

Πανεπιστήμιο Θεσσαλίας
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

Πτυχιακή Εργασία
του Τσότσου Παναγιώτη

**Εφαρμογή επίλυσης πολυωνυμικών εξισώσεων σε
πλατφόρμα android**

Επιβλέπουσα Καθηγήτρια: Δασκαλοπούλου Ασπασία

Βόλος, Νοέμβριος 2016

Περίληψη

Τα μαθηματικά αποτελούν αναπόσπαστο κομμάτι γνώσεων για τον άνθρωπο. Κατά τη μαθητική του πορεία και ανάπτυξη, ο άνθρωπος καλείται να αναγνωρίσει και να μάθει όλο και περισσότερα μαθηματικά θεωρήματα, πράξεις, αριθμούς και συσχετίσεις. Με το πέρασμα των χρόνων και την ανάπτυξη της τεχνολογίας, έχουν δημιουργηθεί νέοι τρόποι για την πιο άμεση αλλά και πρακτική μετάδοση γνώσης των μαθηματικών θεωρημάτων.

Έναν από αυτούς τους τρόπους κληθήκαμε να αναπτύξουμε στην παρούσα εργασία. Πιο συγκεκριμένα, η παρούσα εργασία αποτελείται από την υλοποίηση μιας εφαρμογής για μαθηματικές πράξεις, όπου ο χρήστης θα μπορεί να επιλέξει το είδος των πράξεων που θέλει να εκτελέσει και η εφαρμογή θα παράγει το κατάλληλο αποτέλεσμα. Η εφαρμογή αυτή αποτελεί κομμάτι τόσο εκπαίδευσης, όσο και γνώσης και μπορεί να χρησιμοποιηθεί για εκπαιδευτικούς σκοπούς.

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω την καθηγήτριά μου, Δασκαλοπούλου Ασπασία, αλλά και τον καθηγητή κύριο Ακρίτα Αλκιβιάδη ο οποίος μου εμπιστεύτηκε το συγκεκριμένο θέμα και με καθοδηγούσε καθ' όλη τη διάρκεια της εκπόνησής της εργασίας.

Στη συνέχεια θα ήθελα να ευχαριστήσω την οικογένειά μου, η οποία με στήριξε και ήταν δίπλα μου σε όλα μου τα βήματα.

Ακόμη, δεν θα μπορούσα να μην ευχαριστήσω τους φίλους μου, οι οποίοι με υποστήριξαν, με βοήθησαν και μου χάρισαν ωραίες στιγμές στην μέχρι τώρα πορεία μας.

Περιεχόμενα

Περίληψη	3
Ευχαριστίες	5
Περιεχόμενα.....	6
Κατάλογος Πινάκων.....	9
Κατάλογος Εικόνων	11
Κεφάλαιο 1	13
Εισαγωγή	13
1.1 Αντικείμενο διπλωματικής εργασίας	13
1.2 Σύνοψη διπλωματικής εργασίας.....	14
Κεφάλαιο 2	17
Θεωρητικό υπόβαθρο	17
2.1 Εφαρμογές.....	17
2.2 Λειτουργικά συστήματα	18
2.3 Λογισμικό Android.....	19
2.4 Android Studio	22
Κεφάλαιο 3	23
Θεώρημα του Βίνσεντ και δημιουργία API.....	23
3.1 Θεώρημα του Βίνσεντ.....	23
Εξήγηση του αλγόριθμου.....	24
3.2 Wolfram Mathematica	25
3.3 Διεπαφή προγραμματισμού Εφαρμογών (API).....	28
Κεφάλαιο 4	31
Σχεδιασμός και ανάπτυξη εφαρμογής.....	31
4.1 Απαιτήσεις συστήματος.....	31

4.2 Αρχιτεκτονική εφαρμογής	32
4.3 Στοιχεία Extensible Markup Language (XML)	33
4.3.1 Εμφάνιση (Layout)	34
4.3.2 Εισαγωγή κουμπιού (Button)	36
4.3.3 Εισαγωγή κειμένου (Text View)	37
4.3.4 Ενέργειες (Activity)	38
4.3.5 AndroidManifest	40
4.4 Σχεδιασμός εφαρμογής	41
Κεφάλαιο 5	44
Παρουσίαση εφαρμογής	44
5.1 Αρχική οθόνη	44
5.2 Επίλυση εξισώσεων	45
5.3 Ταυτότητες	46
5.4 Ασκήσεις εξάσκησης	49
Κεφάλαιο 6	52
Συμπεράσματα	52
Βιβλιογραφία	54

Κατάλογος Πινάκων

Πίνακας 1: Πίνακας εκδόσεων λογισμικού Android [1].....	21
---	----

Κατάλογος Εικόνων

Εικόνα 1: Απεικόνιση πλατφόρμας Android Studio.....	22
Εικόνα 2: Περιβάλλον εργασίας εργαλείου Wolfram Mathematica.....	27
Εικόνα 3: Σχεδιάγραμμα αποστολής / λήψης αποτελέσματος μέσω του API της Mathematica.....	29
Εικόνα 4: Διαβάθμιση Layout στο Android.....	36
Εικόνα 5: Εικόνα κουμπιού σε μια εφαρμογή.....	37
Εικόνα 6: Εμφάνιση Textview σε μια εφαρμογή.....	38
Εικόνα 7: Διάγραμμα μεθόδων στο Activity.....	39
Εικόνα 8: Αρχική οθόνη εφαρμογής.....	42
Εικόνα 9: Εξισώσεις Α' βαθμού.....	42
Εικόνα 10: Ασκήσεις εξάσκησης τις οποίες θα μπορεί να κάνει ο χρήστης και να δει αν το αποτέλεσμά του είναι ίδιο με το αποτέλεσμα που δίνεται στην εφαρμογή.....	43
Εικόνα 11: Αρχική καρτέλα εφαρμογής.....	45
Εικόνα 12: Καρτέλα επίλυσης εξισώσεων.....	46
Εικόνα 14: Επιλογή είδους ταυτοτήτων στην κατηγορία "Ταυτότητες".....	47
Εικόνα 15: Καρτέλα επίλυσης είδους "Τετράγωνο διαφοράς".....	48
Εικόνα 16: Καρτέλα επίλυσης είδους "Κύβος διαφοράς".....	49
Εικόνα 17: Καρτέλα "Ασκήσεις εξάσκησης".....	50

Κεφάλαιο 1

Εισαγωγή

Στο παρόν κεφάλαιο γίνεται μια μικρή παρουσίαση του αντικειμένου της εργασίας, της δομής αυτού και των σκοπών του. Ακόμη, παρουσιάζεται η σύνοψη των επόμενων κεφαλαίων ώστε να μπορέσει ο αναγνώστης να κατατοπιστεί επαρκώς σχετικά με το τι πρόκειται να διαβάσει στις επόμενες σελίδες.

1.1 Αντικείμενο διπλωματικής εργασίας

Ο άνθρωπος είναι ένα ον με αυξημένη ανάγκη για γνώση, εκπαίδευση και μόρφωση σε γενικότερο πλαίσιο. Τα τελευταία χρόνια η εκπαίδευση και η τεχνολογία είναι δύο τομείς με παράλληλη ανάπτυξη και στήριξη. Η τεχνολογία πλέον αποτελεί αναπόσπαστο κομμάτι των μαθητών και γενικότερα των ανθρώπων, και πλέον μέσα από διάφορες εφαρμογές, συμβάλει και στην εκπαίδευσή τους.

Η εφαρμογή την οποία κληθήκαμε να αναπτύξουμε έχει εκπαιδευτικούς σκοπούς. Στόχος είναι να βοηθηθεί ο χρήστης στη λύση των μαθηματικών ασκήσεων όχι μόνο από την παραγωγή αποτελέσματος, αλλά και από την επίλυση κάποιων ενδεικτικών ασκήσεων. Ουσιαστικά, ο χρήστης καλείται να επιλέξει το είδος των ασκήσεων των οποίων το αποτέλεσμα θέλει να βρει και το σύστημα θα παράγει το κατάλληλο αποτέλεσμα.

Στην εφαρμογή παρέχονται δύο κατηγορίες αναζήτησης αποτελέσματος, οι εξισώσεις οποιουδήποτε βαθμού και οι ταυτότητες. Εφόσον έχει επιλέξει ο χρήστης την κατηγορία που θέλει, εισάγει την κατάλληλη εξίσωση και μετά από τους

κατάλληλους ελέγχους και συναρτήσεις παράγεται το επιθυμητό αποτέλεσμα. Στην πρώτη κατηγορία, πέραν του αποτελέσματος, ο χρήστης λαμβάνει και το διάστημα στο οποίο βρίσκονται οι ρίζες της εξίσωσης. Η εύρεση του διαστήματος αυτού βασίζεται στο θεώρημα του Βίνσεντ, το οποίο απομονώνει τις πραγματικές ρίζες πολυωνύμων με ρητούς συντελεστές. Για την ολοκλήρωση και την επιτυχή λήψη του διαστήματος αυτού έγινε διασύνδεση, μέσω cloud, της εφαρμογής με το εργαλείο Mathematica.

Εκτός από την παραγωγή αποτελέσματος, η εφαρμογή που αναπτύχθηκε δίνει τη δυνατότητα στο χρήστη να βρει διάφορες, διαφορετικές ασκήσεις στο περιεχόμενο κάθε κατηγορίας. Η ασκήσεις αυτές υπάρχει η δυνατότητα να είναι κοινές σε κάθε εφαρμογή, μπορεί όμως να εισάγει και κάποιος καθηγητής της επιθυμητές ασκήσεις που θέλει, ώστε να οργανώσει τη μελέτη των μαθητών με το δικό του τρόπο.

1.2 Σύνοψη διπλωματικής εργασίας

Το κείμενο κατανέμεται ομοιόμορφα σε έξι κεφάλαια καθένα από τα οποία πραγματεύεται ένα διαφορετικό ζήτημα που αφορά το θέμα της εργασίας. Κάθε κεφάλαιο έχει ως στόχο να επεξηγήσει με ακρίβεια στον αναγνώστη τη λειτουργία της εφαρμογής αλλά και τον τρόπο με τον οποίο αυτή δημιουργήθηκε.

Στο εν λόγω κεφάλαιο γίνεται μια εισαγωγή στο αντικείμενο το οποίο πραγματεύεται η εργασία. Η παρουσίαση του αντικειμένου της εργασίας καθώς και του σκοπού ανάπτυξης της αποτελεί το πιο σημαντικό κομμάτι της εργασίας, καθώς για να γίνει κατανοητός ο τρόπος ανάπτυξης της πρέπει πρώτα να γίνουν αντιληπτοί οι στόχοι και ο σκοπός της.

Στο δεύτερο κεφάλαιο, παρουσιάζονται οι διάφορες τεχνολογίες και οι πλατφόρμες που χρησιμοποιούνται για την ανάπτυξη εφαρμογών σε έξυπνα κινητά. Ακόμη, παρουσιάζεται εκτενέστερα το λογισμικό Android, το οποίο αποτελεί βάση για την ανάπτυξη της παρούσας εφαρμογής. Στο κεφάλαιο αυτό παρουσιάζεται επίσης και η γλώσσα ανάπτυξης των εφαρμογών η οποία δεν είναι άλλη από την γλώσσα προγραμματισμού Java.

Εφόσον έχουν γίνει αντιληπτές οι τεχνολογίες και τα μέσα τα οποία θα χρησιμοποιηθούν για την ανάπτυξη της εφαρμογής, ακολουθεί το επόμενο, τρίτο κεφάλαιο. Στο κεφάλαιο αυτό παρουσιάζεται αναλυτικά η διασύνδεση της εφαρμογής με το εργαλείο Mathematica μέσω του cloud.

Το τέταρτο κεφάλαιο αναλύεται ο σχεδιασμός της εφαρμογής. Ο σχεδιασμός της εφαρμογής αφορά την ανάλυση των απαιτήσεων, της αρχιτεκτονικής της εφαρμογής και την ανάθεση των ρόλων σύμφωνα με τους οποίους θα αναπτυχθεί. Καταγράφονται ουσιαστικά τα στοιχεία τα οποία αποτελούν την κύρια δομή για τη δημιουργία της εφαρμογής και θέτουν τον τρόπο λειτουργίας και ανάπτυξής της.

Στο πέμπτο κεφάλαιο, γίνεται παρουσίαση της λειτουργίας της εφαρμογής, μέσα από εικόνες και σχεδιαγράμματα.

Τέλος, το τελευταίο κεφάλαιο συνοψίζει όλα τα στοιχεία που έχουν παρουσιαστεί στην υπόλοιπη εργασία και καταγράφει τα συμπεράσματα στα οποία έχει οδηγηθεί ο συγγραφέας.

Κεφάλαιο 2

Θεωρητικό υπόβαθρο

Βασικό δομικό στοιχείο για τη σχεδίαση και την ανάπτυξη οποιουδήποτε συστήματος και εφαρμογής αποτελεί το θεωρητικό υπόβαθρο. Το παρόν κεφάλαιο ασχολείται με την κατηγοριοποίηση των εφαρμογών Android, το ιστορικό του λογισμικού αυτού, τα παραπλήσια λειτουργικά συστήματα τα οποία υπάρχουν καθώς και με το θεώρημα το οποίο αποτελεί βασικό δομικό στοιχείο της εφαρμογής.

2.1 Εφαρμογές

Οι εφαρμογές είναι μια γενικότερη κατηγορία προγραμμάτων που μπορούν να χρησιμοποιηθούν από αρκετές συσκευές. Σημαντικό τεχνολογικό επίτευγμα μπορεί να θεωρηθεί η ανάπτυξη εφαρμογών για κινητές συσκευές, όπως τα κινητά τηλέφωνα και τα tablet. Οι εφαρμογές για κινητά τηλέφωνα αποτελούν εφαρμογές λογισμικού που έχουν σχεδιαστεί για να λειτουργούν σε έξυπνα κινητά και άλλες φορητές συσκευές.

Οι εφαρμογές του κάθε λογισμικού είναι διαθέσιμες μέσω των πλατφόρμων διανομής εφαρμογών, όπως τα Apple App Store, Google Play και Windows Phone Store. Κάθε εφαρμογή σχεδιάζεται για ένα συγκεκριμένο λειτουργικό σύστημα όπως Android, Windows ή iOS, ωστόσο μερικές φορές μπορούν να χρησιμοποιηθούν και σε φορητούς ή επιτραπέζιους υπολογιστές.

Οι εφαρμογές για κινητές συσκευές αναπτύχθηκαν με κύριο γνώμονα την επέκταση, την αύξηση παραγωγικότητας και κέρδους των επιχειρήσεων. Κάτι τέτοιο είναι εφικτό μέσα από μια εφαρμογή κάποιας κινητής συσκευής καθώς αυξάνεται ο έλεγχος, οι χρήστες μπορούν να αλληλεπιδρούν παντού και πάντα μεταξύ τους και τα δεδομένα μπορούν πλέον να διανέμονται σε όλους τους χρήστες χωρίς κάποιο περιορισμό. Ακόμη, εφόσον γίνει εγκατάσταση της εφαρμογής σε μία κινητή έξυπνη συσκευή μπορούν να γίνουν όλων των ειδών λειτουργίες και οι εργασίες εκτός γραφείου, χωρίς να απαιτείται η παρουσία όλων των υπαλλήλων.

Οι αρχικές εφαρμογές που αναπτύχθηκαν προσέφεραν βασικές δομές, όπως ημερολόγιο, ανάκτηση πληροφοριών, επαφών και χρηματιστηριακών μετοχών. Ωστόσο όσο μεγάλωνε η ζήτηση σε συνδυασμό με την αύξηση και τη διαθεσιμότητα των εργαλείων για τη δημιουργία τους, οδήγησε στην ταχεία επέκταση των εφαρμογών και σε άλλες κατηγορίες, όπως παιχνίδια, GPS και υπηρεσίες εύρεσης θέσης, τράπεζες και αγορές εισιτηρίων. Η έκρηξη σε αριθμό και ποικιλία από εφαρμογές έκανε την ανακάλυψη μία πρόκληση. Οι πιο γνωστές ηλεκτρονικές εταιρείας ανάπτυξης εφαρμογών και κινητών συσκευών είναι οι:

- App Store
- BlackBerry World
- Google Play
- Nokia Store
- Windows Phone Store
- Windows Store
- Samsung Apps Store

2.2 Λειτουργικά συστήματα

Λειτουργικό σύστημα, στην επιστήμη της πληροφορικής, ονομάζεται το λογισμικό του υπολογιστή που είναι υπεύθυνο για τη διαχείριση και το συντονισμό των εργασιών, καθώς και την κατανομή των διαθέσιμων πόρων. Το λειτουργικό σύστημα παρέχει ένα θεμέλιο, ένα μεσολαβητικό επίπεδο λογικής διασύνδεσης μεταξύ λογισμικού και

υλικού, διαμέσου του οποίου οι εφαρμογές αλληλεπιδρούν εμμέσως με τον υπολογιστή. Μια από τις κεντρικές αρμοδιότητες του λειτουργικού συστήματος είναι η διαχείριση του υλικού, απαλλάσσοντας έτσι το λογισμικό από τον άμεσο και επίπονο χειρισμό του υπολογιστή και καθιστώντας ευκολότερο το χειρισμό και τον προγραμματισμό του. Σχεδόν όλοι οι υπολογιστές και κινητές συσκευές χρησιμοποιούν έναν τύπο λειτουργικού συστήματος. Στη σημερινή εποχή έχει αναπτυχθεί μια πληθώρα λογισμικών ανάλογη και της πληθώρας συσκευών, κινητών ή μη, που υπάρχει στην αγορά.

Τα λειτουργικά συστήματα (Operating Systems) των κινητών τηλεφώνων του παρελθόντος έχουν αντικατασταθεί από πιο σύγχρονα λειτουργικά τα οποία χρησιμοποιούνται από τα σύγχρονα έξυπνα κινητά (smartphones). Συνήθως κάθε συσκευή μπορεί να λάβει πολλές ενημερωμένες εκδόσεις λογισμικού κατά τη διάρκεια ζωής της. Η επέκταση του κάθε λογισμικού συσχετίζεται με τις ανάγκες της αγοράς, τις ανάγκες της συσκευής και των εφαρμογών οι οποίες ολοένα και αυξάνονται. Τα πιο γνωστά λειτουργικά συστήματα για smartphones είναι:

- Android της Google
- Symbian της Nokia
- iOS της Apple
- BlackBerry της RIM
- Windows Phone της Microsoft
- Bada της Samsung

2.3 Λογισμικό Android

Το Android αποτελεί λογισμικό για συσκευές κινητής τηλεφωνίας, του οποίου ο πυρήνας είναι εκείνος του λειτουργικού συστήματος Linux. Το λογισμικό αυτό επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού Java, μέσω μιας πληθώρας βιβλιοθηκών ανεπτυγμένων από τη Google.

Τα πλεονεκτήματα του λογισμικού αυτού είναι αρκετά:

- Ανοιχτή πλατφόρμα ανάπτυξης.
- Απλοποιημένη ανάπτυξη. Τόσο οι πελάτες όσο και οι προγραμματιστές παράγουν εύκολα εργαλεία.
- Ισχυρό λειτουργικό σύστημα. Επειδή, η αρχική ανάπτυξη του λογισμικού γίνεται με τη βοήθεια της εταιρείας Google, το λειτουργικό σύστημα έχει αυξημένη ασφάλεια, επεκτασιμότητα και ανανέωση.
- Υποστηρίζεται από ένα πλήρες σύνολο βιβλιοθηκών, με πολλαπλές δυνατότητες χρήσης πολυμέσων.



Πρώτη παρουσίαση της πλατφόρμας Android έγινε στις 5 Νοεμβρίου 2007, παράλληλα με την ανακοίνωση της ίδρυσης του οργανισμού Open Handset Alliance, μιας κοινοπραξίας 48 τηλεπικοινωνιακών εταιριών, εταιριών λογισμικού καθώς και κατασκευής hardware, οι οποίες είναι αφιερωμένες στην ανάπτυξη και εξέλιξη προτύπων στις συσκευές κινητής τηλεφωνίας.

Κωδικό όνομα	Νούμερο έκδοσης	Ημερομηνία αρχικής κυκλοφορίας	Επίπεδο API
N/A	1.0	23 Σεπτεμβρίου 2008	1

	1.1	9 Φεβρουάριου 2009	2
Cupcake	1.5	27 Απριλίου 2009	3
Donut	1.6	15 Σεπτεμβρίου 2009	4
Eclair	2.0 – 2.1	26 Οκτωβρίου 2009	5–7
Froyo	2.2 – 2.2.3	20 Μαΐου 2010	8
Gingerbread	2.3 – 2.3.7	6 Δεκεμβρίου 2010	9–10
Honeycomb	3.0 – 3.2.6	22 Φεβρουάριου 2011	11–13
Ice Cream Sandwich	4.0 – 4.0.4	18 Οκτωβρίου 2011	14–15
Jelly Bean	4.1 – 4.3.1	9 Ιουλίου 2012	16–18
KitKat	4.4 – 4.4.4	31 Οκτωβρίου 2013	19–20
Lollipop	5.0 – 5.1.1	12 Νοεμβρίου 2014	21–22
Marshmallow	6.0 – 6.0.1	5 Οκτωβρίου 2015	23
Nougat	7.0	22 Αυγούστου 2016	24

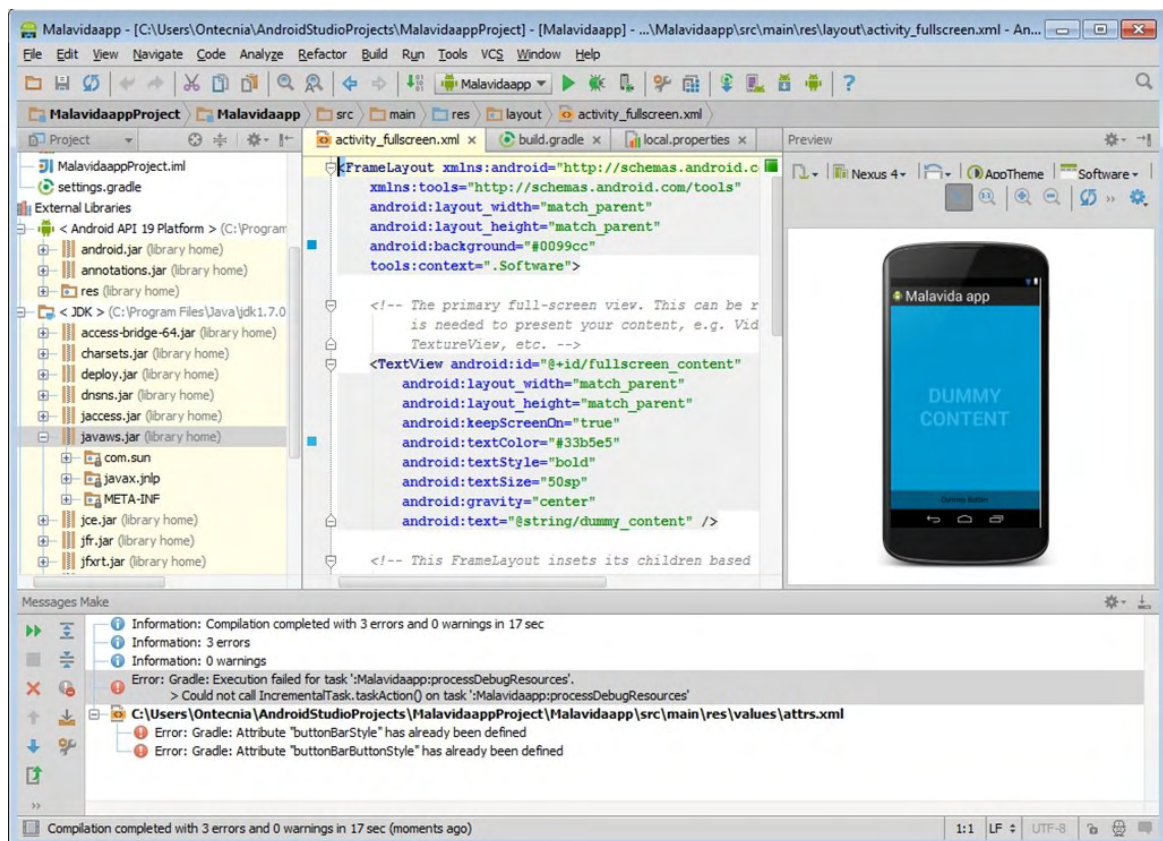
Πίνακας 1: Πίνακας εκδόσεων λογισμικού Android [1].

Η τελευταία έκδοση που έχει αναπτυχθεί έχει όνομα Nougat και η έκδοση του λογισμικού βγήκε σε κυκλοφορία στις 22 Αυγούστου 2016.

2.4 Android Studio

Το Android Studio αποτελεί ολοκληρωμένο προγραμματιστικό περιβάλλον (IDE) και χρησιμοποιείται κυρίως για την ανάπτυξη εφαρμογών της πλατφόρμας Android. Η έναρξη λειτουργίας του ανακοινώθηκε στις 16 Μαΐου 2013 στο συνέδριο Google I/O από την Google Product Manager, Katherine Chou. Το Android Studio είναι διαθέσιμο ελεύθερα με την άδεια Apache License 2.0.

Το πρόγραμμα αυτό βασίζεται στο λογισμικό της JetBrains' IntelliJ IDEA, και σχεδιάστηκε αποκλειστικά για προγραμματισμό εφαρμογών Android [1]. Πλέον, το πρόγραμμα αυτό είναι διαθέσιμο για λειτουργικά συστήματα Windows, Mac OS X και Linux. Εναλλακτικό πρόγραμμα για την ανάπτυξη εφαρμογών σε περιβάλλον Android αποτελεί και το εργαλείο Eclipse.



Εικόνα 1: Απεικόνιση πλατφόρμας Android Studio.

Κεφάλαιο 3

Θεώρημα του Βίνσεντ και δημιουργία API

Προκειμένου να γίνει η εισαγωγή του θεωρήματος του Βίνσεντ, που περιγράφεται στο παρόν κεφάλαιο, στην εφαρμογή Android που έχουμε ετοιμάσει, έπρεπε να γίνει διασύνδεση του εργαλείου Mathematica με το λογισμικό Android. Η διασύνδεση αυτή επιτεύχθηκε με την αποστολή, την επεξεργασία και τη λήψη των αποτελεσμάτων μέσω του cloud και τη δημιουργία API. Στο κεφάλαιο αυτό γίνεται παρουσίαση του εργαλείου Mathematica και της λειτουργίας του καθώς και του τρόπου διασύνδεσης μεταξύ των δύο εφαρμογών.

3.1 Θεώρημα του Βίνσεντ

Το θεώρημα του Βίνσεντ είναι ένα θεώρημα το οποίο έχει ως στόχο να απομονώσει τις πραγματικές ρίζες πολυωνύμων με ρητούς συντελεστές. Το θεώρημα πήρε το όνομά του από τον Αλεξάντρ Ζοζέφ Ιντούλφ Βενσάν και δυστυχώς δεν υπάρχει σε αρκετά σημεία της βιβλιογραφίας. Το θεώρημα αυτό χρησιμοποιούνταν κατά κόρο στην αρχή, ωστόσο επισκιάστηκε από το θεώρημα του Στουρμ (Strum) [2].

Η απομόνωση των πραγματικών ριζών ενός πολυωνύμου είναι η διαδικασία της εύρεσης ανοιχτών ξένων (disjoint) διαστημάτων τέτοιων ώστε κάθε ένα από αυτά να περιέχει ακριβώς μία πραγματική ρίζα η οποία να περιέχεται σε κάποιο διάστημα. Σύμφωνα με τη γαλλική σχολή μαθηματικών του 19ου αιώνα, αυτό είναι το πρώτο

βήμα για την εύρεση των πραγματικών ριζών, ενώ το δεύτερο βήμα είναι η προσέγγισή τους σε οποιοδήποτε βαθμό ακρίβειας. Το θεώρημα συγκεντρώνει μόνο τις θετικές ρίζες, καθώς η απομόνωση των αρνητικών ριζών ενός πολυωνύμου $p(x)$ πραγματοποιείται με την αντικατάσταση του x με $-x$ ($x \leftarrow -x$) και την επανάληψη της διαδικασίας [2].

Η μέθοδος η οποία θα χρησιμοποιηθεί στην εργασία αυτή είναι η δεύτερη κατά σειρά, αναπτύχθηκε από τους Vincent-Akritas-Strzeboński (VAS) το 2005 και έχει ως στόχο να αντιμετωπίσει την εκθετική συμπεριφορά της μεθόδου του Βίνσεντ. Η μέθοδος αυτή αποτελεί μέθοδο συνεχών κλασμάτων και είναι μία απ' ευθείας υλοποίηση του θεωρήματος του Βίνσεντ.

Η μέθοδος του Βίνσεντ μετατράπηκε σε μέθοδο πολυωνυμικής πολυπλοκότητας από τον Α. Ακρίτα το 1978, ο οποίος υπολόγισε κάθε μερικό πηλίκο αι ως ένα κάτω όριο, lb , στις τιμές των θετικών ριζών ενός πολυωνύμου. Η διαδικασία αυτή υπολογίζει το ιδανικό θετικό κάτω όριο που υπολογίζει το ακέραιο μέρος της μικρότερης θετικής ρίζας. Ουσιαστικά, τώρα το $ai \leftarrow lb$ ή, ισοδύναμα, εφαρμόζουμε την αντικατάσταση $x \leftarrow x + lb$, για την οποία απαιτείται περίπου ο ίδιος χρόνος με την αντικατάσταση $x \leftarrow x + 1$. Ο αλγόριθμος VAS αποτελεί τον προεπιλεγμένο αλγόριθμο απομόνωσης ριζών σε πολλά εργαλεία όπως τα Mathematica, Sage, SymPy και Xcas.

Εξήγηση του αλγόριθμου

Παρακάτω παρουσιάζεται ο αλγόριθμος $VAS(p, M)$, στον οποίο δεν περιλαμβάνεται η συνεισφορά του Σεμπόνσκι (Strzeboński).

1. Έστω ότι $p(x)$ είναι ένα πολυώνυμο βαθμού $\deg(p)$ τέτοιο ώστε $p(0) \neq 0$. Για την απομόνωση των θετικών ριζών του, αντιστοίχισε το πολυώνυμο $p(x)$ με τον μετασχηματισμό Μόμπιους (Möbius) $M(x) = x$ και επανέλαβε τα ακόλουθα βήματα όσο υπάρχουν ζεύγη $\{p(x), M(x)\}$ προς επεξεργασία.
2. Για τον υπολογισμό του αριθμού των ριζών που πολυωνύμου μέσα στο διάστημα $(0, \infty)$, χρησιμοποιήθηκε ο κανόνας προσήμων του Ντεκάρτ. Αν

υπάρχουν δύο ή περισσότερες εναλλαγές προσήμου από τον κανόνα προσήμων του Ντεκάρτ συνεπάγεται ότι ίσως υπάρχουν 0, 1, ή περισσότερες πραγματικές ρίζες μέσα στο διάστημα $(0, \infty)$.

3. Για να υπάρχουν με βεβαιότητα ρίζες στο διάστημα $(0, 1)$, χρησιμοποίησε το ιδανικό κάτω όριο "lb". Ουσιαστικά αυτό είναι το ακέραιο μέρος της μικρότερης θετικής ρίζας και υπολογίζεται με τη βοήθεια του κάτω ορίου.
4. Για τον υπολογισμό των ριζών μέσα στο διάστημα $(0, 1)$, εκτέλεσε στο $p(x)$ και $M(x)$ την αντικατάσταση $x \leftarrow \frac{1}{1+x}$ και επεξεργάσου το ζευγάρι.

Ακολουθεί η παρουσίαση της εφαρμογής της μεθόδου VAS στο πολυώνυμο $p(x) = x^3 - 7x + 7$ (σημειώστε ότι $M(x) = x$).

VAS($x^3 - 7x + 7, x$)

var $\leftarrow 2$ // ο αριθμός των εναλλαγών προσήμου στην ακολουθία των συντελεστών του πολυωνύμου $p(x) = x^3 - 7x + 7$

lb $\leftarrow 1$ // το ιδανικό κάτω όριο — βρίσκεται χρησιμοποιώντας lbcomputed και αντικαταστάσεις $x \leftarrow x + 1$

$p \leftarrow x^3 + 3x^2 - 4x + 1$

$M \leftarrow x + 1$

$p_{01} \leftarrow x^3 - x^2 - 2x + 1$

$M_{01} \leftarrow \frac{x+2}{x+1}$

$m \leftarrow 1$

$p_{1\infty} \leftarrow x^3 + 6x^2 + 5x + 1$

$M_{1\infty} \leftarrow x + 2$

RETURN VAS($x^3 - x^2 - 2x + 1, \frac{x+2}{x+1}$) U VAS($x^3 + 6x^2 + 5x + 1, x + 2$)

3.2 Wolfram Mathematica

Το εργαλείο Wolfram Mathematica αποτελεί ένα πρόγραμμα μαθηματικού υπολογισμού που χρησιμοποιείται από πολλές επιστήμες, όπως της μηχανικής, των μαθηματικών και της πληροφορικής. Σχεδιάστηκε από τον Stephen Wolfram και χρησιμοποιεί τη γλώσσα Wolfram, η οποία αποτελεί τη γλώσσα προγραμματισμού που χρησιμοποιείται στο Mathematica.

Το εργαλείο αυτό μπορεί να εκτελέσει κάποιες από τις ακόλουθες διεργασίες:

1. Βασικές και ειδικές μαθηματικές πράξεις με τις κατάλληλες βιβλιοθήκες.
2. Υποστήριξη υπολογισμών για σύνθετους αριθμούς και συναρτήσεις, παραγωγή αποτελεσμάτων υψηλής ακρίβειας, εύρεση διαστήματος για μαθηματικούς υπολογισμούς.
3. Εργαλεία παραγωγής πινάκων, 2D και 3D συστοιχιών για το χειρισμό των δεδομένων συμπεριλαμβανομένης της υποστήριξης για αραιές συστοιχίες, και εργαλεία animation.
4. Παροχή περισσότερων συναρτήσεων με νέες μαθηματικές μεθόδους οι οποίες είναι δύσκολο να αναπτυχθούν.

Ο χρήστης μπορεί να επιλέξει την επιθυμητή συνάρτηση που θέλει, να την εισάγει και με το πάτημα ενός κουμπιού να παραχθεί το κατάλληλο αποτέλεσμα. Η κάθε συνάρτηση έχει διαφορετικά ορίσματα, γι αυτό θα πρέπει ο χρήστης να διαβάσει σωστά τα στοιχεία ώστε να μην παραχθεί λανθασμένο αποτέλεσμα ή κάποιο σφάλμα. Όσον αφορά την απομόνωση των ριζών ενός πολυωνύμου, στη Mathematica, χρησιμοποιήθηκε η εντολή “*RootIntervals*”. Η σύνταξη της εντολής φαίνεται παρακάτω:

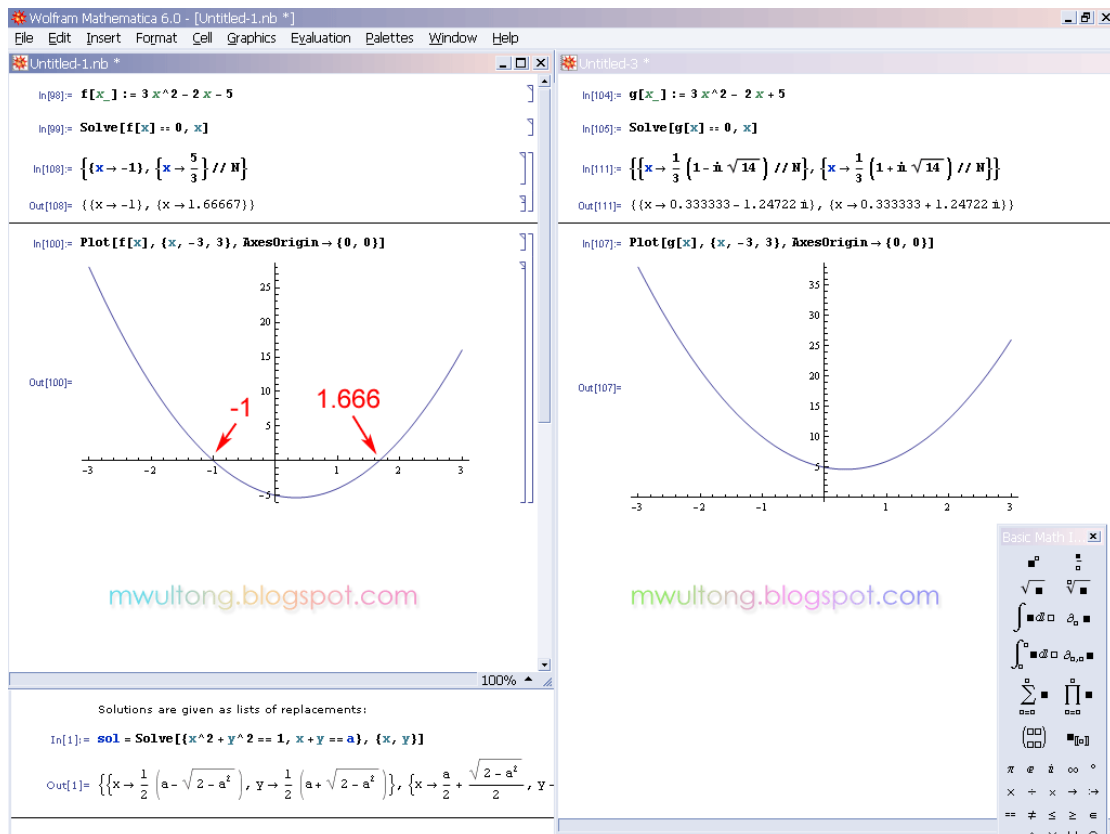
RootIntervals[{p(x)}, Complexes]

όπου $p(x)$ είναι το πολυώνυμο του οποίου τα διαστήματα ριζών θέλουμε να βρούμε. Το “Complexes” χρησιμοποιείται για να λάβουμε μεγαλύτερη ακρίβεια στην παραγωγή του αποτελέσματος.

Ακόμη μια εντολή που χρησιμοποιείται για την παραγωγή αποτελέσματος και υπολογισμού του αποτελέσματος μιας εξίσωσης, είναι η “*Solve*”. Η σύνταξη της εντολής αυτής είναι η εξής:

$$\text{Solve}[x^2 + a x + 1 == 0, x]$$

όπου το πρώτο όρισμα είναι η συνάρτηση της οποίας το αποτέλεσμα θέλουμε να βρούμε και ως δεύτερο όρισμα εισάγεται ο άγνωστος αριθμός της συνάρτησης αυτής.



Εικόνα 2: Περιβάλλον εργασίας εργαλείου Wolfram Mathematica.

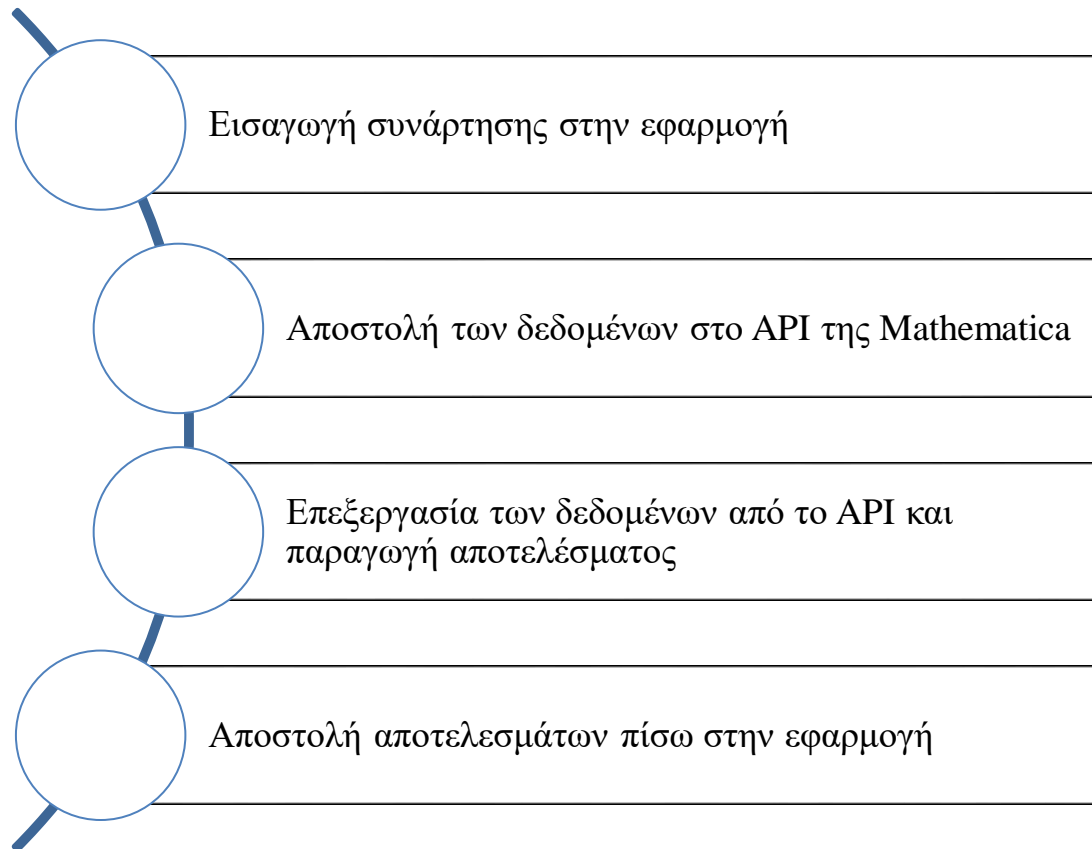
3.3 Διεπαφή προγραμματισμού Εφαρμογών (API)

Η Διεπαφή Προγραμματισμού Εφαρμογών (αγγλ. API, από το Application Programming Interface), είναι η διεπαφή των προγραμματιστικών διαδικασιών που παρέχει ένα λειτουργικό σύστημα, βιβλιοθήκη ή εφαρμογή με σκοπό να επιτρέπει να γίνονται προς αυτά αιτήσεις από άλλα προγράμματα ή/και ανταλλαγή δεδομένων.

Ένας από τους βασικούς σκοπούς μίας διεπαφής είναι να ορίζει το σύνολο των λειτουργιών που μπορεί να παρέχει μια βιβλιοθήκη ή ένα λειτουργικό σύστημα σε άλλα προγράμματα, χωρίς όμως να επιτρέπει πρόσβαση στον κώδικα που υλοποιεί αυτές τις υπηρεσίες. Η διεπαφή αποτελεί ένα «συμβόλαιο κλήσης» μεταξύ καλούντος και καλούμενου προγράμματος – σύνδεσης, ενώ παράλληλα διαχωρίζει την προγραμματιστική υλοποίηση κάποιων υπηρεσιών από τη χρήση τους [5].

Κάθε σύστημα έχει τη δική του διεπαφή, στην οποία καθορίζονται οι ικανότητες και οι δυνατότητες αυτού. Καθορίζονται ακόμη τα επιτρεπτά όρια μέσα στα οποία μπορεί να κινηθεί μια εφαρμογή, το αν θα αποσταλούν ή όχι πληροφορίες ή ακόμη και τον τρόπο επεξεργασίας αυτών από τον παραλήπτη (την εφαρμογή που λαμβάνει το αίτημα για επεξεργασία πληροφοριών).

Η δημιουργία του API είναι απαραίτητη για τη διασύνδεση δύο προγραμμάτων σε περίπτωση που δεν υπάρχει άλλος πιθανός τρόπος διασύνδεσής τους. Στην παρούσα εφαρμογή κρίθηκε απαραίτητη η δημιουργία ενός API στο περιβάλλον του εργαλείου Wolfram Mathematica, το οποίο θα λάμβανε πληροφορίες από την εφαρμογή και θα τις επεξεργαζόταν. Το αποτέλεσμα της επεξεργασίας θα επιστρέψει ξανά στην εφαρμογή ώστε να μπορέσει να γίνει ορατό στο χρήστη. Το API ήταν απαραίτητο για να παραχθεί το σωστό αποτέλεσμα όσον αφορά την παραγωγή αποτελέσματος σύμφωνα με το θεώρημα Βίνσεντ. Η παραπάνω διαδικασία γίνεται κατανοητή μέσα από το ακόλουθο διάγραμμα.



Εικόνα 3: Σχεδιάγραμμα αποστολής / λήψης αποτελέσματος μέσω του API της Mathematica.

Προκειμένου να ολοκληρωθεί η διαδικασία αυτή πρέπει να δημιουργηθεί το αντίστοιχο API στην Mathematica. Για να δημιουργηθεί κάτι τέτοιο θα πρέπει να υπάρχει το εργαλείο της Mathematica εγκατεστημένο και να εισαχθεί από το χρήστη ο «χάρτης» των εντολών τις οποίες θέλει εκείνος να εκτελέσει σε κάθε API. Παραδείγματος χάρη, εάν σε ένα API θέλουμε να παράγεται μόνο το αποτέλεσμα μιας συνάρτησης, τότε πληκτρολογούμε την κατάλληλη εντολή, όπως φαίνεται παρακάτω:

```
αρι6=APIFunction[{"x"->"Expression"},Solve[{{#x==0}, x]&,"String"];  
CloudDeploy[αρι6, Permissions->"Public"]
```

Το αποτέλεσμα που παράγεται από την εκτέλεση της εντολής αυτής είναι ο κωδικός που αντιστοιχεί στο εν λόγω API. Ο κωδικός αυτός χρησιμοποιείται για τη διασύνδεση του API με την εφαρμογή Android από την οποία θα χρησιμοποιηθεί.

Εφαρμογή επίλυσης πολυωνυμικών εξισώσεων σε πλατφόρμα android

Πρέπει να τονισθεί στο σημείο αυτό ότι κάθε API μπορεί να εκτελεί μόνο μια εντολή παραγωγής αποτελέσματος. Σε περίπτωση που πρέπει να εκτελεστούν περισσότερες από μια εντολές, τότε θα πρέπει να δημιουργηθούν πολλαπλά API.

Κεφάλαιο 4

Σχεδιασμός και ανάπτυξη εφαρμογής

Στο παρόν κεφάλαιο γίνεται ανάλυση των απαιτήσεων για τη σχεδίαση και την ανάπτυξη της εφαρμογής. Καθορίζονται οι απαιτήσεις του συστήματος, οι ρόλοι αυτού, παρατίθενται κάποια απαραίτητα χαρακτηριστικά και αρχικές εικόνες – σχεδιαγράμματα απαραίτητα για την κατανόηση της δομής της εφαρμογής.

4.1 Απαιτήσεις συστήματος

Στις απαιτήσεις της εφαρμογής συγκαταλέγονται όλα εκείνα τα στοιχεία τα οποία πρέπει να ληφθούν υπόψη από τους σχεδιαστές της εφαρμογής, ώστε να προκύψει ένα ολοκληρωμένο και ορθό αποτέλεσμα.

- **Το περιβάλλον χρήσης της εφαρμογής θα πρέπει να είναι καθαρό, σαφές και να μην περιέχει περιττά στοιχεία.** Το περιβάλλον της εφαρμογής θα πρέπει να είναι λειτουργικό και κατανοητό στο χρήστη και να μην έχει περιττά στοιχεία τα οποία επιβαρύνουν την εφαρμογή.

- **Δυνατότητα εξατομίκευσης της εφαρμογής.** Με την εισαγωγή των ασκήσεων εξάσκησης, μπορεί ο κάθε χρήστης να έχει την εφαρμογή που επιθυμεί ανάλογα με τις γνώσεις και την αναζήτηση που εκείνος επιθυμεί να κάνει.
- **Εισαγωγή πολλαπλών κατηγοριών και θεωρημάτων για την παροχή αυξημένων υπηρεσιών στους χρήστες.** Θα πρέπει να γίνει εισαγωγή κάποιων κατηγοριών οι οποίες να παρουσιάζουν ενδιαφέρον σε αρκετούς χρήστες, να είναι απαραίτητες και να αναζητούνται από την πλειονότητα αυτών.
- **Παροχή κάποιου θεωρήματος ώστε να υπάρξει ένα καινοτόμο στοιχείο στην εφαρμογή.** Στόχος της εργασίας δεν είναι η απλή παρουσίαση του αποτελέσματος και η επίλυση μιας συνάρτησης, αλλά η εμφάνιση κι ενός ακόμη αποτελέσματος το οποίο θα συνδέεται με κάποιο θεώρημα.
- **Η εφαρμογή θα πρέπει να είναι φορητή.** Θα πρέπει ο χρήστης να μπορεί να χρησιμοποιήσει την εφαρμογή τόσο σε κινητό τηλέφωνο, όσο και σε συσκευή tablet.
- **Λειτουργία της εφαρμογής ακόμη και σε περιοχές με μειωμένη διαδικτυακή πρόσβαση.** Η εφαρμογή θα πρέπει να μπορεί να λειτουργεί και σε μέρη στα οποία δεν θα υπάρχει κάλυψη δικτύου, για να μπορεί να χρησιμοποιείται από οποιονδήποτε χρήστη σε οποιοδήποτε μέρος κι αν βρίσκεται.

4.2 Αρχιτεκτονική εφαρμογής

Για να πληροί τις παραπάνω απαιτήσεις η εφαρμογή την οποία θα αναπτύξουμε, θα πρέπει να λάβουμε αρχικά ως γνώμονα το περιβάλλον στο οποίο αυτή θα λειτουργεί. Θα πρέπει η εφαρμογή να είναι φορητή, να μπορεί δηλαδή να χρησιμοποιηθεί σε οποιαδήποτε κινητή συσκευή. Κάτι τέτοιο είναι εφικτό με τη δημιουργία εφαρμογής για το λογισμικό Android. Το λογισμικό αυτό πλέον υπάρχει σε μια πληθώρα συσκευών, των οποίων ο αριθμός αυξάνεται με το πέρασμα του χρόνου.

Ακόμη μια απαίτηση την οποία θα πρέπει να λάβουμε σοβαρά υπόψη μας σχετίζεται με την εξατομίκευση της εφαρμογής. Θα πρέπει να ορίσουμε παραμέτρους οι οποίες να μπορούν να διαφοροποιηθούν σε κάθε χρήστη και να μπορούν να διαχειρίζονται εύκολα από κάποιον άλλο σε περίπτωση που οριστεί κάποιος ως διαχειριστής της εφαρμογής. Για το λόγο αυτό επιλέξαμε να υπάρχει μια καρτέλα με ασκήσεις εξάσκησης, η οποία θα μπορεί να είναι είτε κοινή ή διαφορετική για κάθε χρήστη. Επιπλέον, ο κώδικας που θα αναπτυχθεί θα πρέπει να είναι ευανάγνωστος και κατανοητός ώστε να υπάρχει η δυνατότητα εισόδου διαχειριστή της εφαρμογής και επέκτασης αυτής μελλοντικά.

Οι κατηγορίες οι οποίες θα εισαχθούν στην εφαρμογή θα πρέπει να είναι χρηστικές να και διερευνηθούν. Οι κατηγορίες θα πρέπει να είναι γενικές, και να χρησιμοποιούνται από αρκετούς χρήστες, όχι μόνο από μια μικρή κατηγορία αυτών. Θα πρέπει να γίνει η σχετική έρευνα σχετικά με τα θέματα στα οποία αναζητούν οι περισσότεροι χρήστες λύση σε παρόμοιες εφαρμογές, καθώς και σε ποια θέματα υστερεί το μεγαλύτερο ποσοστό των μαθητών.

4.3 Στοιχεία Extensible Markup Language (XML)

Στην παράγραφο αυτή θα αναλυθούν τα πιο σημαντικά κομμάτια κώδικα που χρησιμοποιήθηκαν για τη δημιουργία της εφαρμογής.

Η XML (αγγλ. αρκτ. από το Extensible Markup Language) αποτελεί μία γλώσσα σήμανσης, η οποία περιέχει ένα σύνολο κανόνων για την ηλεκτρονική κωδικοποίηση κειμένων. Ορίζεται, κυρίως, στην προδιαγραφή XML 1.0 (XML 1.0 Specification), που δημιούργησε ο διεθνής οργανισμός προτύπων W3C (World Wide Web Consortium), αλλά και σε διάφορες άλλες σχετικές προδιαγραφές ανοιχτών προτύπων [4].

Η XML σχεδιάστηκε δίνοντας έμφαση στην απλότητα, τη γενικότητα και τη χρησιμότητα στο Διαδίκτυο. Είναι μία μορφοποίηση δεδομένων κειμένου, με ισχυρή υποστήριξη Unicode για όλες τις γλώσσες του κόσμου. Αν και η σχεδίαση της XML

εστιάζει στα κείμενα, χρησιμοποιείται ευρέως για την αναπαράσταση αυθαίρετων δομών δεδομένων, που προκύπτουν για παράδειγμα στις υπηρεσίες ιστού.

Υπάρχει μία ποικιλία διεπαφών προγραμματισμού εφαρμογών, που μπορούν να χρησιμοποιούν οι προγραμματιστές, για να προσπελούν δεδομένα XML, αλλά και διάφορα συστήματα σχημάτων XML, τα οποία είναι σχεδιασμένα για να βοηθούν στον ορισμό γλωσσών, που προκύπτουν από την XML.

Έως το 2009, έχουν αναπτυχθεί εκατοντάδες γλώσσες που βασίζονται στην XML, συμπεριλαμβανομένων του RSS, του SOAP και της XHTML. Προεπιλεγμένες κωδικοποιήσεις βασισμένες στην XML, υπάρχουν για τις περισσότερες σουίτες εφαρμογών γραφείου, συμπεριλαμβανομένων του Microsoft Office (Office Open XML), του OpenOffice.org (OpenDocument) και του iWork της εταιρίας Apple.

4.3.1 Εμφάνιση (Layout)

Το βασικό δομικό κομμάτι για την διεπαφή χρήστη στις εφαρμογές Android είναι το αντικείμενο View. Το αντικείμενο **View** είναι το η βασική κλάση για τα widgets που χρησιμοποιούνται για την δημιουργία της διαδραστικής διεπαφής χρήστη, όπως τα buttons τα text fields κ.α.

Το **ViewGroup** είναι μία υποκλάση του View και παρέχει 'αόρατα' δοχεία που μέσα σε αυτά μπαίνουν άλλα Views και ViewGroups.

Το τρίτο επίπεδο της διεπαφής είναι τα **layouts** που είναι υποκλάση της κλάσης ViewGroup και καθορίζει το πως θα φαίνεται η διεπαφή χρήστη. Υπάρχουν πολλά layout που μπορούν να χρησιμοποιηθούν για την δημιουργία της διεπαφής χρήστη, τα οποία αναλύονται παρακάτω:

- **Absolute Layout:** ο χρήστης καθορίζει ακριβώς την τοποθεσία του κάθε αντικειμένου δίνοντας τις συντεταγμένες.

```
<AbsoluteLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
```

- **Frame Layout:** χρησιμοποιείται όταν ο χρήστης θέλει να δείξει μόνο ένα αντικείμενο σε κάθε οθόνη.

```
<FrameLayout
xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="fill_parent"

    android:layout_height="fill_parent">
```

- **Linear Layout:** Στο Linear Layout τοποθετείται ένα αντικείμενο σε κάθε σειρά. Κάθε αντικείμενο μπαίνει σε μία σειρά από πάνω προς τα κάτω.

```
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="fill_parent"

    android:layout_height="fill_parent"

    android:orientation="vertical" >
```

- **Relative Layout:** Σε αυτό το Layout μπορούμε να καθορίσουμε τη θέση του κάθε στοιχείου σε σχέση με τα άλλα στοιχεία ή σε σχέση με την αρχική οθόνη.

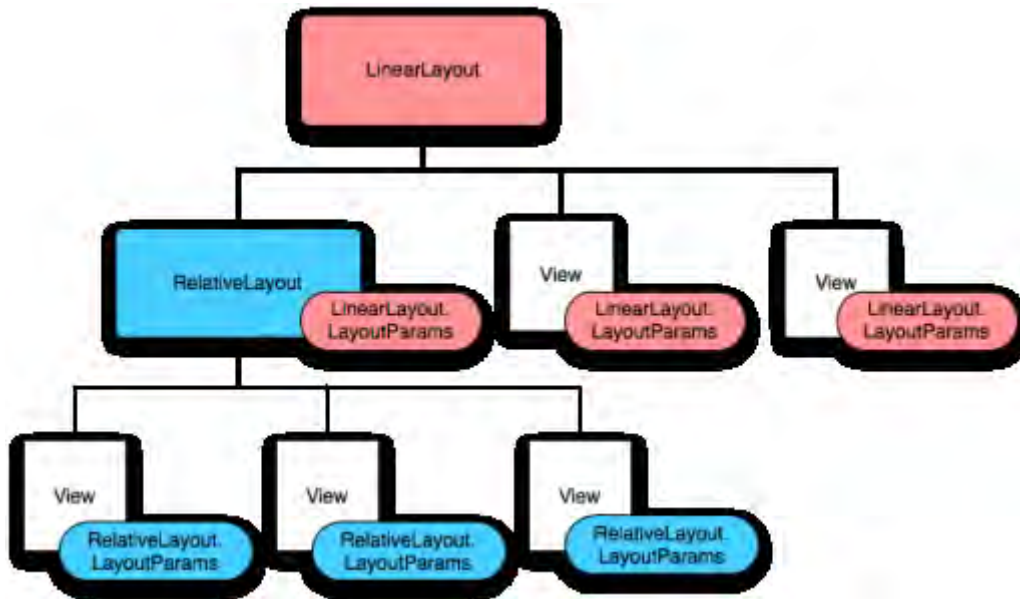
```
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"

    android:layout_width="fill_parent"

    android:layout_height="fill_parent"

    android:paddingLeft="16dp"

    android:paddingRight="16dp" >
```



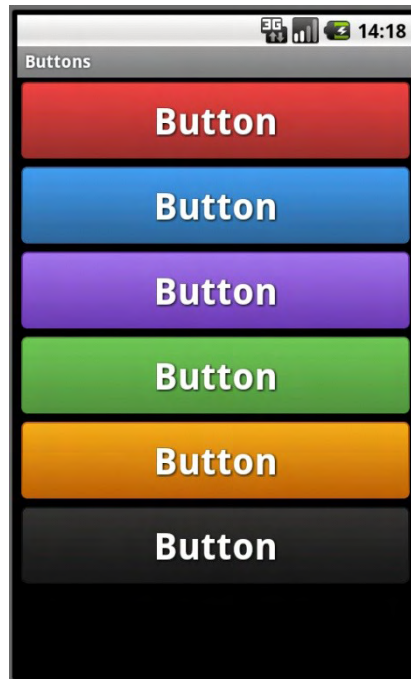
Εικόνα 4: Διαβάθμιση Layout στο Android.

4.3.2 Εισαγωγή κουμπιού (Button)

Το αντικείμενο Button είναι ένα κουμπί το οποίο ο χρήστης μπορεί να πατήσει για να ξεκινήσει μία διαδικασία. Στην συγκεκριμένη εφαρμογή δημιουργήθηκαν πολλά κουμπιά τα οποία είτε αλλάζουν οθόνη, είτε κάνουν έλεγχο στην απάντηση του χρήστη κ.α.

<Button

```
    android:layout_width="100dp"  
    android:layout_height="wrap_content"  
    android:text="OK"  
    android:layout_x="50px"  
    android:layout_y="361px" />
```



Εικόνα 5: Εικόνα κουμπιού σε μια εφαρμογή.

4.3.3 Εισαγωγή κειμένου (Text View)

Το αντικείμενο Text View χρησιμοποιείται για να δείξει κείμενο στον χρήστη και μπορεί να χρησιμοποιηθεί για να μπορεί ο χρήστης να εισάγει δικό του κείμενο.

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Tutorialspoint"  
    android:id="@+id/textView2"/>
```



Εικόνα 6: Εμφάνιση Textview σε μια εφαρμογή.

4.3.4 Ενέργειες (Activity)

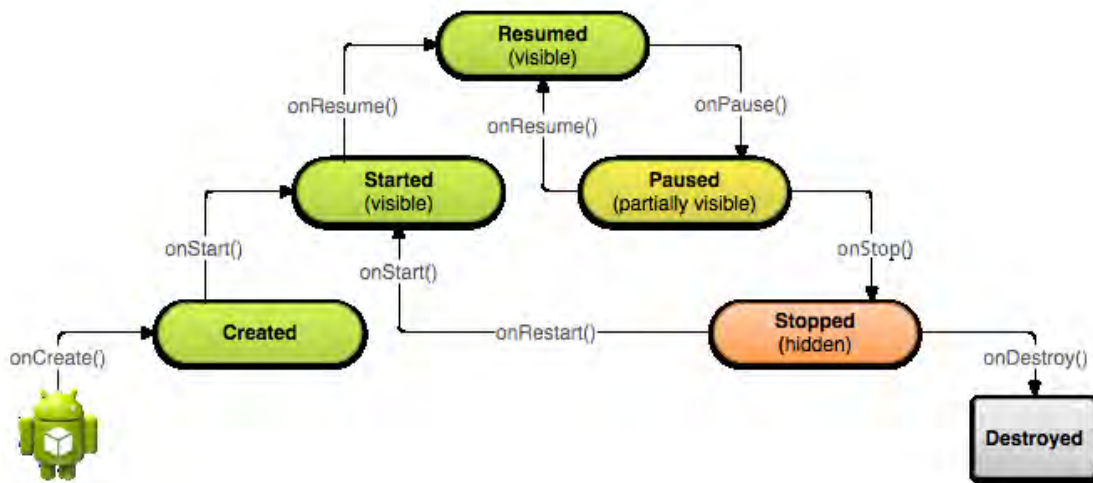
Μία διεργασία Activity είναι μία στοχευόμενη διαδικασία την οποία μπορεί να κάνει ο χρήστης. Σχεδόν όλα τα Activities αλληλεπιδρούν με τον χρήστη και καταλαμβάνει όλη την οθόνη στην οποία τοποθετείται η διεπαφή. Αν και σχεδόν πάντα ένα Activity καταλαμβάνει όλη την οθόνη, μπορούν να χρησιμοποιηθούν και σαν αναδυόμενα παράθυρα, ή ενσωματωμένα μέσα σε άλλα Activities.

Τα Activity αντιμετωπίζονται σαν μία στοίβα. Κάθε φορά που ένα καινούργιο Activity ξεκινάει τοποθετείται στην κορυφή της στοίβας και το προηγούμενο κατεβαίνει πιο κάτω στην στοίβα, όπου και θα μείνει μέχρι να κλείσει το πρώτο στην σειρά Activity.

Υπάρχουν τέσσερις δυνατές καταστάσεις για κάθε Activity

- Ενεργό, λέγεται αυτό που είναι ενεργοποιημένο στην οθόνη.

- Σε παύση, λέγεται όταν το Activity φαίνεται στην οθόνη αλλά δεν μπορεί ο χρήστης να αλληλεπιδράσει με αυτό. Για παράδειγμα όταν ο χρήστης είναι στο διαδίκτυο και δέχεται κλήση.
- Σταματημένο λέγεται όταν το Activity είναι μπλοκαρισμένο από ένα άλλο. Όλα τα στοιχεία του πρώτου Activity παραμένουν στο σύστημα αλλά δεν φαίνεται το παράθυρο πλέον στον χρήστη, και μπορεί να σβήσει από το σύστημα αν χρειαστεί η μνήμη.
- Εάν ένα Activity είναι σε παύση ή σταματημένο το σύστημα μπορεί να το σβήσει από την μνήμη είτε ζητώντας να σβήσει, είτε τερματίζοντας το μόνο του. Όταν ο χρήστης χρησιμοποιήσει αυτό το Activity ξανά τότε πρέπει να ξεκινήσει από την αρχή.



Εικόνα 7: Διάγραμμα μεθόδων στο Activity.

Υπάρχουν δύο μέθοδοι που χρησιμοποιούν όλες οι υποκλάσεις ενός Activity, η *onCreate(Bundle)* και η *onPause()*. Η μέθοδος *onCreate()* χρησιμοποιείται για την αρχικοποίηση των μεταβλητών και τη δημιουργία των γεγονότων που θέλουμε να ξεκινήσουν με την έναρξη της εφαρμογής. Αντίθετα, η δεύτερη μέθοδος καλείται όταν μια άλλη διεργασία λαμβάνει χώρα.

```
public void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
}
```

```
protected void onPause() {  
    super.onPause();  
}
```

4.3.5 AndroidManifest

Απαραίτητο στοιχείο κάθε εφαρμογής είναι το αρχείο AndroidManifest.xml (με ακριβώς αυτό το όνομα). Το αρχείο αυτό περιέχει όλες τις βασικές πληροφορίες που χρειάζεται το σύστημα για να ξεκινήσει την εφαρμογή πριν τρέξει οποιοδήποτε κομμάτι κώδικα και να αναγνωρίσει όλα τα απαραίτητα αρχεία τα οποία θα χρησιμοποιήσει κατά την εκτέλεσή της. Ακολουθεί παρουσίαση του κώδικα που χρησιμοποιείται στην εφαρμογή.

Το αρχείο Manifest είναι υπεύθυνο για:

- Την ονομασία του πακέτου της εφαρμογής. Το όνομα του πακέτου είναι ξεχωριστό και χρησιμοποιείται ως αναγνωριστικό της εφαρμογής.
- Την περιγραφή των συστατικών της εφαρμογής, όπως τα activities τα services, broadcast receivers και content providers.
- Τον καθορισμό του ελάχιστου Android API level που χρειάζεται η εφαρμογή.
- Τον καθορισμό των βιβλιοθηκών που χρειάζεται η εφαρμογή.

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
    package="user.kids_game" >
```

```
<application
```



```
android:allowBackup="true"
android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
android:theme="@style/AppTheme" >
<activity
    android:name=".MainActivity"
    android:label="@string/app_name" >
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
</application>

</manifest>
```

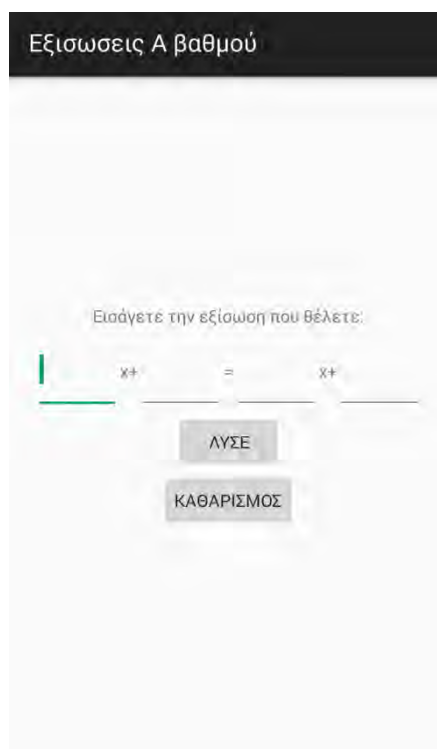
Σημειώνεται ότι στο αρχείο αυτό ο χρήστης αλλάζει όλα τα απαραίτητα στοιχεία μιας εφαρμογής τα οποία πρέπει να γνωστοποιήσει και στο λογισμικό της συσκευής στην οποία θα λειτουργεί η εφαρμογή. Παραδείγματος χάρη, ένα από τα στοιχεία αυτά είναι τα δικαιώματα που θα έχει η εφαρμογή στο λογισμικό και τι πληροφορίες απαιτείται να ανταλλάσει ώστε να ολοκληρώσει τη λειτουργία της.

4.4 Σχεδιασμός εφαρμογής

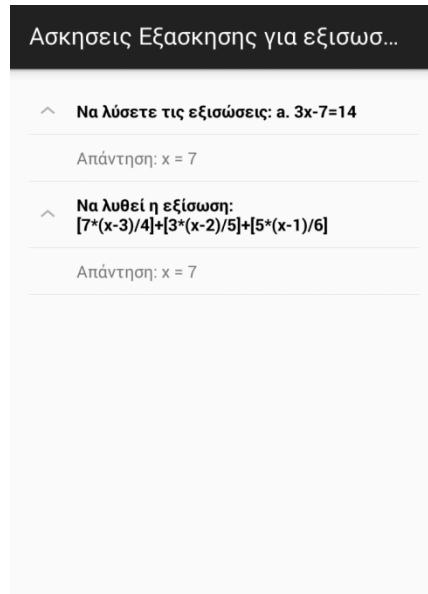
Η εφαρμογή που θα δημιουργηθεί θα είναι στα πλαίσια των απαιτήσεων του συστήματος και των ρόλων αυτού που έχουν τεθεί παραπάνω. Πιο συγκεκριμένα, θα δίνει τη δυνατότητα επιλογής της κατηγορίας του παιχνιδιού στο χρήστη, ώστε να μπορέσει εκείνος να διαλέξει το παιχνίδι με το οποίο θέλει να παίξει και το θέμα της κατηγορίας. Ακολουθούν εικόνες με τα ενδεικτικά σχέδια της εφαρμογής (mockups).



Εικόνα 8: Αρχική οθόνη εφαρμογής.



Εικόνα 9: Εξισώσεις Α' βαθμού.



Εικόνα 10: Ασκήσεις εξάσκησης τις οποίες θα μπορεί να κάνει ο χρήστης και να δει αν το αποτέλεσμα του είναι ίδιο με το αποτέλεσμα που δίνεται στην εφαρμογή.

Με το πάτημα πάνω στην άσκηση της οποίας το αποτέλεσμα θέλουμε να δούμε, κατεβαίνει η οθόνη και εμφανίζεται το αποτέλεσμα της άσκησης.

Με τη δημιουργία και το σχεδιασμό της εφαρμογής αποδείχθηκε πως δεν θα ήταν καθόλα φρόνιμο να γίνει διαχωρισμός των εξισώσεων σε πρώτου και δευτέρου ή τρίτου βαθμού. Για το λόγο αυτό, όπως θα παρουσιαστεί στο κεφάλαιο που ακολουθεί, έχει γίνει η συγχώνευση των δύο πρώτων στοιχείων σε ένα κουμπί.

Κεφάλαιο 5

Παρουσίαση εφαρμογής

Η εφαρμογή αποτελείται από διάφορες, διαφορετικές οθόνες κάθε μια από τις οποίες έχει ως στόχο να εκτελέσει και μια διαφορετική μαθηματική πράξη. Η αρχική οθόνη δίνει την επιλογή στο χρήστη να διαλέξει το σημείο στο οποίο θέλει να περιηγηθεί. Οι επιλογές είναι :

1. Επίλυση εξισώσεων
2. Ταυτότητες

Στο παρόν κεφάλαιο παρουσιάζεται η λειτουργία κάθε οθόνης χωριστά και η αλληλεπίδρασή τους με το χρήστη.

5.1 Αρχική οθόνη

Στην αρχική οθόνη μπορεί ο χρήστης να επιλέξει την κατηγορία που θέλει και να εισάγει την κατάλληλη εξίσωση ή ταυτότητα για να παράγει το επιθυμητό αποτέλεσμα. Σε κάθε κατηγορία θα μπορεί ο χρήστης πέρα από την παραγωγή αποτελέσματος να βρει και ενδεικτικές ασκήσεις τις οποίες έχει εισάγει κάποιος καθηγητής.



Εικόνα 11: Αρχική καρτέλα εφαρμογής.

5.2 Επίλυση εξισώσεων

Στην αρχική γραμμή (input) που φαίνεται στην Εικόνα 12 εισάγει ο χρήστης την εξίσωση που θέλει. Η εξίσωση μπορεί να είναι οποιουδήποτε βαθμού, χωρίς να υπάρχει ο κατάλληλος περιορισμός.

Για να λάβουμε τα διαστήματα των ριζών με βάση το θεώρημα VAS, το οποίο ζητήθηκε, γίνεται σύνδεση της Mathematica, η οποία έχει τη δυνατότητα να παράγει άμεσο αποτέλεσμα. Η διαδικασία της διασύνδεσης γίνεται μέσω του cloud, στο οποίο στέλνονται οι πληροφορίες από την εφαρμογή, γίνεται εκεί η επεξεργασία και η παραγωγή αποτελέσματος και έπειτα επιστρέφει το αποτέλεσμα πίσω στην εφαρμογή

και παρουσιάζεται στο χρήστη. Για το λόγο αυτό η εφαρμογή λειτουργεί αποκλειστικά με σύνδεση στο διαδίκτυο.

Εκτός από τα διαστήματα των ριζών, το αποτέλεσμα της εξίσωσης παρουσιάζεται στο αντίστοιχο πεδίο.



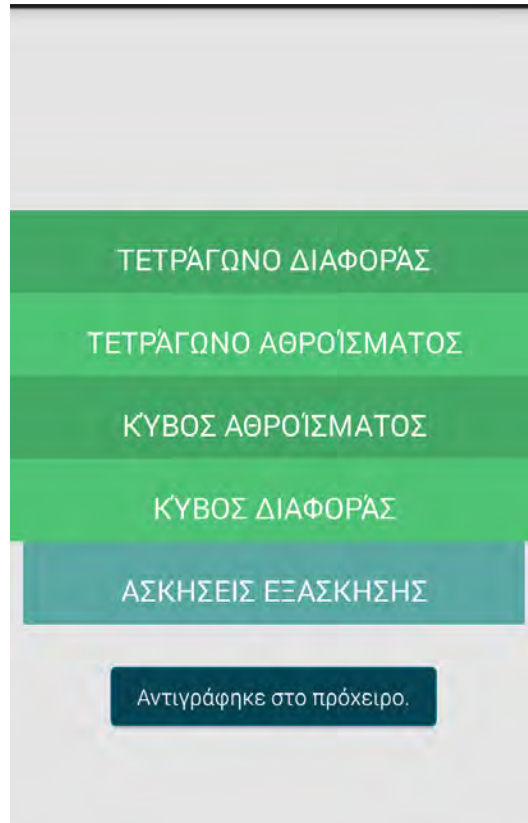
Εικόνα 12: Καρτέλα επίλυσης εξισώσεων.

5.3 Ταυτότητες

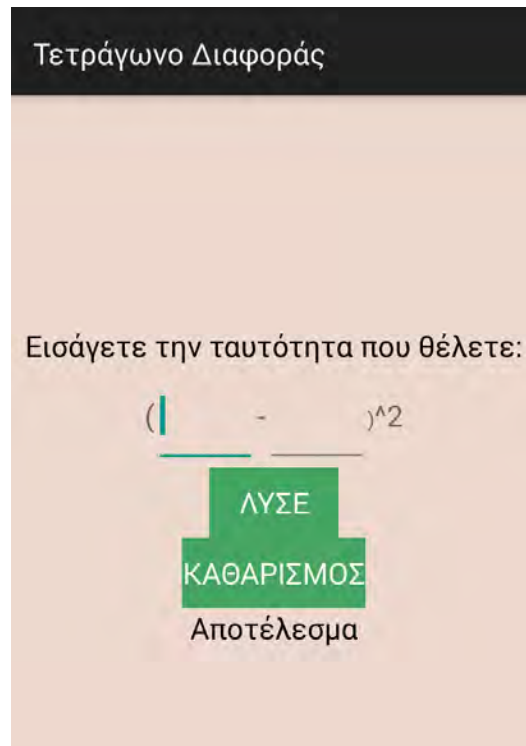
Στην κατηγορία των ταυτοτήτων ο χρήστης καλείται να επιλέξει το είδος των ταυτοτήτων των οποίων το αποτέλεσμα θέλει να βρει. Μόλις ο χρήστης επιλέξει την κατηγορία, μεταβαίνει στην αντίστοιχη καρτέλα για να εισάγει τα στοιχεία που επιθυμεί. Τα είδη ταυτοτήτων που υπάρχουν είναι τα εξής:

1. Τετράγωνο διαφοράς.
2. Τετράγωνο αθροίσματος.
3. Κύβος αθροίσματος.
4. Κύβος διαφοράς.

Σημειώνεται ότι και στην κατηγορία αυτή υπάρχει η επιλογή της προβολής ασκήσεων προς επίλυση.

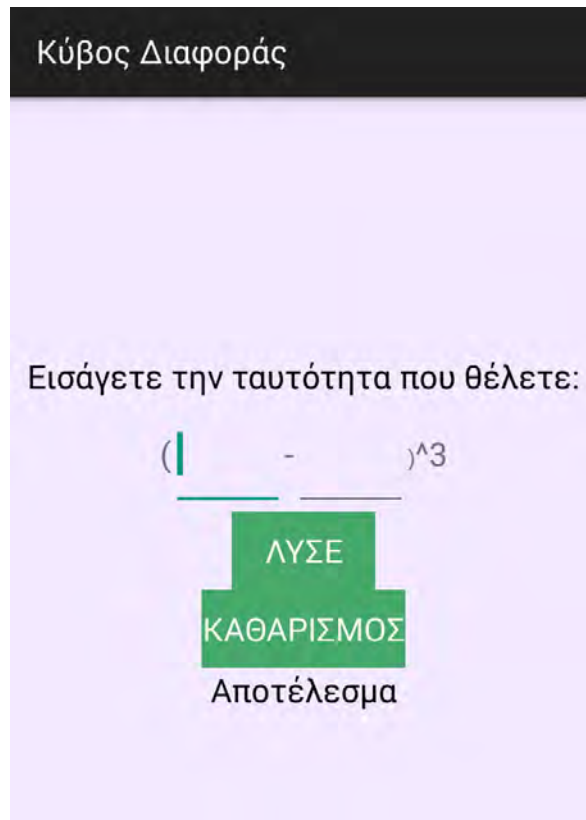


Εικόνα 13: Επιλογή είδους ταυτοτήτων στην κατηγορία "Ταυτότητες".



Εικόνα 14: Καρτέλα επίλυσης είδους "Τετράγωνο διαφοράς".

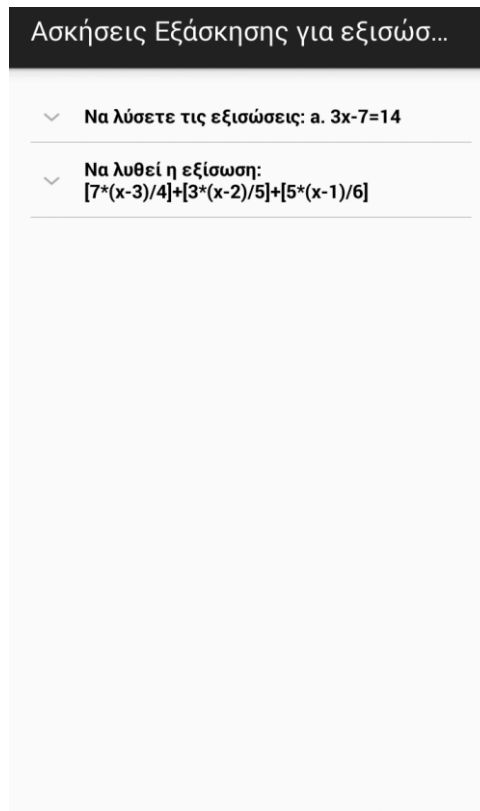
Στην καρτέλα αυτή ο χρήστης εισάγει τους αριθμούς που αντιστοιχούν στην ταυτότητα της οποίας το αποτέλεσμα θέλει να δει και έπειτα πατά το κουμπί «Λύσε». Το αποτέλεσμα που παράγεται παρουσιάζεται κάτω από την αντίστοιχη ετικέτα. Σημειώνεται ότι για την επίλυση των ταυτοτήτων δεν χρησιμοποιείται το εργαλείο Mathematica.



Εικόνα 15: Καρτέλα επίλυσης είδους "Κύβος διαφοράς".

5.4 Ασκήσεις εξάσκησης

Στην εφαρμογή αυτή δίνεται η δυνατότητα στο χρήστη να βρει τις εκφωνήσεις διαφόρων ασκήσεων, κλιμακούμενης δυσκολίας. Στόχος της δημιουργίας μιας τέτοιας καρτέλας είναι αρχικά η εκμάθηση και η περαιτέρω τριβή του μαθητή με το αντικείμενο το οποίο έχει επιλέξει, μέσα από την επίλυση ασκήσεων και στη συνέχεια η αλληλεπίδραση αυτού με τον καθηγητή. Η εφαρμογή έχει αναπτυχθεί με γνώμονα να δίνεται η δυνατότητα στον καθηγητή να μπορεί να εισάγει τις ασκήσεις που θέλει στους χρήστες ώστε να μπορούν εκείνοι να επιλύουν συγκεκριμένες – επιλεγμένες ασκήσεις.



Εικόνα 16: Καρτέλα "Ασκήσεις εξάσκησης".

Κεφάλαιο 6

Συμπεράσματα

Η τεχνολογική ανάπτυξη στις μέρες μας εισέρχεται απροειδοποίητα και με καλούς στόχους σε πολλά πεδία. Ένα από τα πεδία αυτά είναι και η εκπαίδευση. Η εκπαίδευση αποτελεί το σημαντικότερο αγαθό του ανθρώπου, καθώς δεν του παρέχει μόνο γνώσεις, αλλά βοηθά και στην ανάπτυξη του χαρακτήρα του και τη διαμόρφωση αυτού. Κρίσιμο στοιχείο της παράλληλης πορείας αλλά και αλληλεπίδρασης της τεχνολογικής ανάπτυξης με την εκπαίδευση αποτελεί η διατήρηση της ακεραιότητας και του χαρακτήρα της δεύτερης ο οποίος θα πρέπει να μείνει ανέπαφος και να υλοποιεί απρόσκοπτα το στόχο της.

Αρκετές εφαρμογές έχουν αναπτυχθεί στο παρελθόν, οι οποίες εμπεριέχουν εκπαιδευτική δραστηριότητα. Μέσα από τις εφαρμογές αυτές καλούνται οι χρήστες να παίξουν εκπαιδευτικά παιχνίδια, να μάθουν, να αναρωτηθούν και το κυριότερο, να περάσουν το χρόνο τους ευχάριστα, λαμβάνοντας τα απαραίτητα ερεθίσματα. Έχοντας ως γνώμονα την πεποίθηση αυτή, αναπτύχθηκε η παρούσα εφαρμογή. Η παρούσα εφαρμογή αφορά το πεδίο των μαθηματικών και απευθύνεται σε ένα διευρυμένο πεδίο χρηστών, όχι μόνο σε μαθητές κάποιας ηλικίας.

Στόχος της εφαρμογής είναι αρχικά να μπορέσουν οι χρήστες να λάβουν το αποτέλεσμα της εξίσωσης που επιθυμούν, άμεσα και χωρίς την ύπαρξη πιθανοτήτων σφάλματος. Ακόμη, δίνεται η δυνατότητα εξατομίκευσης της εφαρμογής μέσα από την αλληλεπίδραση του καθηγητή με τον χρήστη, ο οποίος θα μπορεί να εισάγει τις κατάλληλες ασκήσεις για τον κάθε χρήστη, στην καρτέλα «Ασκήσεις εξάσκησης». Ένα ακόμη νέο στοιχείο το οποίο υπάρχει στην παρούσα εφαρμογή αφορά την εισαγωγή

του θεωρήματος του Βίνσεντ, το αποτέλεσμα του οποίου φαίνεται σε κάθε καρτέλα επιλογής.

Το μάθημα των μαθηματικών επιλέχθηκε έχοντας ως γνώμονα τη δυσκολία που αυτά παρουσιάζουν και την πληθώρα γνώσεων και επιλογών που εμπεριέχουν. Ο χρήστης μπορεί να επιλέξει ανάμεσα στην επίλυση εξισώσεων πρώτου βαθμού, γενικότερων εξισώσεων οποιουδήποτε βαθμού και στην επίλυση ταυτοτήτων. Οι κατηγορίες αυτές επιλέχθηκαν έπειτα από ενδελεχή έρευνα η οποία αφορά τα ποσοστά αναζήτησης των χρηστών σε παρόμοιες εφαρμογές και εργαλεία. Ένα αρκετά αυξημένο ποσοστό ατόμων αναζητά επίλυση σε εξισώσεις πρώτου, δευτέρου ή μεγαλύτερου βαθμού, ενώ οι μαθητές κυρίως αναζητούν αποτελέσματα για την κατηγορία των ταυτοτήτων.

Μελλοντικά η εφαρμογή μπορεί να επεκταθεί με αρκετούς τρόπους. Αρχικά, υπάρχει η πιθανότητα εισαγωγής και άλλων κατηγοριών οι οποίες θα βοηθήσουν τους χρήστες στην εύρεση αποτελεσμάτων και την απόκτηση περαιτέρω γνώσεων. Ακόμη, υπάρχει η δυνατότητα αλληλεπίδρασης του χρήστη με κάποιο καθηγητή ή διάφορα διαδικτυακά παραδείγματα, ώστε να μπορέσει μέσα από τη λήψη διάφορων εκφωνήσεων να δοκιμάσει να επιλύσει μόνος του κάποιες ασκήσεις. Τέλος, μπορούν να προστεθούν και άλλα θεωρήματα στα οποία οι χρήστες δεν μπορούν να βρουν λύση εύκολα ή δεν τα γνωρίζουν.

Βιβλιογραφία

1. <https://el.wikipedia.org/wiki/Android>
2. <https://el.wikipedia.org/wiki/%CE%98%CE%B5%CF%8E%CF%81%CE%B7%CE%BC%CE%B1%CF%84%CE%BF%CF%85%CE%92%CE%AF%CE%BD%CF%83%CE%B5%CE%BD%CF%84#.CE.98.CE.B5.CF.8E.CF.81.CE.B7.CE.BC.CE.B1.CF.84.CE.BF.CF.85.CE.92.CE.AF.CE.BD.CF.83.CE.B5.CE.BD.CF.84:.CE.97.CF.80.CE.B1.CF.81.CE.B1.CE.BB.CE.BB.CE.B1.CE.B3.CE.AE.CF.84.CF.89.CE.BD.CF.83.CF.85.CE.BD.CE.B5.CF.87.CF.8E.CE.BD.CE.BA.CE.BB.CE.B1.CF.83.CE.BC.CE.AC.CF.84.CF.89.CE.BD.281834.CE.BA.CE.B1.CE.B9.1836.29>
3. <https://www.wolfram.com/mathematica/>
4. <https://el.wikipedia.org/wiki/XML>
5. <https://el.wikipedia.org/wiki/%CE%94%CE%B9%CE%B5%CF%80%CE%B1%CF%86%CE%AE%CF%80%CF%81%CE%BF%CE%B3%CF%81%CE%B1%CE%BC%CE%BC%CE%B1%CF%84%CE%B9%CF%83%CE%BC%CE%BF%CF%8D%CE%B5%CF%86%CE%B1%CF%81%CE%BC%CE%BF%CE%B3%CF%8E%CE%BD>
6. <https://reference.wolfram.com/language/ref/APIFunction.html>
7. <https://reference.wolfram.com/language/ref/Interpreter.html>
8. <https://reference.wolfram.com/language/guide/InterpretingStrings.html>
9. <https://reference.wolfram.com/language/ref/interpreter/Expression.html>
10. https://www.wolframcloud.com/objects/4948e4ff-5e14-498b-9011-a030990163ef?x=x^2+2*x-7

11. https://www.wolframcloud.com/objects/014ab10d-ed22-4aab-8db7-4c104abc8475?x=5*x+2
12. <http://community.wolfram.com/groups/-/m/t/317875>