 Image

Στοιχείο για την εμφάνιση εικόνων. Η εικόνα που θα εμφανίζεται, και άλλες πτυχές της εμφάνισης της, προσδιορίζονται στο περιβάλλον του designer ή του blocks Editor.

Properties

Animation

Αυτή είναι μια περιορισμένη μορφή animation όπου μπορεί ο χρήστης να επισυνάψει ένα μικρό αριθμό τύπων κίνησης σε εικόνες. Οι επιτρεπόμενες κινήσεις είναι ScrollRightSlow, ScrollRight, ScrollRightFast, ScrollLeftSlow, ScrollLeft, ScrollLeftFast, και Stop.

Height

Picture

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false εάν όχι.

Width

Events

Κανένα

Methods

Καμία

1.1.5 Label

Οι ετικέτες είναι στοιχεία που χρησιμοποιούνται για εμφάνιση κειμένου.

Μια ετικέτα κειμένου καθορίζεται από ιδιότητες κειμένου. Άλλες ιδιότητες, οι οποίες μπορούν να ρυθμιστούν στο παράθυρο του design ή του μπλοκ Editor, ελέγχουν την εμφάνιση και τη θέση του κειμένου.

Properties

BackgroundColor

Χρώμα για το φόντο της ετικέτας.

FontBold

Αν επιλεγθεί, το κείμενο της ετικέτας εμφανίζεται με έντονους χαρακτήρες.

FontItalic

Αν επιλεγθεί, το κείμενο της ετικέτας εμφανίζεται με πλάγια γράμματα.

FontSize

Μέγεθος γραμματοσειράς για το κείμενο της ετικέτας.

FontTypeface

Είδος γραμματοσειράς για το κείμενο της ετικέτας.

HasMargins

Αναφέρει εάν η ετικέτα εμφανίζεται με περιθώρια. Και τα τέσσερα περιθώρια (αριστερά, δεξιά, πάνω, κάτω) είναι τα ίδια. Αυτή η ιδιότητα δεν έχει καμία επίδραση στο designer, γιατί οι ετικέτες εμφανίζονται πάντα με περιθώρια.

Height

Ύψος ετικέτας (y-size).

Width

Πλάτος ετικέτας (x-size).

Text

Κείμενο που εμφανίζεται στην ετικέτα.

TextAlignment

Αριστερή, στο κέντρο ή δεξιά στοίχιση.

TextColor

Χρώμα για το κείμενο της ετικέτας.

Visible

Εάν επιλεγεί, η ετικέτα θα είναι ορατή.

1.1.6 ListPicker

Ένα κουμπί που, όταν επιλεγεί, εμφανίζει μια λίστα από κείμενα μεταξύ των οποίων ο χρήστης μπορεί να επιλέξει. Τα κείμενα μπορούν να καθοριστούν μέσω του designer ή του blocks editor ορίζοντας την ιδιότητα ElementsFromString σε string-separated αλληλουχία (για παράδειγμα, επιλογή 1, επιλογή 2, επιλογή 3), είτε ορίζοντας τις ιδιότητες του στοιχείου σε μια λίστα στον blocks editor.

Ορίζοντας την ιδιότητα ShowFilterBar σε true, η λίστα θα μπορεί να φανεί. Άλλες ιδιότητες επηρεάζουν την εμφάνιση του κουμπιού (TextAlignment, BackgroundColor, κλπ) και το κατά πόσο μπορεί να επιλεγεί (ενεργοποιημένο).

Properties

BackgroundColor

Επιστρέφει το χρώμα φόντου του κουμπιού.

Elements

Λίστα των επιλογών για εμφάνιση (ως λίστα).

ElementsFromString

Χωρισμένες με κόμμα οι επιλογές λιστών προς χρήση.

Enabled

Εάν το ListPicker μπορεί να επιλεγεί.

FontBold (μόνο για το designer)

Αν επιλεγθεί, το κείμενο της λίστας εμφανίζεται με έντονους χαρακτήρες.

FontItalic (μόνο για το designer)

Αν επιλεγθεί, το κείμενο της λίστας εμφανίζεται με πλάγια γράμματα.

FontSize (μόνο για το designer)

Μέγεθος γραμματοσειράς για το κείμενο της λίστας.

FontTypeface (μόνο για το designer)

Είδος γραμματοσειράς για το κείμενο της λίστας.

Height

Ύψος κουτιού (y-size).

Image

Καθορίζει την τοποθεσία αποθήκευσης (path) της εικόνας του κουμπιού. Αν υπάρχει και εικόνα και BackgroundColor, μόνο η εικόνα θα είναι ορατή.

Selection

Το επιλεγμένο στοιχείο. Όταν αλλάξει άμεσα από τον προγραμματιστή, η ιδιότητα SelectionIndex, αλλάζει επίσης το πρώτο στοιχείο της ListPicker με την δεδομένη τιμή. Εάν δεν εμφανιστεί η τιμή, το SelectionIndex θα οριστεί σε 0.

SelectionIndex

Ο δείκτης του επιλεγμένου στοιχείου, ξεκινώντας από το 1. Εάν κανένα στοιχείο δεν έχει επιλεγεί, η τιμή θα είναι 0. Εάν γίνει προσπάθεια να τεθεί αυτό σε αριθμό μικρότερο από το 1 ή μεγαλύτερο από τον αριθμό των στοιχείων στο ListPicker, το SelectionIndex θα οριστεί στο 0, και ως επιλογή θα πρέπει να τεθεί το κενό κείμενο.

Σχήμα (μόνο για το designer)

Καθορίζει το σχήμα του κουμπιού (προεπιλογή, στρογγυλεμένα, ορθογώνια, οβάλ). Το σχήμα δεν θα είναι ορατό εάν υπάρχει εικόνα.

ShowFeedback

Καθορίζει αν μια οπτική ανάδραση θα πρέπει να εμφανίζεται για ένα κουμπί που έχει εικόνα ως φόντο.

ShowFilterBar

Επιστρέφει την τρέχουσα κατάσταση της ShowFilterBar δηλώνοντας εάν η μπάρα αναζήτησης φίλτρων (Search Filter Bar) θα εμφανίζεται στο ListPicker ή όχι.

Text

Τίτλος κειμένου που θα εμφανίζεται στο list picker.

TextAlignment (μόνο για το designer)

Στοίχιση κειμένου αριστερά, στο κέντρο ή δεξιά.

TextColor

Χρώμα κειμένου.

Title

Προαιρετικά μπορεί να εμφανιστεί τίτλος στην κορυφή του καταλόγου επιλογών.

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο εμφανίζεται και false αν όχι.

Width

Πλάτος κουτιού (x-size).

ItemTextColor

Το χρώμα του κειμένου των στοιχείων στη ListPicker.

ItemBackgroundColor

Το χρώμα του φόντου των στοιχείων στη ListPicker.

Events

AfterPicking ()

Γεγονός που εμφανίζεται αφότου η δραστηριότητα συλλογής επιστρέψει αποτέλεσμα και οι ιδιότητες έχουν συμπληρωθεί.

BeforePicking ()

Γεγονός που εμφανίζεται όταν το κουμπί ενός στοιχείου επιλεγεί ή εμφανίζεται η λίστα χρησιμοποιώντας το open block. Αυτό το γεγονός συμβαίνει πριν εμφανιστεί η λίστα των αντικειμένων, και μπορεί να χρησιμοποιηθεί για να προετοιμάσει τη λίστα πριν εμφανιστεί.

GotFocus ()

Δείχνει τον κέρσορα να κινείται πάνω στο κουμπί υποδηλώνοντας ότι μπορεί να πατηθεί.

LostFocus ()

Δείχνει τον κέρσορα να μετακινείται μακριά από το κουμπί υποδηλώνοντας ότι δεν μπορεί πλέον να πατηθεί.

Μέθοδοι

Open()

Ανοίγει το ListPicker, όταν ο χρήστης κάνει κλικ σε αυτό.

1.1.7 ListView

Αυτό είναι ένα ορατό στοιχείο που επιτρέπει την τοποθέτηση μιας λίστας στοιχείων κειμένου στην οθόνη.

Η λίστα αυτή μπορεί να ρυθμιστεί χρησιμοποιώντας την ιδιότητα ElementsFromString ή χρησιμοποιώντας τα elements block στον block editor.

Προειδοποίηση: Αυτό το στοιχείο δεν λειτουργεί σωστά στις οθόνες που είναι κυλιόμενες.

Properties

BackgroundColor

Το χρώμα φόντου της listview.

Elements

Λίστα από στοιχεία κειμένου για να δημιουργηθεί η λίστα.

ElementsFromString

Φτιάχνει μια λίστα με μια σειρά από στοιχεία κειμένου χωριζόμενα με κόμμα, όπως: τυρί, φρούτα, μπέικον, ραπανάκι. Κάθε λέξη πριν από το κόμμα θα είναι ένα στοιχείο στη λίστα.

Height

Καθορίζει το ύψος της προβαλλόμενης λίστας.

Selection

Επιστρέφει το κείμενο που επιλέχθηκε τελευταίο στη ListView.

SelectionIndex

Ο δείκτης του επιλεγμένου στοιχείου, ξεκινώντας από το 1. Εάν κανένα στοιχείο δεν έχει επιλεγεί, η τιμή θα είναι 0. Εάν γίνει προσπάθεια να τεθεί

σε αριθμό μικρότερο από το 1 ή μεγαλύτερο από τον αριθμό των στοιχείων στη ListView, το SelectionIndex θα οριστεί στο 0, και ως επιλογή θα πρέπει να τεθεί το κενό κείμενο.

ShowFilterBar

Ορίζει την προβολή της ShowFilterBar. True θα δείξει την μπάρα, False θα την κρύψει.

TextColor

Το χρώμα του κειμένου των στοιχείων στη listview.

Visible

Καθορίζει αν το στοιχείο θα είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false αν όχι.

Width

Καθορίζει το πλάτος της λίστας.

Events

AfterPicking ()

Ένα απλό γεγονός δημιουργείται μετά την επιλογή ενός στοιχείου από την λίστα. Το επιλεγμένο στοιχείο είναι διαθέσιμο στο selection property.

Μέθοδοι

Καμία

1.1.8 Notifier

Τα στοιχεία του Notifier εμφανίζουν διαλόγους ειδοποιήσεων, μηνύματα, προσωρινές ειδοποιήσεις, και δημιουργούν Android καταχωρήσεις ημερολογίου με τις ακόλουθες μεθόδους:

ShowMessageDialog: εμφανίζει ένα μήνυμα το οποίο ο χρήστης πρέπει να το απορρίψει με το πάτημα ενός κουμπιού.

ShowChooseDialog: εμφανίζει ένα μήνυμα με δύο κουμπιά τα οποία επιτρέπουν στο χρήστη να επιλέξει μία από τις δύο απαντήσεις, για παράδειγμα, ναι ή όχι, μετά τις οποίες εμφανίζεται το γεγονός AfterChoosing.

ShowTextDialog: επιτρέπει στο χρήστη να εισάγει κείμενο ως απάντηση στο μήνυμα, μετά το οποίο εμφανίζεται το γεγονός AfterTextInput.

ShowAlert: εμφανίζει μια προσωρινή ειδοποίηση που φεύγει από μόνη της μετά από ένα σύντομο χρονικό διάστημα.

ShowProgressDialog: εμφανίζει μια προειδοποίηση με μια μπάρα φόρτωσης η οποία δεν μπορεί να απορριφθεί από το χρήστη. Μπορεί να απορριφθεί μόνο με τη χρήση του μπλοκ DismissProgressDialog.

DismissProgressDialog: Απορρίπτει το παράθυρο διαλόγου που εμφανίζεται από ShowProgressDialog.

LogError: καταγράφει ένα μήνυμα λάθους στο αρχείο καταγραφής του Android.

Loginfo: καταγράφει ένα μήνυμα πληροφορίας στο αρχείο καταγραφής του Android.

LogWarning: καταγράφει ένα μήνυμα προειδοποίησης στο αρχείο καταγραφής του Android.

Τα μηνύματα των παραθύρων διαλόγου (αλλά όχι η ειδοποίηση) μπορεί να μορφοποιηθεί χρησιμοποιώντας τις ακόλουθες ετικέτες HTML: , <big>, <blockquote>,
, <cite>, <dfn>, <div>, , <small>, , <sub>, <sup>, <tt>, <u>.

Μπορεί επίσης να χρησιμοποιηθεί η ετικέτα font για να καθοριστεί το χρώμα, για παράδειγμα, . Μερικά από τα διαθέσιμα ονόματα χρωμάτων είναι aqua, black, blue, fuchsia, green, grey, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow.

Properties

BackgroundColor

Καθορίζει το χρώμα φόντου για τις ειδοποιήσεις (όχι διαλόγους).

NotifierLength (μόνο για το designer)

Καθορίζει το χρονικό διάστημα που η ειδοποίηση θα εμφανίζεται - είτε "μικρό", είτε "μεγάλο".

TextColor

Καθορίζει το χρώμα του κειμένου για τις ειδοποιήσεις (όχι διαλόγους).

Events

AfterChoosing (επιλογή κειμένου)

Εκδηλώνεται αφού ο χρήστης έχει κάνει μια επιλογή για ShowChooseDialog.

AfterTextInput (απάντηση κειμένου)

Εκδηλώνεται αφού ο χρήστης έχει απαντήσει στο ShowTextDialog.

Methods

DismissProgressDialog ()

Απορρίπτει το προηγούμενο παράθυρο ProgressDialog που εμφανίστηκε.

LogError (γραπτό μήνυμα)

Γράφει ένα μήνυμα λάθους στο αρχείο καταγραφής συστήματος Android.

Loginfo (γραπτό μήνυμα)

Γράφει ένα ενημερωτικό μήνυμα στο αρχείο καταγραφής του Android.

LogWarning (γραπτό μήνυμα)

Γράφει ένα προειδοποιητικό μήνυμα στο αρχείο καταγραφής του Android.

ShowAlert (γραπτό σημείωμα)

Εμφανίζει μια προσωρινή κοινοποίηση.

ShowChooseDialog (μήνυμα κειμένου, τίτλος κειμένου, κείμενο button1Text, κείμενο button2Text, Boolean ακυρώσιμες)

Εμφανίζει ένα παράθυρο διαλόγου με δύο κουμπιά, από τα οποία ο χρήστης μπορεί να επιλέξει. Αν οι ακυρώσιμες τιμές είναι αληθής, θα υπάρξει ένα πρόσθετο κουμπί ακύρωσης. Πατώντας ένα κουμπί θα πραγματοποιηθεί το γεγονός AfterChoosing. Η παράμετρος "choice" στο AfterChoosing θα είναι το κείμενο στο κουμπί που πατήθηκε, ή "cancel" εάν το κουμπί ακύρωσης επιλέχθηκε.

ShowMessageDialog (μήνυμα κειμένου, τίτλος κειμένου, κείμενο ButtonText)

Εμφανίζει ένα παράθυρο ειδοποίησης με ένα μόνο κουμπί το οποίο απορρίπτει την ειδοποίηση.

ShowProgressDialog (μήνυμα κειμένου, τίτλος κειμένου)

Εμφανίζει ένα παράθυρο διαλόγου με ένα προαιρετικό τίτλο και μήνυμα. Αυτό το παράθυρο διαλόγου περιέχει μια περιστρεφόμενη μπάρα για να δείξει ότι το πρόγραμμα λειτουργεί. Δεν μπορεί να ακυρωθεί από το χρήστη αλλά πρέπει να απορριφθεί από το πρόγραμμα App Inventor χρησιμοποιώντας το μπλοκ DismissProgressDialog.

ShowTextDialog (μήνυμα κειμένου, τίτλος κειμένου, boolean ακυρώσιμες)

Εμφανίζει ένα παράθυρο διαλόγου, όπου ο χρήστης μπορεί να εισάγει κείμενο, μετά το οποίο θα πραγματοποιηθεί το γεγονός `AfterTextInput`. Αν οι ακυρώσιμες τιμές είναι αληθής, θα υπάρχει ένα πρόσθετο κουμπί ακύρωσης. Εισάγοντας κείμενο θα πραγματοποιηθεί το γεγονός `AfterTextInput`. Η «απάντηση» της παραμέτρου στο `AfterTextInput` θα είναι το κείμενο που είχε εισαχθεί, ή "cancel" εάν το κουμπί ακύρωσης επιλέχθηκε.

1.1.9 PasswordTextBox

Οι χρήστες εισάγουν τους κωδικούς πρόσβασης στο πλαίσιο κειμένου κωδικού πρόσβασης, το οποίο κρύβει το κείμενο που έχει πληκτρολογηθεί σε αυτό.

Ένα πλαίσιο κειμένου κωδικού πρόσβασης είναι το ίδιο με το συνηθισμένο πλαίσιο κειμένου, εκτός από το ότι δεν εμφανίζει τους χαρακτήρες που πληκτρολογεί ο χρήστης.

Μπορεί κανείς να πάρει ή να ορίσει την τιμή του κειμένου στο κουτί με το `Text` property. Αν το κείμενο είναι κενό, μπορεί να χρησιμοποιηθεί το `Hint` property για να παρέχει στο χρήστη μια πρόταση για το τι μπορεί να πληκτρολογήσει. Το `Hint` εμφανίζεται ως αχνό κείμενο στο πλαίσιο.

Το πλαίσιο κειμένου κωδικού πρόσβασης χρησιμοποιείται συνήθως μαζί με ένα στοιχείο κουμπιού. Ο χρήστης πατάει το κουμπί μετά την εισαγωγή κειμένου.

Methods

`RequestFocus ()`

Ορίζει το `PasswordTextBox` σε ενεργό.

Properties

`BackgroundColor`

Χρώμα για το φόντο του πλαίσιο κειμένου.

`Enabled`

Εάν οριστεί, ο χρήστης μπορεί να πληκτρολογήσει έναν κωδικό πρόσβασης στο πλαίσιο.

FontBold

Εάν οριστεί, το κείμενο εμφανίζεται με έντονους χαρακτήρες.

FontItalic

Εάν οριστεί, το κείμενο εμφανίζεται με πλάγιους χαρακτήρες.

FontSize

Μέγεθος γραμματοσειράς για το κείμενο.

FontTypeface

Είδος γραμματοσειράς για το κείμενο.

Height

Ύψος κουτιού (y-size).

Width

Πλάτος κουτιού (x-size).

TextAlignment

Στοίχιση αριστερά, στο κέντρο ή δεξιά.

TextColor

Χρώμα για το κείμενο.

Hint

Απόκρυψη κωδικού πρόσβασης.

Events

GotFocus ()

Το κουτί γίνεται το εστιασμένο στοιχείο.

LostFocus ()

Το κουτί δεν είναι πλέον το εστιασμένο στοιχείο.

1.1.10 Screen

Στοιχείο ανώτατου επιπέδου που περιέχει όλα τα άλλα στοιχεία του προγράμματος.

Properties

AboutScreen

Πληροφορίες σχετικά με την οθόνη. Εμφανίζεται όταν επιλέγεται από το μενού του συστήματος το "About this Application". Χρησιμοποιείται για να ενημερώνουμε τους χρήστες σχετικά με την εφαρμογή. Σε εφαρμογές με πολλές οθόνες, κάθε οθόνη έχει τις δικές της AboutScreen πληροφορίες.

AlignHorizontal

Ένας αριθμός που κωδικοποιεί τον τρόπο με τον οποίο τα περιεχόμενα της οθόνης ευθυγραμμίζονται οριζόντια. Οι επιλογές είναι: 1 = αριστερή στοίχιση, 2 = οριζόντια στοίχιση στο κέντρο, 3 = δεξιά στοίχιση.

AlignVertical

Ένας αριθμός που κωδικοποιεί τον τρόπο τα περιεχόμενα της οθόνης ευθυγραμμίζονται κάθετα. Οι επιλογές είναι: 1 = στοίχιση στην κορυφή, 2 = κάθετα στο κέντρο, 3 = στοίχιση στο κάτω μέρος. Η κατακόρυφη ευθυγράμμιση δεν έχει κανένα αποτέλεσμα αν η οθόνη είναι κύλισης.

BackgroundColor

AppName (μόνο για το designer)

Το εμφανιζόμενο όνομα της εγκατεστημένης εφαρμογής στο τηλέφωνο. Αν το AppName είναι κενό, ως όνομα της εφαρμογής θα οριστεί αυτό που υπήρχε όταν δημιουργήθηκε η εφαρμογή.

BackgroundImage

Η εικόνα φόντου της οθόνης.

CloseScreenAnimation

Η κίνηση για το κλείσιμο της τρέχουσας οθόνης και επιστροφής στην προηγούμενη οθόνη. Έγκυρες επιλογές είναι η προεπιλογή, το fade, το zoom, το slidehorizontal, το slidevertical, και η καμία.

Height

Ύψος οθόνης (y-size).

Icon (μόνο για το designer)

OpenScreenAnimation

Η κίνηση για τη μετάβαση σε μια άλλη οθόνη. Έγκυρες επιλογές είναι η προεπιλογή, το fade, το zoom, το slidehorizontal, το slidevertical, και η καμία.

ScreenOrientation

Ο προσανατολισμό της ζητούμενης οθόνης, ορίζεται ως τιμή κειμένου. Συνήθως χρησιμοποιούνται οι τιμές landscape, portrait, sensor, user και unspecified.

Scrollable

Όταν επιλεγεί, θα υπάρχει μια κατακόρυφη γραμμή κύλισης στην οθόνη, και το ύψος της εφαρμογής μπορεί να υπερβεί το φυσικό ύψος της συσκευής. Εάν δεν επιλεγεί, το ύψος της εφαρμογής περιορίζεται στο ύψος της συσκευής.

Title

Η λεζάντα η οποία εμφανίζεται στη γραμμή τίτλου.

VersionCode (μόνο για το designer)

Μια ακέραια τιμή που πρέπει να αυξάνεται κάθε φορά που ένα νέο Android πακέτο εφαρμογής αρχείου (APK) έχει δημιουργηθεί για το Google Play Store.

VersionName (μόνο για το designer)

Μια συμβολοσειρά που μπορεί να αλλάξει για να επιτρέπει στους χρήστες του Google Play Store να κάνουν διάκριση μεταξύ των διαφορετικών εκδόσεων του App.

Width

Πλάτος της οθόνης (x-size).

Events

BackPressed ()

Πατημένο το πίσω κουμπί της συσκευής.

ErrorOccurred (component component, text functionName, number errorNumber, text message)

Γεγονός που δημιουργείται όταν παρουσιάζεται ένα σφάλμα. Μόνο συγκεκριμένα λάθη δημιουργούν αυτό το γεγονός. Για τα σφάλματα αυτά, το σύστημα θα εμφανίσει μια προεπιλεγμένη ειδοποίηση. Μπορεί να χρησιμοποιηθεί αυτό το πρόγραμμα χειρισμού συμβάντων για να ορίσει μια λάθος συμπεριφορά διαφορετικής από αυτή της προεπιλογής.

Initialize()

Αρχική οθόνη.

OtherScreenClosed (text otherScreenName, any result)

Γεγονός που δημιουργείται όταν μια άλλη οθόνη έχει κλείσει και ο έλεγχος έχει επιστρέψει στην τρέχουσα οθόνη.

ScreenOrientationChanged ()

Αλλάζει ο προσανατολισμός της οθόνης.

Methods

Καμία

1.1.11 Slider

Το Slider είναι μια μπάρα προόδου που προσθέτει ένα συρόμενο αντίχειρα. Μπορούμε να αγγίξουμε τον αντίχειρα και να σύρουμε αριστερά ή δεξιά για να ορίσουμε τη θέση που επιθυμούμε. Καθώς ο συρόμενος αντίχειρας σέρνεται, θα προκαλέσει την εκδήλωση PositionChanged, αναφέροντας τη θέση του. Η αναφερόμενη θέση του συρόμενου αντίχειρα μπορεί να χρησιμοποιηθεί για να

ενημερώσει τη συμπεριφορά άλλου στοιχείου, όπως το μέγεθος της γραμματοσειράς ενός TextBox ή την ακτίνα της μπάλας.

Properties

ColorLeft

Το χρώμα της μπάρας αριστερά του αντίχειρα.

ColorRight

Το χρώμα της μπάρας δεξιά του αντίχειρα.

MaxValue

Ορίζει τη μέγιστη τιμή του slider. Αλλάζοντας την μέγιστη τιμή επαναφέρεται, επίσης, η θέση του αντίχειρα στα μισά της μπάρας μεταξύ της ελάχιστης και της (νέας) μέγιστης τιμής. Εάν η νέα μέγιστη τιμή είναι μικρότερη από την τρέχουσα ελάχιστη, τότε ελάχιστο και μέγιστο αμφότερα θα ρυθμιστούν σε αυτήν την τιμή. Ρυθμίζοντας το MaxValue επαναφέρουμε τη θέση του αντίχειρα στα μέσα της μπάρας μεταξύ MinValue και MaxValue και σηματοδοτούμε την εκδήλωση PositionChanged.

MinValue

Ορίζει την ελάχιστη τιμή του slider. Αλλάζοντας την ελάχιστη τιμή επαναφέρεται, επίσης, η θέση του αντίχειρα στα μισά της μπάρας μεταξύ της ελάχιστης και της (νέας) μέγιστης τιμής. Εάν η νέα ελάχιστη τιμή είναι μεγαλύτερη από την τρέχουσα μέγιστη, τότε ελάχιστο και το μέγιστο αμφότερα θα πρέπει να ρυθμιστούν σε αυτήν την τιμή. Ρυθμίζοντας το MinValue επαναφέρουμε τη θέση του αντίχειρα στα μισά της μπάρας μεταξύ MinValue και MaxValue και σηματοδοτούμε την εκδήλωση PositionChanged.

ThumbPosition

Ορίζει τη θέση του συρόμενου αντίχειρα. Εάν αυτή η τιμή είναι μεγαλύτερη από MaxValue, τότε θα πρέπει να ρυθμιστεί στην ίδια τιμή με το MaxValue. Εάν αυτή η τιμή είναι μικρότερη από MinValue, τότε θα πρέπει να οριστεί στην ίδια τιμή με το MinValue.

ThumbEnabled

Ορίζει αν θα εμφανίζεται ή όχι ο συρόμενος αντίχειρας.

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο εμφανίζεται και false αν όχι.

Width

Events

PositionChanged (number thumbPosition)

Δείχνει ότι η θέση του συρόμενου αντίχειρα έχει αλλάξει.

Methods

Καμία

1.1.12 Spinner

Το στοιχείο spinner εμφανίζει ένα αναδυόμενο παράθυρο με μια λίστα στοιχείων. Αυτά τα στοιχεία μπορούν να ρυθμιστούν στον designer ή στο blocks editor είτε ορίζοντας την ιδιότητα ElementsFromString σε μια σειρά διαχωρισμένων αλληλουχιών (για παράδειγμα, επιλογή 1, επιλογή 2, επιλογή 3), είτε ορίζοντας την ιδιότητα elements σε λίστα, στο blocks editor. Τα spinners είναι δημιουργημένα με το πρώτο στοιχείο τους να είναι ήδη επιλεγμένο. Έτσι, η επιλογή δεν δημιουργεί γεγονός After Picking. Κατά συνέπεια, είναι χρήσιμο ως πρώτο στοιχείο του Spinner να είναι μια επιλογή όπως "Επιλέξτε από τα παρακάτω ...".

Properties

Elements

Επιστρέφει μια λίστα στοιχείων κειμένου προς επιλογή.

ElementsFromString

Ορίζει τη λίστα του Spinner με τα στοιχεία διαχωρισμένα με κόμμα.

Height

Prompt

Κείμενο με τον τρέχοντα τίτλο του παραθύρου του spinner.

Selection

Επιστρέφει το τρέχον επιλεγμένο στοιχείο του spinner.

SelectionIndex

Ο δείκτης του τρέχοντος επιλεγμένου στοιχείου, ξεκινώντας από το 1. Αν δεν έχει επιλεγεί κανένα στοιχείο, η τιμή θα είναι 0.

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false εάν όχι.

Width

Events

AfterSelecting (επιλογή κειμένου)

Γεγονός που καλείται αφότου ο χρήστης επιλέξει ένα στοιχείο από τη λίστα.

Methods

DisplayDropdown ()

Εμφανίζει την πτυσσόμενη προς επιλογή λίστα, ίδια ενέργεια όπως όταν ο χρήστης κάνει κλικ στο spinner.

1.1.13 TextBox

Ο χρήστης εισάγει κείμενο σε ένα πλαίσιο κειμένου.

Η αρχική τιμή του κειμένου ή αυτή που εισάγεται από τον χρήστη σε ένα πλαίσιο κειμένου είναι στις ιδιότητες text. Αν το text είναι κενό, μπορεί να χρησιμοποιηθεί το Hint από τις ιδιότητες για να παρέχει στο χρήστη μια πρόταση του τι μπορεί να πληκτρολογήσει. Το Hint εμφανίζεται ως αχνό κείμενο στο πλαίσιο.

Η ιδιότητα `MultiLine` καθορίζει αν το κείμενο μπορεί να έχει περισσότερες από μία γραμμές. Για μία μόνο γραμμή κειμένου, το πληκτρολόγιο θα κλείσει αυτόματα όταν ο χρήστης πατήσει το πλήκτρο `Done`. Για να κλείσει το πληκτρολόγιο για πλαίσια κειμένου με πολλές γραμμές, η εφαρμογή θα πρέπει να χρησιμοποιήσει τη μέθοδο `HideKeyboard` ή να βασιστεί στον χρήστη να πατήσει το πλήκτρο πίσω.

Η ιδιότητα `NumbersOnly` περιορίζει το πληκτρολόγιο έτσι ώστε να δέχεται μόνο αριθμητικές τιμές.

Άλλες ιδιότητες επηρεάζουν την εμφάνιση του πλαισίου κειμένου (`TextAlignment`, `BackgroundColor`, κλπ) και το αν μπορεί να χρησιμοποιηθεί (`Enabled`).

Τα πλαίσια κειμένου χρησιμοποιούνται συνήθως με το στοιχείο `Button`, με το χρήστη να κάνει κλικ στο κουμπί όταν τελειώσει την εισαγωγή κειμένου.

Εάν δεν πρέπει να εμφανίζεται το κείμενο που εισάγεται από το χρήστη, μπορεί να χρησιμοποιηθεί το `PasswordTextBox` αντ' αυτού.

Methods

`HideKeyboard ()`

Απόκρυψη του πληκτρολογίου. Χρειάζεται μόνο σε πλαίσια κειμένου με πολλές γραμμές. Στα πλαίσια κειμένου με μια γραμμή το πληκτρολόγιο κλείνει όταν οι χρήστες πατήσουν το πλήκτρο `Done`.

`RequestFocus ()`

Ορίζει το πλαίσιο κειμένου σε ενεργό.

Properties

`BackgroundColor`

Το χρώμα του φόντου του πλαισίου εισαγωγής. Μπορεί να επιλεγθεί ένα χρώμα με το όνομα του είτε από το `designer`, είτε από το `blocks editor`. Το προεπιλεγμένο χρώμα φόντου είναι «default» (shaded 3-D look).

`Enabled`

Εάν ο χρήστης μπορεί να εισαγάγει κείμενο σε αυτό το πλαίσιο εισαγωγής. Από προεπιλογή, η τιμή αυτή ορίζεται σε αληθής (true).

FontBold (μόνο για το designer)

Αν η γραμματοσειρά του κείμενο θα είναι έντονη. Από προεπιλογή, δεν είναι.

FontItalic (μόνο για το designer)

Αν το κείμενο θα είναι με πλάγια γράμματα. Από προεπιλογή, δεν είναι.

FondSize

Το μέγεθος της γραμματοσειράς του κειμένου. Από προεπιλογή, είναι 14,0 πόντοι.

FontTypeface (μόνο για το designer)

Ο τύπος γραμματοσειρά για το κείμενο. Η τιμή μπορεί να αλλάξει στο Designer.

Height

Hint

Κείμενο που θα εμφανίζεται αγνά στο πλαίσιο κειμένου για να παράσχει μια υπόδειξη ως προς το τι ο χρήστης θα πρέπει να εισάγει. Αυτό μπορεί να γίνει ορατό μόνο αν η ιδιότητα στο text είναι κενή.

MultiLine

Αν η τιμή είναι αληθής, τότε το πλαίσιο κειμένου δέχεται πολλαπλές γραμμές εισόδου, οι οποίες εισάγονται χρησιμοποιώντας το πλήκτρο return. Για μια μόνο γραμμή πλαίσια κειμένου υπάρχει ένα κουμπί done αντί του πλήκτρου return, και πατώντας το κρύβετε το πληκτρολόγιο. Η εφαρμογή θα πρέπει να καλέσει τη μέθοδο HideKeyboard για να κρύψει το πληκτρολόγιο για πλαίσιο κειμένου πολλών γραμμών.

NumbersOnly

Αν η τιμή είναι αληθής, τότε αυτό το πλαίσιο κειμένου δέχεται μόνο αριθμούς ως τιμές. Οι αριθμοί μπορεί να περιλαμβάνουν ένα δεκαδικό ψηφίο και ένα προαιρετικό μείον. Αυτό ισχύει μόνο για είσοδο πληκτρολογίου. Ακόμα και αν το NumbersOnly

είναι αληθές, μπορεί να χρησιμοποιηθεί το [set Text to] για να εισάγετε οποιοδήποτε κείμενο.

Text

Το κείμενο στο πλαίσιο κειμένου, το οποίο μπορεί να ρυθμιστεί από τον προγραμματιστή στο designer ή blocks editor, ή μπορεί να εισαχθεί από το χρήστη (εκτός εάν η ιδιότητα Enabled είναι ψευδής).

TextAlignment (μόνο για το designer)

Εάν το κείμενο θα είναι ευθυγραμμισμένο αριστερά, στο κέντρο ή δεξιά. Από προεπιλογή, το κείμενο είναι στοιχισμένο αριστερά .

TextColor

Το χρώμα του κειμένου. Μπορεί να επιλεγεί ένα χρώμα με βάση το όνομά του από το designer ή το blocks editor. Το προκαθορισμένο χρώμα κειμένου είναι μαύρο.

Visible

Αν το στοιχείο είναι ορατό.

Width

Events

GotFocus ()

Το γεγονός εκδηλώνεται όταν το στοιχείο αυτό επιλεγεί για να γίνει εισαγωγή, όπως η επιλογή αυτού από τον χρήστη.

LostFocus ()

Το γεγονός εκδηλώνεται όταν αυτό το στοιχείο δεν είναι πλέον επιλεγμένο για να γίνει εισαγωγή, όπως όταν ο χρήστης αγγίζει ένα διαφορετικό πλαίσιο κειμένου.

1.1.13 TimePicker

Ένα κουμπί το οποίο, όταν επιλεγεί, ανοίγει ένα παράθυρο pop-up για να επιτρέψει στο χρήστη να επιλέξει την ώρα.

Properties

BackgroundColor

Επιστρέφει το χρώμα του φόντου του κουμπιού.

Enabled

FontBold (μόνο για το designer)

FontItalic (μόνο για το designer)

FondSize (μόνο για το designer)

FontTypeface (μόνο για το designer)

Height

Hour

Η ώρα της τελευταίας φοράς που χρησιμοποιήθηκε το time picker. Η ώρα είναι σε μορφή 24 ωρών. Αν η τελευταία καθορισμένη ώρα ήταν 23:53, αυτή η ιδιότητα θα επιστρέψει 23.

Image

Καθορίζει το path (σημείο αποθήκευσης) της εικόνας του κουμπιού. Αν υπάρχει και εικόνα και BackgroundColor, μόνο η εικόνα θα είναι ορατή.

Minute

Το λεπτό του τελευταίου χρόνου που χρησιμοποιήθηκε στο time picker.

Shape (μόνο για το designer)

Καθορίζει το σχήμα του κουμπιού (προεπιλογή, στρογγυλεμένα, ορθογώνια, οβάλ). Το σχήμα δεν θα είναι ορατό εάν υπάρχει εικόνα.

ShowFeedback

Καθορίζει αν μια οπτική ανάδραση θα εμφανίζεται για ένα κουμπί που έχει εικόνα ως φόντο.

Text

TextAlignment (μόνο για το designer)

TextColor

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false αν όχι.

Width

Events

AfterTimeSet ()

Το γεγονός αυτό εκτελείται όταν ο χρήστης έχει ορίσει την ώρα στο pop up παράθυρο διαλόγου.

GotFocus ()

Δείχνει τον κέρσορα να κινείται πάνω από το κουμπί δηλώνοντας ότι τώρα είναι δυνατό να πατηθεί.

LostFocus ()

Δείχνει τον κέρσορα να μετακινείται μακριά από το κουμπί ώστε δεν είναι πλέον δυνατόν να πατηθεί.

Methods

Καμία

1.1.14 WebViewer

Στοιχείο για την προβολή ιστοσελίδων. Το κεντρικό URL μπορεί να καθοριστεί είτε στο designer, είτε στο blocks editor. Η προβολή μπορεί να ρυθμιστεί ώστε να ακολουθούνται τα links όταν επιλεγούν, και οι χρήστες μπορούν να συμπληρώνουν Web forms.

Προειδοποίηση: Αυτό δεν είναι πλήρες πρόγραμμα περιήγησης. Για παράδειγμα, αν πατηθεί το πλήκτρο back (πίσω) του τηλεφώνου ο χρήστης θα βγει από το την

εφαρμογή, αντί να μετακινηθεί προς τα πίσω στο ιστορικό περιήγησης του προγράμματος.

Μπορεί να χρησιμοποιηθεί η ιδιότητα `WebView`. Η ιδιότητα `WebViewString` τρέχει στην σελίδα `Webviewer` έτσι ώστε να επικοινωνούν μεταξύ τους η εφαρμογή με τον κωδικό `Javascript`. Στην εφαρμογή, ορίζονται και λαμβάνονται `WebViewString`. Στο `WebView` περιλαμβάνονται `Javascript` τα οποία αναφέρονται στο `window`. Τα αντικείμενα του `AppInventor`, χρησιμοποιούν `methods` και `setWebViewString (text)`.

Για παράδειγμα, εάν ο `WebView` ανοίγει σε μια σελίδα που περιέχει την εντολή `Javascript`

```
document.write ("The answer is" + window.AppInventor.getWebViewString());
```

και αν οριστεί το `WebView.WebViewString` σε "hello", τότε η ιστοσελίδα θα δείξει

The answer is hello.

Και αν η ιστοσελίδα περιέχει `Javascript` που εκτελεί την εντολή

```
windowAppInventor.setWebViewString("hello from Javascript"),
```

τότε η τιμή της ιδιότητας `WebViewString` θα είναι

hello from Javascript.

Properties

`CurrentPageTitle`

Τίτλος της τρέχουσας σελίδας.

`CurrentUrl`

Το `URL` της σελίδας που προβάλλεται τη δεδομένη στιγμή. Αυτό θα μπορούσε να είναι διαφορετικό από την αρχική διεύθυνση `URL` (`home URL`), εάν οι σελίδες που επισκέφθηκαν ακολουθούνται από άλλα `links`.

`FollowLinks`

Καθορίζει αν θα ακολουθήσει τους συνδέσμους (`links`) όταν επιλεγούν στο `WebView`. Εάν ακολουθηθούν τα `links`, μπορεί να χρησιμοποιηθεί το `GoBack` και

GoForward για να γίνει η περιήγηση στο ιστορικό του προγράμματος περιήγησης (browser).

Height

HomeUrl

Το URL της σελίδας του που πρέπει να ανοίγει αρχικά ο WebViewer. Ρυθμίζοντας αυτό θα φορτώσει η σελίδα.

IgnoreSslError

Ορίζει εάν πρέπει ή όχι να αγνοηθούν τα σφάλματα SSL. Ορίζουμε την τιμή σε true για να αγνοηθούν τα λάθη. Χρησιμοποιούμε αυτό για να γίνουν δεχτές οι self-signed πιστοποιήσεις από ιστοσελίδες.

PromptforPermission

Εάν είναι αληθές, τότε ζητήσει από το χρήστη του WebView να δώσει άδεια για πρόσβαση στο API του γεωγραφικού εντοπισμού. Εάν είναι αναληθές, ζητείται η χορήγηση άδειας.

UsesLocation (μόνο για το designer)

Εάν θα δοθεί ή όχι η άδεια στην αίτηση χρησιμοποίησης του Javascript για το API του γεωγραφικού εντοπισμού. Αυτή η ιδιότητα είναι διαθέσιμη μόνο στο designer.

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false αν όχι.

WebViewString

Παίρνει την τιμή string του WebView, η οποία είναι ορατή μέσω του Javascript στο WebView ως window.AppInventor object

Width

Events

Κανένα

Methods

Boolean CanGoBack ()

Επιστρέφει true αν ο WebView μπορεί να πάει πίσω στη λίστα ιστορικού.

Boolean CanGoForward ()

Επιστρέφει true αν ο WebView μπορεί να πάει προς τα εμπρός στη λίστα ιστορικού.

ClearCaches ()

Καταργεί τις κρυφές μνήμες του WebView.

ClearLocations ()

Διαγράφει τις αποθηκευμένες άδειες των τοποθεσιών.

GoBack()

Επιστρέφει στην προηγούμενη σελίδα στη λίστα του ιστορικού. Αν δεν υπάρχει προηγούμενη σελίδα δεν συμβαίνει τίποτα.

GoForward ()

Πήγαινε στην επόμενη σελίδα στη λίστα ιστορικού. Αν δεν υπάρχει προηγούμενη σελίδα δεν συμβαίνει τίποτα.

GoHome()

Φορτώνει την αρχική σελίδα URL. Αυτό συμβαίνει αυτόματα όταν η αρχική διεύθυνση URL έχει αλλάξει.

GoToUrl (url κειμένου)

Φορτώνει τη σελίδα του δοθέντος URL.

1.2 Layout

1.2.1 Horizontal Arrangement

Χρησιμοποιούμε το στοιχείο της οριζόντιας διάταξης για να παρουσιάσουμε μια ομάδα στοιχείων από αριστερά προς τα δεξιά.

Αυτό το στοιχείο είναι ένα στοιχείο μορφοποίησης στο οποίο τοποθετούνται στοιχεία τα οποία πρέπει να εμφανίζονται από τα αριστερά προς τα δεξιά. Αν θέλουμε τα στοιχεία να εμφανίζονται το ένα πάνω στο άλλο, χρησιμοποιούμε το VerticalArrangement.

Σε μια HorizontalArrangement, τα στοιχεία διατάσσονται κατά μήκος του οριζόντιου άξονα, στοιχισμένα κατακόρυφα.

Εάν η ιδιότητα Height ενός HorizontalArrangement έχει οριστεί σε Αυτόματη (Automatic), το πραγματικό ύψος της διάταξης καθορίζεται από το ψηλότερο στοιχείο στη ρύθμιση του οποίου η ιδιότητα Height δεν έχει οριστεί σε Fill Parent. Εάν η ιδιότητα Height ενός HorizontalArrangement έχει οριστεί σε αυτόματη και περιέχει μόνο στοιχεία των οποίων οι ιδιότητες έχουν οριστεί σε Fill Parent, το πραγματικό ύψος της διάταξης υπολογίζεται χρησιμοποιώντας τα αυτόματα ύψη των στοιχείων. Εάν η ιδιότητα Height του HorizontalArrangement έχει οριστεί σε αυτόματη και δεν περιέχει στοιχεία, το ύψος θα είναι 100.

Εάν η ιδιότητα Width ενός HorizontalArrangement έχει οριστεί σε αυτόματη (Automatic), το πραγματικό πλάτος της διάταξης καθορίζεται από το άθροισμα του πλάτους των στοιχείων. Αν η ιδιότητα Width ενός HorizontalArrangement έχει οριστεί σε αυτόματη, οποιαδήποτε στοιχεία των οποίων η ιδιότητα Width έχει οριστεί σε Fill Parent θα συμπεριφέρονται σαν να έχει οριστεί σε αυτόματη.

Εάν η ιδιότητα Width ενός HorizontalArrangement έχει οριστεί σε Fill Parent ή στο να οριστούν τα pixels, οποιαδήποτε στοιχεία των οποίων η ιδιότητα Width έχει οριστεί σε Fill Parent, θα καταλάβει το πλάτος που δεν καταλαμβάνεται από τα υπόλοιπα στοιχεία.

Properties

Visible

Αν η τιμή είναι αληθής, το στοιχείο και το περιεχόμενό του είναι ορατό.

Height

Οριζόντια στοίχιση ύψους (y-size).

Width

Οριζόντια στοίχιση πλάτους (x-size).

1.2.2 TableArrangement

Χρησιμοποιούμε το στοιχείο διάταξης πίνακα για να παρουσιάσουμε μια ομάδα στοιχείων σε έναν πίνακα.

Αυτό το στοιχείο είναι ένα στοιχείο μορφοποίησης στο οποίο μπορούμε να τοποθετήσετε τα στοιχεία που θα πρέπει να εμφανίζονται σε μορφή πίνακα.

Σε ένα TableArrangement, τα στοιχεία διατάσσονται σε ένα πλέγμα σειρών και στηλών, με όχι περισσότερα από ένα στοιχείο ορατό σε κάθε κελί. Εάν πολλαπλά στοιχεία καταλαμβάνουν το ίδιο κελί, μόνο το τελευταίο θα είναι ορατό.

Μέσα σε κάθε σειρά, τα στοιχεία είναι στοιχισμένα κατακόρυφα.

Το πλάτος της στήλης καθορίζεται από το πλατύτερο στοιχείο της εν λόγω στήλης. Κατά τον υπολογισμό του πλάτους της στήλης, το αυτόματο πλάτος (automatic width) χρησιμοποιείται για τα στοιχεία των οποίων η ιδιότητα του πλάτους έχει οριστεί σε Fill Parent. Ωστόσο, κάθε στοιχείο θα γεμίσει πάντα ολόκληρο το πλάτος της στήλης που καταλαμβάνει.

Το ύψος της γραμμής καθορίζεται από το ψηλότερο στοιχείο σε αυτή τη γραμμή του οποίου η ιδιότητα του ύψους δεν έχει οριστεί σε Fill Parent. Εάν μια γραμμή περιέχει μόνο στοιχεία των οποίων η ιδιότητα του ύψους έχει οριστεί σε Fill Parent, το ύψος της γραμμής υπολογίζεται χρησιμοποιώντας τα αυτόματα ύψη των στοιχείων.

Properties

Visible

Αν η τιμή είναι αληθής, το στοιχείο και το περιεχόμενό του είναι ορατό.

Rows (number-of-rows)

Ο αριθμός των γραμμών του πίνακα.

Columns (number-of-columns)

Ο αριθμός των στηλών του πίνακα.

Height

Στοιχίση ύψους του πίνακα (y-size).

Width

Στοιχίση πλάτους του πίνακα (x-size).

1.2.3 VerticalArrangement

Χρησιμοποιούμε το στοιχείο της κατακόρυφης διάταξης για να παρουσιάσουμε μια ομάδα στοιχείων που ορίζονται από πάνω προς τα κάτω, αριστερά στοιχισμένα.

Αυτό το στοιχείο είναι ένα στοιχείο μορφοποίησης στο οποίο μπορούμε να τοποθετήσουμε τα στοιχεία που πρέπει να εμφανίζονται το ένα κάτω από το άλλο. Το πρώτο στοιχείο είναι αποθηκευμένο στην κορυφή, το δεύτερο κάτω από αυτό, και ούτω καθεξής. Αν θέλουμε να έχουμε τα στοιχεία που εμφανίζονται το ένα δίπλα στο άλλο, χρησιμοποιούμε το HorizontalArrangement.

Σε μια VerticalArrangement, τα στοιχεία διατάσσονται κατά μήκος του κατακόρυφου άξονα, αριστερά στοιχισμένα.

Εάν η ιδιότητα του πλάτους ενός VerticalArrangement έχει οριστεί σε αυτόματη, το πραγματικό πλάτος της στοιχίσης καθορίζεται από το στοιχείο που κατέχει το μεγαλύτερο πλάτος η ιδιότητα του οποίου δεν έχει οριστεί σε Fill Parent. Αν η ιδιότητα του πλάτους ενός VerticalArrangement έχει οριστεί σε αυτόματη και περιέχει μόνο στοιχεία των οποίων η ιδιότητα του πλάτους έχει οριστεί σε Fill Parent, το πραγματικό πλάτος της διάταξης υπολογίζεται χρησιμοποιώντας τα αυτόματα πλάτη των συνιστωσών. Αν η ιδιότητα του πλάτους ενός VerticalArrangement έχει οριστεί σε αυτόματη και αυτό δεν περιέχει στοιχεία, το πλάτος θα είναι 100.

Εάν η ιδιότητα του ύψους ενός VerticalArrangement έχει οριστεί σε αυτόματη, το πραγματικό ύψος της στοιχίσης καθορίζεται από το άθροισμα των υψών των συνιστωσών. Εάν η ιδιότητα του ύψους ενός VerticalArrangement έχει οριστεί σε αυτόματη, οποιαδήποτε στοιχεία των οποίων η ιδιότητα του ύψους έχει οριστεί σε Fill Parent θα συμπεριφέρονται σαν να είχε οριστεί σε αυτόματη.

Εάν η ιδιότητα του ύψους ενός VerticalArrangement έχει οριστεί σε Fill Parent ή σε ορισμό των pixels, οποιουδήποτε στοιχείου του οποίου η ιδιότητα του ύψους έχει οριστεί σε Fill Parent, θα λάβει το ύψος που δεν καταλαμβάνεται από τα υπόλοιπα στοιχεία.

Properties

Visible

Αν η τιμή είναι αληθής, το στοιχείο και το περιεχόμενό του είναι ορατό.

Height

Κάθετη στοίχιση ύψους (y-size).

Width

Οριζόντια στοίχιση ύψους (x-size).

1.3 Media Components

1.3.1 Camcorder

Ένα στοιχείο για την εγγραφή βίντεο χρησιμοποιώντας camcorder. Αφού γίνει η εγγραφή του βίντεο, το όνομα του αρχείου στο τηλέφωνο που περιέχει το κλιπ είναι διαθέσιμο ως argument για την εκδήλωση AfterRecording. Το όνομα του αρχείου μπορεί να χρησιμοποιηθεί, για παράδειγμα, για να οριστεί η source (πηγή) ιδιότητα ενός στοιχείου MediaPlayer.

Properties

Καμία

Events

AfterRecording(text clip)

Υποδηλώνει ότι ένα βίντεο καταγράφηκε με την κάμερα και παρέχει το path προς την αποθηκευμένη εικόνα.

Methods

RecordVideo ()

Καταγράφει ένα βίντεο, τότε ανακύπτει το γεγονός AfterRecording.

1.3.2 Camera

Χρησιμοποιούμε το στοιχείο της κάμερας για να τραβήξουμε μια φωτογραφία από το κινητό τηλέφωνο.

Η φωτογραφική μηχανή είναι ένα μη ορατό στοιχείο που αποθανατίζει μια εικόνα χρησιμοποιώντας την κάμερα της συσκευής. Μετά τη λήψη της φωτογραφίας, το path του αρχείου στο τηλέφωνο που περιέχει την εικόνα είναι διαθέσιμο ως argument για το event AfterPicture. Το path μπορεί να χρησιμοποιηθεί, για παράδειγμα, ως ιδιότητα του picture ενός στοιχείου image.

Properties

UseFront

Καθορίζει πότε πρέπει να χρησιμοποιείται η μπροστινή κάμερα (όταν είναι διαθέσιμη). Εάν η συσκευή δεν έχει κάμερα στο μπροστινό μέρος, η επιλογή αυτή θα αγνοηθεί και η κάμερα θα ανοίξει κανονικά.

Methods

TakePicture ()

Ανοίγει την κάμερα του τηλεφώνου για να μπορέσει να γίνει λήψη φωτογραφίας.

Events

AfterPicture(Text image)

Καλείται μετά από τη λήψη της φωτογραφίας. Το κείμενο του argument image είναι το path που μπορεί να χρησιμοποιηθεί για να εντοπιστεί η εικόνα στο τηλέφωνο.

1.3.3 ImagePicker

Ένα κουμπί ειδικού σκοπού. Όταν ο χρήστης επιλέγει ένα image picker, η γκαλερί εικόνων της συσκευής εμφανίζεται, και ο χρήστης μπορεί να επιλέξει μια εικόνα. Αφού επιλεγεί η εικόνα, αποθηκεύεται στην κάρτα SD και η ιδιότητα ImageFile θα πάρει το όνομα του αρχείου όπου θα αποθηκευτεί η εικόνα. Για να μην γεμίσει ο

χώρος αποθήκευσης, μπορούν να αποθηκευτούν το πολύ μέχρι 10 εικόνες. Επιλέγοντας περισσότερες από 10 εικόνες θα διαγραφούν οι προηγούμενες, με σειρά από την παλαιότερη στην νεότερη.

Properties

BackgroundColor

Επιστρέφει το χρώμα φόντου του κουμπιού.

Enabled

FontBold (μόνο για το designer)

FontItalic (μόνο για το designer)

FontSize (μόνο για το designer)

FontTypeface (μόνο για το designer)

Height

Image

Καθορίζει το path της εικόνας του κουμπιού. Αν υπάρχει και εικόνα και BackgroundColor, μόνο η εικόνα θα είναι ορατή.

Selection

Το path προς το αρχείο που περιέχει την εικόνα που επιλέχθηκε.

Shape (μόνο για το designer)

Καθορίζει το σχήμα του κουμπιού (προεπιλογή, στρογγυλεμένα, ορθογώνια, οβάλ). Το σχήμα δεν θα είναι ορατό εάν υπάρχει προβαλλόμενη εικόνα.

ShowFeedback

Καθορίζει αν μια οπτική ανάδραση θα πρέπει να εμφανίζεται για ένα κουμπί που έχει εικόνα ως φόντο.

Text

TextAlignment (μόνο για το designer)

TextColor

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false εάν όχι.

Width

Events

AfterPicking ()

Απλό γεγονός που προκαλείται αφότου η δραστηριότητα του picker επιστρέψει τα αποτελέσματα της και οι ιδιότητες έχουν συμπληρωθεί.

BeforePicking ()

Απλό γεγονός που προκαλείται όταν το στοιχείο επιλεγεί, αλλά πριν η δραστηριότητα του picker ξεκινήσει.

GotFocus ()

Δείχνει τον κέρσορα να μετακινείται πάνω στο κουμπί έτσι ώστε είναι δυνατόν να επιλεγεί.

LostFocus ()

Δείχνει τον κέρσορα να μετακινείται μακριά από το κουμπί ώστε δεν είναι πλέον δυνατόν να γίνει η επιλογή.

Methods

Open()

Ανοίγει τον picker, όταν ο χρήστης κάνει κλικ πάνω του.

1.3.4 Player

Στοιχείο πολυμέσων το οποίο παίζει ήχο και ελέγχει τη δόνηση του τηλεφώνου. Το όνομα του τομέα των πολυμέσων καθορίζεται στις ιδιότητες στο πεδίο source, το οποίο μπορεί να ρυθμιστεί είτε στο Designer, είτε στο Block Editor. Το χρονικό

διάστημα της δόνησης ορίζεται στο block editor σε χιλιοστά του δευτερολέπτου (milliseconds).

Αυτό το στοιχείο είναι καλύτερο για μεγάλα αρχεία ήχου, όπως τραγούδια, ενώ το στοιχείο sound είναι πιο αποδοτικό για μικρά αρχεία, όπως ηχητικά εφέ.

Properties

IsPlaying

Αναφέρει αν τα πολυμέσα παίζουν.

Loop

Αν η τιμή είναι αληθής, ο player θα κάνει επανάληψη (loop) όταν παίζει. Ρυθμίζοντας το Loop, ενώ ο player παίζει, θα επηρεάσει το τρέχον παίξιμο.

PlayOnlyInForeground

Αν η τιμή είναι αληθής, ο player θα σταματήσει να παίζει κατά την έξοδο από την τρέχουσα οθόνη. Αν η τιμή δεν είναι αληθής (προεπιλογή), ο player συνεχίζει να παίζει, είτε η τρέχουσα οθόνη εμφανίζει είτε όχι.

Source

Volume

Ρυθμίζει την ένταση του ήχου μεταξύ των τιμών 0 και του 100.

Events

Completed()

Υποδηλώνει ότι τα πολυμέσα έχουν ολοκληρωθεί.

OtherPlayerStarted ()

Αυτό το γεγονός εμφανίζεται όταν ένας άλλος player έχει ξεκινήσει (και ο τρέχων player παίζει ή είναι σε παύση, αλλά δεν σταμάτησε).

Methods

Pause()

Αναστέλλει την λειτουργία των πολυμέσων, αν παίζουν.

Start()

Αναπαράγει τα πολυμέσα. Εάν είχε επιλεγεί η παύση, η αναπαραγωγή επαναλαμβάνεται. Εάν είχε σταματήσει η αναπαραγωγή, ξεκινά από την αρχή.

Stop()

Σταματά την αναπαραγωγή των πολυμέσων και ξεκινά από την αρχή την αναπαραγωγή του τραγουδιού.

Vibrate(number milliseconds)

Δόνηση για ορισμένο αριθμό των χιλιοστών του δευτερολέπτου.

1.3.5 Sound

Ένα στοιχείο πολυμέσων που αναπαράγει αρχεία ήχου και προαιρετικά δονείται για χιλιοστά του δευτερολέπτου (milliseconds) καθορίζεται στο Blocks Editor. Το όνομα του αρχείου ήχου μπορεί να καθοριστεί είτε στο designer, είτε στο blocks editor.

Το στοιχείο sound είναι καλύτερο για μικρά αρχεία ήχου, όπως ηχητικά εφέ, ενώ το στοιχείο Player είναι πιο αποτελεσματικό για ήχους που διαρκούν μεγαλύτερο χρονικό διάστημα, όπως τα τραγούδια.

Properties

MinimumInterval

Το ελάχιστο διάστημα μεταξύ ήχων. Εάν αναπαράγεται ένας ήχος, όλα τα περαιτέρω Play () που έχουν κληθεί, θα πρέπει να αγνοηθούν μέχρι να παρέλθει το χρονικό διάστημα που έχει οριστεί.

Source

Το όνομα του αρχείου ήχου.

Events

Κανένα

Methods

Pause()

Διακόπτει προσωρινά την αναπαραγωγή του ήχου, που εκτελείται.

Play ()

Αναπαράγει τον ήχο.

Resume()

Συνεχίζει να παίζει τον ήχο μετά από παύση.

Stop()

Σταματά την αναπαραγωγή του ήχου που εκτελείται.

Vibrate(number millisecs)

Δόνηση για καθορισμένο αριθμό χιλιοστών του δευτερολέπτου.

1.3.6 SoundRecorder

Στοιχείο πολυμέσων που καταγράφει ήχο.

Properties

Καμία

Events

AfterSoundRecorded(text sound)

Παρέχει τη θέση του νεοσύστατου ήχου.

StartedRecording ()

Δηλώνει ότι η εγγραφή έχει ξεκινήσει, και μπορεί να διακοπεί.

StoppedRecording ()

Δηλώνει ότι η εγγραφή έχει σταματήσει, και μπορεί να ξεκινήσει και πάλι.

Methods

Start ()

Ξεκινά την εγγραφή.

Stop ()

Διακόπτει την εγγραφή.

1.3.6 SpeechRecognizer

Χρησιμοποιούμε το στοιχείο αναγνώρισης ομιλίας για να μετατρέψουμε την ομιλία του χρήστη σε κείμενο χρησιμοποιώντας τη λειτουργία αναγνώρισης ομιλίας του Android.

Properties

Result

Το τελευταίο κείμενο που παράγεται από το πρόγραμμα αναγνώρισης.

Methods

Gettext ()

Ζητά από το χρήστη να μιλήσει, και μετατρέπει την ομιλία σε κείμενο. Σηματοδοτεί το συμβάν AfterGettingText όταν το αποτέλεσμα είναι διαθέσιμο.

Events

AfterGetting(Text result)

Σηματοδοτείται αφού το πρόγραμμα αναγνώρισης παράγει το κείμενο. Το argument είναι το αποτέλεσμα του κειμένου που παρήχθη.

BeforeGettingText ()

Σηματοδοτείται λίγο πριν από το πρόγραμμα αναγνώρισης κληθεί.

1.3.7 TextToSpeech

Το στοιχείο TextToSpeech εκφωνεί το δοθέν κείμενο. Μπορεί να ρυθμιστεί ο τόνος και ο ρυθμός της ομιλίας.

Μπορεί, επίσης, να οριστεί μια γλώσσα παρέχοντας τον κώδικα της γλώσσας αυτής. Αυτό αλλάζει την προφορά των λέξεων, όχι την πραγματική γλώσσα που ομιλείται.

Για παράδειγμα, η ρύθμιση της γλώσσας είναι στα Γαλλικά, προφέροντας αγγλικό κείμενο θα ακούγεται σαν κάποιον που μιλάει Αγγλικά με γαλλική προφορά.

Μπορεί, ακόμη, να οριστεί μια χώρα με την παροχή του ταχυδρομικού κώδικα αυτής. Αυτό μπορεί να επηρεάσει την προφορά. Για παράδειγμα, τα Βρετανικά αγγλικά (GBR) θα ακούγονται διαφορετικά από τα Αγγλικά των ΗΠΑ (USA). Ο ταχυδρομικός κώδικας κάθε χώρας, δεν επηρεάζει κάθε γλώσσα.

Οι γλώσσες και οι χώρες οι οποίες είναι διαθέσιμες εξαρτώνται από τη κάθε συσκευή, και μπορούν να αναγράφονται στις ιδιότητες AvailableLanguages και AvailableCountries.

Properties

AvailableCountries

Κατάλογος των ταχυδρομικών κωδικών των χωρών που είναι διαθέσιμες σε αυτή τη συσκευή για χρήση TextToSpeech.

AvailableLanguages

Κατάλογος των γλωσσών που είναι διαθέσιμες σε αυτή τη συσκευή για χρήση TextToSpeech.

Country

Χρήση του ταχυδρομικού κώδικα της χώρας για την παραγωγή λόγου. Αυτό μπορεί να επηρεάσει την προφορά. Για παράδειγμα, Βρετανικά αγγλικά (GBR) θα ακούγονται διαφορετικά από τα Αγγλικά των ΗΠΑ (USA). Ο ταχυδρομικός κώδικας κάθε χώρας, δεν επηρεάζει κάθε γλώσσα.

Language

Ρυθμίζει τη γλώσσα για το TextToSpeech. Αυτό αλλάζει τον τρόπο που προφέρονται οι λέξεις, όχι την πραγματική γλώσσα. Για παράδειγμα, ρυθμίζουμε τη γλώσσα στα Αγγλικά, μιλάμε Αγγλικά σαν κάποιος να μιλάει αγγλικά με Γαλλική προφορά.

Pitch

Καθορίζει το βαθμό έντασης για το TextToSpeech. Οι τιμές θα πρέπει να είναι μεταξύ 0 και 2, όπου οι χαμηλότερες τιμές μειώνουν τον τόνο της φωνής που συντίθενται και οι μεγαλύτερες τιμές τον αυξάνουν.

Result

SpeechRate

Ορίζει τον ρυθμό ομιλίας για το TextToSpeech. Οι τιμές θα πρέπει να είναι μεταξύ 0 και 2, όπου οι χαμηλότερες τιμές επιβραδύνουν τον ρυθμό ομιλίας και οι μεγαλύτερες τιμές τον επιταχύνουν.

Events

AfterSpeaking(boolean result)

Γεγονός που πραγματοποιείται μετά την ολοκλήρωση του μηνύματος.

BeforeSpeaking ()

Γεγονός που πραγματοποιείται, όταν καλείται να γίνει η ομιλία, πριν το μήνυμα ειπωθεί.

Methods

Speak(text message)

Αναπαράγει το δοθέν μήνυμα.

1.3.8 VideoPlayer

Ένα στοιχείο πολυμέσων το οποίο μπορεί να αναπαράγει βίντεο. Όταν η εφαρμογή εκτελείται, το VideoPlayer θα εμφανίζεται ως ένα ορθογώνιο στην οθόνη. Εάν ο χρήστης αγγίζει το ορθογώνιο, θα εμφανιστούν οι επιλογές: αναπαραγωγή / παύση, μετάβαση εμπρός, και μετάβαση προς τα πίσω στο βίντεο. Η εφαρμογή μπορεί επίσης να ελέγξει τη συμπεριφορά καλώντας τις μεθόδους start, pause, και SeekTo.

Τα αρχεία βίντεο θα πρέπει να είναι σε μορφή 3GPP (.3gp) ή MPEG-4 (.mp4).

Το App Inventor για Android επιτρέπει μόνο αρχεία βίντεο κάτω από 1 MB και περιορίζει το συνολικό μέγεθος της εφαρμογής έως 5 MB, εκ των οποίων δεν είναι όλα διαθέσιμα για τα πολυμέσα (video, audio, and sound) των αρχείων. Αν τα αρχεία

πολυμέσων είναι πολύ μεγάλα, μπορεί υπάρχουν σφάλματα κατά το packaging ή την εγκατάσταση της εφαρμογής, σε αυτή την περίπτωση θα πρέπει να μειωθεί ο αριθμός των αρχείων πολυμέσων ή τα μεγέθη τους. Τα περισσότερα λογισμικά επεξεργασίας βίντεο, όπως το Windows Movie Maker και το Apple iMovie, μπορούν να βοηθήσουν στο να μειωθεί το μέγεθος των βίντεο είτε μικραίνοντας τα, είτε κωδικοποιώντας τα εκ νέου σε μια πιο συμπαγή μορφή.

Μπορεί, επίσης, να ρυθμιστεί η πηγή των πολυμέσων σε μια διεύθυνση URL που παραπέμπει σε ένα βίντεο συνεχούς ροής, αλλά η διεύθυνση URL πρέπει να δείχνει προς το ίδιο το αρχείο βίντεο, όχι σε ένα πρόγραμμα που αναπαράγει το βίντεο.

Properties

FullScreen

Height

Source

Το «path» για το βίντεο. Συνήθως, αυτό θα είναι το όνομα του αρχείου του βίντεο, το οποίο θα πρέπει να προστεθεί στο Designer.

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false, αν όχι.

Volume

Ρυθμίζει την ένταση του ήχου σε τιμές μεταξύ 0 και 100. Οι τιμές λιγότερο από 0, αντιμετωπίζονται ως 0, και οι τιμές πάνω από 100 αντιμετωπίζονται ως 100.

Width

Events

Completed()

Δηλώνει ότι το βίντεο έχει φτάσει στο τέλος.

Methods

number GetDuration()

Επιστρέφει τη διάρκεια του βίντεο σε χιλιοστά του δευτερολέπτου.

Pause()

Διακόπτει προσωρινά την αναπαραγωγή του βίντεο. Η αναπαραγωγή μπορεί να συνεχιστεί από το ίδιο σημείο καλώντας τη μέθοδο start.

SeekTo(number ms)

Αναζητά τον χρόνο που ζητήθηκε (ορίζεται σε χιλιοστά του δευτερολέπτου) στο βίντεο. Αν το βίντεο είναι σε παύση, το στιγμιότυπο που φαίνεται δεν θα ανανεωθεί κατά την αναζήτηση.

Start()

Ξεκινά η αναπαραγωγή του βίντεο.

1.3.9 YandexTranslate

Χρησιμοποιούμε αυτό το στοιχείο για να μεταφράσουμε λόγια και φράσεις διαφορετικών γλωσσών. Θα πρέπει να έχουμε πρόσβαση στο Διαδίκτυο, δεδομένου ότι θα ζητηθεί μετάφραση από την υπηρεσία Yandex.Translate. Καθορίζουμε την πηγή και τη γλώσσα, στο πεδίο source-target με τη χρήση δύο κωδικών γραμμμάτων της γλώσσας. Έτσι, το "en-es" θα μεταφράσει από τα αγγλικά στα ισπανικά, ενώ το "es-ru" θα μεταφράσει από τα ισπανικά στα ρώσικα. Αν δεν δηλωθεί η γλώσσα προέλευσης, η υπηρεσία θα προσπαθήσει να την ανιχνεύσει. Παρέχοντας έτσι απλά "es" θα προσπαθήσει να ανιχνεύσει τη γλώσσα προέλευσης και να τη μεταφράσει στα ισπανικά.

Σημείωση: Η μετάφραση γίνεται ασύγχρονα στο παρασκήνιο. Όταν η μετάφραση ολοκληρωθεί, το γεγονός "GotTranslation" ενεργοποιείται.

Properties

Καμία

Events

GotTranslation(text responseCode, text translation)

Το γεγονός ενεργοποιείται όταν η υπηρεσία Yandex.Translate επιστρέφει το μεταφρασμένο κείμενο. Αυτό το γεγονός παρέχει επίσης έναν κωδικό απόκρισης για την αντιμετώπιση των λαθών. Αν η responseCode δεν είναι 200, τότε κάτι πήγε στραβά με το κάλεσμα της υπηρεσίας, και η μετάφραση δεν θα είναι διαθέσιμη.

Methods

RequestTranslation(text languageToTranslateTo, text textToTranslate)

Παρέχοντας μια γλώσσα προς μετάφραση (για παράδειγμα, «es» για την ισπανικά, «en» για τα αγγλικά, ή «ru» για τα Ρώσικα), και μια λέξη ή φράση προς μετάφραση, η μέθοδος αυτή θα ζητήσει τη μετάφραση στην υπηρεσία Yandex.Translate. Μόλις το κείμενο μεταφραστεί από την υπηρεσία, το γεγονός GotTranslation θα εκτελεστεί.

Σημείωση: Το Yandex.Translate θα επιχειρήσει να ανιχνεύσει τη γλώσσα προέλευσης. Μπορεί, επίσης, να προσδιοριστεί από την αρχή η γλώσσα της μετάφρασης. Δηλαδή, es-ru θα καθορίσει την μετάφραση από τα ισπανικά στα ρώσικα.

1.4 Drawing and Animation

1.4.1 Ball

Ένα κυκλικό «Sprite», το οποίο μπορεί να τοποθετηθεί στον καμβά, όπου αντιδράει είτε με άγγιγμα, είτε με σύρσιμο, αλληλεπιδρά με άλλα sprites (ImageSprites και άλλα balls) καθώς και με τα πλαίσια του καμβά, και κινείται σύμφωνα με τις τιμές που έχουν οριστεί στις ιδιότητές του (property).

Για παράδειγμα, για να κινείται μια μπάλα 4 pixels προς την κορυφή του καμβά κάθε 500 χιλιοστά του δευτερολέπτου (μισό δευτερόλεπτο), θα πρέπει να ορίσουμε την ιδιότητα speed (ταχύτητα) σε 4 [pixels], την ιδιότητα interval (διάστημα) σε 500 [milliseconds], την ιδιότητα heading (κατεύθυνση) σε 90 μοίρες [degree], και την ιδιότητα enabled σε True. Αυτές καθώς και οι υπόλοιπες ιδιότητες μπορούν να αλλαχθούν ανά πάσα στιγμή.

Η διαφορά μεταξύ ενός ball και ενός ImageSprite είναι ότι το τελευταίο μπορεί να πάρει την μορφή ενός αρχείου εικόνας, ενώ η μορφή μιας μπάλας μπορεί να αλλάξει μόνο με τη μεταβολή των ιδιοτήτων του PaintColor (χρώμα) και της ακτίνας (radius).

Properties

Enabled

Ελέγχει πότε θα κινηθεί το sprite όταν η ταχύτητά του δεν είναι μηδενική.

Heading

Επιστρέφει την κατεύθυνση του sprite σε μοίρες πάνω στον άξονα του x. Οι αρνητικές μοίρες βρίσκονται κάτω από το δεξιό μέρος της οθόνης. Οι 90 μοίρες είναι πάνω από την κορυφή της οθόνης.

Interval

Το διάστημα σε χιλιοστά του δευτερολέπτου κατά το οποίο η θέση του sprite αλλάζει. Για παράδειγμα, εάν το διάστημα είναι 50 και η ταχύτητα είναι 10, τότε το sprite θα κινηθεί 10 pixels κάθε 50 χιλιοστά του δευτερολέπτου.

PaintColor

Radius

Speed

Η ταχύτητα με την οποία κινείται το sprite. Το Sprite κινείται πολλά pixels κάθε διάστημα.

Visible

Η τιμή είναι True αν το Sprite είναι ορατό.

X

Η οριζόντια συντεταγμένη στο αριστερό άκρο του Sprite, αυξάνεται όταν το Sprite μετακινείται προς τα δεξιά.

Y

Η κατακόρυφη συντεταγμένη στην κορυφή του Sprite, αυξάνεται όταν το Sprite κινείται προς τα κάτω.

Z

Πώς το sprite θα πρέπει να διαβαθμιστεί σε σχέση με άλλα sprites, με μεγαλύτερα αριθμητικώς επίπεδα μπροστά σε χαμηλότερα αριθμητικώς επίπεδα.

Events

CollidedWith(component other)

Χειριστής για τα γεγονότα CollidedWith, που καλείται όταν δύο sprites συγκρούονται. Σημειώνεται ότι ο έλεγχος για συγκρούσεις με ένα περιστρεφόμενο ImageSprite είναι διαθέσιμος επί του παρόντος, σε αντίθεση με την μη περιστρεφόμενη θέση του sprite. Ως εκ τούτου, ο έλεγχος σύγκρουσης θα είναι ανακριβής για ψηλά στενά ή μικρά στενά sprites που περιστρέφονται.

Dragged(number startX, number startY, number prevX, number prevY, number currentX, number currentY)

Χειριστής για dragged (συρόμενα) γεγονότα. Σε όλα τα καλέσματα (calls), οι αρχικές συντεταγμένες είναι εκεί που η οθόνη ακουμπήθηκε για πρώτη φορά, και οι «current» (τρέχουσες) συντεταγμένες περιγράφουν το τελικό σημείο του τρέχοντος σημείου της γραμμής. Στο πρώτο κάλεσμα με δοσμένο drag, οι "προηγούμενες" (previous) συντεταγμένες είναι οι ίδιες με τις συντεταγμένες εκκίνησης. Στη συνέχεια, είναι οι "τρέχουσες" (current) συντεταγμένες από το προηγούμενο κάλεσμα. Σημειώστε ότι το Sprite δεν θα μετακινηθεί πραγματικά οπουδήποτε ως απάντηση στο γεγονός που σύρθηκε, εκτός αν κληθεί το MoveTo.

EdgeReached(number edge)

Το κάλεσμα του χειριστή των γεγονότων (event handler) καλείται όταν το sprite φτάσει στην άκρη της οθόνης. Εάν αναπηδήσει τότε καλείται από αυτό το άκρο, το sprite θα εμφανιστεί για να αναπηδήσει μακριά από την άκρη στην οποία έφτασε. Η άκρη εδώ παρουσιάζεται ως ένας ακέραιος αριθμός που δείχνει μία από τις οκτώ κατευθύνσεις βόρεια (1), βορειοανατολικά (2), ανατολικά (3), νοτιοανατολικά (4), νότια (-1), νοτιοδυτικά (-2), δυτικά (-3) και βορειοδυτικά (-4).

Flung(number x, number y, number speed, number heading, number xvel, number yvel)

Όταν ένα sprite εκσφενδονίζεται (quick swipe): παρέχει τις συντεταγμένες του πετάγματος (x, y), σε σχέση με το επάνω αριστερό μέρος του καμβά. Επίσης, παρέχει την ταχύτητα (pixels ανά χιλιοστό του δευτερολέπτου) και την κατεύθυνση (0-360 μοίρες) του πετάγματος, καθώς και την ταχύτητα x και y των στοιχείων του διανύσματος του πετάγματος.

NoLongerCollidingWith(component other)

Γεγονός που δείχνει ότι ένα ζευγάρι sprites δεν συγκρούονται πλέον.

TouchDown(number x, number y)

Όταν ο χρήστης αρχίζει να αγγίζει το sprite (τοποθετήσει δάχτυλο στο sprite και το αφήσει εκεί): παρέχεται το (x, y) η θέση της αφής, σε σχέση με το επάνω αριστερό μέρος του καμβά.

TouchUp(number x, number y)

Όταν ο χρήστης σταματήσει να αγγίζει το sprite (σηκώσει το δάχτυλο μετά από ένα γεγονός touchdown): παρέχεται το (x, y) η θέση της αφής, σε σχέση με το επάνω αριστερό μέρος του καμβά.

Touched(number x, number y)

Όταν ο χρήστης αγγίζει το sprite και στη συνέχεια σηκώνει αμέσως το δάχτυλο: παρέχεται το (x, y) η θέση της αφής, σε σχέση με το επάνω αριστερό μέρος του καμβά.

Methods

Bounce(number edge)

Κάνει το sprite να αναπηδά, σαν να ήταν έξω από τον τοίχο. Για κανονική αναπήδηση, το edge argument θα πρέπει να είναι αυτό που επιστρέφεται από το EdgeReached.

Boolean CollidingWith(component other)

Υποδεικνύει εάν μια σύγκρουση έχει καταχωρηθεί μεταξύ αυτού του Sprite και του προηγούμενου.

MoveIntoBounds ()

Μετακινεί το sprite πίσω στις αναπηδήσεις εάν μέρος αυτού εκτείνεται από τα όρια, διαφορετικά δεν υπάρχει καμία επίπτωση. Αν το Sprite είναι πολύ μεγάλο για να χωρέσει στον καμβά, αυτό ευθυγραμμίζει την αριστερή πλευρά του sprite με την αριστερή πλευρά του καμβά. Αν το Sprite είναι πολύ ψηλό για να χωρέσει στον καμβά, αυτό ευθυγραμμίζει την πάνω πλευρά του Sprite με την πάνω πλευρά του καμβά.

MoveTo (number x, number y)

Μετακινεί το sprite, έτσι ώστε η πάνω αριστερή γωνία του να είναι στις συγκεκριμένες συντεταγμένες x και y.

PointInDirection(number x, number y)

Ενεργοποιεί το sprite έτσι ώστε να κατευθύνεται προς το σημείο με συντεταγμένες (x, y).

PointTowards(component target)

Ενεργοποιεί το sprite έτσι ώστε να κατευθύνεται προς ένα καθορισμένο sprite στόχο. Η νέα κατεύθυνση θα είναι παράλληλη με τη γραμμή που ενώνει τα κεντρικά σημεία των δύο sprites.

1.4.2 Canvas

Ένα ορθογώνιο πλαίσιο διδιάστατης αφής επί του οποίου το μπορούν να δημιουργηθούν σχέδια και να μετακινηθούν sprites.

Το BackgroundColor, PaintColor, BackgroundImage, width και height του καμβά μπορούν να ρυθμιστούν είτε από το designer, είτε από το blocks Editor. Το πλάτος και το ύψος μετράται σε pixel και πρέπει να έχουν θετική τιμή.

Κάθε θέση στον καμβά μπορεί να οριστεί ως ένα ζεύγος τιμών (X, Y), όπου

- X είναι ο αριθμός των pixels από το αριστερό άκρο του καμβά.
- Y είναι ο αριθμός των pixels από το άνω άκρο του καμβά.

Υπάρχουν γεγονότα για να δηλώσουν πότε και πού ένας καμβάς έχει ακουμπηθεί ή ένα Sprite (ImageSprite or ball) έχει συρθεί. Υπάρχουν επίσης μέθοδοι για να σχεδιάζονται σημεία, γραμμές και κύκλοι.

Properties

BackgroundColor

Το χρώμα του φόντου του καμβά.

BackgroundImage

Το όνομα του αρχείου που περιέχει την εικόνα φόντου του καμβά.

FontSize

Το μέγεθος της γραμματοσειράς του κειμένου που αποτυπώνεται στον καμβά.

Height

LineWidth

Το πλάτος των γραμμών που αποτυπώνεται στον καμβά.

PaintColor

Το χρώμα με το οποίο δημιουργούνται οι γραμμές.

TextAlignment

Καθορίζει τη στοίχιση του κειμένου που έχει δημιουργηθεί από το DrawText () ή το DrawAngle () σε σχέση με το σημείο που ορίζεται από την εν λόγω εντολή.

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο είναι ορατό και false αν όχι.

Width

Events

Dragged(number startX, number startY, number prevX, number prevY, number currentX, number currentY, boolean draggedSprite)

Όταν ο χρήστης σέρνει από ένα σημείο (PrevX, prevY) σε ένα άλλο (x, y). Το ζεύγος (startX, startY) δείχνει σε ποιο σημείο ο χρήστης άγγιξε πρώτα την οθόνη, και το "draggedSprite" δείχνει αν ένα sprite έχει συρθεί.

Flung(number x, number y, number speed, number heading, number xvel, number yvel, boolean flungSprite)

Όταν μια εκσφενδόνιση (quick swipe) γίνεται πάνω στον καμβά: παρέχεται το (x, y) η θέση της έναρξης, σε σχέση με το επάνω αριστερό μέρος του καμβά. Επίσης, παρέχεται η ταχύτητα (pixels ανά χιλιοστό του δευτερολέπτου) και η κατεύθυνση (0-360 μοίρες) της, καθώς και η ταχύτητα x και y των στοιχείων του διανύσματος του πετάγματος. Η τιμή "flungSprite" είναι αληθής, αν ένα sprite βρισκόταν κοντά στο σημείο της αφετηρίας του πετάγματος.

TouchDown(number x, number y)

Όταν ο χρήστης αρχίζει να αγγίζει τον καμβά (τοποθετεί το δάχτυλο στον καμβά και το αφήνει εκεί): παρέχεται το (x, y) η θέση του αγγίγματος, σε σχέση με το επάνω αριστερό μέρος του καμβά.

TouchUp(number x, number y)

Όταν ο χρήστης σταματήσει να αγγίζει τον καμβά (σηκώσει το δάχτυλο μετά από ένα γεγονός touchdown): παρέχεται το (x, y) η θέση του αγγίγματος, σε σχέση με το επάνω αριστερό μέρος του καμβά.

Touched(number x, number y, boolean touchedSprite)

Όταν ο χρήστης αγγίζει τον καμβά και, στη συνέχεια, αμέσως σηκώσει το δάχτυλο: παρέχεται το (x, y) η θέση του αγγίγματος, σε σχέση με το επάνω αριστερό μέρος του καμβά. Το TouchedSprite είναι αληθές, αν το ίδιο άγγιγμα άγγιξε ταυτόχρονα ένα sprite, και ψευδές διαφορετικά.

Methods

Clear ()

Καθαρίζει οτιδήποτε έχει ζωγραφιστεί στον καμβά, αλλά όχι το χρώμα του φόντου ή τις εικόνες που μπορεί να υπάρχουν στον καμβά.

`DrawCircle(number x, number y, number r)`

Σχεδιάζει ένα κύκλο (fill in) με τις δοθείσες συντεταγμένες και ακτίνα πάνω στον καμβά.

`DrawLine(number x1, number y1, number x2, number y2)`

Σχεδιάζει μια γραμμή μεταξύ των συγκεκριμένων συντεταγμένων πάνω στον καμβά.

`DrawPoint(number x, number y)`

Σχεδιάζει ένα σημείο στις δοθείσες συντεταγμένες πάνω στον καμβά.

`DrawText(text text, number x, number y)`

Σχεδιάζει το καθορισμένο κείμενο στις καθορισμένες συντεταγμένες χρησιμοποιώντας από τις ιδιότητες τις τιμές των `Font Size` και `TextAlignment`.

`DrawTextAtAngle(text text, number x, number y, number angle)`

Σχεδιάζει το συγκεκριμένο κείμενο από το σημείο όπου θα οριστεί ως σημείο εκκίνησης στις καθορισμένες συντεταγμένες, με καθορισμένη γωνία χρησιμοποιώντας τις τιμές των ιδιοτήτων `Fond Size` και `TextAlignment`.

`number GetBackgroundPixelColor(number x, number y)`

Παίρνει το χρώμα από ένα καθορισμένο σημείο. Αυτό περιλαμβάνει το χρώμα του φόντου και οποιαδήποτε σχεδιασμένα σημεία, γραμμές ή κύκλους, αλλά όχι sprites.

`number GetPixelColor(number x, number y)`

Παίρνει το χρώμα του καθορισμένου σημείου.

`text Save()`

Αποθηκεύει μια εικόνα του καμβά σε μια εξωτερική μονάδα αποθήκευσης. Αν συμβεί κάποιο σφάλμα, καλείται η οθόνη του `ErrorOccurred`.

`text SaveAs(text fileName)`

Αποθηκεύει μια εικόνα του καμβά σε μια συσκευή εξωτερικής αποθήκευσης σε ένα αρχείο με το όνομα filename. Το όνομα του αρχείου πρέπει να τελειώνει με μία από τις παρακάτω καταλήξεις .jpg, .jpeg, .png, οι οποίες καθορίζουν τον τύπο αρχείου.

SetBackgroundPixelColor(number x, number y, number color)

Ορίζει το χρώμα του συγκεκριμένου σημείου. Αυτό διαφέρει από το DrawPoint έχοντας ένα argument για το χρώμα.

1.4.3 ImageSprite

Ένα «Sprite», το οποίο μπορεί να τοποθετηθεί σε ένα καμβά, όπου μπορεί να αντιδρά σε πινελιές και συρσίματα, να αλληλεπιδρά με άλλα sprites (balls και άλλα ImageSprites) καθώς και με την άκρη του καμβά, και να κινείται σύμφωνα με τις τιμές που δίνονται στις ιδιότητες (properties). Η εμφάνισή του ορίζεται από την ιδιότητα picture (εκτός αν ιδιότητα Visible είναι ψευδής).

Για να έχουμε μια κίνηση ImageSprite 10 pixel προς τα αριστερά κάθε 1.000 χιλιοστά του δευτερολέπτου (ένα δευτερόλεπτο), για παράδειγμα, θα πρέπει να οριστεί η ιδιότητα speed (ταχύτητα) σε 10 [pixels], η ιδιότητα interval (διάστημα) 1000 [χιλιοστά του δευτερολέπτου], η ιδιότητα της κατεύθυνσης σε 180 [μοίρες] , και η ιδιότητα Enabled να πάρει την τιμή True. Ένα sprite του οποίου η ιδιότητα rotates είναι αληθής, θα περιστρέφει την εικόνα μέχρις ότου αλλάξει η κατεύθυνση του sprite. Ο έλεγχος για συγκρούσεις με έναν περιστρεφόμενο ImageSprite ελέγχεται επί του παρόντος, σε αντίθεση με τα sprites που έχουν μη περιστρεφόμενη θέση. Ως εκ τούτου, ο έλεγχος σύγκρουσης θα είναι ανακριβής για ψηλά στενά ή μικρού εύρους sprites που περιστρέφονται. Οποιαδήποτε από τις ιδιότητες του Sprite μπορεί να αλλάξει ανά πάσα στιγμή από τον έλεγχο του προγράμματος.

Properties

Enabled

Ελέγχει πότε ένα sprite κινείται, όταν η ταχύτητά του δεν είναι μηδενική.

Heading

Επιστρέφει την κατεύθυνση του sprite σε μοίρες σε σχέση με τον άξονα των x. Οι μηδέν βαθμοί είναι προς τα δεξιά της οθόνης. Οι 90 βαθμοί είναι προς την κορυφή της οθόνης.

Height

Interval

Το διάστημα σε χιλιοστά του δευτερολέπτου κατά το οποίο η θέση του sprite αλλάζει. Για παράδειγμα, εάν το διάστημα είναι 50 και η ταχύτητα είναι 10, τότε το sprite θα κινηθεί 10 pixels κάθε 50 χιλιοστά του δευτερολέπτου.

Picture

Η εικόνα που καθορίζει την εμφάνισή του sprite.

Rotates

Αν η τιμή είναι αληθής, η εικόνα sprite περιστρέφεται ώστε να ταιριάζει με την κατεύθυνση του sprite. Αν είναι false, η εικόνα sprite δεν περιστρέφεται όταν το sprite αλλάξει κατεύθυνση. Το Sprite περιστρέφεται γύρω από κεντρικό σημείο του.

Speed

Η ταχύτητα με την οποία κινείται το sprite. Το Sprite κινείται πολλά pixels ανά διάστημα.

Visible

Η τιμή είναι true όταν το Sprite είναι ορατό.

Width

X

Η οριζόντια συντεταγμένη στο αριστερό άκρο του Sprite, αυξάνεται όσο το Sprite μετακινείται προς τα δεξιά.

Y

Η κατακόρυφη συντεταγμένη στην κορυφή του Sprite, αυξάνεται καθώς το Sprite μετακινείται προς τα κάτω.

Z

Πώς το sprite θα πρέπει να τοποθετείται σε σχέση με τα υπόλοιπα sprites, με τα μεγαλύτερα αριθμητικώς στρώματα να τοποθετούνται μπροστά από αυτά με τα λιγότερα αριθμητικώς στρώματα.

Events

CollidedWith(component other)

Ο χειριστής για CollidedWith γεγονότα, καλείται όταν δυο sprites συγκρούονται. Σημειώνεται ότι ο έλεγχος για συγκρούσεις με έναν περιστρεφόμενο ImageSprite ελέγχεται επί του παρόντος, σε αντίθεση με τα sprites που έχουν μη περιστρεφόμενη θέση. Ως εκ τούτου, ο έλεγχος σύγκρουσης θα είναι ανακριβής για ψηλά στενά ή μικρού εύρους sprites που περιστρέφονται.

Dragged(number startX, number startY, number prevX, number prevY, number currentX, number currentY)

Διαχειριστής για συρόμενα γεγονότα. Σε κάθε κάλεσμα της εντολής, οι αρχικές συντεταγμένες είναι όπου η οθόνη για πρώτη φορά ακουμπήθηκε, και οι "τρέχουσες" (current) συντεταγμένες περιγράφουν το τελικό σημείο του τρέχοντος τμήματος γραμμής. Στο πρώτο κάλεσμα με δοσμένο drag (σύρσιμο), οι "προηγούμενες" (previous) συντεταγμένες είναι οι ίδιες με τις συντεταγμένες εκκίνησης. Στη συνέχεια, είναι οι "τρέχουσες" (current) συντεταγμένες από το προηγούμενο κάλεσμα. Σημειώστε ότι το Sprite δεν θα μετακινηθεί πραγματικά οπουδήποτε ως απάντηση στο γεγονός που σύρθηκε, εκτός αν κληθεί το MoveTo.

EdgeReached(number edge)

Ο χειρισμός συμβάντων καλείται όταν το sprite φτάνει σε μια άκρη της οθόνης. Εάν αναπηδήσει στη συνέχεια καλείται από αυτό το άκρο, έπειτα το sprite θα εμφανιστεί για να αναπηδήσει μακριά από το άκρο στο οποίο έφτασε. Η άκρη εδώ παρουσιάζεται ως ένας ακέραιος αριθμός που δείχνει μία από τις οκτώ κατευθύνσεις βόρεια (1), βορειοανατολικά (2), ανατολικά (3), νοτιοανατολικά (4), νότια (-1), νοτιοδυτικά (-2), δυτικά (-3) και βορειοδυτικά (-4).

Flung(number x, number y, number speed, number heading, number xvel, number yvel)

Όταν ένα sprite εκσφενδονίζεται (quick swipe): παρέχει τις συντεταγμένες του πετάγματος (x, y), σε σχέση με το επάνω αριστερό μέρος του καμβά. Επίσης, παρέχει την ταχύτητα (pixels ανά χιλιοστό του δευτερολέπτου) και την κατεύθυνση (0-360 μοίρες) του πετάγματος, καθώς και την ταχύτητα x και y των στοιχείων του διανύσματος του πετάγματος.

NoLongerCollidingWith(component other)

Γεγονός που δείχνει ότι δυο sprites δεν συγκρούονται πλέον.

TouchDown(number x, number y)

Όταν ο χρήστης αρχίζει να αγγίζει το sprite (τοποθετεί το δάχτυλο στο sprite και το αφήνει εκεί): παρέχεται το (x, y) η θέση του αγγίγματος, σε σχέση με το επάνω αριστερό μέρος του καμβά.

TouchUp(number x, number y)

Όταν ο χρήστης σταματήσει να αγγίζει το sprite (σηκώσει το δάχτυλο μετά από ένα γεγονός touchdown): παρέχεται το (x, y) η θέση του αγγίγματος, σε σχέση με το επάνω αριστερό μέρος του καμβά.

Touched(number x, number y)

Όταν ο χρήστης αγγίζει το sprite και, στη συνέχεια, αμέσως σηκώσει το δάχτυλο: παρέχεται το (x, y) η θέση του αγγίγματος, σε σχέση με το επάνω αριστερό μέρος του καμβά.

Methods

Bounce(number edge)

Κάνει το sprite να αναπηδά, σαν να ήταν έξω από τον τοίχο. Για κανονική αναπήδηση, το edge argument θα πρέπει να είναι αυτό που επιστρέφεται από το EdgeReached.

Boolean CollidingWith(component other)

Υποδεικνύει εάν μια σύγκρουση έχει καταχωρηθεί μεταξύ αυτού του Sprite και του προηγούμενου.

MoveIntoBounds ()

Μετακινεί το sprite πίσω στις αναπηδήσεις εάν μέρος αυτού εκτείνεται από τα όρια, διαφορετικά δεν υπάρχει καμία επίπτωση. Αν το Sprite είναι πολύ μεγάλο για να χωρέσει στον καμβά, αυτό ευθυγραμμίζει την αριστερή πλευρά του sprite με την αριστερή πλευρά του καμβά. Αν το Sprite είναι πολύ ψηλό για να χωρέσει στον καμβά, αυτό ευθυγραμμίζει την πάνω πλευρά του Sprite με την πάνω πλευρά του καμβά.

MoveTo (number x, number y)

Μετακινεί το sprite, έτσι ώστε η πάνω αριστερή γωνία του να είναι στις συγκεκριμένες συντεταγμένες x και y.

PointInDirection(number x, number y)

Ενεργοποιεί το sprite έτσι ώστε να κατευθύνεται προς το σημείο με συντεταγμένες (x, y).

PointTowards(component target)

Ενεργοποιεί το sprite έτσι ώστε να κατευθύνεται προς ένα καθορισμένο sprite στόχο. Η νέα κατεύθυνση θα είναι παράλληλη με τη γραμμή που ενώνει τα κεντρικά σημεία των δύο sprites.

1.5 Sensor Components

1.5.1 Accelerometer Sensor

Μη ορατό στοιχείο που μπορεί να ανιχνεύσει το κούνημα και να μετρήσει την επιτάχυνση περίπου σε τρεις διαστάσεις, χρησιμοποιώντας μονάδες SI (m/s²). Τα στοιχεία είναι:

- xAccel: 0, όταν το τηλέφωνο είναι σε κατάσταση ηρεμίας σε επίπεδη επιφάνεια, θετική όταν το τηλέφωνο έχει κλίση προς τα δεξιά (δηλαδή, η αριστερή του πλευρά υψώνεται), και αρνητική όταν το τηλέφωνο έχει κλίση προς τα αριστερά (δηλαδή, η δεξιά του πλευρά υψώνεται).

- yAccel: 0, όταν το τηλέφωνο είναι σε κατάσταση ηρεμίας σε μια επίπεδη επιφάνεια, θετική όταν το κάτω μέρος του υψώνεται, και αρνητική όταν το πάνω μέρος του υψώνεται.
- zAccel: Ίση με -9.8 (βαρύτητας της Γης σε μέτρα ανά δευτερόλεπτο, όταν η συσκευή βρίσκεται σε κατάσταση ηρεμίας παράλληλα με το έδαφος και την οθόνη προς τα επάνω, 0 όταν είναι κάθετα στο έδαφος, και 9.8 όταν είναι στραμμένη προς τα κάτω. Η τιμή μπορεί επίσης να επηρεαστεί επιταχύνοντάς το με ή ενάντια στη βαρύτητα.

Properties

Available

Enabled

MinimumInterval

Το ελάχιστο διάστημα μεταξύ των κουνημάτων του τηλεφώνου.

Sensitivity

Ένας αριθμός που υποδηλώνει πόσο ευαίσθητο είναι το accelerometer. Οι επιλογές είναι: 1 = αδύναμο, 2 = μέτριο, 3 = ισχυρό.

XAccel

YAccel

ZAccel

Events

AccelerationChanged(number xAccel, number yAccel, number zAccel)

Δείχνει την αλλαγή της επιτάχυνσης στις διαστάσεις X, Y, και / ή Z.

Shaking()

Δηλώνει ότι η συσκευή άρχισε να κουνιέται ή συνεχίζει να κουνιέται.

Methods

Καμία

1.5.2 Barcode Scanner

Στοιχείο που χρησιμοποιεί τον σαρωτή των barcodes για να διαβάσει ένα barcode.

Properties

Result

Αποτελέσματα με την μορφή κειμένου που προκύπτουν από την σάρωση.

UseExternalScanner

Εάν η τιμή είναι αληθής, το App Inventor θα αναζητήσει προς χρήση ένα εξωτερικό πρόγραμμα σάρωσης, όπως "Bar Code Scanner."

Events

AfterScan(text result)

Υποδεικνύει ότι ο σαρωτής έχει διαβάσει ένα (κείμενο) αποτέλεσμα και παρέχει το αποτέλεσμα.

Methods

DoScan ()

Ξεκινά μια σάρωση barcode, χρησιμοποιώντας την κάμερα. Όταν ολοκληρωθεί η σάρωση, θα πραγματοποιηθεί το γεγονός AfterScan.

1.5.3 Clock

Μη ορατό στοιχείο που παρέχει τη χρονική στιγμή χρησιμοποιώντας το εσωτερικό ρολόι του τηλεφώνου. Μπορεί να θέτει ένα χρονόμετρο ανά τακτά χρονικά διαστήματα και να εκτελεί τους υπολογισμούς του χρόνου, χειρισμούς και μετατροπές. Επίσης, υπάρχουν διαθέσιμες μέθοδοι για την μορφοποίηση της ημερομηνίας και της ώρας.

Properties

TimerAlwaysFires

Θα ενεργοποιηθεί ακόμα και όταν η εφαρμογή δεν εμφανίζεται στην οθόνη.

TimerEnabled

Εάν η τιμή είναι αληθής, ξεκινάει το χρονόμετρο.

TimerInterval

Διάστημα μεταξύ των χρονικών γεγονότων σε ms.

Events

Timer ()

Το χρονόμετρο έχει σταματήσει.

Methods

InstantInTime AddDays(InstantInTime instant, number days)

Μια στιγμή στο χρόνο μερικές ημέρες μετά το όρισμα.

InstantInTime AddHours(InstantInTime instant, number hours)

Μια στιγμή στο χρόνο μερικές ώρες μετά το όρισμα.

InstantInTime AddMinutes(InstantInTime instant, number minutes)

Μια στιγμή στο χρόνο λίγα λεπτά μετά το όρισμα.

InstantInTime AddMonths(InstantInTime instant, number months)

Μια στιγμή στο χρόνο μερικούς μήνες μετά το όρισμα.

InstantInTime AddSeconds(InstantInTime instant, number seconds)

Μια στιγμή στο χρόνο μερικά δευτερολέπτα μετά το όρισμα.

InstantInTime AddWeeks(InstantInTime instant, number weeks)

Μια στιγμή στο χρόνο μερικές εβδομάδες μετά το όρισμα.

InstantInTime AddYears(InstantInTime instant, number years)

Μια χρονική στιγμή κάποια χρόνια μετά το όρισμα.

number DayOfMonth(InstantInTime instant)

Η ημέρα του μήνα.

number Duration(InstantInTime start, InstantInTime end)

Χιλιοστά του δευτερολέπτου τα οποία μεσολάβησαν μεταξύ στιγμών.

text FormatDate(InstantInTime instant)

Κείμενο που αντιπροσωπεύει την ημερομηνία της στιγμής.

text FormatDateTime(InstantInTime instant)

Κείμενο που αντιπροσωπεύει την ημερομηνία και την ώρα της στιγμής.

text FormatTime(InstantInTime instant)

Κείμενο που αντιπροσωπεύει το χρόνο της στιγμής.

number GetMillis(InstantInTime instant)

Η στιγμή στο χρόνο μετράται ως χιλιοστά του δευτερολέπτου από το 1970.

number Hour(InstantInTime instant)

Η ώρα της ημέρας.

InstantInTime MakeInstant(text from)

Μια στιγμή καθορίζεται από MM/DD/YYYY hh:mm:ss ή MM/DD/YYYY ή hh:mm.

InstantInTime MakeInstantFromMillis(number millis)

Η στιγμή στο χρόνο που καθορίζεται από χιλιοστά του δευτερολέπτου από το 1970.

number Minute(InstantInTime instant)

Το λεπτό της ώρας.

number Month(InstantInTime instant)

Οι μήνες του χρόνου παριστάνονται ως ένας αριθμοί από το 1 έως το 12.

text MonthName(InstantInTime instant)

Το όνομα του μήνα.

`InstantInTime Now()`

Η τρέχουσα χρονική στιγμή που διαβάζεται από το ρολόι του τηλεφώνου.

`number Second(InstantInTime instant)`

Το δευτερόλεπτο του λεπτού.

`number SystemTime()`

Η εσωτερική ώρα του τηλεφώνου.

`number Weekday(InstantInTime instant)`

Η ημέρα της εβδομάδας παριστάνεται με αριθμούς από το 1 (Κυριακή) έως το 7 (Σάββατο).

`text WeekdayName(InstantInTime instant)`

Το όνομα της ημέρας της εβδομάδας.

`number Year(InstantInTime instant)`

Το έτος.

1.5.4 LocationSensor

Μη ορατό στοιχείο που παρέχει πληροφορίες για τη θέση, συμπεριλαμβανομένων γεωγραφικού μήκους, πλάτους, ύψους (εάν υποστηρίζεται από τη συσκευή), και διεύθυνσης. Αυτό μπορεί επίσης να εκτελέσει "γεωκωδικοποίηση» (geocoding), μετατρέποντας μια συγκεκριμένη διεύθυνση (όχι απαραίτητα την τρέχουσα) σε γεωγραφικό πλάτος (με τη μέθοδο `LatitudeFromAddress`) και γεωγραφικό μήκος (με τη μέθοδο `LongitudeFromAddress`).

Για να λειτουργήσει, το στοιχείο θα πρέπει να έχει θέσει την ιδιότητα `enabled` σε `True`, και η συσκευή να έχει τον αισθητήρα θέσης ενεργοποιημένο μέσω ασύρματων δικτύων ή δορυφόρων GPS (εάν βρίσκεται σε εξωτερικό χώρο).

Οι πληροφορίες τοποθεσίας μπορεί να μην είναι άμεσα διαθέσιμες όταν ξεκινήσει η εφαρμογή. Θα πρέπει να περιμένουμε ένα μικρό χρονικό διάστημα έτσι ώστε να

βρεθεί και να χρησιμοποιηθεί ένας πάροχος τοποθεσίας (location provider), ή να περιμένουμε για το γεγονός OnLocationChanged.

Properties

Accuracy

Altitude

AvailableProviders

CurrentAddress

DistanceInterval

Καθορίζει το ελάχιστο διάστημα απόστασης, σε μέτρα, όπου ο αισθητήρας θα προσπαθήσει να χρησιμοποιήσει για την αποστολή ενημερώσεων τοποθεσίας. Για παράδειγμα, αν αυτό έχει οριστεί σε 5, τότε ο αισθητήρας θα εκκινήσει ένα γεγονός LocationChanged μόνο όταν τα 5 μέτρα έχουν παρέλθει. Ωστόσο, ο αισθητήρας δεν εγγυάται ότι η ενημέρωση θα γίνει ακριβώς σε αυτό το διάστημα. Για παράδειγμα, μπορεί να χρειαστεί περισσότερο από 5 μέτρα για την εκκίνηση ενός γεγονότος.

Enabled

HasAccuracy

HasAltitude

HasLongitudeLatitude

Latitude

Longitude

ProviderLocked

ProviderName

TimeInterval

Καθορίζει τον ελάχιστο χρονικό διάστημα, σε χιλιοστά του δευτερολέπτου, όπου ο αισθητήρας θα προσπαθήσει να χρησιμοποιηθεί για την αποστολή ενημερώσεων

τοποθεσίας. Ωστόσο, οι ενημερώσεις τοποθεσίας θα λάβουν χώρα μόνο όταν η θέση του τηλεφώνου αλλάξει πραγματικά, και η χρήση του καθορισμένου χρονικού διαστήματος δεν είναι εγγυημένη. Για παράδειγμα, εάν το 1000 χρησιμοποιηθεί ως χρονικό διάστημα, οι ενημερώσεις τοποθεσίας δεν πρόκειται ποτέ να ενεργοποιηθούν νωρίτερα από τα 1000ms, αλλά μπορούν να ενεργοποιηθούν ανά πάσα στιγμή μετά.

Events

LocationChanged(number latitude, number longitude, number altitude)

Υποδεικνύει ότι μια νέα θέση έχει εντοπιστεί.

StatusChanged(text provider, text status)

Δηλώνει ότι η κατάσταση της υπηρεσίας παρόχου τοποθεσίας έχει αλλάξει, όπως όταν ένας πάροχος έχει χαθεί ή ένας νέος πάροχος αρχίζει να χρησιμοποιείται.

Methods

number LatitudeFromAddress(text locationName)

Απορρέει από το γεωγραφικό πλάτος της δοθείσας διεύθυνσης.

number LongitudeFromAddress(text locationName)

Απορρέει από το γεωγραφικό μήκος της δοθείσας διεύθυνσης.

1.5.5 NearField

Μη ορατό στοιχείο το οποίο παρέχει NFC (near field communication) δυνατότητες. Προς το παρόν το στοιχείο αυτό στηρίζει μόνο την ανάγνωση και τη γραφή κειμένου ετικετών κειμένου (εάν υποστηρίζεται από τη συσκευή).

Για να διαβάσει και να γράφει ετικέτες κειμένου, το στοιχείο πρέπει να έχει την ιδιότητα ReadMode σε true ή false, αντίστοιχα.

Properties

LastMessage

ReadMode

TextToWrite

WriteType

Events

TagRead(text message)

Υποδεικνύει ότι μια νέα ετικέτα έχει ανιχνευθεί. Επί του παρόντος, αυτό είναι μόνο μια απλή ετικέτα κειμένου.

TagWritten ()

Γεγονός για TagWritten.

Methods

Καμία

1.5.6 Orientation Sensor

Χρησιμοποιούμε ένα στοιχείο αισθητήρα προσανατολισμού για να προσδιορίσουμε τον χωρικό προσανατολισμός του τηλεφώνου.

Ένας αισθητήρας προσανατολισμού είναι ένα μη ορατό στοιχείο που αναφέρει τις ακόλουθες τρεις τιμές, σε μοίρες:

- Roll: 0 μοίρες όταν η συσκευή είναι σε επίπεδο, αυξάνεται σε 90 μοίρες, όταν η συσκευή έχει κλίση προς την επάνω αριστερή πλευρά, και μειώνεται σε -90 μοίρες όταν η συσκευή έχει κλίση προς την επάνω δεξιά πλευρά.
- Pitch: 0 μοίρες όταν η συσκευή είναι σε επίπεδο, αυξάνεται σε 90 μοίρες, όταν η συσκευή έχει κλίση έτσι ώστε η κορυφή της να είναι προς τα κάτω, και μειώνεται στις 0 μοίρες όταν η συσκευή αναποδογυρίσει. Παρομοίως, όταν η συσκευή έχει κλίση τέτοια ώστε το κάτω μέρος της να είναι στραμμένο προς τα κάτω, μειώνεται σε -90 μοίρες, στη συνέχεια, αυξάνεται σε 0 μοίρες όταν η συσκευή αναποδογυρίσει.
- Azimuth: 0 μοίρες, όταν η κορυφή της συσκευής δείχνει βόρεια, 90 μοίρες όταν είναι στραμμένη ανατολικά, 180 μοίρες όταν είναι στραμμένη νότια, 270 μοίρες όταν είναι στραμμένη δυτικά, κ.λπ.

Αυτές οι μετρήσεις προϋποθέτουν ότι η ίδια η συσκευή δεν κινείται.

Properties

Available

Δηλώνει πότε ο αισθητήρας προσανατολισμού παρουσιάζεται στη συσκευή Android.

Enabled

Εάν οριστεί, ο αισθητήρας προσανατολισμού είναι ενεργοποιημένος.

Azimuth

Επιστρέφει την αζιμούθια γωνία της συσκευής.

Pitch

Επιστρέφει την γωνία κλίσης της συσκευής.

Roll

Επιστρέφει την γωνία περιστροφής της συσκευής.

Magnitude

Επιστρέφει έναν αριθμό μεταξύ 0 και 1 που δείχνει την κλίση της συσκευής. Δίνει το μέγεθος της δύναμης που θα γίνει αισθητή από ένα ρολάρισμα μιας μπάλας στην επιφάνεια της συσκευής.

Angle

Επιστρέφει μια γωνία που λέει την κατεύθυνση προς την οποία η συσκευή έχει κλίση. Δηλαδή, μας δίνει την κατεύθυνση της δύναμης που θα γίνει αισθητή από ρολάρισμα μιας μπάλας στην επιφάνεια της συσκευής.

Events

OrientationChanged(number azimuth, number pitch, number roll)

Καλείται όταν ο προσανατολισμός έχει αλλάξει.

1.5.6 ProximitySensor

Ένα στοιχείο αισθητήρα που μπορεί να μετρήσει την εγγύτητα ενός αντικειμένου (σε εκατοστά) σε σχέση με την οθόνη προβολής μιας συσκευής. Αυτός ο αισθητήρας χρησιμοποιείται συνήθως για να προσδιοριστεί αν ένα ακουστικό κρατείται στο αυτί

ενός ανθρώπου, δηλαδή επιτρέπει τον καθορισμό του πόσο μακριά είναι ένα αντικείμενο από μια συσκευή. Πολλές συσκευές επιστρέφουν την απόλυτη απόσταση, σε cm, αλλά κάποια επιστρέφουν μόνο τις τιμές κοντά και μακριά. Στην περίπτωση αυτή, ο αισθητήρας αναφέρει συνήθως τη μέγιστη τιμή του εύρους της σε μακρινές αποστάσεις και την μικρότερη τιμή σε κοντινές αποστάσεις. Αναφέρει την ακόλουθη τιμή:

Distance: Η απόσταση του αντικείμενου από τη συσκευή.

Properties

Available

Αναφέρει εάν η συσκευή διαθέτει ή όχι αισθητήρα εγγύτητας.

Enabled

Αν είναι ενεργοποιημένο, τότε η συσκευή θα ακούγεται με τις αλλαγές στην εγγύτητα.

KeepRunningWhenOnPause

Αν πάρει την τιμή true, θα κρατήσει τον αισθητήρα για την αλλαγή της εγγύτητας ακόμα και όταν η εφαρμογή δεν είναι ορατή.

Distance

Επιστρέφει την απόσταση του αντικειμένου από τη συσκευή.

MaximumRange

Αναφέρει το μέγιστο εύρος του ProximitySensor της συσκευής.

Events

ProximityChanged(number distance)

Καλείται όταν η απόσταση (σε εκατοστά) του αντικειμένου προς τη συσκευή, αλλάξει.

1.6 Social Components

1.6.1 ContactPicker

Ένα κουμπί που, όταν το επιλέξουμε, εμφανίζει μια λίστα με τις επαφές ανάμεσα των οποίων μπορούμε να επιλέξουμε. Αφού ο χρήστης κάνει την επιλογή, θα πρέπει να οριστούν οι παρακάτω ιδιότητες, ώστε να μας δώσουν πληροφορίες σχετικά με την επιλεγμένη επαφή:

- **ContactName**: το όνομα της επαφής.
- **EmailAddress**: η κύρια διεύθυνση ηλεκτρονικού ταχυδρομείου της επαφής.
- **EmailAddressList**: μια λίστα με τις διευθύνσεις ηλεκτρονικού ταχυδρομείου της επαφής.
- **PhoneNumber**: ο κύριος αριθμός τηλεφώνου της επαφής (σε αργότερη έκδοση Android).
- **PhoneNumberList**: μια λίστα των αριθμών τηλεφώνου της επαφής (σε αργότερη έκδοση Android).
- **Image**: το όνομα του αρχείου το οποίο περιέχει την εικόνα της επαφής και μπορεί να χρησιμοποιηθεί ως τιμή στην ιδιότητα του `picture` για το στοιχείο `image` ή `ImageSprite`.

Άλλες ιδιότητες επηρεάζουν και την εμφάνιση του κουμπιού (`TextAlignment`, `BackgroundColor`, κλπ) και το πότε αυτό θα είναι διαθέσιμο προς επιλογή (`Enabled`).

Το `ContactPicker` δεν μπορεί να λειτουργήσει σε όλα τα τηλέφωνα. Για παράδειγμα, σε συστήματα Android πριν την έκδοση 3.0, δεν μπορεί να πάρει τους αριθμούς τηλεφώνου και η λίστα με τις διευθύνσεις ηλεκτρονικού ταχυδρομείου θα περιέχει μόνο ένα e-mail.

Properties

BackgroundColor

Επιστρέφει το χρώμα του φόντου του κουμπιού.

ContactName

EmailAddress

Enabled

FontBold (μόνο για το designer)

FontItalic (μόνο για το designer)

FontSize (μόνο για το designer)

FontTypeface (μόνο για το designer)

Height

Image

Καθορίζει τη διαδρομή της εικόνας του κουμπιού. Αν υπάρχει και εικόνα και BackgroundColor, μόνο η εικόνα θα είναι ορατή.

Picture

Shape (μόνο για το designer)

Καθορίζει το σχήμα του κουμπιού (προεπιλογή, στρογγυλεμένα, ορθογώνια, οβάλ). Το σχήμα δεν θα είναι ορατό εάν εμφανίζεται εικόνα.

ShowFeedback

Καθορίζει αν ένα οπτικό feedback θα πρέπει να εμφανίζεται για ένα κουμπί που έχει εικόνα ως φόντο.

Text

TextAlignment (μόνο για το designer)

TextColor

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο φαίνεται και false αν όχι.

Width

Events

AfterPicking()

Απλό γεγονός που εμφανίζεται αφότου η δραστηριότητα του picker επιστρέψει τα αποτελέσματά της και οι ιδιότητες έχουν συμπληρωθεί.

BeforePicking ()

Απλό γεγονός που εμφανίζεται όταν το στοιχείο επιλεγεί, αλλά πριν η δραστηριότητα του picker ξεκινήσει.

GotFocus ()

Δείχνει τον κέρσορα να μετακινείται πάνω στο κουμπί έτσι ώστε να είναι δυνατή η επιλογή του.

LostFocus ()

Δείχνει τον κέρσορα να μετακινείται μακριά από το κουμπί ώστε πλέον δεν είναι δυνατή η επιλογή του.

Methods

Open()

Ανοίγει τον picker, όταν ο χρήστης κάνει κλικ πάνω του.

1.6.2 EmailPicker

Ένα EmailPicker είναι ένα είδος πλαισίου κειμένου. Εάν ο χρήστης αρχίζει να πληκτρολογεί το όνομα ή τη διεύθυνση e-mail μιας επαφής, το τηλέφωνο θα εμφανίσει ένα πτυσσόμενο μενού επιλογών που συμπληρώνει τα στοιχεία που εισάγονται. Εάν υπάρχουν πολλές επαφές, η πτυσσόμενη λίστα μπορεί να χρειαστεί μερικά δευτερόλεπτα για να εμφανιστεί και να δείξει τα ενδιάμεσα αποτελέσματα, ενώ τα αποτελέσματα υπολογίζονται.

Τα αρχικά περιεχόμενα του πλαισίου κειμένου και τα περιεχόμενα μετά την εισαγωγή του χρήστη, εμφανίζονται στην ιδιότητα του text. Αν η ιδιότητα του text είναι αρχικά κενή, τα περιεχόμενα της ιδιότητας του hint θα φαίνονται αχνά στο πλαίσιο κειμένου ως υπόδειξη προς το χρήστη.

Άλλες ιδιότητες επηρεάζουν την εμφάνιση του πλαισίου κειμένου (TextAlignment, BackgroundColor, κλπ) και το κατά πόσο αυτό μπορεί να χρησιμοποιηθεί (Enabled).

Τα πλαίσια κειμένου σαν αυτό χρησιμοποιούνται συνήθως με στοιχείο κουμπιού (Button), με το χρήστη να πατάει το κουμπί όταν εισάγει το τελικό του κείμενο.

Methods

RequestFocus()

Θέτει το EmailPicker ενεργό.

Properties

BackgroundColor

Το χρώμα του φόντου του πλαισίου εισαγωγής. Μπορεί ο χρήστης να επιλέξει ένα χρώμα με το όνομα του είτε στο designer, είτε στο blocks editor. Το προεπιλεγμένο χρώμα φόντου είναι «default» (shaded 3-D look).

Enabled

Πότε ο χρήστης μπορεί να εισάγει κείμενο στο πλαίσιο εισαγωγής. Από προεπιλογή, η τιμή είναι αληθής.

FontBold (μόνο για το designer)

Αν η γραμματοσειρά του κειμένου θα είναι έντονη. Από προεπιλογή, δεν είναι.

FontItalic (μόνο για το designer)

Πότε το κείμενο θα εμφανίζεται με πλάγια γράμματα. Από προεπιλογή, δεν είναι ενεργοποιημένο.

FontSize

Το μέγεθος της γραμματοσειράς για το κείμενο. Από προεπιλογή, είναι 14.0 πόντοι.

FontTypeface (μόνο για το designer)

Η γραμματοσειρά του κειμένου. Η τιμή μπορεί να αλλάξει στο Designer.

Height

Hint

Κείμενο που θα εμφανίζεται αχνά στο πλαίσιο εισαγωγής για να παράσχει μια υπόδειξη ως προς το τι ο χρήστης θα πρέπει να εισάγει. Αυτό μπορεί να διαπιστωθεί μόνο αν η ιδιότητα `property` είναι κενή.

Text

Το κείμενο στο πλαίσιο εισαγωγής, το οποίο μπορεί να ρυθμιστεί είτε από τον προγραμματιστή στο `designer` ή `Blocks Editor`, είτε μπορεί να εισαχθεί από το χρήστη (εκτός εάν η ιδιότητα `Enabled` είναι ψευδής).

TextAlignment (μόνο για το `designer`)

Πότε το κείμενο θα πρέπει να είναι στοιχισμένο αριστερά, στο κέντρο ή δεξιά. Από προεπιλογή, το κείμενο είναι στοιχισμένο αριστερά.

TextColor

Το χρώμα του κειμένου. Μπορείτε να επιλέξετε ένα χρώμα με το όνομά στο `designer` ή στο `Blocks Editor`. Το προκαθορισμένο χρώμα κειμένου είναι το μαύρο.

Visible

Καθορίζει αν το στοιχείο θα είναι ορατό στην οθόνη. Η τιμή είναι `true` αν το στοιχείο φαίνεται και `false` αν όχι.

Width

Events

GotFocus ()

Το γεγονός ξεκινάει όταν επιλέγεται το πλαίσιο για να γίνει εισαγωγή του κειμένου, π.χ. όταν ο χρήστης το αγγίξει.

LostFocus ()

Το γεγονός ξεκινάει όταν αυτό το στοιχείο δεν είναι πλέον επιλεγμένο για την εισαγωγή κειμένου, π.χ. ο χρήστης αγγίζει ένα διαφορετικό πλαίσιο κειμένου.

Methods

Καμία

1.6.3 Phone Call

Ένα μη-ορατό στοιχείο που κάνει ένα τηλεφώνημα στον αριθμό που καθορίζεται στην ιδιότητα `PhoneNumber`, η οποία μπορεί να ρυθμιστεί είτε στο `designer`, είτε στο `Blocks Editor`. Το στοιχείο διαθέτει μια μέθοδο `MakePhoneCall`, διευκολύνοντας το πρόγραμμα να ξεκινήσει μια τηλεφωνική κλήση.

Συχνά, αυτό το στοιχείο χρησιμοποιείται με το στοιχείο `ContactPicker`, επιτρέποντας έτσι στον χρήστη να επιλέξει μια επαφή από αυτές που είναι αποθηκευμένες στο τηλέφωνο και να ορίσει στην ιδιότητα `PhoneNumber` τον αριθμό τηλεφώνου της επαφής.

Για να οριστεί άμεσα ο τηλεφωνικός αριθμός (π.χ. 650-555-1212), πρέπει να ρυθμιστεί η ιδιότητα `PhoneNumber` σε `Text` με καθορισμένα ψηφία (π.χ. "6505551212"). Μπορούν να συμπεριληφθούν παύλες, τελείες και παρενθέσεις (π.χ. "(650)-555-1212»), αλλά μπορούν να αγνοηθούν. Κενά δεν μπορούν να συμπεριληφθούν.

Properties

`PhoneNumber`

Events

`IncomingCallAnswered` (text `phoneNumber`)

Γεγονός που δείχνει ότι μια εισερχόμενη κλήση έχει απαντηθεί. Το `phoneNumber` δηλώνει τον εισερχόμενο τηλεφωνικό αριθμό κλήσης.

`PhoneCallEnded` (number status, text `phoneNumber`)

Γεγονός που δείχνει ότι μια τηλεφωνική κλήση έχει λήξει. Αν η κατάσταση είναι 1, η εισερχόμενη κλήση δεν απαντήθηκε ή απορρίφθηκε. Αν η κατάσταση είναι 2, η εισερχόμενη κλήση απαντήθηκε πριν κλείσει το τηλέφωνο. Αν η κατάσταση είναι 3, η εξερχόμενη κλήση τελείωσε. Το `phoneNumber` δηλώνει τον αριθμό της κλήσης που τερματίστηκε.

`PhoneCallStarted` (number status, text `phoneNumber`)

Γεγονός που δείχνει ότι ένα τηλεφώνημα έχει αρχίσει. Αν η κατάσταση είναι 1, χτυπά εισερχόμενη κλήση. Εάν η κατάσταση είναι 2, πραγματοποιείται εξερχόμενη κλήση. Το `phoneNumber` δηλώνει τον αριθμό τηλεφώνου της εισερχόμενης ή εξερχόμενης κλήσης.

Methods

`MakePhoneCall ()`

Κάνει ένα τηλεφώνημα χρησιμοποιώντας τον αριθμό της ιδιότητας `PhoneNumber`.

1.6.4 `PhoneNumberPicker`

Ένα κουμπί που, όταν πατηθεί, εμφανίζει μια λίστα με τους αριθμούς των επαφών του τηλεφώνου για να επιλέξει ο χρήστης. Αφού ο χρήστης έχει κάνει μια επιλογή, θα πρέπει να οριστούν οι ακόλουθες ιδιότητες για να δώσουν πληροφορίες σχετικά με την επιλεγμένη επαφή:

- `ContactName`: το όνομα της επαφής.
- `PhoneNumber`: ο αριθμός τηλεφώνου της επαφής.
- `EmailAddress`: η διεύθυνση ηλεκτρονικού ταχυδρομείου της επαφής.
- `Picture`: το όνομα του αρχείου που περιέχει την εικόνα της επαφής, η οποία μπορεί να χρησιμοποιηθεί ως τιμή στην ιδιότητα `property` για τα στοιχεία `Image` ή `ImageSprite`.

Άλλες ιδιότητες επηρεάζουν την εμφάνιση του κουμπιού (`TextAlignment`, `BackgroundColor`, κλπ) και αν μπορεί αυτό να επιλεγεί (`Enabled`).

Το στοιχείο `PhoneNumberPicker` μπορεί να μην λειτουργεί σε όλες τις συσκευές Android. Για παράδειγμα, σε συστήματα Android πριν την έκδοση 3.0, οι κατάλογοι με τους αριθμούς τηλεφώνου και τις διευθύνσεις e-mail που θα επιστραφούν θα είναι άδειοι.

Properties

`BackgroundColor`

Επιστρέφει το χρώμα του φόντου του κουμπιού.

`ContactName`

EmailAddress

Enabled

FontBold (μόνο για το designer)

FontItalic (μόνο για το designer)

FontSize (μόνο για το designer)

FontTypeface (μόνο για το designer)

Height

Image

Καθορίζει τη διαδρομή (path) της εικόνας του κουμπιού. Αν υπάρχει και εικόνα και BackgroundColor, μόνο η εικόνα θα είναι ορατή.

PhoneNumber

Picture

Shape (μόνο για το designer)

Καθορίζει το σχήμα του κουμπιού (προεπιλογή, στρογγυλεμένα, ορθογώνια, οβάλ). Το σχήμα δεν θα είναι ορατό εάν υπάρχει εικόνα.

ShowFeedback

Καθορίζει αν μια οπτική ανάδραση θα πρέπει να εμφανίζεται για ένα κουμπί που θα έχει εικόνα ως φόντο.

Text

TextAlignment (μόνο για το designer)

TextColor

Visible

Καθορίζει αν το στοιχείο θα πρέπει να είναι ορατό στην οθόνη. Η τιμή είναι true αν το στοιχείο φαίνεται και false, εάν όχι.

Width

Events

AfterPicking()

Απλό γεγονός που λαμβάνει χώρα αφού η δραστηριότητα συλλογής (picker activity) επιστρέψει τα αποτελέσματα και οι ιδιότητες έχουν συμπληρωθεί.

BeforePicking ()

Απλό γεγονός που λαμβάνει χώρα όταν η συνιστώσα επιλεγεί, αλλά πριν ξεκινήσει η δραστηριότητα συλλογής.

GotFocus ()

Δείχνει τον κέρσορα να μετακινείται πάνω στο κουμπί έτσι ώστε να είναι δυνατή η επιλογή του.

LostFocus ()

Δείχνει τον κέρσορα μετακινείται μακριά από το κουμπί έτσι ώστε δεν είναι πλέον δυνατή η επιλογή του.

Methods

Open()

Ανοίγει τον επιλογέα (picker), όταν ο χρήστης κάνει κλικ πάνω του.

1.6.5 Sharing

Το μοίρασμα (sharing) είναι ένα μη ορατό στοιχείο που επιτρέπει την κοινή χρήση αρχείων ή/ και μηνυμάτων μεταξύ εφαρμογών του χρήστη, αλλά και άλλων εφαρμογών που είναι εγκατεστημένες στη συσκευή. Το στοιχείο θα εμφανίζει μια λίστα με τις εγκατεστημένες εφαρμογές εκ των οποίων μπορεί να χειριστεί τις πληροφορίες που παρέχονται, και επιτρέπει στο χρήστη να επιλέξει κάποιον με τον οποίο θα μοιραστεί το περιεχόμενο, για παράδειγμα, μια εφαρμογή ηλεκτρονικού ταχυδρομείου, ένα application κοινωνικού δικτύου, ένα application μηνυμάτων, και ούτω καθεξής.

Η διαδρομή του αρχείου μπορεί να ληφθεί απευθείας από τα άλλα στοιχεία, όπως η φωτογραφική μηχανή ή το ImagePicker, αλλά μπορεί επίσης να καθοριστεί άμεσα από το σημείο αποθήκευσης. Θα πρέπει να τονιστεί ότι κάθε συσκευή χειρίζεται με διαφορετικό τρόπο την αποθήκευση, έτσι μερικά πράγματα που πρέπει να δοκιμαστούν εάν, για παράδειγμα, υπάρχει αρχείο που ονομάζεται arrow.gif στο φάκελο Appinventor/assets, θα είναι:

- "file:///sdcard/Appinventor/assets/arrow.gif"

ή

- "/storage/Appinventor/assets/arrow.gif"

Properties

Καμία

Events

Κανένα

Methods

ShareFile(text file)

Μοιράζεται ένα αρχείο με οποιαδήποτε ικανή εφαρμογή εγκατεστημένη στο τηλέφωνο, εμφανίζοντας μια λίστα με τις διαθέσιμες εφαρμογές, επιτρέποντας στο χρήστη να επιλέξει μια από τη λίστα. Η επιλεγμένη εφαρμογή θα ανοίξει με το αρχείο να έχει εισαχθεί σε αυτό.

ShareFileWithMessage(text file, text message)

Μοιράζεται και αρχεία και μηνύματα μέσω οποιασδήποτε ικανής εφαρμογής εγκατεστημένης στο τηλέφωνο, εμφανίζοντας μια λίστα με τις διαθέσιμες εφαρμογές και επιτρέπει στο χρήστη να επιλέξει μια από τη λίστα. Η επιλεγμένη εφαρμογή θα ανοίξει με το αρχείο να έχει εισαχθεί σε αυτό.

ShareMessage(text message)

Μοιράζεται ένα μήνυμα μέσω οποιασδήποτε ικανής εφαρμογής εγκατεστημένης στο τηλέφωνο, εμφανίζοντας μια λίστα με τις διαθέσιμες εφαρμογές επιτρέποντας στο

χρήστη να επιλέξει ένα από τη λίστα. Η επιλεγμένη εφαρμογή θα ανοίξει με το αρχείο να έχει εισαχθεί σε αυτό.

1.6.6 Texting

Ένα στοιχείο που, όταν η μέθοδος `SendMessage` καλείται, στέλνει το μήνυμα κειμένου που ορίζεται στην ιδιότητα `message`, στον αριθμό τηλεφώνου που καθορίζεται στην ιδιότητα `PhoneNumber`.

Εάν η ιδιότητα `ReceivingEnabled` ρυθμιστεί σε 1, τα μηνύματα δεν θα ληφθούν. Αν η `ReceivingEnabled` έχει οριστεί σε 2, τα μηνύματα θα λαμβάνονται μόνο όταν η εφαρμογή εκτελείται. Τέλος, αν η `ReceivingEnabled` έχει οριστεί σε 3, τα μηνύματα θα λαμβάνονται όταν η εφαρμογή εκτελείται και όταν η εφαρμογή δεν είναι ενεργή θα είναι στην ουρά και μια ειδοποίηση θα εμφανίζεται στο χρήστη.

Όταν ένα μήνυμα ληφθεί, το γεγονός `MessageReceived` ξεκινάει και παρέχει τον αριθμό αποστολής και το μήνυμα.

Μια εφαρμογή που περιλαμβάνει αυτό το στοιχείο, θα λαμβάνει μηνύματα, ακόμη και όταν είναι στο παρασκήνιο (δηλαδή όταν δεν είναι ορατό στην οθόνη) και ακόμη και αν η εφαρμογή δεν εκτελείται, εφ' όσον είναι εγκατεστημένη στο τηλέφωνο. Αν το τηλέφωνο λάβει ένα μήνυμα κειμένου όταν η εφαρμογή δεν είναι σε πρώτο πλάνο, το τηλέφωνο θα εμφανίσει μια ειδοποίηση στη γραμμή ειδοποιήσεων. Επιλέγοντας την ειδοποίηση θα εμφανιστεί η εφαρμογή.

Εάν η ιδιότητα `GoogleVoiceEnabled` είναι αληθής, τα μηνύματα μπορούν να σταλούν μέσω Wi-Fi χρησιμοποιώντας το Google Voice. Η επιλογή αυτή απαιτεί ο χρήστης να έχει ένα λογαριασμό στο Google Voice και ότι το Voice app έχει εγκατασταθεί στο τηλέφωνο. Η επιλογή του Google Voice λειτουργεί μόνο σε τηλέφωνα που υποστηρίζουν Android 2.0 (Eclair) και πάνω.

Για να καθοριστεί ο αριθμός τηλεφώνου (π.χ. 650-555-1212), πρέπει να ρυθμιστεί η ιδιότητα `PhoneNumber` σε μια συμβολοσειρά κειμένου με καθορισμένα ψηφία (π.χ. 6505551212). Παύλες, τελείες και παρενθέσεις μπορούν να συμπεριληφθούν (π.χ. (650) -555 - 1212), αλλά θα αγνοηθούν. Κενά δεν μπορούν να συμπεριληφθούν.

Ένας άλλος τρόπος για μια εφαρμογή να καθορίσει έναν αριθμό τηλεφώνου είναι να περιλαμβάνει ένα στοιχείο `PhoneNumberPicker`, το οποίο επιτρέπει στους χρήστες να

επιλέξουν έναν αριθμό τηλεφώνου από αυτούς που είναι αποθηκευμένοι στις επαφές του τηλεφώνου.

Properties

GoogleVoiceEnabled

Αν η τιμή είναι αληθής, τότε το SendMessage θα επιχειρήσει να στείλει μηνύματα μέσω WiFi χρησιμοποιώντας το Google Voice. Αυτό προϋποθέτει ότι το Google Voice app είναι εγκατεστημένο στο τηλέφωνο ή το tablet, με ένα λογαριασμό Google Voice. Αν η τιμή GoogleVoiceEnabled είναι ψευδής, η συσκευή πρέπει να διαθέτει υπηρεσίες τηλεφώνου και γραπτών μηνυμάτων για να στείλει ή να λάβει μηνύματα με αυτό το στοιχείο.

Message

Το μήνυμα που θα σταλεί όταν η μέθοδος SendMessage κληθεί.

PhoneNumber

Ο αριθμός που το μήνυμα θα αποσταλεί, όταν η μέθοδος SendMessage κληθεί. Ο αριθμός είναι μια συμβολοσειρά κειμένου με καθορισμένες ψηφία (π.χ. 6505551212). Παύλες, τελείες και παρενθέσεις μπορούν να συμπεριληφθούν (π.χ., (650) -555 - 1212), αλλά θα αγνοηθούν. Κενά δεν πρέπει να περιλαμβάνονται.

ReceivingEnabled

Αν οριστεί σε 1 (OFF) δεν μπορούν να ληφθούν μηνύματα. Αν οριστεί σε 2 (FOREGROUND) ή σε 3 (ALWAYS) το στοιχείο θα απαντά σε μηνύματα εάν η εφαρμογή βρίσκεται σε λειτουργία. Αν η εφαρμογή δεν εκτελείται τότε το μήνυμα θα πρέπει να απορρίπτεται αν οριστεί στο 2 (FOREGROUND). Αν οριστεί σε 3 (ALWAYS) και η εφαρμογή δεν λειτουργεί το τηλέφωνο θα εμφανίσει μια ειδοποίηση. Επιλέγοντας την ειδοποίηση, θα φέρει μπροστά την εφαρμογή την εκδήλωση MessageReceived. Τα μηνύματα που λαμβάνονται όταν η εφαρμογή είναι αδρανής θα είναι στην ουρά, και έτσι πολλά MessageReceived γεγονότα μπορεί να εμφανιστούν όταν η εφαρμογή ξεκινήσει.

Events

MessageReceived(text number, text messageText)

Γεγονός που ξεκινάει όταν ένα μήνυμα κειμένου ληφθεί από το τηλέφωνο.

Methods

SendMessage()

Στέλνει ένα μήνυμα κειμένου.

1.6.7 Twitter

Ένα μη ορατό στοιχείο που επιτρέπει την επικοινωνία με το Twitter. Μόλις ο χρήστης συνδεθεί στο λογαριασμό του στο Twitter (και η άδεια έχει επιβεβαιωθεί επιτυχώς από το γεγονός IsAuthorized), πολλές περισσότερες λειτουργίες είναι διαθέσιμες:

- Ψάχνει στο Twitter για tweets ή ετικέτες (SearchTwitter).
- Στέλνει tweet (Tweet).
- Στέλνει tweet με εικόνα (TweetWithImage).
- Κατευθύνει ένα μήνυμα σε συγκεκριμένο χρήστη (DirectMessage).
- Λαμβάνει τα πιο πρόσφατα μηνύματα που απευθύνονται στον συνδεδεμένο χρήστη (RequestDirectMessages).
- Ακολουθεί ένα συγκεκριμένο χρήστη (follow).
- Παύει να ακολουθεί ένα συγκεκριμένο χρήστη (StopFollowing).
- Παίρνει μια λίστα χρηστών που ακολουθούν τον συνδεδεμένο χρήστη (RequestFollowers).
- Παίρνει τα πιο πρόσφατα μηνύματα από χρήστες που ακολουθούνται από τον συνδεδεμένο χρήστη (RequestFriendTimeline).
- Παίρνει την πιο πρόσφατη αναφορά του συνδεδεμένου χρήστη (RequestMentions).

Ο χρήστης θα πρέπει να προμηθευτεί ένα consumer key και ένα Consumer Secret για το Twitter, εξουσιοδοτούμενο ειδικά για την συγκεκριμένη εφαρμογή από το http://twitter.com/oauth_clients/new.

Properties

ConsumerKey

Το consumer key χρησιμοποιείται όταν εγκρίνεται από το Twitter μέσω του OAuth.

ConsumerSecret

Το consumer secret χρησιμοποιείται όταν εγκρίνεται από το Twitter μέσω του OAuth.

DirectMessages

Η ιδιότητα περιλαμβάνει μια λίστα με τα πιο πρόσφατα μηνύματα αναφέροντας τους συνδεδεμένους χρήστες. Αρχικά, η λίστα είναι κενή. Για να οριστεί, το πρόγραμμα πρέπει να:

1. καλέσει τη μέθοδο Authorize,
2. περιμένει για το γεγονός Authorized,
3. καλέσει τη μέθοδο RequestDirectMessages,
4. περιμένει για το γεγονός DirectMessagesReceived.

Η τιμή αυτής της ιδιότητας στη συνέχεια θα οριστεί στον κατάλογο των άμεσων μηνυμάτων που ανακτώνται (και θα παραμείνει αυτή η τιμή μέχρι την επόμενη κλήση του RequestDirectMessages).

Followers

Αυτή η ιδιότητα περιέχει μια λίστα των ακολούθων (followers) του συνδεδεμένου χρήστη. Αρχικά, η λίστα είναι κενή. Για να οριστεί, το πρόγραμμα πρέπει να:

1. καλέσει τη μέθοδο Authorize,
2. περιμένει για το γεγονός IsAuthorized,
3. καλέσει τη μέθοδο RequestFollowers,
4. περιμένει για την εκδήλωση FollowersReceived.

Η τιμή αυτής της ιδιότητας στη συνέχεια θα οριστεί στον κατάλογο των ακολούθων (followers) (και θα διατηρήσει την τιμή του μέχρι την επόμενη κλήση του RequestFollowers).

FriendTimeline

Αυτή η ιδιότητα περιλαμβάνει τα 20 πιο πρόσφατα μηνύματα από χρήστες που ακολουθήθηκαν. Αρχικά, η λίστα είναι κενή. Για να οριστεί, το πρόγραμμα πρέπει να:

1. καλέσει τη μέθοδο `Authorize`,
2. περιμένει το γεγονός `IsAuthorized`,
3. καθορίσει τους χρήστες να ακολουθούν με μία ή περισσότερες κλήσεις τη μέθοδο `follow`,
4. καλέσει τη μέθοδο `RequestFriendTimeline`,
5. περιμένει το γεγονός `FriendTimelineReceived`.

Η τιμή αυτής της ιδιότητας στη συνέχεια θα οριστεί στη λίστα των μηνυμάτων (και θα διατηρήσει την τιμή της μέχρι την επόμενη κλήση του `RequestFriendTimeline`).

Mentions

Αυτή η ιδιότητα περιέχει μια λίστα με αναφορές του συνδεδεμένου χρήστη. Αρχικά, η λίστα είναι κενή. Για να οριστεί, το πρόγραμμα πρέπει να:

1. καλέσει τη μέθοδο `Authorize`,
2. περιμένει το γεγονός `IsAuthorized`,
3. καλέσει τη μέθοδο `RequestMentions`,
4. περιμένει το γεγονός `MentionsReceived`.

Η τιμή αυτής της ιδιότητας στη συνέχεια θα οριστεί ο κατάλογος αναφορών (και θα διατηρήσει την τιμή του μέχρι τις επόμενες κλήσεις του `RequestMentions`).

SearchResults

Αυτή η ιδιότητα, η οποία αρχικά είναι κενή, ορίζει μια λίστα αποτελεσμάτων αναζήτησης αφού το πρόγραμμα:

1. καλέσει τη μέθοδο `SearchTwitter`,
2. περιμένει για το γεγονός `SearchSuccessful`.

Η τιμή της ιδιότητας τότε θα είναι η ίδια με την παράμετρο του `SearchSuccessful`. Σημειώνεται, ότι δεν είναι απαραίτητο να κληθεί η μέθοδος `Authorize` πριν κληθεί το `SearchTwitter`.

Username

Το όνομα χρήστη του εξουσιοδοτημένου χρήστη. Είναι κενό αν δεν υπάρχει εξουσιοδοτημένος χρήστης.

Events

DirectMessagesReceived(list messages)

Αυτό το συμβάν προκύπτει, όταν τα τελευταία μηνύματα που ζητήθηκαν μέσω του RequestDirectMessages έχουν ανακτηθεί. Μια λίστα με τα μηνύματα μπορεί στη συνέχεια να βρεθεί στην παράμετρο messages ή στην ιδιότητα messages.

FollowersReceived(list followers2)

Αυτό το γεγονός προκύπτει, όταν το σύνολο των ακολούθων του συνδεδεμένου χρήστη που ζητείται μέσω του RequestFollowers έχει ανακτηθεί. Ο κατάλογος των ακολούθων μπορεί στη συνέχεια να βρεθεί στην παράμετρο followers ή στην ιδιότητα followers.

FriendTimelineReceived(list timeline)

Αυτό το γεγονός προκύπτει, όταν τα μηνύματα που ζητήθηκαν μέσω του RequestFriendTimeline έχουν ανακτηθεί. Η παράμετρος timeline και η ιδιότητα Timeline θα περιέχουν μια λίστα από τις λίστες, όπου κάθε υπο-λίστα περιέχει μια ενημερωμένη κατάσταση της φόρμας (username message).

IsAuthorized ()

Αυτό το γεγονός προκύπτει αφού το πρόγραμμα καλέσει το Authorize εάν η έγκριση ήταν επιτυχής. Καλείται, επίσης, μετά από κλήση του CheckAuthorized αν έχουμε ήδη ένα έγκυρο διακριτικό πρόσβασης. Εφόσον αυτό το γεγονός έχει ξεκινήσει, οποιαδήποτε άλλη μέθοδο για αυτό το στοιχείο μπορεί να κληθεί.

MentionsReceived(list mentions)

Αυτό το γεγονός προκύπτει, όταν οι αναφορές που έχουν ζητηθεί μέσω του RequestMentions από τους συνδεδεμένους χρήστες έχουν ανακτηθεί. Μια λίστα με τις αναφορές μπορεί τότε να βρεθεί στη παράμετρο mentions ή στην ιδιότητα mentions.

SearchSuccessful(list searchResults)

Αυτό το γεγονός προκύπτει, όταν τα αποτελέσματα της αναζήτησης που έχουν ζητηθεί μέσω του SearchSuccessful έχουν ανακτηθεί. Μια λίστα με τα αποτελέσματα μπορεί να βρεθεί στην παράμετρο results ή στην ιδιότητα results.

Methods

Authorize()

Ανακατευθύνει το χρήστη για να συνδεθεί στο Twitter μέσω του προγράμματος περιήγησης στο Web χρησιμοποιώντας το πρωτόκολλο OAuth, αν δεν έχει ήδη άδεια.

CheckAuthorized ()

Ελέγχει αν υπάρχει ήδη πρόσβαση, και αν ναι, καλεί το IsAuthorized του event handler να ξεκινήσει.

DeAuthorize()

Αφαιρεί την άδεια του Twitter για παράδειγμα από την τρέχουσα εφαρμογή.

DirectMessage(text user, text message)

Αυτό στέλνει ένα άμεσο (ιδιωτικό) μήνυμα σε συγκεκριμένο χρήστη. Το μήνυμα θα πρέπει να μειωθεί εάν υπερβαίνει 160 χαρακτήρες.

Απαιτήσεις: Αυτό θα πρέπει να κληθεί μόνο αφού έχει ξεκινήσει το γεγονός IsAuthorized, που δείχνει ότι ο χρήστης έχει συνδεθεί επιτυχώς στο Twitter.

Follow(text user)

Ξεκινά να ακολουθεί ένα χρήστη.

RequestDirectMessages ()

Ζητά τα 20 πιο πρόσφατα άμεσα μηνύματα να σταλούν στο συνδεδεμένο χρήστη. Όταν τα μηνύματα ανακτηθούν, το σύστημα θα εκτελέσει το γεγονός DirectMessagesReceived και θα ρυθμίσει την ιδιότητα DirectMessages στη λίστα των μηνυμάτων.

Απαιτήσεις: Αυτό θα πρέπει να κληθεί εφόσον έχει ξεκινήσει το γεγονός IsAuthorized, που δείχνει ότι ο χρήστης έχει συνδεθεί επιτυχώς στο Twitter.

RequestFollowers ()

Δείχνει ποιος ακολουθεί το χρήστη.

RequestFriendTimeline ()

Παίρνει τα 20 πιο πρόσφατα μηνύματα στο χρονοδιάγραμμα του χρήστη.

RequestMentions ()

Ζητά τις 20 πιο πρόσφατες αναφορές του συνδεδεμένου χρήστη. Όταν οι αναφορές ανακτηθούν, το σύστημα θα ξεκινήσει το γεγονός MentionsReceived και θα ορίσει την ιδιότητα mentions στη λίστα αναφορών.

Απαιτήσεις: Αυτό θα πρέπει να κληθεί εφόσον έχει ξεκινήσει το γεγονός IsAuthorized, που δείχνει ότι ο χρήστης έχει συνδεθεί επιτυχώς στο Twitter.

SearchTwitter(text query)

Αυτό ψάχνει στο Twitter για συγκεκριμένο ερώτημα String.

Απαιτήσεις: Αυτό θα πρέπει να κληθεί εφόσον έχει ξεκινήσει το γεγονός IsAuthorized, που δείχνει ότι ο χρήστης έχει συνδεθεί επιτυχώς στο Twitter.

StopFollowing(text user)

Σταματά να ακολουθεί ένα χρήστη.

Tweet(text status)

Αυτό στέλνει ένα tweet ως συνδεδεμένος χρήστης με καθορισμένο κείμενο, το οποίο θα πρέπει να μειωθεί εάν υπερβαίνει τους 160 χαρακτήρες.

Απαιτήσεις: Αυτό θα πρέπει να κληθεί εφόσον έχει ξεκινήσει το γεγονός IsAuthorized, που δείχνει ότι ο χρήστης έχει συνδεθεί επιτυχώς στο Twitter.

TweetWithImage(text status, text imagePath)

Αυτό στέλνει ένα tweet, ως συνδεδεμένος χρήστης με καθορισμένο κείμενο και μια διαδρομή για την εικόνα που πρόκειται να μεταφορτωθεί, το οποίο θα πρέπει να μειωθεί εάν υπερβαίνει τους 160 χαρακτήρες. Αν μια εικόνα δεν έχει βρεθεί ή δεν είναι έγκυρη, η ενημέρωση δεν θα σταλεί.

Απαιτήσεις: Αυτό θα πρέπει να κληθεί εφόσον έχει ξεκινήσει το γεγονός `IsAuthorized`, που δείχνει ότι ο χρήστης έχει συνδεθεί επιτυχώς στο Twitter.

1.7 Storage

1.7.1 File

Μη ορατό στοιχείο για την αποθήκευση και την ανάκτηση αρχείων. Χρησιμοποιείται για να γραφούν ή να διαβαστούν αρχεία στη συσκευή του χρήστη. Η προεπιλεγμένη συμπεριφορά είναι να γραφούν αρχεία στον ιδιωτικό κατάλογο δεδομένων που σχετίζονται με την εφαρμογή. Το `companion` γράφει αρχεία στο `/sdcard/AppInventor/data` για εύκολο εντοπισμό σφαλμάτων. Αν η διαδρομή του αρχείου ξεκινά με κάθετο (`/`), τότε το αρχείο δημιουργείται στο `/sdcard`. Για παράδειγμα, γράφοντας ένα αρχείο στο `/myFile.txt` το αρχείο θα γραφτεί στο `/sdcard/myFile.txt`.

Properties

Καμία

Events

`GotText(text text)`

Γεγονός που δείχνει ότι τα περιεχόμενα του αρχείου έχουν διαβαστεί.

Methods

`AppendToFile(text text, text fileName)`

Προσαρτά κείμενο στο τέλος του αρχείου. Δημιουργεί το αρχείο, εάν δεν υπάρχει ήδη.

`Delete(text fileName)`

Διαγράφει ένα αρχείο από τα αποθηκευμένα αρχεία. Το πρόθεμα του ονόματος του αρχείου με κάθετο (`/`) χρησιμοποιείται για να διαγραφεί ένα συγκεκριμένο αρχείο στην κάρτα SD (για παράδειγμα, το `/myFile.txt` θα διαγράψει το `/sdcard/myFile.txt` αρχείο). Εάν το όνομα του αρχείου δεν αρχίζει με κάθετο (`/`), τότε το αρχείο που βρίσκεται αποθηκευμένο σε ιδιωτικό μέρος του προγράμματος θα διαγραφεί.

Ξεκινώντας το αρχείο με // είναι λάθος, διότι δεν μπορούν να διαγραφούν τα αρχεία του ενεργητικού.

ReadFrom(text fileName)

Διαβάζει το κείμενο από ένα αρχείο που είναι αποθηκευμένο. Ορίζοντας ως πρόθεμα στο όνομα του αρχείου την κάθετο (/) διαβάζεται ένα συγκεκριμένο αρχείο στην κάρτα SD (για παράδειγμα, /myFile.txt θα διαβάσει το /sdcard/myFile.txt αρχείο). Για να διαβάσει τα στοιχεία που διατίθενται μαζί με μια εφαρμογή (λειτουργεί, επίσης, και για το Companion) πρέπει το όνομα του αρχείου να ξεκινάει με δύο καθέτους (//). Εάν το όνομα του αρχείου δεν αρχίζει με κάθετο, θα διαβαστεί από τα ιδιωτικά αποθέματα της εφαρμογής (για πακέτα εφαρμογών) και από το /sdcard /AppInventor/data για το Companion.

SaveFile(text text, text fileName)

Αποθηκεύει το κείμενο σε ένα αρχείο. Αν το όνομα του αρχείου αρχίζει με κάθετο (/), το αρχείο είναι γραμμένο στην sdcard (για παράδειγμα, γράφοντας /myFile.txt το αρχείο θα γραφτεί στο /sdcard/myFile.txt). Εάν το όνομα του αρχείου δεν αρχίζει με κάθετο, θα γραφτεί στον ιδιωτικό κατάλογο δεδομένων του προγράμματος όπου δεν θα είναι προσβάσιμο σε άλλα προγράμματα του τηλεφώνου. Υπάρχει μια εξαίρεση για το AI Companion όπου αυτά τα αρχεία γράφονται στο /sdcard/AppInventor/data για τη διευκόλυνση του εντοπισμού σφαλμάτων. Πρέπει να σημειωθεί ότι αυτό το μπλοκ θα αντικαταστήσει το αρχείο εάν υπάρχει ήδη. Αν θέλουμε να προσθέσουμε περιεχόμενο σε ένα αρχείο χρησιμοποιούμε το append block.

1.7.2 FusionTablesControl

Ένα μη-ορατό στοιχείο που επικοινωνεί με το Google Fusion Tables. Το Fusion Tables μας επιτρέπει να αποθηκεύουμε, να μοιραζόμαστε, να θέτουμε ερωτήματα και να απεικονίζουμε πίνακες δεδομένων. Αυτό το στοιχείο μας επιτρέπει να ερωτάμε, να δημιουργούμε και να τροποποιούμε αυτούς τους πίνακες.

Εφαρμογές που χρησιμοποιούν το Fusion Tables πρέπει να επικυρωθούν από τους διακομιστές της Google. Υπάρχουν δύο τρόποι για να γίνει αυτό. Ο πρώτος τρόπος χρησιμοποιεί μόνο ένα κλειδί API που κατέχει ο χρήστης (ο προγραμματιστής). Με

την προσέγγιση αυτή οι τελικοί χρήστες πρέπει, επίσης, να συνδεθούν για να αποκτήσουν πρόσβαση σε ένα Fusion Table.

Η δεύτερη προσέγγιση είναι να χρησιμοποιηθεί η υπηρεσία ελέγχου ταυτότητας. Με αυτή την προσέγγιση ο χρήστης δημιουργηθεί πιστοποιήσεις και ένα ειδικό «λογαριασμό υπηρεσίας διεύθυνσης ηλεκτρονικού ταχυδρομείου», ο οποίος επιτρέπει στους τελικούς χρήστες να χρησιμοποιούν Fusion tables χωρίς να κάνουν login. Ο λογαριασμό της υπηρεσίας πιστοποιεί κάθε πρόσβαση.

Χρησιμοποιώντας το στοιχείο FusiontablesControl

Δημιουργία Fusion Tables

Είναι τόσο εύκολο ο χρήστης να δημιουργήσει τον δικό του Fusion Table, όσο είναι και η δημιουργία ενός εγγράφου Google. Τα βήματα είναι τα εξής:

1. Σύνδεση στο λογαριασμό Gmail ή σε οποιαδήποτε άλλη υπηρεσία της Google (π.χ., Drive, YouTube). Πλοήγηση στο Google Drive.
2. Επιλογή του κουμπιού «New» και να περιήγησε στο «More». Εάν δεν υπάρχει η επιλογή Google Fusion Tables, επιλογή του «Connect more apps» και περιήγηση στη σελίδα των υπηρεσιών της Google για εύρεση της υπηρεσίας Fusion tables και σύνδεση με το προσωπικό Google Drive.
3. Υπάρχουν παραδείγματα και tutorials ώστε ο χρήστης να μπορέσει να μάθει τα βασικά.
4. Επιλογή του κουμπιού «See my tables» (στο επάνω δεξιό μέρος της σελίδας). Η επιλογή αυτή θα φέρει στο προσκήνιο την προσωπική σελίδα του χρήστη.
5. Εμφάνιση λίστας των πινάκων του χρήστη ή των πινάκων που μπορεί άλλοι να έχουν μοιραστεί με αυτόν.
6. Χρησιμοποίηση του κουμπιού «create» για τη δημιουργία νέου πίνακα. Ονομασία μερικών στηλών και αποθήκευση.
7. Επιλογή του κουμπιού «Share» (πάνω δεξιά) για να τροποποιηθούν τα δικαιώματα (άδειες) του πίνακα.

Η δημιουργία ενός Fusiontable App

Όταν σύρουμε το στοιχείο του FusionTablesControl μέσα στο πλαίσιο του designer, δεν πρέπει να ξεχάσουμε να ορίσουμε την ιδιότητα του ApiKey του, η οποία είναι αρχικά κενή. Θα πρέπει να την αντιγράψουμε από το [Google Developers Console](#) και να την επικολλήσουμε στο πεδίο property.

Για να πάρουμε ένα API κλειδί, ακολουθούμε τις παρακάτω οδηγίες.

1. Πηγαίνουμε στο Google Developers Console και συνδεόμαστε εάν είναι απαραίτητο.
2. Από το «APIs & auth» επιλέγουμε το «APIs» από το μενού στα αριστερά.
3. Επιλέγουμε το «Fusion Tables API» από τη λίστα που εμφανίζεται και την ενεργοποιούμε.
4. Στην αριστερή μπάρα, επιλέξτε το στοιχείο «Credentials».
5. Από το «Public API access» επιλέγουμε το «Create new Key», και διαλέγουμε το «create» για να δημιουργήσουμε ένα API Key.

Το API κλειδί (ή κλειδιά) θα εμφανιστεί στο παράθυρο δίπλα στο «Public API access». Πρέπει να δοθεί το κλειδί αυτό ως τιμή για την ιδιότητα ApiKey στην εφαρμογή του Fusion Tables.

Αφού δημιουργήσουμε το API Key, θέτουμε την τιμή της ιδιότητα «Query» σε ένα έγκυρο FusionTables SQL query και καλούμε το SendQuery να εκτελέσει το query. Το App Inventor θα στείλει το ερώτημα στο διακομιστή του Fusion table και το μπλοκ του GotResult θα ξεκινήσει όταν ένα αποτέλεσμα επιστραφεί από το διακομιστή. Τα αποτελέσματα των ερωτημάτων θα επιστραφούν σε μορφή CSV, και μπορούν να μετατραπούν σε μορφή λίστας, χρησιμοποιώντας τα μπλοκ του «list from csv table» ή «list from csv row».

Για να συσταθεί υπηρεσία ελέγχου ταυτότητας για το Fusion Table, πρέπει να ακολουθηθούν τα παρακάτω πρόσθετα βήματα:

1. Στο Google APIs Console κάτω από το APIs & auth επιλέγουμε το στοιχείο APIs από το μενού στα αριστερά.
2. Επιλέγουμε το κουμπί «Create new Client ID». Διαλέγουμε την επιλογή «Service account», και κάνουμε κλικ στην επιλογή «Create Client ID».

3. Ένα αρχείο που ονομάζεται «KeyFile» (έχοντας την κατάληξη .p12) θα κατέβει αυτόματα στον υπολογιστή. Μόλις η δημιουργία ολοκληρωθεί, θα πάρουμε έναν πίνακα με τα στοιχεία του λογαριασμού της υπηρεσίας.
4. Στο παράθυρο σχεδιαστή του App Inventor, επιλέγουμε το FusionTablesControl. Στο παράθυρο ιδιοτήτων, προσθέτουμε το ServiceAccountEmail (από τον πίνακα της κονσόλας), ανεβάζουμε το KeyFile και επιλέγουμε το πλαίσιο UseServiceAuthentication.
5. Μοιραζόμαστε το Fusion Table με το προσωπικό μας ServiceAccountEmail, και του δίνουμε δικαιώματα επεξεργασίας, όπως ακριβώς θα μοιραζόμασταν οποιαδήποτε άλλο έγγραφο Google με μια διεύθυνση ηλεκτρονικού ταχυδρομείου.

Properties

ApiKey

Το κλειδί Google API.

KeyFile

Καθορίζει τη διαδρομή του αρχείου του ιδιωτικού κλειδιού. Αυτό το αρχείο κλειδιού, χρησιμοποιείται για να αποκτήσουμε πρόσβαση στο FusionTables API μέσω της υπηρεσία ελέγχου ταυτότητας.

Query

Το query που αποστέλλεται στο Fusion Tables API.

Για έγκυρα query formats και παραδείγματα, μπορούμε να ανατρέξουμε στο [Fusion Tables API v1.0 reference manual](#).

ServiceAccountEmail

Η υπηρεσία λογαριασμού ηλεκτρονικού ταχυδρομείου που χρησιμοποιείται για τον έλεγχο ταυτότητας της υπηρεσίας.

UseServiceAuthentication

Υποδηλώνει πότε ένας λογαριασμός υπηρεσίας χρησιμοποιείται για τον έλεγχο ταυτότητας.

Events

GotResult(text result)

Δηλώνει ότι το query του Fusion Tables έχει ολοκληρώσει την διαδικασία και επιστρέφει το αποτέλεσμα. Το αποτέλεσμα του query γενικά επιστρέφεται σε μορφή CSV, και μπορεί να μετατραπεί σε μορφή λίστας, χρησιμοποιώντας τα μπλοκ «list from csv table» ή «list from csv row».

Methods

DoQuery ()

Καταργημένη. Αυτή η κατηγορία έχει καταργηθεί από το τέλος του 2012. Χρησιμοποιούμε το SendQuery αντ 'αυτού.

ForgetLogin ()

Ξεχνάει τα στοιχεία σύνδεσης του τελικού χρήστη. Δεν έχει καμία επίδραση στην υπηρεσία ελέγχου ταυτότητας.

GetRows(text tableId, text columns)

Παίρνει όλες τις γραμμές από ένα συγκεκριμένο Fusion Table. Το αναγνωριστικό πεδίο του πίνακα (υποχρεωτικό) είναι το id του Fusion Table. Το πεδίο των στηλών είναι μια λίστα διαχωρισμένη με κόμματα από τις στήλες για να μπορεί να ανακτηθεί.

GetRowsWithConditions(text tableId, text columns, text conditions)

Παίρνει όλες τις σειρές από το Fusion Table που πληρούν ορισμένες προϋποθέσεις. Το αναγνωριστικό πεδίο του πίνακα (υποχρεωτικό) είναι το id του Fusion Table. Το πεδίο των στηλών είναι μια λίστα διαχωρισμένη με κόμματα από τις στήλες για να μπορεί να ανακτηθεί. Το πεδίο των συνθηκών καθορίζει τι σειρές του πίνακα προς ανάκτηση (για παράδειγμα, οι σειρές στις οποίες μια συγκεκριμένη τιμή στήλης δεν είναι μηδέν).

InsertRow(text tableId, text columns, text values)

Εισάγει μια σειρά στον καθορισμένο Fusion Table. Το αναγνωριστικό πεδίο του πίνακα (υποχρεωτικό) είναι το id του Fusion Table. Το πεδίο των στηλών είναι μια

λίστα διαχωρισμένη με κόμματα μέσω των οποίων εισάγουμε τιμές. Το πεδίο των τιμών καθορίζει τι τιμές πρέπει να εισαχθούν σε κάθε στήλη.

SendQuery ()

Στέλνει το query στον διακομιστή του Fusion Tables.

1.7.3 TinyDB

Το TinyDB είναι ένα μη ορατό στοιχείο που αποθηκεύει δεδομένα για μια εφαρμογή. Οι εφαρμογές που δημιουργούνται με το App Inventor αρχικοποιούν τις τιμές που τρέχουν κάθε φορά . Αυτό σημαίνει ότι αν μια εφαρμογή ορίσει την τιμή μιας μεταβλητής και ο χρήστης στη συνέχεια κλείσει την εφαρμογή, η τιμή αυτής της μεταβλητής δεν θα υπάρχει την επόμενη φορά που η εφαρμογή θα εκτελεστεί. Σε αντίθεση, το TinyDB είναι ένα μόνιμο μέρος αποθήκευσης δεδομένων για την εφαρμογή. Τα δεδομένα που είναι αποθηκευμένα σε μια TinyDB θα είναι διαθέσιμα κάθε φορά που η εφαρμογή εκτελείται. Ένα παράδειγμα θα μπορούσε να είναι ένα παιχνίδι που σώζει την υψηλότερη βαθμολογία και την ανακτά κάθε φορά που παίζεται το παιχνίδι.

Τα στοιχεία των δεδομένων είναι συμβολοσειρές που αποθηκεύονται κάτω από ετικέτες. Για να αποθηκεύσουμε ένα στοιχείο δεδομένων, πρέπει να καθορίσουμε την ετικέτα (tag) κάτω από την οποία θα αποθηκεύεται. Στη συνέχεια, μπορούμε να ανακτήσουμε τα δεδομένα που είναι αποθηκευμένα κάτω από το δοθέν tag.

Κάθε εφαρμογή έχει την δική της αποθήκη δεδομένων. Υπάρχει μόνο μια αποθήκη δεδομένων ανά εφαρμογή. Ακόμα κι αν έχουμε πολλά στοιχεία TinyDB, θα χρησιμοποιούν την ίδια αποθήκη δεδομένων. Για να πάρουμε το αποτέλεσμα σε ξεχωριστά καταστήματα, χρησιμοποιούμε διαφορετικά κλειδιά. Δεν μπορούμε να χρησιμοποιήσουμε το TinyDB για να περάσουμε δεδομένα ανάμεσα σε δύο διαφορετικές εφαρμογές στο τηλέφωνο, αν και μπορούμε να χρησιμοποιήσουμε το TinyDB για να μοιραζόμαστε δεδομένα μεταξύ διαφορετικών οθονών σε εφαρμογές που περιέχουν πολλές οθόνες.

Κατά την ανάπτυξη εφαρμογών με τη χρήση του AI Companion, όλες οι εφαρμογές που χρησιμοποιούν το Companion θα μοιράζονται το ίδιο TinyDB. Αυτή η κατανομή θα εξαφανιστεί μόλις οι εφαρμογές συσκευάζονται για να εγκατασταθούν στο

τηλέφωνο. Κατά τη διάρκεια της ανάπτυξης θα πρέπει να είμαστε προσεκτικοί ώστε να διαγράφουμε τα δεδομένα του Companion της εφαρμογής κάθε φορά που θα αρχίζουμε να δημιουργούμε μια νέα εφαρμογή.

Properties

Καμία

Events

Κανένα

Methods

ClearAll ()

Καθαρίζει ολόκληρο τον χώρο αποθήκευσης δεδομένων στο TinyDB.

ClearTag(text tag)

Καταργεί την καταχώρηση με τη δοσμένη ετικέτα.

any GetTags()

Επιστρέφει μια λίστα με όλες τις ετικέτες στο TinyDB.

any GetValue(text tag, any valueIfTagNotThere)

Ανακτά την τιμή που είναι αποθηκευμένη κάτω από τη δοσμένη ετικέτα. Αν δεν υπάρχει τέτοια ετικέτα, επιστρέφεται τη valueIfTagNotThere.

StoreValue(text tag, any valueToStore)

Αποθηκεύει την τιμή με βάση τη δοθείσα ετικέτα. Η αποθήκευση παραμένει στο τηλέφωνο όταν η εφαρμογή επανεκκινείται.

1.7.4 TinyWebDB

Μη ορατό στοιχείο που επικοινωνεί με μια υπηρεσία Web για την αποθήκευση και ανάκτηση πληροφοριών.

Properties

ServiceURL

Η διεύθυνση URL της βάσης δεδομένων με την οποία το στοιχείο θα επικοινωνεί.

Events

GotValue(text tagFromWebDB, any valueFromWebDB)

Υποδεικνύει ότι ένα αίτημα GetValue του διακομιστή έχει πετύχει.

ValueStored ()

Γεγονός που υποδηλώνει ότι το αίτημα StoreValue του διακομιστή έχει πετύχει.

WebServiceError(text message)

Δηλώνει ότι η επικοινωνία με την υπηρεσία Web σηματοδότησε ένα σφάλμα.

Methods

GetValue(text tag)

Στέλνει ένα αίτημα προς την υπηρεσία Web για να πάρει την τιμή η οποία είναι αποθηκευμένη κάτω από τη δοθείσα ετικέτα. Η υπηρεσία Web πρέπει να αποφασίσει τι πρέπει να επιστρέψει αν δεν υπάρχει αποθηκευμένη τιμή κάτω από αυτή την ετικέτα. Αυτό το στοιχείο δέχεται οτιδήποτε επιστραφεί.

StoreValue(text tag, any valueToStore)

Στέλνει ένα αίτημα προς την υπηρεσία Web για να αποθηκεύσει την δοθείσα τιμή κάτω από τη δοθείσα ετικέτα.

1.8 Connectivity Components

1.8.1 Activity Starter

Ένα στοιχείο που μπορεί να ξεκινήσει μια δραστηριότητα χρησιμοποιώντας τη μέθοδο StartActivity.

Δραστηριότητες που μπορεί να ξεκινήσουν περιλαμβάνουν:

- Ξεκίνημα ενός άλλου App Inventor για Android εφαρμογές. Για να γίνει αυτό, πρώτα πρέπει να μάθουμε το «class» της άλλης εφαρμογής κατεβάζοντας (download) τον πηγαίο κώδικα και χρησιμοποιώντας ένα αρχείο εξερευνητής ή ξεξιπάροντας (unzip) το βοηθητικό πρόγραμμα για να βρούμε ένα αρχείο με

όνομα «youngandroidproject/project.properties». Η πρώτη γραμμή του αρχείου θα αρχίζει με «main =» και θα ακολουθείται από το όνομα της κλάσης. Για παράδειγμα, main=com.gmail.Bitdiddle.Ben.HelloPurr.Screen1. (Τα πρώτα στοιχεία δείχνουν ότι δημιουργήθηκε από τον Ben.Bitdiddle@gmail.com). Για να κάνουμε το ActivityStarter να ξεκινήσει αυτή την εφαρμογή, πρέπει να ρυθμίσουμε τις ακόλουθες ιδιότητες:

1. Εισάγουμε στο ActivityPackage το όνομα της κλάσης, αφαιρώντας το τελευταίο στοιχείο (για παράδειγμα, com.gmail.Bitdiddle.Ben.HelloPurr).
 2. Εισάγουμε στο ActivityClass ολόκληρο το όνομα της κατηγορίας (για παράδειγμα, com.gmail.Bitdiddle.Ben.HelloPurr.Screen1).
- Ξεκίνημα της εφαρμογής της κάμερας, καθορίζοντας τις ακόλουθες ιδιότητες:
 1. Action: android.intent.action.MAIN
 2. ActivityPackage: com.android.camera
 3. ActivityClass: com.android.camera.Camera
 - Εκτέλεση αναζήτησης στο web. Ας υποθέσουμε ότι ο όρος που θέλουμε να αναζητήσουμε είναι το «vampire» οι ιδιότητες πρέπει να ρυθμιστούν ως εξής:
 1. Action: android.intent.action.WEB_SEARCH
 2. ExtraKey: query
 3. ExtraValue: vampire
 4. ActivityPackage: com.google.android.providers.enhancedgooglesearch
 5. ActivityClass: com.google.android.providers.enhancedgooglesearch.Launcher
 - Άνοιγμα προγράμματος περιήγησης σε μια συγκεκριμένη ιστοσελίδα. Αν υποθέσουμε ότι η σελίδα που θέλουμε να πάμε είναι η «www.facebook.com», οι ιδιότητες πρέπει να ρυθμιστούν ως εξής:
 1. Action: android.intent.action.VIEW
 2. DataUri: <http://www.facebook.com>

Properties

Action

ActivityClass

ActivityPackage

DataType

DataUri

ExtraKey

ExtraValue

Result

ResultName

ResultType

ResultUri

Events

AfterActivity(text result)

Γεγονός που ξεκινάει μετά τα αποτελέσματα που επιστρέφει το ActivityStarter.

Methods

text ResolveActivity()

Επιστρέφει το όνομα της δραστηριότητας που αντιστοιχεί σε αυτό το ActivityStarter, ή ένα κενό string αν δεν μπορεί να βρεθεί καμία αντίστοιχη δραστηριότητα.

StartActivity ()

Ξεκινά τη δραστηριότητα που αντιστοιχεί σε αυτό το ActivityStarter.

1.8.2 BluetoothClient

Στοιχείο του Bluetooth του πελάτη.

Properties

AddressesAndNames

Οι διευθύνσεις και τα ονόματα των συνδεδεμένων συσκευών Bluetooth.

Available

Εάν το Bluetooth είναι διαθέσιμο στη συσκευή.

CharacterEncoding

DelimiterByte

Enabled

Εάν το Bluetooth είναι ενεργοποιημένο.

HighByteFirst

IsConnected

Secure

Εάν θα εφαρμόσει το SSP (Simple Secure Pairing), το οποίο υποστηρίζεται σε συσκευές με Bluetooth v2.1 ή νεότερες εκδόσεις. Κατά την εργασία με φορητές συσκευές Bluetooth, αυτή η ιδιότητα μπορεί να χρειαστεί να οριστεί σε False. Για το Android 2.0-2.2, αυτή η ρύθμιση της ιδιότητας θα αγνοηθεί.

Events

Καμία

Methods

number BytesAvailableToReceive()

Επιστρέφει μια εκτίμηση του αριθμού των bytes που μπορούν να ληφθούν χωρίς να μπλοκαριστούν.

boolean Connect(text address)

Συνδέει τη συσκευή Bluetooth με την καθορισμένη διεύθυνση και το Serial Port Profile (SPP). Επιστρέφει την τιμή true, αν η σύνδεση ήταν επιτυχής.

boolean ConnectWithUUID(text address, text uuid)

Συνδέει τη συσκευή Bluetooth με την καθορισμένη διεύθυνση και το UUID. Επιστρέφει την τιμή true, αν η σύνδεση ήταν επιτυχής.

Disconnect()

Αποσύνδεση από τη συνδεδεμένη συσκευή Bluetooth.

boolean IsDevicePaired(text address)

Ελέγχει αν η συσκευή Bluetooth, με την καθορισμένη διεύθυνση, έχει αντιστοιχιστεί.

number ReceiveSigned1ByteNumber()

Λαμβάνει ένα υπογεγραμμένο νούμερο 1-byte από τη συνδεδεμένη συσκευή Bluetooth.

number ReceiveSigned2ByteNumber()

Λαμβάνει ένα υπογεγραμμένο αριθμό 2-byte από τη συνδεδεμένη συσκευή Bluetooth.

number ReceiveSigned4ByteNumber()

Λαμβάνει ένα υπογεγραμμένο αριθμό 4-byte από τη συνδεδεμένη συσκευή Bluetooth.

list ReceiveSignedBytes(number numberOfBytes)

Λαμβάνει πολλαπλές υπογεγραμμένες τιμές byte από τη συνδεδεμένη συσκευή Bluetooth. Αν ο αριθμός των bytes είναι μικρότερος από μηδέν (numberOfBytes), διαβάζει μέχρι να ληφθεί μια οριοθετημένη τιμή byte.

text ReceiveText(number numberOfBytes)

Λήψη κειμένου από τη συνδεδεμένη συσκευή Bluetooth. Αν ο αριθμός των byte (numberOfBytes) είναι μικρότερος από μηδέν, διαβάζει μέχρι να ληφθεί μια οριοθετημένη τιμή byte.

number ReceiveUnsigned1ByteNumber()

Λαμβάνει έναν ανυπόγραφο αριθμός 1-byte από τη συνδεδεμένη συσκευή Bluetooth.

`number ReceiveUnsigned2ByteNumber()`

Λαμβάνει έναν ανυπόγραφο αριθμό 2-byte από τη συνδεδεμένη συσκευή Bluetooth.

`number ReceiveUnsigned4ByteNumber()`

Λαμβάνει έναν ανυπόγραφο αριθμό 4-byte από τη συνδεδεμένη συσκευή Bluetooth.

`list ReceiveUnsignedBytes(number numberOfBytes)`

Λαμβάνει πολλαπλά ανυπόγραφα byte από τη συνδεδεμένη συσκευή Bluetooth. Αν ο αριθμός των byte (`numberOfBytes`) είναι μικρότερος από μηδέν, διαβάζει μέχρι να ληφθεί μια οριοθετημένη τιμή byte.

`Send1ByteNumber(text number)`

Στέλνει έναν αριθμό 1-byte στη συνδεδεμένη συσκευή Bluetooth.

`Send2ByteNumber(text number)`

Στέλνει έναν αριθμό 2-byte στη συνδεδεμένη συσκευή Bluetooth.

`Send4ByteNumber(text number)`

Στέλνει έναν αριθμό 4-byte στη συνδεδεμένη συσκευή Bluetooth.

`SendBytes(list list)`

Στέλνει μια λίστα με τιμές byte στη συνδεδεμένη συσκευή Bluetooth.

`SendText(text text)`

Στέλνει κείμενο στη συνδεδεμένη συσκευή Bluetooth.

1.8.3 BluetoothServer

Στοιχείο του διακομιστή Bluetooth.

Properties

Available: Boolean (δυναδικός)

Εάν το Bluetooth είναι διαθέσιμο στη συσκευή Android.

CharacterEncoding: text (κείμενο)

Η κωδικοποίηση χαρακτήρων που θα χρησιμοποιηθεί κατά την αποστολή και λήψη κειμένου.

DelimiterByte: number (αριθμός)

Το οριοθετημένο byte που χρησιμοποιείται όταν εισάγεται ένας αρνητικός αριθμός στην παράμετρο numberOfBytes όταν καλείται ReceiveText, ReceiveSignedBytes, ή ReceiveUnsignedBytes.

Enabled: boolean

Εάν το Bluetooth είναι ενεργοποιημένο.

HighByteFirst: boolean

Εάν οι αριθμοί με 2 και 4 byte πρέπει να αποστέλλονται και να λαμβάνονται με το μεγαλύτερο (ή πιο σημαντικό) byte πρώτο. Πρέπει να ελεγχτεί το εγχειρίδιο της συσκευής με την οποία η εφαρμογή θα επικοινωνεί για να οριστούν οι κατάλληλες ρυθμίσεις. Αυτό είναι επίσης γνωστό ως big-endian.

IsAccepting: boolean

Ενημερώνει για το αν το συγκεκριμένο στοιχείο του BluetoothServer δέχεται μια εισερχόμενη σύνδεση.

IsConnected: boolean

Ενημερώνει για το αν η σύνδεση με το Bluetooth έχει πραγματοποιηθεί.

Events

ConnectionAccepted ()

Δηλώνει ότι η σύνδεση με Bluetooth έχει γίνει αποδεκτή.

Methods

AcceptConnection(text serviceName)

Αποδοχή μιας εισερχόμενης σύνδεσης με το προφίλ σειριακής θύρας (SPP).

`AcceptConnectionWithUUID(text serviceName, text uuid)`

Αποδοχή μιας εισερχόμενης σύνδεσης με ένα συγκεκριμένο UUID.

`number BytesAvailableToReceive()`

Επιστρέφει μια εκτίμηση του αριθμού των bytes που μπορούν να ληφθούν χωρίς να εμποδιστούν.

`Disconnect()`

Αποσύνδεση από τη συνδεδεμένη συσκευή Bluetooth.

`number ReceiveSigned1ByteNumber()`

Λαμβάνει έναν αριθμό 1-byte από τη συνδεδεμένη συσκευή Bluetooth.

`number ReceiveSigned2ByteNumber()`

Λαμβάνει έναν αριθμό 2-byte από τη συνδεδεμένη συσκευή Bluetooth.

`number ReceiveSigned4ByteNumber()`

Λαμβάνει έναν αριθμό 4-byte από τη συνδεδεμένη συσκευή Bluetooth.

`list ReceiveSignedBytes(number numberOfBytes)`

Λαμβάνει πολλαπλές τιμές byte από τη συνδεδεμένη συσκευή Bluetooth. Αν ο `numberOfBytes` είναι μικρότερος από μηδέν, διαβάζει μέχρι να ληφθεί μια τιμή για το οριοθετημένο byte.

`text ReceiveText(number numberOfBytes)`

Λήψη κειμένου από τη συνδεδεμένη συσκευή Bluetooth. Αν το `numberOfBytes` είναι μικρότερο από μηδέν, διαβάζει μέχρι να ληφθεί μια τιμή για το οριοθετημένο byte.

`number ReceiveUnsigned1ByteNumber()`

Λαμβάνει ένα ανυπόγραφο αριθμό 1-byte από τη συνδεδεμένη συσκευή Bluetooth.

`number ReceiveUnsigned2ByteNumber()`

Λαμβάνει ένα ανυπόγραφο αριθμό 2-byte από τη συνδεδεμένη συσκευή Bluetooth.

`number ReceiveUnsigned4ByteNumber()`

Λαμβάνει ένα ανυπόγραφο αριθμό 4-byte από τη συνδεδεμένη συσκευή Bluetooth.

`list ReceiveUnsignedBytes(number numberOfBytes)`

Λαμβάνει πολλαπλές τιμές ανυπόγραφα byte από τη συνδεδεμένη συσκευή Bluetooth. Αν το `numberOfBytes` είναι μικρότερο από μηδέν, διαβάζει μέχρι να ληφθεί μια τιμή για ένα οριοθετημένο byte.

`Send1ByteNumber(text number)`

Στέλνει έναν 1-byte αριθμό στη συνδεδεμένη συσκευή Bluetooth.

`Send2ByteNumber (text number)`

Στέλνει έναν 2-byte αριθμό στη συνδεδεμένη συσκευή Bluetooth.

`Send4ByteNumber (text number)`

Στέλνει έναν 1-byte αριθμό στη συνδεδεμένη συσκευή Bluetooth.

`SendBytes (list list)`

Στέλνει μια λίστα με τιμές byte στη συνδεδεμένη συσκευή Bluetooth.

`SendText (text text)`

Στέλνει κείμενο στη συνδεδεμένη συσκευή Bluetooth.

`StopAccepting ()`

Σταματάει να δέχεται μια εισερχόμενη σύνδεση.

1.8.4 Web

Μη ορατό στοιχείο που παρέχει λειτουργίες για τα αιτήματα HTTP GET, POST, PUT και DELETE.

Properties

AllowCookies

Εάν τα cookies από μια απάντηση θα πρέπει να αποθηκευτούν και να χρησιμοποιηθούν στις επόμενες αιτήσεις. Τα cookies υποστηρίζονται μόνο σε έκδοση Android 2.3 ή μεγαλύτερη.

RequestHeaders

Το αίτημα των κεφαλίδων, σαν μια λίστα με υπολίστες δύο στοιχείων. Το πρώτο στοιχείο της κάθε υπολίστας αντιπροσωπεύει το αίτημα της κεφαλίδας του πεδίου του ονόματος. Το δεύτερο στοιχείο της κάθε υπολίστας αντιπροσωπεύει το αίτημα της κεφαλίδας των τιμών του πεδίου, είτε μια τιμή είτε μια λίστα που περιέχει πολλαπλές τιμές.

ResponseFileName

Το όνομα του αρχείου στο οποίο θα αποθηκευτεί η απάντηση. Εάν η τιμή του SaveResponse είναι αληθής και το ResponseFileName είναι κενό, τότε θα δημιουργηθεί ένα νέο όνομα αρχείου.

SaveResponse

Εάν η απάντηση θα πρέπει να αποθηκευτεί σε ένα αρχείο.

Url

Η διεύθυνση URL για το αίτημα του web.

Events

GotFile(text url, number responseCode, text responseType, text fileName)

Γεγονός που δείχνει ότι το αίτημα έχει τελειώσει.

GotText(text url, number responseCode, text responseType, text responseContent)

Γεγονός που δείχνει ότι το αίτημα έχει τελειώσει.

Methods

text BuildRequestData(list list)

Μετατρέπει μια λίστα με υπολίστες δύο στοιχείων, που αντιπροσωπεύουν το όνομα και την αξία, σε μια σειρά που έχει μορφοποιηθεί ως application/x-www-form-urlencoded media type, κατάλληλο για να περάσει στο PostText.

ClearCookies ()

Καθαρίζει όλα τα cookies για αυτό το στοιχείο Web.

Delete()

Εκτελεί ένα αίτημα HTTP DELETE, χρησιμοποιώντας την ιδιότητα URL και ανακτά την απάντηση.

Εάν η τιμή της ιδιότητας SaveResponse είναι αληθής, η απάντηση θα αποθηκευτεί σε ένα αρχείο και το γεγονός GotFile θα ενεργοποιηθεί. Η ιδιότητα ResponseFileName μπορεί να χρησιμοποιηθεί για να καθοριστεί το όνομα του αρχείου.

Εάν η ιδιότητα SaveResponse είναι ψευδής, το γεγονός GotText θα ενεργοποιηθεί.

Get ()

Εκτελεί μια αίτηση HTTP GET χρησιμοποιώντας την ιδιότητα URL και ανακτά την απάντηση.

Εάν η τιμή της ιδιότητας SaveResponse είναι αληθής, η απάντηση θα αποθηκευτεί σε ένα αρχείο και το γεγονός GotFile θα ενεργοποιηθεί. Η ιδιότητα ResponseFileName μπορεί να χρησιμοποιηθεί για να καθοριστεί το όνομα του αρχείου.

Εάν η ιδιότητα SaveResponse είναι ψευδής, το γεγονός GotText θα ενεργοποιηθεί.

text HtmlTextDecode(text htmlText)

Αποκωδικοποιεί τη δοθείσα τιμή κειμένου HTML. Οι οντότητες χαρακτήρων της HTML, όπως & ;, < ;, > ;, &apos ;, και ", έχουν αλλάξει σε &, <,>, "and". Οντότητες όπως &#xhhhh, και &#nnnn έχουν αλλάξει σε κατάλληλους χαρακτήρες.

any JsonTextDecode(text jsonText)

Αποκωδικοποιεί τη δοθείσα κωδικοποιημένη JSON τιμή ώστε να παράγει μια αντίστοιχη τιμή AppInventor. Μια λίστα JSON [x, y, z] αποκωδικοποιείται σε μια

λίστα (x y z), ένα αντικείμενο JSON με όνομα A και αξία B, (συμβολίζεται ως A: B κλεισμένο σε αγκύλες) αποκωδικοποιείται σε μια λίστα ((AB)), η οποία είναι μια λίστα που περιέχει τη λίστα δύο στοιχείων (AB).

PostFile(text path)

Εκτελεί μια αίτηση HTTP POST χρησιμοποιώντας την ιδιότητα url και στοιχεία από συγκεκριμένο αρχείο.

Εάν η τιμή της ιδιότητας SaveResponse είναι αληθής, η απάντηση θα αποθηκεύεται σε ένα αρχείο και το γεγονός GotFile θα ενεργοποιείται. Η ιδιότητα ResponseFileName μπορεί να χρησιμοποιηθεί για να καθοριστεί το όνομα του αρχείου.

Εάν η τιμή της ιδιότητας SaveResponse είναι ψευδής, το γεγονός GotText θα ενεργοποιηθεί.

PostText(text text)

Εκτελεί μια αίτηση HTTP POST χρησιμοποιώντας το URL της ιδιότητας και το συγκεκριμένο κείμενο.

Οι χαρακτήρες του κειμένου κωδικοποιούνται χρησιμοποιώντας κωδικοποίηση UTF-8.

Εάν η τιμή της ιδιότητας SaveResponse είναι αληθής, η απάντηση θα αποθηκεύεται σε ένα αρχείο και το γεγονός GotFile θα ενεργοποιείται. Η ιδιότητα responseFileName μπορεί να χρησιμοποιηθεί για να καθοριστεί το όνομα του αρχείου.

Εάν η τιμή της ιδιότητας SaveResponse είναι ψευδής, το γεγονός GotText θα ενεργοποιηθεί.

PostTextWithEncoding(text text, text encoding)

Εκτελεί μια αίτηση HTTP POST χρησιμοποιώντας το URL της ιδιότητας και το συγκεκριμένο κείμενο.

Οι χαρακτήρες του κειμένου κωδικοποιούνται χρησιμοποιώντας τη δοθείσα κωδικοποίηση.

Εάν η τιμή της ιδιότητα `SaveResponse` είναι αληθής, η απάντηση θα αποθηκεύεται σε ένα αρχείο και το γεγονός `GotFile` θα ενεργοποιείται. Η ιδιότητα `ResponseFileName` μπορεί να χρησιμοποιηθεί για να καθοριστεί το όνομα του αρχείου.

Εάν η τιμή της ιδιότητας `SaveResponse` είναι ψευδής, το γεγονός `GotText` θα ενεργοποιηθεί.

`PutFile(text path)`

Εκτελεί μια αίτηση HTTP PUT χρησιμοποιώντας την ιδιότητα `url` και τα στοιχεία από το καθορισμένο αρχείο.

Εάν η τιμή της ιδιότητα `SaveResponse` είναι αληθής, η απάντηση θα πρέπει να αποθηκεύεται σε ένα αρχείο και το γεγονός `GotFile` θα ενεργοποιείται. Η ιδιότητα `ResponseFileName` μπορεί να χρησιμοποιηθεί για να καθοριστεί το όνομα του αρχείου.

Εάν η τιμή της ιδιότητας `SaveResponse` είναι ψευδής, το γεγονός `GotText` θα ενεργοποιηθεί.

`PutText(text text)`

Εκτελεί μια αίτηση HTTP PUT χρησιμοποιώντας το URL της ιδιότητας και το συγκεκριμένο κείμενο.

Οι χαρακτήρες του κειμένου κωδικοποιούνται χρησιμοποιώντας κωδικοποίηση UTF-8.

Εάν η τιμή της ιδιότητας `SaveResponse` είναι αληθής, η απάντηση θα πρέπει να αποθηκεύεται σε ένα αρχείο και το γεγονός `GotFile` θα ενεργοποιείται. Η ιδιότητα `responseFileName` μπορεί να χρησιμοποιηθεί για να καθοριστεί το όνομα του αρχείου.

Εάν η τιμή της ιδιότητας `SaveResponse` είναι ψευδής, το γεγονός `GotText` θα ενεργοποιηθεί.

`PutTextWithEncoding(text text, text encoding)`

Εκτελεί μια αίτηση HTTP PUT χρησιμοποιώντας το URL της ιδιότητας και το συγκεκριμένο κείμενο.

Οι χαρακτήρες του κειμένου κωδικοποιούνται χρησιμοποιώντας τη δοθείσα κωδικοποίηση.

Εάν η τιμή της ιδιότητας `SaveResponse` είναι αληθής, η απάντηση θα αποθηκεύεται σε ένα αρχείο και το γεγονός `GotFile` θα ενεργοποιείται. Η ιδιότητα `ResponseFileName` μπορεί να χρησιμοποιηθεί για να καθοριστεί το όνομα του αρχείου.

Εάν η τιμή της ιδιότητας `SaveResponse` είναι ψευδής, το γεγονός `GotText` θα ενεργοποιηθεί.

`text UriEncode(text text)`

Κωδικοποιεί την δεκτή τιμή του κειμένου έτσι ώστε να μπορεί να χρησιμοποιηθεί σε μια διεύθυνση URL.

`any XMLTextDecode(text XmlText)`

Αποκωδικοποιεί το δοθέν XML string για να παράγει τη δομή λίστας.

1.9 LEGO MINDSTORMS

Αυτά τα στοιχεία παρέχουν έλεγχο των ρομπότ LEGO® MINDSTORMS® NXT μέσω Bluetooth.

Τα LEGO και MINDSTORMS είναι σήματα κατατεθέντα του Ομίλου LEGO.

ΣΗΜΑΝΤΙΚΟ: Όλα αυτά τα στοιχεία έχουν μια ιδιότητα `BluetoothClient` που πρέπει να ρυθμιστεί στο designer του App Inventor (στο πρόγραμμα περιήγησης). Η ιδιότητα δεν μπορεί να οριστεί στο blocks editor. Η ιδιότητα δείχνει ποιο στοιχείο `BluetoothClient` θα χρησιμοποιηθεί για την επικοινωνία με το ρομπότ. Θα πρέπει να προσθέσουμε ένα στοιχείο `BluetoothClient` στο έργο μας. Αν έχουμε ένα ρομπότ, θα πρέπει να έχουμε ένα στοιχείο `BluetoothClient`. Εάν έχουμε δύο ρομπότ και θέλουμε να ελέγξουμε και τα δύο ταυτόχρονα από μία εφαρμογή, θα χρειαστούμε δύο στοιχεία `BluetoothClient` στο έργο μας. Το στοιχείο `BluetoothClient` είναι διαθέσιμο στο σημείο της παλέτας "Not ready for prime time".

Εδώ είναι μια λίστα με τα αρχικά βήματα που θα πρέπει να εκτελέσουμε για να χρησιμοποιήσουμε ένα ή περισσότερα από τα στοιχεία NXT:

1. Πηγαίνουμε στην παλέτα και κάνουμε κλικ στο "Not ready for prime time".
2. Σύρουμε ένα στοιχείο BluetoothClient και το αφήνουμε στο Viewer.
3. Το στοιχείο αυτόματα θα ονομαστεί BluetoothClient1.
4. Στην παλέτα κάνουμε κλικ στο "MINDSTORMS LEGO».
5. Σύρουμε ένα από τα στοιχεία, για παράδειγμα NxtDirectCommands, το αφήνουμε στο Viewer.
6. Στο κουτί των ιδιοτήτων (properties), κάνουμε κλικ στην περιοχή μετά το BluetoothClient (προσωρινά «None...»).
7. Ένα πλαίσιο εμφανίζεται με μια λίστα με όλα τα BluetoothClient στοιχεία στο έργο μας.
8. Κάνουμε κλικ στο BluetoothClient1 και κλικ στο OK.
9. Αν θέλουμε, προσθέτουμε ένα άλλο στοιχείο, για παράδειγμα το NxtColorSensor, και επαναλαμβάνουμε τα βήματα 6-8 για να ρυθμίσουμε την ιδιότητα του BluetoothClient.

1.9.1 NxtDirectCommands

Ένα στοιχείο που παρέχει ένα περιβάλλον χαμηλού επιπέδου σε ένα ρομπότ Lego Mindstorms NXT, με λειτουργίες για να στέλνει άμεσες εντολές NXT (Direct Commands).

Properties

BluetoothClient

Το στοιχείο BluetoothClient που θα πρέπει να χρησιμοποιηθεί για την επικοινωνία. Η ρύθμισή του πρέπει να γίνει στο Designer.

Methods

DeleteFile (text fileName)

Διαγραφή αρχείου στο ρομπότ.

DownloadFile (text source, text destination)

Κατέβασμα ενός αρχείου στο ρομπότ.

GetBatteryLevel ()

Μας δίνει το επίπεδο φόρτισης της μπαταρίας για το ρομπότ. Επιστρέφει την τάση σε millivolt.

GetBrickName ()

Μας δίνει το όνομα του ρομπότ.

GetCurrentProgramName ()

Μας δίνει το όνομα του τρέχοντος προγράμματος του ρομπότ.

GetFirmwareVersion ()

Μας δίνει σε λίστα τους αριθμούς του firmware και της έκδοσης πρωτοκόλλου για το ρομπότ, όπου το πρώτο στοιχείο είναι ο αριθμός έκδοσης του firmware και το δεύτερο στοιχείο είναι ο αριθμός έκδοσης του πρωτοκόλλου.

GetInputValues (text sensorPortLetter)

Διαβάζει τις τιμές ενός αισθητήρα εισόδου στο ρομπότ. Θεωρείται ότι ο τύπος του αισθητήρα έχει διαμορφωθεί μέσω του SetInputMode.

GetOutputState (text motorPortLetter)

Διαβάζει την κατάσταση της εξόδου του κινητήρα για το ρομπότ.

KeepAlive ()

Κράτημα σε ζωή. Επιστρέφει το τρέχον όριο ύπνου σε χιλιοστά του δευτερολέπτου (milliseconds).

ListFiles (text wildcard)

Επιστρέφει μια λίστα με τα ονόματα των αρχείων που ταιριάζουν με αυτά που βρίσκονται στο ρομπότ.

LsGetStatus (κείμενο sensorPortLetter)

Επιστρέφει τον αριθμό των διαθέσιμων bytes προς ανάγνωση.

LsRead (text sensorPortLetter)

Διαβάζει ανυπόγραφα δεδομένα χαμηλής ταχύτητας από έναν αισθητήρα εισόδου στο ρομπότ. Θεωρεί ότι ο τύπος του αισθητήρα έχει διαμορφωθεί μέσω του SetInputMode.

LsWrite (text sensorPortLetter, list list, number rxDataLength)

Γράφει δεδομένα χαμηλής ταχύτητας σε έναν αισθητήρα εισόδου στο ρομπότ. Θεωρεί ότι ο τύπος του αισθητήρα έχει διαμορφωθεί μέσω του SetInputMode.

MessageRead (number mailbox)

Διαβάζει ένα μήνυμα από ένα mailbox (1-10) στο ρομπότ.

MessageWrite (number mailbox, text message)

Γράφει ένα μήνυμα στο mailbox (1-10) για το ρομπότ.

PlaySoundFile (text fileName)

Παίζει ένα αρχείο ήχου για το ρομπότ.

PlayTone (number frequencyHz, number durationMs)

Κάνει το ρομπότ να παίζει έναν ήχο.

ResetInputScaledValue (text sensorPortLetter)

Επαναφέρει την κλιμακούμενη τιμή του αισθητήρα εισόδου στο ρομπότ.

ResetMotorPosition (text motorPortLetter, boolean relative)

Επαναφέρει τη θέση του μοτέρ.

SetBrickName (text name)

Ορίζει το brick name του ρομπότ.

SetInputMode (text sensorPortLetter, number sensorType, number sensorMode)

Διαμορφώνει έναν αισθητήρα εισόδου στο ρομπότ.

SetOutputState (text motorPortLetter, number power, number mode, number regulationMode, number turnRatio, number runState, number tachoLimit)

Ορίζει την κατάσταση της εξόδου του κινητήρα στο ρομπότ.

StartProgram (text programName)

Ξεκινάει την εκτέλεση ενός προγράμματος, που έγινε προσφάτως η λήψη του, για το ρομπότ.

StopProgram ()

Σταματάει την εκτέλεση του τρέχοντος προγράμματος στο ρομπότ.

StopSoundPlayback ()

Διακόπτει την αναπαραγωγή του ήχου.

1.9.2 NxtColorSensor

Ένα στοιχείο που παρέχει μια διεπαφή υψηλού επιπέδου σε έναν αισθητήρα χρώματος σε ένα ρομπότ Lego Mindstorms NXT.

Properties

BluetoothClient

Η συνιστώσα BluetoothClient που θα πρέπει να χρησιμοποιηθεί για επικοινωνία. Οι ρυθμίσεις της γίνονται στο Designer.

SensorPort

Η θύρα του αισθητήρα με την οποία ο αισθητήρας είναι συνδεδεμένος. Οι ρυθμίσεις πραγματοποιούνται στο Designer.

DetectColor

Πότε ο αισθητήρας θα πρέπει να εντοπίσει το χρώμα ή το φως.

Εάν η τιμή είναι αληθής, υποδεικνύει ότι ο αισθητήρας θα πρέπει να εντοπίσει το χρώμα. Εάν είναι ψευδής, υποδεικνύει ότι ο αισθητήρας θα πρέπει να ανιχνεύσει το φως.

Αν η ιδιότητα DetectColor έχει οριστεί σε True, τα γεγονότα BelowRange, WithinRange και AboveRange δεν θα πραγματοποιηθούν και ο αισθητήρας δεν θα παράγει το χρώμα.

Αν η ιδιότητα DetectColor έχει οριστεί σε False, το γεγονός ColorChanged δεν θα πραγματοποιηθεί.

ColorChangedEventEnabled

Ορίζει εάν το γεγονός ColorChanged πρέπει να πραγματοποιηθεί, όταν η ιδιότητα DetectColor έχει οριστεί σε True και έχει ανιχνεύσει τις αλλαγές του χρώματος.

GenerateColor

Το χρώμα που θα παράγεται από τον αισθητήρα.

Μόνο οι τιμές «None», «Red», «Green», και «Blue» είναι έγκυρες.

Ο αισθητήρας δεν θα παράγει χρώμα, όταν η ιδιότητα DetectColor έχει οριστεί σε True.

BottomOfRange

Το bottom of range, χρησιμοποιείται για τα γεγονότα BelowRange, WithinRange, και AboveRange.

TopOfRange

Το top of range χρησιμοποιείται για τα γεγονότα BelowRange, WithinRange, και AboveRange.

BelowRangeEventEnabled

Αν το γεγονός BelowRange πρέπει να πραγματοποιηθεί, όταν η ιδιότητα DetectColor έχει οριστεί σε False και το επίπεδο φωτός πηγαίνει κάτω από το BottomOfRange.

WithinRangeEventEnabled

Αν το γεγονός WithinRange πρέπει να πραγματοποιηθεί, όταν η ιδιότητα DetectColor έχει οριστεί σε False και το επίπεδο φωτός κυμαίνεται μεταξύ του BottomOfRange και το TopOfRange.

AboveRangeEventEnabled

Αν το γεγονός AboveRange πρέπει να πραγματοποιηθεί, όταν η ιδιότητα DetectColor έχει οριστεί σε False και το επίπεδο φωτός είναι πάνω από την TopOfRange.

Events

ColorChanged(number color)

Το εντοπισμένο χρώμα έχει αλλάξει.

Το γεγονός ColorChanged δεν θα πραγματοποιηθεί αν η ιδιότητα DetectColor ή η ιδιότητα ColorChangedEventEnabled έχουν οριστεί σε False.

BelowRange ()

Το επίπεδο του φωτός έχει πέσει κάτω από του εύρους.

Το γεγονός BelowRange δεν θα πραγματοποιηθεί αν η ιδιότητα DetectColor έχει οριστεί σε True ή εάν η ιδιότητα BelowRangeEventEnabled έχει οριστεί σε False.

WithinRange ()

Το επίπεδο του φωτός είναι μέσα στο εύρος.

Το γεγονός WithinRange δεν θα πραγματοποιηθεί αν η ιδιότητα DetectColor έχει οριστεί σε True ή εάν η ιδιότητα WithinRangeEventEnabled έχει οριστεί σε False.

AboveRange ()

Το επίπεδο του φωτός έχει περάσει πάνω από το εύρος.

Το γεγονός AboveRange δεν θα πραγματοποιηθεί αν η ιδιότητα DetectColor έχει οριστεί σε True ή αν η ιδιότητα AboveRangeEventEnabled έχει οριστεί σε False.

Methods

GetColor ()

Επιστρέφει το τρέχον ανιχνεύσιμο χρώμα ή το χρώμα «None», αν το χρώμα δεν μπορεί να διαβαστεί ή εάν η ιδιότητα DetectColor έχει οριστεί σε False.

GetLightLevel ()

Επιστρέφει το τρέχον επίπεδο φωτός ως τιμή μεταξύ 0 και 1023, ή -1 αν το επίπεδο φωτός δεν μπορεί να διαβαστεί ή εάν η ιδιότητα DetectColor έχει οριστεί σε True.

1.9.3 NxtLightSensor

Ένα στοιχείο που παρέχει μια διεπαφή υψηλού επιπέδου σε έναν αισθητήρα φωτός για ένα ρομπότ Lego Mindstorms NXT.

Properties

BluetoothClient

Το στοιχείο BluetoothClient που θα πρέπει να χρησιμοποιηθεί για την επικοινωνία. Πρέπει να ρυθμιστεί στο Designer.

SensorPort

Η θύρα του αισθητήρα με την οποία ο αισθητήρας είναι συνδεδεμένος. Πρέπει να ρυθμιστεί στο Designer.

GenerateLight

Πότε ο αισθητήρας φωτός θα πρέπει να παράγει φως.

BottomOfRange

Το μικρότερο μέρος του φάσματος που χρησιμοποιείται για τα γεγονότα BelowRange, WithinRange, και AboveRange.

TopOfRange

Το μεγαλύτερο μέρος του φάσματος που χρησιμοποιείται για τα γεγονότα BelowRange, WithinRange, και AboveRange.

BelowRangeEventEnabled

Εάν το γεγονός BelowRange πρέπει να εκκινηθεί, όταν το επίπεδο φωτός πηγαίνει κάτω από το BottomOfRange.

WithinRangeEventEnabled

Εάν το γεγονός WithinRange πρέπει να εκκινηθεί, όταν το επίπεδο φωτός κυμαίνεται μεταξύ του BottomOfRange και του TopOfRange.

AboveRangeEventEnabled

Εάν το γεγονός AboveRange πρέπει να εκκινηθεί, όταν το επίπεδο φωτός περνά πάνω από το TopOfRange.

Events

BelowRange ()

Το επίπεδο φωτός έχει πέσει κάτω από το φάσμα.

WithinRange ()

Το επίπεδο φωτός βρίσκεται μέσα στο φάσμα.

AboveRange ()

Το επίπεδο φωτός έχει περάσει πάνω από το φάσμα.

Methods

GetLightLevel ()

Επιστρέφει το τρέχον επίπεδο φωτός ως τιμή μεταξύ 0 και 1023, ή -1 αν το επίπεδο φωτός δεν μπορεί να διαβαστεί.

1.9.4 NxtSoundSensor

Ένα στοιχείο που παρέχει μια διεπαφή υψηλού επιπέδου σε ένα αισθητήρα ήχου σε ένα ρομπότ Lego Mindstorms NXT.

Properties

BluetoothClient

Το στοιχείο BluetoothClient που θα πρέπει να χρησιμοποιείται για την επικοινωνία. Πρέπει να ρυθμιστεί στο Designer.

SensorPort

Η θύρα του αισθητήρα με την οποία ο αισθητήρας είναι συνδεδεμένος. Πρέπει να ρυθμιστεί στο Designer.

BottomOfRange

Το μικρότερο σημείο του φάσματος που χρησιμοποιείται για τα γεγονότα BelowRange, WithinRange, και AboveRange.

TopOfRange

Το μεγαλύτερο σημείο του φάσματος που χρησιμοποιείται για τα γεγονότα BelowRange, WithinRange, και AboveRange.

BelowRangeEventEnabled

Εάν το γεγονός BelowRange πρέπει να εκκινηθεί, όταν το επίπεδο του ήχου πηγαίνει κάτω από το BottomOfRange.

WithinRangeEventEnabled

Εάν το γεγονός WithinRange πρέπει να εκκινηθεί, όταν το επίπεδο του ήχου κυμαίνεται μεταξύ του BottomOfRange και του TopOfRange.

AboveRangeEventEnabled

Εάν το γεγονός AboveRange πρέπει να εκκινηθεί, όταν το επίπεδο του ήχου πηγαίνει πάνω από το TopOfRange.

Events

BelowRange ()

Το επίπεδο του ήχου έχει πέσει κάτω από το φάσμα.

WithinRange ()

Το επίπεδο του ήχου έχει περάσει μέσα στο φάσμα.

AboveRange ()

Το επίπεδο του ήχου έχει περάσει πάνω από το φάσμα.

Methods

GetSoundLevel ()

Επιστρέφει το τρέχον επίπεδο του ήχου ως μια τιμή μεταξύ 0 και 1023, ή -1 αν το ηχητικό επίπεδο δεν μπορεί να διαβαστεί.

1.9.5 NxtTouchSensor

Ένα στοιχείο που παρέχει μια διεπαφή υψηλού επιπέδου σε έναν αισθητήρα αφής σε ένα ρομπότ Lego Mindstorms NXT.

Properties

BluetoothClient

Το στοιχείο BluetoothClient που θα πρέπει να χρησιμοποιείται για την επικοινωνία. Πρέπει να ρυθμιστεί στο Designer.

SensorPort

Η θύρα του αισθητήρα με την οποία ο αισθητήρας είναι συνδεδεμένος. Πρέπει να ρυθμιστεί στο Designer.

PressedEventEnabled

Εάν το γεγονός Pressed θα πρέπει να εκκινηθεί, όταν ο αισθητήρας αφής έχει πιεστεί.

ReleasedEventEnabled

Εάν το γεγονός Released θα πρέπει να εκκινηθεί, όταν ο αισθητήρας αφής απελευθερωθεί.

Events

Pressed()

Ο αισθητήρας αφής έχει πατηθεί.

Released()

Ο αισθητήρας αφής έχει απελευθερωθεί.

Methods

IsPressed ()

Επιστρέφει true αν ο αισθητήρας αφής έχει πιεστεί.

1.9.6 NxtUltrasonicSensor

Ένα στοιχείο που παρέχει μια διεπαφή υψηλού επιπέδου σε έναν αισθητήρα υπερήχων σε ένα ρομπότ Lego Mindstorms NXT.

Properties

BluetoothClient

Το στοιχείο BluetoothClient που θα πρέπει να χρησιμοποιείται για την επικοινωνία. Πρέπει να ρυθμιστεί στο Designer.

SensorPort

Η θύρα του αισθητήρα με την οποία ο αισθητήρας είναι συνδεδεμένος. Πρέπει να ρυθμιστεί στο Designer.

BottomOfRange

Το μικρότερο μέρος του φάσματος που χρησιμοποιείται για τα γεγονότα BelowRange, WithinRange, και AboveRange.

TopOfRange

Το μικρότερο μέρος του φάσματος που χρησιμοποιείται για τα γεγονότα BelowRange, WithinRange, και AboveRange.

BelowRangeEventEnabled

Εάν το γεγονός BelowRange πρέπει να εκκινηθεί, όταν η απόσταση πηγαίνει κάτω από το BottomOfRange.

WithinRangeEventEnabled

Εάν το γεγονός WithinRange πρέπει να εκκινηθεί, όταν η απόσταση κυμαίνεται μεταξύ του BottomOfRange και την TopOfRange.

AboveRangeEventEnabled

Εάν το γεγονός AboveRange πρέπει να εκκινηθεί, όταν η απόσταση πηγαίνει πάνω από το TopOfRange.

Events

BelowRange ()

Η απόσταση έχει πέσει κάτω από το φάσμα.

WithinRange ()

Η απόσταση κυμαίνεται μέσα στην περιοχή.

AboveRange ()

Η απόσταση έχει περάσει πάνω από την περιοχή.

Methods

GetDistance ()

Επιστρέφει την τρέχουσα απόσταση σε εκατοστόμετρα ως μια τιμή μεταξύ 0 και 254, ή -1 αν η απόσταση δεν μπορεί να διαβαστεί.

1.9.7 NxtDrive

Ένα στοιχείο που παρέχει μια διεπαφή υψηλού επιπέδου σε ένα ρομπότ Lego Mindstorms NXT, με λειτουργίες που μπορούν να κινούν και να στρίβουν το ρομπότ.

Properties

BluetoothClient

Το στοιχείο BluetoothClient που θα πρέπει να χρησιμοποιείται για την επικοινωνία. Πρέπει να ρυθμιστεί στο Designer.

DriveMotors

Οι θύρες motor που χρησιμοποιούνται για την οδήγηση.

WheelDiameter

Η διάμετρος των τροχών που χρησιμοποιούνται για την οδήγηση.

StopBeforeDisconnect

Αν σταματήσουν οι κινητήρες πριν από την αποσύνδεση.

Methods

MoveForwardIndefinitely (number power)

Μετακινεί το ρομπότ προς τα εμπρός επ' αόριστον, με το ορισμένο ποσοστό της μέγιστης ισχύος.

MoveForward (number power, number distance)

Μετακινεί το ρομπότ μπροστά ανάλογα με τη δοθείσα απόσταση, με το καθορισμένο ποσοστό της μέγιστης ισχύος.

MoveBackwardIndefinitely (number power)

Μετακινεί το ρομπότ προς τα πίσω επ' αόριστον, με το ορισμένο ποσοστό της μέγιστης ισχύος.

MoveBackward (number power, number distance)

Μετακινεί το ρομπότ προς τα πίσω ανάλογα με τη δοθείσα απόσταση, με το συγκεκριμένο ποσοστό της μέγιστης ισχύος.

Stop ()

Σταματάει τα μοτέρ κίνησης του ρομπότ.

TurnClockwiseIndefinitely (number power)

Γυρνάει τα ρομπότ δεξιόστροφα επ' αόριστον, με το ορισμένο ποσοστό της μέγιστης ισχύος.

TurnCounterClockwiseIndefinitely (number power)

Γυρνάει το ρομπότ αριστερόστροφα επ' αόριστον, με το ορισμένο ποσοστό της μέγιστης ισχύος.

3.1.5 Ανάλυση της οθόνης των Blocks

1 Control Blocks

1.1 (if & if else)

1.1.1 (if...then)



Εικόνα 32-if...then

Τεστάρει μια δοθείσα συνθήκη. Εάν η συνθήκη είναι αληθής, εκτελεί τις ενέργειες στην δοθείσα ακολουθία των μπλοκ, αλλιώς τα μπλοκ αγνοούνται.

1.1.2 (if...then...else)



Εικόνα 33- if...then...else

Τεστάρει μια δοθείσα συνθήκη. Εάν το αποτέλεσμα είναι αληθές εκτελεί τις ενέργειες στην ακολουθία των μπλοκ του then, αλλιώς εκτελεί τις ενέργειες στην ακολουθία των μπλοκ του else.

1.1.3 (if...then...else if...then...else)



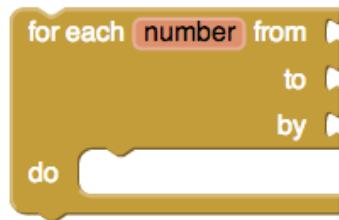
Εικόνα 34- if...then if...else

Τεστάρει μια δοθείσα συνθήκη. Εάν το αποτέλεσμα είναι αληθές, εκτελεί τις ενέργειες στην ακολουθία then, αλλιώς τεστάρει την ακολουθία else if. Εάν τα

αποτελέσματα είναι αληθή εκτελεί τις ενέργειες του then, αλλιώς εκτελεί τις ενέργειες της ακολουθίας του else.

1.2 For each from to

1.2.1 (for each (number)...from...to...by...do)

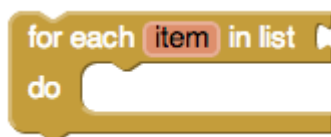


Εικόνα 35- for each (number)...from...to...by...do

Τρέχει τα μπλοκ που βρίσκονται στο πεδίο του do για κάθε αριθμητική τιμή, ξεκινώντας από το from και καταλήγοντας στο to, αυξάνοντας τον αριθμό που έχει οριστεί στο πεδίο “by”. Κάθε φορά χρησιμοποιούμε το δοθέν όνομα της μεταβλητής “number” για να αναφερθούμε στην τρέχουσα αξία. Μπορούμε να αλλάξουμε το όνομα “number” σε κάτι άλλο αν το επιθυμούμε.

1.3 For each in list

1.3.1 (for each...in list...do)



Εικόνα 36- for each...in list...do

Τρέχει τα μπλοκ που βρίσκονται στο πεδίο του “do” για κάθε αντικείμενο που ανήκει στην λίστα. Χρησιμοποιούμε το δοθέν όνομα της μεταβλητής “item” για να αναφερθούμε στο τρέχον αντικείμενο της λίστας. Μπορούμε να αλλάξουμε το όνομα “item” σε κάτι άλλο αν επιθυμούμε.

1.4 While

1.4.1 (while test...do)



Εικόνα 37- while test...do

Τεστάρει την συνθήκη test. Εάν είναι αληθής, εκτελεί την πράξη που δίνεται στο πεδίο του “do” και έπειτα τεστάρει ξανά. Εάν η συνθήκη είναι ψευδής, το μπλοκ τερματίζεται και η πράξη η οποία δίνεται στο πεδίο του “do” δεν εκτελείται πλέον.

1.5 If then else

1.5.1 (if...then...else)



Εικόνα 38- if...then...else

Τεστάρει τη δοθείσα συνθήκη. Εάν η δήλωση είναι αληθής, εκτελείται η πράξη στην ακολουθία των μπλοκ then- return και επιστρέφεται η τιμή then- return, διαφορετικά, εκτελείται η πράξη στην ακολουθία των μπλοκ else- return και επιστρέφεται η τιμή else- return.

1.6 do

1.6.1 (do...result)



Εικόνα 39- do...result

Μερικές φορές σε μια διαδικασία ή σε ένα άλλο κομμάτι του κώδικα, μπορεί να χρειαστεί να κάνουμε κάτι και να επιστρέψουμε κάτι, αλλά για διάφορους λόγους

μπορεί να επιλέξουμε να χρησιμοποιήσουμε αυτό το μπλοκ, αντί για τη δημιουργία μιας νέας διαδικασίας.

1.7 Evaluate but ignore result

1.7.1 (evaluate but ignore result)



evaluate but ignore result

Εικόνα 40- evaluate but ignore result

Παρέχει ένα “εικονικό κενό” για να ταιριάζει ένα μπλοκ που έχει εσοχή αριστερά όπου δεν υπάρχει κενό, όπως σε μια ακολουθία από μπλοκ στο κομμάτι του “do” ή σε ένα μπλοκ “if”. Το μπλοκ το οποίο θα ταιριάζει, θα τρέξει αλλά τα αποτελέσματα τα οποία θα επιστρέψει, θα αγνοηθούν. Αυτό μπορεί να φανεί χρήσιμο εάν έχουμε ορίσει μια διαδικασία η οποία επιστρέφει ένα αποτέλεσμα, αλλά θέλουμε να το καλέσουμε σε ένα πλαίσιο το οποίο δεν δέχεται το αποτέλεσμα.

1.8 Open another screen

1.8.1 (open another screen screen name)



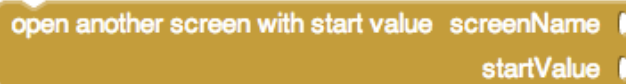
open another screen screenName

Εικόνα 41- open another screen screenName

Ανοίγει την οθόνη με το αντίστοιχο όνομα.

1.9 Open another screen with start value

1.10 (Open another screen with start value screenName...startValue)



open another screen with start value screenName
startValue

Εικόνα 42- open another screen start value screen name...startValue

Ανοίγει μια άλλη οθόνη και βάζει μια τιμή σε αυτή.

1.11 Get start value

1.11.1 (get start value)



Εικόνα 43- get start value

Επιστρέφει την αρχική τιμή που έχει δοθεί στην τρέχουσα οθόνη. Αυτή η τιμή δίνεται για να χρησιμοποιηθεί ώστε να ανοίγει μια άλλη οθόνη με τιμή εκκίνησης (open another screen with start value) ή να κλείνει μια οθόνη με μια τιμή (close screen with value).

1.12 Close screen

1.12.1 (close screen)



Εικόνα 44- close screen

Κλείνει την τρέχουσα οθόνη.

1.13 Close screen with value

1.13.1 (close screen with value result)



Εικόνα 45- close screen with value result

Κλείνει την τρέχουσα οθόνη και επιστρέφει μια τιμή στην οθόνη την οποία αυτή άνοιξε.

1.14 Close application

1.14.1 (close application)



Εικόνα 46- close application

Κλείνει την εφαρμογή.

1.15 Get plain start text

1.15.1 (get plain start text)



Εικόνα 47- get plain start text

Επιστρέφει το κείμενο το οποίο έχει περαστεί σε αυτή την οθόνη, όταν ανοιχτεί από άλλη εφαρμογή. Εάν καμία τιμή δεν έχει καταχωρηθεί, επιστρέφει κενό πεδίο. Για εφαρμογές με πολλές οθόνες, χρησιμοποιούμε το “get start value”, έναντι του “get plain start value”.

1.16 Close screen with plain text

1.16.1 (close screen with plain text)

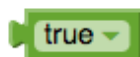


Εικόνα 48- close screen with plain text text

Κλείνει την τρέχουσα οθόνη και εισάγει κείμενο στην εφαρμογή που άνοιξε την οθόνη. Για εφαρμογές με πολλές οθόνες, χρησιμοποιούμε το “close screen with value”, έναντι του “close screen with text”.

2. Logic Blocks

2.1(True)



Εικόνα 49- true

Δηλώνει τη σταθερή τιμή True. Χρησιμοποιείται για να ορίζουμε δυαδικές ιδιότητες στις τιμές των συνιστωσών ή ως τιμή μιας μεταβλητής που αντιπροσωπεύει μια συνθήκη.

2.2(False)



Εικόνα 50- false

Αντιπροσωπεύει τη σταθερή τιμή false. Χρησιμοποιείται για να ορίζουμε δυαδικές ιδιότητες στις τιμές των συνιστωσών ή ως τιμή μιας μεταβλητής που αντιπροσωπεύει μια συνθήκη.

2.3(not)



Εικόνα 51- not

Εκτελεί λογική άρνησης, επιστρέφοντας false εάν το input είναι αληθές και true εάν το input είναι ψευδές.

2.4(=)



Εικόνα 52- =

Τεστάρει εάν τα arguments είναι ίσα.

- Δύο αριθμοί είναι ίσοι εάν αριθμητικά είναι ίσοι, για παράδειγμα το 1 είναι ίσο με το 1.0.
- Δύο πλαίσια κειμένου είναι ίσα εάν έχουν τους ίδιους χαρακτήρες στην ίδια σειρά με την ίδια πτώση (γραμματική). Για παράδειγμα, η λέξη banana δεν είναι ίδια με την λέξη Banana.
- Αριθμοί και γράμματα είναι ίσα εάν ο αριθμός είναι αριθμητικά ίσος με τον αριθμό που θα τυπωθεί με αυτά τα γράμματα. Για παράδειγμα, το 12.0 είναι ίσο με το αποτέλεσμα της πρόσθεσης του πρώτου χαρακτήρα του 1A με τον τελευταίο χαρακτήρα του Teafor2.
- Δύο λίστες είναι ίσες εάν έχουν τον ίδιο αριθμό στοιχείων και τα αντίστοιχα στοιχεία είναι ίσα.

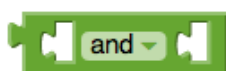
2.5 (\neq)



Εικόνα 53- διάφορο

Τεστάρει πότε δυο στοιχεία δεν είναι ίσα.

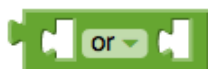
2.6 (and)



Εικόνα 54- and

Τεστάρει αν όλες οι λογικές συνθήκες ενός συνόλου είναι αληθής. Το αποτέλεσμα είναι αληθές μόνο όταν και μόνο όταν όλες οι τεσταρισμένες συνθήκες είναι αληθής. Οι συνθήκες ελέγχονται από αριστερά προς τα δεξιά και ο έλεγχος σταματάει όταν μια από τις συνθήκες είναι ψευδής. Εάν δεν υπάρχουν συνθήκες για έλεγχο, τότε το αποτέλεσμα είναι αληθές.

2.7 (or)



Εικόνα 55- or

Τεστάρει αν κάποιο από το σύνολο των λογικών υποθέσεων είναι αληθές. Τα αποτελέσματα είναι αληθή εάν μια ή περισσότερες από τις τεσταρισμένες συνθήκες είναι αληθής. Όταν προσθέτεις μια συνθήκη στο κενό πεδίο του test , ένα ακόμη κενό πεδίο εμφανίζεται ώστε να μπορέσουμε να προσθέσουμε και άλλη συνθήκη. Οι συνθήκες ελέγχονται από αριστερά προς τα δεξιά και ο έλεγχος σταματάει μόλις μια από τις συνθήκες είναι αληθής. Εάν δεν υπάρχουν συνθήκες για έλεγχο, τότε το αποτέλεσμα είναι ψευδές.

3. Math Blocks

Σημείωση: Κάθε μαθηματικό μπλοκ που έχει ασυμπλήρωτα κενά, το ασυμπλήρωτο πεδίο διαβάζεται ως 0.

Μερικά μαθηματικά μπλοκ περιέχουν πτυσσόμενο (dropdown) μενού, που σημαίνει ότι μπορούν να μετατραπούν σε διαφορετικά μπλοκ.

3.1 Basic Number Block (0)



Εικόνα 56- 0

Μπορεί να χρησιμοποιηθεί ως θετικός ή αρνητικός αριθμός (περιλαμβάνονται και δεκαδικοί). Με διπλό κλικ πάνω στο «0» που υπάρχει πάνω στο μπλοκ μπορούμε να αλλάξουμε τον αριθμό.

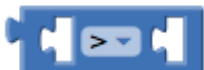
3.2 (=)



Εικόνα 57- =

Τεστάρει εάν δυο αριθμοί είναι ίσοι και επιστρέφει true ή false.

3.3 (>)



Εικόνα 58- >

Τεστάρει εάν ο πρώτος αριθμός είναι μεγαλύτερος του δεύτερου και επιστρέφει true ή false.

3.4 (>=)



Εικόνα 59- >=

Τεστάρει εάν ο πρώτος αριθμός είναι μεγαλύτερος ή ίσος με το δεύτερο αριθμό και επιστρέφει true ή false.

3.5 (<)



Εικόνα 60- <

Τεστάρει εάν ο πρώτος αριθμός είναι μικρότερος του δεύτερου και επιστρέφει true ή false.

3.6 (<=)



Εικόνα 61- <=

Τεστάρει εάν ο πρώτος αριθμός είναι μικρότερος ή ίσος του δεύτερου και επιστρέφει true ή false.

3.7 (+)



Εικόνα 62- +

Επιστρέφει τα αποτελέσματα της προσθήκης οποιουδήποτε αριθμού μπλοκ που έχουν μια κοινή αριθμητική τιμή. Μπλοκ με αριθμητική τιμή περιλαμβάνουν το βασικό αριθμητικό μπλοκ, το μήκος μιας λίστας ή κειμένου, μεταβλητές με αριθμητικές τιμές, κλπ. Αυτό το μπλοκ είναι μεταλλασσόμενο και μπορεί να επεκταθεί για να επιτρέψει περισσότερους αριθμούς προς άθροισμα.

3.8 (-)



Εικόνα 63- (-)

Επιστρέφει το αποτέλεσμα της αφαίρεσης του δεύτερου αριθμού από τον πρώτο.

3.9 (*)



Εικόνα 64- *

Επιστρέφει τα αποτελέσματα του πολλαπλασιασμού οποιουδήποτε αριθμού μπλοκ που έχουν μια κοινή αριθμητική τιμή. Είναι ένα μεταλλασσόμενο μπλοκ το οποίο μπορεί να επεκταθεί για να επιτρέψει περισσότερους αριθμούς στην πράξη.

3.10 (/)



Εικόνα 65- /

Επιστρέφει τη διαίρεση του πρώτου αριθμού από τον δεύτερο.

3.11 (^)



Εικόνα 66- ^

Επιστρέφει το αποτέλεσμα του πρώτου αριθμού υψωμένο στη δύναμη του δεύτερου.

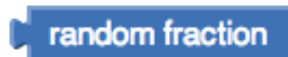
3.12 (random integer)



Εικόνα 67- random integer

Επιστρέφει ένα τυχαίο ακέραιο αριθμό μεταξύ των δοσμένων τιμών, χωρίς αποκλεισμούς. Η σειρά των δηλώσεων δεν έχει σημασία.

3.13 (random fraction)



Εικόνα 68- random fraction

Επιστρέφει μια τυχαία τιμή μεταξύ του 0 και του 1.

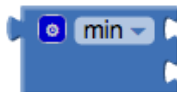
3.14 (random set seed to)



Εικόνα 69- random set seed to

Χρησιμοποιούμε αυτό το μπλοκ για να δημιουργούμε επαναλαμβανόμενες ακολουθίες τυχαίων αριθμών. Μπορούμε να δημιουργήσουμε την ίδια ακολουθία τυχαίων αριθμών καλώντας πρώτα τυχαία στοιχεία με την ίδια τιμή. Αυτό είναι χρήσιμο για να τεστάρουμε προγράμματα που περιλαμβάνουν τυχαίες τιμές.

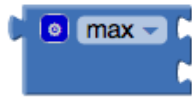
3.15 (min)



Εικόνα 70- min

Επιστρέφει την μικρότερη τιμή από ένα σύνολο αριθμών. Εάν υπάρχουν ασυμπλήρωτα πεδία στο μπλοκ, το «min» θα θεωρηθεί ως 0. Το μπλοκ αυτό είναι μεταλλασσόμενο και αναπτυσσόμενο.

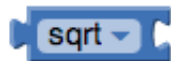
3.16 (max)



Εικόνα 71- max

Επιστρέφει την μεγαλύτερη τιμή από ένα σύνολο αριθμών. Εάν υπάρχουν ασυμπλήρωτα πεδία στο μπλοκ, το «max» θα θεωρηθεί ως 0. Το μπλοκ αυτό είναι μεταλλασσόμενο και αναπτυσσόμενο.

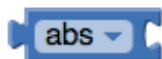
3.17 (sqrt)



Εικόνα 72- sqrt

Επιστρέφει την τετραγωνική ρίζα του δοσμένου αριθμού.

3.18 (abs)



Εικόνα 73- abs

Επιστρέφει την απόλυτη τιμή του δοσμένου αριθμού.

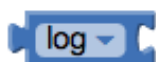
3.19 (-)



Εικόνα 74- (-)

Επιστρέφει την αρνητική τιμή του δοσμένου αριθμού.

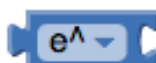
3.20 (log)



Εικόνα 75- log

Επιστρέφει το φυσικό λογάριθμο ενός δεδομένου αριθμού που είναι ο λογάριθμος προς την βάση e (2,71828 ...).

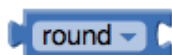
3.21 (e[^])



Εικόνα 76- e[^]

Επιστρέφει το e (2,71828 ...) υψωμένο στη δύναμη του συγκεκριμένου αριθμού.

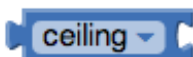
3.22 (round)



Εικόνα 77- round

Επιστρέφει το δεδομένο αριθμό στρογγυλοποιημένο στον πλησιέστερο ακέραιο αριθμό. Αν το δεκαδικό μέρος είναι $<0,5$ θα στρογγυλοποιείται προς τα κάτω. Εάν είναι $>0,5$ θα στρογγυλοποιείται προς τα πάνω. Αν είναι ακριβώς ίσο με 0,5, οι άρτιοι αριθμοί θα στρογγυλοποιούνται προς τα κάτω, και οι περιττοί αριθμοί θα στρογγυλοποιούνται προς τα πάνω.

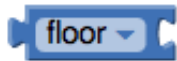
3.23 (ceiling)



Εικόνα 78- ceiling

Επιστρέφει το μικρότερο ακέραιο που είναι μεγαλύτερος ή ίσος με το δοσμένο αριθμό.

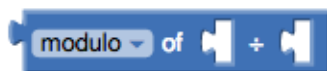
3.24 (floor)



Εικόνα 79- floor

Επιστρέφει το μεγαλύτερο ακέραιο που είναι μικρότερος ή ίσος με το δοσμένο αριθμό.

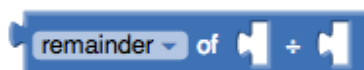
3.25 (modulo)



Εικόνα 80- modulo

Η ατελής διαίρεση (a, b) είναι η ίδια με το υπόλοιπο (a, b) όταν τα a και b είναι θετικοί. Γενικότερα, ως ατελής διαίρεση (a, b) ορίζεται για κάθε a και b, έτσι ώστε $(\text{floor}(a / b) \times b) + \text{modulo}(a, b) = a$. Για παράδειγμα, $\text{modulo}(11, 5) = 1$, $\text{modulo}(-11, 5) = 4$, $\text{modulo}(11, -5) = -4$, $\text{modulo}(-11, -5) = -1$. Η ατελής διαίρεση (a, b) έχει πάντα το ίδιο πρόσημο με το b, ενώ το υπόλοιπο (a, b) έχει πάντα το ίδιο πρόσημο με το a.

3.26 (remainder)



Εικόνα 81- remainder

Το υπόλοιπο (a, b) επιστρέφει το αποτέλεσμα της διαίρεσης του a από το b και λαμβάνει το υπόλοιπο. Το υπόλοιπο είναι το κλασματικό μέρος του αποτελέσματος πολλαπλασιαζόμενο με το b.

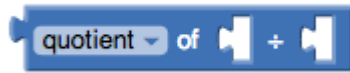
Για παράδειγμα, το υπόλοιπο $(11,5) = 1$ επειδή

$$11/5 = 2 \frac{1}{5}$$

Στην περίπτωση αυτή, το $1/5$ είναι το κλασματικό μέρος. Πολλαπλασιάζουμε αυτό με το b, σε αυτήν την περίπτωση 5 και παίρνουμε 1, το υπόλοιπο μας.

Άλλα παραδείγματα είναι υπόλοιπο $(-11, 5) = -1$, υπόλοιπο $(11, -5) = 1$, και υπόλοιπο $(-11, -5) = -1$.

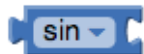
3.27 (quotient)



Εικόνα 82- quotient

Επιστρέφει το αποτέλεσμα της διαίρεσης του πρώτου αριθμού από τον δεύτερο και απορρίπτει οποιοδήποτε κλασματικό μέρος του αποτελέσματος.

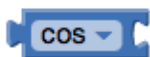
3.28 (sin)



Εικόνα 83- sin

Επιστρέφει το ημίτονο του δοσμένου αριθμού σε μοίρες.

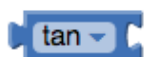
3.29 (cos)



Εικόνα 84- cos

Επιστρέφει το συνημίτονο του δοσμένου αριθμού σε μοίρες.

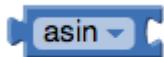
3.30 (tan)



Εικόνα 85- tan

Επιστρέφει την εφαπτομένη του δοσμένου αριθμού σε μοίρες.

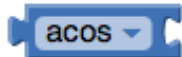
3.31 (asin)



Εικόνα 86- asin

Επιστρέφει το ημίτονο τόξου του δοσμένου αριθμού σε μοίρες.

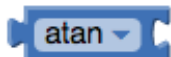
3.32 (acos)



Εικόνα 87- acos

Επιστρέφει το συνημίτονο τόξου του δοσμένου αριθμού σε μοίρες.

3.33 (atan)



Εικόνα 88- atan

Επιστρέφει το τόξο εφαπτομένης του δοσμένου αριθμού σε μοίρες.

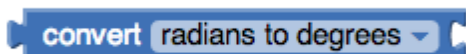
3.34 (atan2)



Εικόνα 89- atan2

Επιστρέφει το τόξο της εφαπτομένης του y / x , του δοσμένου y και x .

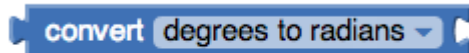
3.35 (convert radians to degrees)



Εικόνα 90- convert to degrees

Επιστρέφει την τιμή σε μοίρες του δοσμένου σε ακτίνια αριθμού. Το αποτέλεσμα θα είναι μια γωνία στο εύρος $[0, 360)$.

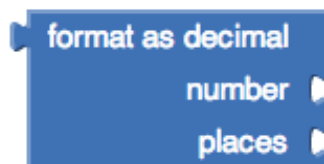
3.36 (convert degrees to radians)



Εικόνα 91- convert degrees to radians

Επιστρέφει την τιμή σε ακτίνια του δοσμένου σε μοίρες αριθμού. Το αποτέλεσμα θα είναι μια γωνία στην περιοχή $[-\pi, +\pi)$.

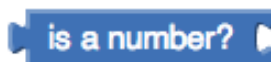
3.37 (format as decimal)



Εικόνα 92- format as decimal

Διαμορφώνει ένα νούμερο ως δεκαδικό αριθμό με ένα συγκεκριμένο αριθμό θέσεων μετά την υποδιαστολή. Ο αριθμός των θέσεων θα πρέπει να είναι ένας μη αρνητικός ακέραιος. Το αποτέλεσμα παράγεται από στρογγυλοποίηση του αριθμού (αν υπήρχαν πάρα πολλές θέσεις) ή προσθέτοντας μηδενικά στα δεξιά (αν υπήρχαν λίγες).

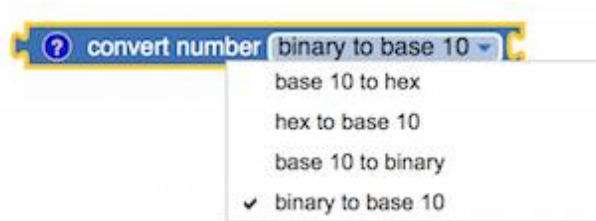
3.38 (is a number)



Εικόνα 93- is a number

Επιστρέφει true αν το συγκεκριμένο αντικείμενο είναι ένας αριθμός, και false εάν όχι.

3.39 (convert number)



Εικόνα 94- convert number

Παίρνει μια συμβολοσειρά κειμένου που αντιπροσωπεύει ένα θετικό ακέραιο σε μία βάση και επιστρέφει ένα string που αντιπροσωπεύει τον ίδιο αριθμό σε μια άλλη βάση. Για παράδειγμα, εάν η συμβολοσειρά εισόδου είναι 10, στη συνέχεια μετατρέπόμενη από τη βάση 10 σε δυαδικό θα παράγει το string 1010, ενώ αν η συμβολοσειρά εισόδου είναι το ίδιο 10, στη συνέχεια, η μετατροπή από δυαδική σε βάση 10 θα παράγει το string 2. Εάν η συμβολοσειρά εισόδου είναι το ίδιο 10, στη συνέχεια, η μετατροπή από τη βάση 10 έως εξάγωνο θα παράγει το string A.

4. Text Blocks

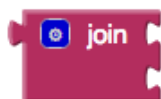
4.1 (strings “”)



Εικόνα 95- strings

Περιέχει μια συμβολοσειρά κειμένου. Αυτή η συμβολοσειρά μπορεί να περιέχει χαρακτήρες (γράμματα, αριθμούς ή άλλους ειδικούς χαρακτήρες). Στο App Inventor θεωρείται ως αντικείμενο κειμένου.

4.2 (join)



Εικόνα 96- join

Προσαρτά όλες τις εισόδους για να κάνει μία συμβολοσειρά. Αν δεν υπάρχουν δεδομένα, επιστρέφει μια κενή συμβολοσειρά.

4.3 (length)



Εικόνα 97- length

Επιστρέφει τον αριθμό των χαρακτήρων, συμπεριλαμβανομένων των διαστημάτων στη συμβολοσειρά. Αυτό είναι το μήκος της δεδομένης συμβολοσειρά κειμένου.

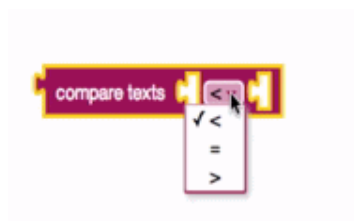
4.4 (is empty)



Εικόνα 98- is empty

Επιστρέφει το αν η συμβολοσειρά περιέχει ή όχι χαρακτήρες (συμπεριλαμβανομένων των κενών). Όταν το μήκος συμβολοσειράς είναι 0, επιστρέφει true, αλλιώς επιστρέφει false.

4.5 (compare texts <>=)



Εικόνα 99- compare texts

Επιστρέφει το αν η πρώτη συμβολοσειρά είναι ή όχι λεξικογραφικά <,>, ή = με τη δεύτερη συμβολοσειρά. Αυτό εξαρτάται ανάλογα με την επιλογή που έχει γίνει στο πτυσσόμενο μενού.

Μια συμβολοσειρά θεωρείται λεξικογραφικά μεγαλύτερη από μια άλλη, αν είναι αλφαβητικά μεγαλύτερη. Όλα τα κεφαλαία γράμματα θεωρούνται ως μικρότερα ή

υπάρχουν πριν από τα πεζά γράμματα. Η λέξη «γάτα» θα είναι μεγαλύτερη της λέξης «Γάτα».

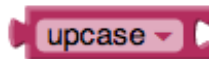
4.6 (trim)



Εικόνα 100- trim

Αφαιρεί τα κενά στην αρχή ή στο τέλος της συμβολοσειράς εισόδου και επιστρέφει το αποτέλεσμα.

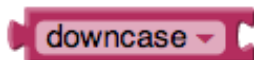
4.7 (upcase)



Εικόνα 101- upcase

Επιστρέφει ένα αντίγραφο της συμβολοσειράς κειμένου μετατρέποντας όλα τα σύμβολα σε κεφαλαία.

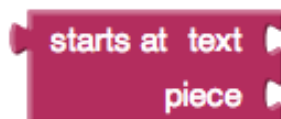
4.8 (downcase)



Εικόνα 102- downcase

Επιστρέφει ένα αντίγραφο της συμβολοσειράς κειμένου μετατρέποντας όλα τα σύμβολα σε πεζά.

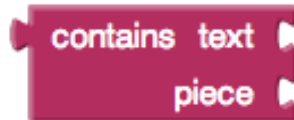
4.9 (starts at)



Εικόνα 103- starts at

Επιστρέφει τη θέση του χαρακτήρα, όπου ο πρώτος χαρακτήρας του κομματιού εμφανίζεται για πρώτη φορά στο κείμενο, ή 0 αν δεν υπάρχει. Για παράδειγμα, η θέση της “ana” στην havana banana είναι 4.

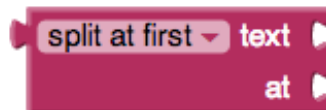
4.10 (contains)



Εικόνα 104- contains

Επιστρέφει true αν το κομμάτι εμφανίζεται στο κείμενο, αλλιώς, επιστρέφει false.

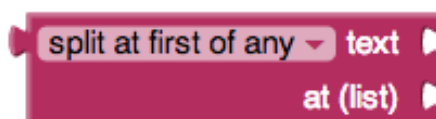
4.11 (split at first)



Εικόνα 105- split at first

Διαιρεί το δεδομένο κείμενο σε δύο κομμάτια χρησιμοποιώντας τη θέση της πρώτης εμφάνισης του στο ως το σημείο διαίρεσης και επιστρέφει μία λίστα δύο στοιχείων που αποτελείται από το κομμάτι πριν από το σημείο διαίρεσης και το κομμάτι μετά το σημείο διαίρεσης. Χωρίζοντας τα μήλο, μπανάνα, κεράσι, σκυλοτροφή με ένα κόμμα ως σημείο διάσπασης επιστρέφει μια λίστα με δύο στοιχεία: το πρώτο είναι το κείμενο "μήλο" και το δεύτερο είναι το κείμενο "μπανάνα", "κεράσι", "σκυλοτροφή". Παρατηρούμε ότι το κόμμα μετά το μήλο δεν φαίνεται στο αποτέλεσμα, γιατί αυτό είναι το σημείο διαίρεσης.

4.12 (split at first of any)



Εικόνα 106- split at first of any

Διαιρεί το συγκεκριμένο κείμενο σε μια λίστα δύο στοιχείων, χρησιμοποιώντας την πρώτη θέση του κάθε στοιχείου της λίστας ως σημείο διαχωρισμού. Διασπώντας την πρόταση "i love apples bananas apples grapes" από τη λίστα [ba, ap] θα παίρναμε μια λίστα αποτελούμενη από δύο στοιχεία το πρώτο θα ήταν το *i love* και το δεύτερο *ples apples bananas grapes*.

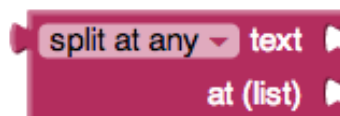
4.13 (split)



Εικόνα 107- split

Χωρίζει το κείμενο σε κομμάτια χρησιμοποιώντας τα ως σημεία διαχωρισμού και παράγει μία λίστα των αποτελεσμάτων. Ο διαχωρισμός ένα, δύο, τρία, τέσσερα, (κόμμα) επιστρέφει τη λίστα ένα δύο τρία τέσσερα. Ο διαχωρισμός μια- πατάτα, δύο-πατάτες, τρεις-πατάτες, τέσσερις-πατάτες, επιστρέφει τη λίστα ένα δύο τρία τέσσερα.

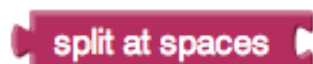
4.13 (split at any)



Εικόνα 108- split at any

Διαιρεί το συγκεκριμένο κείμενο σε μια λίστα, χρησιμοποιώντας οποιοδήποτε από τα στοιχεία ως σημείο διαχωρισμού, και επιστρέφει μια λίστα με τα αποτελέσματα. Διασπώντας τα appleberry, banana, cherry, dogfood ως λίστα δύο στοιχείων των οποίων το πρώτο στοιχείο είναι ένα κόμμα και το δεύτερο στοιχείο είναι ry, επιστρέφεται μια λίστα με τέσσερα στοιχεία: (applebe, banana, che, dogfood).

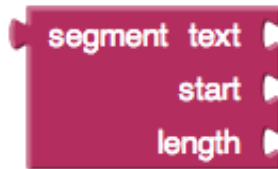
4.14 (split at spaces)



Εικόνα 109- split at spaces

Διαιρεί το συγκεκριμένο κείμενο, παράγοντας μια λίστα με τα κομμάτια.

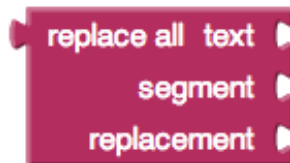
4.15 (segment)



Εικόνα 110- segment

Αποσπά μέρος του κειμένου εκκίνησης στην αρχική θέση και συνεχίζει με το μήκος των χαρακτήρων.

4.16 (replace all)



Εικόνα 111- replace all

Επιστρέφει μια νέα συμβολοσειρά κειμένου που λαμβάνεται με την αντικατάσταση όλων των υπαρχόντων υποσυμβολοσειρών. Αντικαθιστώντας όλα τα “she” με το “Hannah” στις προτάσεις *She loves writing. She loves coding as the text, She as the segment*, θα έχουμε ως αποτέλεσμα *Hannah loves eating. Hannah loves writing. Hannah loves coding.*

5. List Blocks

5.1 (create empty list)



Εικόνα 112- create empty list

Δημιουργεί κενή λίστα χωρίς στοιχεία.

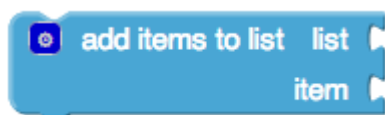
5.2 (make a list)



Εικόνα 113- make a list

Δημιουργεί μια λίστα από τα δοθέντα μπλοκ. Εάν δεν παρέχεται κανένα στοιχείο, αυτό δημιουργεί μια κενή λίστα, στην οποία μπορούμε να προσθέσουμε στοιχεία αργότερα. Αυτό το μπλοκ είναι ένα μεταλλασσόμενο. Κάνοντας κλικ στο μπλε σύμβολο θα σας επιτρέψει να προσθέσετε επιπλέον στοιχεία στη λίστα.

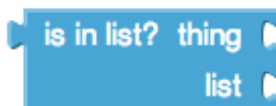
5.3 (add items to list)



Εικόνα 114- add items to list

Προσθέτει τα δοθέντα στοιχεία στο τέλος της λίστας. Η διαφορά μεταξύ αυτού και του να προσαρτούνται στη λίστα είναι ότι η προσάρτηση στη λίστα παίρνει τα στοιχεία που πρέπει να προσαρτηθούν ως μια ενιαία λίστα ενώ προσθέτοντας στοιχεία στη λίστα παίρνει τα στοιχεία ως μεμονωμένα επιχειρήματα. Αυτό το μπλοκ είναι μεταλλασσόμενο.

5.4 (is in list?)



Εικόνα 115- is in list?

Αν το πράγμα (thing) είναι ένα από τα στοιχεία της λίστας, επιστρέφει true, αλλιώς, επιστρέφει false. Σημειώνεται ότι εάν ένα κατάλογος περιέχει υποκαταλόγους, τα μέλη των υποκαταλόγων δεν είναι τα ίδια μέλη της λίστας. Για παράδειγμα, τα μέλη

της λίστας (1 2 (3 4)) είναι 1, 2, και η λίστα (3 4). Τα 3 και 4 δεν είναι τα ίδια μέλη της λίστας.

5.5 (length of list)



Εικόνα 116- length of list

Επιστρέφει τον αριθμό των στοιχείων μιας λίστας.

5.6 (is list empty?)



Εικόνα 117- is list empty?

Εάν η λίστα δεν περιέχει αντικείμενα επιστρέφει true, αλλιώς επιστρέφει false.

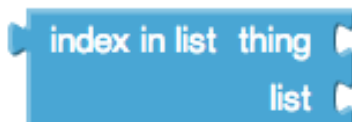
5.7 (pick a random item)



Εικόνα 118- pick a random item

Διαλέγει τυχαία ένα στοιχείο από τη λίστα.

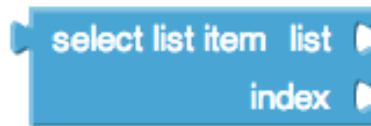
5.8 (index in list)



Εικόνα 119- index in list

Επιστρέφει τη θέση του πράγματος (thing) στη λίστα. Εάν δεν περιλαμβάνεται στη λίστα, επιστρέφει 0.

5.9 (select list item)



Εικόνα 120- select list item

Επιλέγει το αντικείμενο στο δεδομένο δείκτη στη συγκεκριμένη λίστα. Το πρώτο στοιχείο της λίστας είναι το index 1.

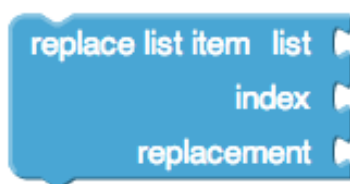
5.10 (insert list item)



Εικόνα 121- insert list item

Εισάγει ένα στοιχείο στη λίστα στη δοθείσα θέση.

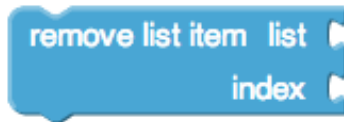
5.11 (replace list item)



Εικόνα 122- replace list item

Εισάγει την αντικατάσταση στη δοθείσα λίστα στη θέση index. Το προηγούμενο στοιχείο σε αυτή τη θέση αφαιρείται.

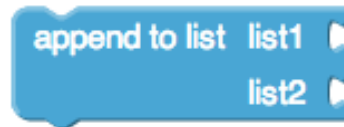
5.12 (remove list item)



Εικόνα 123- remove list item

Αφαιρεί τα αντικείμενα στη δοθείσα θέση.

5.13 (append to list)



Εικόνα 124- append to list

Προσθέτει τα στοιχεία στη δεύτερη λίστα στο τέλος της πρώτης λίστας.

5.14 (copy list)



Εικόνα 125- copy list

Δημιουργεί ένα αντίγραφο μιας λίστας, συμπεριλαμβανομένης της αντιγραφής όλων των υποκαταλόγων.

5.15 (is a list?)



Εικόνα 126- is a list?

Αν το πράγμα είναι μία λίστα, επιστρέφει true, αλλιώς επιστρέφει false.

5.16 (list to csv row)

A blue rectangular icon with rounded corners and a white border. Inside, the text "list to csv row list" is written in white, lowercase, sans-serif font. The icon has a slight 3D effect with a darker blue shadow on the right side.

Εικόνα 127- list to csv row

Ερμηνεύει τη λίστα ως μια γραμμή του πίνακα και επιστρέφει ένα CSV (τιμές διαχωρισμένες με κόμματα) κείμενο που αντιπροσωπεύει την σειρά. Κάθε στοιχείο στη γραμμή της λίστας θεωρείται ότι είναι ένα πεδίο, και αναγράφεται με διπλά εισαγωγικά στο προκύπτον κείμενο CSV. Τα στοιχεία διαχωρίζονται με κόμμα. Για παράδειγμα, η μετατροπή της λίστας (a b c d) σε μια σειρά CSV παράγει ("a", "b", "c", "d"). Η επιστρεφόμενη γραμμή κειμένου δεν έχει διαχωριστική γραμμή στο τέλος.

5.17 (list from csv row)

A blue rectangular icon with rounded corners and a white border. Inside, the text "list from csv row text" is written in white, lowercase, sans-serif font. The icon has a slight 3D effect with a darker blue shadow on the right side.

Εικόνα 128- list from csv row

Αναλύει ένα κείμενο ως CSV (τιμές διαχωρισμένες με κόμμα) διαμορφωμένη γραμμή για να παράγει μια λίστα πεδίων. Για παράδειγμα, μετατρέπει το ("a", "b", "c", "d") σε μία λίστα η οποία παράγει (a b c d).

5.18 (list to csv table)

A blue rectangular icon with rounded corners and a white border. Inside, the text "list from csv table text" is written in white, lowercase, sans-serif font. The icon has a slight 3D effect with a darker blue shadow on the right side.

Εικόνα 129- list from csv table

Ερμηνεύει τη λίστα ως έναν πίνακα row-major μορφής και επιστρέφει ένα CSV κείμενο που αντιπροσωπεύει τον πίνακα. Κάθε στοιχείο της λίστας θα πρέπει να είναι το ίδιο ένας κατάλογος που θα αντιπροσωπεύει μια γραμμή του πίνακα CSV. Κάθε στοιχείο στη σειρά λίστας θεωρείται ότι είναι ένα πεδίο, και αναγράφεται με διπλά εισαγωγικά στο προκύπτον κείμενο CSV. Στο προκύπτον κείμενο, τα στοιχεία στις σειρές διαχωρίζονται με κόμμα και οι σειρές διαχωρίζονται από CRLF (`\r\n`).

5.19 (list from csv table)



Εικόνα 130- list from csv table

Αναλύει ένα κείμενο ως CSV (τιμές διαχωρισμένες με κόμμα) μορφοποιώντας τον πίνακα ώστε να παράγει μια λίστα από γραμμές, καθεμιά από τις οποίες είναι μια λίστα από πεδία. Οι σειρές μπορούν να διαχωριστούν από νέες γραμμές (\n) ή CRLF (r\n).

5.20 (lookup in pairs)



Εικόνα 131- lookup in pairs

Χρησιμοποιείται για την αναζήτηση πληροφοριών στο λεξικό που μοιάζει με τη δομή που παρουσιάζεται με τη μορφή λίστας. Η λειτουργία αυτή λαμβάνει τρεις εισόδους, ένα κλειδί, ένα ζεύγος λίστας, και ένα αποτέλεσμα not found, οι οποίες εξ ορισμού, έχουν οριστεί σε "not found".

Εδώ τα ζεύγη πρέπει να είναι μια λίστα ζευγών, δηλαδή, μια λίστα όπου κάθε στοιχείο είναι το ίδιο μια λίστα αποτελούμενη από δύο στοιχεία.

Η αναζήτηση σε ζεύγη βρίσκει το πρώτο ζευγάρι στη λίστα του οποίου το πρώτο στοιχείο είναι το κλειδί, και επιστρέφει το δεύτερο στοιχείο. Για παράδειγμα, αν η λίστα είναι ((a apple) (d dragon) (b boxcar) (cat 100)) τότε ανατρέχοντας στο «b» θα επιστρέφει «boxcar». Αν δεν υπάρχει τέτοιο ζεύγος στη λίστα, τότε η αναζήτηση σε ζευγάρια θα επιστρέφει το NotFound αποτέλεσμα. Εάν τα ζεύγη δεν είναι λίστα από ζεύγη, τότε η λειτουργία θα σηματοδοτήσει σφάλμα.

6. Colors Blocks

6.1 (basic color block)



Εικόνα 132- basic color block

Το βασικό color block έχει ένα μικρό τετράγωνο σχήμα και ένα χρώμα στη μέση το οποίο εκπροσωπεί το χρώμα που αποθηκεύεται εσωτερικά σε αυτό το μπλοκ. Αν κάνουμε κλικ στο χρώμα στη μέση, ένα pop-up μενού εμφανίζεται στην οθόνη με έναν πίνακα 70 χρωμάτων μεταξύ των οποίων μπορούμε να επιλέξουμε. Κάνοντας κλικ σε ένα νέο χρώμα αλλάζουμε το τρέχον χρώμα του βασικού μπλοκ.



Εικόνα 133- table of colors

6.2 (make color)



Εικόνα 134- make color

Το makecolor παίρνει μια λίστα από 3 ή 4 αριθμούς. Αυτοί οι αριθμοί σε αυτή τη λίστα αντιπροσωπεύουν τις τιμές στον κώδικα RGB. Οι κωδικοί RGB χρησιμοποιούνται για να δημιουργούνται τα χρώματα στο διαδίκτυο. Ο πρώτος αριθμός σε αυτή τη λίστα αντιπροσωπεύει την τιμή R του κώδικα. Ο δεύτερος αντιπροσωπεύει το G και ο τρίτος το B. Η τέταρτη τιμή είναι προαιρετική και αντιπροσωπεύει την αξία άλφα ή το πόσο κορεσμένο είναι το χρώμα. Η

προεπιλεγμένη τιμή άλφα είναι 100. Μπορούμε να πειραματιστούμε με διαφορετικές τιμές και να δούμε πώς τα χρώματα να αλλάζουν τη χρήση αυτού του μπλοκ.

6.3 (split color)



Εικόνα 135- split color

Το splitcolor είναι το αντίθετο του makecolor. Παίρνει από ένα χρώμα: το χρώμα του μπλοκ, τη μεταβλητή που κρατάει το χρώμα, ή την ιδιότητα από ένα εκ των στοιχείων που αντιπροσωπεύουν ένα χρώμα και επιστρέφει μια λίστα με τις τιμές RGB σε κώδικα RGB αυτού του χρώματος.

Πώς λειτουργούν τα χρώματα στο app inventor:

Εσωτερικά, το App Inventor αποθηκεύει κάθε χρώμα ως ενιαίο αριθμό. Όταν χρησιμοποιούμε το makeColor και λαμβάνουμε μια λίστα ως επιχείρημα, ο κατάλογος αυτός μετατρέπεται εσωτερικά με χρήση χρωμάτων του App Inventor και αποθηκεύεται ως αριθμός. Αν γνωρίζαμε τους αριθμούς για τα χρώματα, θα μπορούσαμε να προσδιορίσουμε τι χρώμα θέλουμε να είναι κάτι απλά θέτοντας συγκεκριμένο αριθμό στην ιδιότητα του «color».

7. Variable Blocks

7.1 (initialize global name to)



Εικόνα 136- initialize global name to

Αυτό το μπλοκ χρησιμοποιείται για να δημιουργήσει τις global μεταβλητές. Παίρνει κάθε είδους τιμή ως επιχείρημα. Κάνοντας κλικ στο όνομα μπορούμε να αλλάξουμε το όνομα αυτής της μεταβλητής. Οι global μεταβλητές χρησιμοποιούνται σε όλες τις διαδικασίες ή τα γεγονότα έτσι ώστε το block να μπορέσει να σταθεί μόνο του.

Οι Global μεταβλητές μπορούν να αλλάξουν ενώ η εφαρμογή τρέχει και μπορούν να αναφέρονται και να αλλάζουν από οποιοδήποτε κομμάτι της εφαρμογής, ακόμη και στα πλαίσια των διαδικασιών και των eventhandlers. Μπορούμε να μετονομάσουμε αυτό το μπλοκ, ανά πάσα στιγμή και όλα τα συναφή μπλοκ που αναφέρονται στο παλιό όνομα θα ενημερώνονται αυτόματα.

7.2 (get)



Εικόνα 137- get

Αυτό το μπλοκ παρέχει έναν τρόπο έτσι ώστε να μπορούμε να πάρουμε οποιαδήποτε μεταβλητή έχουμε δημιουργήσει.

7.3 (set to)



Εικόνα 138- set to

Αυτό το μπλοκ ακολουθεί τους ίδιους κανόνες με αυτούς του get. Μόνο μεταβλητές στο πεδίο εφαρμογής θα είναι διαθέσιμες στην πτυσσόμενη λίστα. Μόλις επιλεγεί μια μεταβλητή *v*, ο χρήστης μπορεί να επισυνάψει ένα νέο μπλοκ και να δώσει μια νέα τιμή στη μεταβλητή *v*.

7.4 (initialize Local name to –in (do))



Εικόνα 139- initialize Local name to- in (do)

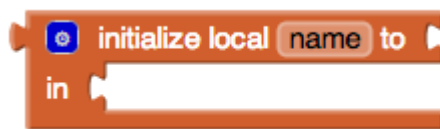
Αυτό το μπλοκ είναι μεταλλασσόμενο και μας επιτρέπει να δημιουργούμε νέες μεταβλητές οι οποίες χρησιμοποιούνται μόνο κατά τη διαδικασία που τρέχει το τμήμα

«DO» του μπλοκ. Με αυτό τον τρόπο όλες οι μεταβλητές στη διαδικασία αυτή θα ξεκινάν με την ίδια τιμή κάθε φορά που η διαδικασία εκτελείται.

Note: Αυτό το μπλοκ διαφέρει από το μπλοκ που περιγράφεται παρακάτω επειδή είναι ένα “DO” μπλοκ. Μπορούμε να επισυνάψουμε δηλώσεις σε αυτό. Οι δηλώσεις κάνουν πράγματα. Αυτός είναι ο λόγος για τον οποίο αυτό το μπλοκ έχει χώρο στο εσωτερικό του για να μπορούν να επισυνάπτονται μπλοκ δηλώσεων.

Μπορούμε να μετονομάσουμε τις μεταβλητές σε αυτό το μπλοκ ανά πάσα στιγμή και σε κάθε αντίστοιχο μπλοκ οπουδήποτε αλλού αναφέρεται το παλιό όνομα θα ενημερώνεται αυτόματα.

7.5 (initialize Local name to –in (return))



Εικόνα 140- initialize Local name to- in (return)

Αυτό το μπλοκ είναι μεταλλασσόμενο και μας επιτρέπει να δημιουργούμε νέες μεταβλητές οι οποίες χρησιμοποιούνται μόνο κατά τη διαδικασία που τρέχει το «return» μέρους του μπλοκ. Με αυτό τον τρόπο όλες οι μεταβλητές στη διαδικασία αυτή θα ξεκινάν με την ίδια τιμή κάθε φορά που η διαδικασία εκτελείται.

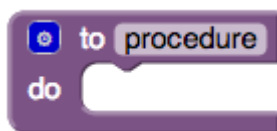
Note: Αυτό το μπλοκ διαφέρει από το μπλοκ που περιγράφηκε προηγουμένως, διότι είναι ένα «return» μπλοκ. Μπορούμε να επισυνάψουμε εκφράσεις σε αυτό. Οι εκφράσεις επιστρέφουν μια τιμή. Αυτός είναι ο λόγος για τον οποίο αυτό το μπλοκ έχει υποδοχή για να συνδέουμε εκφράσεις.

Μπορούμε να μετονομάσουμε τις μεταβλητές σε αυτό το μπλοκ ανά πάσα στιγμή και σε κάθε αντίστοιχο μπλοκ οπουδήποτε αλλού αναφέρεται το παλιό όνομα θα ενημερώνεται αυτόματα.

8. Procedures Blocks

Μια διαδικασία είναι μια σειρά από μπλοκ ή ο κωδικός ο οποίος αποθηκεύεται κάτω από ένα όνομα, το όνομα της διαδικασίας του μπλοκ μας. Αντί να χρειάζεται να κρατήσουμε μαζί την ίδια μακρά ακολουθία των μπλοκ, μπορούμε να δημιουργήσουμε μια διαδικασία και απλά να καλέσουμε το μπλοκ διαδικασίας όποτε θέλουμε να τρέξει η ακολουθία των μπλοκ. Στην επιστήμη των υπολογιστών, μια διαδικασία θα μπορούσε να ονομάζεται επίσης μια λειτουργία ή μια μέθοδος.

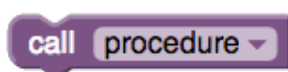
8.1 (procedure do)



Εικόνα 141- procedure do

Συγκεντρώνει μια σειρά μπλοκ μαζί σε μια ομάδα. Μπορούμε να χρησιμοποιήσουμε την ακολουθία των μπλοκ επανειλημμένα καλώντας τη διαδικασία. Εάν η διαδικασία έχει επιχειρήματα, μπορούμε να τα προσδιορίσουμε χρησιμοποιώντας το μεταλλασσόμενο κουμπί του μπλοκ. Αν κάνουμε κλικ στο μπλε σύμβολο, μπορούμε να σύρουμε επιπλέον επιχειρήματα στη διαδικασία.

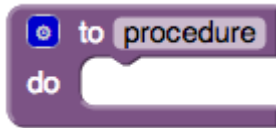
Όταν δημιουργούμε ένα νέο μπλοκ διαδικασίας, το App Inventor επιλέγει ένα μοναδικό όνομα αυτόματα. Μπορούμε να κάνουμε κλικ στο όνομα και να το αλλάξουμε. Τα ονόματα διαδικασίας σε μια εφαρμογή πρέπει να είναι μοναδικά. Το App Inventor δεν μας επιτρέπει να ορίσουμε δύο διαδικασίες στην ίδια εφαρμογή με το ίδιο όνομα. Μπορούμε να μετονομάσουμε μια διαδικασία ανά πάσα στιγμή, ενώ χτίζουμε την εφαρμογή, αλλάζοντας την ετικέτα στο μπλοκ. Το App Inventor θα μετονομάσει αυτόματα τις σχετιζόμενες κλήσεις των μπλοκ έτσι ώστε να ταιριάζουν.



Εικόνα 142- call procedure

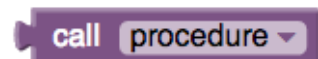
Όταν δημιουργούμε μια διαδικασία, το App Inventor δημιουργεί αυτόματα ένα μπλοκ κλήσεων και την τοποθετεί στο «My Definitions». Χρησιμοποιούμε την κλήση του μπλοκ για να κινήσουμε τη διαδικασία.

8.2 (procedure result)



Εικόνα 143- procedure result

Ίδια με μια διαδικασία «do», αλλά καλώντας τη διαδικασία αυτή επιστρέφεται ένα αποτέλεσμα.



Εικόνα 144- call procedure

Μετά τη δημιουργία αυτής της διαδικασίας, θα δημιουργηθεί ένα μπλοκ κλήσης που χρειάζεται για να προστεθεί. Αυτό συμβαίνει επειδή το αποτέλεσμα από την εκτέλεση αυτής της διαδικασίας θα επιστραφεί στο μπλοκ κλήσης και η τιμή θα περάσει σε οποιοδήποτε μπλοκ είναι συνδεδεμένο στην εσοχή.

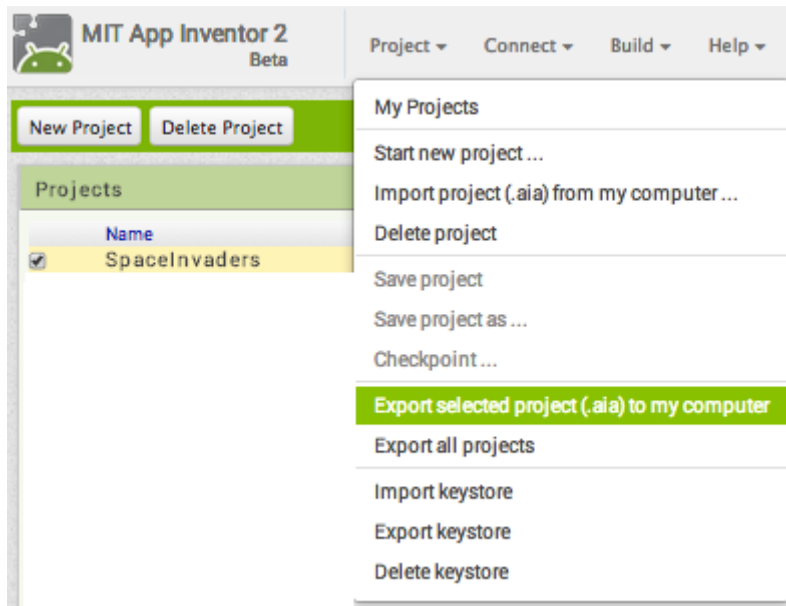
3.1.6 Πακετάρισμα και Μοίρασμα

Μπορούμε να μοιραστούμε την εφαρμογή που έχουμε δημιουργήσει σε εκτελέσιμη μορφή (.apk) όπου μπορεί να εγκατασταθεί σε μια συσκευή, ή σε μορφή πηγαίου κώδικα (.aia), όπου μπορεί να φορτωθεί στο App Inventor και να γίνουν αλλαγές. Μπορούμε, επίσης, να διανείμουμε την εφαρμογή μας στο Google Play Store.

- Μοίρασμα της εφαρμογής που δημιουργήσαμε έτσι ώστε οι άλλοι να μπορούν να την επεξεργαστούν (αρχείο .aia).

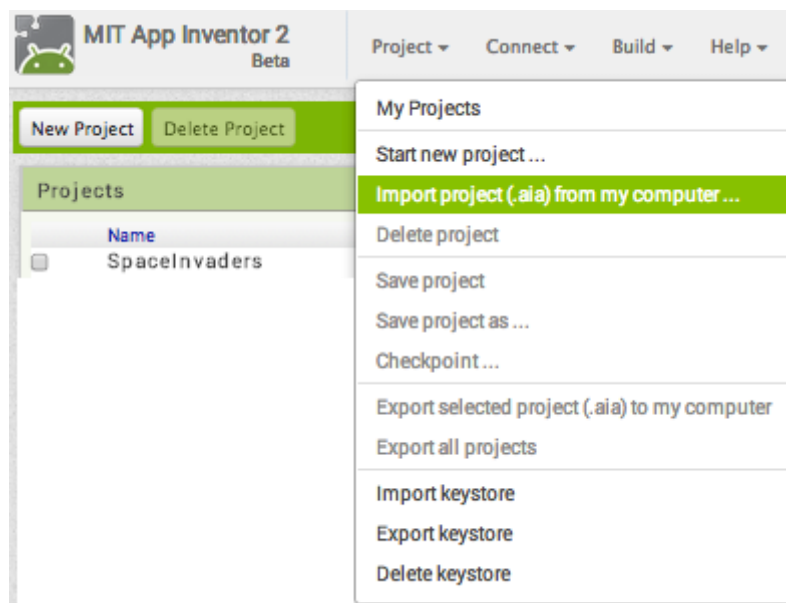
Βεβαιωνόμαστε ότι βλέπουμε την λίστα με όλα τα έργα μας (αν δεν την βλέπουμε, επιλέγουμε Projects | My Projects). Επιλέγουμε το πρόγραμμα που θέλουμε να μοιραστούμε, επιλέγοντας το πλαίσιο δίπλα σε αυτό. Επιλέγουμε το Project | Export

selected project (.aia) to my computer για να εξάγουμε τον πηγαίο κώδικα (blocks) για το έργο μας. Ο πηγαίος κώδικας φορτώνεται σε ένα αρχείο .aia.



Εικόνα 145- Εξαγωγή πηγαίου κώδικα

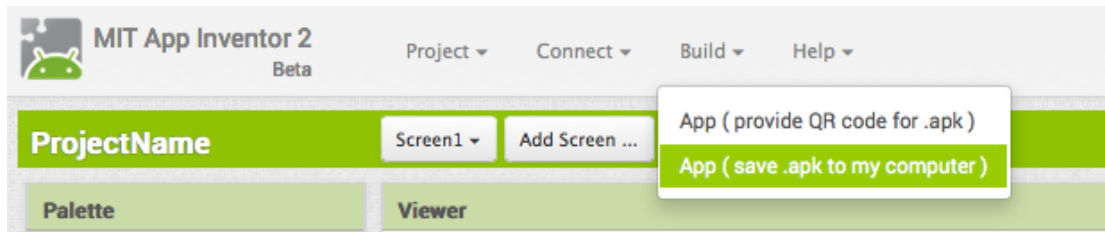
Εάν θέλουμε να το στείλουμε κάπου, μπορούμε να ανοίξουμε το αρχείο επιλέγοντας Project | Import project (.aia) from my computer.



Εικόνα 146- Εισαγωγή project από υπολογιστή

- Μοίρασμα της εφαρμογής που δημιουργήσαμε έτσι ώστε οι άλλοι να μπορούν να κάνουν εγκατάσταση στο τηλέφωνο ή το tablet τους (αρχείο .apk).

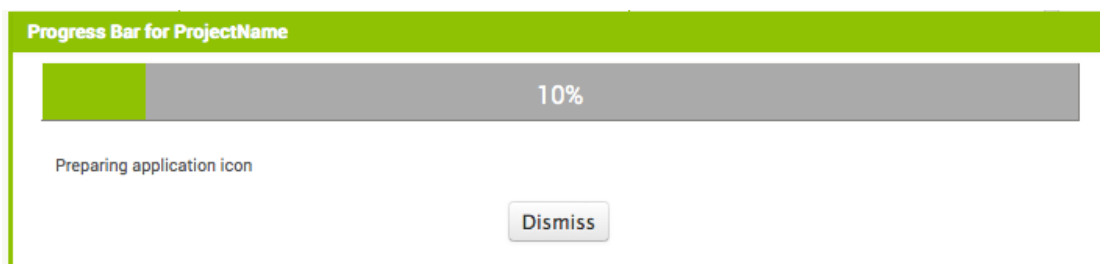
Πακετάρουμε την εφαρμογή (.apk αρχείο) πηγαίνοντας στο μενού "Build" στη γραμμή εργαλείων του app inventor.



Εικόνα 147- Πακετάρισμα εφαρμογής

Επιλέγουμε "App (save .apk to my computer)". Ένα pop-up παράθυρο θα πρέπει να μας ενημερώσει ότι η λήψη μας έχει αρχίσει.

Note: Η άλλη επιλογή (provide QR code for .apk) παράγει έναν κωδικό QR που μπορεί να σκαναριστεί και να κατεβάσει την εφαρμογή για δύο ώρες. Μπορούμε να μοιραστούμε αυτόν τον κώδικα με άλλους, αλλά θα πρέπει να το χρησιμοποιήσουν μέσα σε 2 ώρες από την λήψη του.



Εικόνα 148- Λήψη κώδικα

Όταν η κατασκευή ολοκληρωθεί, μπορούμε να στείλουμε με e-mail την εφαρμογή (".apk" file) στους φίλους μας, όπου μπορούν να την εγκαταστήσουν ανοίγοντας το e-mail από το τηλέφωνό τους. Αν θέλουμε να το διανεύουμε ευρύτερα, μπορούμε να το ανεβάσουμε σε ένα δικτυακό τόπο που τόσο εμείς όσο και ο φίλος μας μπορεί να έχει πρόσβαση. Μπορούμε, επίσης, να διανεύουμε την εφαρμογή μας στο Google Play Store.

Note: Οποιοσδήποτε εγκαταστήσει την εφαρμογή μας (η οποία είναι ένα αρχείο ".apk") θα πρέπει να αλλάξει τη ρύθμιση στο τηλέφωνό του για να επιτρέπεται η εγκατάσταση εφαρμογών εκτός αγοράς.

Για να βρούμε αυτή τη ρύθμιση για τις εκδόσεις του Android πριν από 4.0, πηγαίνουμε στο "Settings > Applications" και έπειτα επιλέγουμε το πλαίσιο δίπλα από το "Unknown Sources". Για συσκευές που τρέχουν Android 4.0 και πάνω, πηγαίνουμε στο "Settings > Security" ή στο "Settings > Security & Screen Lock" και έπειτα επιλέγουμε το πλαίσιο δίπλα στο "Unknown Sources" και επιβεβαιώνουμε την επιλογή μας.

Note: Τα αρχεία πηγαίου κώδικα (.aia) δεν είναι εκτελέσιμα προγράμματα Android - αυτά είναι αρχεία .apk. Ο πηγαίος κώδικας δεν είναι επίσης κώδικα Java SDK - μπορεί να φορτωθεί μόνο σε App Inventor.

3.2 Χρήση

Αυτό που κάνει το MIT App Inventor ξεχωριστό είναι το πώς η παγκόσμια κοινότητα χρηστών έχει αγκαλιάσει το εργαλείο και το έχει χρησιμοποιήσει με διάφορους τρόπους. Παρακάτω αναφέρονται μερικοί από τους τομείς και τους τρόπους χρήσης του MIT App Inventor.

- Εκπαίδευση. Οι μαθητές δεν εισχωρούν μόνο στην λογική του προγραμματισμού αλλά δημιουργούν και εφαρμογές που μπορούν να τους φανούν χρήσιμες και σε άλλα μαθήματα του σχολείου τους. Ο George Shao, ένα 11χρονο αγόρι από τον Καναδά, δημιούργησε μια εφαρμογή για να τον βοηθάει με τις εργασίες των μαθηματικών του. Δημιούργησε μια αριθμομηχανή όπου μπορεί να υπολογίσει τον όγκο πολλών 3D σχημάτων, την περιοχή και την περίμετρο 2D σχημάτων, καθώς και ημίτονα, συνημίτονα, και εφαπτόμενες.

Note: Σε πρόσφατη έρευνα η οποία πραγματοποιήθηκε από την οργάνωση Ανθρωπιστική δράση με Δωρεάν Λογισμικό Ανοικτού Κώδικα (Humanitarian Free and Open Source Software –HFOSS) και χρηματοδοτήθηκε από το Εθνικό Ίδρυμα Επιστημών των Ηνωμένων Πολιτειών (National Science Foundation) έγινε προσπάθεια να απαντηθεί το ερώτημα αν το app inventor είναι κατάλληλο για τη

διδασκαλία του προγραμματισμού και ειδικότερα της υπολογιστικής σκέψης στη σχολική εκπαίδευση (Morreli et al. 2011). Τα πρώτα αποτελέσματα είναι ενθαρρυντικά αφού οι ερευνητές επισημαίνουν τα εξής πλεονεκτήματα: (α) Εύκολο στη χρήση περιβάλλον με πολλές δυνατότητες (β) Αντικειμενοστραφές μοντέλο οπτικού προγραμματισμού με δομές ελέγχου καθοδηγούμενες από γεγονότα (event-driven) (γ) μάθηση μέσω της λύσης προβλημάτων (δ) Επιπλέον κίνητρα στους μαθητές σε σχέση με το Scratch και Alice εξαιτίας της φορητότητας και της πρακτικής χρήσης των εφαρμογών που δημιουργούνται (ε) ύπαρξη emulator που σημαίνει ότι δεν χρειάζονται πολλές συσκευές για την εισαγωγή στη σχολική τάξη (ζ) υποστήριξη από τη Google.

- Άθληση. Οι χρήστες μπορούν να γυμναστούν με διασκεδαστικό τρόπο. Το Bunny Bolt είναι μια εφαρμογή που αναπτύχθηκε από μια ομάδα φοιτητών του Wellesley που χρησιμοποιεί ως απειλή τα λαγουδάκια- δολοφόνους για να βγάλει τα παιδιά ηλικίας 8-14 έξω στη φύση. Το παιχνίδι βάζει τους παίκτες να κινούνται αποσκοπώντας στη σύλληψη των εικονικών κουνελιών που δραπέτευσαν από τον κακό μάγο.
- Διασκέδαση. Δημιουργία απλών παιχνιδιών καθαρά για διασκέδαση. Ο Carlos Hernandez Sangabriel από το Μεξικό χρησιμοποίησε το App Inventor για να δημιουργήσει τη δική του υλοποίηση του MasterMind, ενός δημοφιλούς παιχνιδιού παζλ, όπου δίνονται δέκα προσπάθειες για να σπάσει ένας μυστικός κώδικα που χρησιμοποιεί διαφορετικούς συνδυασμούς έξι χρωμάτων.
- Ενημέρωση. Υπάρχουν εφαρμογές που μπορούν να δημιουργηθούν μέσω του app inventor και να χρησιμοποιηθούν για την ενημέρωση των χρηστών σχετικά με διάφορα θέματα. Μια τέτοια εφαρμογή είναι το Stats Analyzer. Το Stats Analyzer είναι μια εφαρμογή η οποία δημιουργήθηκε από τον Gonzalo Arilla, ένα 17χρονο αγόρι από την Ισπανία για τους λάτρεις του NBA που τους επιτρέπει να παρακολουθούν τα στατιστικά στοιχεία για τους αγαπημένους τους ποδοσφαιριστές.
- Χρήσιμες

- Εφαρμογές της καθημερινότητας. Μπορούμε, επίσης, μέσω του app inventor να δημιουργήσουμε εφαρμογές που βρίσκουν χρήση στην καθημερινότητα και κάνουν τη ζωή μας ευκολότερη. Μια τέτοια εφαρμογή είναι αυτή του 12χρονου Arjun Kumar, η οποία μάλιστα κέρδισε και την 1η θέση στο διαγωνισμό του 2012 MIT App Inventor. Η νικητήρια εφαρμογή ονομάζεται Ez School Bus Locator. Η εφαρμογή βοηθά τους γονείς με ή χωρίς smartphones να βλέπουν ανά πάσα στιγμή την τοποθεσία του σχολικού λεωφορείου των παιδιού τους, να ελέγχουν αν το παιδί τους είναι μέσα στο λεωφορείο, και να ξέρουν πότε το σχολικό λεωφορείο θα φτάσει στον προορισμό του. Μια ακόμη τέτοια χρήσιμη εφαρμογή είναι αυτή που αναπτύχθηκε από τον Noel Zamora, ένα εργαζόμενο σε ένα εργοστάσιο της Ιταλίας. Ο Noel, λοιπόν, δημιούργησε ένα σημειωματάριο εμπνευσμένος από την ανάγκη στην δουλειά να σημειώνει κάτι γρήγορα χωρίς να πρέπει να χάνει χρόνο να βρίσκει χαρτί κ στυλό.
- Αλληλεγγύη και δημιουργικότητα. Ορίζοντας τις ανάγκες, τα μέσα και την φαντασία τους, το app inventor είναι ένας σύμμαχος των χρηστών προς δημιουργία ακόμη πιο χρήσιμων εφαρμογών. Μια ομάδα μαθητών γυμνασίου από το Los Fresnos του Τέξας σχεδίασε μια εφαρμογή για να βοηθήσει άτομα με προβλήματα όρασης έτσι ώστε να είναι ευκολότερη η περιήγηση στο σχολείο τους.

Κεφάλαιο 4

4.1 Σχεδιασμός Εφαρμογής

Το υπό ανάπτυξη σύστημα θα αποτελείται από ένα webserver ο οποίος θα διατηρεί τα στοιχεία που είναι απαραίτητα για την παροχή των υπηρεσιών και μία εφαρμογή για κινητές συσκευές η οποία θα είναι το εργαλείο για πρόσβαση του χρήστη σε αυτές τις υπηρεσίες.

Ο web server θα αποτελεί στην ουσία το back end της εφαρμογής. Θα υποστηρίζει τις λειτουργίες της εφαρμογής καθώς θα περιλαμβάνει τους μηχανισμούς διατήρησης των πληροφοριών που θα είναι προς διάθεση στους χρήστες της εφαρμογής καθώς

και του τρόπου διάθεσης αυτών. Πιο συγκεκριμένα οι λειτουργίες του θα περιλαμβάνουν:

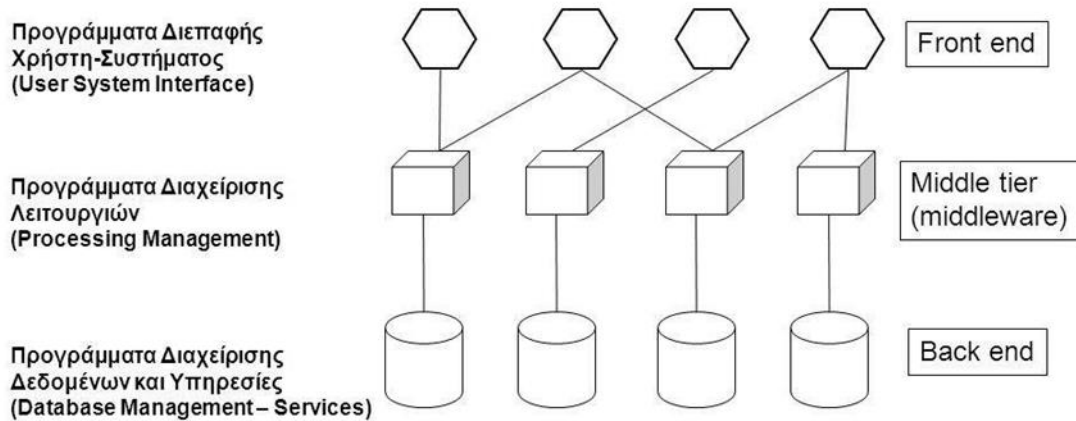
1. Διατήρηση πληροφοριών για ιστοσελίδες κατηγοριοποιημένων σε θεματολογίες.
2. Διατήρηση καταλόγου ενεργών χρηστών της εφαρμογής και των προτιμήσεων τους.
3. Διάθεση περιεχομένου στους χρήστες της εφαρμογής.

4.2 Αρχιτεκτονική Εφαρμογής

Η σχεδίαση της εφαρμογής βασίζεται στα πρότυπα της αρχιτεκτονικής των τριών επιπέδων (αρχιτεκτονική Three Tier) και εγκαθίσταται στο δίκτυο υπολογιστών με χρήση του πρωτοκόλλου επικοινωνίας TCP/IP (Transmission Control Protocol/Internet Protocol= Πρωτόκολλο Ελέγχου Μετάδοσης και πρωτόκολλο του Internet). Το σύστημα αποτελείται από τρία διακριτά επίπεδα, που στην γενική περίπτωση, υλοποιούνται σε διαφορετικές μηχανές:

1. Το υψηλότερο επίπεδο περιλαμβάνει την διεπαφή του πελάτη (**επίπεδο παρουσίασης/ client**).
2. Το ενδιάμεσο επίπεδο περιέχει την επιχειρησιακή λογική (**επίπεδο εφαρμογής/ application server**).
3. Το κατώτερο επίπεδο διαχειρίζεται τα δεδομένα που χρησιμοποιούνται από την εφαρμογή (**επίπεδο δεδομένων/database server**).

Βαθμοτή Αρχιτεκτονική Τεχνοτροπία Τριών Επιπέδων (Three Tier) - Σχηματικά

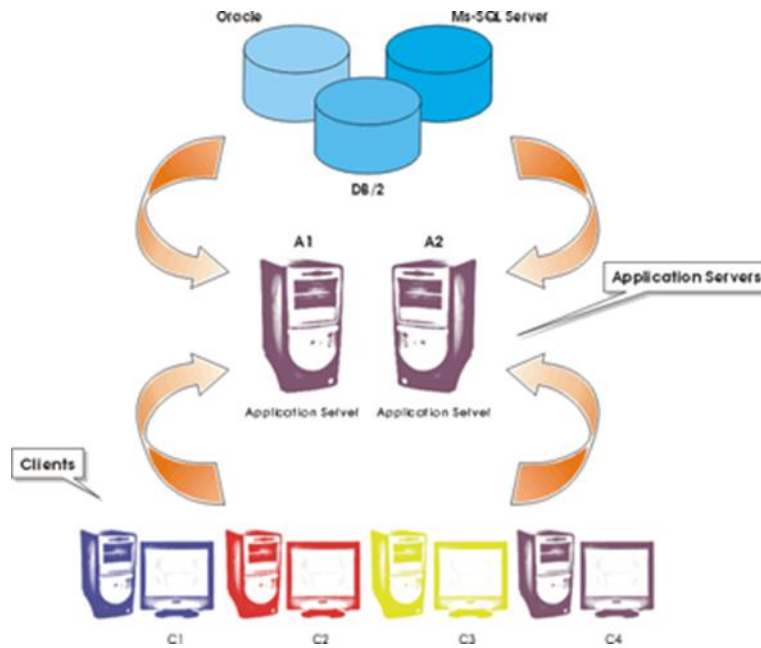


Σχήμα 1- Βαθμοτή Αρχιτεκτονική Τεχνοτροπία Τριών Επιπέδων

Αναλυτικότερα:

1. Υψηλότερο Επίπεδο- Client

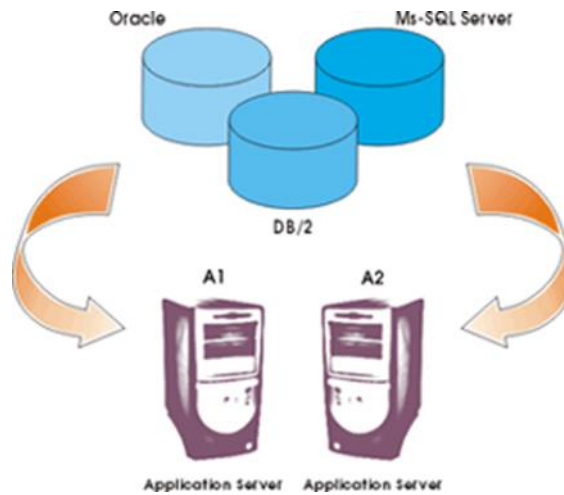
Το υψηλότερο επίπεδο του λογισμικού αποτελεί η επαφή του χρήστη με το σύστημα (User Interface). Στο επίπεδο αυτό, πραγματοποιείται η διαχείριση των Οθονών Εργασίας (User Screens) καθώς επίσης και η μορφοποίηση των δεδομένων που εμφανίζονται. Η επικοινωνία του Client με τον Application ή τους Application Servers πραγματοποιείται κάνοντας χρήση ενός μόνο πακέτου δεδομένων κάθε φορά. Έτσι, επιτυγχάνεται ο βέλτιστος χρόνος απόκρισης μεταξύ του Client και του Application Server, δεδομένου ότι τα δυο αυτά επίπεδα μπορούν να λειτουργήσουν πάνω σε μια τηλεπικοινωνιακή γραμμή (Leased Line, Dialup, Internet Connection), εξασφαλίζοντας έτσι μικρούς χρόνους απόκρισης σε όλο το σύστημα.



Σχήμα 2- Υψηλότερο Επίπεδο- Client

2. Ενδιάμεσο Επίπεδο- Application Server

Το ενδιάμεσο επίπεδο αποτελεί το κύριο τμήμα του λογισμικού, στο οποίο εκτελούνται οι περισσότερες λειτουργίες, εκτός εκείνων που σχετίζονται με τη διαμόρφωση των οθονών εργασίας. Υπάρχει δυνατότητα εγκατάστασης περισσότερων του ενός Application Servers σε διαφορετικά μηχανήματα, αξιοποιώντας, με τον τρόπο αυτό, οποιαδήποτε διαθέσιμη υπολογιστική ισχύ και εξασφαλίζοντας εξαιρετικά αποτελέσματα ανταπόκρισης, αξιοπιστίας και επεκτασιμότητας.

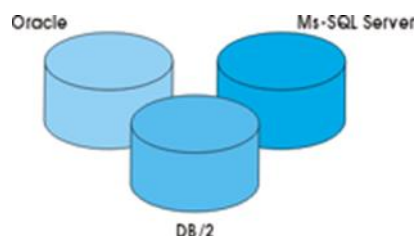


Σχήμα 3- Ενδιάμεσο Επίπεδο- Application Server

Με την κατανομή των Application Servers σε ανεξάρτητα μηχανήματα, επιτυγχάνεται αποσυμφόρηση του συνολικού φόρτου του συστήματος, αφού κάθε Application Server είναι σε θέση να υποστηρίξει ένα υποσύνολο του συνολικού αριθμού των Remote Clients (π.χ. Ο Application Server A θα εξυπηρετεί τους Clients του υποκαταστήματος A, ενώ ο Application Server B θα εξυπηρετεί τους Clients του υποκαταστήματος B).

3. Κατώτερο Επίπεδο- Database Server

Αποτελώντας το κατώτερο επίπεδο του συστήματος, ο Database Server παρέχει όλες τις απαραίτητες λειτουργίες για την αποθήκευση, ανάκτηση, ενημέρωση και συντήρηση των δεδομένων του συστήματος καθώς επίσης και όλους τους απαραίτητους μηχανισμούς για την ακεραιότητα των δεδομένων (Data Integrity).



Σχήμα 4- Κατώτερο Επίπεδο- Database Server

Η αρχιτεκτονική Client - Server τριών επιπέδων (Three Tier) έχει διεθνώς αποδειχθεί ως η πλέον κατάλληλη για δικτυακές εγκαταστάσεις, σε αντίθεση με την

αρχιτεκτονική Client - Server δύο επιπέδων (Two Tier), είτε Fat-Client, είτε Fat-Server.

Συνηθίζεται, επίσης, σε Web εφαρμογές, όπου το ενδιάμεσο επίπεδο είναι ένας Web Server που αποθηκεύει το λογισμικό σύνδεσης και το τμήμα της λογικής της εφαρμογής που χρησιμοποιείται για προσπέλαση των αντίστοιχων δεδομένων από τον DB Server.

4.3 Πλεονεκτήματα αρχιτεκτονικής 3 επιπέδων

Η συγκρότηση του συστήματος σε τρία επίπεδα εξασφαλίζει:

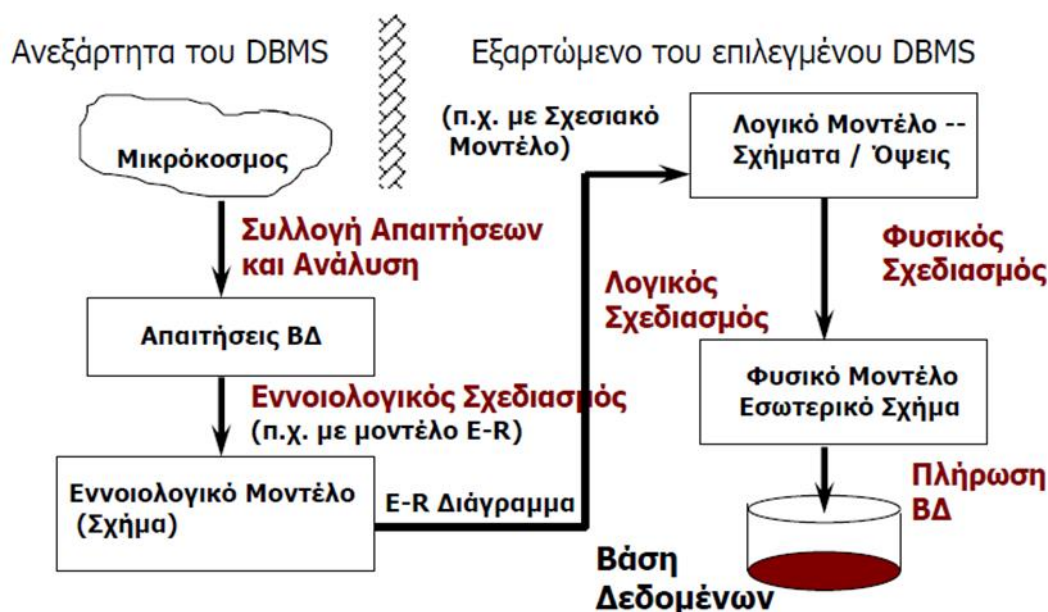
- Την ελαχιστοποίηση της επιβάρυνσης του δικτύου λόγω μεταφοράς μεγάλου όγκου δεδομένων π.χ. η εκτέλεση ενός Query για την ανάκτηση μερικών εγγραφών από έναν πίνακα με δεκάδες χιλιάδες εγγραφές γίνεται στο διακομιστή εφαρμογής (Application Server), από τον οποίο μεταφέρεται στο χρήστη μόνο το αποτέλεσμα.
- Τη δυνατότητα διαχωρισμού του διακομιστή δεδομένων (Database Server) από το διακομιστή ή τους διακομιστές εφαρμογής (Application Servers), ώστε να εκτελούνται σε διαφορετικά μηχανήματα. Κατά συνέπεια, ο καθορισμός των κρίσιμων μεγεθών απόδοσης των αντίστοιχων μηχανών (sizing) μπορεί να γίνεται ανεξάρτητα, ενώ παράλληλα εξασφαλίζεται απεριόριστη επεκτασιμότητα, χωρίς ανακατασκευή, του λογισμικού.
- Τη μέγιστη ευελιξία στην επιλογή του διακομιστή δεδομένων, καθώς επιτρέπεται η χρήση οποιουδήποτε μηχανήματος με οποιοδήποτε λειτουργικό σύστημα (π.χ. Windows NT ή UNIX κλπ), με μοναδική απαίτηση τη δυνατότητα επικοινωνίας δια μέσου TCP/IP πρωτοκόλλου. Έτσι, είναι δυνατή η μεταγενέστερη αναβάθμιση ως προς τη βάση δεδομένων με την αλλαγή/ αναβάθμιση του μηχανήματος, χωρίς να επηρεάζεται το υπόλοιπο σύστημα.
- Την ενίσχυση της ασφάλειας των δεδομένων (προσπέλαση στον εξυπηρετητή της βάσης δεδομένων μόνο μέσω του ενδιάμεσου επιπέδου).
- Την ανάπτυξη του μεσαίου επιπέδου με τεχνολογίες όπως η αντικειμενοστραφής ή η βασισμένη σε ψηφίδες λογισμικού, ενισχύει την δυνατότητα επαναχρησιμοποίησης τμημάτων του λογισμικού.

4.4 Σχεδίαση Βάσης Δεδομένων (Database)

Η διαδικασία της σχεδίασης μιας βάσης δεδομένων είναι η διαδικασία μετατροπής της γνώσης που έχουμε για έναν μικρόκοσμο σε μια βάση δεδομένων την οποία θα διαχειρίζεται ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ).

Για εννοιολογική διευκόλυνση της όλης διαδικασίας, αυτή διακρίνεται σε διαδοχικές αυτοτελείς διαδικασίες (φάσεις):

1. Συλλογή και Ανάλυση Απαιτήσεων (Requirements Collection and Analysis)
2. Εννοιολογικός Σχεδιασμός ΒΔ (Conceptual Database Design)
3. Λογικός Σχεδιασμός ΒΔ (Logical Database Design)
4. Φυσικός Σχεδιασμός ΒΔ (Physical Database Design)



Σχήμα 5- Σχεδίαση Βάσης Δεδομένων

1. Συλλογή & Ανάλυση Απαιτήσεων

- Η εφαρμογή περιέχει κάποιες κατηγορίες. Για τις κατηγορίες μας ενδιαφέρει ο μοναδικός κωδικός που τις χαρακτηρίζει (code) και η ονομασία τους (title).

- Ο χρήστες οι οποίοι χρησιμοποιούν την εφαρμογή. Για τους χρήστες μας ενδιαφέρει να γνωρίζουμε τον μοναδικό κωδικό που τους χαρακτηρίζει (code), το ονοματεπώνυμό τους, το όνομα χρήστη που έχει δηλώσει για την είσοδό του στο σύστημα (username), τον κωδικό πρόσβασης που χρησιμοποιεί για να εισέλθει στο σύστημα (password), τον λογαριασμό ηλεκτρονικού ταχυδρομείου που χρησιμοποίησε κατά την εγγραφή του στην εφαρμογή (e-mail), το φύλο του (sex) και τέλος, την ημερομηνία γέννησής του (birthday).
- Η εφαρμογή περιέχει ιστοσελίδες. Για τις ιστοσελίδες μας ενδιαφέρει ο μοναδικός κωδικός που τις χαρακτηρίζει (code), η ηλεκτρονική διεύθυνση στην οποία βρίσκεται (url), η ημερομηνία εισαγωγής της στο σύστημα (insertdate) και η διάρκεια διατήρησής της στο σύστημα (duration).
- Οι χρήστες επιλέγουν κάποιες κατηγορίες. Για τις κατηγορίες χρηστών μας ενδιαφέρει ο χρήστης (user) και οι κατηγορίες που έχει επιλέξει (category).
- Οι κατηγορίες περιέχουν κάποιες ιστοσελίδες. Για τις κατηγορίες των ιστοσελίδων μας ενδιαφέρει η κατηγορία στην οποία ανήκουν (category) και οι ιστοσελίδες που ανήκουν σε κάθε κατηγορία (webpage).

2. Εννοιολογικός Σχεδιασμός Βάσης Δεδομένων

Οι οντότητες (έννοιες) που συμμετέχουν στην οργάνωση των δεδομένων είναι:

1. Χρήστες (USERS): Αντιπροσωπεύει τους χρήστες της εφαρμογής που θα χρειάζονται να αντλήσουν πληροφορίες καθώς και τους διαχειριστές του web server. Τα χαρακτηριστικά της είναι:
 - Κωδικός (CODE): Μοναδικός κωδικός αριθμός που τους χαρακτηρίζει.
 - Ονοματεπώνυμο (FULLNAME): Το ονοματεπώνυμο του χρήστη.
 - Είδος (TYPE): Το είδος του χρήστη (απλός χρήστης ή διαχειριστής).
 - Username (USERNAME): Το username που χρησιμοποιεί ο χρήστης για είσοδο στο σύστημα.
 - Password (PASSWORD): Το password που χρησιμοποιεί ο χρήστης για είσοδο στο σύστημα.
 - Email: Το email που χρησιμοποίησε ο χρήστης για την εγγραφή του.

- Sex: Το φύλο του χρήστη, αρσενικό ή θηλυκό.
 - Birthday: Η ημερομηνία γέννησης του χρήστη.
2. Κατηγορίες (CATEGORIES): Αντιπροσωπεύει τις κατηγορίες των πληροφοριών που διατίθενται από την εφαρμογή. Τα χαρακτηριστικά της είναι:
- Κωδικός (CODE): Μοναδικός κωδικός αριθμός που την χαρακτηρίζει.
 - Ονομασία (TITLE): Η ονομασία της κατηγορίας.
3. Ιστοσελίδα (WEBPAGE): Αντιπροσωπεύει μία ιστοσελίδα. Τα χαρακτηριστικά της είναι:
- Κωδικός (CODE): Μοναδικός κωδικός που χαρακτηρίζει την ιστοσελίδα.
 - URL (URL): Το url της ιστοσελίδας.
 - Ημερομηνία Εισαγωγής (INSERTDATE): Η ημερομηνία εισόδου της ιστοσελίδας στο σύστημα.
 - Διάρκεια Διατήρησης (DURATION): Το χρονικό διάστημα διατήρησης της ιστοσελίδας στο σύστημα σε ημέρες.
4. Κατηγορίες Χρηστών (USERSCATEGORIES): Αντιπροσωπεύει τις κατηγορίες πληροφοριών που έχουν επιλεγεί από τον χρήστη. Τα χαρακτηριστικά της είναι:
- Χρήστης (USER): Δηλώνει τον χρήστη.
 - Κατηγορία (CATEGORY): Σε ποια κατηγορία αντιστοιχίζεται ο συγκεκριμένος χρήστης.
5. Κατηγορίες Ιστοσελίδων (WEBPAGESCATEGORIES): Αντιπροσωπεύει τις ιστοσελίδες που συγκαταλέγονται σε κάθε κατηγορία. Τα χαρακτηριστικά της είναι:
- Κατηγορία (CATEGORY): Δηλώνει την κατηγορία.
 - Ιστοσελίδα (WEBPAGE): Η ιστοσελίδα/ες που αντιστοιχίζονται με τις κατηγορίες.

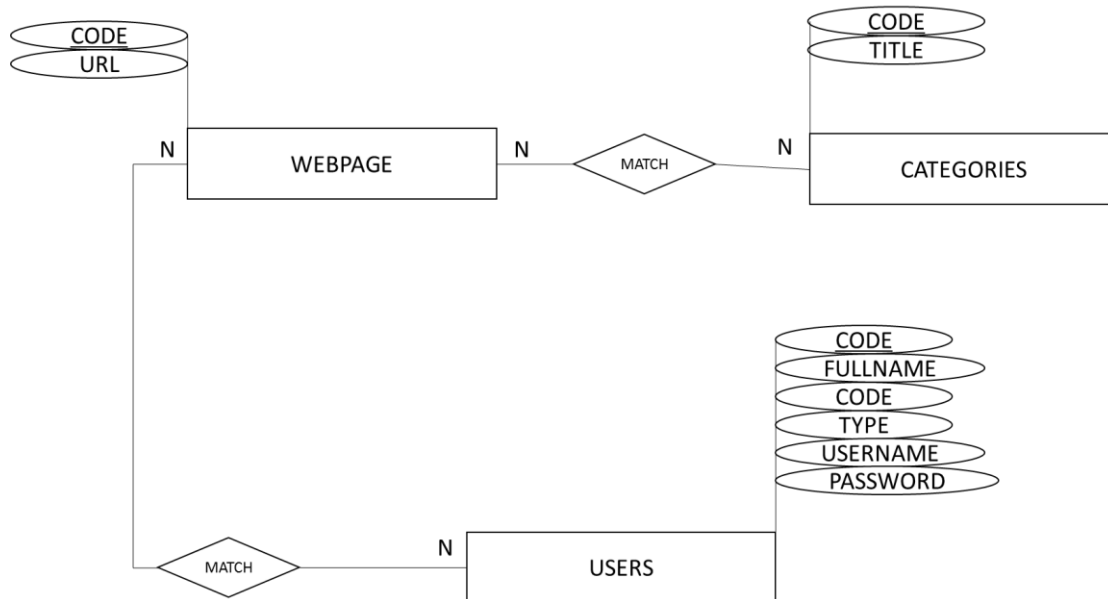
Οι σχέσεις μεταξύ των οντοτήτων περιγράφονται στον επόμενο πίνακα:

Πίνακας 9- Σχέσεις Οντοτήτων

ΟΝΤΟΤΗΤΕΣ	ΣΥΣΧΕΤΙΣΗ	ΠΕΡΙΓΡΑΦΗ
USERS CATEGORIES	MATCH	Κάθε χρήστης προτιμάει μία ή περισσότερες κατηγορίες και μία κατηγορία μπορεί να προτιμάται από έναν ή περισσότερους χρήστες. Συσχέτιση: Πολλά προς Πολλά.
CATEGORIES WEBPAGE	MATCH	Κάθε ιστοσελίδα μπορεί να αναφέρεται σε μία ή περισσότερες κατηγορία και μία κατηγορία μπορεί να περιλαμβάνει μία ή περισσότερες ιστοσελίδες. Συσχέτιση: Πολλά προς Πολλά.

3. Λογικός Σχεδιασμός Βάσης Δεδομένων

Το παραπάνω εννοιολογικό μοντέλο περιγράφεται από το ακόλουθο διάγραμμα οντοτήτων συσχετίσεων.



Σχήμα 6- Διάγραμμα Οντοτήτων- Συσχετίσεων

Σχεσιακό Μοντέλο

Το Εννοιολογικό Μοντέλο μετασχηματίζεται σε σχεσιακό ως εξής:

Η οντότητα USERS μετασχηματίζεται στον πίνακα USERS με τα παρακάτω πεδία.

Πίνακας 10- Πεδία του πίνακα Users

CODE	INTEGER	Μοναδικός κωδικός αριθμός που χαρακτηρίζει τους χρήστες. Πρωτεύον κλειδί.
FULLNAME	VARCHAR(60)	Το ονοματεπώνυμο του χρήστη.
TYPE	CHAR(1)	Το είδος του χρήστη (απλός χρήστης ή διαχειριστής ή ακαθόριστος).
USERNAME	VARCHAR(12)	Το username που χρησιμοποιεί ο χρήστης

		για είσοδο στο σύστημα.
PASSWORD	VARCHAR(100)	Το password που χρησιμοποιεί ο χρήστης για είσοδο στο σύστημα.

Η οντότητα CATEGORIES μετασχηματίζεται στον πίνακα CATEGORIES με πεδία αυτά που φαίνονται παρακάτω.

Πίνακας 11- Πεδία του πίνακα Categories

CODE	INTEGER	Μοναδικός κωδικός αριθμός που χαρακτηρίζει την κατηγορία. Πρωτεύον κλειδί.
TITLE	VARCHAR(20)	Η ονομασία της κατηγορίας.

Η οντότητα WEBPAGE μετασχηματίζεται στον πίνακα WEBPAGE με πεδία αυτά τα οποία παρατίθενται στο παρακάτω πίνακα:

Πίνακας 12- Πεδία του πίνακα Webpage

CODE	INTEGER	Μοναδικός κωδικός που χαρακτηρίζει μοναδικά την ιστοσελίδα. Πρωτεύον κλειδί.
URL	VARCHAR(100)	Το url της ιστοσελίδας.
INSERTDATE	DATE	Η ημερομηνία εισόδου της ιστοσελίδας στο σύστημα.
DURATION	INTEGER	Το χρονικό διάστημα διατήρησης της ιστοσελίδας στο σύστημα σε ημέρες.

--	--	--

Η συσχέτιση MATCH (USERS–CATEGORIES) μετασχηματίζεται στον πίνακα USERSCATEGORIES με πεδία που φαίνονται παρακάτω:

Πίνακας 13- Πεδία του πίνακα UsersCategories

USER	INTEGER	Ο συσχετιζόμενος με την κατηγορία χρήστης. Συμμετέχει στο πρωτεύον κλειδί. Ξένο κλειδί στον πίνακα USERS στο πεδίο CODE.
CATEGORY	INTEGER	Η συσχετιζόμενη κατηγορία. Συμμετέχει στο πρωτεύον κλειδί. Ξένο κλειδί στον πίνακα CATEGORIES στο πεδίο CODE.

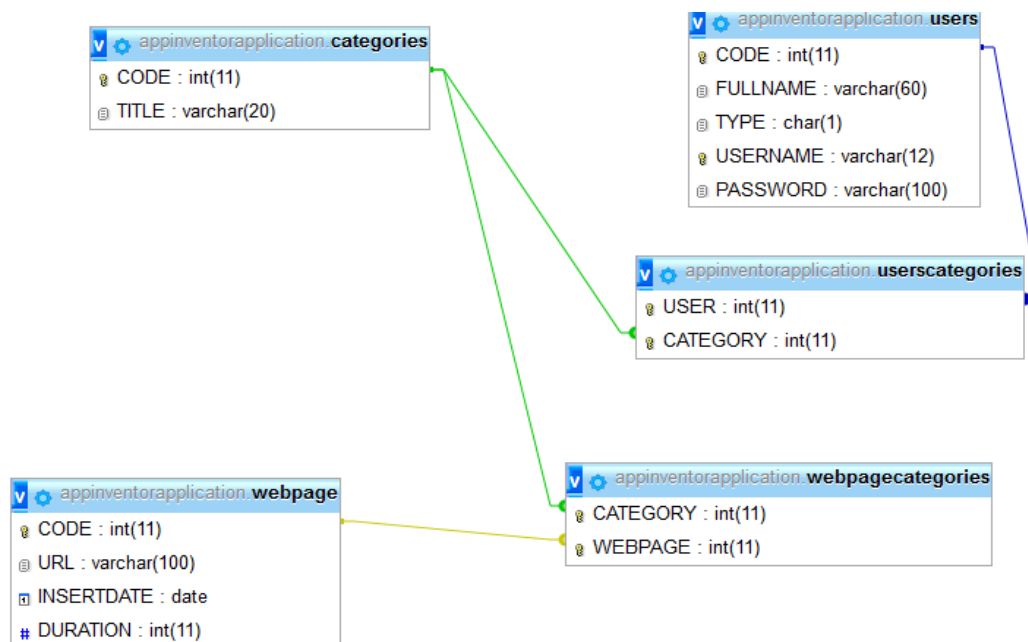
Η συσχέτιση MATCH (WEBPAGE–CATEGORIES) μετασχηματίζεται στον πίνακα WEBPAGECATEGORIES με πεδία αυτά που φαίνονται παρακάτω:

Πίνακας 14- Πεδία του πίνακα WebpageCategories

WEBPAGE	INTEGER	Η συσχετιζόμενη ιστοσελίδα. Συμμετέχει στο πρωτεύον κλειδί. Ξένο κλειδί στον πίνακα WEBPAGE στο πεδίο CODE.
CATEGORY	INTEGER	Η συσχετιζόμενη κατηγορία. Συμμετέχει στο πρωτεύον κλειδί. Ξένο κλειδί στον πίνακα

		CATEGORIES στο πεδίο CODE.
--	--	----------------------------

Το σχεσιακό μοντέλο αποτυπώνεται στο παρακάτω σχεδιάγραμμα



Σχήμα 7- Σχεσιακό Μοντέλο

4.5 Επίπεδο Λειτουργικότητας

4.5.1 Περιπτώσεις χρήσης

Στο επίπεδο αυτό προσδιορίζεται η λειτουργικότητα του webserver. Αρχικά πρέπει να προσδιοριστούν οι χρήστες που θα τον χρησιμοποιούν καθώς και οι λειτουργίες που χρειάζεται να εκτελεί.

Οι πιθανοί χρήστες είναι:

1. Η εφαρμογή για κινητές συσκευές που έχει αναπτυχθεί.
2. Οι διαχειριστές του webserver.

Οι λειτουργίες που χρειάζεται να εκτελεί κάθε ένας από αυτούς είναι:

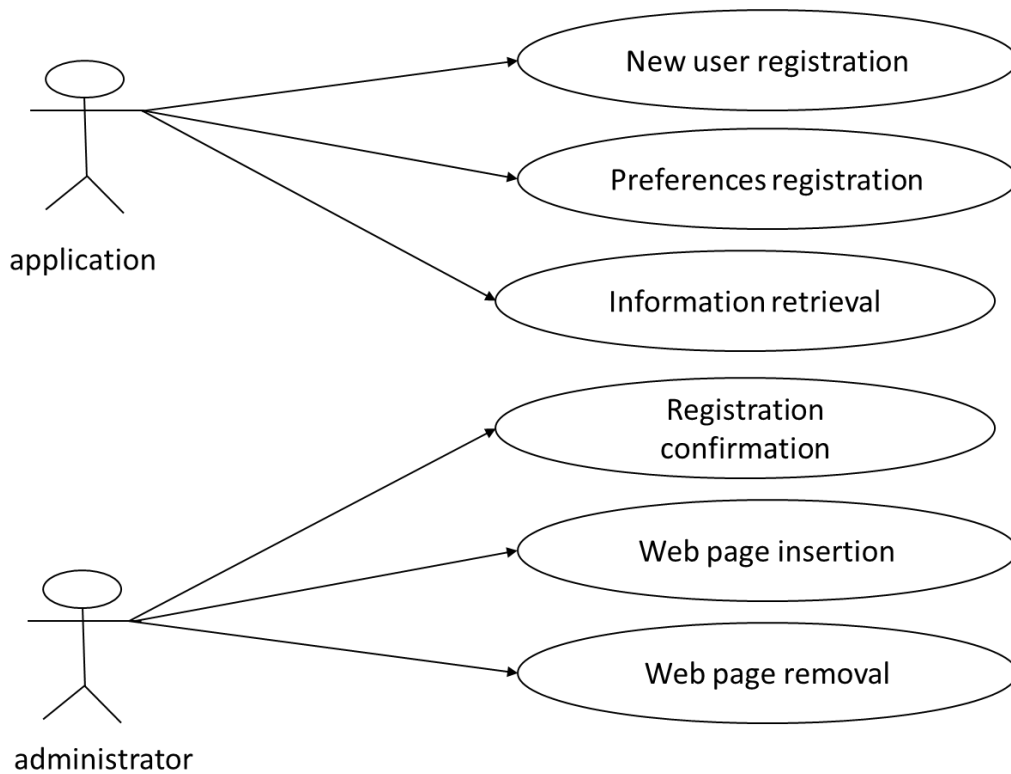
1. Εφαρμογή για κινητές συσκευές:

- Εγγραφή νέου χρήστη: Στην περίπτωση αυτή οι χρήστες της κινητής συσκευής χρησιμοποιούν την δυνατότητα που τους παρέχεται από την εφαρμογή για να εγγραφούν στην υπηρεσία.
- Καταχώρηση προτιμήσεων χρήστη: Οι χρήστες χρησιμοποιούν την εφαρμογή για κινητή συσκευή προκειμένου να δηλώσουν την προτίμησή τους σε θεματολογία για ενημέρωση.
- Ανάκτηση πληροφοριών: Οι χρήστες αιτούνται, μέσω της εφαρμογής για κινητές συσκευές, πληροφορίες σχετικές με την θεματολογία που έχουν επιλέξει.

2. Διαχειριστές web server:

- Έγκριση εγγραφής χρήστη: Οι διαχειριστές ελέγχουν την αίτηση εγγραφής του υποψήφιου μέλους και την κάνουν δεκτή ή όχι.
- Καταχώρηση ιστοσελίδας: Οι διαχειριστές προθέτουν μία νέα ιστοσελίδα στο σύστημα αποδίδοντας την σε μία ή περισσότερες κατηγορίες.
- Διαγραφή ιστοσελίδας: Οι διαχειριστές αφαιρούν μία ιστοσελίδα από τις διαθέσιμες του συστήματος.

Οι παραπάνω προδιαγραφές φαίνονται στο παρακάτω διάγραμμα use-case.



Σχήμα 8- Διάγραμμα use- case

4.5.2 Κλάσεις Αντικείμενων

Στην λειτουργικότητα της εφαρμογής συμμετέχουν αντικείμενα από τις ακόλουθες κλάσεις αντικειμένων:

- Administrator: Αντιπροσωπεύει έναν διαχειριστή του συστήματος. Τα χαρακτηριστικά του είναι:

1. Κωδικός: Μοναδικός κωδικός που τον χαρακτηρίζει.
2. Fullname: Το ονοματεπώνυμο του διαχειριστή.
3. Username: Το username που χρησιμοποιεί ο διαχειριστής για είσοδο στο σύστημα.
4. Password: Το password που χρησιμοποιεί ο διαχειριστής για είσοδο στο σύστημα.

Οι λειτουργίες που καθορίζουν την συμπεριφορά των αντικειμένων της κλάσης είναι:

1. setCode: Καταχωρεί τον κωδικό του χρήστη.
2. getCode: Επιστρέφει τον κωδικό του χρήστη.

3. setFullname: Καταχωρεί το ονοματεπώνυμο του χρήστη.
 4. getFullname: Επιστρέφει το ονοματεπώνυμο του χρήστη.
 5. setUsername: Καταχωρεί το username του χρήστη.
 6. getUsername: Επιστρέφει το username του χρήστη.
 7. setPassword: Καταχωρεί το password του χρήστη.
 8. getPassword: Επιστρέφει το password του χρήστη.
 9. login: Ελέγχει αν τα credentials του χρήστη είναι σωστά και ανάλογα επιτρέπει την είσοδο στην λειτουργικότητα του webserver.
 10. get: Ανακτά από την βάση δεδομένων τα στοιχεία του χρήστη και τα αποδίδει στα χαρακτηριστικά του αντικειμένου.
 11. set: Καταχωρεί στην βάση δεδομένων τα χαρακτηριστικά του αντικειμένου.
 12. update: Ενημερώνει την βάση δεδομένων με τα χαρακτηριστικά του αντικειμένου.
 13. delete: Διαγράφει τα στοιχεία του χρήστη από την βάση δεδομένων.
- Web Page: Αντιπροσωπεύει μία ιστοσελίδα. Τα χαρακτηριστικά της είναι:
 1. code: Ο μοναδικός κωδικός που αντιστοιχεί στην ιστοσελίδα.
 2. url: Το url της ιστοσελίδας.
 3. category: Η κατηγορία στην οποία ανήκει η ιστοσελίδα.
 4. Insertdate: Η ημερομηνία καταγραφής της ιστοσελίδας.
 5. Duration: Η διάρκεια εγκυρότητας της ιστοσελίδας σε μέρες.

Οι μέθοδοι που προσδιορίζουν την συμπεριφορά της είναι:

1. setCode: Καταχωρεί τον κωδικό της ιστοσελίδας.
2. getCode: Επιστρέφει τον κωδικό της ιστοσελίδας.
3. setUrl: Καταχωρεί το url της ιστοσελίδας.
4. getUrl: Επιστρέφει το url της ιστοσελίδας.
5. setInsertdate: Καταχωρεί την ημερομηνία καταχώρησης της ιστοσελίδας.
6. getInsertdate: Επιστρέφει την ημερομηνία καταχώρησης της ιστοσελίδας.
7. setDuration: Καταχωρεί τη διάρκεια εγκυρότητας της ιστοσελίδας.
8. getDuration: Επιστρέφει τη διάρκεια εγκυρότητας της ιστοσελίδας.
9. setCategory: Καταχωρεί κατηγορία.
10. getCategory: Επιστρέφει την κατηγορία στην οποία ανήκει η ιστοσελίδα.
11. set: Καταχωρεί τα χαρακτηριστικά του αντικειμένου στην βάση δεδομένων.

12. get: Επιστρέφει τα χαρακτηριστικά του αντικειμένου από την βάση δεδομένων.
 13. update: Ενημερώνει τα χαρακτηριστικά του αντικειμένου στην βάση δεδομένων.
 14. delete: Διαγράφει την ιστοσελίδα από την βάση δεδομένων.
- Category: Αντιπροσωπεύει τη κατηγορία των ιστοσελίδων. Τα χαρακτηριστικά της είναι:
 1. Code: Είναι ο μοναδικός κωδικός της κατηγορίας.
 2. Title: Είναι η ονομασία της κατηγορίας.

Η συμπεριφορά των αντικειμένων προσδιορίζονται από τις ακόλουθες μεθόδους:

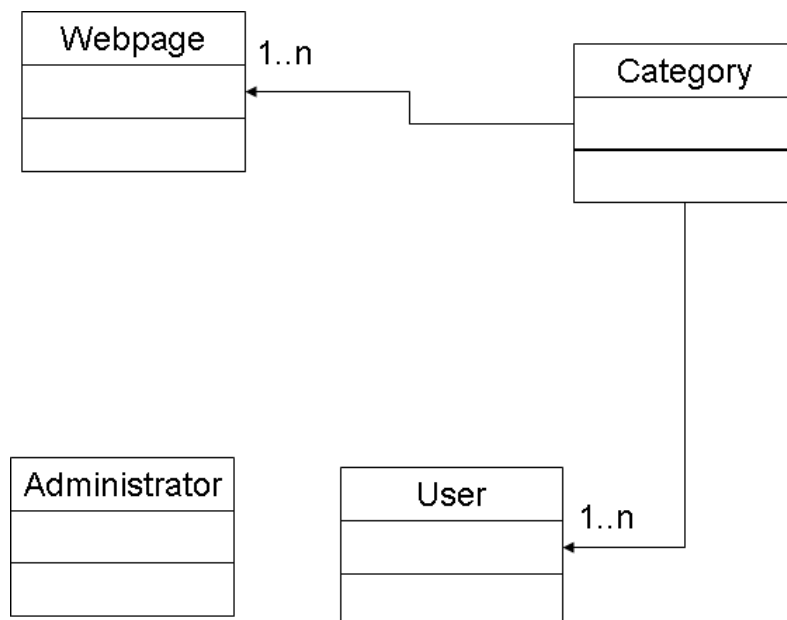
1. setCode: Καταχωρεί τον κωδικό του αντικειμένου.
 2. getCode: Επιστρέφει τον κωδικό του αντικειμένου.
 3. setTitle: Καταχωρεί την ονομασία της κατηγορίας.
 4. getTitle: Επιστρέφει την ονομασία της κατηγορίας.
 5. set: Καταγράφει την κατηγορία στην βάση δεδομένων.
 6. get: Καταχωρεί τα χαρακτηριστικά του αντικειμένου όπως ανακτώνται από την βάση δεδομένων.
 7. update: Ενημερώνει την βάση δεδομένων με τα χαρακτηριστικά του αντικειμένου.
 8. delete: Διαγράφει τα στοιχεία του αντικειμένου από την βάση δεδομένων.
- User: Αντιπροσωπεύει έναν χρήστη του συστήματος. Τα χαρακτηριστικά του είναι:
 1. Κωδικός: Μοναδικός κωδικός που τον χαρακτηρίζει.
 2. Fullname: Το ονοματεπώνυμο του χρήστη.
 3. Username: Το username που χρησιμοποιεί ο χρήστης για είσοδο στο σύστημα.
 4. Password: Το password που χρησιμοποιεί ο χρήστης για είσοδο στο σύστημα.
 5. Categories: Το σύνολο των κατηγοριών που ενδιαφέρουν τον χρήστη.

Οι λειτουργίες που καθορίζουν την συμπεριφορά των αντικειμένων της κλάσης είναι:

1. setCode: Καταχωρεί τον κωδικό του χρήστη.
2. getCode: Επιστρέφει τον κωδικό του χρήστη.
3. setFullname: Καταχωρεί το ονοματεπώνυμο του χρήστη.

4. `getFullname`: Επιστρέφει το ονοματεπώνυμο του χρήστη.
5. `setUsername`: Καταχωρεί το `username` του χρήστη.
6. `getUsername`: Επιστρέφει το `username` του χρήστη.
7. `setPassword`: Καταχωρεί το `password` του χρήστη.
8. `getPassword`: Επιστρέφει το `password` του χρήστη.
9. `setCategory`: Καταχωρεί κατηγορία.
10. `getCategory`: Επιστρέφει κατηγορία στην οποία ανήκει η ιστοσελίδα.
11. `login`: Ελέγχει αν τα `credentials` του χρήστη είναι σωστά και ανάλογα επιτρέπει την είσοδο στην λειτουργικότητα του `webserver`. Αν η είσοδος είναι επιτυχής επιστρέφεται κατάλογος με τις ιστοσελίδες που αντιστοιχούν στις κατηγορίες ενδιαφέροντος του χρήστη.
12. `get`: Ανακτά από την βάση δεδομένων τα στοιχεία του χρήστη και τα αποδίδει στα χαρακτηριστικά του αντικειμένου.
13. `set`: Καταχωρεί στην βάση δεδομένων τα χαρακτηριστικά του αντικειμένου.
14. `update`: Ενημερώνει την βάση δεδομένων με τα χαρακτηριστικά του αντικειμένου.
15. `delete`: Διαγράφει τα στοιχεία του χρήστη από την βάση δεδομένων.

Η παραπάνω περιγραφή απεικονίζεται στο ακόλουθο διάγραμμα κλάσεων

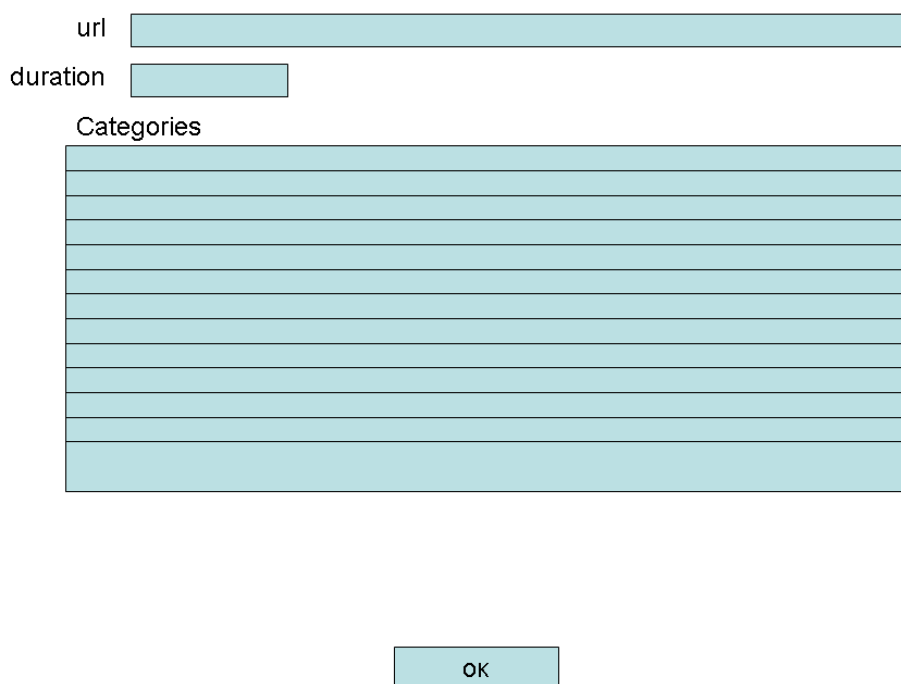


Σχήμα 9- Διάγραμμα κλάσεων

4.5.3 Σχεδίαση Διεπαφών

Καταχώρηση Ιστοσελίδας

Η διεπαφή για την λειτουργία αυτή αποτελείται από ένα textbox στο οποίο ο διαχειριστής θα καταχωρεί το url της ιστοσελίδας, ένα textbox για την καταχώρηση της επιθυμητής διάρκειας εγκυρότητας της ιστοσελίδας και ένα listbox στο οποίο θα μπορεί να επιλέγει τις κατηγορίες στις οποίες μπορεί να ανήκει. Στην συνέχεια τα στοιχεία αυτά θα καταχωρούνται στον server κάνοντας κλικ στο OK.



The image shows a web form with the following elements:

- A label "url" followed by a long text input field.
- A label "duration" followed by a shorter text input field.
- A label "Categories" above a list box containing 10 empty rows.
- An "OK" button centered below the form.

Εικόνα 149- Καταχώρηση Ιστοσελίδας

Τροποποίηση Ιστοσελίδας

Η διεπαφή για την λειτουργία αυτή αποτελείται από ένα textbox στο οποίο ο διαχειριστής θα μπορεί να ελέγχει το url της ιστοσελίδας και να το αλλάζει αν το επιθυμεί και ένα listbox στο οποίο θα μπορεί να επιλέγει τις κατηγορίες στις οποίες μπορεί να ανήκει. Στην συνέχεια τα στοιχεία αυτά θα ενημερώνονται στον server κάνοντας κλικ στο OK.

Έγκριση Χρήστη

Ο διαχειριστής επιλέγει (σε checkbox) από μία λίστα αιτουμένων προς εγγραφή χρηστών το ποιοι θα εγκριθούν για να χρησιμοποιούν τις υπηρεσίες του συστήματος. Κάνοντας κλικ στο OK οι εγκρίσεις καταχωρούνται στην βάση δεδομένων.

<input type="checkbox"/>	User 1
<input type="checkbox"/>	User 2
<input type="checkbox"/>	User 3
<input type="checkbox"/>	User 4

Εικόνα 152- Έγκριση Χρήστη

Καταχώρηση Προτιμήσεων Χρήστη

Από την εφαρμογή λαμβάνονται σε μορφή JSON οι προτιμήσεις του χρήστη σε μορφή JSON και καταγράφονται στην βάση δεδομένων.

Είσοδος Χρήστη

Από την εφαρμογή λαμβάνονται σε μορφή JSON τα στοιχεία του χρήστη. Στην συνέχεια ελέγχεται αν ο χρήστης είναι έγκυρος και σε θετική περίπτωση ανακτώνται οι ιστοσελίδες που ανήκουν στις κατηγορίες των προτιμήσεων του. Αυτές αποστέλλονται στην εφαρμογή σε μορφή JSON.

4.6 Υλοποίηση webserver

Η υλοποίηση σε κάθε επίπεδο της αρχιτεκτονικής του webserver έγινε με την χρήση των παρακάτω τεχνολογιών:

1. Επίπεδο Δεδομένων: Σχεσιακό Σύστημα Διαχείρισης Βάσεων Δεδομένων MySQL
2. Επίπεδο Λειτουργικότητας: Γλώσσα προγραμματισμού PHP
3. Επίπεδο Παρουσίασης: Περιγραφική γλώσσα HTML.

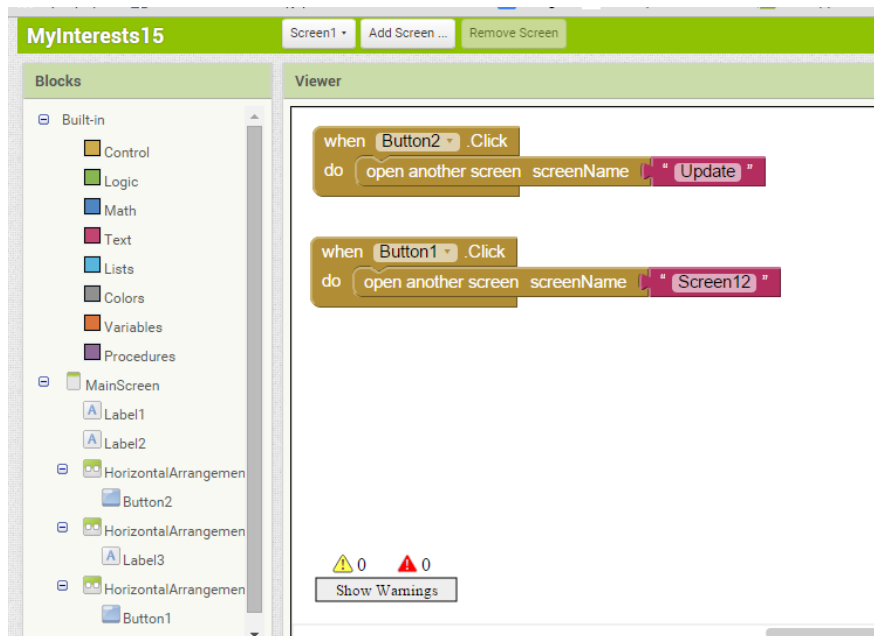
4.7 Υλοποίηση Εφαρμογής με App Inventor

Η υλοποίηση της εφαρμογής με το app inventor, απαρτίζεται από 2 μέρη όπως περιγράφηκε και στο τρίτο κεφάλαιο, από το σχεδιαστικό/ γραφιστικό κομμάτι που υλοποιείται στο designer και από το κομμάτι του κώδικα που υλοποιείται στο blocks editor.

Η πρώτη οθόνη που εμφανίζεται με το άνοιγμα της εφαρμογής είναι αυτή που ζητάει από τον χρήστη, είτε να κάνει εγγραφή (Sign up), όντας νέος χρήστης της εφαρμογής, είτε να πραγματοποιήσει είσοδο (Sign in), όντας ήδη χρήστης αυτής.



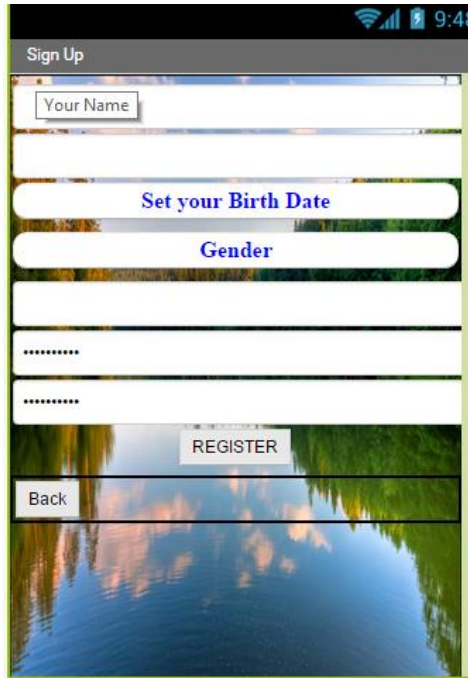
Εικόνα 153- Αρχική Οθόνη- Designer



Εικόνα 154- Αρχική Οθόνη- Blocks Editor

Εάν ο χρήστης είναι νέος και επιθυμεί να κάνει εγγραφή στην εφαρμογή, επιλέγει το Sign up και μεταβαίνει στην οθόνη όπου ζητείται να εισαχθούν το όνομα, το e-mail, η ημερομηνία

γέννησης και το φύλο του χρήστη. Ακόμη, πρέπει να οριστεί ένα username, ένα password και η ορθή επανάληψη του password αυτού.



Sign Up

Your Name

Set your Birth Date

Gender

.....

.....

REGISTER

Back

Εικόνα 155- Οθόνη sign up- Designer

Αφού γίνει η δήλωση των προσωπικών στοιχείων και προκειμένου να ολοκληρωθεί η εγγραφή, θα πρέπει ο χρήστης να επιλέξει ανάμεσα στις κατηγορίες που του εμφανίζονται.



Εικόνα 157- Οθόνη register- Designer


```

when Web1 .GetText
  url responseCode responseType responseContent
do
  if [get responseCode = 200]
  then
    set TextBox2 .Text to [get responseContent]
    set global checklist0 to [split text [get responseContent] at '<>']
    if [length of list list [get global checklist0] ≠ 0]
    then
      set global checklist to [split text [select list item list [get global checklist0] index 2] at ',']
      if [select list item list [get global checklist] index 1]
      then
        set C1 .Checked to true
      if [select list item list [get global checklist] index 2]
      then
        set C2 .Checked to true
      if [select list item list [get global checklist] index 3]
      then
        set C3 .Checked to true
      if [select list item list [get global checklist] index 4]
      then
        set C4 .Checked to true
      if [select list item list [get global checklist] index 5]
    
```

```

initialize global checklist0 to [create empty list]
initialize global profielist to [create empty list]
initialize global checklist to [create empty list]
initialize global parameter to ''

st0

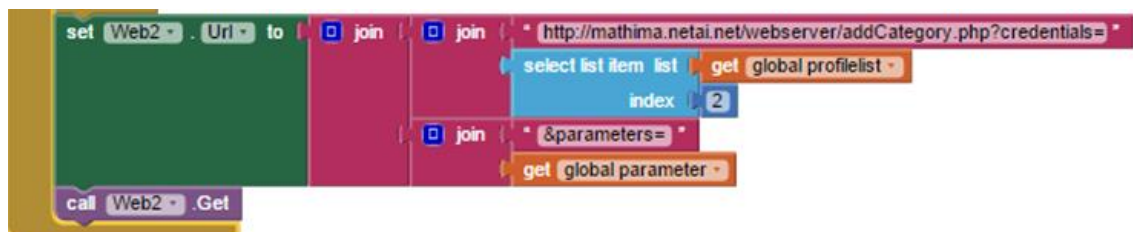
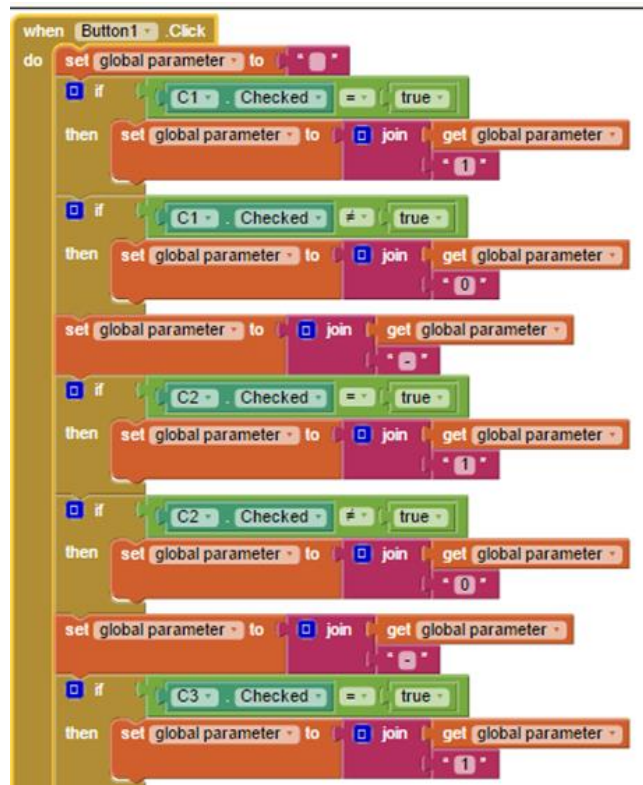
when Button3 .Click
do
  open another screen screenName 'Browser'

when File1 .GetText
  text
do
  set global profielist to [split text [get text] at '<']
  set Web1 .Uri to [join ['http://mathima.netai.net/webserver/getusercategories.php?u=' [select list item list [get global profielist] index 2]]]
  call Web1 .Get

when register .Initialize
do
  call File1 .ReadFrom
  fileName 'credentials.txt'

when Web2 .GetText
  url responseCode responseType responseContent
do
  if [contains text [get responseContent] piece 'profileupdatedok']
  then
    call Notifier1 .ShowAlert
    notice 'Profile Updated!'
  if [contains text [get responseContent] piece 'profileupdatedok']
  then
    call Notifier1 .ShowAlert
    notice 'Profile Updated!'

```

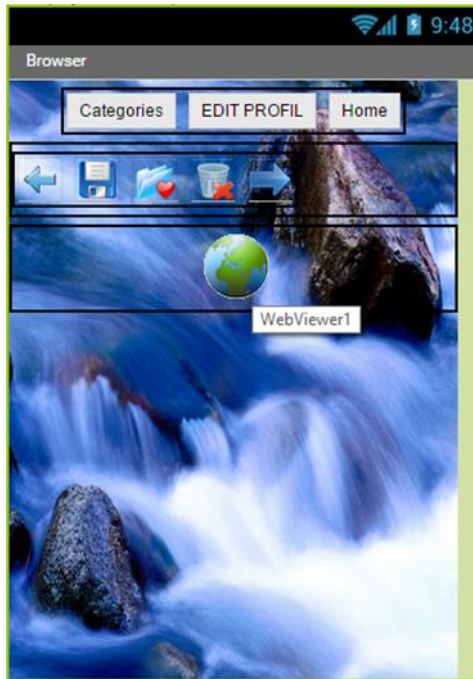



Εικόνα 158- Οθόνη register- Blocks Editor






Σημείωση: Στα blocks «when web1. Got Text» και «when button1. Click», οι υποθέσεις (if) συνεχίζονται κατά τον ίδιο τρόπο για όλες (21) τις κατηγορίες της εφαρμογής.

Αν ο χρήστης επιλέξει το κουμπί «back» θα μεταβεί πάλι στην αρχική οθόνη της εφαρμογής.

Αφού ο χρήστης έχει επιλέξει τις κατηγορίες, πατώντας το κουμπί «register» μεταβαίνει στην οθόνη περιήγησης (browser), όπου προβάλλεται περιεχόμενο από τις κατηγορίες της οποίες επέλεξε στο προηγούμενο βήμα.



Εικόνα 159- Οθόνη περιήγησης- Designer

Στην οθόνη αυτή του παρέχεται η δυνατότητα αποθήκευσης κάποιου από τα περιεχόμενα που θέλει να κρατήσει ως αρχείο για να επισκεφτεί αργότερα, επιλέγοντας το σύμβολο της δισκέτας . Μπορεί να ανοίξει τις σελίδες τις οποίες αποθήκευσε με την επιλογή του συμβόλου του φακέλου , μπορεί να διαγράψει κάποια από τις αποθηκευμένες σελίδες με την επιλογή του κάδου , και τέλος με τα βελάκια περιήγησης μπορεί να μεταβεί σε προηγούμενες (αριστερό βέλος ) ή επόμενες ιστοσελίδες (δεξί βέλος ) που του παρέχει η εφαρμογή.

Επιπλέον, στην οθόνη περιήγησης, υπάρχουν τρεις ακόμη επιλογές: «Categories», «EDIT PROFIL» και «Home». Επιλέγοντας το «Categories», ο χρήστης μεταβαίνει στην σελίδα επιλογής των κατηγοριών, επιλέγοντας το «EDIT PROFIL» μεταβαίνει στην σελίδα όπου έκανε την εγγραφή, οπότε και μπορεί να αλλάξει τα στοιχεία του προφίλ του, και επιλέγοντας το «Home» βρίσκεται πάλι στην αρχική σελίδα της εφαρμογής.

```

when Button1 .Click
do
  if [get global name2 = 2] or [get global name2 = 1]
  then
    set global name2 to [length of list list]
    call WebView1 .GoToUrl
    url [select list item list] [get global urlist]
    index [get global name2]
    set global elsepointer to 1
  if [get global elsepointer = 0] and not [get global name2 = 2] or [get global name2 = 1]
  then
    set global name2 to [get global name2] - 1
    call WebView1 .GoToUrl
    url [select list item list] [get global urlist]
    index [get global name2]
    set global elsepointer to 0
  set global elsepointer to 0

when Web1 .GotText
url [responseCode] [responseType] [responseContent]
do
  if [get responseCode = 200]
  then
    set global name to [split text] [get responseContent]
    at [ <h> ]
    set global urlist to [split text] [select list item list] [get global name]
    at [ <br> ]
    call WebView1 .GoToUrl
    url [select list item list] [get global urlist]
    index [2]
    set global name2 to 2
  
```

0 0
Show Warnings

```

when Button6 .Click
do
  call File2 .ReadFrom
  fileName [myfavourites.txt]
  set Button5 .Enabled to true

when Browser .Initialize
do
  call File1 .ReadFrom
  fileName [credentials.txt]

when Button9 .Click
do
  call File1 .ReadFrom
  fileName [credentials.txt]
  set Button5 .Enabled to false

when Button8 .Click
do
  open another screen screenName [Register]

when Button3 .Click
do
  open another screen screenName [Screen1]

when Button7 .Click
do
  open another screen screenName [register]

initialize global fileurlist to [create empty list]
initialize global name2 to 1
initialize global name to [create empty list]
initialize global urlist to [create empty list]
initialize global file_contents to [ ]
initialize global myurl to [http://mathima.net.in/webserver/login1.php?credentials=]
initialize global elsepointer to 0
initialize global profilelist to [create empty list]
initialize global credentials to [ ]

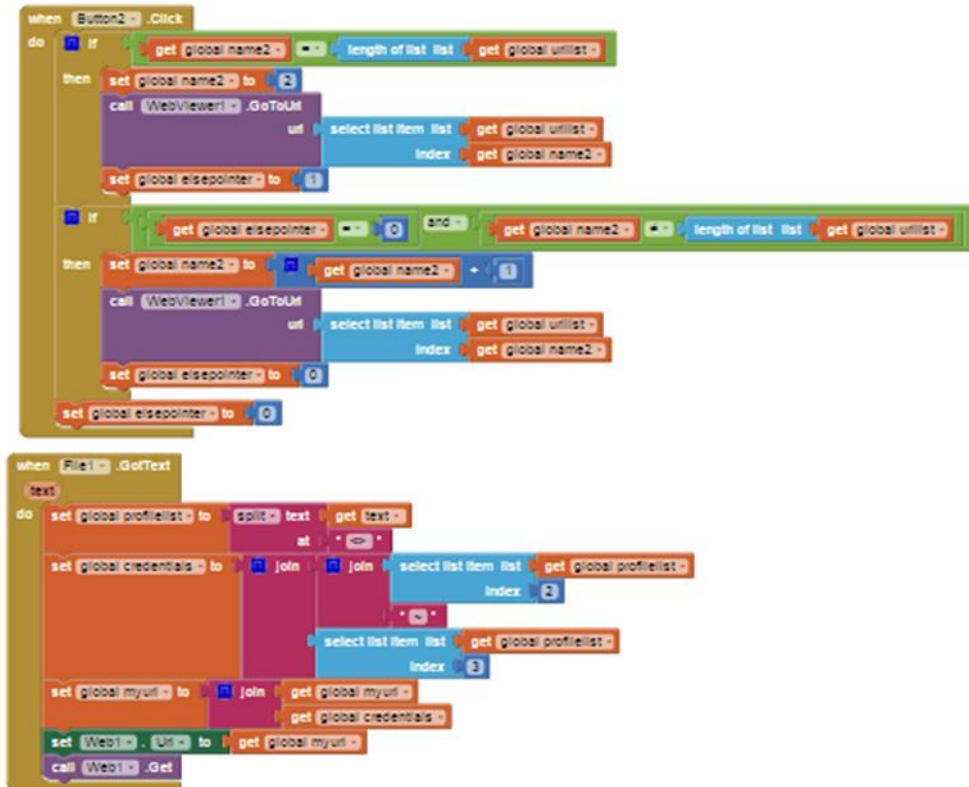
when Button4 .Click
do
  call File2 .AppendToFile
  text [join] [WebView1 .CurrentUri]
  fileName [myfavourites.txt]
  call Notifier1 .ShowAlert
  notice [Content added to favourites]
  
```

```

when File2 . GotText
do
  set global urlist to split text get text
  at . @ .
  set global fileurlist to split text get text
  at . @ .
  if length of list list get global urlist < 0
  then
    call WebView1 . GoToURL
    ut select list item list get global urlist
    index
    set global name2 to
  if length of list list get global urlist < 0
  then
    call Notification . ShowAlert
    notice Favorite list is empty
    call WebView1 . GoToURL
    ut http://mathima.netal.net/webserver/empty.htm

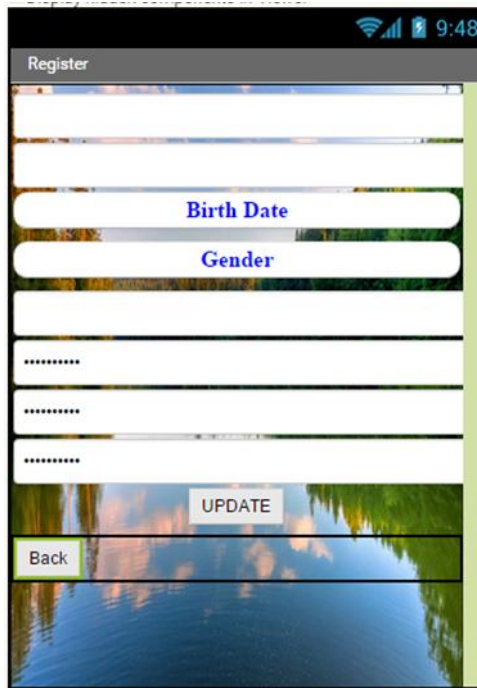
when Button5 . Click
do
  call File2 . Delete
  fileName myfavouites.txt
  call File2 . SaveFile
  text
  fileName myfavouites.txt
  for each number from
  to length of list list get global fileurlist
  by
  do
    if select list item list get global fileurlist . @ . @ . and select list item list get global fileurlist . @ . WebView1 . CurrentURL
    then
      set global file_contents to join get global file_contents
      join
      select list item list get global fileurlist
      index
      get number
  call File2 . AppendToFile
  text get global file_contents
  fileName myfavouites.txt
  call Notification . ShowAlert
  notice Web page has been deleted from favorites
  call File2 . ReadFrom
  fileName myfavouites.txt

```

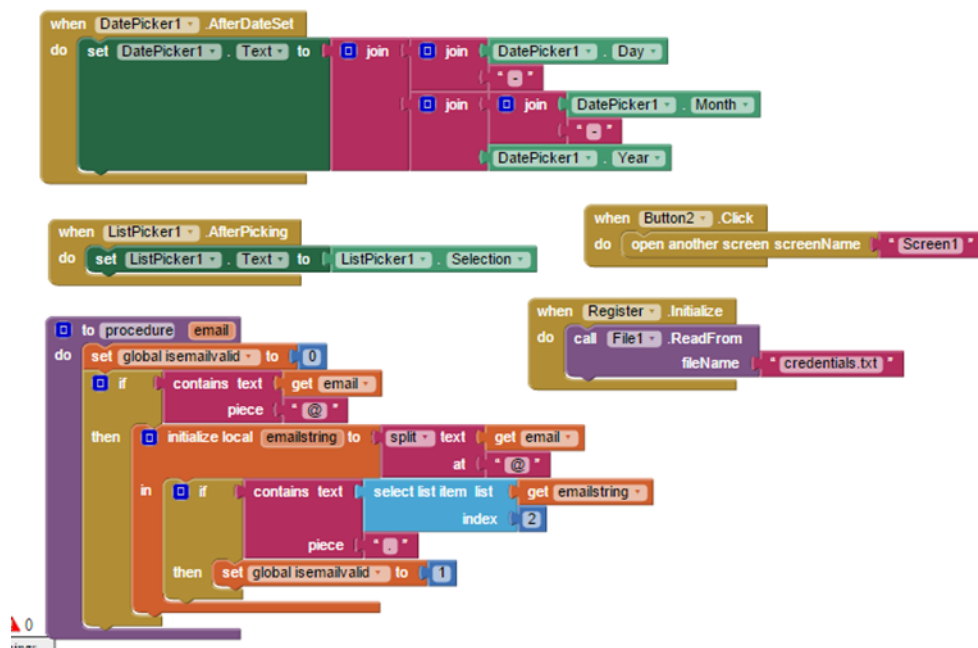


Εικόνα 160- Οθόνη περιήγησης- Blocks Editor

Στην περίπτωση, λοιπόν, όπου ο χρήστης θέλει να κάνει αλλαγές στις πληροφορίες στο προφίλ του, επιλέγοντας το πλήκτρο «EDIT PROFIL» μεταβαίνει στην οθόνη «update» όπου μπορεί να αλλάξει τις προσωπικές του πληροφορίες.



Εικόνα 161- Οθόνη update- Designer




```

initialize global datelist to create empty list
initialize global profilist to create empty list
initialize global url1 to http://mathima.netal.net/webserver/register.php
initialize global f to ?f= initialize global b to 
initialize global u to &p= initialize global s to 
initialize global isemalvalid to 0
initialize global np to 
initialize global url to 
initialize global e to 
initialize global p to ?f=

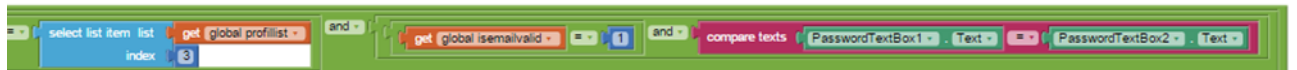
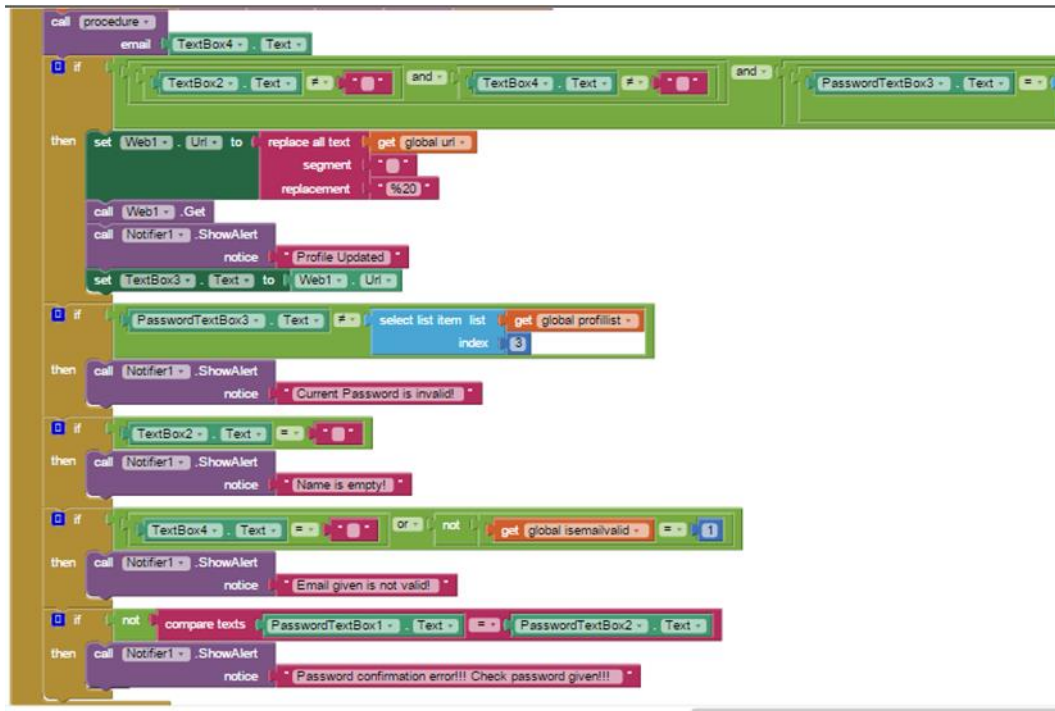
when Web1 GotText
  url responseCode responseType responseContent
  do
    if get responseCode != 200
    then call Notifier1 ShowAlert
         notice Connection Error!!
    if get responseCode == 200
    then if contains text get responseContent
         piece Successfull Registration!
    then call Notifier1 ShowAlert
         notice Successfull Update!!
         call File1 SaveFile
              text join join TextBox1 . Text
                  ~
                  PasswordTextBox1 . Text
              fileName credentias.txt
              open another screen screenName register
  end
end

```

```

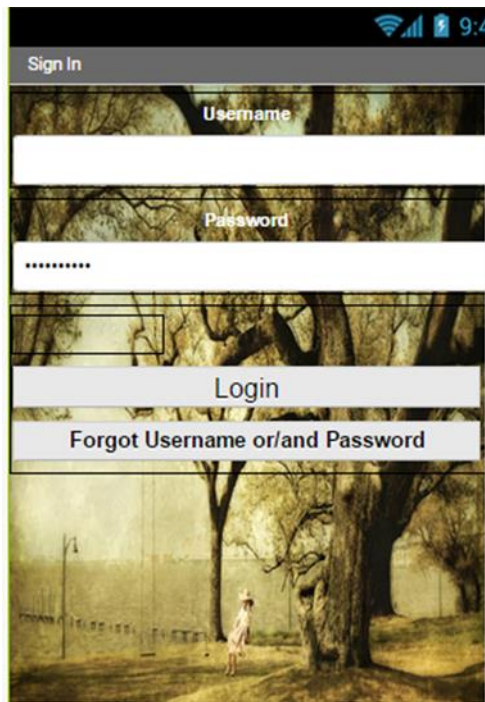
when Button1 .Click
do
  set global f to join ?f=
                  TextBox2 . Text
  set global p to join &p=
                  PasswordTextBox3 . Text
  set global np to join &np=
                  PasswordTextBox1 . Text
  set global u to join &u=
                  TextBox1 . Text
  set global e to join &e=
                  TextBox4 . Text
  set global s to join &s=
                  ListPicker1 . Selection
  set global b to join &b=
                  join join DatePicker1 . Day
                      ~
                      join DatePicker1 . Month
                      ~
                      join DatePicker1 . Year
                      ~
  set global url to join get global url1
                      join join get global f
                          join get global e
                          join get global np
                          join get global s
                          join get global p
                          join get global u
                          join get global b

```



Εικόνα 162- Οθόνη update- Blocks Editor

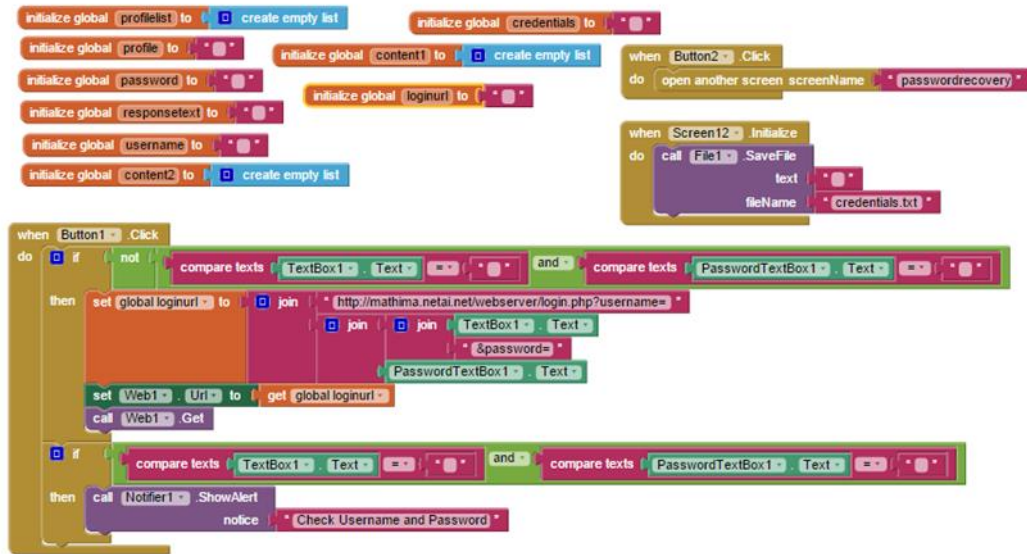
Σε περίπτωση που ο χρήστης είναι ήδη εγγεγραμμένος στην εφαρμογή, εισερχόμενος σε αυτή επιλέγει το «log in», έτσι ώστε να μεταβεί στην οθόνη όπου ζητείται η εισαγωγή του username και του password που έχει δηλώσει κατά την εγγραφή του.



Εικόνα 163- Οθόνη Sign in- Designer

```

when Web1 - GotText
  url responseCode responseType responseContent
do
  if get responseCode = 200
  then
    if contains text get responseContent
      piece <<>usernfound<<>
    then
      call Notifier1 - showAlert
      notice Login Error!
      open another screen screenName Screen1
    if not contains text get responseContent
      piece <<>usernfound<<>
    then
      call Notifier1 - showAlert
      notice Successfull Login!
      set global profilelist to split text get responseContent
      at <<>
      set global profile to select list item list get global profilelist
      index 2
      call File1 - SaveFile
      text get global profile
      fileName credentials.txt
      open another screen screenName Browser
  
```

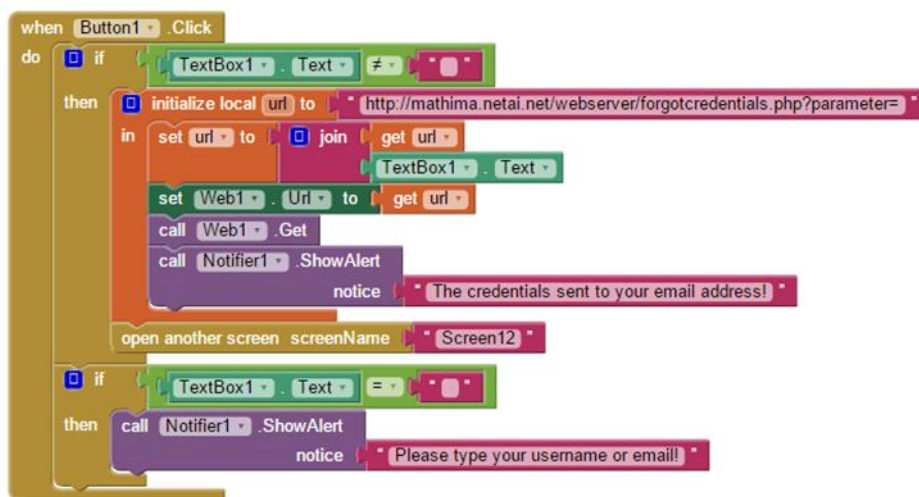


Εικόνα 164- Οθόνη Sign in- Blocks Editor

Σε περίπτωση που ο χρήστης έχει ξεχάσει είτε το username του, είτε το password του, υπάρχει η επιλογή «forgot username or/and password», όπου επιλέγοντάς τη ο χρήστης μεταβαίνει σε μια οθόνη όπου του ζητείται να πληκτρολογήσει είτε το username του, είτε το e-mail του για να μπορέσει να γίνει ανάκτηση των πληροφοριών που ζητούνται.



Εικόνα 165- Οθόνη Recovery- Designer



Εικόνα 166- Οθόνη recovery- Blocks Editor

Κεφάλαιο 5

5.1 Αξιολόγηση Εφαρμογής

Ο όρος αξιολόγηση με την ευρεία έννοια, χρησιμοποιείται όταν θέλουμε να κάνουμε μια εκτίμηση των γνωρισμάτων και των ιδιοτήτων ενός «πράγματος». Η αξιολόγηση ενός αντικειμένου, μιας δραστηριότητας, μιας κατάστασης μπορεί να αποδοθεί με γραφικό, αριθμητικό ή γλωσσικό τρόπο.

Η εκπαιδευτική αξιολόγηση είναι η διαδικασία τεκμηρίωσης, συνήθως με μετρήσιμους όρους, των γνώσεων, δεξιοτήτων, στάσεων και πεποιθήσεων. Η αξιολόγηση μπορεί να επικεντρωθεί σε έναν εκπαιδευόμενο, μια μαθησιακή κοινότητα, (τάξη, εργαστήριο ή οργανωμένες ομάδες εκπαιδευόμενων), σε ένα εκπαιδευτικό ίδρυμα ή στο εκπαιδευτικό σύστημα στο σύνολό του.

Οι πρακτικές αξιολόγησης στην εκπαίδευση εξαρτώνται από το θεωρητικό πλαίσιο που ορίζουν οι ερευνητές, τις υποθέσεις και τις πεποιθήσεις τους για το ερευνώμενο δείγμα, την προέλευση της γνώσης και τις μαθησιακές διαδικασίες.

Η αξιολόγηση διακρίνεται σε τρεις κατηγορίες, ανάλογα με τα στάδια στα οποία εφαρμόζεται:

1. Προκαταρκτική ή προγνωστική αξιολόγηση

Συντελείται στην αρχή διδασκαλίας ενός μαθήματος ή έργου και προσπαθεί να ανιχνεύσει προϋπάρχουσες γνώσεις και στάσεις γύρω από το αντικείμενο που θα διδαχθεί.

2. Διαμορφωτική αξιολόγηση

Γίνεται κατά τη διάρκεια της διδασκαλίας και σαν στόχο της έχει να υποβοηθήσει τη μάθηση. Διαμορφωτική αξιολόγηση έχουμε όταν ένας μαθητής ή καθηγητής προσφέρει ανατροφοδότηση για το έργο κάποιου άλλου μαθητή. Οι διαμορφωτικές αξιολογήσεις έχουν διαγνωστικό ρόλο, εντοπίζοντας τις μαθησιακές δυσκολίες με σκοπό να τις εξυγιάνουν.

3. Αθροιστική ή τελική αξιολόγηση

Λαμβάνει χώρα στο τέλος ενός μαθήματος ή έργου. Σε ένα εκπαιδευτικό περιβάλλον, οι αθροιστικές αξιολογήσεις χρησιμοποιούνται κυρίως για να βαθμολογήσουν την επίδοση των μαθητών στους μαθητές και να προσφέρουν αποτίμηση όλης της διαδικασίας.

Για την αξιολόγηση τόσο της εργασίας, όσο και του ίδιου του εργαλείου του app *inventor*, επιλέχθηκε η «αθροιστική/ τελική αξιολόγηση». Τα στοιχεία συγκεντρώθηκαν μέσα από τη συμπλήρωση ενός ανώνυμου ερωτηματολογίου, αποτελούμενο από 15 ερωτήσεις οι οποίες απαντήθηκαν από ένα τυχαίο σύνολο 80 ατόμων.

Το ερωτηματολόγιο αποτελεί «το μέσον επικοινωνίας (Interface) μεταξύ του ερευνητή και των ερωτώμενων, με άμεσο ή έμμεσο τρόπο, ανάλογα με τη μέθοδο συλλογής των δεδομένων» (Ρόντος & Παπάνης, 2007). Είναι ένα έντυπο που απαρτίζεται από μια ακολουθία ερωτήσεων, που παρουσιάζονται με μια συγκεκριμένη σειρά, που εξυπηρετεί τους σκοπούς του ερευνητή (Ζαφειρίου, 2003).

Τα βασικότερα πλεονεκτήματα των ερωτηματολογίων είναι:

- Το χαμηλό κόστος υλοποίησης,
- Η αποστολή σε μεγάλο πλήθος ανθρώπων,
- Η ευκολία στην κατασκευή και χρήση του,

- Η ελευθερία στην έκφραση των ερωτηθέντων,
- Η τυποποίηση των μεθόδων ανάλυσης του υλικού,
- Οι απαντήσεις των ερωτηθέντων δεν επηρεάζονται από τον ερευνητή,
- Η μικρότερη απαίτηση χρόνου σε σχέση με άλλα μέσα συλλογής δεδομένων.

Το ερωτηματολόγιο υποβλήθηκε στους ερωτώμενους διαδικτυακά. Όπως αναφέρεται σε άρθρο των Ζωγόπουλου και Μπαγουλή, η διαδικτυακή αξιολόγηση έχει πολλά πλεονεκτήματα, όπως:

- Μεγαλύτερη ασφάλεια καθώς τα μέσα συλλογής αποστέλλονται μέσω ηλεκτρονικής μετάδοσης και κρυπτογράφησης.
- Σημαντική μείωση του κόστους και των χρησιμοποιούμενων πόρων.
- Ευχρηστία στη διαχείριση και αξιολόγηση των ευρημάτων.
- Εξαγωγή αποτελεσμάτων, αναφορών, στατιστικών.

Ακόμη, η διαδικτυακή αξιολόγηση έχει πιο φιλική μορφή και προδιαθέτει θετικά τους ερωτώμενους, μειώνοντας το άγχος που εμφανίζεται όταν αντιμετωπίζουν τον παραδοσιακό τρόπο αξιολόγησης.

Το αρνητικό στοιχείο στην μέσω διαδικτύου αξιολόγηση αποτελεί το γεγονός ότι δεν εγγυάται η ανατροφοδότηση από όλους τους καλούμενους προς αξιολόγηση. Στην προκειμένη περίπτωση στάλθηκαν 100 ερωτηματολόγια εκ των οποίων απαντήθηκαν τα 80.

5.1.1 Ευρήματα ερωτηματολογίου

Με το πέρας της συμπλήρωσης του ερωτηματολογίου και έπειτα από ανάλυση και ερμηνεία των απαντήσεων που πάρθηκαν, εξήχθησαν συμπεράσματα για την πληρότητα και χρησιμότητα του εγχειριδίου, καθώς και για την ευκολία εκμάθησης και χρήσης του app inventor ως εργαλείο δημιουργίας εφαρμογών για κινητά τηλέφωνα. Οι ερωτήσεις που έλαβαν χώρα στο ερωτηματολόγιο κινούνταν στους παρακάτω άξονες:

- Πρότερες γνώσεις στον προγραμματισμό και τη χρήση υπολογιστικών συστημάτων.

- Ανίχνευση στάσεων.
- Δεξιότητες και γνώσεις που αποκτήθηκαν.
- Συμβολή του πρότυπου παραδείγματος στην εκμάθηση του app inventor.

Στην συνέχεια παρουσιάζονται οι ερωτήσεις που τέθηκαν στο ερωτηματολόγιο (παρατίθεται στο παράρτημα) ομαδοποιημένες στους άξονες που αναφέρθηκαν παραπάνω.

Πίνακας 15- Ερωτήσεις Ερωτηματολογίου

Άξονες	Ερωτήσεις
Πρότερες γνώσεις	5. Χρησιμοποιείται τον υπολογιστή για: 6. Έχετε ασχοληθεί με τον προγραμματισμό; 7. Έχετε ξανασχοληθεί με παρόμοια εργαλεία όπως το app inventor;
Στάσεις	8. Το app inventor ως εργαλείο ανάπτυξης μιας εφαρμογής, είναι: 13. Τι θα θέλατε να περιέχει σε μεγαλύτερο βαθμό το tutorial; 14. Το tutorial σας βοήθησε: 15. Θα συνεχίσετε την δημιουργία εφαρμογών με το app inventor;
Δεξιότητες/ γνώσεις	9. Μπορείτε να δημιουργήσετε την δική σας εφαρμογή μέσω του app inventor; 10. Μέχρι ποιο σημείο σας βοήθησε το εγχειρίδιο να κατανοήσετε το εργαλείο του app inventor; 12. Με την ολοκλήρωση του tutorial, είστε σε θέση να:
Πρότυπο παράδειγμα	11. Σας βοήθησε το παράδειγμα/ εφαρμογή να κατανοήσετε το εργαλείο app inventor;

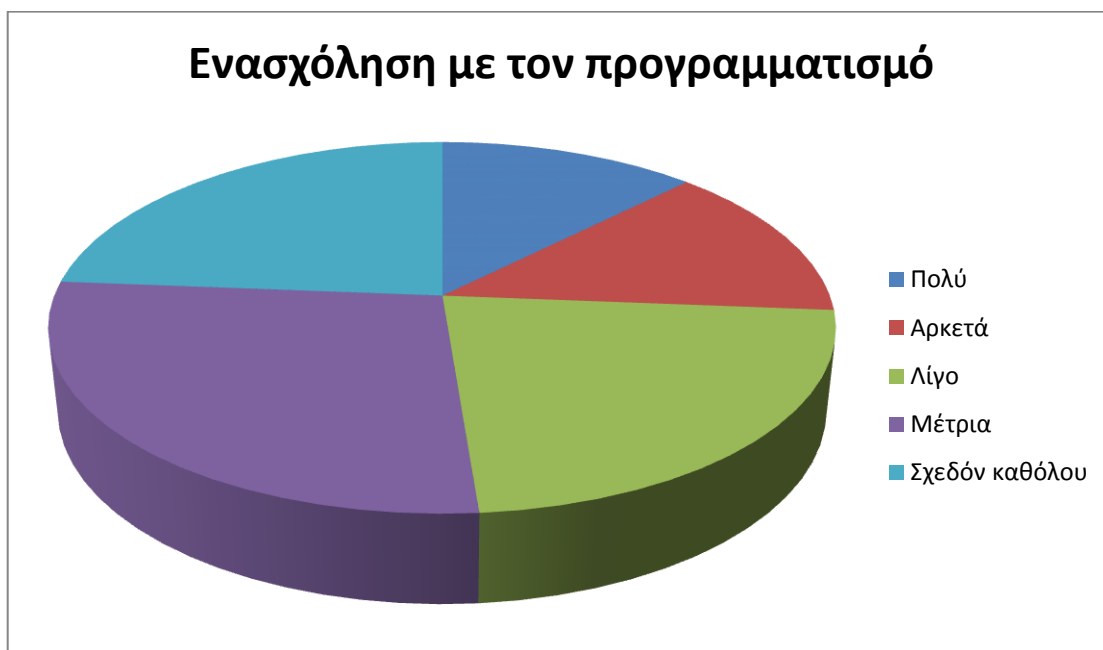
Πρέπει να σημειωθεί ότι το δείγμα των ερωτηθέντων κυμαίνεται σε όλες τις ηλικίες, και στα δυο φύλα και καλύπτει ένα αρκετά ευρύ μορφωτικό φάσμα. Ακόμη, οι ερωτηθέντες ως επί το πλείστον είναι τεχνολογικά αλφαριθμητικοί και χρησιμοποιούν τον υπολογιστή αρκετές ώρες μέσα στην ημέρα.

Παρακάτω αναλύονται τα ευρήματα του ερωτηματολογίου σύμφωνα με τους άξονες στους οποίους κινήθηκε το ερωτηματολόγιο.

- **Πρότερες Γνώσεις**

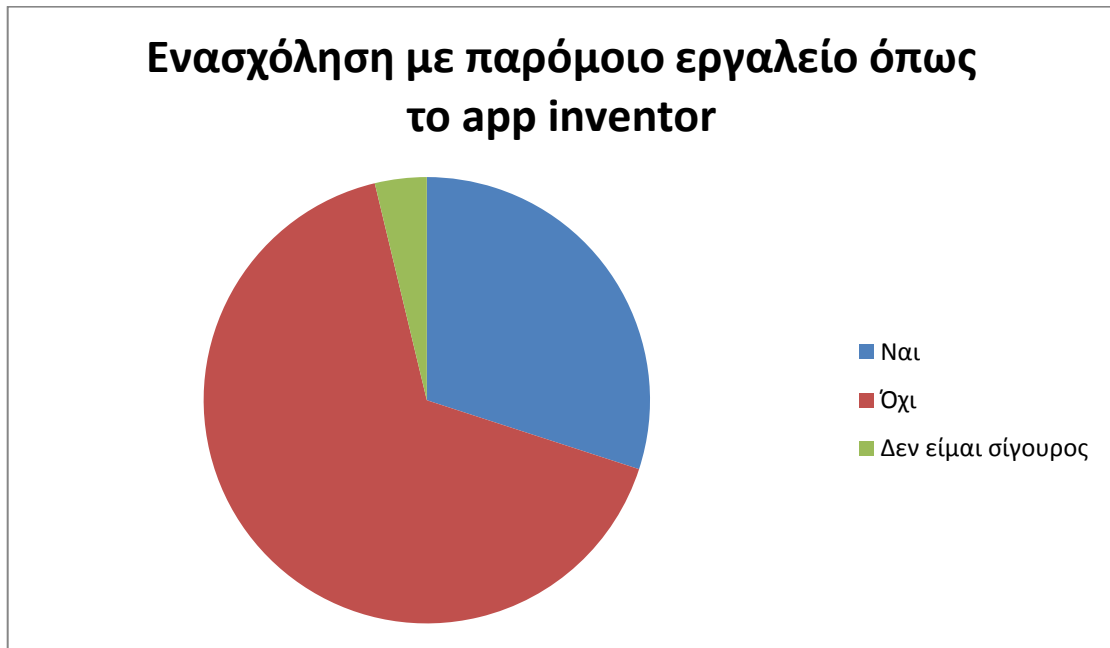
Οι ερωτηθέντες οι οποίοι συμμετείχαν στο δείγμα, χρησιμοποιούν τον ηλεκτρονικό υπολογιστή κυρίως για επαγγελματικούς λόγους σε ηλικίες από 30 και πάνω και για μαθησιακούς σκοπούς σε μικρότερες ηλικίες. Μεγάλο ποσοστό, επίσης, δήλωσε ότι εκμεταλλεύεται τα μέσα κοινωνικής δικτύωσης για λόγους επικοινωνίας.

Στην ερώτηση αν έχουν ξαναασχοληθεί με τον προγραμματισμό, το μεγαλύτερο ποσοστό 27,5% δήλωσε ότι έχει ασχοληθεί «μέτρια», ένα μεγάλο ποσοστό 23,75% δήλωσε ότι δεν έχει ασχοληθεί καθόλου, ενώ το 12,5% ότι έχει ασχοληθεί πολύ.



Σχήμα 10- Ερώτηση 6/ Ενασχόληση με προγραμματισμό

Επίσης, το μεγαλύτερο ποσοστό 66,25% δεν έχει ξανασχοληθεί με παρόμοιο εργαλείο όπως αυτό του app inventor, ενώ ένα 3,75% δήλωσε ότι «δεν είμαι σίγουρος», αυτό προκύπτει είτε λόγω του ότι δεν μπορούν να εντάξουν επακριβώς το app inventor σε μια κατηγορία εργαλείων, είτε γιατί δεν έχουν ξεκάθαρη εικόνα των εργαλείων που έχουν ενασχοληθεί στον παρελθόν.

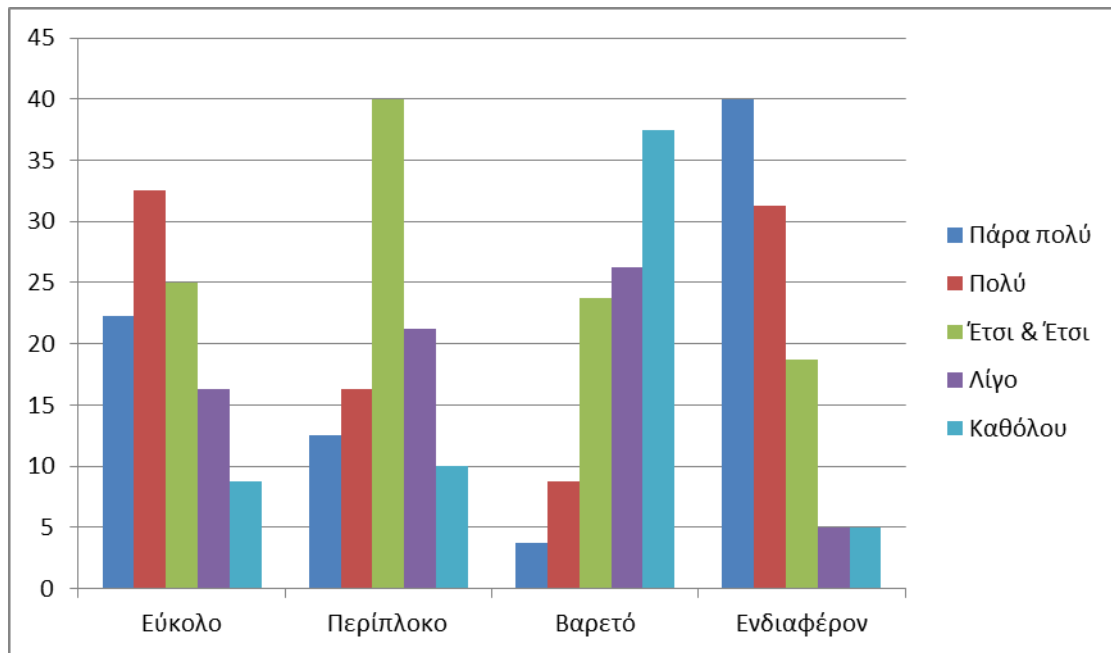


Σχήμα 11- Ερώτηση 7/ Ενασχόληση με παρόμοιο εργαλείο όπως το app inventor

- **Στάσεις**

Στην ερώτηση πώς βρίσκουν το εργαλείο του app inventor, βάση των μεγαλύτερων ποσοστών ανά επιλογή, προκύπτει ότι θεωρούν το app inventor «πολύ» εύκολο κατά 32,5%, περίπλοκο «έτσι και έτσι» κατά 40%, «καθόλου» βαρετό κατά 37,5% και «πάρα πολύ» ενδιαφέρον κατά 40%.

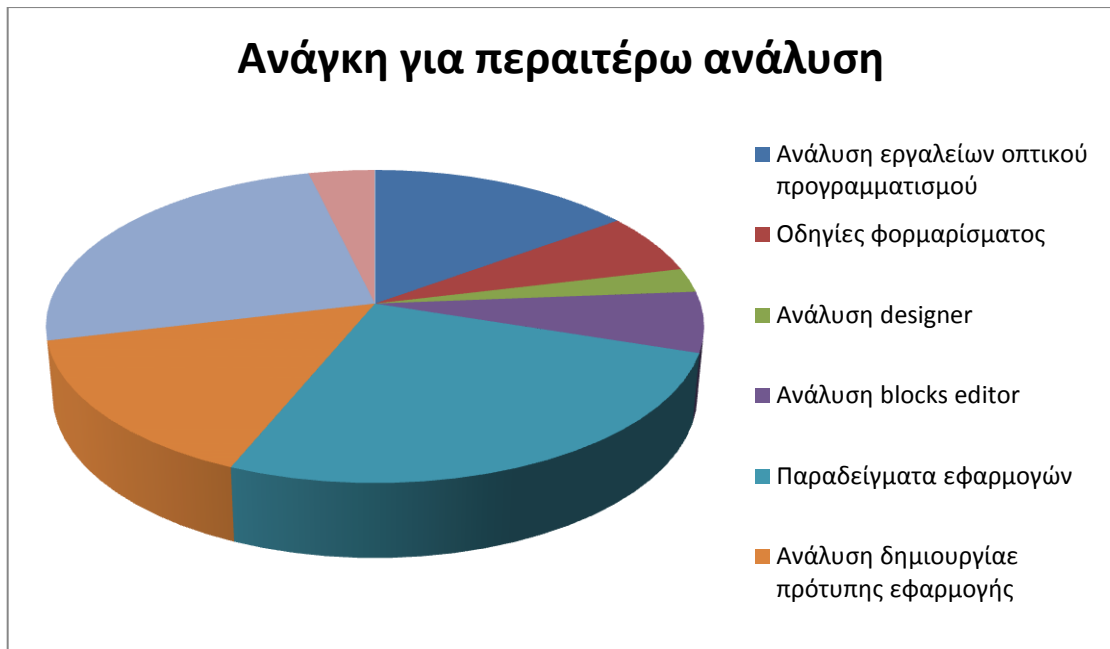
Βάση των παραπάνω, βγάζουμε το συμπέρασμα ότι το app inventor θεωρείτε ως ένα σχετικά εύκολο εργαλείο αλλά όχι βαρετό, όχι ιδιαίτερα περίπλοκο, με μεγάλο ενδιαφέρον για το χρήστη.



Σχήμα 12- Ερώτηση 8/ Πώς βρίσκετε το εργαλείο app inventor

Προκειμένου να αντληθούν συμπεράσματα για την πληρότητα του tutorial, οι ερωτώμενοι κλήθηκαν να επιλέξουν στο τι θα ήθελαν να περιείχε σε μεγαλύτερο βαθμό το εγχειρίδιο ούτως ώστε να κατανοήσουν και να αναπτύξουν δεξιότητες πάνω στο app inventor. Κρίνοντας από τις απαντήσεις αυτό που λείπει περισσότερο από το εγχειρίδιο είναι περισσότερα παραδείγματα εφαρμογών δημιουργημένα με το συγκεκριμένο εργαλείο, σε ποσοστό 26,25%. Επίσης, ένα ποσοστό της τάξεως του 15% θα ήθελε μεγαλύτερη ανάλυση των εργαλείων οπτικού προγραμματισμού, καθώς επίσης και μεγαλύτερη επεξήγηση της πρότυπης εργασίας που αναλύθηκε στο tutorial.

Μικρότερα ποσοστά της τάξεως του 6,25% κατέλαβαν οι ανάγκες για μεγαλύτερη ανάλυση του blocks designer και των οδηγιών φορμαρίσματος. Σημαντικό είναι, τέλος, το ποσοστό 25% που βρήκε το εγχειρίδιο πλήρες σε όλες τις πτυχές του.

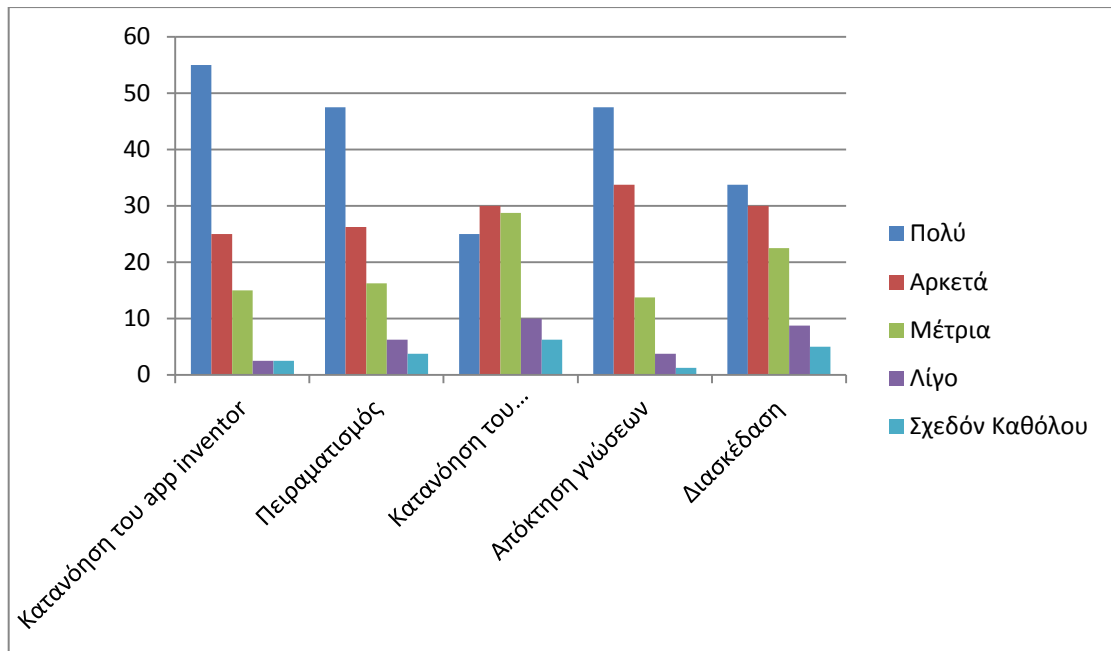


Σχήμα 13- Ερώτηση 13/ Ανάγκη για περαιτέρω ανάλυση του εργαλείου

Συνεχίζοντας την ανάλυση των απαντήσεων του ερωτηματολογίου, προκύπτει ότι το tutorial βοήθησε κατά πολύ στην κατανόηση του app inventor (ποσοστό 55%), καθώς και στον πειραματισμό των εμπλεκόμενων με αυτό (ποσοστό 47,5%).

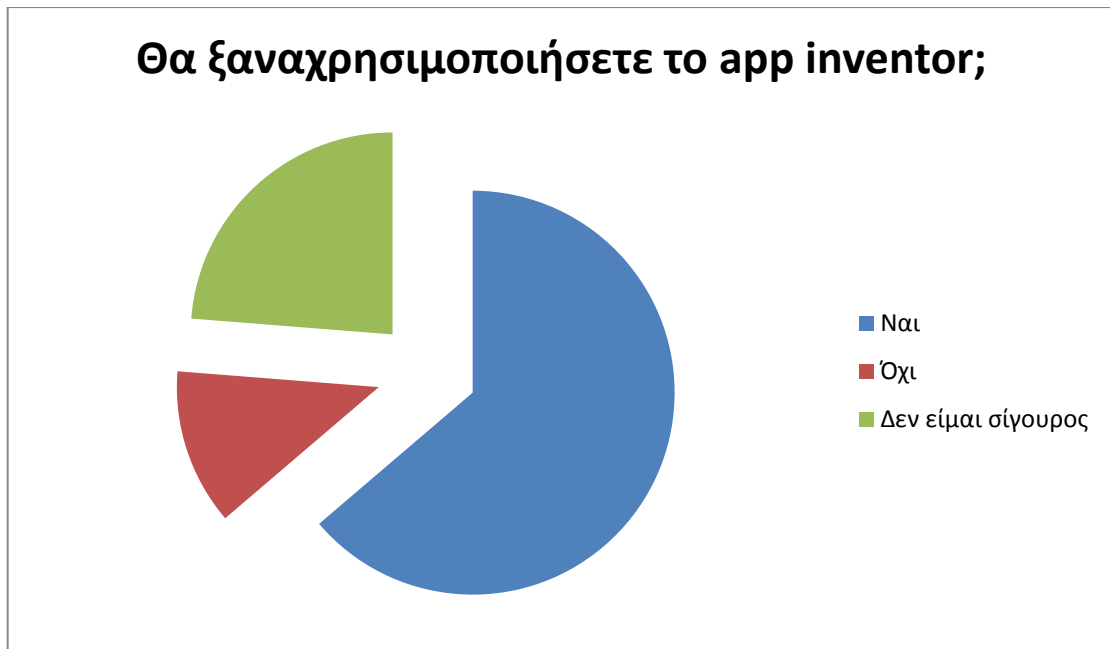
Οι απόψεις σχετικά με το αν το εγχειρίδιο βοηθάει στην γενικότερη κατανόηση του προγραμματισμού πέρα από τα πλαίσια του συγκεκριμένου εργαλείου, μοιράζονται μεταξύ «πολύ», «αρκετά» και «μέτρια» με ποσοστά 25%, 30% και 28,75% αντίστοιχα, πράγμα που είναι λογικό αν λάβουμε υπόψη την πρότερη γνώση που κατείχαν οι ερωτηθέντες σχετικά με τον προγραμματισμό.

Ακόμη, τα μεγαλύτερα ποσοστά, δήλωσαν ότι με το πέρας του εγχειριδίου, διασκέδασαν αλλά κυρίως απέκτησαν γνώσεις σε αρκετά μεγάλο βαθμό.



Σχήμα 14- Ερώτηση 14/ Ανάπτυξη της γενικότερης κατανόησης του προγραμματισμού

Απαντώντας στην ερώτηση αν θα ξανασχοληθούν με το app inventor μελλοντικά, λαμβάνοντας το ποσοστό 63,75%, βγάζουμε το συμπέρασμα ότι κατανόησαν και έμειναν ικανοποιημένοι τόσο με τις λειτουργίες και τις δυνατότητες του εργαλείου, όσο και με το ίδιο το εγχειρίδιο.



Σχήμα 15- Ερώτηση 15/ Θα ξαναχρησιμοποιήσετε το app inventor?

- **Δεξιότητες/ Γνώσεις**

Οι ερωτηθέντες δήλωσαν ότι μετά την ολοκλήρωση του tutorial έχουν τις γνώσεις και τις δεξιότητες «σε μεγάλο βαθμό» να φτιάξουν μια δική τους εφαρμογή κατά ένα ποσοστό 35% και «ως ένα σημείο» σε ποσοστό 32,5%. Το ποσοστό που είπε με σιγουριά ότι μπορεί να φτιάξει μόνο του μια εφαρμογή εξ ολοκλήρου φτάνει το 18,75%, ενώ τελείως αρνητικοί ήταν το ποσοστό 13,75%.



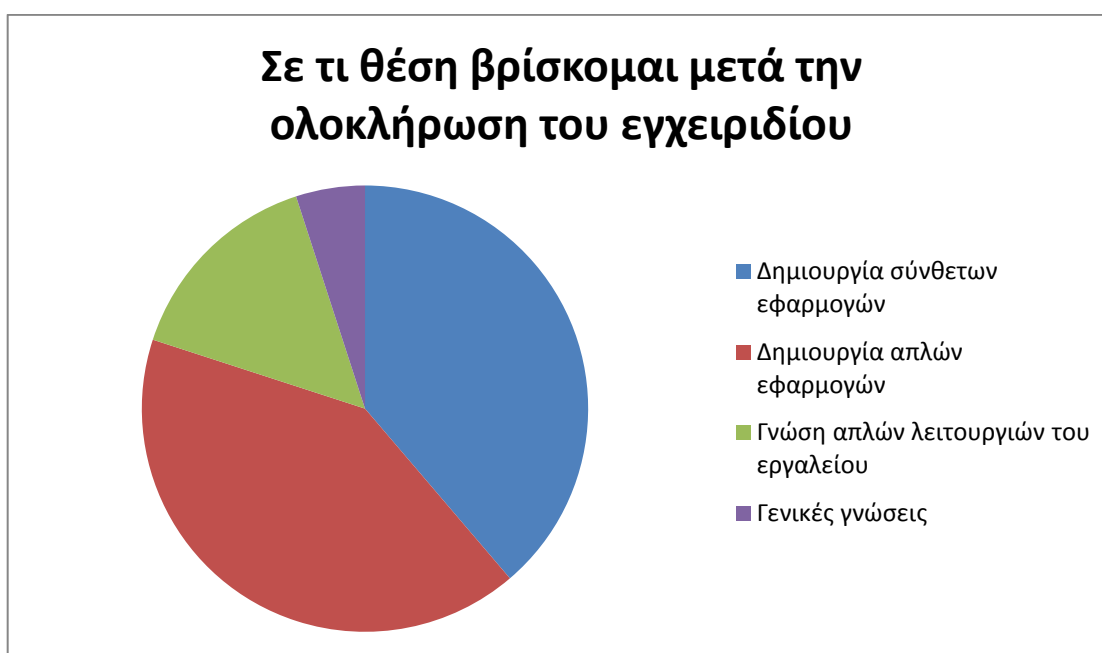
Σχήμα 16- Ερώτηση 9/ Δημιουργία Εφαρμογής με app inventor

Στη συνέχεια, απαντώντας στην ερώτηση μέχρι ποιο σημείο σας βοήθησε το εγχειρίδιο στην κατανόηση του app inventor, το 51,25% των ερωτηθέντων είπε ότι βοηθήθηκε «πολύ», το 23,75% «σε μεγάλο βαθμό», το 20% «ως ένα σημείο» και μόνο ένα 5%, δήλωσε ότι δεν βοηθήθηκε καθόλου.



Σχήμα 17- Ερώτηση 10/ Πόσο βοήθησε το εγχειρίδιο στην κατανόηση του app inventor

Απαντώντας στην ερώτηση «ολοκληρώνοντας το tutorial τι είστε σε θέση να κάνετε», οι περισσότεροι (41,25%) θεωρούν ότι μπορούν να φτιάξουν μια απλή εφαρμογή μόνοι τους με το εργαλείο app inventor, λιγότερο ποσοστό (38,75%) νιώθει τη σιγουριά ότι μπορεί να φτιάξει ολοκληρωμένες σύνθετες εφαρμογές, ενώ ένα 15% πιστεύει ότι γνωρίζει βασικές λειτουργίες του εργαλείου και τέλος, ένα 5% θεωρεί ότι απέκτησε μια γενική εικόνα του τι είναι το app inventor.



Σχήμα 18- Ερώτηση 12/ Σε τι θέση βρίσκομαι μετά την ολοκλήρωση του εγχειριδίου

- **Πρότυπο παράδειγμα**

Το μεγαλύτερο ποσοστό του δείγματος (45%) πιστεύει ότι το πρότυπο παράδειγμα που παρουσιάστηκε στο εγχειρίδιο για την πληρέστερη κατανόηση του εργαλείου app inventor, λειτούργησε θετικά ως προς το στόχο του. Το 30% θεωρεί ότι βοήθησε κατά ένα μεγάλο βαθμό (30%) και ακολουθούν τα ποσοστά 17,5% (ως ένα σημείο) και 7,5% (όχι).



Σχήμα 19- Ερώτηση 11/ Συμβολή του πρότυπου παραδείγματος στην κατανόηση του app inventor

5.2 Πλεονεκτήματα και Μειονεκτήματα του App Inventor

Το App Inventor μπορεί να χρησιμοποιηθεί για εκπαίδευση στον προγραμματισμό με τέτοιο τρόπο που να μην είναι δύσκολο στην εκμάθηση του από τον εκπαιδευόμενο. Αυτό επιτρέπει στους χρήστες ακόμη και από πολύ μικρή ηλικία να εξοικειώνονται με το αντικείμενο του προγραμματισμού και παράλληλα να αποκομίζουν πολύτιμες εμπειρίες.

Το App inventor ανταποκρίνεται στη θεωρία μάθησης του εποικοδομισμού η οποία αναφέρει πως ο μαθητής ανακαλύπτει μόνος του τη γνώση και θεωρείται ερευνητής και σχεδιαστής της δραστηριότητας, όπου μέσω της οποίας θα κατακτήσει αυτή.

Επίσης, θεωρείται ένας από τους πιο γρήγορους και αξιόπιστους τρόπους για τη δημιουργία εφαρμογών για Android κινητά και tablets καθώς παρουσιάζει ένα σύνολο πλεονεκτημάτων, όπως:

- Είναι εύκολο στη χρήση περιβάλλον με πολλές δυνατότητες.
- Είναι εύκολη η εγκατάσταση (για την πρώτη έκδοση του App Inventor ή χωρίς εγκατάσταση για τη 2η έκδοση αυτού).

- Δυνατότητα επίσκεψης του λογαριασμού και των εφαρμογών που δημιουργεί ο χρήστης οποιαδήποτε στιγμή αρκεί να υπάρχει σύνδεση στο διαδίκτυο, καθώς τα πάντα είναι αποθηκευμένα στο cloud.
- Είναι ελεύθερο/ δωρεάν λογισμικό, που σημαίνει ότι δίνεται η δυνατότητα σε όλους να το χρησιμοποιήσουν.
- Ύπαρξη emulator που σημαίνει ότι δεν χρειάζονται πολλές συσκευές.
- Υποστήριξη από τη Google και το MIT.
- Εμφάνιση της εφαρμογής καθώς κατασκευάζεται (είτε μέσω του Android phone ή Tablet συνδέοντας την αντίστοιχη συσκευή με την εφαρμογή, είτε μέσω Android emulator που παρέχει η ίδια η εφαρμογή).
- Παρέχει τη δυνατότητα των πολλαπλών διεργασιών ταυτόχρονα.
- Δυνατότητα βοήθειας και επικοινωνίας μέσω των forums.
- Βοηθά στην ανάπτυξη κριτικής σκέψης διαιρώντας ένα πρόβλημα, σε μικρότερα και απλούστερα με στόχο την ευκολότερη αντιμετώπισή τους.
- Εισάγει το μαθητή στην επίλυση λογικών και μη προβλημάτων.
- Αποφεύγονται τα συντακτικά λάθη, λόγω του ότι οι εντολές είναι έτοιμες.
- Ο χρήστης αντιλαμβάνεται τη σχέση αιτίας και αποτελέσματος.
- Οι χρήστες μαθαίνουν να συνεργάζονται, να κατανοούν τη λογική σειρά των πραγμάτων, να αναπτύσσουν τη φαντασία τους.
- Αντικειμενοστραφές μοντέλο οπτικού προγραμματισμού με δομές ελέγχου καθοδηγούμενες από γεγονότα (event-driven).
- Προάγει τη μάθηση μέσω της λύσης προβλημάτων.

Παρόλα αυτά, υπάρχουν και μειονεκτήματα σχετικά με το app inventor. Μερικά από αυτά είναι:

- Δεν υπάρχει η δυνατότητα μετονομασίας των σελίδων που δημιουργούμε.
- Χρειάζεται απαραίτητα σύνδεση με το διαδίκτυο για να έχουμε πρόσβαση στις εφαρμογές που έχουμε δημιουργήσει.
- Η έκδοσή του περιορίζεται σε λίγες μόνο γλώσσες.
- Παρουσιάζονται καθυστερήσεις σε κάποια μοντέλα, ανάλογα με τα τεχνικά χαρακτηριστικά τους αλλά και την έκδοση που χρησιμοποιούν.

5.3 Μελλοντικές Επεκτάσεις Εφαρμογής

Καθώς η ανάπτυξη και η υλοποίηση μίας Android εφαρμογής είναι μια ιδιαίτερα απαιτητική και ατέρμονη διαδικασία, γίνονται ολοένα και περισσότερες προσπάθειες δημιουργίας εργαλείων που να καθιστούν τη δημιουργία πιο εύκολη και ευχάριστη για τους χρήστες.

Ως επί το πλείστον οι γλώσσες προγραμματισμού είναι ιδιαίτερα δύσκολες στην εκμάθησή τους, πόσο μάλλον για αρχάριους χρήστες που δεν έχουν το απαιτούμενο υπόβαθρο ή δεν έχουν εισχωρήσει στην λογική που ο προγραμματισμός απαιτεί.

Για το λόγο αυτό ενισχύεται όλο και περισσότερο η δημιουργία εργαλείων βασισμένων στην οπτική γλώσσα προγραμματισμού. Οι οπτικές γλώσσες προγραμματισμού (visual programming language, συντ. VPL) επιτρέπουν στο χρήστη τη δημιουργία προγραμμάτων μέσα από το γραφικό χειρισμό προγραμματιστικών στοιχείων (αντί κειμένου). Μια VPL επιτρέπει τον προγραμματισμό με οπτικές εκφράσεις που είναι η τοποθέτηση στο χώρο κειμένου και γραφικών συμβόλων, είτε σαν σύνταξη, είτε σαν βοηθητική σύνταξη.

Στην παρούσα εργασία έγινε μία προσπάθεια δημιουργίας ενός tutorial, το οποίο αποσκοπεί σε μία εν δυνάμει ολοκληρωμένη παρουσίαση του εργαλείου οπτικού προγραμματισμού app inventor. Παρόλα αυτά, υπάρχουν πάντα περιθώρια βελτίωσης.

Έτσι, λαμβάνοντας υπόψη μας και την αξιολόγηση των ερωτηματολογίων που απαντήθηκαν από τους χρήστες, ως μελλοντική επέκταση της εργασίας, θα μπορούσε να αποτελεί ένα εγχειρίδιο που θα περιείχε περισσότερα παραδείγματα ανάπτυξης εφαρμογών με το app inventor αλλά και περαιτέρω ανάπτυξη της πρότυπης εφαρμογής που αναλύεται στο εγχειρίδιο.

Μια επέκταση αυτή της εφαρμογής θα μπορούσε να αποτελεί η κατανόηση από την εφαρμογή των κατηγοριών που δείχνει μεγαλύτερο ενδιαφέρον ο χρήστης, μέσω της μεγαλύτερης επισκεψιμότητας από αυτόν, και να του επιστρέφεται μεγαλύτερο περιεχόμενο από σχετικές σελίδες. Ακόμη, ως κάτι επιπρόσθετο, θα μπορούσε να υπάρχει η δυνατότητα αποστολής οποιουδήποτε περιεχομένου ο χρήστης επιθυμεί

στο e-mail ή σε κάποιο άλλο μέσο κοινωνικής δικτύωσης (facebook, twitter, pinterest, κλπ) ενός άλλου προσώπου.

5.4 Συμπεράσματα

Με το πέρασμα των χρόνων οι αλλαγές που επέρχονται τόσο στην τεχνολογία, όσο και στην εκπαίδευση είναι ραγδαίες. Και ενώ η σχέση των δυο αυτών τομέων αρχικά ήταν μικρή, πλέον είναι άρρηκτα συνδεδεμένοι.

Οι σύγχρονες μαθησιακές θεωρίες δίνουν έμφαση στην απόκτηση εμπειριών από τους μαθητές μέσω της εκπαιδευτικής διαδικασίας. Η επίτευξη του στόχου αυτού επιτάσσει την ένταξη και αξιοποίηση νέων μεθόδων, εργαλείων και τεχνικών στην μαθησιακή διαδικασία.

Σύμφωνα με τους Resnick et al. (2009), “οι χρήστες των νέων τεχνολογιών πρέπει να είναι ικανοί όχι μόνο να καταναλώνουν περιεχόμενο αλλά και να το κατασκευάζουν”. Ως εκ τούτου η νέα γενιά εκπαιδευόμενων θα πρέπει να είναι σε θέση να παράγει περιεχόμενο. Ένας από τους βασικότερους τρόπους να επιτευχθεί αυτό είναι μέσω του προγραμματισμού. Παρόλα αυτά, το αντικείμενο του προγραμματισμού κάθε άλλο παρά απλό θεωρείται από τους νέες χρήστες.

Λόγω αυτής της δυσκολίας έχουν αναπτυχθεί διάφορα ειδικά εκπαιδευτικά περιβάλλοντα προγραμματισμού, μεταξύ των οποίων και το app inventor με το οποίο ασχοληθήκαμε στην παρούσα εργασία. Πρόκειται για ένα εξαιρετικό εκπαιδευτικό εργαλείο το οποίο μπορεί ο εκπαιδευτικός να αξιοποιήσει στο έπακρο για την εκπαιδευτική διαδικασία, είτε ως μέσω αξιολόγησης των γνώσεων που αποκτά ο μαθητής, είτε ως προγραμματιστικό εργαλείο. Οι γραφικές γλώσσες, δίνουν τη δυνατότητα να ασχοληθούν οι μαθητές με το αντικείμενο του προγραμματισμού από μικρή ηλικία. Παράλληλα, αποκτούν πολλά οφέλη όπως πρωτοβουλία, κριτική ικανότητα κτλ. Αυτό είναι πολύ σημαντικό, γιατί τα παιδιά από πολύ νωρίς εξοικειώνονται με τις πλέον σύγχρονες τεχνολογίες και αυτό οδηγεί στην ακόμα ταχύτερη ανάπτυξη της τεχνολογίας, μέσω της εμφάνισης νέων εφαρμογών και εργαλείων.

Δημιουργώντας, λοιπόν, αυτό το εγχειρίδιο χρήσης του MIT app inventor, συμπεριλαμβανομένης και της πρότυπης εφαρμογής, προσπαθήσαμε να δώσουμε μια

πιο σφαιρική και ολοκληρωμένη προσέγγιση του εργαλείου αυτού καθώς διαπιστώθηκε ότι υπήρχε έλλειψη επαρκών πληροφοριών και αναπτυγμένων εφαρμογών. Βάση του ερωτηματολογίου το οποίο συμπληρώθηκε από ένα αρκετά μεγάλο αριθμό χρηστών, κρίθηκε ότι το tutorial πετυχαίνει τους στόχους του ως ένα ικανοποιητικό βαθμό με περιθώρια φυσικά βελτίωσης και εξέλιξης.

Βιβλιογραφικές Αναφορές

Βασιλάκης, Β., Χατζηνικολάκης Γ., Σύλλογος Εκπαιδευτικών Πληροφορικής Χίου (2014). Προγραμματισμός σε App Inventor. Ανακτήθηκε 23 Οκτωβρίου 2014, από [http://sepchiou.gr/docs/appinventor_code_club/App%20Inventor%20Programming%20\(Part%20A\).pdf](http://sepchiou.gr/docs/appinventor_code_club/App%20Inventor%20Programming%20(Part%20A).pdf)

Βέμμου, Ε. & Γκότζος, Δ. (2009). Αξιολόγηση των λογισμικών: Kidspiration, Βικιπαίδεια και 21' Εν πλω. (Αδημοσίευτη εργασία στα πλαίσια του μαθήματος «Χρήση και Αξιολόγηση του Εκπαιδευτικού Λογισμικού», Πανεπιστήμιο Αθήνας, 2009). Ανακτήθηκε 2 Ιουνίου 2015, από <http://users.sch.gr/dmgtz/Ergasies/wikipedia,%20kidspiration,%2021%20en%20plo.pdf>

Γιακουμέλης, Εμμ. (2009). Υλοποίηση εφαρμογής κινητού τηλεφώνου με χρήση της πλατφόρμας Google Android. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2009). Ανακτήθηκε 7 Νοεμβρίου 2014, από <http://digilib.lib.unipi.gr/dspace/bitstream/unipi/3492/1/Giakoumelis.pdf>

Δασκαλάκης, Ελ. (2012). Ανάλυση και χρήση εργαλείων για τον προγραμματισμό εφαρμογών σε android. (Αδημοσίευτη πτυχιακή εργασία, Α-ΤΕΙ Κρήτης, 2012). Ανακτήθηκε 3 Νοεμβρίου 2014, από <http://nefeli.lib.teicrete.gr/browse/stef/epp/2012/DaskalakisEleftherios/document-1332243045-389070-17065.tkl>

Δημογιάννη, Δ. (2013). *Σχεδίαση και Ανάπτυξη Διαδικτυακής Εφαρμογής Υποστήριξης Μελετών Χρηστών*. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πατρών, Πολυτεχνική σχολή, 2013). Ανακτήθηκε 3 Νοεμβρίου 2014, από http://nemertes.lis.upatras.gr/jspui/bitstream/10889/6409/1/Diploma_thesis_Dimogian_nis_Dimitrios.pdf

Ζαχαριάδης, Στ. (2014). Σχεδίαση και ανάπτυξη εφαρμογής android για διαχείριση έξυπνου σπιτιού. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2014). Ανακτήθηκε 7 Νοεμβρίου 2014, από <http://digilib.lib.unipi.gr/dspace/handle/unipi/5970>

Ζερβούδη, Ε. (2014). Μελέτη του ανταγωνισμού της αγοράς των λειτουργικών συστημάτων κινητών συσκευών. (Αδημοσίευτη πτυχιακή εργασία, Χαροκόπειο Πανεπιστήμιο, 2014). Ανακτήθηκε 5 Νοεμβρίου 2014, από <http://estia.hua.gr:8080/dspace/handle/123456789/2359>

Ζωγόπουλος, Ε. & Μπαγούλη, Κ. (2011). *Λογισμικά διαδικτυακής μαθησιακής εξέτασης και αξιολόγησης. Μια πρώτη προσέγγιση*. 2^ο Πανελλήνιο συνέδριο. Πάτρα 28-30/04/2011.

Θεοχαρόπουλος, Ι. (2008). Αξιοποίηση της Ενοποιημένης Πλατφόρμας Eclipse για την Ανάπτυξη και διαχείριση Ανοιχτού Εκπαιδευτικού Λογισμικού σε Java. Άρθρο που παρουσιάστηκε στο Πανελλήνιο εκπαιδευτικό συνέδριο Ημαθίας «Ψηφιακό υλικό για την υποστήριξη του παιδαγωγικού έργου των εκπαιδευτικών Πρωτοβάθμιας & Δευτεροβάθμιας Εκπαίδευσης», 9-11 Μαΐου 2008. Ανακτήθηκε 10 Φεβρουαρίου 2015, από <http://www.pe04.net/lib/exe/fetch.php?media=bio:papers:pap6.pdf>

Καπνόπουλος, Κ. (2010). Ανάπτυξη κινητού συστήματος ασφαλούς διαχείρισης συναντήσεων. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Μακεδονίας, 2010). Ανακτήθηκε 28 Μαΐου 2015, από https://dspace.lib.uom.gr/bitstream/2159/13879/1/Kapnopoulos_Msc2010.pdf

Κατσαράκης, Ζ. (2013). Ανάπτυξη Android εφαρμογής με το MIT app inventor. (Αδημοσίευτη πτυχιακή εργασία. Α-ΤΕΙ Κρήτης, 2013). Ανακτήθηκε 15 Μαΐου 2015, από <http://nefeli.lib.teicrete.gr/browse/stef/epp/2013/KatsarakisZacharias/attached-document-1387448598-488980-30038/KatsarakisZacharias2013.pdf>

Κολόσακα, Ολ. (2011). Εκπαιδευτικά παιχνίδια με το εργαλείο προγραμματισμού Kodu. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2011). Ανακτήθηκε 5 Μαρτίου 2015, από <http://digilib.lib.unipi.gr/dspace/handle/unipi/4107>

Κουφόπουλος, Γ. (2011). Σχεδιασμός και ανάπτυξη εφαρμογής για Mobile Social Location Based Space. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2011). Ανακτήθηκε 3 Νοεμβρίου 2014, από <http://digilib.lib.unipi.gr/dspace/bitstream/unipi/4216/1/Koufopoulos.pdf>

Μαυρόγιαννη, Σ. (2007). Semantic Web Tutorial. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2007). Ανακτήθηκε 17 Δεκεμβρίου 2014, από <http://digilib.lib.unipi.gr/dspace/bitstream/unipi/2230/1/Maurogianni.pdf>

Μέφερης, Κ. (2010). Σχεδιασμός και ανάπτυξη εφαρμογής για Mobile Social Space. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2010). Ανακτήθηκε 5 Νοεμβρίου 2014, από <http://digilib.lib.unipi.gr/dspace/bitstream/unipi/4823/1/Meferis.pdf>

Μονάδα καινοτομίας και επιχειρηματικότητας (2012). Ανάπτυξη μελετών περιπτώσεων, οδηγός προδιαγραφών. (Πανεπιστήμιο Αιγαίου, 2012). Ανακτήθηκε 18 Μαΐου 2015, από <file:///C:/Users/%CE%95%CE%BB%CE%AD%CE%BD%CE%B7/Downloads/case%20writing%20manual.pdf>

Ξανθόπουλος, Σ. (2013). Επισκόπηση προσεγγίσεων ανάπτυξης εφαρμογών για έξυπνες κινητές συσκευές και ανάπτυξη πρότυπης εφαρμογής. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Μακεδονίας, 2013). Ανακτήθηκε 3 Νοεμβρίου 2014, από <https://dspace.lib.uom.gr/bitstream/2159/15490/6/XanthopoulosSpyridonMsc2013.pdf>

Παλαιοκρασάς, Στ. (2007). Οδηγίες για τη διενέργεια μελέτης περίπτωσης επιχειρηματικότητας. Άρθρο που παρουσιάστηκε στα πλαίσια του έργου ΕΠΕΑΕΚ II, 2007. Ανακτήθηκε 13 Ιανουαρίου 2015, από <http://repository.edulll.gr/edulll/retrieve/3458/1022.pdf>

Παλάσκας, Αλ. (2014). Διεπαφές Φορητών Συσκευών και Τεχνολογίες Αλληλεπίδρασης - Mobile Interfaces and Interaction Technologies. (Αδημοσίευτη διπλωματική εργασία, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης, 2014). Ανακτήθηκε 18 Ιουνίου 2015, από <http://docplayer.gr/3021250-Diepafes-foriton-syskeyon-kai-tehnologies-allilepidrasis-mobile-interfaces-and-interaction-technologies.html>

Παπαδάκης, Στ. (2014). Διδακτικό σενάριο: Δημιουργία παιχνιδιού στο περιβάλλον App Inventor. Άρθρο που παρουσιάστηκε στο 7ο Πανελλήνιο Συνέδριο «Διδακτική της Πληροφορικής» στο Πανεπιστήμιο Κρήτης, 3-5 Οκτωβρίου 2014. Ανακτήθηκε 19 Ιουνίου 2015, από

http://www.researchgate.net/publication/275099638_A_learning_scenario_for_teaching_programming_in_Secondary_Education_Creating_a_game_with_App_Inventor_for_Android.%28%20App%20Inventor%29

Παπαδάκης, Στ., Καλογιαννάκης, Μ., Ζαράνης, Ν. (2013). Δημιουργώντας εφαρμογές για έξυπνες φορητές συσκευές με το AppInventor. Άρθρο που παρουσιάστηκε στο 7ο Πανελλήνιο Συνέδριο Καθηγητών Πληροφορικής «Η Πληροφορική στην Πρωτοβάθμια και Δευτεροβάθμια Εκπαίδευση - Προκλήσεις και Προοπτικές», Θεσσαλονίκη, 12- 14 Απριλίου 2013. Ανακτήθηκε 15 Μαΐου 2015, από <http://docplayer.gr/830117-Dimioyrgontas-efarmoges-gia-exypnes-forites-syskeyes-me-to-appinventor.html>

Πνευματικός, Δ. (2013). Σχεδίαση και ανάπτυξη εφαρμογής με Web Services. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2013). Ανακτήθηκε 5 Νοεμβρίου 2014, από <http://digilib.lib.unipi.gr/dspace/bitstream/unipi/5165/1/Pneumatikos.pdf>

Ρόντος, Κ. και Παπάνης, Ε. (2007). *Μέθοδοι συλλογής στατιστικών δεδομένων*. Στατιστική Έρευνα, Εκδόσεις Σιδέρη.

Σοφίος, Κ. (2013). Ανάλυση, σχεδιασμός και υλοποίηση εφαρμογής για έξυπνες συσκευές για επικοινωνία με γεωγραφικό πληροφοριακό σύστημα μέσω διαδικτυακών υπηρεσιών. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2013). Ανακτήθηκε 5 Νοεμβρίου 2014, από <http://digilib.lib.unipi.gr/dspace/bitstream/unipi/5726/1/Sofios.pdf>

Τζήμας, Δ.& Ζήκος, Γ. (2015). Σχετικά με τη διδασκαλία της Python. Άρθρο που παρουσιάστηκε στο 9ο Πανελλήνιο συνέδριο καθηγητών πληροφορικής Καστοριάς «Η Πληροφορική στην Πρωτοβάθμια και Δευτεροβάθμια Εκπαίδευση, Καινοτόμες παιδαγωγικές πρακτικές», 24-26 Απριλίου 2015. Ανακτήθηκε 15 Μαΐου 2015, από http://users.sch.gr/bstefan/keplinet/files/praktika_pekap_2015/labs/E138-tzimas-2.pdf

Φαρμάκης, Χ. & Φιλίππιδης, Φ. (2013). Παρουσίαση του λειτουργικού συστήματος Android και ανάπτυξη διαδραστικών εφαρμογών. (Αδημοσίευτη πτυχιακή εργασία, Α-ΤΕΙ Λάρισας, 2013). Ανακτήθηκε 18 Ιουνίου 2015, από <http://www.teilar.gr/dbData/ProfAnn/profann-9667dcb7.pdf>

Φουντουλάκη, Μ. (2011). *Η συμβολή του Scratch στη διδασκαλία του προγραμματισμού στη Β'θμια εκπαίδευση*. (Αδημοσίευτη διπλωματική εργασία, Πανεπιστήμιο Πειραιά, 2011). Ανακτήθηκε 13 Ιανουαρίου 2015, από <http://dione.lib.unipi.gr/xmlui/handle/unipi/4155>

Φωτόπουλος, Β., Φαλιέρης Σ., Νικολόπουλος, Γ., Νικολόπουλος, Κ. (2007). *Πολυμέσα και Web Publishing*. Άρθρο που παρουσιάστηκε στα πλαίσια του έργου ΕΠΕΑΕΚ ΙΙ, 2007. Ανακτήθηκε 10 Νοεμβρίου 2014, από <http://repository.edulll.gr/edulll/retrieve/2873/887.pdf>

Παράρτημα

Ερωτηματολόγιο

Παρακαλείσθε να απαντήσετε με ειλικρίνεια, καθώς το συγκεκριμένο ερωτηματολόγιο έχει δημιουργηθεί για να εξυπηρετήσει στατιστικούς σκοπούς.

Σημείωση: Το ερωτηματολόγιο είναι ανώνυμο.

1. Ποιο είναι το φύλο σας;

- Γυναίκα
- Άντρας

2. Σε ποια ηλικιακή ομάδα ανήκετε;

- 15-20
- 20-30
- 30-40
- 40-50
- 50 και άνω

3. Ποια είναι η εκπαίδευσή σας;

- Πρωτοβάθμια
- Δευτεροβάθμια
- Τριτοβάθμια
- Άλλο (προσδιορίστε)

4. Πόσες ώρες την εβδομάδα περνάτε στον υπολογιστή;

- <2

- 2-4
- 5-9
- 10-14
- >14

5. Χρησιμοποιείται τον υπολογιστή για:

	Συνέχεια	Πολλές φορές	Κάποιες φορές	Λίγες φορές	Σχεδόν ποτέ
Να σερφάρετε στο διαδίκτυο					
Την εργασία/σχολείο σας					
Επικοινωνία με φίλους/συνεργάτες (chat, e-mail, forum, VoIP, κοινωνική δικτύωση, κλπ)					
Καθημερινές δραστηριότητες (web banking, ηλεκτρονικές αγορές, αναζήτηση πληροφοριών, κλπ)					
Ψυχαγωγία					
Ενημέρωση					
Άλλο (προσδιορίστε)					

6. Έχετε ασχοληθεί με τον προγραμματισμό;

- Πολύ
- Αρκετά
- Μέτρια
- Λίγο
- Καθόλου

7. Έχετε ξαναασχοληθεί με παρόμοια εργαλεία όπως το app inventor;

- Ναι
- Όχι
- Δεν είμαι σίγουρος (προσδιορίστε)

8. Το app inventor ως εργαλείο ανάπτυξης μιας εφαρμογής, είναι:

	Πάρα πολύ	Πολύ	Έτσι και έτσι	Λίγο	Καθόλου
Εύκολο					
Περίπλοκο					
Βαρετό					
Ενδιαφέρον					

9. Μπορείτε να δημιουργήσετε την δική σας εφαρμογή μέσω του app inventor;

- Ναι
- Σε μεγάλο βαθμό
- Ως ένα σημείο
- Όχι

10. Μέχρι ποιο σημείο σας βοήθησε το εγχειρίδιο να κατανοήσετε το εργαλείο του app inventor;

- Πολύ
- Σε μεγάλο βαθμό
- Ως ένα σημείο

- Καθόλου

11. Σας βοήθησε το παράδειγμα/ εφαρμογή να κατανοήσετε το εργαλείο app inventor;

- Ναι
- Σε μεγάλο βαθμό
- Ως ένα σημείο
- Όχι

12. Με την ολοκλήρωση του tutorial, είστε σε θέση να:

- Φτιάχνετε ολοκληρωμένες, σύνθετες εφαρμογές με τη χρήση του app inventor.
- Φτιάχνετε απλές εφαρμογές με το app inventor.
- Κατανοείτε βασικές λειτουργίες του app inventor.
- Γνωρίζετε τι είναι γενικά το app inventor.

13. Τι θα θέλατε να περιέχει σε μεγαλύτερο βαθμό το tutorial;

- Ανάλυση των εργαλείων οπτικού προγραμματισμού.
- Οδηγίες για το φορμάρισμα.
- Ανάλυση της οθόνης του designer.
- Ανάλυση του blocks editor.
- Παραδείγματα εφαρμογών.
- Ανάλυση δημιουργίας της πρότυπης εφαρμογής.
- Θεωρώ ότι το εγχειρίδιο είναι πλήρες.
- Άλλο (προσδιορίστε).

14. Το tutorial σας βοήθησε:

	Πολύ	Αρκετά	Μέτρια	Λίγο	Σχεδόν καθόλου
Στην κατανόηση του app inventor					
Να πειραματιστείτε					

Στην κατανόηση του προγραμματισμού γενικότερα					
Στην απόκτηση γνώσεων					
Να διασκεδάσετε					

15. Θα συνεχίσετε την δημιουργία εφαρμογών με το app inventor;

- Ναι
- Όχι
- Δεν είμαι σίγουρος