



## Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Προηγμένα Συστήματα Πληροφορικής»

### Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Ανάπτυξη εφαρμογής σε λειτουργικό σύστημα Android για σημειώσεις φοιτητών. <b>Android Application for Students Notes.</b>
Όνοματεπώνυμο Φοιτητή	<b>Χαρέας Παναγιώτης</b>
Πατρώνυμο	<b>Δημήτριος</b>
Αριθμός Μητρώου	<b>ΜΠΣΠ/15093</b>
Επιβλέπων	<b>Ευθύμιος Αλέπης, Επίκουρος Καθηγητής</b>



---

Ημερομηνία Παράδοσης , 15-10-2018

---

### Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Ευθύμιος Αλέπης  
Επίκουρος Καθηγητής

(υπογραφή)

Κωνσταντίνος Πατσάκης  
Επίκουρος Καθηγητής

(υπογραφή)

Μαρία Βίρβου  
Καθηγήτρια





## ΠΕΡΙΕΧΟΜΕΝΑ

<b>1.</b>	<b>ΠΕΡΙΛΗΨΗ</b> .....	<b>4</b>
<b>2.</b>	<b>ΕΙΣΑΓΩΓΗ</b> .....	<b>6</b>
<b>3.</b>	<b>ΑΝΑΣΚΟΠΗΣΗ ΠΕΔΙΟΥ</b> .....	<b>7</b>
<b>4.</b>	<b>ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ</b> .....	<b>15</b>
<b>5.</b>	<b>ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ</b> .....	<b>32</b>
5.1	ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ .....	32
5.2	ANDROID OS .....	33
5.3	ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ .....	39
5.4	ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ .....	52
5.5	ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ SQLITE .....	52
5.6	FIREBASE ΤΕΧΝΟΛΟΓΙΕΣ .....	61
5.7	ΕΡΓΑΛΕΙΟ ΑΝΑΠΤΥΞΗΣ ΚΩΔΙΚΑ ANDROID STUDIO .....	72
5.8	ΔΟΜΗ ΕΝΟΣ ΕΡΓΟΥ ΣΕ ANDROID STUDIO .....	74
<b>6.</b>	<b>ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ</b> .....	<b>77</b>
<b>7.</b>	<b>ΒΙΒΛΙΟΓΡΑΦΙΑ</b> .....	<b>78</b>



## 1. ΠΕΡΙΛΗΨΗ

Βασικός στόχος της παρούσας διπλωματικής εργασίας είναι η ανάπτυξη μιας εφαρμογής σε λειτουργικό σύστημα Android για κινητές συσκευές η οποία εστιάζει στην διαχείριση και οργάνωση των αμέτρητων πληροφοριών που λαμβάνει ένας άνθρωπος σε καθημερινή βάση. Πιο συγκεκριμένα περιγράφεται η ανάγκη αυτή και προτείνεται μια πιθανή λύση που αφορά τον τομέα της εκπαίδευσης και ειδικότερα την φοιτητική κοινότητα καθώς προσπαθεί να δώσει λύση στην οργάνωση και διαχείριση των σημειώσεων που λαμβάνουν οι φοιτητές ενός εκπαιδευτικού ιδρύματος. Ο τομέας αυτός της εκπαίδευσης δεν επιλέχθηκε τυχαία αλλά αντιθέτως αποτελείται την πλέον αντιπροσωπευτική κοινότητα στην οποία το πρόβλημα αυτό της διαχείρισης και οργάνωσης των πληροφοριών αποτελεί καθημερινή ανάγκη των μελών αυτής. Έτσι λοιπόν η εφαρμογή που κατασκευάστηκε για τον σκοπό αυτό πρόκειται για μία εφαρμογή σημειώσεων όπου οι χρήστες της μπορούν να καταχωρήσουν αλλά και να μοιραστούν σημειώσεις σε πραγματικό χρόνο. Αυτές οι λειτουργίες προσφέρονται στο χρήστη μέσω ενός φιλικού και εύχρηστου περιβάλλοντος. Οι ενδιαφερόμενοι χρήστες καταχωρούν τόσο κείμενο όσο και εικόνες σχετικές με την σημείωση τους και με το πάτημα ενός κουμπιού πραγματοποιούν την ανάρτηση αυτής που είναι ορατή σε όλους. Επίσης δίνεται η δυνατότητα ενημέρωσης των σημειώσεων από τον συγγραφέα αυτής καθώς προσφέρονται παράλληλα και ειδοποιήσεις στους ενδιαφερόμενους χρήστες τόσο σε πιο ειδικό επίπεδο που αφορά μια συγκεκριμένη σημείωση όσο και σε πιο γενικό που αφορά ολόκληρη κατηγορία σημειώσεων. Τέλος όσον αφορά τις λειτουργικές δυνατότητες της εφαρμογής προσφέρεται επιπλέον πέραν της προβολής και η αποθήκευση μίας σημείωσης που έχει καταχωρηθεί από άλλους χρήστες. Η παρούσα διπλωματική ωστόσο δεν περιγράφει μονό τις λειτουργικές δυνατότητες αυτής της εφαρμογής αλλά εστιάζει και στην τεχνική υλοποίησή της. Παρουσιάζει αναλυτικά τις τεχνολογίες που χρησιμοποιήθηκαν για την κατασκευή της και κάνει αναφορά σε παρόμοιες εφαρμογές που αναπτύχθηκαν στην Ελλάδα και το εξωτερικό. Κλείνοντας αναφέρεται στα συμπεράσματα που προέκυψαν κατά την ανάπτυξη αυτής έτσι ώστε να προσφέρει μια πλήρη εικόνα στους αναγνώστες της σχετικά με το πρόβλημα που κλήθηκε να αντιμετωπίσει.

## Abstract

The main goal of this diploma thesis is to develop an application on Android mobile operating system that focuses on managing and organizing the innumerable information which a person receives on a daily basis. More specifically, this document describes this need and propose a possible solution which concern the education sector and particular the student community as it attempts to solve the need of management of the notes which receive the students of an educational institution. This area of education has not been chosen by chance but instead is the most representative community in which this problem of managing and organizing information is a daily need of its members. So the application which built for this purpose is about a note application where users can record and share notes in real-time. These functions are offered to the user through a friendly and easy for using environment. The Interested users can enter both text and images related to their note and with a touch of a button can post it and make it visible to everyone. Furthermore, the application gives the possibility of updating a note by her author and it also offers notifications to the interested users both at a specific level referring



---

to a particular note and at general regarding to a whole note category. Finally, concerning to the functionality of the application, it is offered beyond the view of the note, the storage of it which has registered by other users. This diploma thesis, however, does not only describe the functional capabilities of this application but also focuses on its technical implementation. It presents in detail the technologies used for its construction and makes reference to similar applications developed in Greece and abroad. In closing, it refers to the conclusions that emerged in its development so as to give a full picture to its readers about the problem that it was called upon to deal with.



## 2. ΕΙΣΑΓΩΓΗ

Την εποχή την οποία διανύουμε την χαρακτηρίζουν πολλές εξελίξεις σε διαφορετικούς τομείς. Δυο από αυτές αποτελούν και ο καταϊγισμός των πληροφοριών και η έξαρση της τεχνολογίας. Συγκεκριμένα την τελευταία δεκαετία ίσως και λίγο περισσότερο η τεχνολογία έχει εξελιχθεί σε τόσο μεγάλο βαθμό ώστε πλέον πολλές από τις καθημερινές ανάγκες ενός ανθρώπου να εξαρτώνται ως επί το πλείστον από αυτή. Αυτό το γεγονός αποδεικνύεται από την μετάβαση που έχει γίνει τα τελευταία χρόνια από τις συσκευές των υπολογιστών και των laptop σε αυτές των κινητών smartphone και των tablet και γενικότερα σε κάθε είδους συσκευή τσέπης. Ο λόγος αυτής της μετάβασης δεν είναι άλλος από το ότι πλέον ανάγκες όπως η παροχή πληροφοριών, υπηρεσιών και επικοινωνίας που κάποτε μπορούσαν να προσφερθούν μόνο μέσω της οθόνης ενός υπολογιστή πλέον παρέχονται ανά πάσα ώρα και στιγμή από την οθόνη μια κινητής συσκευής. Ωστόσο το γεγονός αυτής της έξαρσης στον τομέα της τεχνολογίας επηρέασε με την εμφάνιση του και άλλους τομείς άμεσα αλληλένδετους με αυτήν όπως αυτόν της παροχής των πληροφοριών. Αυτό συνέβη καθώς η τεχνολογία αλλά και το internet έδωσε την δυνατότητα στους χρήστες να μπορούν από οπουδήποτε και αν βρίσκονται να είναι στο επίκεντρο των εξελίξεων, να αλληλοεπιδρούν και να ενημερώνονται άμεσα για αυτές με το πάτημα ενός κουμπιού. Οι εξελίξεις λοιπόν αυτές όπως αντιλαμβανόμαστε αποτέλεσαν την αφορμή ώστε να γεννηθεί ακόμα μια ανάγκη της καθημερινότητας ενός χρήστη όπως αυτή της οργάνωσης και διαχείρισης των αμέτρητων πληροφοριών που λαμβάνει. Την ανάγκη αυτή κλήθηκε να καλύψει λοιπόν ο τομέας της τεχνολογίας και ειδικότερα αυτός της πληροφορικής μέσω συγκεκριμένων εφαρμογών που ασχολούνται με την διαχείριση και οργάνωση της πληροφορίας. Έτσι λοιπόν πλήθος εφαρμογών που σχετίζεται με διαφορετικούς κλάδους και που σκοπό τους έχουν την διαχείριση και οργάνωση της πληροφορίας εμφανίζονται κατά καιρούς προκειμένου να καλύψουν την ανάγκη αυτή που προέκυψε. Ένα τέτοιο είδος οργανωτικών εφαρμογών αποτελούν και αυτές των σημειώσεων τις οποίες ένας χρήστης χρειάζεται να κρατάει σε καθημερινή βάση είτε για προσωπική, είτε για επαγγελματική ή εκπαιδευτική χρήση. Τις εφαρμογές λοιπόν αυτές όπως ήταν και αναμενόμενο τις συναντάμε πλέον σε μεγάλο πλήθος συσκευών (εφαρμογές Desktop, Web εφαρμογές και εφαρμογές για Android και IOS) έτσι ώστε να καλυφθεί η ανάγκη από οποιαδήποτε συσκευή χρησιμοποιεί ένας χρήστης. Με αφορμή λοιπόν την ανάγκη αυτή η παρούσα διπλωματική εργασία επικεντρώνεται στην ανάλυση αυτής ενώ παράλληλα προτείνει μία λύση για την κάλυψη της μέσω μίας εφαρμογής σημειώσεων που εστιάζει στις ανάγκες διαχείρισης και οργανώσεων σημειώσεων συγκεκριμένα για φοιτητές ενός εκπαιδευτικού ιδρύματος. Ο λόγος που η συγκεκριμένη διπλωματική εστιάζει στο χώρο της εκπαίδευσης και συγκεκριμένα στους φοιτητές είναι ότι αποτελεί μία από τις πιο αντιπροσωπευτικές κοινότητες ανθρώπων όπου η ανάγκη αυτή για οργάνωση των αμέτρητων πληροφοριών που λαμβάνονται καθημερινά από τα άτομα αυτής σε συνδυασμό με την έλλειψη χρόνου που υφίσταται καθιστά απαραίτητη την ύπαρξη μίας τέτοιας εφαρμογής σημειώσεων. Η εφαρμογή λοιπόν που υλοποιήθηκε και περιγράφεται στην παρούσα πτυχιακή εργασία όπως είπαμε αφορά μια εφαρμογή σε λειτουργικό σύστημα Android και πρόκειται για μια εφαρμογή σημειώσεων στην οποία οι χρήστες – φοιτητές μπορούν να κάνουν είσοδο σε αυτή με λογαριασμό ηλεκτρονικού ταχυδρομείου, να αναρτήσουν μια σημείωση αλλά και εικόνα αυτής, να αναζητήσουν αλλά και να ανταλλάξουν σημειώσεις με όλους τους χρήστες τις εφαρμογής σε πραγματικό χρόνο από οποιαδήποτε τοποθεσία και αν βρίσκονται. Κλείνοντας προσφέρει ενημερώσεις σχετικές με σημειώσεις που ήδη έχουν στην κατοχή τους οι χρήστες της και πιθανόν άλλαξαν αλλά και

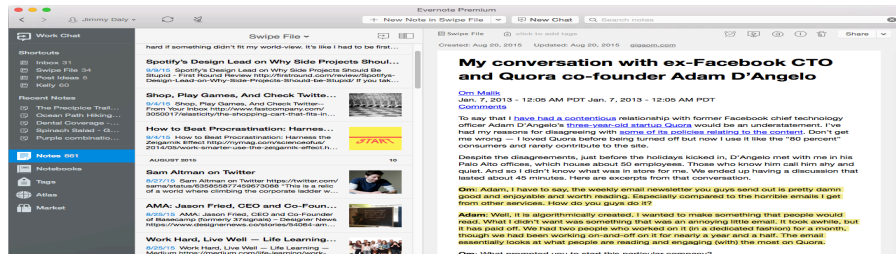


ενημερώσεις για καινούργιες σημειώσεις που αναρτήθηκαν από άλλους χρήστες της εφαρμογής και αφορούν τον ενδιαφερόμενο χρήστη.

### 3. ΑΝΑΣΚΟΠΗΣΗ ΠΕΔΙΟΥ

Στην εποχή που ζούμε την οποία χαρακτηρίζουν ο καταρτισμός των πληροφοριών και η έξαρση της τεχνολογίας γεννιέται η ανάγκη για οργάνωση και διαχείριση των αμέτρητων πληροφοριών που λαμβάνουμε καθημερινά. Η ανάγκη αυτή καλύφθηκε λοιπόν μέσω της τεχνολογίας και ειδικότερα μέσω συγκεκριμένων εφαρμογών που ασχολούνται με την διαχείριση και οργάνωση της πληροφορίας. Ένα τέτοιο είδος οργανωτικών εφαρμογών αποτελούν και αυτές των σημειώσεων οι οποίες πλέον είναι συμβατές με όλο και μεγαλύτερο πλήθος συσκευών (εφαρμογές Desktop, Web εφαρμογές και εφαρμογές για Android και IOS) προκειμένου να καλύψουν τις ανάγκες της καθημερινότητας ενός χρήστη. Με αφορμή λοιπόν την παρούσα πτυχιακή εργασία που περιγράφει μία τέτοια εφαρμογή σημειώσεων παρακάτω σας παραθέτουμε άλλες παρόμοιες εφαρμογές στην Ελλάδα και στο εξωτερικό.

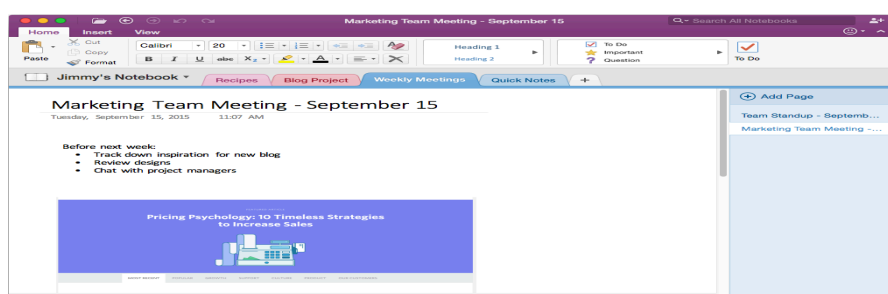
#### 1. Evernote (IOS, Android, Windows, Web)



Η εφαρμογή Evernote πρόκειται για ένα πολύ ισχυρό εργαλείο οργάνωσης και διαχείρισης σημειώσεων σε ηλεκτρονική μορφή. Μπορεί να διαχειριστεί αρχεία προσθέτοντας σε αυτά κείμενο, εικόνες ή ήχο δίνοντας την δυνατότητα να τα οργανώσει κανείς σε σημειωματάρια. Παράλληλα προσφέρει την δυνατότητα να σαρώσει κάποιος χειρόγραφα αρχεία και να μπορεί να τα αναζητήσει εύκολα μαζί με όλα τα υπόλοιπα έγγραφα όπως pdf ή αρχεία office μέσω μιας πολύ ισχυρής αναζήτησης η οποία χάρεις το ευρετήριο που κατασκευάζει και χρησιμοποιεί μπορεί να αναζητήσει κείμενο μέσα από οποιαδήποτε αρχείο ακόμα και αν αυτό είναι εικόνα. Η συγκεκριμένη εφαρμογή προσφέρετε για λειτουργικά συστήματα Android, IOS για κινητές συσκευές, για Windows για Desktop αλλά και ως Web εφαρμογή γενικότερα.



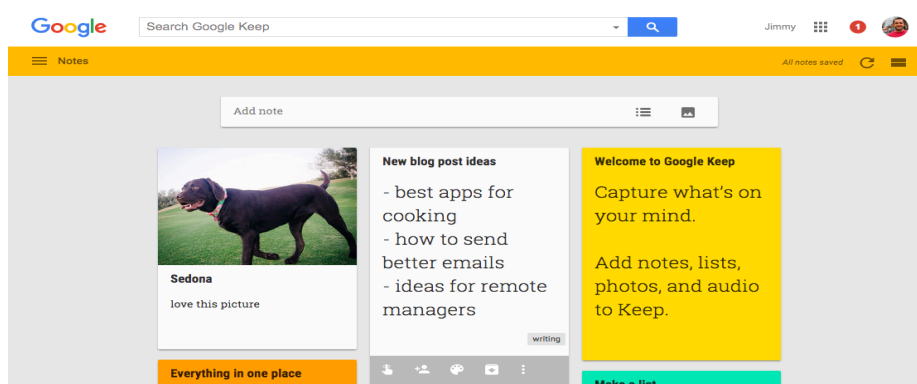
## 2. Microsoft OneNote (IOS,Android,Mac,Windows,Web)



Η εφαρμογή OneNote αποτελεί ένα από τα πολλά εργαλεία της σουίτας Microsoft Office που προσφέρατε από την εταιρεία Microsoft. Το συγκεκριμένο λοιπόν εργαλείο ασχολείται αποκλειστικά με την δυνατότητα καταχώρησης και οργάνωσης σημειώσεων και δίνει την αίσθηση στους χρήστες του ότι καταχωρούν σημειώσεις σε ένα κλασικό σημειωματάριο που προσφέρει απίστευτες δυνατότητες. Σε σύγκριση με την εφαρμογή Evernote που αναφέραμε και πιο πάνω υπάρχει μία ειδοποιός διαφορά μεταξύ τους που τις κάνει μοναδικές και ο κάθε χρήστης μπορεί να επιλέξει ανάμεσα τους ανάλογα με το ποια εξυπηρετεί τις ανάγκες του. Η πρώτη λοιπόν εφαρμογή που σας παρουσιάσαμε δηλαδή η Evernote πέρα από την βασική ανάγκη που και οι δύο αυτές εφαρμογές εξυπηρετούν δηλαδή την καταχώρηση σημειώσεων παρέχει μία πολύ ισχυρή δυνατότητα αναζήτησης για μπορούν οι χρήστες της να βρουν οποιαδήποτε σημείωση έχουν καταχώρηση εντός της εφαρμογής. Σε αντίθεση με αυτήν η εφαρμογή OneNote δεν στηρίζεται τόσο στην αναζήτηση αλλά προσφέρει μία δομή οργάνωσης που βασίζεται σε καρτέλες και κατηγορίες όπως αυτές των κλασικών σημειωμάτων έτσι ώστε οι σημειώσεις να παραμένουν οργανωμένες και να μπορούν εύκολα να αναζητηθούν από τους χρήστες της. Όσον αφορά το λειτουργικό κομμάτι αυτής η Microsoft έχει κατασκευάσει ένα εργαλείο καταχώρησης σημειώσεων με απίστευτες δυνατότητες. Πιο συγκεκριμένα το εργαλείο αυτό δίνει την αίσθηση στους χρήστες της ότι καταγράφουν σημειώσεις σε χαρτί καθώς μπορούν να καταχωρήσουν ότι θέλουν μέσα σε αυτές και σε οποιοδήποτε σημείο της σελίδας επιθυμούν. Παράλληλα παρέχει την δυνατότητα ενσωμάτωσης εικόνων, βίντεο, σκίτσων, αρχείων ήχου, συνημμένων αρχείων καθώς και λίστα υποχρεώσεων εντός της σημείωσης που πραγματοποιείτε απλά με το σύρσιμο του αρχείου ενσωμάτωσης προσφέροντας τεράστια ευελιξία χρήσης. Επιπλέον δίνει την δυνατότητα στους χρήστες της να σκιαγραφούν τα σημαντικότερα σημεία της σημείωσης όπως ακριβώς κάνει ένας χρήστης σε μία απλή σημείωση που καταγράφεται σε χαρτί. Ωστόσο οι δυνατότητες αυτής όσον αφορά την επεξεργασία ενός κειμένου και ειδικότερα στην περίπτωση μία σημείωσης δεν περιορίζεται μόνο σε όσα αναφέραμε καθώς προσφέρατε και σε αυτό το εργαλείο της Microsoft η γνωστή για τους εξισωμένους με την συγκεκριμένη σουίτα (Microsoft Office) γραμμή εργαλείων της. Τέλος η συγκεκριμένη εφαρμογή προσφέρατε για λειτουργικά συστήματα Android, IOS για κινητές συσκευές, σε Mac και Windows για εφαρμογές Desktop αλλά και ως Web εφαρμογή για αυτούς που προτιμούν να τα διαχειρίζονται όλα μέσω του διαδικτύου.



### 3. Google Keep (IOS, Android, Web)

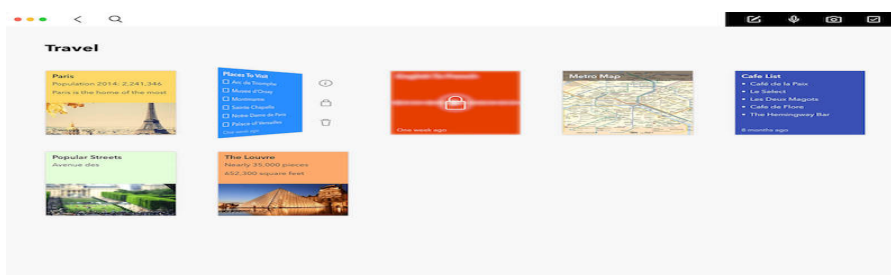


Η εφαρμογή Google Keep μπορεί με μια πρώτη ματιά να μοιάζει με μια κοινή εφαρμογή αλλά παρόλα αυτά η εταιρεία Google έχει σχεδιάσει μία πλήρη εφαρμογή καταχώρησης σημειώσεων όπου οι χρήστες της μπορούν να καταγράψουν τις σημειώσεις τους, να δημιουργήσουν λίστες ελέγχου και πολλά άλλα. Είναι φανερό ότι σε σχέση με τις προηγούμενες δύο εφαρμογές που έχουν αναφερθεί έως τώρα δηλαδή την Evernote και την OneNote η συγκεκριμένη εφαρμογή υστερεί σε λειτουργικές δυνατότητες αλλά παρόλα αυτά χωρίς να παρεκκλίνει από τον βασικό σκοπό ύπαρξής της δηλαδή την καταγραφή σημειώσεων. Πιο συγκεκριμένα η εφαρμογή αυτή εστιάζει περισσότερο στη βασική ανάγκη που είναι η καταγραφή της σημείωσης χωρίς να αποπροσανατολίζει και να μπερδεύει τους χρήστες της με περαιτέρω λειτουργικότητες κάνοντάς την ίσως πιο αποτελεσματική σε σχέση με τις υπόλοιπες. Όσον αφορά τώρα το λειτουργικό κομμάτι της εφαρμογής αυτής δίνει την βασική δυνατότητα που δεν είναι άλλη από την καταγραφή σημειώσεων προσφέροντας παράλληλα την δυνατότητα προεπισκόπησης της σημείωσης χωρίς να χρειάζεται ο χρήστης να την ανοίξει ολόκληρη για να δει το περιεχόμενό της. Επιπλέον μπορεί το ίδιο το εργαλείο αυτό να μην παρέχει από μόνο του πλούσιες λειτουργικές δυνατότητες επεξεργασίας ωστόσο προσφέρει την δυνατότητα αντιγραφής μία σημείωσης στο Google Doc όπου εκεί κάθε σημείωση της εφαρμογής μετατρέπεται σε επεξεργάσιμο αρχείο και ο χρήστης μπορεί να επέμβει σε αυτό για περαιτέρω επεξεργασία. Επιπρόσθετα κατά την αποθήκευση κάθε σημείωσης μέσω της εφαρμογής αυτής καταγράφεται η τοποθεσία όπου πραγματοποιείται η ανάρτηση από τον χρήστη έτσι ώστε να τον βοηθήσει στο μέλλον να την αναζητήσει ευκολότερα. Τέλος όσον αφορά το λειτουργικό κομμάτι αυτής μια από τις βασικές και πιο χρήσιμες λειτουργίες της αποτελεί η ηχογράφηση ηχητικών σημειώσεων μέσω της εφαρμογής προσφέροντας παράλληλα πέρα από την καταγραφή του ήχου κατά την απαγγελία του χρήστη και την καταγραφή της σημείωσης ως κείμενο έτσι ώστε οι χρήστες της να μην χρειαστεί να την



μετατρέψουν σε δεύτερο χρόνο από ηχητική μορφή σε έντυπη. Κλείνοντας η συγκεκριμένη εφαρμογή προσφέρετε για λειτουργικά συστήματα Android και IOS για κινητές συσκευές αλλά και ως Web εφαρμογή.

#### 4. Zoho Notebook (Mac, IOS, Android)

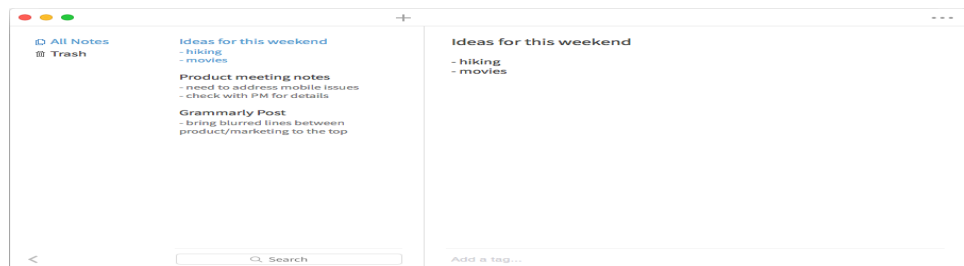


Η εφαρμογή Zoho αποτελεί μέρος των προϊόντων της οικογένεια Zoho. Πρόκειται για μία εφαρμογή καταχώρησης σημειώσεων όπου οι χρήστες τις μπορούν να επιλέξουν ποιο από τα μοντέρνα σχέδια της εφαρμογής που χρησιμοποιούνται ως ταπετσαρία ταιριάζει με το προφίλ τους ή και να σχεδιάσουν το δικό τους. Επίσης επιλογή δίνεται στους χρήστες της και κατά την εκτέλεση της βασική λειτουργίας δηλαδή την καταχώριση μίας σημειώσεων έτσι ώστε να μπορεί να διαλέξει ανάμεσα σε πέντε καρτέλες καταχώρισης όπου η κάθε μια από αυτές προσφέρει τις δικές της οθόνες και επιλογές. Πιο συγκεκριμένα η εφαρμογή αυτή προσφέρει δυνατότητες όπως η καταχώριση κειμένου στην σημείωση, την εγγραφή φωνής σε συνεργασία με την συσκευή Apple Watch, την ενσωμάτωση εικόνων και λιστών ελέγχου αλλά και την δημιουργία χειρόγραφων σημειώσεων με το δάχτυλο για κινητές συσκευές ή την χρήση ειδικής γραφίδας για σταθερούς και φορητούς υπολογιστές. Κλείνοντας με ότι αφορά το λειτουργικό κομμάτι της εφαρμογής προσφέρεται ως δυνατότητα στους χρήστες της η καταγραφή σημειώσεων επάνω σε ιστοσελίδες, σε στιγμιότυπα οθονών και σε διάφορα άλλα. Ωστόσο αξίζει να σημειωθεί ότι πέρα από τις λειτουργικές δυνατότητες που προσφέρονται προς εκτέλεση από έναν χρήστη η συγκεκριμένη εφαρμογή έχει εξασφαλίσει ότι καμία σημείωση δεν πρόκειται να χαθεί ακόμα και αν σβήστηκε κατά λάθος καθώς συγχρονίζει κάθε σημείωση της εφαρμογής με το σύννεφο έτσι ώστε να μπορεί να είναι ανακτήσιμη οποιαδήποτε στιγμή. Τέλος την εφαρμογή αυτή θα την βρουν οι χρήστες διαθέσιμη για κινητές συσκευές σε λειτουργικό σύστημα IOS και Android ενώ για σταθερούς υπολογιστές σε Mac.



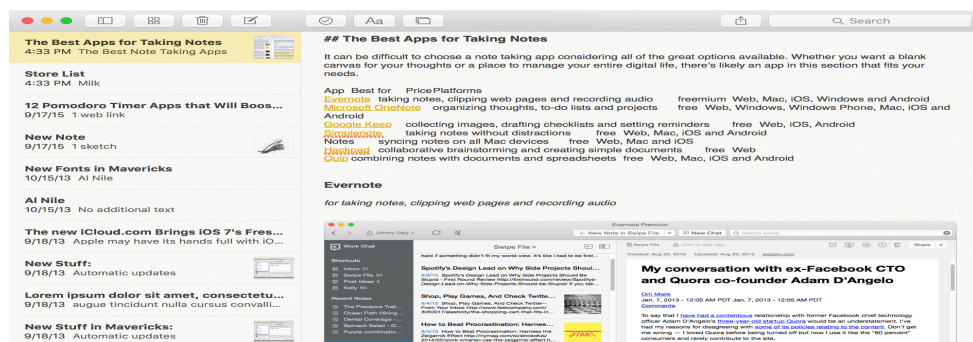


## 5. Simplenote (IOS, Android, Mac, Web)



Η εφαρμογή Simplenote αποτελεί μία απλή εφαρμογή σημειώσεων όπως δηλώνει και το όνομα της. Πρόκειται για μια εφαρμογή στην οποία δίνεται απλά ένας χώρος καταγραφής σημειώσεων με σχετικά πεδία για σημαντικές συναντήσεις, σχέδια για Σαββατοκύριακο και πολλά άλλα. Δεν παρέχει δυνατότητες όπως η καταχώρηση αρχείων εικόνας, ήχου και βίντεο αλλά παρέχει ένα τόσο εύχρηστο τρόπο καταγραφής και αναζήτησης σημειώσεων που πραγματικά διευκολύνει τους χρήστες της. Επίσης προσφέρει την δυνατότητα ανάκτησης οποιασδήποτε έκδοσης κάποιας σημείωσης έτσι ώστε τίποτα να μην χάνεται ακόμα και αν σβήστηκε κατά λάθος. Παράλληλα δίνει την δυνατότητα στους χρήστες της να μοιράζονται τις σημειώσεις τους με άλλους χρήστες της εφαρμογής διευκολύνοντας έτσι την συνεργασία μεταξύ τους. Τέλος η εφαρμογή αυτή είναι διαθέσιμη για λειτουργικά συστήματα IOS και Android για κινητές συσκευές, για Mac για φορητούς και σταθερούς υπολογιστές αλλά διατίθεται και ως Web εφαρμογή για τους online χρήστες.

## 6. Apple Notes (IOS,Mac,Web)



Η εφαρμογή Apple Notes αποτελεί μια εφαρμογή σχεδιασμένη από την εταιρεία Apple αποκλειστικά για συσκευές της εταιρείας όπως iPhone, iPad και Mac. Πρόκειται για μια απλή εφαρμογή καταγραφής σημειώσεων η οποία επιτρέπει σαν βασικό της μέλημα την καταγραφή σημειώσεων σε αυτές τις συσκευές, πραγματοποιεί προσθήκη σελιδοδεικτών σε ιστότοπους καθώς και αποθήκευση online άρθρων, ενώ προσφέρει επιπλέον την δυνατότητα κατασκευής σκίτσων και χειρόγραφων σημειώσεων με την βοήθεια ειδικής γραφίδας αλλά και την δημιουργία λίστας εργασιών. Παράλληλα παρέχεται η δυνατότητα συγχρονισμού μέσω του iCloud όλων των συσκευών Apple του χρήστη έτσι ώστε να μπορεί να έχει πρόσβαση στις σημειώσεις του μέσω κάθε μίας από



αυτές. Επιπλέον προσφέρει την δυνατότητα αποστολές αυτών μέσω ηλεκτρονικού ταχυδρομείου καθώς και των συγχρονισμό αυτών με οποιαδήποτε υπηρεσία ηλεκτρονικού ταχυδρομείου ο χρήστης της επιθυμεί (Gmail, Outlook). Τέλος όπως ήδη αναφέραμε την εφαρμογή αυτή οι χρήστες της θα την βρουν διαθέσιμη σε λειτουργικό σύστημα IOS για κινητές συσκευές, Mac για φορητούς και σταθερούς υπολογιστές αλλά και ως Web εφαρμογή.

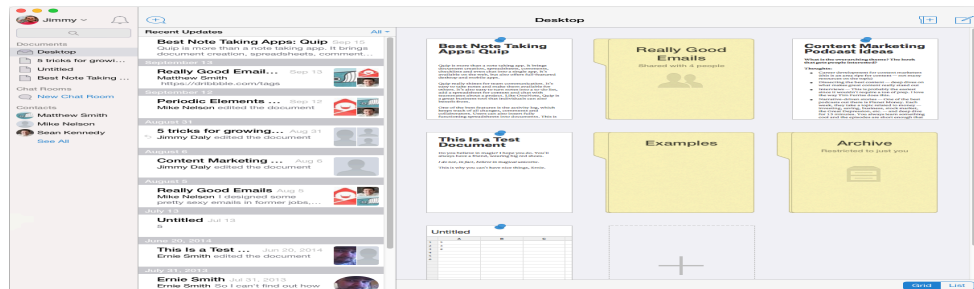
## 7. HackPad (IOS, Web)



Η εφαρμογή HackPad αποτελεί μια εφαρμογή που βασίζεται εξολοκλήρου στο σύννεφο και είναι σχεδιασμένη με τέτοιο τρόπο ώστε να μπορούν οι χρήστες της να καταγράψουν και να οργανώσουν οποιαδήποτε σημείωση τους. Όπως ακριβώς όλες οι εφαρμογές που έχουμε αναφέρει έως τώρα έτσι και η HackPad προσφέρει στους χρήστες της την δυνατότητα καταγραφής σημειώσεων που αποτελεί και τον βασικό σκοπό ύπαρξης της εφαρμογής αλλά παρέχοντας παράλληλα και αρκετές άλλες ισχυρές δυνατότητες. Επιγραμματικά θα μπορούσαμε να αναφέρουμε την μεταφόρτωσή αρχείων εικόνας στην σημείωση, την επεξεργασία κειμένου με μια πολύ ισχυρή και εύχρηστη γραμμή εργαλείων αλλά και την οργάνωση των σημειώσεων σε ομάδες χωρίς να υπάρχουν περιορισμοί που να δεσμεύουν την σημείωση με την ομάδα προσδίδοντας ευελιξία στους χρήστες ώστε να καταχωρούν την κάθε σημείωση όσες φορές το επιθυμούν και να την καταχωρούν όπου αυτοί κρίνουν ότι χρειάζεται να συμπεριλαμβάνεται. Επιπλέον το εργαλείο αυτό προσφέρει την δυνατότητα μετατροπής της σημείωσης σε online σύνδεσμο προσθέτοντας απλά τον χαρακτήρα @ ακολουθούμενο από το όνομα της αυτής καθώς και την χρήση οποιοδήποτε συνδέσμου σε υπηρεσίες όπως το Youtube και το Gmail. Ωστόσο η δυνατότητες της εφαρμογής αυτής δεν σταματάνε εδώ καθώς οι σημειώσεις μπορούν να γίνουν αντικείμενο κοινής χρήσης από τους χρήστες της την ίδια ακριβώς στιγμή ενώ παράλληλα όταν υπάρχει και η ανάγκη κοινής επεξεργασίας ο κάθε χρήστης που συμμετέχει στην ομάδα επεξεργασίας να μπορεί να καταγράψει τις σημειώσεις του σε αποκλειστικά δικό του χώρο και όλες να παρουσιάζονται στην ίδια σημείωση κάτω από μία κοινή σελίδα προβολής τους. Κλείνοντας η εφαρμογή αυτή διατίθεται σε λειτουργικό σύστημα IOS για κινητές συσκευές αλλά και ως Web εφαρμογή.

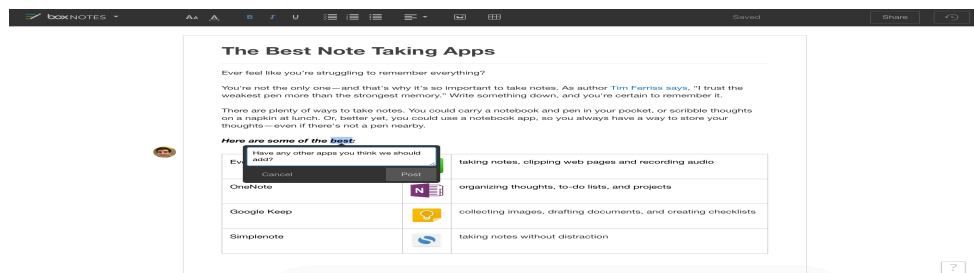


## 8. Quip (IOS, Android, Mac, Windows, Web)



Η εφαρμογή Quip δεν πρόκειται για μία απλή εφαρμογή για κινητές συσκευές και φορητούς υπολογιστές αλλά για μία ισχυρή εφαρμογή με αρκετές δυνατότητες. Η βασική της λειτουργικότητα αποτελεί η καταχώρηση σημειώσεων χωρίς αυτό να σημαίνει ότι οι δυνατότητες της περιορίζονται μόνο σε αυτή. Όλες οι δυνατότητες της εφαρμογής ξεκινούν από την καταχώρηση της σημείωσης και έπειτα. Αυτό με λίγα λόγια σημαίνει ότι ο χρήστης μπορεί να ενσωματώσει σε αυτή αρχεία κειμένου, υπολογιστικά φύλλα αλλά και συνομιλίες που αφορούν την σχετική σημείωση. Επιπλέον η εφαρμογή μπορεί να μην προσφέρει κάποια γραμμή εργαλείων για μορφοποιήσει παρόλα αυτά δίνει την δυνατότητα στο χρήστη να επιλέξει μεταξύ κάποιων προτύπων μορφοποιήσεις εγγράφου που προσφέρονται από την εφαρμογή. Επίσης παρέχονται ως επιπλέον λειτουργικότητα η δημοσίευση της σημείωσης στην σελίδα της εφαρμογής στο διαδίκτυο, η κοινοποίηση αυτής μεταξύ άλλων χρηστών της εφαρμογής που ο ίδιος ο χρήστης επιλέγει καθώς και η αποθήκευση της ως αρχείο pdf. Τέλος σε ότι αφορά το λειτουργικό κομμάτι αυτής παρότι κάθε σημείωση του χρήστη αποθηκεύεται στο δικό του προσωπικό φάκελο μέσα στην εφαρμογή δίνεται ως λειτουργικότητα η αναζήτηση σημειώσεων αλλά και περιεχομένων μέσα σε αυτές τόσο στο φάκελο του κάθε χρήστη όσο και σε όλες εκείνες τις σημειώσεις που ανήκουν σε άλλους και έχουν δημοσιευτεί ή κοινοποιηθεί μεταξύ των χρηστών της εφαρμογής. Κλείνοντας η εφαρμογή αυτή διατίθεται σε λειτουργικό σύστημα IOS και Android για κινητές συσκευές, σε Mac και Windows για υπολογιστές αλλά και ως Web εφαρμογή.

## 9. Box Notes (IOS, Android, Web)

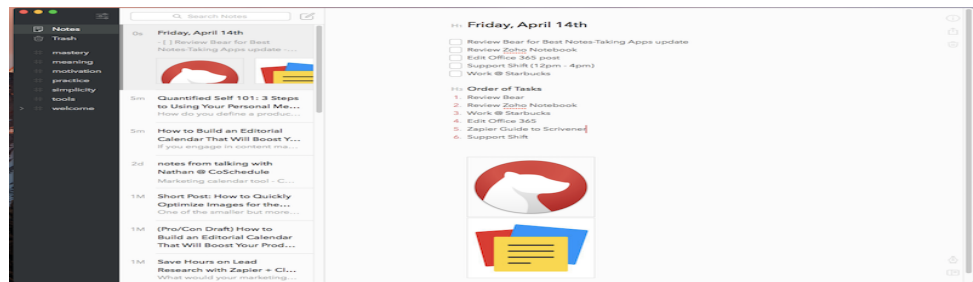


Την εφαρμογή Box Notes θα μπορούσαμε να την παρομοιάσουμε όσον αφορά την λειτουργικότητα της με την εφαρμογή Quip που αναφέραμε πιο πάνω καθώς παρουσιάζουν αρκετά κοινά στοιχεία. Πρόκειται για μία απλή εφαρμογή σημειώσεων



που ως βασική λειτουργία της έχει την καταχώρηση και επεξεργασία σημειώσεων μέσω μιας απλής γραμμής εργαλείων για μορφοποίηση κειμένου. Επιπλέον και αυτή η εφαρμογή προσφέρει την δυνατότητα εισαγωγής συνομιλιών αλλά και σχολίων μεταξύ των ενδιαφερόμενων χρηστών κάτω από την ίδια σημείωση. Ωστόσο η συγκεκριμένη εφαρμογή σε αντίθεση με άλλες παρόμοιες εφαρμογές αποθηκεύει τις σημειώσεις κάθε χρήστη σε ξεχωριστούς φακέλους κάτι που μπορεί να αποδειχθεί πολύ χρήσιμο αν και εφόσον η σημείωση αφορά κάποιο συγκεκριμένο θέμα και συνεργάζονται για αυτό παραπάνω από ένας χρήστες. Σε κάθε άλλη περίπτωση ίσως το μοντέλο αυτό αποθήκευσης της εφαρμογής να δυσκολεύει τους χρήστες της να αναζητήσουν τις σημειώσεις τους καθώς μπορεί να είναι διασκορπισμένες σε διαφορετικούς φακέλους μέσα στην εφαρμογή ανάλογα με το θέμα ή το έργο με το οποίο σχετίζονται. Τέλος η εφαρμογή αυτή είναι διαθέσιμη για κινητές συσκευές σε λειτουργικό σύστημα IOS και Android ενώ παράλληλα διατίθεται και ως Web εφαρμογή γενικότερα.

## 10. Bear (IOS, Mac)



Η συγκεκριμένη εφαρμογή πρόκειται καθαρά για ένα εργαλείο κοινής χρήσης σημειώσεων. Με την πρώτη μάτια η εφαρμογή αυτή θα μπορούσαμε να πούμε πως μοιάζει με την εφαρμογή Apple Note καθώς συγκεντρώνει και αυτή όλες τις σημειώσεις του χρήστη σε μία λίστα σημειώσεων με δυνατότητα προεπισκόπησης της κάθε μίας με ένα απλό πάτημα πάνω σε αυτή. Ωστόσο οι ομοιότητες μεταξύ των δύο περιορίζονται έως εδώ. Η συγκεκριμένη εφαρμογή προσφέρει μια πλούσια σε δυνατότητες επεξεργασίας γραμμή εντολών καθώς και την δυνατότητα καταχώρησης ετικετών στις σημειώσεις ώστε η οργάνωση και η αναζήτηση αυτών να μπορεί να γίνει εύκολα από τους χρήστες της. Επίσης μία ακόμα δυνατότητα που προσφέρεται σε αυτή είναι ότι οι χρήστες της μπορούν να καταχωρήσουν σημειώσεις πέραν από τον κλασικό τρόπο εισαγωγής δηλαδή γραπτός αλλά και μέσω απαγγελίας της σημείωσης. Επιπλέον λειτουργικότητες όπως η εξαγωγή της σημείωσης σε διάφορες μορφές αρχείων όπως Pdf, docx, html και άλλες αλλά και η δυνατότητα συγχρονισμού των συσκευών του χρήστη ώστε να μπορεί από οποιαδήποτε από αυτές να έχει πρόσβαση στις σημειώσεις του προσφέρονται από την έκδοση επί πληρωμής. Κλείνοντας η εφαρμογή αυτή είναι διαθέσιμη για κινητές συσκευές σε λειτουργικό σύστημα IOS και σε Mac για σταθερούς και φορητούς υπολογιστές.

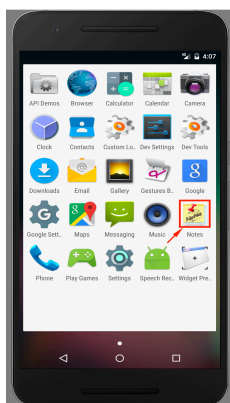


## 4. Εγχειρίδιο Χρήσης

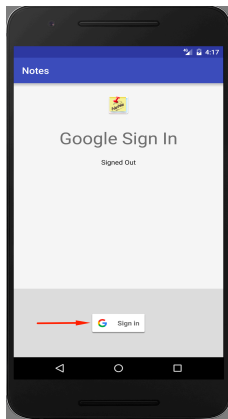
Στόχος του συγκεκριμένου εγχειριδίου είναι η αποσαφήνιση της λειτουργικότητας και η εξοικείωση του χρήστη με την εφαρμογή της παρούσας πτυχιακής εργασίας. Το θέμα που εξετάζεται στην συγκεκριμένη πτυχιακή αφορά μια εφαρμογή για σημειώσεις που απευθύνεται σε σπουδαστές κάποιου εκπαιδευτικού ιδρύματος και τον τρόπο λειτουργίας αυτής προκειμένου να εξυπηρετήσει τις ανάγκες τους για την ανταλλαγή σημειώσεων και την οργανωτική αποθήκευση και εύρεση αυτών. Η εφαρμογή θα μπορούσαμε να αναφέρουμε ότι διαρθρώνεται από επτά βασικές λειτουργίες. Επιγραμματικά αυτές είναι :

- Η είσοδος του χρήστη στην εφαρμογή.
- Η καταχώρηση νέας σημείωσης.
- Η επεξεργασία ή διαγραφή κάποιας υπάρχουσας.
- Η κατηγοριοποίηση των σημειώσεων.
- Η εύρεση σημειώσεων ανά κατηγορία.
- Η αναζήτηση όλων των σημειώσεων που έχουν καταχωρηθεί στην εφαρμογή από όλους τους χρήστες της.
- Η ενημέρωση των ενδιαφερόμενων χρηστών για την μεταβολή ή την δημιουργία κάποιας σημείωσης.

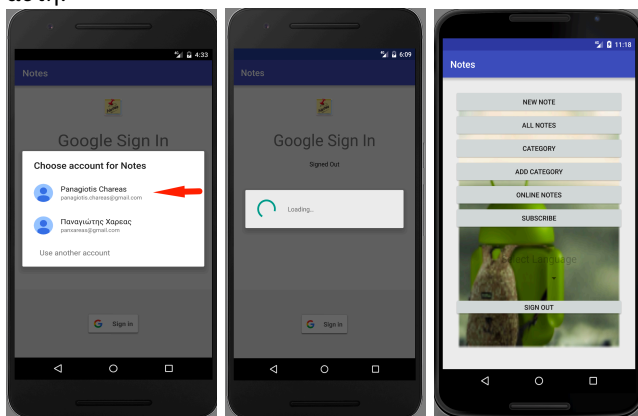
Παρακάτω αναλύονται τα βασικά βήματα εκτέλεσης και χρήσης της εφαρμογής. Αφού ο χρήστης κατεβάσει και εγκαταστήσει επιτυχώς την εφαρμογή εμφανίζεται στις εγκατεστημένες εφαρμογές της εκάστοτε συσκευής με λειτουργικό σύστημα Android και η εφαρμογή των σημειώσεων. Συνεπώς ο χρήστης με την επιτυχημένη εγκατάσταση της εφαρμογής αναμένει να δει στην οθόνη της συσκευής του το εικονίδιο της εφαρμογής σημειώσεων.



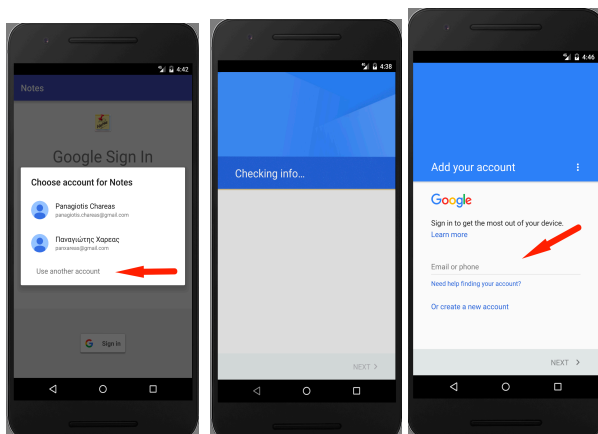
Πατώντας επάνω στο εικονίδιο της εφαρμογής ο χρήστης εκκινεί το πρόγραμμα για τις σημειώσεις. Μετά την εκκίνηση της ζητείται από τον χρήστη να συνδεθεί με ηλεκτρονικό λογαριασμό ( gmail ) στην εφαρμογή πατώντας το κουμπί με την επιγραφή sign in.



Πατώντας το κουμπί ο χρήστης μεταβαίνει στην επόμενη οθόνη που του δίνονται δύο επιλογές. Η πρώτη επιλογή που έχει είναι αν έχει συνδεθεί στην εφαρμογή με κάποιο λογαριασμό αυτός έχει αποθηκευτεί στην συγκεκριμένη συσκευή και μπορεί άμεσα να τον επιλέξει και να κάνει είσοδο σε αυτή.



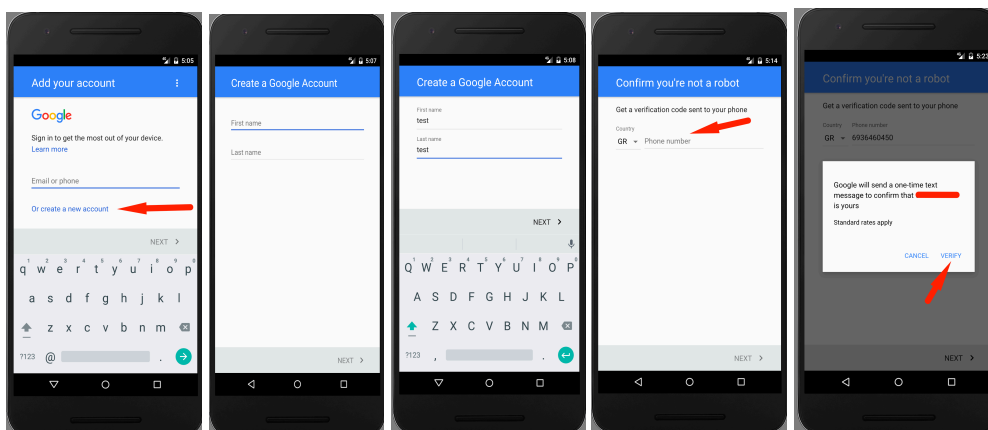
Η δεύτερη επιλογή που έχει είναι αν ο χρήστης συνδέεται για πρώτη φορά στην παρούσα εφαρμογή ή επιθυμεί να συνδεθεί με κάποιο άλλο λογαριασμό. Στην προκειμένη περίπτωση επιλέγει το κουμπί Use another account.



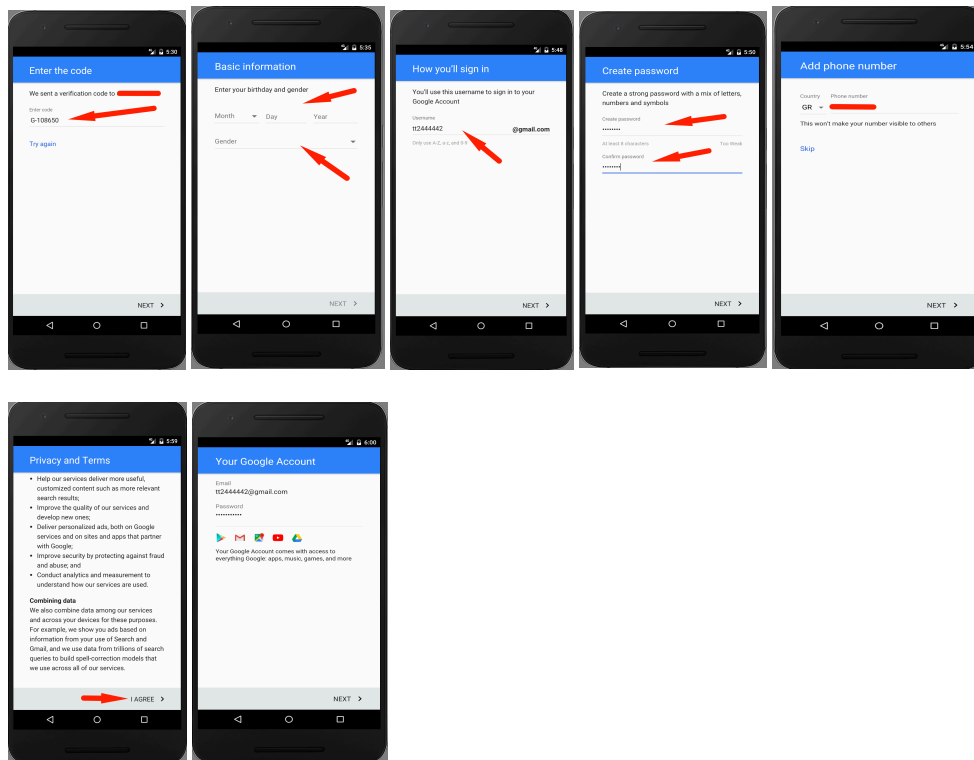


Στο σημείο αυτό ο χρήστης βρίσκεται πάλι ανάμεσα σε δύο επιλογές. Η πρώτη επιλογή που έχει είναι να συμπληρώσει το email ή τον αριθμό τηλεφώνου που είχε καταχωρήσει κατά την δημιουργία λογαριασμού gmail και να επιλέξει το κουμπί NEXT για να προχωρήσει και να γίνει εύρεση του σχετικού ηλεκτρονικού λογαριασμού. Έπειτα από την επιτυχή εύρεση του λογαριασμού αυτός εμφανίζεται στο μενού επιλογής λογαριασμών όπως είδαμε και παραπάνω και ο χρήστης επιλέγοντας τον μπορεί να κάνει είσοδο στην εφαρμογή.

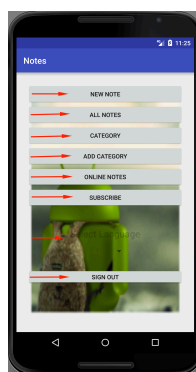
Η δεύτερη επιλογή που έχει είναι η δημιουργία εκ νέου λογαριασμού gmail επιλέγοντας το κουμπί Or create a new account. Καταχωρεί τα στοιχεία όνομα και επώνυμο και επιλέγει το κουμπί NEXT για να συνεχίσει. Εν συνεχεία επιλέγει την χώρα και καταχωρεί τον αριθμό τηλεφώνου κάτι που απαιτείται από την google προκειμένου να επιβεβαιώσει πως η ενέργεια δημιουργίας λογαριασμού έγινε από κάποιο χρήστη και όχι μέσω κάποιου κακόβουλου λογισμικού και στην συνέχεια επιλέγει το κουμπί NEXT. Η google ενημερώνει τον χρήστη ότι θα αποσταλεί μήνυμα στον αριθμό τηλεφώνου που δηλώθηκε στο προηγούμενο βήμα και ζητά επιβεβαίωση από αυτόν ο οποίος στην περίπτωση που συμφωνεί επιλέγει την επιλογή VERIFY. Αμέσως μετά την επιβεβαίωση ένα μήνυμα με ένα κωδικό αποστέλλεται στον αριθμό που δηλώθηκε και ο χρήστη καλείται να τον συμπληρώσει στο σχετικό πεδίο και να πατήσει NEXT για να συνεχίσει. Στην επόμενη οθόνη συμπληρώνει τα στοιχεία γέννησης του και το φύλο του και επιλέγει το NEXT. Στο σημείο αυτό ζητείται η καταχώρηση του καινούργιου ηλεκτρονικού λογαριασμού και επιλέγει το NEXT. Εν συνεχεία επιλέγει ένα μυστικό κωδικό και τον συμπληρώνει στα αντίστοιχα πεδία και πατάει το κουμπί NEXT για να προχωρήσει. Ο χρήστης μπορεί αμέσως μετά να καταχωρήσει τον αριθμό τηλεφώνου του εάν το επιθυμεί και να συνεχίσει πατώντας NEXT. Παρακάτω διαβάζει τους όρους χρήσης της google για το λογαριασμό και σε περίπτωση που συμφωνεί επιλέγει το I AGREE για να συνεχίσει παρακάτω. Τέλος εμφανίζεται ο λογαριασμός που μόλις δημιούργησε για να μπορεί να κάνει είσοδο στην εφαρμογή.







Με την είσοδο του στην εφαρμογή όπως είδαμε και παραπάνω ο χρήστης μεταβαίνει στην ακόλουθη οθόνη.



Η οθόνη αυτή αποτελεί το βασικό μενού της εφαρμογής. Το μενού αυτό συντίθεται από οκτώ λειτουργίες οι οποίες αναλύονται αναλυτικότερα στην συνέχεια του εγχειριδίου :

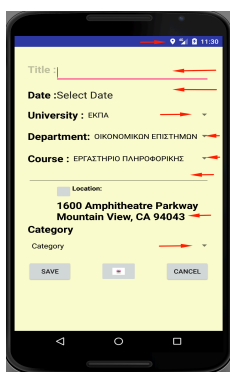
- Την δημιουργία νέας σημείωσης.
- Την προβολή όλων των σημειώσεων που έχει ο χρήστης στην κατοχή του.
- Την Εύρεση των αποθηκευμένων σημειώσεων αν κατηγορία.
- Την δημιουργία κατηγορίας σημειώσεων.
- Την προβολή όλων των σημειώσεων που έχουν ανέβει από χρήστες της εφαρμογής.
- Την εγγραφή για ενημέρωση σχετικά με νέες σημειώσεις.



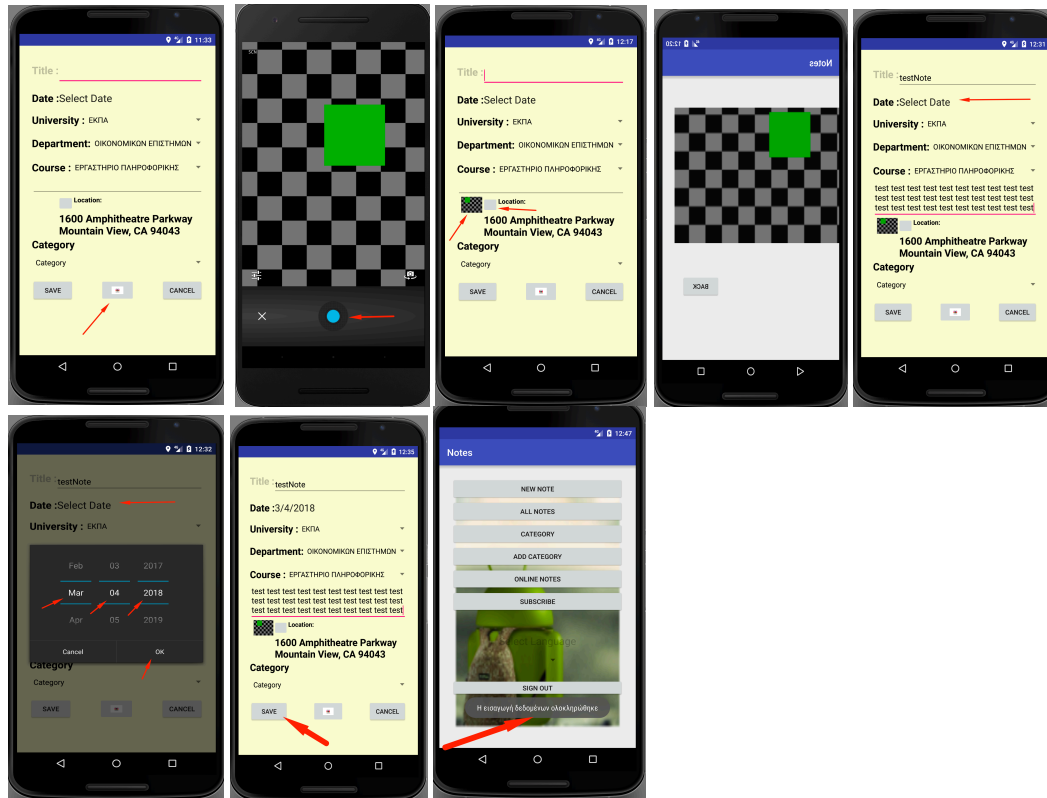


- Την επιλογή γλώσσας της εφαρμογής.
- Την έξοδο από την εφαρμογή.

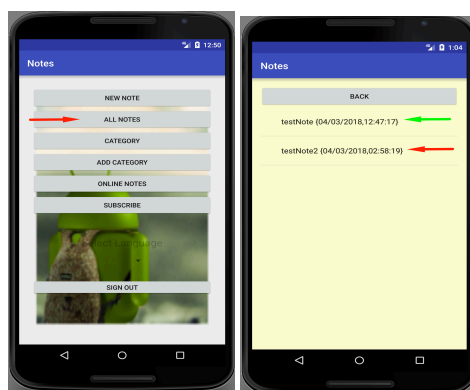
Η πρώτη επιλογή που έχει ο χρήστης μεταβαίνοντας στο βασικό μενού της εφαρμογής είναι η δημιουργία νέας σημείωσης. Με την επιλογή αυτή λοιπόν δίνεται η δυνατότητα σε ένα χρήστη της εφαρμογής να δημιουργήσει μια νέα σημείωση και αυτή αμέσως να μπορεί να είναι ορατή από όλους τους χρήστες της εφαρμογής. Αναλυτικότερα ο χρήστης πατώντας το κουμπί NEW NOTE (Νέα Σημείωση) οδηγείται στην παρακάτω οθόνη.



Στο σημείο αυτό έχει την δυνατότητα να καταχωρήσει στα αντίστοιχα πεδία τα βασικά στοιχεία μια σημείωσης. Αυτά είναι σύμφωνα με την εφαρμογή ο τίτλος, η ημερομηνία, το πανεπιστήμιο, το τμήμα, το μάθημα, το κείμενο της σημείωσης, η καταγραφή της τοποθεσίας (αν επιθυμεί ο χρήστης να καταγραφεί πρέπει να ανοίξει το gps), μια κατηγορία αν έχει δημιουργήσει πιο πριν κάποια συγκεκριμένη και θέλει να την εντάξει σε αυτή για να διευκολύνει την αναζήτηση της. Επιπλέον δίνεται η δυνατότητα στο χρήστη (εάν και εφόσον το επιθυμεί) να τραβήξει μια φωτογραφία μια σημείωση και να την ανεβάσει και αυτή μαζί με τα υπόλοιπα πεδία της σημείωσης που μόλις δημιούργησε. Η ενέργεια αυτή μπορεί να πραγματοποιηθεί πατώντας το κουμπί με την φωτογραφική μηχανή. Έπειτα από την ενέργεια αυτή η κάμερα της εκάστοτε συσκευής ανοίγει και ο χρήστης μπορεί να τραβήξει την φωτογραφία. Μετά την λήψη της η εφαρμογή επιστρέφει στο σημείο που γίνεται η καταχώρηση νέας σημείωσης με την διαφορά ότι εμφανίζεται σε μικρογραφία η φωτογραφία που μόλις τραβήχτηκε. Από την στιγμή που ο χρήστης έχει τραβήξει κάποια φωτογραφία και αυτή εμφανιστεί στην οθόνη το γκριζό κουμπί πάνω από αυτή ενεργοποιείται. Το κουμπί αυτό πατώντας το δίνει τη δυνατότητα προβολής της φωτογραφίας και οδηγεί τον χρήστη σε μία άλλη οθόνη όπου γίνεται η προβολή. Από το σημείο αυτό για να επιστρέψει ο χρήστης στην προηγούμενη οθόνη πατάει το κουμπί BACK. Εφόσον ο χρήστης έχει καταγράψει την νέα σημείωση και επιθυμεί να την αποθήκευση και τοπικά στην συσκευή του αλλά και ταυτόχρονα να μπορεί να την κατεβάσει οποιοσδήποτε άλλος χρήστης πατάει το κουμπί SAVE. Μετά την επιτυχημένη αποθήκευση της σημείωσης μήνυμα επιτυχής καταχώρησης εμφανίζεται στο κάτω μέρος της οθόνης. Σε περίπτωση που ο χρήστης δεν επιθυμεί να ολοκληρώσει την καταχώρηση ή θέλει να επιστρέψει στην αρχική σελίδα πατάει το κουμπί CANCEL.



Επιστρέφοντας πάλι στο αρχικό μενού της εφαρμογής η δεύτερη επιλογή που έχει ο χρήστης είναι η προβολή όλων των σημειώσεων που έχει στην κατοχή του. Αυτό σημαίνει πως στην ενότητα αυτή της εφαρμογής ο χρήστης μπορεί να αναζητήσει τόσο τις σημειώσεις που ο ίδιος ανέβασε όσο και σημειώσεις που έχει κατεβάσει και ανήκουν σε άλλους χρήστες. Για να προχωρήσει σε αυτή πατάει το κουμπί ALL NOTES (ΟΛΕΣ ΟΙ ΣΗΜΕΙΩΣΕΙΣ) και μεταβαίνει στην επόμενη οθόνη. Στην οθόνη αυτή ο χρήστης βλέπει μια λίστα με όλες τις σημειώσεις που έχει είτε αυτές είναι σημειώσεις που ο ίδιος δημιούργησε (το πράσινο βέλος αποτελεί μια τέτοια σημείωση) είτε σημειώσεις άλλων (το κόκκινο βέλος αποτελεί μια τέτοια) που ο χρήστης έχει κατεβάσει τοπικά στην συσκευή του. Το κατέβασμα μιας σημείωσης που δεν έχει δημιουργηθεί από τον ίδιο τον χρήστη αλλά από κάποιον άλλον θα το δούμε παρακάτω σε αυτή την ενότητα.

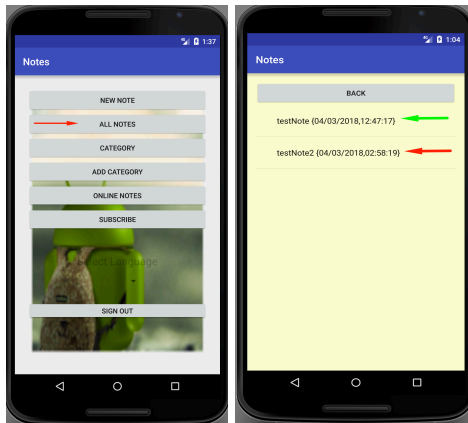




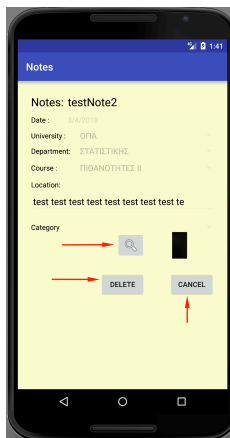
Έχει σημασία για την εφαρμογή σε ποιον ανήκει η σημείωση καθώς δίνει διαφορετικές επιλογές στο χρήστη ανάλογος αν είναι ο ιδιοκτήτης ή απλός χρήστης της σημείωσης. Ας δούμε αρχικά την περίπτωση που ο χρήστης θέλει να δει κάποια από τις σημειώσεις που αυτός έχει δημιουργήσει (περίπτωση σημείωσης με το πράσινο βέλος). Πατώντας λοιπόν στην συγκεκριμένη σημείωση από την λίστα ο χρήστης μεταβαίνει στην επόμενη οθόνη. Η εφαρμογή στο σημείο αυτό γνωρίζει αν ο χρήστης είναι ιδιοκτήτης της σημείωσης ή όχι. Στην προκειμένη περίπτωση η οθόνη ανήκει σε χρήστη με την ιδιότητα του ιδιοκτήτη. Συνεπώς από την στιγμή που η σημείωση ανήκει στον χρήστη αυτόν δίνεται η δυνατότητα επεξεργασίας όλων των πεδίων της. Μία άλλη δυνατότητα που δίνεται σε αυτή την οθόνη είναι η προβολή της φωτογραφίας που είχε η συγκεκριμένη σημείωση εάν αυτή υπάρχει. Πατώντας λοιπόν επάνω στο κουμπί με τον μεγεθυντικό φακό ο χρήστης μεταβαίνει σε μία οθόνη όπου προβάλλεται η φωτογραφία. Από την οθόνη εκείνη ο χρήστης επιστρέφει πίσω στο προηγούμενο σημείο πατώντας το κουμπί BACK. Επιπλέον δίνεται η δυνατότητα στον χρήστη αν επιθυμεί να αλλάξει και την φωτογραφία που έχει ανεβάσει με την συγκεκριμένη σημείωση πατώντας το κουμπί CHANGE. Με το πάτημα του συγκεκριμένου κουμπιού ανοίγει η κάμερα της συσκευής και ο χρήστης μπορεί να ξανατραβήξει εκ νέου μια καινούργια φωτογραφία. Μετά την λήψη επιστρέφει στην οθόνη που ήταν και η φωτογραφία έχει αντικατασταθεί με την νέα και το κουμπί με τον μεγεθυντικό φακό έχει κρυφτεί καθώς η φωτογραφία μόλις τραβήχτηκε από τον χρήστη και δεν υπάρχει λόγος προβολής της. Παρόλα αυτά η δυνατότητα προβολής δεν καταργείται οριστικά μετά την λήψη νέας φωτογραφίας. Εμφανίζεται πάλι όταν ο χρήστης αποφασίσει πάλι να ξανά δει τη σημείωση της οποίας είναι ιδιοκτήτης όπως συνέβη και πιο πάνω. Μετά την επεξεργασία των στοιχείων της σημείωσης ο χρήστης προκειμένου να καταχωρηθούν οι αλλαγές τόσο στην συσκευή αλλά και προκειμένου να γίνουν ορατές και από άλλους είτε αυτοί έχουν ήδη κατεβάσει την σημείωση αυτή είτε όχι πρέπει να πατήσει το κουμπί UPDATE. Με την επιτυχημένη ενημέρωση της σημείωσης μήνυμα εμφανίζεται στο κάτω μέρος της οθόνης. Επιπλέον με την ενημέρωση της εμφανίζεται επάνω και αριστερά στην οθόνη ένα push notification με το σύμβολο της εφαρμογής. Σκοπός αυτής της ειδοποίησης που την βλέπει τόσο ο ίδιος ο χρήστης που έχει την ιδιότητα του ιδιοκτήτη όσο και χρήστες που απλά την έχουν κατεβάσει είναι να τους ενημερώσει πως η σημείωση άλλαξε από τον ιδιοκτήτη προκειμένου να μπου να την δουν και αν επιθυμούν να λάβουν τις αλλαγές που έκανε ο ιδιοκτήτης σβήνοντας την παλιά και ξανακατεβάζοντας την νέα σημείωση. Πατώντας επάνω στην ειδοποίηση ο χρήστης μεταφέρεται στη σημείωση που άλλαξε και μπορεί αν θελήσει να την κατεβάσει ξανά. Επίσης στην οθόνη αυτή δίνεται η δυνατότητα στον χρήστη – ιδιοκτήτη να διαγράψει την σημείωση. Ωστόσο ακόμα και στην περίπτωση του χρήστη με ιδιότητα ιδιοκτήτη δεν διαγράφεται από τις online notes πάρα μόνο τοπικά από τον χώρο αποθήκευσης της συσκευής. Η διαδικασία αυτή επιτυγχάνεται πατώντας το κουμπί DELETE. Με την ολοκλήρωση της διαγραφής μήνυμα εμφανίζεται στο κάτω μέρος της οθόνης. Συνεπώς ο χρήστης επιστρέφονται στην οθόνη με όλες τις σημειώσεις που έχει μετά την διαγραφή η σημείωση που έσβησε δεν θα εμφανίζεται πλέον εκεί. Τελειώνοντας την λειτουργικότητα της οθόνης του χρήστη - ιδιοκτήτη προκειμένου να ακύρωση μια ενέργεια ή να επιστρέψει στην προηγούμενη οθόνη πατάει το κουμπί CANCEL.



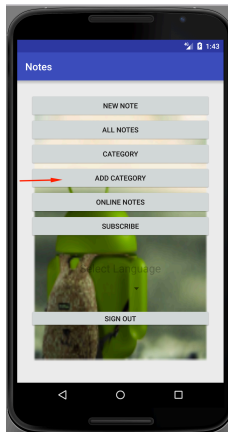
Όπως αναφέρθηκε η εφαρμογή γνωρίζει σε ποιόν ανήκει η σημείωση και ποιος την έχει απλά κατεβάσει. Παραπάνω είδαμε την περίπτωση που ο χρήστης είναι ιδιοκτήτης της σημείωσης. Εν συνεχεία λοιπόν θα εξηγήσουμε τι συμβαίνει στην περίπτωση που ο χρήστης δεν είναι ιδιοκτήτης της σημείωσης και έχει πατήσει στην ενότητα ALL NOTES και επιλέγει μια τέτοια σημείωση (σημείωση με κόκκινο βέλος).



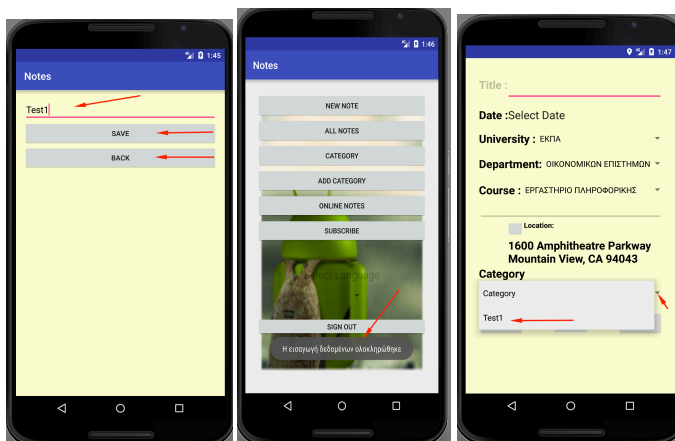
Πατώντας επάνω στο όνομα της σημείωσης αυτή ανοίγει και ο χρήστης πλέον μπορεί να δει μόνο την σημείωση που έχει κατεβάσει χωρίς να μπορεί να επεξεργαστεί κανένα από τα πεδία της εφόσον αυτή δεν του ανήκει. Οι επιλογές που έχει ο χρήστης στην προκειμένη περίπτωση είναι παρόμοιες με αυτές του χρήστη – ιδιοκτήτη αλλά σίγουρα λιγότερες. Αναλυτικότερα ο χρήστης στη περίπτωση αυτή έχει δυνατότητα προβολής φωτογραφίας μέσω του μεγεθυντικού φακού, διαγραφής της σημείωσης τοπικά από την συσκευή μέσω του κουμπιού DELETE και ακύρωση - επιστροφή μέσω του κουμπιού CANCEL κατά τον ίδιο τρόπο λειτουργίας αυτών όπως και στην περίπτωση του χρήστη ιδιοκτήτη.



Επιστρέφοντας πάλι στο βασικό Μενού της εφαρμογής η επόμενη επιλογή που δίνεται στο χρήστη και που θα αναλύσουμε είναι η δημιουργία κατηγορίας. Σκοπός της επιλογής αυτής είναι η δημιουργία μιας ευρύτερης κατηγορίας κάτω από την οποία μπορεί ο χρήστης να εντάξει τις μελλοντικές του σημειώσεις. Αυτή η κατηγοριοποίηση προσφέρει εν μέρει μια καλύτερη οργανωτική δομή των σημειώσεων με σκοπό να διευκολύνει την εύρεση τους.



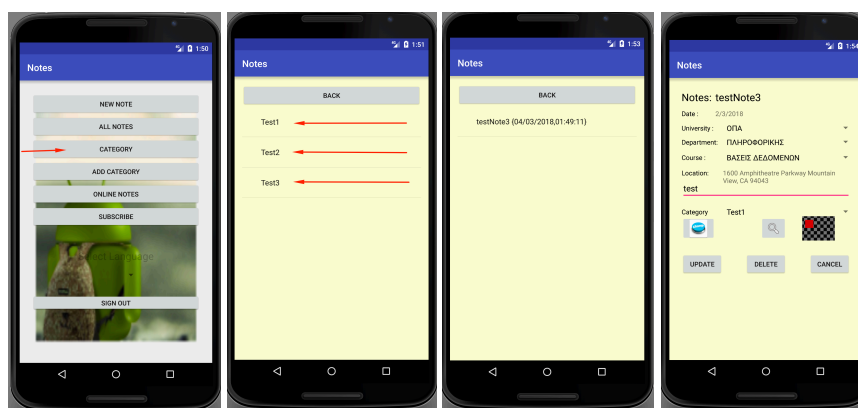
Πατώντας λοιπόν την επιλογή ADD CATEGORY ο χρήστης οδηγείται στην παρακάτω οθόνη. Στην οθόνη αυτή ο χρήστης μπορεί να δημιουργήσει μια νέα κατηγορία σημειώσεων πληκτρολογώντας το όνομα αυτής στο πλαίσιο που βρίσκεται στο πάνω μέρος της οθόνης. Εφόσον είναι σίγουρος για την ονομασία της κατηγορίας πατάει το κουμπί SAVE προκειμένου να αποθηκευτεί. Με την ολοκλήρωση της διαδικασίας μήνυμα επιτυχούς αποθήκευσης εμφανίζεται στο κάτω μέρος της οθόνης. Πατώντας τώρα ο χρήστης στη δημιουργία νέας σημείωσης (NEW NOTE) στο βασικό μενού επιλογών ύστερα από την δημιουργία κατηγορίας που δείξαμε μόλις τώρα εμφανίζεται στην αντίστοιχη λίστα Category και η κατηγορία που μόλις δημιουργήθηκε. Με αυτόν τον τρόπο ο χρήστης εφόσον έχει προνοήσει να δημιουργήσει κατηγορίες μπορεί κάθε νέα σημείωση να την εντάξει σε κάποια από αυτές. Επιστρέφονται πάλι στην οθόνη δημιουργίας νέας κατηγορίας ο χρήστης μπορεί να επιστρέψει στο βασικό μενού πατώντας το κουμπί BACK.



Με αφορμή λοιπόν τα όσα ειπώθηκαν παραπάνω για την κατηγοριοποίηση των σημειώσεων η εφαρμογή δίνει την δυνατότητα στους χρήστες που έχουν κατηγοριοποιήσει τις σημειώσεις τους να μπορούν να τις αναζητήσουν πιο εύκολα. Η επιλογή αυτή δίνεται από το αρχικό μενού πατώντας το κουμπί CATEGORY. Πατώντας την επιλογή αυτή ο χρήστης οδηγείται σε μία οθόνη η οποία περιλαμβάνει μια λίστα με όλες τις κατηγορίες που έχει δημιουργήσει μέχρι στιγμής στην εφαρμογή. Ο χρήστης επιλέγοντας κάποια από τις κατηγορίες που εμφανίζονται στην οθόνη θα του εμφανιστεί μια λίστα που θα περιλαμβάνει σημειώσεις τις οποίες έχει εντάξει κάτω από την συγκεκριμένη κατηγορία που επέλεξε. Στο παρακάτω παράδειγμα θεωρούμε ότι επέλεξε την κατηγορία HISTORY και

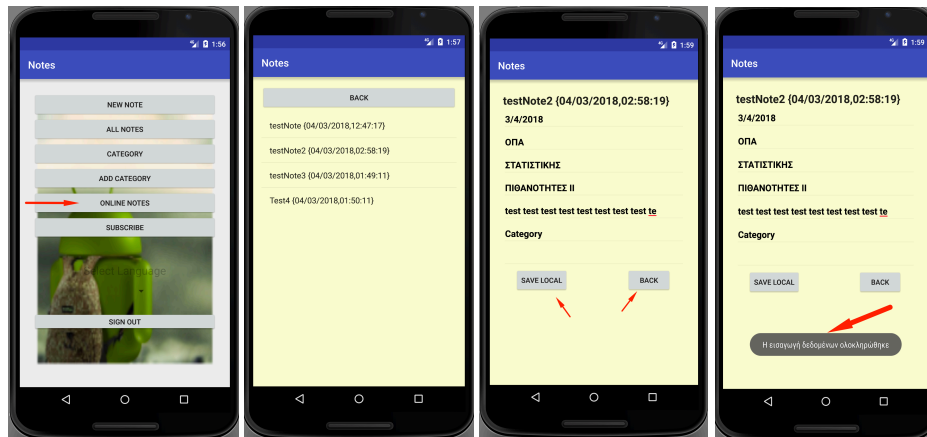


εμφανίστηκε η παρακάτω λίστα. Πατώντας επάνω σε κάποια σημείωση μεταβαίνει σε αυτή με δυνατότητες επεξεργασίας όπως αυτές που εξηγήθηκαν παραπάνω στην περίπτωση του χρήστη με ιδιότητα ιδιοκτήτης. Αυτό συμβαίνει καθώς ο χρήστης αναζητά ανά κατηγορία μόνο τις σημειώσεις που ο ίδιος έχει δημιουργήσει και κατηγοριοποιήσει. Επιγραμματικά αναφέρουμε από τις λειτουργίες που παρέχονται στην συγκεκριμένη οθόνη την επεξεργασία των πεδίων της σημείωσης, την ενημέρωση αυτών αλλά και όλων των ενδιαφερόμενων σχετικά με την σημείωση χρηστών, την εκ νέου λήψη φωτογραφίας είτε αυτή υπάρχει ήδη είτε όχι, την αλλαγή κατηγορίας της σημείωσης, την διαγραφή της σημείωσης και την ακύρωση – επιστροφή στην προηγούμενη οθόνη.

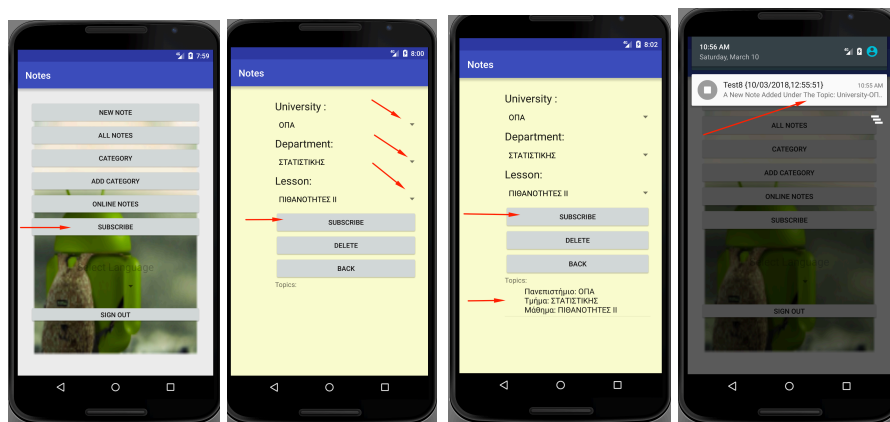


Επιστρέφοντας πάλι στο αρχικό μενού η εφαρμογή παρέχει στον χρήστη της μια ακόμα λειτουργία αυτή της προβολής όλων των σημειώσεων που έχουν καταχωρηθεί στην εφαρμογή από όλους τους χρήστες. Συνεπώς με την λειτουργία αυτή δίνεται η δυνατότητα σε οποιονδήποτε χρήστη της εφαρμογής να αναζητήσει όλες τις δημοσιευμένες σημειώσεις με σκοπό να τις κατεβάσει αν το επιθυμεί στην συσκευή του. Η επιλογή αυτή παρέχεται μέσω του κουμπιού ONLINE NOTES. Πατώντας την συγκεκριμένη επιλογή ο χρήστης μεταφέρεται σε μια οθόνη όπου του παρουσιάζονται όλες οι σημειώσεις που έχει η εφαρμογή. Πατώντας ο χρήστης επάνω στο όνομα κάποιας σημείωσης αυτή ανοίγει. Στην οθόνη που προβάλλεται ο χρήστης μπορεί να διαβάσει την σημείωση αλλά του δίνεται και η επιλογή να την κατεβάσει τοπικά στην συσκευή του. Η ενέργεια αυτή επιτυγχάνεται πατώντας το κουμπί SAVE LOCAL. Με το πάτημα του κουμπιού στην περίπτωση που η σημείωση δεν υπάρχει ήδη στις σημειώσεις του χρήστη και ύστερα από την λήψη της, μήνυμα επιτυχίας εμφανίζεται στην οθόνη. Αντιθέτως αν η σημείωση υπάρχει σχετικό μήνυμα εμφανίζεται και η σημείωση δεν ξανακατεβαίνει. Επιστροφή στην προηγούμενη οθόνη ο χρήστης κάνει με πατώντας το BACK.

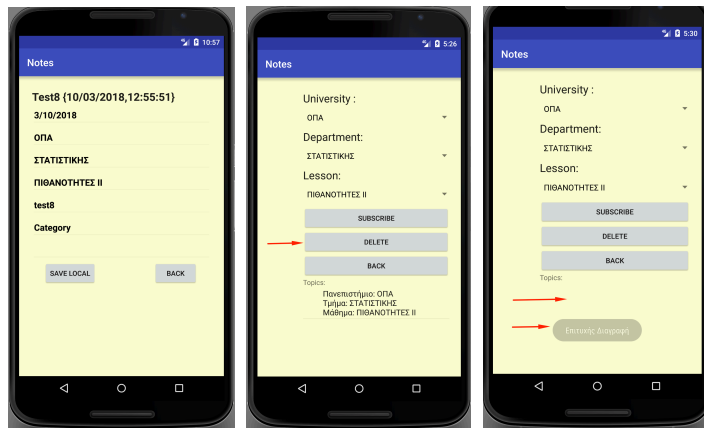




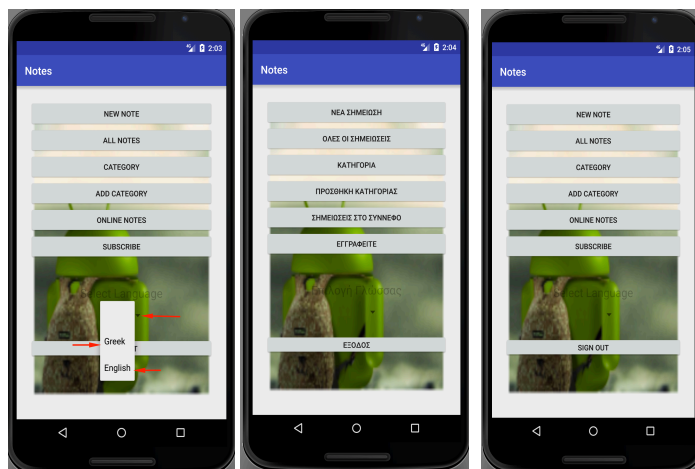
Η εφαρμογή επιπλέον παρέχει σαν επιλογή στον χρήστη την δυνατότητα να γραφτεί και λαμβάνει ενημερώσεις σχετικά με σημειώσεις που τον αφορούν. Η δυνατότητα αυτή παρέχεται στο βασικό μενού πατώντας την επιλογή Εγγραφέιτε (Subscribe). Επιλέγοντας την λοιπόν οδηγείτε στην παρακάτω οθόνη. Στην οθόνη αυτή ο χρήστης επιλέγει το Πανεπιστήμιο, το Τμήμα και το μάθημα για το οποίο θέλει να λαμβάνει ενημερώσεις για σημειώσεις που ανέβηκαν σχετικά με αυτό. Μόλις ολοκληρώσει με την επιλογή του μαθήματος που θέλει να λαμβάνει ενημερώσεις πατάει το κουμπί Subscribe (Εγγραφέιτε) και εμφανίζεται η εγγραφή του στην λίστα με τα σχετικά θέματα που ακολουθεί. Από την στιγμή που ο χρήστης γραφέιτε σε ένα θέμα ενημερώσεις σχετικά με αυτό θα εμφανίζονται στο πάνω μέρος της οθόνης. Πατώντας επάνω σε αυτές οδηγείτε στην σχετική σημείωση όπου μπορεί να την διαβάσει και να την κατεβάσει αν το επιθυμεί. Επιστρέφονται στην οθόνη όπου ο χρήστης μπορεί να κάνει σχετικές εγγραφές για ειδοποιήσεις παρέχετε και η δυνατότητα απεγγραφής από κάποιο θέμα. Αυτό επιτυγχάνετε επιλέγοντας το συνδυασμό Πανεπιστήμιο, Τμήμα και Μάθημα από το οποίο θέλει να διαγραφεί και πατάει το κουμπί Διαγραφή (Delete). Με την ολοκλήρωση της ενέργεια αυτής μήνυμα επιτυχούς διαγραφής εμφανίζεται στην οθόνη και το αντίστοιχο θέμα εξαφανίζεται από την λίστα των θεμάτων που ακολουθεί ο χρήστης. Τέλος και σε αυτή την οθόνη ο χρήστης μπορεί να επιστρέψει στο αρχικό μενού των επιλογών επιλέγοντας το κουμπί Πίσω (Back).



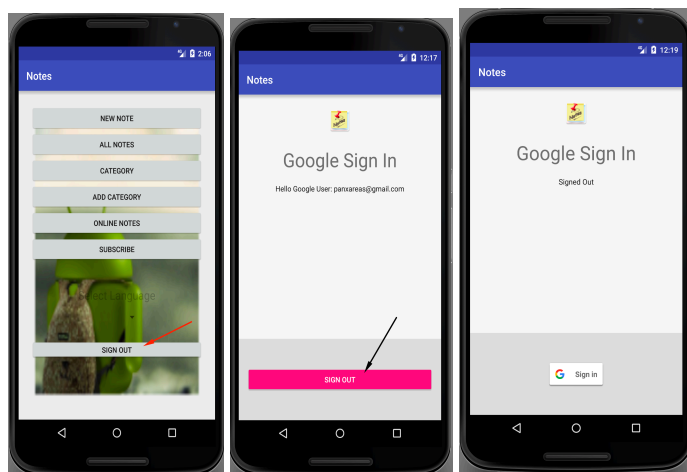




Τελειώνοντας αναφέρουμε ότι η εφαρμογή είναι δίγλωσση. Οι γλώσσες που υποστηρίζονται από την εφαρμογή είναι αγγλικά και ελληνικά. Η δυνατότητα αυτή παρέχεται από την επιλογή Select Language στο αρχικό μενού. Πατώντας την συγκεκριμένη επιλογή εμφανίζεται στην οθόνη μια λίστα με της γλώσσες που υποστηρίζει η εφαρμογή. Επιλέγοντας από την λίστα κάποια από τις γλώσσες η εφαρμογή αυτόματα παρουσιάζεται στην γλώσσα επιλογής του χρήστη.

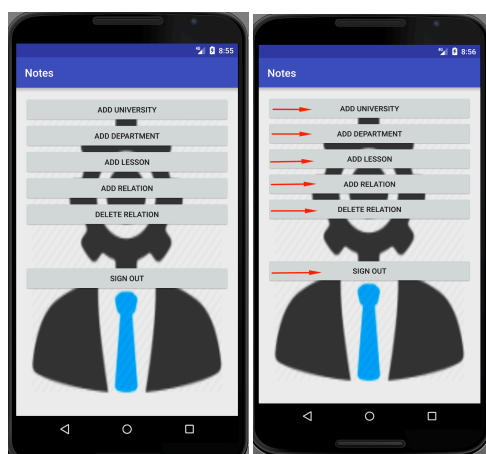


Για την Έξοδο από την εφαρμογή ο χρήστης πατάει το κουμπί SIGN OUT από το αρχικό μενού. Πατώντας το οδηγείται στην παρακάτω οθόνη όπου ξανά επιλέγει το κουμπί SIGN OUT. Με την επιτυχής έξοδο από την εφαρμογή ο χρήστης βρίσκεται ξανά στην οθόνη εισόδου.



Στην εφαρμογή επιπλέον δίνεται στους χρήστες με τον ρόλο Admin ένα διαχειριστικό με το οποίο μπορούν να παραμετροποιήσουν στοιχεία της εφαρμογής. Συνδεδεμένος ένας χρήστη που έχει δικαίωμα στην εφαρμογή ως Admin οδηγείται στην παρακάτω οθόνη. Η οθόνη αυτή αποτελεί το βασικό μενού επιλογών του διαχειριστή. Σκοπός αυτό του διαχειριστικού είναι να παραμετροποίηση τα dropdown list της εφαρμογής. Με την είσοδο του χρήστη στο διαχειριστικό εμφανίζεται το βασικό μενού επιλογών που αποτελείται από τις παρακάτω έξι επιλογές:

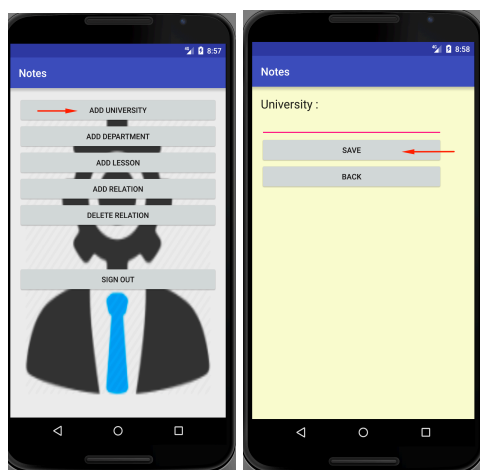
- Την εισαγωγή Πανεπιστημίου.
- Την εισαγωγή Τμήματος.
- Την εισαγωγή Μαθήματος.
- Την δημιουργία σχέσης μεταξύ των παραπάνω επιλογών.
- Την διαγραφή κάποιας υπάρχουσας σχέσης.
- Την έξοδο από το διαχειριστικό.



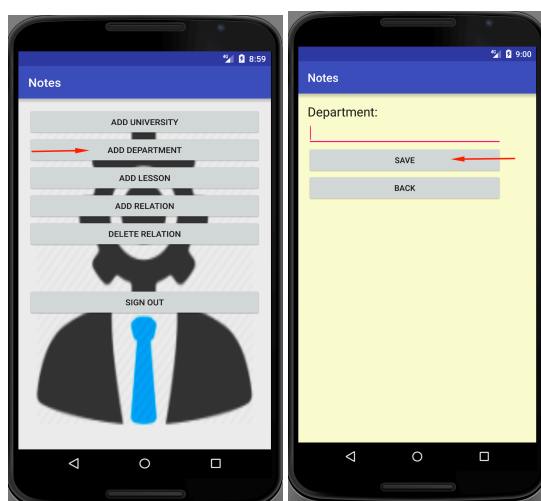
Πατώντας την πρώτη επιλογή στο βασικό μενού δηλαδή την προσθήκη ενός νέου πανεπιστημίου στα dropdown της εφαρμογής ο χρήστης μεταβαίνει στην επόμενη οθόνη. Στην συγκεκριμένη οθόνη πληκτρολογεί την ονομασία του νέου Πανεπιστημίου που θέλει να εισάγει και πατάει Αποθήκευση (Save). Με την επιτυχή ολοκλήρωση της διαδικασίας μήνυμα επιτυχής καταχώρησης εμφανίζεται στο



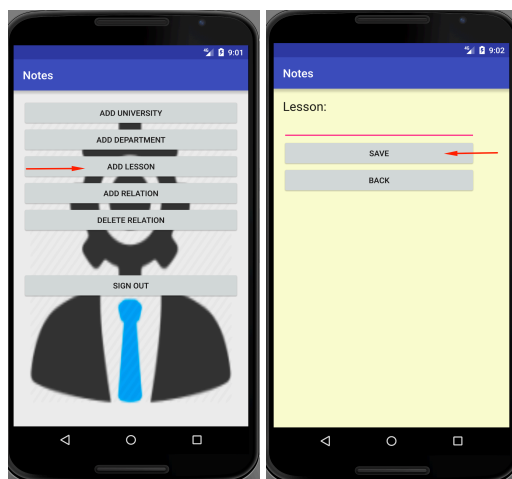
κάτω μέρος της σελίδας. Στην περίπτωση που θέλει να μεταβεί πάλι πίσω στο βασικό μενού επιλογών πατάει το κουμπί Πίσω (Back).



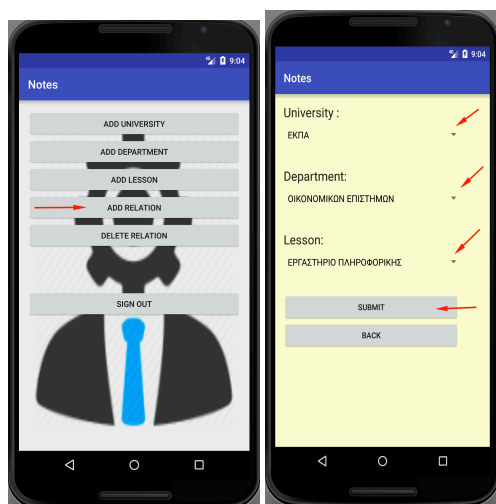
Η επομένη επιλογή στη σειρά που έχει ένας διαχειριστής είναι η δημιουργία καινούργιου τμήματος για τα dropdown list της εφαρμογής. Επιλέγοντας την ο χρήστης μεταβαίνει στην επόμενη οθόνη. Εδώ του δίνεται η δυνατότητα να καταχωρήσει ένα νέο τμήμα πληκτρολογώντας το όνομα του και πατώντας το κουμπί της αποθήκευσης. Μετά την επιτυχημένη καταχώρηση σχετικό μήνυμα εμφανίζεται στην οθόνη του χρήστη. Εάν πάλι επιθυμεί να επιστρέψει στο αρχικό μενού πατάει το κουμπί Πίσω (Back).



Η τρίτη επιλογή του χρήστη αφορά την δημιουργία νέου μαθήματος στα dropdown της εφαρμογής. Με την επιλογή αυτή ο χρήστης μεταβαίνει στην σχετική οθόνη όπου και μπορεί να εισάγει την ονομασία του νέου μαθήματος που θέλει να καταχωρήσει. Με την επιτυχή ολοκλήρωση μήνυμα εμφανίζεται στο κάτω μέρος της οθόνης. Για την επιστροφή στο αρχικό μενού επιλέγει το κουμπί Πίσω (Back).



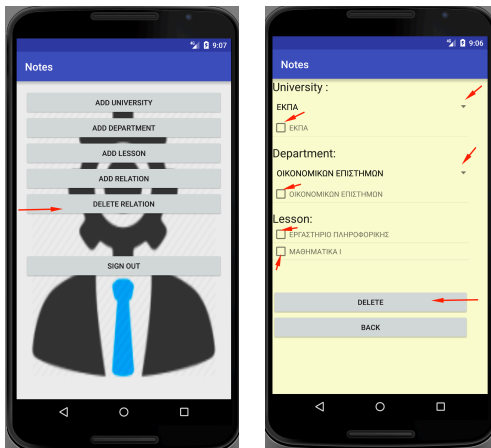
Την τέταρτη κατά σειρά επιλογή στο βασικό μενού του διαχειριστή αποτελεί η δημιουργία «συσχέτισης» μεταξύ Πανεπιστημίου, Τμήματος και Μαθήματος. Με αυτή την επιλογή ο διαχειριστής έχει την δυνατότητα να συνδυάσει τις τρεις προηγούμενες κατηγορίες που αναφέραμε έτσι ώστε να εμφανίζονται κατά αυτό τον τρόπο στην εφαρμογή. Επιλέγοντας την λοιπόν από το βασικό μενού οδηγείται στην παρακάτω οθόνη. Εκεί του δίνεται η δυνατότητα να επιλέξει αρχικά ένα από τα Πανεπιστήμια που έχει καταχωρήσει από το διαχειριστικό έως τώρα και ένα Τμήμα καθώς και ένα Μάθημα και να τα συσχετίσει μεταξύ τους επιλέγοντας την επιλογή του κουμπιού καταχώρησης. Με την επιτυχή ολοκλήρωση της διαδικασίας σχετικό μήνυμα εμφανίζεται στην οθόνη του χρήστη. Για την επιστροφή του στο βασικό μενού επιλογών και σε αυτή την οθόνη υπάρχει η επιλογή του κουμπιού Πίσω (Back)



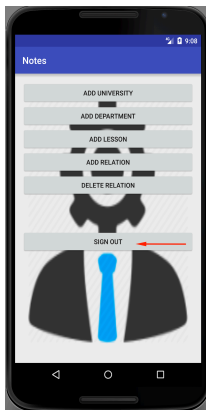
Η τελευταία επιλογή στο μενού του διαχειριστή αφορά την διαγραφή μια υπάρχουσας συσχέτισης μεταξύ Πανεπιστημίου, Τμήματος και Μαθήματος. Πατώντας την επιλογή αυτή ο χρήστης μεταφέρεται στην παρακάτω εικόνα. Σε αυτή την οθόνη ο χρήστης ανάλογα με το Πανεπιστήμιο που θα επιλέξει του εμφανίζονται οι αντίστοιχες συσχετίσεις που έχουν δημιουργηθεί για αυτό. Επιλέγοντας είτε το Πανεπιστήμιο, είτε το Τμήμα είτε τα Μαθήματα ο διαχειριστής μπορεί να



διαγράψει οποιαδήποτε συσχέτιση είχε δημιουργήσει μέχρι στιγμής. Μετά την επιλογή των εγγραφών που θέλει ο χρήστης να αφαιρέσει από την συσχέτιση για την οριστική διαγραφή του πατάει το κουμπί Διαγραφή (Delete). Με την ολοκλήρωση της διαδικασίας μήνυμα επιτυχούς ολοκλήρωσης της διαγραφής εμφανίζεται στο κάτω μέρος της οθόνης. Τέλος και στην οθόνη αυτή προσφέρεται η επιλογή του κουμπιού Πίσω που επιστρέφει τον χρήστη στο μενού των επιλογών.



Τελεύοντας με το διαχειριστικό της εφαρμογής αναφέρουμε ότι το κουμπί Έξοδος αποσυνδέει τον χρήστη από τον διαχειριστικό και από την εφαρμογή.





## 5. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ

Η πτυχιακή εργασία που εξετάζεται όπως ήδη έχει αναφερθεί και πιο πάνω ασχολείται με την λειτουργία μιας εφαρμογής για κινητές συσκευές. Σκοπός αυτής είναι η ανταλλαγή σημειώσεων ανάμεσα σε φοιτητές κάποιου εκπαιδευτικού ιδρύματος. Στην παρούσα λοιπόν ενότητα θα εξετάσουμε την συγκεκριμένη εφαρμογή όχι από την λειτουργική της αλλά από την τεχνική της σκοπιά. Με το όρο τεχνική σκοπιά αναφερόμαστε σε όλα εκείνα τα τεχνικά μέρη που συνέθεσαν την κατασκευή αυτής της εφαρμογής. Πιο συγκεκριμένα θα μπορούσαμε να αναφέρουμε πως η ανάλυση αυτής της ενότητας θα κινηθεί γύρω από τεχνικά κομμάτια όπως το λειτουργικό σύστημα, οι γλώσσες προγραμματισμού, οι βάσεις, τα εργαλεία και άλλες τεχνολογίες που χρησιμοποιήθηκαν για την κατασκευή της παρούσας εφαρμογής.

### 5.1 ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ

Αναφερόμενοι στον όρο λειτουργικό σύστημα γενικά εννοούμε ένα πρόγραμμα που λειτουργεί με την ιδιότητα του μεσάζοντα μεταξύ ενός χρήστη υπολογιστή, της συσκευής και των περιφερειακών συσκευών αυτής. Παρατηρούμε ότι στις μέρες μας χρησιμοποιείται τόσο η αγγλική συντομογραφία OS (Operation System) όσο και η πλήρης ονομασία λειτουργικό σύστημα αναφερόμενοι σε αυτό. Ως βασικούς στόχους ενός λειτουργικού συστήματος θα μπορούσαμε να αναφέρουμε διάφορους. Επιγραμματικά αναφέρουμε την ομαλή εκτέλεση των προγραμμάτων, την εύκολη αντιμετώπιση προβλημάτων για τους χρήστες, την ευκολία διαχείρισης και γενικά χρήσης ενός πληροφορικού συστήματος και τέλος την αποτελεσματική αξιοποίηση του υλικού του υπολογιστικού συστήματος. Το λειτουργικό σύστημα προσφέρει υπηρεσίες τόσο στους χρήστες όσο και στους προγραμματιστές που καθιστούν δυνατή τη λειτουργία ενός συστήματος χωρίς να είναι απαραίτητη η ανάγκη χρήσης ελέγχων χαμηλού επιπέδου που είναι δύσκολο να τρέξουν. Επιπλέον παρέχει ένα ευρύ φάσμα ομοιόμορφων διεπαφών που αλληλοεπιδρούν προκειμένου να εξασφαλίσουν την επικοινωνία μεταξύ των περιφερειακών συσκευών όπως συσκευές εισόδου/εξόδου πχ εκτυπωτές και του υπολογιστή. Επιπρόσθετα προσφέρει μια οργανωτική δομή προκειμένου οι χρήστες του να μπορούν να διαχειρίζονται διάφορους τύπους αρχείων αλλά και ένα γραφικό περιβάλλον που καθιστά την λειτουργία ενός OS εύκολη στη χρήση του. Οι τρεις πιο γνωστοί τύποι λειτουργικών συστημάτων σε επίπεδο υπολογιστών τόσο για προσωπική όσο και επαγγελματική χρήση θα μπορούσαμε να πούμε πως είναι αυτοί των Windows, Linux και Mac. Τα λειτουργικά συστήματα συνήθως παρέχουν ένα σύνολο API's, που ονομάστηκαν κλήσεις συστήματος. Ωστόσο, ασφαλέστερα λειτουργικά συστήματα ήταν αυτά που υιοθετήθηκε πρόσφατα. Αυτή τη στιγμή έχουμε παράλληλες και πολυεπίπεδες αρχιτεκτονικές ενσωματωμένες στα λειτουργικά μας συστήματα. Η εξέλιξη και η πρόοδος που έχει σημειωθεί τα τελευταία χρόνια αντιλαμβανόμαστε πόσο μεγάλη είναι. Κάτι τέτοιο αποδεικνύει και το γεγονός ότι τα λειτουργικά συστήματα στις μέρες μας με την πρόοδο της τεχνολογίας δεν έχουν εξελιχθεί μόνο σε επίπεδο υπολογιστών αλλά και σε επίπεδο κινητών συσκευών. Η όλο ένα και μεγαλύτερη ανάγκη για χρήση κινητών συσκευών καθιστούσε πλέον αναγκαία την εξέλιξη των λειτουργικών συστημάτων και σε επίπεδο κινητών συσκευών. Με τον ίδιο τρόπο που ένα λειτουργικό σύστημα επιτραπέζιων ή και φορητών υπολογιστών ελέγχει τον υπολογιστή έτσι και ένα λειτουργικό σύστημα κινητής τηλεφωνίας αποτελεί την πλατφόρμα λογισμικού πάνω από την οποία εκτελούνται προγράμματα για κινητές συσκευές όπως smartphones, PDA και υπολογιστές Tablet. Το λειτουργικό σύστημα και σε αυτή την περίπτωση των κινητών συσκευών είναι υπεύθυνο για τον προσδιορισμό των λειτουργικών δυνατοτήτων που είναι διαθέσιμες σε μια τέτοια συσκευή όπως τα πληκτρολόγια,



το WAP, οι αισθητήρες όπως οθόνες αφής, GPS, επιταχυνσιόμετρα και άλλοι, ο συγχρονισμός με τις εφαρμογές, το ηλεκτρονικό ταχυδρομείο, η ανταλλαγή μηνυμάτων, η ασφάλεια της συσκευής και πολλά άλλα. Γνωστοί τύποι λειτουργικών συστημάτων για κινητές συσκευές αποτελούν οι εξής: Android OS (Google Inc), Bada (Samsung Electronics), BlackBerry OS (Research In Motion), iPhone OS (Apple), MeeGo OS (Nokia and Intel), Palm OS (Garnet OS), Symbian OS (Nokia), WebOS (Palm/HP) και Windows Mobile (Windows Phone). Από αυτούς τρεις είναι οι πιο διαδεδομένοι τύποι λειτουργικών συστημάτων αυτή την στιγμή και αυτοί είναι το Android OS της Google, το iPhone OS της Apple και το Windows Mobile της Microsoft με τα δύο πρώτα να έχουν κατακτήσει το μεγαλύτερο μέρος της αγοράς.

Period	Android	iOS	Windows Phone	Others
2015Q4	79.6%	18.7%	1.2%	0.5%
2016Q1	83.5%	15.4%	0.8%	0.4%
2016Q2	87.6%	11.7%	0.4%	0.3%
2016Q3	86.8%	12.5%	0.3%	0.4%

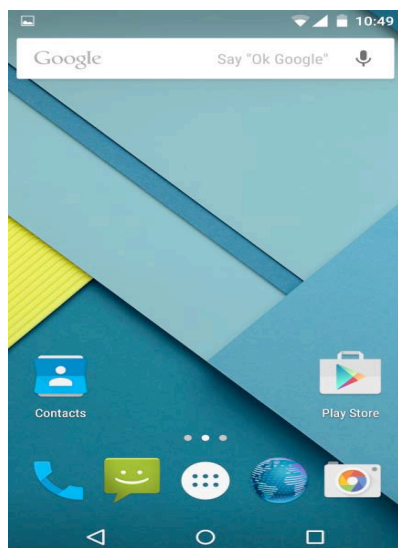
Source: IDC, Nov 2016

*Image Source: Smartphone Mobile OS Market Share, 2016 Q3 (IDC)*

Στην παρούσα λοιπόν πτυχιακή η εφαρμογή για της σημείωσης φοιτητών που κατασκευάστηκε σχεδιάστηκε έτσι ώστε να λειτουργεί στην πλατφόρμα Android για κινητές συσκευές και είναι αυτή για την οποία θα παραθέσουμε παρακάτω ορισμένες πληροφορίες σχετικές με αυτή.

## 5.2 Android OS

Το λειτουργικό σύστημα Android και οι υπηρεσίες κλήσεων βίντεο δείχνουν πώς αυτό το λειτουργικό σύστημα είναι διαφορετικό από άλλα. Βασίζεται στο λειτουργικό σύστημα Linux και μπορεί να θεωρηθεί ως παραλλαγή του ενσωματωμένου Linux. Το συγκεκριμένο λειτουργικό σύστημα δημιουργήθηκε από την εταιρεία Google Inc και θα μπορούσαμε να πούμε πως αυτή την στιγμή είναι πρωτοπόρος ως πλατφόρμα για κινητές συσκευές. Ο λόγος που αυτή η άποψη έχει επικρατήσει είναι ότι παρέχει πλήθος χαρακτηριστικών που αρμονικά λειτουργούν μεταξύ τους ώστε να εξασφαλίζουν αύξηση στην ταχύτητα και στην απόδοση της κινητής συσκευής. Κάτι τέτοιο γίνεται ήδη αντιληπτό από την έκδοση 5 και μετά του Android που έφερε πλούσια χαρακτηριστικά σχεδιασμένα ώστε να παρέχει μια πλήρως βελτιωμένη εμπειρία κινητής τηλεφωνίας.



The 5.0 Android (Lollipop) operating system

Το Android είναι μια πλατφόρμα αλλά και ένα λειτουργικό σύστημα για κινητά τηλέφωνα και άλλες συσκευές. Ο αρχικός του προορισμός ήταν να τρέξει για οθόνες αφής κινητών συσκευών αλλά εξαπλώθηκε και σε άλλες συσκευές όπως τα κινητά τηλέφωνα, οι τηλεοράσεις, τα αυτοκίνητα και τα ρολόγια. Συνεπώς αντιλαμβανόμαστε πόσο μεγάλος είναι πλέον ο αριθμός των συσκευών που σήμερα χρησιμοποιούν Android. Ο λόγος αυτής της τάσης προς το Android δεν αποτελεί άλλον από το ότι μετέτρεψε τις κινητές συσκευές σε προσωπικό υπολογιστή τσέπης. Αναπτύχθηκε από τον Andy Rubino διευθυντής των κινητών πλατφορμών για την Google. Το Android είναι ένα λειτουργικό σύστημα ανοιχτού κώδικα (Η χρήση του όρου Open source ορίζει τον πηγαίο κώδικα για οποιαδήποτε εφαρμογή που αναπτύσσεται για πώληση ή κέρδος με προσανατολισμό σκοπό. Αυτό εξασφαλίζει επίσης ότι χωρίς άδειες επί πληρωμή, χρήστες και προγραμματιστές χρησιμοποιούν τον πηγαίο κώδικα, ακολουθώντας τους κανόνες και τους όρους χρήσης που προϋποθέτει η άδεια). Με άλλα λόγια οποιοσδήποτε χρήστης μπορεί να φέρει βελτιώσεις στο λειτουργικό σύστημα, προσθέτοντας επεκτάσεις χωρίς να είναι διαθέσιμες στην κοινότητα ανοιχτού κώδικα. Ωστόσο κάποια από τα API's και τις βιβλιοθήκες του συστήματος δεν είναι πλήρως ανοικτά γεγονός για το οποίο επικρίθηκε. Η πιο πρόσφατη έκδοση του Android είναι η 8.0 γνωστή με την ονομασία Oreo αναπτύχθηκε από την Google και κυκλοφόρησε τον Αύγουστο του 2017. Η λίστα με τις εκδόσεις android ξεκινάει με την είσοδο της έκδοσης Android 1.0 τον Νοέμβριο του 2007 και υπάρχει μέχρι σήμερα ένας μεγάλος αριθμός αναβαθμίσεων από αυτή την αρχική κυκλοφορία του. Τα ονόματα αυτών των εκδόσεων είναι εμπνευσμένα από τις ονομασίες γνωστών επιδορπίων. Επιγραμματικά αναφέρονται αυτές οι εκδόσεις παρακάτω :

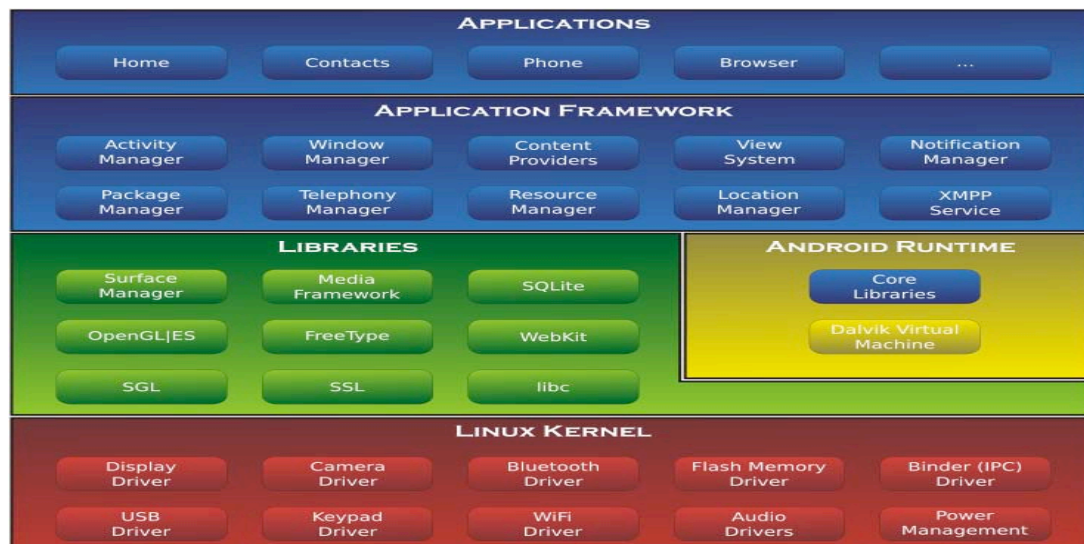
- Alpha (1.0)
- Beta (1.1)
- Cupcake (1.5)
- Donut (1.6)
- Iclair (2.0-2.1)
- Frozen Yogurt "Froyo" (2.2-2.2.3)
- Gingerbread (2.3-2.3.7)
- Honeycomb (3.0-3.2.6)
- Ice Cream Sandwich (4.0-4.0.4)





- Jelly Bean (4.1-4.3.1)
- KitKat (4.4-4.4.4, 4.4W-4.4W.2)
- Lollipop (5.0-5.1)
- Marshmallow (6.0-6.0.1)
- Nougat (7.0-7.1.1)
- Oreo (8.0)

Από την αρχή μέχρι την έκδοση 5.0, η πλατφόρμα Android χρησιμοποίησε μια εικονική μηχανή που ονομάζεται Dalvik ως διαδικασία just-in-time (JIT) για την εκτέλεση του κώδικα Java. Ο Dalvik μεταγλωττίζει τις εφαρμογές κάθε φορά που αυτές εκκινούνται. Παρακάτω παρατίθεται η αρχιτεκτονική δομή του Android μέχρι και την έκδοση 5.0.



The architecture of the Android operating system with Dalvik virtual machine

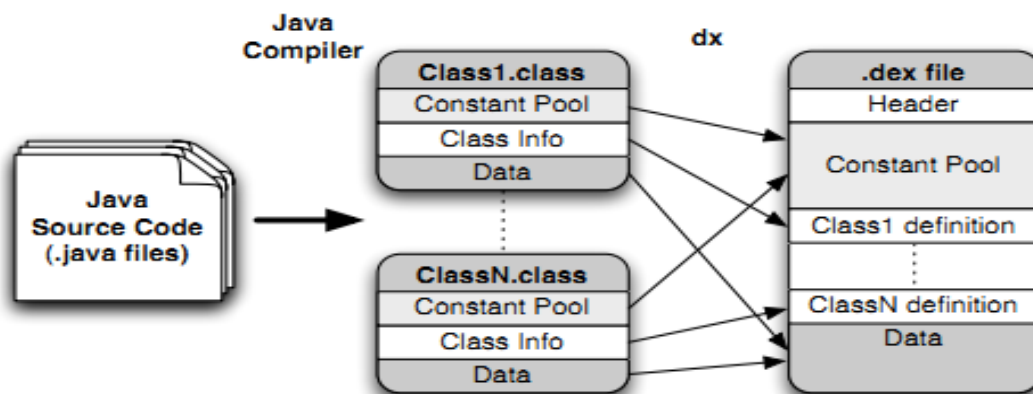
Συμπερασματικά από την παραπάνω εικόνα παρατηρούμε ότι το Android OS αποτελείται από ένα Linux Kernel, ένα περιβάλλον runtime, ένα πλαίσιο ανάπτυξης, βιβλιοθήκες και βασικές εφαρμογές βασισμένα σε Java. Επιπρόσθετα έχει την δικιά του εικονική μηχανή της Google γνωστή με την ονομασία Dalvik όπως αναφέραμε και πιο πάνω και το λειτουργικό σύστημα Linux αποτελεί μόνο ένα μέρος αυτής της αρχιτεκτονική και μάλιστα τοποθετείται στο τελευταίο επίπεδο αυτής δίνοντας πρόσβαση μόνο σε ορισμένους χρήστες. Όσον αφορά τα επίπεδα εφαρμογών και το περιβάλλον ανάπτυξης είναι γραμμένα σε γλώσσα Java. Ωστόσο οι προγραμματιστές εφαρμογών περιορίζονται μόνο στο επίπεδο εφαρμογών ενώ οι κατασκευαστές συσκευών σε όλα τα επίπεδα αυτής της αρχιτεκτονικής. Αναλυτικότερα ξεκινώντας από το επίπεδο εφαρμογών αποτελεί το κορυφαίο επίπεδο της αρχιτεκτονικής Android και είναι αυτό στο οποίο αποθηκεύονται όλες οι εγκατεστημένες εφαρμογές της συσκευής. Ως επί το πλείστον το συγκεκριμένο επίπεδο περιλαμβάνει native εφαρμογές όπως το Google maps, η κάμερα, προγράμματα περιήγησης στο διαδίκτυο (browsers) και άλλες πολλές. Οι εφαρμογές αυτές εκτελούνται από το επίπεδο ανάπτυξης (Application Framework) όταν οι τελικοί χρήστες επιθυμούν να ξεκινήσει η λειτουργία τους. Εν συνεχεία αναφέρεται το επίπεδο ανάπτυξης της αρχιτεκτονικής Android. Σε αυτό το επίπεδο περιλαμβάνονται διάφορα πακέτα υπηρεσιών (Services) αλλά και όλες οι κλάσεις που χρειάζονται οι προγραμματιστές για την υλοποίηση μιας εφαρμογής σε λειτουργικό σύστημα Android. Παρ' όλα αυτά το συγκεκριμένο επίπεδο παρέχει στους προγραμματιστές επιπλέον την δυνατότητα χρήσης οποιασδήποτε ήδη υλοποιημένης



εφαρμογής που περιλαμβάνεται στο πιο πάνω επίπεδο της αρχιτεκτονικής δηλαδή αυτό των εφαρμογών. Τελειώνοντας την σύντομη αναφορά μας στο επίπεδο αυτό αναφέρουμε πώς περιλαμβάνει την απαραίτητη γκάμα «διαχειριστικών» που είναι απαραίτητη για την πρόσβαση και την προσπέλαση των δεδομένων. Παρακάτω παρουσιάζονται και επιγραμματικά αναφέρονται σε μια λίστα οι αναφερόμενες ως «διαχειριστικές» υπηρεσίες που προσφέρονται από το επίπεδο αυτό.

- **Activity Manager:** Ελέγχει και διαχειρίζεται όλες τις δραστηριότητες και ασχολείται με τον κύκλο ζωής των εφαρμογών.
- **Resource Manager:** Οποιαδήποτε πρόσβαση σε πόρους όπως για παράδειγμα γραφικά, εκτός αυτών του κώδικα διαχειρίζονται από τον Resource Manager.
- **Notification Manager:** Είναι αυτός που διαχειρίζεται και επιτρέπει να εμφανίζονται ειδοποιήσεις από τις εφαρμογές.
- **Location Manager:** Διαχειρίζεται τις αλλαγές που γίνονται όσον αφορά την γεωγραφική θέση του τελικού χρήστη και παρέχει ειδοποιήσεις σχετικά με αυτή.
- **Package Manager:** Όλα τα δεδομένα που είναι εγκατεστημένα στη συσκευή ανακτώνται μέσω του συγκεκριμένου διαχειριστή.
- **Telephony Manager:** Οι ρυθμίσεις σύνδεσης δικτύου αλλά και όλες οι απαραίτητες πληροφορίες για τις υπηρεσίες διαχειρίζονται από τον Telephony Manager.
- **Window Manager:** Όλες οι «οθόνες» αλλά και η χωροταξική τους διάταξη αναλαμβάνονται από τον Window Manager.
- **Content Provider:** Είναι αυτός που είναι υπεύθυνος να παρέχει την απαιτούμενη πρόσβαση στις εφαρμογές για τη χρήση άλλων εφαρμογών μέσω αυτών.

Το παρακάτω επίπεδο αρχιτεκτονικής περιλαμβάνει το περιβάλλον runtime αλλά και όλες τις απαραίτητες βιβλιοθήκες. Όσον αφορά το runtime περιβάλλον είναι αυτό που είναι υπεύθυνο για την εκτέλεση όλων των εφαρμογών. Μέσα σε αυτό το περιβάλλον διατίθεται ένα VM με την ονομασία Dalvik Virtual Machine όπως ήδη έχουμε αναφέρει. Σε αυτή λοιπόν την εικονική μηχανή εκτελούνται ταυτόχρονα όλες οι εφαρμογές Android και πιο συγκεκριμένα αρχεία της μορφής .dexff



#### Dalvik Virtual Machine Process

Από την έκδοση 4.4 KitKat όπου πρωτοεμφανίστηκε το περιβάλλον runtime έχει αντικαταστήσει πλήρως την εικονική συσκευή Dalvik εξασφαλίζοντας την απαραίτητη συμβατότητα που απαιτούνταν για την εκτέλεση εφαρμογών προηγούμενων εκδόσεων Android βασισμένες στο VM Dalvik παρέχοντας επιπλέον πλεονεκτήματα όσον αφορά την απόδοση. Όσον αφορά τώρα τις βιβλιοθήκες



που περιλαμβάνει το επίπεδο αυτό είναι γραμμένες σε C και C++ γλώσσες προγραμματισμού. Οι συγκεκριμένες βιβλιοθήκες δεν είναι άμεσα προσπελάσιμες παρά μόνο μέσω του επιπέδου ανάπτυξης (Application Framework). Τέτοιες βιβλιοθήκες είναι οι βιβλιοθήκες ιστού για πρόσβαση στο διαδίκτυο, προγράμματα περιήγησης, βιβλιοθήκες για μορφές ήχου και βίντεο. Φτάνοντας στο χαμηλότερο επίπεδο αυτής της αρχιτεκτονικής συναντάμε τον Linux Kernel. Αυτό το στρώμα της αρχιτεκτονικής έχει ζωτική σημασία για αυτήν. Αυτό συμπεραίνεται από το γεγονός ότι ο Linux Kernel παρέχει μια σταθερότητα και ένα λειτουργικό σύστημα για κινητές συσκευές προσφέροντας παράλληλα τόσο υπηρεσίες όπως την διαχείριση μνήμης, ενέργειας και ασφάλειας όσο και ένα αρμονικά συνδεδεμένο περιβάλλον στο οποίο επιτυγχάνεται με τον καλύτερο δυνατό τρόπο η επικοινωνία του λογισμικού με το υλικό της εκάστοτε συσκευής. Μέσα από την πάροδο του χρόνου το λειτουργικό σύστημα Linux εξελίχθηκε και πολλές από τις εξελίξεις και τα προνόμια που αναπτύχθηκαν για αυτό έχουν περάσει πλέον και στο λειτουργικό σύστημα Android. Μια από τις κύριες εξελίξεις που ενσωματώθηκε στο λειτουργικό σύστημα Android μέσω του Linux Kernel είναι και αυτή στο χώρο της ασφάλειας. Επιγραμματικά αναφέρουμε κάποια από τα χαρακτηριστικά ασφάλειας που συμπεριλαμβάνει πλέον το Android OS:

1. Ένα μοντέλο δικαιωμάτων που καθορίζεται από τον τελικό χρήστη.
2. Απομόνωση των διαδικασιών .
3. Το λειτουργικό σύστημα επιτρέπει στις διαδικασίες να διαχειρίζονται κοινά δεδομένα με ασφαλή τρόπο.
4. Η ικανότητα απομόνωσης του περιττού και μη ασφαλούς μέρος του πυρήνα.
5. Την δυνατότητα διαχωρισμού και διαχείρισης των πόρων όταν το λειτουργικό σύστημα χρησιμοποιείται από πολλαπλούς χρήστες. Με αλλά λόγια δεν επιτρέπει τη πρόσβαση ενός χρήστη στα δεδομένα, στο χώρο αποθήκευσης, στους πόρους CPU, στο GPS και σε αλλά που ανήκουν σε άλλο χρήστη.

Κλείνοντας τα όσα ειπώθηκαν για το λειτουργικό σύστημα Android αξίζει να αναφέρουμε κάποια στοιχεία σχετικά με τη δομή του συγκεκριμένου λειτουργικού. Το Android θα μπορούσαμε εν συντομία να πούμε πως αποτελείται από δύο βασικές «οντότητες» τα Intents και τα Components. Τα Intents είναι μηνύματα που περιέχουν πληροφορίες σχετικά με ένα παραλήπτη αλλά ταυτόχρονα έχουν την δυνατότητα να μεταφέρουν και δεδομένα. Συνήθως χρησιμοποιούνται για να επιτύχουν την επικοινωνία μιας εφαρμογής και του λειτουργικού συστήματος και αποκαλείται για την λειτουργία του αυτή και Broadcast Intent. Τα Intents κατηγοριοποιούνται σε δύο είδη. Το ένα είδος αποτελούν τα Explicit Intent και το άλλο τα Implicit Intent. Οι διαφορές μεταξύ των δύο κατηγοριών είναι ότι η πρώτη ανιχνεύει τον παραλήπτη του μηνύματος από το όνομα τού ενώ το δεύτερο είδος αφήνει το λειτουργικό σύστημα να αποφασίσει σε ποια εφαρμογή θα το στείλει. Όσον αφορά τώρα τα Components αποτελούν μέρη του λειτουργικού συστήματος κυρίως γραμμένα σε γλώσσα προγραμματισμού Java. Παρακάτω αναφέρουμε επιγραμματικά τα είδη Component που υπάρχουν στο λειτουργικό σύστημα αυτό και βρίσκονται δηλωμένα μέσα σε κάθε εφαρμογή Android μέσα στο αρχείο Manifest κατά την συγγραφεί τους από τους προγραμματιστές.

- **Activity:** Είναι μια ορατή διαδικασία που παρέχεται στην οθόνη, λειτουργεί στο παρασκήνιο και αλληλοεπιδρά με τον χρήστη μέσω του User Interface. Όσον αφορά την παρούσα πτυχιακή εργασία έχουν κατασκευαστεί και αναφέρονται επιγραμματικά σε αυτή την παράγραφο τα εξής Activities:
  1. **BaseActivity:** Περιέχει βασικές λειτουργίες που χρησιμοποιούνται μέσα σε όλη την εφαρμογή.
  2. **LoginActivity:** Περιέχει την απαραίτητη λειτουργικότητα για το Authentication ενός user.

















































3. **MainActivity:** Αποτελείται από την λειτουργικότητα της βασικής οθόνης επιλόγων της εφαρμογής.
  4. **MainActivity1:** Περιέχει την λειτουργικότητα που είναι απαραίτητη για την εισαγωγή μιας νέας σημείωσης.
  5. **MainActivity2:** Περιέχει την λειτουργικότητα για την εμφάνιση όλων των αποθηκευμένων σημειώσεων του εκάστοτε χρήστη.
  6. **MainActivity3:** Περιέχει την λειτουργικότητα για την προβολή και επεξεργασία μίας αποθηκευμένης σημείωσης του χρήστη.
  7. **MainActivity4:** Περιέχει την λειτουργικότητα που χρειάζεται για να εμφανιστεί η λίστα με τις κατηγορίες που έχει δημιουργήσει ο χρήστης προκειμένου να κατηγοριοποιήσει τις σημειώσεις του.
  8. **MainActivity5:** Περιέχει την απαραίτητη λειτουργικότητα για την δημιουργία μιας νέας κατηγορίας σημειώσεων.
  9. **MainActivity6:** Περιέχει την λειτουργικότητα για την εμφάνιση των σημειώσεων ανάλογα με την κατηγορία που έχει επιλεγεί.
  10. **MainActivity7:** Περιλαμβάνει την λειτουργικότητα για την εμφάνιση όλων των σημειώσεων που υπάρχουν στο «σύννεφο».
  11. **MainActivity8:** Περιέχει την λειτουργικότητα για την προβολή και αποθήκευση μιας σημείωσης από το σύννεφο τοπικά στην συσκευή του χρήστη.
  12. **ImageViewerActivity:** Περιέχει την απαραίτητη λειτουργικότητα για την προβολή των εικόνων που έχουν τραβηχτεί στην εφαρμογή από τον χρήστη.
  13. **ImageViewerActivity3:** Περιέχει την απαραίτητη λειτουργικότητα για την προβολή των εικόνων που έχουν τραβηχτεί στην εφαρμογή από τον χρήστη.
  14. **SubscribeActivity:** Περιέχει την απαραίτητη λειτουργικότητα που απαιτείται για να εγγραφεί και λαμβάνει ένας χρήστης ειδοποιήσεις σχετικά με σημειώσεις μαθημάτων που τον αφορούν.
  15. **AdminActivity:** Περιλαμβάνει την λειτουργικότητα που αφορά τις επιλογές που παρέχει το βασικό μενού στο διαχειριστικό της εφαρμογής.
  16. **AddDepartmentActivity:** Το Activity αυτό περιέχει την λειτουργικότητα για την εισαγωγή ενός νέου τμήματος από το διαχειριστικό του χρήστη με ρόλο admin.
  17. **AddLessonActivity:** Το συγκεκριμένο Activity περιέχει την λειτουργικότητα για την εισαγωγή ενός νέου μαθήματος στα dropdown list της εφαρμογής από το διαχειριστικό.
  18. **AddUniversityActivity:** Περιλαμβάνει την λειτουργικότητα για την εισαγωγή ενός νέου πανεπιστημίου στις λίστες της εφαρμογής.
  19. **AddRelationActivity:** Περιέχει την λειτουργικότητα για την δημιουργία μιας συσχέτισης μεταξύ Πανεπιστημίου, Τμήματος και Μαθήματος που θα εμφανίζεται στις λίστες καταχώρησης της εφαρμογής.
  20. **DeleteRelationActivity:** Το Activity αυτό αποτελείται από την απαραίτητη λειτουργικότητα που απαιτείται για την διαγραφή μιας ήδη υπάρχουσας συσχέτισης μεταξύ Πανεπιστημίου, Τμήματος και Μαθήματος που έχει δημιουργηθεί κάνοντας χρήση του διαχειριστικού εργαλείου.
- **Service:** Είναι μια αόρατη στο χρήστη διαδικασία που τρέχει στο παρασκήνιο χωρίς να γίνεται χρήση της οθόνης της συσκευής. Στην παρούσα πτυχιακή εργασία έχει κατασκευαστεί το εξής Service:
    1. **MyFirebaseMessagingService:** Είναι υπεύθυνο για την αποστολή και την εμφάνιση όλων των ειδοποιήσεων που εμφανίζει η εφαρμογή των σημειώσεων.



- **Broadcast Receiver:** Είναι ένα Component που είναι υπεύθυνο να λαμβάνει τα Broadcast Intents που στέλνει το λειτουργικό σύστημα
- **Content Provider:** Αποτελεί ένα Component που λειτουργεί σαν βάση δεδομένων και διαχειρίζεται όλες τις απαραίτητες πληροφορίες μεταξύ των εφαρμογών.

### 5.3 ΓΛΩΣΣΑ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Με τον όρο γλώσσα προγραμματισμού αναφερόμαστε σε μια τεχνητή γλώσσα που μπορεί να χρησιμοποιηθεί για τον έλεγχο μιας μηχανής, συνήθως ενός υπολογιστή. Οι γλώσσες προγραμματισμού (όπως άλλωστε και οι ανθρώπινες γλώσσες) ορίζονται από ένα σύνολο συντακτικών και εννοιολογικών κανόνων, που ορίζουν τη δομή και το νόημα, αντίστοιχα, των προτάσεων της γλώσσας. Οι γλώσσες προγραμματισμού χρησιμοποιούνται για να διευκολύνουν την οργάνωση και διαχείριση πληροφοριών, αλλά και για την ακριβή διατύπωση αλγορίθμων. Μέχρι σήμερα παρατηρείται ότι οι γλώσσες προγραμματισμού με το πέρασμα του χρόνου ολοένα και αυξάνονται. Παρακάτω παρατίθεται μια λίστα με τις εικοσιένα επικρατέστερες γλώσσες προγραμματισμού για το έτος 2017 σύμφωνα με το επιστημονικό περιοδικό IEEE.

Language Rank	Types	Spectrum Ranking
1. Python	 	100.0
2. C	  	99.7
3. Java	  	99.4
4. C++	  	97.2
5. C#	  	88.6
6. R	 	88.1
7. JavaScript	 	85.5
8. PHP	 	81.4
9. Go	  	76.1
10. Swift	 	75.3
11. Arduino		73.0
12. Ruby	 	72.4
13. Assembly		72.1
14. Scala	 	68.3
15. Matlab	 	68.0
16. HTML	 	67.0
17. Shell	 	66.3
18. Perl	 	57.6
19. Visual Basic	 	55.4
20. Cuda	 	53.9
21. Lua	  	52.7

<https://spectrum.ieee.org/static/interactive-the-top-programming-languages-2017>

Στην παρούσα πτυχιακή εργασία η γλώσσα με την οποία «χτίστηκε» η εφαρμογή είναι η γλώσσα προγραμματισμού Java. Η επιλογή της συγκεκριμένης γλώσσα δεν αποτελεί τυχαία επιλογή αλλά σχετίζεται άμεσα με το γεγονός ότι η εφαρμογή υλοποιήθηκε για κινητές συσκευές με σκοπό να εκτελείτε σε λειτουργικό σύστημα Android. Η γλώσσα Java είναι μια ευρέως διαδεδομένη γλώσσα προγραμματισμού σύμφωνα και με τις παραπάνω τελευταίες δημοσκοπήσεις από τον χώρο της επιστήμης των υπολογιστών. Η Java ξεκίνησε να υπάρχει σαν πρώτη έκδοση το 1991 και δημιουργήθηκε από τον James Gosling επικεφαλής μιας ομάδας της Sun Microsystems. Η sun χρησιμοποίησε UNIX για τους υπολογιστές της οπότε οι προγραμματιστές της βάσισαν την νέα γλώσσα σε C++. Ωστόσο η συγκεκριμένη επιλογή τους σε σχέση με το ότι η γλώσσα προγραμματισμού που βασίστηκαν ήταν η C++ και όχι η C δεν ήταν καθόλου τυχαία. Σκοπός τους ήταν η νέα γλώσσα που





ανέπτυξαν να ήταν αντικειμενοστραφής. Το αρχικό όνομα της γλώσσας ήταν Oak. Ωστόσο σύντομα κατάλαβαν πώς υπήρχε ήδη υλοποιημένη γλώσσα με αυτό το όνομα, έτσι άλλαξαν το όνομα της σε Java. Η πρώτη έκδοση της Java σχεδιάστηκε για τον προγραμματισμό οικιακών συσκευών όπως πλυντήρια και τηλεοράσεις και η κωδική ονομασία που δόθηκε για το εγχείρημα αυτό ήταν Green. Παρότι το συγκεκριμένο πεδίο δεν είχε ιδιαίτερη αίγλη ο σχεδιασμός μιας γλώσσας για το σκοπό αυτό δεν ήταν καθόλου εύκολος. Ο λόγος που ένα τέτοιο εγχείρημα αποτελούσε σχεδόν ακατόρθωτο ήταν ότι απευθυνόταν σε συσκευές με μεγάλο φάσμα μικροεπεξεργαστών. Επίσης κάτι ακόμα που δυσκόλευε ακόμα περισσότερο την επίτευξη αυτού του σκοπού ήταν και το γεγονός ότι απευθυνόταν σε συσκευές χαμηλού κόστους κάτι που σήμαινε πώς οι κατασκευαστές δεν ήταν πρόθυμοι να δαπανήσουν μεγάλα χρηματικά ποσά για την ανάπτυξη πολύπλοκων μεταγλωττιστών. (Ο μεταγλωττιστής είναι ένα πρόγραμμα που μεταφράζει το πρόγραμμα σε μια γλώσσα κατανοητή από τον μικροεπεξεργαστή.) Για την απλοποίηση του σκοπού αυτού οι προγραμματιστές της Sun αποφάσισαν να χρησιμοποιήσουν μια διαδικασία μετάφρασης δύο βημάτων. Η ιδέα τους αυτή υλοποιήθηκε μεταφράζοντας το πρόγραμμα αρχικά σε μια ενδιάμεση γλώσσα που ήταν κοινός σε όλες τις ηλεκτρικές συσκευές και έπειτα σε ένα εύκολο στην συγγραφή και συνεπώς οικονομικό πρόγραμμα για την εκάστοτε ηλεκτρική συσκευή ή υπολογιστή. Η ενδιάμεση αυτή γλώσσα ονομάζεται κώδικας-byte της java ή απλά κώδικας-byte. Το 1994 ο Patrick Naughton και ο Jonathan Payne στη Sun Microsystems ανέπτυξαν ένα περιηγητή ιστού που μπορούσε να τρέξει προγράμματα σε Java στο διαδίκτυο γνωστό με την ονομασία HotJava. Στην πορεία μέχρι σήμερα ακολούθησαν και άλλες εταιρίες που ανέπτυξαν λογισμικό που εκτελούσε προγράμματα σε Java. Συμπερασματικά από την παραπάνω αναδρομή η γλώσσα για τον προγραμματισμό των οικιακών συσκευών δεν έγινε τότε ιδιαίτερα γνωστή στους κατασκευαστές αλλά η γλώσσα Java στην οποία εξελίχθηκε χρησιμοποιείται ευρέως ως γλώσσα προγραμματισμού. Όπως ήδη αναφέρθηκε και πιο πάνω η γλώσσα Java είναι μια αντικειμενοστραφής γλώσσα ωστόσο το ερώτημα που γεννάται είναι τι θα πει ότι μια γλώσσα προγραμματισμού είναι αντικειμενοστραφής. Ο κόσμος γύρω μας αποτελείται από διάφορα αντικείμενα όπως είναι οι άνθρωποι, τα αυτοκίνητα, τα κινητά, οι μηχανές και άλλα. Κάθε ένα από αυτά τα αντικείμενα έχει την δυνατότητα να παράγει κάποια ή και κάποιων αριθμό ενεργειών που επιφέρουν κάποιο αποτέλεσμα σε κάποια αντικείμενα του ίδιου κόσμου. Ο αντικειμενοστραφής προγραμματισμός βασίστηκε λοιπόν στην γενική ιδέα μίας τέτοιας αλληλεπίδρασης αντικειμένων θεωρώντας πώς ένα πρόγραμμα όπως και στον κόσμο γύρω μας αποτελείται αντίστοιχα από αντικείμενα που αλληλοεπιδρούν μεταξύ τους μέσω ενεργειών. Βασιζόμενοι λοιπόν στην ιδέα αυτή ο αντικειμενοστραφής προγραμματισμός απέκτησε δικιά του ορολογία και δομή που αποτελείται από αντικείμενα και μεθόδους, θεωρώντας μάλιστα ότι αντικείμενα του ίδιου είδους κατηγοριοποιούνται μέσα στην ίδια ομάδα ή όπως ορολογιακά αναφέρεται στην Java ανήκουν στην ίδια κλάση. Τελειώνοντας την αναφορά μας στην γλώσσα προγραμματισμού Java δεν θα μπορούσαμε να μην αναφερθούμε στα δύο βασικά είδη εφαρμογών που παράγονται κάνοντας χρήση της συγκεκριμένης γλώσσας. Τα προγράμματα που παράγονται λοιπόν ύστερα από την υλοποίηση του με κώδικα Java κατηγοριοποιούνται σε δύο είδη αυτά των μικροεφαρμογών (applets) και σε αυτά των εφαρμογών (applications). Οι μικροεφαρμογές με τις εφαρμογές είναι σχεδόν πανομοιότυπες. Η διαφορά που υπάρχει αναμεσά τους και τις διαφοροποιεί ωστόσο είναι ότι, οι εφαρμογές είναι σχεδιασμένες για να τρέχουν σε υπολογιστή όπως οποιοδήποτε πρόγραμμα ενώ οι μικροεφαρμογές εκτελούνται μόνο μέσω κάποιου προγράμματος περιήγησης ιστού (Web browser) ή με άλλα λόγια κάποιου φυλλομετρητή ιστού (Web browser).

Σε ότι αφορά τώρα την παρούσα πτυχιακή εργασία όπως ήδη έχει αναφερθεί είναι υλοποιημένη σε γλώσσα Java, για να τρέχει σε κινητές συσκευές και αφορά μια εφαρμογή για σημειώσεις φοιτητών ενός εκπαιδευτικού ιδρύματος. Αυτό που δεν έχει αναφερθεί όμως έως τώρα και επ'ευκαιρία μετά την αναφορά μας στην java είναι τι ακριβώς λειτουργικότητα προσφέρει αυτός ο κώδικας στην εφαρμογή και σε ποια αρχεία είναι αυτός υλοποιημένος. Σε μια Android εφαρμογή τα αρχεία που



περιλαμβάνουν τον εκτελέσιμο κώδικα σε Java ονομάζονται Activities όπως έχουμε αναφέρει και πιο πάνω στην παρούσα ενότητα. Τα Activities λοιπόν που αναπτύχθηκαν για την εξυπηρέτηση της λειτουργικότητας της παρούσας εφαρμογής σημειώσεων παρουσιάζονται και αναλύονται διεξοδικότερα παρακάτω.

- 1. BaseActivity:** Το συγκεκριμένο Activity περιέχει βασικές λειτουργίες που χρησιμοποιούνται μέσα στην εφαρμογή. Περιλαμβάνει δύο βασικές συναρτήσεις που απαιτούνται από την εφαρμογή για την εμφάνιση της «μπάρας προόδου» κατά το Authentication του user με ηλεκτρονικό λογαριασμό Gmail κατά την είσοδο και έξοδο του από την εφαρμογή. Πιο συγκεκριμένα οι συναρτήσεις αυτές είναι η ShowProgressDialog που είναι υπεύθυνη όταν καλείται μέσα στον κώδικα για την εμφάνιση της μπάρας προόδου και η hideProgressDialog που κρύβει τη μπάρα προόδου.
- 2. LoginActivity:** Περιέχει την απαραίτητη λειτουργικότητα για το Authentication ενός user. Γενικότερα γίνεται χρήση μιας τεχνολογίας με την ονομασία Firebase και πιο συγκεκριμένα στο αρχείο αυτό γίνεται χρήση της λειτουργικότητας για authentication ενός χρήστη που θα εξηγήσουμε παρακάτω στην ενότητα αυτή. Με την κλήση του συστήματος για την συγκεκριμένη οθόνη ο κώδικας που τρέχει στο παρασκήνιο αρχικά αρχικοποιεί τα απαραίτητα για την οθόνη γραφικά στοιχεία. Εν συνεχεία κάνει της απαραίτητες ρυθμίσεις (configurations) που απαιτούνται από την κλήση προς το Api της Google που εξασφαλίζει το authentication με ηλεκτρονικό λογαριασμό Gmail. Ύστερα καλεί την τεχνολογία Firebase προκειμένου να αποφασίσει εάν ο χρήστης είναι αρχικά καταχωρημένος στην εφαρμογή και εν συνεχεία εξετάζει αν είναι ήδη συνδεδεμένος σε αυτή ή όχι. Ανάλογα με την κατάσταση του χρήστη που λαμβάνει από το Firebase ενημερώνει τα απαραίτητα γραφικά στοιχεία της οθόνης. Στην περίπτωση τώρα που χρήστης δεν είναι συνδεδεμένος στην εφαρμογή και από την στιγμή που πατήσει το κουμπί Sing In για να συνδεθεί σε αυτή ξεκινά η διαδικασία καταχώρησης και εισόδου του χρήστη στην εφαρμογή. Ειδικότερα περνάει όλη την απαραίτητη πληροφορία που έχει καταχωρήσει ο χρήστης μέσω του Api της Google στην τεχνολογία Firebase και μέσω της λειτουργίας αυτής για authentication εκ κινούνται οι απαραίτητες διαδικασίες για την καταχώρηση και είσοδο του στην εφαρμογή. Τώρα όσον αφορά την περίπτωση που ο χρήστης επιθυμεί να κάνει έξοδο από την εφαρμογή μέσω πάλι της τεχνολογία Firebase για authentication ενημερώνεται η κατάσταση του τρέχον λογαριασμού του χρήστη ως αποσυνδεδεμένος.
- 3. MainActivity:** Το συγκεκριμένο Activity αποτελείται από την λειτουργικότητα της βασικής οθόνης επιλόγων της εφαρμογής. Μετά την επιτυχής είσοδο ενός χρήστη στην εφαρμογή είναι η πρώτη οθόνη που του εμφανίζεται καθώς αποτελεί το βασικό μενού επιλογών. Στο κάλεσμα της συγκεκριμένης λοιπόν οθόνης ο κώδικας στο παρασκήνιο αρχικά ελέγχει αν υπάρχει βάση δεδομένων τοπικά στην εκάστοτε συσκευή. Εάν αυτή δεν υπάρχει κατασκευάζει τους απαραίτητους πίνακες για την σωστή λειτουργία της εφαρμογής. Για την κατασκευή της τοπικής βάσης δεδομένων γίνεται χρήση της SQLite που θα περιγράψουμε παρακάτω στην ενότητα αυτή. Εν συνεχεία αρχικοποιεί όλα τα απαραίτητα γραφικά στοιχεία της οθόνη αυτής ώστε να εμφανιστούν στο χρήστη. Επίσης γίνεται αρχικοποίηση όλων των resources της εφαρμογής ανάλογα με την προκαθορισμένη γλώσσα αυτής. Επιπλέον λαμβάνει το αναγνωριστικό Id που επιστρέφεται από την κλήση στο Firebase για το authentication



του χρήστη κατά το login του στην εφαρμογή με σκοπό είτε να μπορεί να ανακτήσει τις σημειώσεις ή τις κατηγορίες του εκάστοτε χρήστη είτε να μπορεί να γίνει καταχώρηση νέων. Επίσης ένας ακόμα λόγος που χρειάζεται η λήψη του Id του χρήστη πέρα από την ταυτοποίηση του στην εφαρμογή είναι γιατί μετά την προβολή της οθόνης αυτής ο χρήστης θα μπορούσαμε να πούμε ότι συμπεριλαμβάνεται σε μια λίστα ώστε να λαμβάνει ειδοποιήσεις για σημειώσεις που τον αφορούν μέσω της τεχνολογίας Firebase. Η λειτουργικότητα που χρησιμοποιείται αυτή τη φορά μέσω του Firebase ονομάζεται push notification και την οποία και θα αναλύσουμε παρακάτω στην ενότητα αυτή. Κλείνοντας με την λειτουργικότητα του συγκεκριμένου Activity γίνεται επίσης αρχικοποίηση όλων των listeners της οθόνης που είναι υπεύθυνη για να ανακατευθύνουν τον χρήστη μέσα στην εφαρμογή ανάλογα με τις επιλογές του.

- 4. MainActivity1:** Το Activity αυτό περιέχει την λειτουργικότητα που είναι απαραίτητη για την εισαγωγή μιας νέας σημείωσης. Επιλέγοντας ο χρήστης την σχετική επιλογή για καταχώρηση νέας σημείωσης από το αρχικό μενού επιλογών ο κώδικας στο παρασκήνιο τρέχει το συγκεκριμένο Activity. Με την κλήση αυτού του Activity ο κώδικας που εμπεριέχεται μέσα σε αυτό αρχικοποιεί τους απαραίτητους listeners προκειμένου να μπορούν να αναπαραχθούν οι απαραίτητες ενέργειες κατά την χρήση των κουμπιών που προσφέρει το συγκεκριμένο Activity. Εν συνεχεία αρχικοποιούνται όλα τα απαραίτητα γραφικά στοιχεία της οθόνης. Επιπροσθέτως λαμβάνονται όλες οι απαραίτητες πληροφορίες που απαιτούνται για την ομαλή λειτουργία της εφαρμογής πχ η αρχικοποίηση όλων των dropdown λιστών της συγκεκριμένης οθόνης αλλά και πληροφορίες που προσκομίζονται από το ένα Activity στο άλλο όπως για παράδειγμα το αναγνωριστικό του χρήστη κατά την είσοδο του στην εφαρμογή (userId). Με την εμφάνιση των γραφικών στοιχείων που συνθέτουν την οθόνη αυτή ο χρήστης μπορεί να καταχωρήσει στα αντίστοιχα πεδία τις πληροφορίες που αφορούν την σημείωση που θέλει να καταγράψει. Αυτό σημαίνει ότι μπορεί να επιλέξει σε ορισμένα πεδία της οθόνης εγγραφές τυποποιημένες από την εφαρμογή όπως για παράδειγμα τα διαθέσιμα πανεπιστήμια και τα σχετικά τμήματα και μαθήματα αυτών. Ένα από τα πεδία που εμπεριέχονται σε αυτή την οθόνη είναι και αυτό της τοποθεσίας από όπου γίνεται η ανάρτηση της σημείωσης. Στην περίπτωση λοιπόν που ο χρήστης το επιθυμεί ανοίγοντας την κεραία του GPS ο κώδικας στο παρασκήνιο ελέγχει αν η εφαρμογή έχει πάρει τα απαραίτητα δικαιώματα ώστε να επιτραπεί από το λειτουργικό σύστημα να γίνει χρήση του GPS από την παρούσα εφαρμογή. Μετά την επιτυχημένη επιβεβαίωση για την χρήση του GPS λαμβάνονται οι γεωγραφικές συντεταγμένες του χρήστη και ο κώδικας συνθέτη την διεύθυνση της τοποθεσίας. Μια από τις επιλογές που δίνονται στην συγκεκριμένη οθόνη είναι και αυτή της λήψης φωτογραφίας. Πατώντας λοιπόν το κουμπί για το άνοιγμα της κάμερας καλείται ο απαραίτητος για την συγκεκριμένη λειτουργία listener και η κάμερα της εκάστοτε συσκευής ανοίγει. Μετά την λήψη της φωτογραφίας ο κώδικας ελέγχει αν ήταν επιτυχής η λήψη, ενημερώνει το σχετικό γραφικό στοιχείο της οθόνης με τα δεδομένα της φωτογραφίας που μόλις τραβήχτηκε και ενεργοποιεί την επιλογή του κουμπιού προεπισκόπησης της σε περίπτωση που ο χρήστης επιθυμεί να ξανά δει πέρα από την μικρογραφία που παρέχεται στην οθόνη σε μεγέθυνση την εικόνα που μόλις τράβηξε. Στην περίπτωση που επιθυμεί να δει την φωτογραφία σε μεγέθυνση καλείται ο σχετικός listener ο οποίος «αποστέλλει» μέσω του intent την απαραίτητη για την εικόνα πληροφορία αλλά και όλες τις αλλαγές που τυχόν έχουν





γίνει στη σημείωση στο σχετικό για την προεπισκόπηση Activity. Αφού ο χρήστης συμπληρώσει τις πληροφορίες της σημείωσης που θέλει να καταχωρήσει μπορεί να επιλέξει το κουμπί της αποθήκευσης που παράλληλα κάνει ανάρτηση της σημείωσης σε όλους τους χρήστες της εφαρμογής. Πατώντας λοιπόν το κουμπί της αποθήκευσης καλείται ο listener που είναι υπεύθυνος για την αποθήκευση και την ανάρτηση της σημείωσης. Συγκεκριμένα συλλέγει από όλα τα καταχωρημένα πεδία της οθόνης τις πληροφορίες που έχει καταχώρηση ή επιλέξει ο χρήστης και συνθέτει τα queries με τους σχετικούς πίνακες που πρέπει να τρέξουν στην τοπική βάση (SQLite) που έχει δημιουργηθεί στη συσκευή καταχωρώντας παράλληλα και το αναγνωριστικό χρήστη που δηλώνει ότι η σημείωση ανήκει στο συγκεκριμένο. Επιπλέον ελέγχει αν ο χρήστης έχει τραβήξει κάποια φωτογραφία. Στην περίπτωση που έχει γίνει λήψη φωτογραφίας ο κώδικας καλεί και πάλι την τεχνολογία firebase αυτή την φορά κάνοντας χρήση της λειτουργικότητας storage που θα εξηγήσουμε παρακάτω προκειμένου να αποθήκευση στο cloud την φωτογραφία που τραβήχτηκε για την σημείωση ώστε αυτή να μπορεί να είναι ορατή με την προβολή της συγκεκριμένης σημείωσης απ' όλους τους χρήστες. Επίσης στο συγκεκριμένο σημείο του κώδικα γίνεται χρήση πάλι της τεχνολογίας firebase για την λειτουργικότητα της βάσης δεδομένων στο cloud. Οι πληροφορίες που επρόκειτο να καταχωρηθούν στην τοπική βάση συμπεριλαμβάνονται σε σχετική λίστα ώστε να καταχωρηθούν και στο cloud μέσω της τεχνολογίας firebase για να είναι προσβάσιμη και από άλλους χρήστες. Μετά την λήψη και την προετοιμασία της σχετικής με την σημείωση πληροφορίας εμφανίζεται στο χρήστη η μπάρα προόδου της εφαρμογής. Ο κώδικας ανάλογα με το αποτέλεσμα που έλαβε από τον έλεγχο της ύπαρξης ή όχι φωτογραφίας «ανεβάζει» την σημείωση με την φωτογραφία ή χωρίς στην περίπτωση που δεν υπάρχει στο cloud. Παράλληλα τρέχει και τα queries με την σχετική πληροφορία στην τοπική βάση προκειμένου να αποθηκευτεί η πληροφορία και στην συσκευή. Μετά την επιτυχημένη ολοκλήρωση των ενεργειών αυτών η μπάρα προόδου κρύβεται από την οθόνη του χρήστη. Σε κάθε άλλη περίπτωση ο χρήστης ενημερώνεται με μήνυμα σχετικά με το σφάλμα που οδήγησε σε ανεπιτυχή αποθήκευση και ανάρτηση της σημείωσης. Τέλος στην οθόνη αυτή περιλαμβάνεται και η λειτουργία cancel η οποία όταν καλείται ο κώδικας ανακατευθύνει τον χρήστη στην βασική οθόνη αποστέλλοντας μαζί και το αναγνωριστικό εισόδου που έλαβε κατά την είσοδο του στην εφαρμογή.

- 5. MainActivity2:** Το συγκεκριμένο Activity περιέχει την λειτουργικότητα για την εμφάνιση όλων των αποθηκευμένων σημειώσεων του εκάστοτε χρήστη. Με το κάλεσμα της οθόνης αυτής στον κώδικα λαμβάνεται το αναγνωριστικό του χρήστη που έλαβε κατά την είσοδο του στην εφαρμογή προκειμένου να μπορεί να γίνει ανάκτηση όλων των αποθηκευμένων σημειώσεων από την τοπική βάση που έχει δημιουργηθεί στην εκάστοτε συσκευή που χρησιμοποιεί. Μετά την λήψη του αναγνωριστικού χρήστη αρχικοποιούνται τα γραφικά στοιχεία της οθόνης αλλά και οι listeners αυτής της ώστε να μπορεί να αναπαραχθεί η απαραίτητη λειτουργικότητα μετά το πάτημα των κουμπιών που προσφέρονται σε αυτή. Εν συνέχεια σε συνάρτηση με το αναγνωριστικό χρήστη που έχει ληφθεί διαμορφώνεται και τρέχει το query στην τοπική βάση ανακτώντας τις σημειώσεις του συγκεκριμένου χρήστη γεμίζοντας την λίστα με τις σημειώσεις που εμφανίζονται στην συγκεκριμένη οθόνη. Πατώντας ο χρήστης επάνω στον τίτλο κάποιας από τις αποθηκευμένες σημειώσεις του που εμφανίζονται στην οθόνη ο κώδικας στο παρασκήνιο ενεργοποιεί τον απαραίτητο listener. Στην περίπτωση



αυτή ο συγκεκριμένος listener λαμβάνει τον τίτλο της σημείωσης που επιλέχθηκε αλλά και το αναγνωριστικό του χρήστη προκειμένου να σταλεί μέσω intent στο Activity προβολής της σημείωσης. Τέλος και σε αυτή την οθόνη προσφέρεται κουμπί επιστροφής στο βασικό μενού επιλογών που όταν πατηθεί από τον χρήστη τον ανακατευθύνει στο βασικό μενού στέλνοντας το αναγνωριστικό που έλαβε κατά την είσοδο του.

6. **MainActivity3:** Το Activity αυτό περιέχει την λειτουργικότητα για την προβολή και την επεξεργασία μίας αποθηκευμένης σημείωσης του χρήστη. Καλώντας ο χρήστης από το βασικό μενού των επιλογών το συγκεκριμένο Activity ο κώδικας αυτού αρχικοποιεί τους απαραίτητους για την λειτουργικότητα της οθόνης listeners. Εν συνεχεία δημιουργεί όλα τα γραφικά στοιχεία που πλαισιώνουν την οθόνη αυτή. Προχωρώντας η εκτέλεση του κώδικα λαμβάνει τις απαραίτητες για την λειτουργία της οθόνης αυτής πληροφορίες που έχουν παραχθεί σε αυτό ή και σε κάποια άλλα Activity. Τέτοιες πληροφορίες αποτελούν το αναγνωριστικό χρήστη που λαμβάνεται από την οθόνη εισόδου ή πληροφορίες με τυχόν αλλαγές που έχουν συμβεί από τον χρήστη κατά την επεξεργασία της σημείωσης και χρειάστηκαν να σταλθούν σε κάποιο άλλο Activity. Μία τέτοια περίπτωση όπου γίνεται αποστολή πληροφορίας από το ένα Activity στο άλλο αποτελεί η προεπισκόπηση της εικόνας. Σκοπός της διαδικασίας αυτής είναι το συγκεκριμένο Activity (**MainActivity3**) να λάβει ξανά πίσω όλη την πληροφορία που αποσταλθεί στην οθόνη προεπισκόπησης μαζί με τις οποιοσδήποτε αλλαγές έχει υποστεί κατά την επεξεργασία της από τον χρήστη. Από την στιγμή που ληφθούν πληροφορίες από άλλο Activity ο κώδικας στο παρασκήνιο ελέγχει αν η πληροφορία προήλθε από την οθόνη προεπισκόπησης εικόνας. Αν ναι παρουσιάζει αυτή στα αντίστοιχα πεδία της σημείωσης διαφορετικά επικοινωνεί με την τοπική βάση αλλά και την βάση στο cloud μέσω firebase προκειμένου να την λάβει από εκεί. Επιπρόσθετα ένα επιπλέον στοιχείο που ελέγχει ο κώδικας είναι και αυτό της εικόνας. Επικοινωνεί λοιπόν με την τεχνολογία firebase και αναζητεί για την συγκεκριμένη σημείωση αν υπάρχει φωτογραφία ή όχι. Στην περίπτωση που υπάρχει εξετάζει αν έχει ενημερωθεί με κάποια άλλη ή όχι και έπειτα την φορτώνει στην οθόνη του χρήστη ενεργοποιώντας και την δυνατότητα προεπισκόπησης της. Επιπλέον πέρα από τον έλεγχο της ύπαρξης ή όχι φωτογραφίας ο κώδικας επικοινωνεί με την τεχνολογία firebase προκειμένου να λάβει την σχετική πληροφορία για το αν ο χρήστης στον οποίο προβάλλεται η σημείωση είναι ο συντάκτης της ή ένας απλός χρήστης που την έχει κατεβάσει για προβολή. Στην περίπτωση που ο κώδικας μέσα από τον έλεγχο που κάνει οδηγηθεί στο συμπέρασμα ότι ο τρέχον χρήστης της εφαρμογής είναι και ιδιοκτήτης της σημείωσης ενεργοποιεί όλα τα σχετικά πεδία για επεξεργασία καθώς εμφανίζει μαζί με το κουμπιά της διαγραφής και της ακύρωσης και το κουμπί της ενημέρωσης της σημείωσης καθώς και αυτό της λήψης φωτογραφίας. Στην αντίθετη περίπτωση δηλαδή αυτή που ο χρήστης δεν είναι ιδιοκτήτης της σημείωσης τότε ο κώδικας διατηρεί ανενεργά όλα τα πεδία της σημείωσης καθώς αυτή πλέον προσφέρεται μόνο για προβολή. Όσον αφορά της δυνατότητες που έχει ο χρήστης σε αυτή την περίπτωση περιορίζονται σε αυτές της διαγραφής, της ακύρωσης και της προεπισκόπησης εικόνα αν αυτή είναι διαθέσιμη με τα αντίστοιχα κουμπιά να εμφανίζονται στην οθόνη του χρήστη. Όσον αφορά το λειτουργικό κομμάτι αυτών των κουμπιών θα ξεκινήσουμε την αναφορά μας από την περίπτωση του κουμπιού της ενημέρωσης της σημείωσης. Όταν το κουμπί αυτό εμφανιστεί στην συγκεκριμένη οθόνη σημαίνει ότι ο χρήστης είναι ιδιοκτήτης της σημείωσης και έχει



δικαίωμα να την επεξεργαστεί. Αυτό πρακτικά για τον κώδικα σημαίνει ότι από την στιγμή που τα πεδία της σημείωσης είναι ελεύθερα προς επεξεργασία στο πάτημα του κουμπιού της ενημέρωσης λαμβάνει όλες της σχετικές πληροφορίες από τα πεδία της σημείωσης τα οποία έχει επεξεργαστεί ο χρήστης. Μετά την λήψη αυτών επικοινωνεί με την τεχνολογία firebase προκειμένου να βρει και να ενημερώσει με τις αλλαγές την σημείωση. Στο σημείο αυτό εξετάζει και πάλι αν έχει γίνει ενημέρωση της φωτογραφίας ή όχι προκειμένου να ενημερώσει αντίστοιχα την βάση στο cloud μέσω της τεχνολογίας firebase. Παράλληλα με την ενημέρωση της βάσης στο cloud ο κώδικας ενημερώνει και την τοπική βάση στην εκάστοτε συσκευή που τρέχει. Επιπλέον στο σημείο αυτό του κώδικα και μετά τις επιτυχημένες ενημερώσεις της βάσης τόσο σε τοπικό όσο και σε επίπεδο cloud ο κώδικας επικοινωνεί και πάλι με την τεχνολογία firebase. Αυτή την φορά η επικοινωνία γίνεται για να γίνει χρήση της συγκεκριμένης τεχνολογίας αλλά για την λειτουργικότητα notification την οποία θα αναλύσουμε στην συνέχεια της ενότητας αυτής. Ο κώδικας λοιπόν λαμβάνει μία σχετική λίστα με όλους τους χρήστες που έχουν κατεβάσει την συγκεκριμένη σημείωση από την βάση cloud και κάνοντας χρήση της λειτουργικότητας notification του firebase ενημερώνει όλους του ενδιαφερόμενους ότι η σημείωση ενημερώθηκε από τον ιδιοκτήτη της προκειμένου αν το επιθυμούν να την ξανακατεβάσουν με τις αλλαγές που έχουν γίνει σε αυτή. Όσον αφορά τώρα την λειτουργικότητα στο κουμπί της διαγραφής της σημείωσης ο κώδικας λαμβάνει τον τίτλο της σημείωσης και σε συνδυασμό με το αναγνωριστικό του χρήστη όταν κληθεί να γίνει διαγραφή την σβήνει από τις σημείωσης του χρήστη από την τοπική βάση της εκάστοτε συσκευής που έχει αποθηκευτεί. Επίσης στην οθόνη αυτή όπως και στην οθόνη εισαγωγής μιας νέας σημείωσης παρέχεται εξίσου η δυνατότητα λήψης και προεπισκόπησης φωτογραφίας. Έτσι λοιπόν πατώντας ο χρήστης το άνοιγμα της φωτογραφικής μηχανής και αμέσως με τα την λήψη της στον κώδικα ελέγχεται αν αυτή τραβήχτηκε επιτυχώς ή όχι. Στην περίπτωση της επιτυχούς λήψης φορτώνει την φωτογραφία στο αντίστοιχο γραφικό στοιχείο της οθόνης και ενημερώνει τον κώδικα πώς η εικόνα υπέστη αλλαγή. Επίσης όπως αναφέραμε και παραπάνω στην οθόνη αυτή παρέχεται και η δυνατότητα προεπισκόπησης της εικόνας. Στην περίπτωση λοιπόν που υπάρχει ήδη κάποια εικόνα για την συγκεκριμένη σημείωση τότε το κουμπί της προεπισκόπησης έχει ενεργοποιηθεί. Πατώντας το λοιπόν ο χρήστης ο κώδικας καλεί τον απαραίτητο listener. Αυτός με την σειρά του λαμβάνει όλες τις αλλαγές που μπορεί να έχει κάνει ο χρήστης στην σημείωση κατά την επεξεργασία της μαζί με την πληροφορία για την εικόνα και τις αποστέλλει μέσω intent στην οθόνη προεπισκόπησης. Τέλος στην οθόνη αυτή περιλαμβάνεται και η λειτουργία cancel η οποία όταν καλείται ο κώδικας ανακατευθύνει τον χρήστη στην οθόνη που περιλαμβάνει την λίστα των σημειώσεων του τρέχοντος χρήστη αποστέλλοντας μαζί και το αναγνωριστικό εισόδου που έλαβε κατά την είσοδο του στην εφαρμογή.

- 7. MainActivity4:** Η οθόνη αυτή περιλαμβάνει την λειτουργικότητα που χρειάζεται για να εμφανιστεί η λίστα με τις κατηγορίες που έχει δημιουργήσει ο χρήστης προκειμένου να κατηγοριοποιήσει τις σημείωσης του. Με την επιλογή της συγκεκριμένη οθόνης από το βασικό μενού των επιλογών ο κώδικας ανατρέχει το συγκεκριμένο αρχείο προκειμένου να προβάλει στον χρήστη την αναμενόμενη λειτουργικότητα. Με την κλήση λοιπόν αυτού του Activity ο κώδικας αρχικά λαμβάνει και σε αυτή την περίπτωση το αναγνωριστικό χρήστη που έλαβε από την εφαρμογή κατά την είσοδο του σε αυτή. Εν συνέχεια αρχικοποιεί τους



απαραίτητους για την λειτουργία της οθόνη Listeners και ενημερώνει όλα τα γραφικά στοιχεία της οθόνης αυτής. Αφού ολοκληρώσει τις παραπάνω ενέργειες επικοινωνεί με την τοπική βάση (SQLite) και λαμβάνει τις κατηγορίες που έχει δημιουργήσει ο χρήστης μέχρι εκείνη την στιγμή και τις παραθέτει σε μια λίστα την οποία και εμφανίζει στην οθόνη. Επιλέγοντας ο χρήστης κάποια από αυτές καλείτε ο σχετικός Listener ο οποίος λαμβάνει τον τίτλο της κατηγορίας που επιλέχθηκε και τον αποστέλλει μέσω Intent μαζί με το αναγνωριστικό χρήστη στην οθόνη εμφανίσεις των σχετικών με την κατηγορία σημειώσεων (**MainActivity6**). Επιπλέον και στην οθόνη αυτή παρέχετε η δυνατότητα επιστροφής στο βασικό μενού όπου όταν καλείτε μέσω του κουμπιού επιστροφής ο σχετικός για την λειτουργία Listener ανακατευθύνει τον χρήστη στο μενού επιλογών επιστρέφοντας και το αναγνωριστικό του κωδικό.

8. **MainActivity5:** Το συγκεκριμένο Activity περιέχει την απαραίτητη λειτουργικότητα για την δημιουργία μιας νέας κατηγορίας σημειώσεων. Όταν αυτό καλείται από το βασικό μενού των επιλογών ο κώδικας του αρχείου αυτού αρχικά ελέγχει αν έχει δημιουργηθεί στην εκάστοτε συσκευή που τρέχει η εφαρμογή ο σχετικός πίνακας που θα περιλαμβάνει τις εκάστοτε κατηγορίες του χρήστη. Στην περίπτωση που αυτός δεν υπάρχει στην συσκευή τον δημιουργεί. Στην συνέχεια και σε αυτή την οθόνη ο κώδικας προετοιμάζει τους απαραίτητους για την λειτουργικότητα αυτής Listeners και αρχικοποιεί τα γραφικά στοιχεία που την συνθέτουν. Παράλληλα και εδώ λαμβάνεται το αναγνωριστικό του χρήστη από την είσοδο του στην εφαρμογή. Μόλις οι παραπάνω ενέργειες ολοκληρωθούν η οθόνη εμφανίζεται στον χρήστη και μπορεί να γίνει η καταχώρηση νέας κατηγορίας. Ο χρήστης μπορεί να πληκτρολογήσει την ονομασία της νέας κατηγορίας και αφού γίνουν οι απαραίτητοι έλεγχοι για την εγκυρότητα της εγγραφής αλλά και την διασφάλιση ότι η κατηγορία που καταχωρήθηκε δεν υπάρχει ήδη η νέα κατηγορία εισάγεται στην τοπική βάση της συσκευής. Με την επιτυχημένη ολοκλήρωση ή όχι της καταχώρησης σχετικό μήνυμα εμφανίζεται στην οθόνη του χρήστη. Ολοκληρώνοντας με την λειτουργικότητα της οθόνης αυτής προσφέρεται και εδώ κουμπί επιστροφής στο αρχικό μενού όπου και αποστέλλεται μέσω Intent το αναγνωριστικό χρήστη.
9. **MainActivity6:** Στο αρχείο αυτό περιλαμβάνεται η λειτουργικότητα για την εμφάνιση των σημειώσεων ανάλογα με την κατηγορία που έχει επιλεγεί ως αποτέλεσμα της επιλογής που έγινε από την οθόνη με τις σχετικές κατηγορίες του χρήστη. Μετά λοιπόν την επιλογή αυτή που έγινε στο **MainActivity4** ο κώδικας ανακατευθύνεται στο συγκεκριμένο αρχείο. Στο σημείο αυτό ο κώδικας του αρχείου προετοιμάζει τους απαραίτητους για την λειτουργία της οθόνης Listeners καθώς και τα γραφικά της οθόνης. Εν συνεχεία λαμβάνει το αναγνωριστικό χρήστη καθώς και την ονομασία της επιλεγμένης κατηγορία που πραγματοποιήθηκε σε προηγούμενη οθόνη της εφαρμογής (**MainActivity4**). Έχοντας λοιπόν τις πληροφορίες αυτές ανατρέχει στην τοπική βάση της συσκευής και αναζητά της σημειώσεις του τρέχον χρήστη που έχουν κατηγοριοποιηθεί κάτω από την επιλεγμένη κατηγορία. Με την λήψη αυτών από την τοπική βάση τις ενσωματώνει σε μια λίστα και τις εμφανίζει στην οθόνη του χρήστη. Με την επιλογή λοιπόν μίας από τις σχετικές με την κατηγορία σημειώσεις του χρήστη καλείται ο υπεύθυνος για την απαραίτητη λειτουργικότητα Listener ο οποίος με την σειρά του ανακατευθύνει τον χρήστη στην οθόνη προβολής της επιλεγμένης σημείωσης (**MainActivity3**) στέλνοντας παράλληλα και το αναγνωριστικό χρήστη. Τελειώνοντας με την λειτουργικότητα της



συγκεκριμένης οθόνης παρέχεται και η δυνατότητα επιστροφής στο βασικό μενού επιλογών αποστέλλοντας μαζί το αναγνωριστικό κωδικό εισόδου του χρήστη.

10. **MainActivity7:** Στην οθόνη αυτή περιλαμβάνετε η λειτουργικότητα για την εμφάνιση όλων των σημειώσεων που υπάρχουν στο «σύννεφο». Συνεπώς μιλάμε όχι μόνο για τις σημειώσεις που έχει δημιουργήσει ο τρέχον χρήστης στην εφαρμογή αλλά για κάθε σημείωση που έχει αναρτηθεί μέσω αυτής από οποιονδήποτε χρήστη. Με την επιλογή λοιπόν αυτής της λειτουργικότητας από το βασικό μενού ο κώδικας ανατρέχει στο αρχείο αυτό. Ως πρώτη του προτεραιότητα λαμβάνει το αναγνωριστικό κωδικό με τον οποίο ο χρήστης έκανα είσοδο στην εφαρμογή. Εν συνεχεία επικοινωνεί με το cloud μέσω της τεχνολογίας firebase και πιο συγκεκριμένα με την βάση αυτού. Σκοπός της επικοινωνίας αυτής είναι να λάβει όλες τις σημειώσεις των χρηστών που έχουν αναρτηθεί μέσω της εφαρμογής. Μόλις ολοκληρώσει την ενέργεια αυτή συγκεντρώνει τις σημειώσεις και τις τοποθετεί σε μια λίστα προκειμένου να τις προβάλλει στην οθόνη του χρήστη. Από το σημείο αυτό όταν ο χρήστης επιλέξει μία από τις προβαλλόμενες σημειώσεις καλείτε ο σχετικός Listener προκειμένου να προωθήσει την σημείωση στην οθόνη προβολής και αποθήκευσης από το «σύννεφο» (**MainActivity8**) στέλνοντας παράλληλα το αναγνωριστικό χρήστη, το αναγνωριστικό κωδικό της σημείωσης καθώς και τον τίτλος. Τέλος και αυτή η οθόνη περιλαμβάνει την σχετική λειτουργικότητα για την επιστροφή στο αρχικό μενού μέσω του κουμπιού Πίσω στέλνοντας μαζί και το αναγνωριστικό χρήστη μέσω Intent.
11. **MainActivity8:** Περιέχει την λειτουργικότητα για την προβολή και αποθήκευση μιας σημείωσης από το σύννεφο τοπικά στην συσκευή του χρήστη. Μόλις γίνει η επιλογή μια σημείωσης από την λίστα των online σημειώσεων σε προηγούμενη οθόνη (**MainActivity7**) ο κώδικας οδηγείται στο αρχείο αυτό. Όταν λοιπόν αυτό εκτελείται από την εφαρμογή αρχικοποιεί ως πρώτο βήμα τα γραφικά στοιχεία της οθόνης και εν συνεχεία λαμβάνει το αναγνωριστικό χρήστη, το αναγνωριστικό κωδικό της σημείωσης καθώς και τον τίτλο αυτής που επιλέχθηκε κατά το προηγούμενο βήμα (**MainActivity7**). Έχοντας λοιπόν όλη την απαραίτητη πληροφορία που χρειάζεται ανατρέχει στην βάση στο cloud μέσω του firebase και βρίσκει την επιλεγμένη σημείωση από τον χρήστη με σκοπό την προβολή της στην οθόνη του. Με το που την βρει λοιπόν συμπληρώνει τα σχετικά πεδία αυτής και «κλειδώνει» τα γραφικά στοιχεία της οθόνης καθώς η σημείωση στο σημείο αυτό είναι μόνο για προβολή και αποθήκευση τοπικά στην συσκευή αν ο χρήστης το επιθυμεί. Αν ο χρήστης αποφασίσει να συμπεριλάβει πατώντας το κουμπί της αποθήκευσης την προβαλλόμενη σημείωση μαζί με αυτές που έχει αποθηκευμένες στην συσκευή του τότε ο κώδικας αρχικά ανατρέχει στις σημειώσεις που έχει ο τρέχον χρήστης τοπικά και αν δεν την βρει ήδη καταχωρημένη τότε την αποθηκεύει. Σε αντίθετη περίπτωση τον ενημερώνει πως η σημείωση που επέλεξε να αποθήκευση υπάρχει ήδη. Σε κάθε περίπτωση κατά την αποθήκευση μιας σημείωσης είτε αυτή καταχωρηθεί επιτυχώς στις σημειώσεις του χρήστη είτε όχι λόγο κάποιου πιθανού προβλήματος που μπορεί να προέκυψε ενημερωτικό μήνυμα εμφανίζεται στη οθόνη του χρήστη. Τέλος κουμπί επιστροφή στο βασικό μενού των επιλογών προσφέρεται και σε αυτή την οθόνη επιστρέφοντας μαζί και το αναγνωριστικό του χρήστη.
12. **ImageViewerActivity:** Περιέχει την απαραίτητη λειτουργικότητα για την προβολή των εικόνων που έχουν τραβηχτεί στην εφαρμογή από τον χρήστη. Όταν ο χρήστης επιλέξει να κάνει προβολή της φωτογραφίας που συμπεριλαμβάνεται με την σχετική σημείωση από την οθόνη εισαγωγής μιας νέας σημείωσης τότε ο κώδικας





ανακατευθύνεται στο συγκεκριμένο Activity. Με το που κληθεί για εκτέλεση λοιπόν το σχετικό αρχείο λαμβάνονται όλες οι απαραίτητες πληροφορίες από τον αρχείο που έγινε η κλήση προς την οθόνη προεπισκόπησης της φωτογραφίας και εν συνεχεία προβάλετε η φωτογραφία της σημείωσης στην οθόνη του χρήστη. Με την λειτουργία του κουμπιού Πίσω συλλέγεται και πάλι όλη η πληροφορία που λήφθηκε κατά την κλήση του συγκεκριμένου Activity και ανακατευθύνεται μαζί με αυτή στην οθόνη από όπου ο χρήστης σε πρώτο βήμα επέλεξε να γίνει προβολή της φωτογραφίας (**MainActivity1**) που συμπεριλαμβάνεται με την σημείωση που έβλεπε.

13. **ImageViewerActivity3:** Περιέχει την απαραίτητη λειτουργικότητα για την προβολή των εικόνων που έχουν τραβηχτεί στην εφαρμογή. Όταν ο χρήστης επιλέξει από την οθόνη προβολής μια αποθηκευμένης τοπικά σημείωσης να κάνει προβολή της φωτογραφίας που συμπεριλαμβάνεται με την σχετική σημείωση τότε ο κώδικας ανακατευθύνεται στο συγκεκριμένο Activity. Με το που κληθεί για εκτέλεση λοιπόν το σχετικό αρχείο λαμβάνονται όλες οι απαραίτητες πληροφορίες από τον αρχείο που έγινε η κλήση προς την οθόνη προεπισκόπησης της φωτογραφίας και εν συνεχεία προβάλετε η φωτογραφία της σημείωσης στην οθόνη του χρήστη. Με την λειτουργία του κουμπιού Πίσω συλλέγεται και πάλι όλη η πληροφορία που λήφθηκε κατά την κλήση του συγκεκριμένου Activity και ανακατευθύνεται μαζί με αυτή στην οθόνη από όπου ο χρήστης σε πρώτο βήμα επέλεξε να γίνει προβολή της φωτογραφίας που συμπεριλαμβάνεται με την σημείωση που έβλεπε.
14. **MyFirebaseMessagingService:** Το αρχείο αυτό εμπεριέχει όλη την απαραίτητη λειτουργικότητα για την αποστολή και την λήψη των ειδοποιήσεων μέσω της λειτουργικότητας push notification του firebase. Με αλλά λόγια εμπεριέχει τις απαραίτητες συναρτήσεις που τρέχουν στην εφαρμογή όταν χρειάζεται να αποσταλεί ή να εμφανιστεί κάποια ειδοποίηση. Συγκεκριμένα στο αρχείο αυτό υπάρχουν δύο βασικές συναρτήσεις που όταν καλούνται μέσα από τον κώδικα της εφαρμογής τότε εκτελούν τις ενέργειες που αναφέρθηκαν παραπάνω. Οι συναρτήσεις αυτές είναι:
  - **SendNotificationWithTopic:** Η συνάρτηση αυτή όταν καλείτε συνθέτει ένα Post Request που επικοινωνεί με την τεχνολογία firebase και το οποίο εμπεριέχει τον τίτλο της σημείωσης, το μήνυμα της ειδοποίησης, καθώς και τον χρήστη ή την ομάδα χρηστών στους οποίους απευθύνεται.
  - **OnMessageReceived:** Η συνάρτηση αυτή είναι υπεύθυνη να λαμβάνει τα μηνύματα που επικοινωνούν με την τεχνολογία firebase και από εκεί ανακατευθύνονται στους τελικούς παραλήπτες. Όταν ένα μήνυμα ληφθεί τότε λαμβάνεται ο τίτλος της σημείωσης καθώς και το μήνυμα που μεταφέρει. Εν συνεχεία καλεί την συνάρτηση Show Notification προκειμένου να την εμφανίσει στην οθόνη του χρήστη.
  - **ShowNotification:** Η συνάρτηση αυτή λαμβάνει τα στοιχεία της ειδοποίησης που μόλις λήφθηκαν από την συνάρτηση OnMessageReceived και δημιουργεί μια νέα ειδοποίηση με τα στοιχεία αυτά προκειμένου να την εμφανίσει στον χρήστη. Επιπλέον σε αυτή την συνάρτηση ορίζεται το που θα ανακατευθυνθεί ο κώδικας όταν ο χρήστης πατήσει επάνω στην ειδοποίηση για την δει. Στην προκειμένη περίπτωση



θα τον οδηγήσει στην οθόνη προβολής της σημείωση για την οποία ειδοποιήθηκε.

- 15. SubscribeActivity:** Το Activity αυτό περιλαμβάνει την λειτουργικότητα για την εγγραφή ενός χρήστη σε κάποιο σχετική θεματολογία προκειμένου να λαμβάνει ειδοποιήσεις που τον αφορούν. Ειδικότερα όταν καλείται από την εφαρμογή το σχετικό αρχείο προς εκτέλεση ο κώδικας αρχικά λαμβάνει το αναγνωριστικό χρήστη. Εν συνεχεία προετοιμάζει τους Listeners της οθόνης και αρχικοποιεί τα γραφικά στοιχεία αυτής. Αφού ολοκληρώσει με τις ενέργειες αυτές επικοινωνεί με την βάση στο firebase προκειμένου να λάβει τα πανεπιστήμια και τα σχετικά τμήματα και μαθήματα αυτών με τα οποία θα ενημερώσει τις σχετικές λίστες στην οθόνη αυτή. Από την στιγμή που ο χρήστης επιλέξει το πανεπιστήμιο, το τμήμα και το μάθημα για το οποίο θέλει να λαμβάνει ειδοποιήσεις για σχετικές σημειώσεις που ανεβαίνουν από άλλους χρήστες της εφαρμογής πατάει το κουμπί της εγγραφής. Με το πάτημα του κουμπιού αυτού ο σχετικός Listener καλείτε. Αυτός με την σειρά του επικοινωνεί με την βάση στο firebase και ανατρέχει στον σχετικό πίνακα προκειμένου να διαπιστώσει αν ο τρέχον χρήστης είναι ήδη εγγεγραμμένος να λαμβάνει ειδοποιήσεις για το μάθημα του συγκεκριμένου τμήματος και πανεπιστημίου που επέλεξε. Στην περίπτωση που κάτι τέτοιο συμβαίνει ήδη τον ενημερώνει με μήνυμα ότι είναι εγγεγραμμένος διαφορετικά τον εγγράφει στο σχετικό θέμα. Μετά την επιτυχημένη εγγραφή του χρήστη μήνυμα εμφανίζεται στην οθόνη καθώς και ο συνδυασμός πανεπιστήμιο, τμήμα και μάθημα που μόλις εγγράφηκε εμφανίζεται στην λίστα με τα θέματα που ακολουθεί και θέλει να ενημερώνετε σχετικά με αυτά. Επίσης στην οθόνη αυτή παρέχεται και η δυνατότητα απεγράφης από ένα μάθημα ώστε να μην λαμβάνονται σχετικές ειδοποιήσεις. Αυτή η ενέργεια επιτυγχάνεται όταν επιλέξει πάλι από τις λίστες με τα πανεπιστήμια και τα σχετικά τμήματα και μαθήματα αυτών εκείνο το μάθημα από το οποίο θέλει να απεγραφεί και πατάει το κουμπί της διαγραφής. Ο κώδικας λοιπόν σε αυτό το σημείο καλεί τον σχετικό listener και αυτός ελέγχει αρχικά αν η εγγραφή που ζητείται από τον χρήστη για διαγραφή είναι καταχωρημένη στις εγγραφές του. Στην περίπτωση που αυτή βρεθεί γίνεται διαγραφή και ενημερώνει τον χρήστη για την ολοκλήρωση της ενέργειας αυτής. Διαφορετικά του εμφανίζει μήνυμα ότι η εγγραφή που επιθυμεί να διαγράψει δεν υπάρχει. Τέλος και στην οθόνη αυτή υπάρχει η δυνατότητα επιστροφής στο βασικό μενού με το πάτημα του κουμπιού Πίσω όπου και αποστέλλεται μαζί και το αναγνωριστικό του χρήστη.
- 16. AdminActivity:** Το Activity αυτό περιλαμβάνει την λειτουργικότητα για το μενού των επιλογών στο διαχειριστικό της εφαρμογής. Όταν στην εφαρμογή συνδεθεί χρήστης με τον ρόλο του διαχειριστή τότε καλείτε το συγκεκριμένο αρχείο. Ο κώδικας λοιπόν του αρχείου αυτού λαμβάνει αρχικά το αναγνωριστικό εισόδου του χρήστη και εν συνεχεία προετοιμάζει τους Listeners της οθόνης οι οποίοι θα ανακατευθύνουν τον χρήστη στις επόμενες οθόνες ανάλογα με τις επιλογές του και αρχικοποιεί τα γραφικά στοιχεία αυτής. Με την ολοκλήρωση των ενεργειών αυτών εμφανίζονται οι επιλογές που παρέχει το διαχειριστικό στην οθόνη του χρήστη. Επιγραμματικά αναφέρουμε τις επιλογές αυτές:
- Δημιουργία νέου Πανεπιστημίου.
  - Δημιουργία νέου Τμήματος.
  - Δημιουργία νέου Μαθήματος.



- Δημιουργία νέας σχέσης.
- Διαγραφή υπάρχουσας σχέσης.
- Έξοδος.

**17. AddUniversityActivity:** Το Activity αυτό περιέχει την λειτουργικότητα για την εισαγωγή ενός νέου Πανεπιστημίου στις λίστες τις εφαρμογής. Όταν λοιπόν αυτό κληθεί από το βασικό μενού του διαχειριστικού τότε ως πρώτη ενέργεια λαμβάνει το αναγνωριστικό κωδικό του χρήστη. Επιπλέον προετοιμάζει τους Listeners της οθόνης για να μπορεί να αναπαραχθεί η απαιτούμενη λειτουργικότητα της και αρχικοποιούνται τα γραφικά στοιχεία που αφορούν την εμφάνιση αυτής. Εν συνεχεία πραγματοποιείται επικοινωνία με την βάση στο firebase όπου ζητούνται από τον σχετικό πίνακα όλα τα καταχωρημένα πανεπιστήμια στην εφαρμογή. Μόλις συγκεντρωθεί η πληροφορία αυτή τα ενσωματώνει σε μια λίστα. Την στιγμή λοιπόν που ο χρήστης καταχωρήσει το όνομα ενός νέου πανεπιστημίου τότε ο σχετικός Listener καλείται. Αυτός με την σειρά του πραγματοποιεί ελέγχους εγκυρότητας της καταχώρησης που έκανα ο χρήστης όπως το αν η εγγραφή είναι κενή ή αν υπάρχει ήδη καταχωρημένη στην λίστα με τα πανεπιστήμια της εφαρμογής που ανακτήθηκε προηγουμένως. Σε κάθε περίπτωση ενημερώνει τον χρήστη με σχετικό μήνυμα. Στην περίπτωση τώρα που οι έλεγχοι ολοκληρωθούν επιτυχώς τότε η καταχώρηση πραγματοποιείται και σχετικό μήνυμα εμφανίζεται. Λειτουργικότητα επιστροφής στο βασικό μενού του διαχειριστικού προσφέρεται και στην οθόνη αυτή επιστρέφοντας μαζί και το αναγνωριστικό χρήστη.

**18. AddDepartmentActivity:** Η οθόνη αυτή παρέχει την λειτουργικότητα για την εισαγωγή ενός νέου Τμήματος στην λίστες της εφαρμογής από το διαχειριστικό αυτής. Όταν λοιπόν ο διαχειριστής επιλέγει αυτή την λειτουργικότητα τότε εκτελείτε το συγκεκριμένο αρχείο. Ως συνέπεια της εκτέλεσης αυτής είναι αρχικά ο κώδικας να λάβει το αναγνωριστικό του χρήστη, να προετοιμάσει τους Listeners που χρειάζονται για την λειτουργία της οθόνης και να αρχικοποιήσει τα γραφικά της στοιχεία. Εν συνεχεία πραγματοποιείται επικοινωνία με την βάση στο firebase όπου ζητούνται από τον σχετικό πίνακα όλα τα καταχωρημένα τμήματα στην εφαρμογή. Μόλις συγκεντρωθεί η πληροφορία αυτή τα ενσωματώνει σε μια λίστα. Την στιγμή λοιπόν που ο χρήστης καταχωρήσει το όνομα ενός νέου τμήματος τότε ο σχετικός Listener καλείται. Αυτός με την σειρά του πραγματοποιεί ελέγχους εγκυρότητας της καταχώρησης που έκανα ο χρήστης όπως το αν η εγγραφή είναι κενή ή αν υπάρχει ήδη καταχωρημένη στην λίστα με τα τμήματα της εφαρμογής που ανακτήθηκε προηγουμένως. Σε κάθε περίπτωση ενημερώνει τον χρήστη με σχετικό μήνυμα. Στην περίπτωση τώρα που οι έλεγχοι ολοκληρωθούν επιτυχώς τότε η καταχώρηση πραγματοποιείται και σχετικό μήνυμα εμφανίζεται στον χρήστη. Λειτουργικότητα επιστροφής στο βασικό μενού του διαχειριστικού προσφέρεται και στην οθόνη αυτή επιστρέφοντας μαζί και το αναγνωριστικό χρήστη.

**19. AddLessonActivity:** Η οθόνη αυτή παρέχει την λειτουργικότητα για την εισαγωγή ενός νέου Μαθήματος στις λίστες της εφαρμογής από το διαχειριστικό αυτής. Όταν λοιπόν ο διαχειριστής επιλέγει αυτή την λειτουργικότητα τότε εκτελείτε το συγκεκριμένο αρχείο. Ως συνέπεια της εκτέλεσης αυτής είναι αρχικά ο κώδικας να λάβει το αναγνωριστικό του χρήστη, να προετοιμάσει τους Listeners που χρειάζονται για την λειτουργία της οθόνης και να αρχικοποιήσει τα γραφικά της στοιχεία. Εν συνεχεία πραγματοποιείται επικοινωνία με την βάση στο firebase όπου





ζητούνται από τον σχετικό πίνακα όλα τα καταχωρημένα μαθήματα στην εφαρμογή. Μόλις συγκεντρωθεί η πληροφορία αυτή τα ενσωματώνει σε μια λίστα. Την στιγμή λοιπόν που ο χρήστης καταχωρήσει το όνομα ενός νέου μαθήματος τότε ο σχετικός Listener καλείται. Αυτός με την σειρά του πραγματοποιεί ελέγχους εγκυρότητας της καταχώρησης που έκανα ο χρήστης όπως το αν η εγγραφή είναι κενή ή αν υπάρχει ήδη καταχωρημένη στην λίστα με τα τμήματα της εφαρμογής που ανακτήθηκε προηγουμένως. Σε κάθε περίπτωση ενημερώνει τον χρήστη με σχετικό μήνυμα. Στην περίπτωση τώρα που οι έλεγχοι ολοκληρωθούν επιτυχώς τότε η καταχώρηση πραγματοποιείται και σχετικό μήνυμα εμφανίζεται στον χρήστη. Λειτουργικότητα επιστροφής στο βασικό μενού του διαχειριστικού προσφέρεται και στην οθόνη αυτή επιστρέφοντας μαζί και το αναγνωριστικό χρήστη.

- 20. AddRelationActivity:** Η οθόνη αυτή περιλαμβάνει την λειτουργικότητα για την δημιουργία μιας συσχέτισης μεταξύ ενός καταχωρημένου Πανεπιστημίου, ενός Τμήματος και ενός Μαθήματος τα οποία θα εμφανίζονται ανάλογα με βάση την συσχέτιση αυτή στις λίστες της εφαρμογής. Όταν ο διαχειριστής επιλέξει από το διαχειριστικό μενού την λειτουργικότητα αυτή ο κώδικας οδηγεί σε αυτή την οθόνη. Με το που εκτελείτε λοιπόν το αρχείο αυτό λαμβάνεται και σε αυτή την οθόνη το αναγνωριστικό χρήστη και προετοιμάζονται οι Listeners αλλά και τα γραφικά αυτής. Υστέρα γίνεται επικοινωνία μέσω της τεχνολογίας firebase με την βάση δεδομένων και συγκεντρώνεται σε λίστες και εμφανίζονται στην οθόνη του χρήστη τα καταχωρημένα ως τώρα Πανεπιστήμια, Τμήματα και Μαθήματα στην εφαρμογή. Αφού ο χρήστης επιλέξει έναν συνδυασμό μεταξύ Πανεπιστημίου, Τμήματος και Μαθήματος τον οποίο θέλει να συσχετίσει μεταξύ τους πατώντας το κουμπί της καταχώρησης καλεί το απαραίτητο για την ενέργεια αυτή Listener. Στο σημείο αυτό γίνεται πάλι επικοινωνία με την βάση στο firebase και ελέγχεται αν ο συγκεκριμένος συνδυασμός που επιλέχθηκε από τον χρήστη υπάρχει ήδη καταχωρημένος. Στην περίπτωση που αυτός υπάρχει σχετικό μήνυμα εμφανίζεται και ενημερώνει τον χρήστη. Σε αντίθετη περίπτωση πραγματοποιείται ενέργεια καταχώρησης και μόλις αυτή ολοκληρωθεί ενημερώνει τον χρήστη για την επιτυχημένη εξέλιξη της. Τελειώνοντας και με την οθόνη αυτή και εδώ υπάρχει κουμπί επιστροφής στο διαχειριστικό μενού που όταν κληθεί ανακατευθύνει τον χρήστη πίσω στο διαχειριστικό μενού επιστρέφοντας μαζί και το αναγνωριστικό εισόδου του.
- 21. DeleteRelationActivity:** Στο Activity αυτό υπάρχει όλη η απαραίτητη λειτουργικότητα για την διαγραφή μία υπάρχουσας συσχετίσεις μεταξύ Πανεπιστημίου, Τμήματος και Μαθήματος που εμφανίζεται στις λίστες της εφαρμογής. Όταν ο διαχειριστής επιλέξει την συγκεκριμένη λειτουργικότητα από το μενού τότε το συγκεκριμένο αρχείο εκτελείτε. Ειδικότερα λαμβάνεται το αναγνωριστικό χρήστη και εν συνεχεία προετοιμάζονται οι listeners που είναι απαραίτητη για την λειτουργία της οθόνης αυτής και αρχικοποιούνται τα γραφικά στοιχεία της. Επιπρόσθετα πραγματοποιείται μια επικοινωνία με την βάση στο firebase όπου ανατρέχετε ο σχετικός πίνακας με τις υπάρχουσες συσχετίσεις προκειμένου να συγκεντρωθεί η απαραίτητη πληροφορία. Αφού αυτό συμβεί ενημερώνονται οι λίστες και τα check box της οθόνης αυτής με τα δεδομένα να εμφανίζονται με τέτοιο τρόπο ώστε να είναι ορατές οι συσχετίσεις που έχουν γίνει. Για τον λόγο αυτό κάθε φορά που ο χρήστης πραγματοποιεί μια επιλογή σε ένα check box ανάλογα πραγματοποιείται και μια σειρά άλλων ενεργειών που



επηρεάζουν και τις υπόλοιπες επιλογές κάτω από αυτή. Από την στιγμή που ο χρήστης αποφασίσει και πραγματοποιήσει τις επιλογές που θέλει να διαγράψει για το πετύχει αυτό αρκεί να επιλέξει μόνο το κουμπί της διαγραφής. Σε εκείνο το σημείο ο σχετικός για την λειτουργικότητα listener καλείτε. Αυτό πρακτικά σημαίνει ότι ανάλογα με τις επιλογές που έχουν γίνει από τον χρήστη ο κώδικας πραγματοποιεί διαγραφές των αντίστοιχων επιλεγμένων πεδίων. Με την ολοκλήρωση των διαγραφών μήνυμα επιτυχίας εμφανίζεται στον χρήστη. Σε αντίθετη περίπτωση πάλι τον πληροφορεί σχετικά με το ζήτημα που μπορεί να προέκυψε. Ολοκληρώνοντας την λειτουργικότητα της σθόνης αυτής και εδώ παρέχεται η δυνατότητα επιστροφής στο διαχειριστικό μενού πατώντας το κουμπί πίσω όπου παράλληλα στέλνεται μέσω Intent και ο αναγνωριστικός κωδικός του χρήστη.

#### 5.4 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

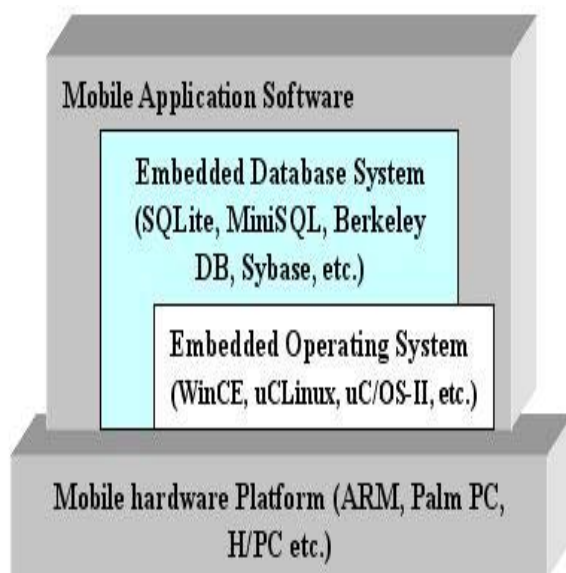
Με τον όρο βάση δεδομένων εννοείται μία συλλογή από *συστηματικά μορφοποιημένα* σχετιζόμενα δεδομένα στα οποία είναι δυνατή η ανάκτηση δεδομένων μέσω αναζήτησης κατ' απαίτηση. Με άλλα λόγια θα μπορούσαμε να αποδώσουμε τον όρο βάση δεδομένων (database) με μια απλούστερη έννοια αναφέροντας ότι είναι μια συλλογή από εγγραφές και αρχεία, τα οποία είναι οργανωμένα έτσι ώστε να εξυπηρετούν ένα συγκεκριμένο σκοπό. Στην παρούσα λοιπόν πτυχιακή εργασία γίνεται χρήση βάσης δεδομένων προκειμένου να αποθηκευτεί, να ανακτηθεί, να ενημερωθεί ή να διαγραφεί η απαραίτητη για την εφαρμογή πληροφορία. Ωστόσο στην εργασία που εξετάζουμε δεν έχει γίνει χρήση απλής SQL βάσης δεδομένων αλλά SQLite εφόσον η εφαρμογή τρέχει σε λειτουργικό σύστημα Android και απευθύνεται σε κινητές συσκευές καθώς και μία NoSQL cloud βάση δεδομένων μέσω της τεχνολογίας Firebase Realtime Database την οποία και θα αναλύσουμε παρακάτω.

#### 5.5 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ SQLite

Ξεκινώντας λοιπόν την ανάλυση μας σε ότι αφορά τις βάσεις δεδομένων που χρησιμοποιήθηκαν στην παρούσα πτυχιακή εργασία αναφερόμαστε στην ενσωματωμένη βάση δεδομένων SQLite. Η ενσωματωμένη βάση δεδομένων SQLite εφαρμόζεται ευρέως στη διαχείριση δεδομένων των ενσωματωμένων περιβάλλον, όπως οι κινητές συσκευές, ο βιομηχανικός έλεγχος και αλλά έχοντας γίνει το επίκεντρο της ανάπτυξης συναφών τομέων. Βασικός λόγος αυτής της έξαρσης προς την συγκεκριμένη τεχνολογία όσον αφορά τις βάσεις δεδομένων αποτελούν τα πλεονεκτήματα που προσφέρει αυτή σε σχέση με την σταθερότητα, την αξιοπιστία, τις γρήγορες και υψηλές αποδόσεις και την φορητότητα της που την καθιστούν μοναδική μεταξύ ενός μεγάλου πλήθους κύριων ενσωματωμένων βάσεων δεδομένων. Η πρόοδος της τεχνολογίας γενικότερα και ειδικότερα των ευφυών συσκευών δικτύου, των κινητών συσκευών και εφαρμογών κατέστησαν αναγκαία την εξέλιξη των ενσωματωμένων λειτουργικών συστημάτων και ως επακόλουθο αυτής και την εξέλιξη στις βάσεις δεδομένων σύμφωνα με τις απαιτήσεις που δημιούργησε η απαίτηση για βάση δεδομένων κινητής τηλεφωνίας. Τα τελευταία χρόνια μεγάλη προσπάθεια σημειώθηκε από διάφορες εταιρίες ώστε οι κλασικές βάσεις δεδομένων να καλύψουν την ανάγκη αυτή που παρουσιάστηκε. Αποτέλεσμα αυτού ήταν να εμφανίζονταν στην αγορά προϊόντα βάσεων δεδομένων όπως IBM DB2, Oracle, Sybase, SQL Server και ούτω καθεξής χωρίς όμως να επιτυγχάνεται άμεσο αποτέλεσμα στην ανάγκη που



προέκυψε. Ο λόγος αυτών των αποτυχημένων προσπαθειών βασίζεται κυρίως στο γεγονός ότι οι βάσεις αυτές περιορίζοντας από χωρητικότητα και κατανάλωση ισχύος ενσωματωμένων συστημάτων κάτι που γινόταν ακόμα πιο εμφανές όταν οι κλίμακες των χρηστών κινητών συσκευών γίνονταν όλο ένα και πιο μεγάλες. Οι παραδοσιακές βάσεις δεδομένων φάνταζαν όλο και πιο αργές και η απαίτηση τους για υπολογιστική μνήμη γινόταν μεγαλύτερη σε μία εποχή που καθιστούσε αναγκαία την απαίτηση δεδομένων σε πραγματικό χρόνο. Την απαίτηση λοιπόν αυτή που προέκυψε ήρθαν να εξαλείψουν οι ενσωματωμένες βάσεις δεδομένων εμφανίζοντας εξαιρετικές αποδόσεις σε συναφή πεδία σε σχέση με τις κλασικές βάσεις δεδομένων που γνωρίζαμε μέχρι τώρα. Επί του παρόντος, έχει εμφανιστεί μεγάλος αριθμός ενσωματωμένων βάσεων δεδομένων όπως Xiao-Jin-Ling της KingbaseLite, OpenBASE Mini της Neusoft Group, sybase iAnywhere της Sybase, OracleLite της Oracle αλλά και άλλες όπως Mysql, Berkeley DB, SQLite, και ούτω καθεξής. Ωστόσο το ενδιαφέρον πολλών προγραμματιστών στις μέρες μας έχει προσελκύσει η SQLite λόγω των πλεονεκτημάτων που προσφέρει όπως το πόσο ελαφρύ προϊόν είναι, η εύκολη φορητότητα της καθώς και ότι από την στιγμή που ανήκει στα προϊόντα ανοικτού κώδικα (Open Source) δεν δεσμεύεται από πνευματικά δικαιώματα και περιορισμούς. Με τον όρο ενσωματωμένη βάση δεδομένων αναφερόμαστε σε ένα σύστημα διαχείριση βάσεων δεδομένων που εξειδικεύεται κυρίως στην κινητή υπολογιστική. Οι βάσεις αυτές συνήθως βρίσκονται ενσωματωμένες στο λειτουργικό σύστημα ή εμπεριέχονται μέσα στις εφαρμογές και κυρίως «τρέχουν» σε ενσωματωμένες ή κινητές συσκευές. Για τον λόγο αυτό αλλά και επειδή είναι άμεσα αλληλένδετες με την κινητή υπολογιστική αποκαλούνται και ως ενσωματωμένες βάσεις για κινητά. Παρόλα αυτά οι ενσωματωμένες τεχνολογίες βάσεις δεδομένων έχουν επηρεάσει με την παρουσία τους πολλούς τομείς πέρα από αυτόν των κινητών συσκευών όπως τις βάσεις γενικότερα, τα συστήματα πραγματικού χρόνου, την κατανεμημένη υπολογιστική τεχνολογία και την τεχνολογία κινητών επικοινωνιών δημιουργώντας καινούργια πεδία έρευνας και ανάπτυξης. Η δομή ενός ενσωματωμένου συστήματος απεικονίζεται στην παρακάτω εικόνα.



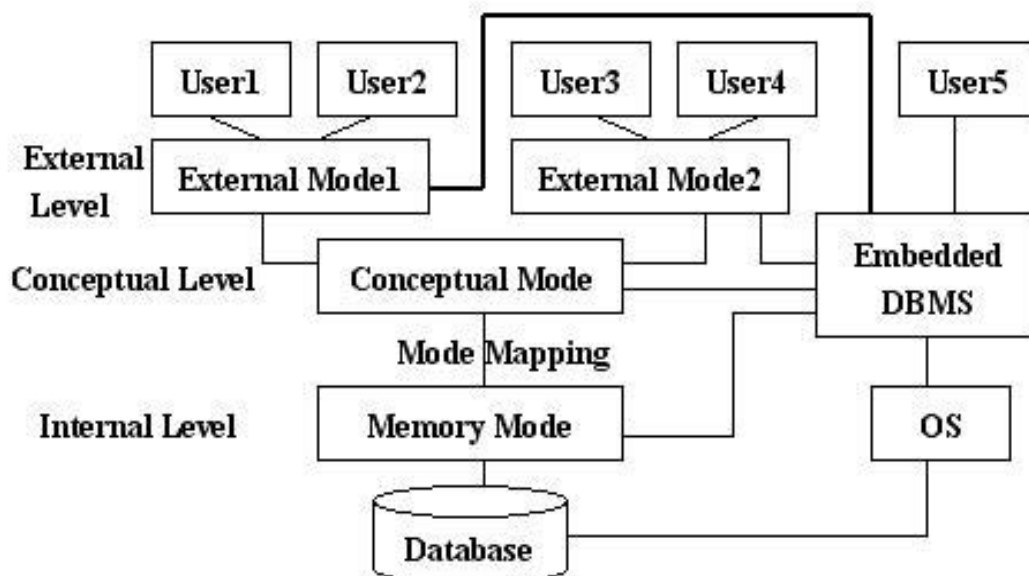
(Research and Application of SQLite Embedded Database Technology CHUNYUE BI School of Computer Science and Information

Technology Zhejiang Wanli University No.8, South Qian Hu Road Ningbo, Zhejiang P.R.CHINA <http://www.zwu.edu.cn>)

Σύμφωνα με την παραπάνω εικόνα παρατηρείται ότι μια ενσωματωμένη βάση δεδομένων ως ενδιάμεσο ενσωματωμένο λογισμικό είναι υποχρεωμένη να περιορίζεται από την ταχύτητα, τους



πόρους αλλά και τις εφαρμογές των ενσωματωμένων συστημάτων που εμπεριέχεται. Το ενσωματωμένο σύστημα βάσης δεδομένων είναι ουσιαστικά παρόμοιο με το συνηθισμένο σύστημα βάσης δεδομένων, καθώς μπορεί να είναι και αυτές δικτυακές, είναι σχεσιακές και είναι και αντικειμενοστραφής. Η βασική διαφορά τους όσον αφορά τον τρόπο που λειτουργούν είναι ότι οι ενσωματωμένες βάσεις καλούν την αντίστοιχη λειτουργία API του συστήματος που εμπεριέχονται προκειμένου να αποκτήσουν πρόσβαση στα δεδομένα ενώ οι παραδοσιακές βάσεις που δεν εμπεριέχονται σε κάποιο σύστημα ανακτούν τα δεδομένα από μόνες τους. Παρόλα αυτά αρχιτεκτονικά οι ενσωματωμένες βάσεις δεδομένων είναι παρόμοιες με τις κλασσικές καθώς αποτελούνται από τρία μέρη: 1. το εσωτερικό επίπεδο, 2. το εννοιολογικό επίπεδο 3. το εξωτερικό επίπεδο.



(Research and Application of SQLite Embedded Database Technology CHUNYUE BI School of Computer Science and Information

Technology Zhejiang Wanli University No.8, South Qian Hu Road Ningbo, Zhejiang P.R.CHINA <http://www.zwu.edu.cn>)

Αναλύοντας τα επίπεδα αυτά και ξεκινώντας από το εννοιολογικό επίπεδο θα μπορούσαμε να πούμε πως περιλαμβάνει μία κοινή λογική περιγραφή των βάσεων αυτών ώστε να μπορούν να επικοινωνούν τα υπόλοιπα επίπεδα μεταξύ τους κατά τον ίδιο τρόπο και είναι άμεσα εξαρτώμενο από τα υπόλοιπα επίπεδα. Όσον αφορά το εξωτερικό επίπεδο αποτελεί την διεπαφή μεταξύ χρηστών, DBS και προγραμματιστών που βοηθάει στην αποθήκευση και ανάκτηση δεδομένων σύμφωνα με τους κανόνες και την ορολογία που ορίζει το εννοιολογικό επίπεδο. Τέλος το εσωτερικό επίπεδο είναι αυτό που έχει άμεση επικοινωνία με το υλικό και αυτό μέσω του οποίου δίνονται, εκτελούνται αλλά και ορίζονται οι εντολές που έχουν ζητηθεί από τα πιο πάνω επίπεδα. Παρόλα αυτά κάτι που διακρίνει αυτού του είδους τις βάσεις είναι ότι σε περιπτώσεις που οι απαιτήσεις σε πραγματικό χρόνο είναι πολύ υψηλές πραγματοποιείται επικοινωνία άμεσα με το εσωτερικό επίπεδο παρακάμπτοντας τα άλλα δύο στάδια δηλαδή την εκτέλεση χαρτογράφησης από το εξωτερικό επίπεδο στο εννοιολογικό και από αυτό σε λειτουργία μνήμης. Γενικά το ενσωματωμένο σύστημα διαχείρισης βάσης δεδομένων (DBMS) είναι ένα σύστημα λογισμικού το οποίο διαχειρίζεται τα δεδομένα στο ενσωματωμένο σύστημα βάσης δεδομένων, και το οποίο αποτελείται από τρεις προγραμματιστικές λειτουργίες δηλαδή το πρόγραμμα επεξεργασίας του μεταγλωττιστή, το πρόγραμμα ελέγχου λειτουργίας του



συστήματος και το πρόγραμμα εξυπηρέτησης. Επιγραμματικά αναφέρουμε για τις παραπάνω λειτουργίες:

1. **Πρόγραμμα επεξεργασίας γλώσσας μεταγλωττιστή:** Περιλαμβάνει το πρόγραμμα επεξεργασίας γλώσσας σε όλα τα επίπεδα, το πρόγραμμα επεξεργασίας μεταγλωττιστή γλώσσας χειρισμού βάσεων δεδομένων και το πρόγραμμα επεξήγησης γλώσσας επερώτησης (query).

2. **Πρόγραμμα ελέγχου λειτουργίας του συστήματος:** Αποτελεί την καρδιά του ενσωματωμένου προγράμματος DBMS, χρησιμοποιείται για τον έλεγχο και τον συντονισμό των δραστηριοτήτων του προγράμματος, συμπεριλαμβανομένου του προγράμματος παράλληλου ελέγχου, του προγράμματος προγραμματισμού σε πραγματικό χρόνο, του προγράμματος επεξεργασίας συναλλαγών, του προγράμματος ενημέρωσης της βάσης δεδομένων και του προγράμματος διαχείρισης αποθήκευσης.

3. **Πρόγραμμα εξυπηρέτησης:** Συμπεριλαμβάνει το πρόγραμμα αποκατάστασης σφαλμάτων, το αρχείο καταγραφής προγράμματος εργασιών και προγράμματος αναδιοργάνωσης βάσης δεδομένων. Ωστόσο από όλες τις ενσωματωμένες βάσεις που υπάρχουν αυτή την στιγμή στην αγορά στην παρούσα πτυχιακή εργασία έγινε χρήση της SQLite η οποία έχει κεντρίσει το ενδιαφέρον πολλών προγραμματιστών για τα πλεονεκτήματα που προσφέρει. Κάνοντας χρήση της ονομασίας SQLite αναφερόμαστε σε μια ενσωματωμένη βάση δεδομένων ανοιχτού κώδικα η οποία έχει βασιστεί σε διάφορες γλώσσες όπως C, C++, BASIC, C#, Python, Java και Delphi παρότι είναι γραμμένη σε λιγότερες από 30,000 γραμμές σε γλώσσα C και ο δημιουργός αυτής ονομάζεται D. Richard Hipp. Με απλά λόγια θα μπορούσαμε να αναφέρουμε ότι η SQLite αποτελεί μια βιβλιοθήκη που εξελίσσεται και υλοποιεί ένα αυτοματοποιημένο μηχανισμό βάσης δεδομένων με μηδενικές παραμέτρους και χωρίς την απαίτηση κάποιου διακομιστή. Μιλάμε για μια βάση δεδομένων που αποδεικνύεται πιο προσβάσιμη σε γλώσσες script όπως η VB των Windows ή η JavaScript δίνοντας έτσι ακόμα περισσότερες δυνατότητες ανάπτυξης σε εφαρμογές HTML. Ωστόσο η χρήση της SQLite δεν περιορίζεται μόνο σε τέτοιου είδους εφαρμογές αλλά είναι διαθέσιμη και σε ενσωματωμένα λειτουργικά συστήματα όπως IOS, Android, Symbian OS, Maemo, Blackberry και WebOS, λόγω του μικρού μεγέθους και της ευκολίας χρήσης. Ο αριθμός των υπηρεσιών κινητής τηλεφωνίας αυξήθηκε κατά παράγοντες κατά την τελευταία δεκαετία και η μέση χρήση των φορητών συσκευών εδραίωσε πολλαπλά κοινωνικά δίκτυα και εφαρμογές μηνυμάτων στην καθημερινότητα ενός χρήστη που πλέον τα χρησιμοποιεί όχι μόνο για επαγγελματικό σκοπό αλλά και ως μέσω επικοινωνίας και διασκέδασης. Η ανάγκη λοιπόν που προέκυψε για περισσότερο αποθηκευτικό χώρο στις κινητές συσκευές, οι προτιμήσεις των χρηστών αλλά και άλλα τμήματα δεδομένων άσκησαν πίεση στις κατασκευάστριες εταιρίες κινητών συσκευών που επωμίστηκαν να αντιμετωπίσουν το πρόβλημα φόρτου εργασίας και να υποστηρίξουν την επιθυμητή εμπειρία χρήσης που προσδοκούσαν οι χρήστες των συσκευών αυτών. Η βιομηχανία προκειμένου να αντιμετωπίσει το πρόβλημα που παρουσιάστηκε προσέφερε μια σειρά λύσεων μέσα σε αυτές και αυτή της SQLite. Η SQLite είναι ένα ευρέως υιοθετημένο σύστημα διαχείρισης κινητών βάσεων δεδομένων (Kreibich, 2010). Εφαρμόζει το σχεσιακό μοντέλο βάσεων δεδομένων και υποστηρίζει σε πλήρη βαθμό το βιομηχανικό πρότυπο της SQL γλώσσας. Όλες οι μεγάλες βιομηχανίες λειτουργικών συστημάτων για κινητές συσκευές στην αγορά την υποστηρίζουν, ενώ το Android έρχεται με την προ εγκατεστημένη έκδοση από προεπιλογή. Ωστόσο σε αυτό το σημείο γεννάται το ερώτημα τί ήταν αυτό που έκανε την SQLite να διακριθεί ανάμεσα στις υπόλοιπες λύσεις που παρουσιάστηκαν κατά καιρούς και την έκανε να εδραιωθεί ως την βασική λύση στο πρόβλημα που προέκυψε. Κάτι τέτοιο μπορεί λοιπόν να γίνεται αντιληπτό αναφέροντας τα κύρια χαρακτηριστικά αυτής που την ξεχώρισαν σε σχέση με τις υπόλοιπες αλλά και τα πλεονεκτήματα που παρέχει σε σύγκριση με τις συμβατικές βάσεις δεδομένων. Ξεκινώντας λοιπόν με τα βασικά χαρακτηριστικά αυτής αναφέρουμε παρακάτω τα εξής:

- Η SQLite αποτελεί ένα προϊόν ανοιχτού κώδικα (open source) κάτι που σημαίνει σε πρώτο στάδιο ότι μειώνει το κόστος κατασκευής ενός προϊόντος αλλά παρέχοντας επίσης την δυνατότητα τροποποιήσεως της σε ένα βαθμό γεγονός





που εξασφαλίζει με αυτό τον τρόπο την συντήρηση και την σταθερότητα των εκάστοτε προϊόντων που κάνουν χρήση αυτής ανάλογες με τις ανάγκες που παρουσιάζουν το καθένα.

- Η SQLite αποτελεί ένα ατομικό, σταθερό, απομονωμένο και ανθεκτικό σύστημα βάσεων δεδομένων που παραμένει λειτουργικό ακόμα και σε διακοπές του βασικού συστήματος ή του ρεύματος.
- Η SQLite δεν απαιτεί καμία ενέργεια εγκατάστασης ή ρύθμισης, δεν απαιτεί την χρήση κάποιου νήματος (thread) προκειμένου να ξεκινήσει ή να σταματήσει κάτι που πρακτικά θα σήμαινε ανάγκη για περισσότερους πόρους αλλά ούτε και την δημιουργία βάσης δεδομένων αλλά και λογαριασμό χρήση για πρόσβαση σε αυτή με δικαιώματα δοσμένα από τον διαχειριστή του συστήματος.
- Η SQLite μπορεί να διαβάσει και να γράφει αρχεία βάσης δεδομένων στον σκληρό δίσκο απευθείας και δεν χρειάζονται επιπλέον διαδικασία εξυπηρέτησης. Ωστόσο και μια πλήρης βάση δεδομένων αντιστοιχεί σε αρχεία στο δίσκο. Μπορούν να χρησιμοποιηθούν τα ίδια αρχεία βάσης δεδομένων από διαφορετικά μηχανήματα, αλλά μπορούν επίσης να μοιραστούν ελεύθερα μεταξύ αυτών.
- Υποστηρίζει βάσεις δεδομένων μεγέθους terabyte.
- Δεν χρειάζεται τύπους δεδομένων. Αυτό σημαίνει ότι μπορεί να υποστηρίξει οποιαδήποτε δεδομένα χωρίς να πρέπει να δηλωθεί ο τύπος του ή να πρέπει να δηλωθούν σε συγκεκριμένο πίνακα ή κελί.
- Είναι άμεσα προσβάσιμη μέσω του αντίστοιχου API.
- Είναι γρήγορη, αποδοτική με δυνατότητα αναβαθμίσεις και δεν βασίζεται σε κανένα λειτουργικό σύστημα παρότι είναι συμβατή με ένα μεγάλο αριθμό λειτουργικών συστημάτων (Linux, Android, iOS, Windows και ούτω καθεξής.)
- Παρέχει πολλαπλή υποστήριξη για το API και υποστηρίζει κύριες γλώσσες προγραμματισμού συμπεριλαμβανομένων αυτές C, C++, PHP, Java, Perl οι οποίες για να κάνουν χρήση της SQLite επικοινωνούν με τα αρχεία βάσεων δεδομένων μέσω του API.
- Διατίθεται ως κώδικας ANSI-C και είναι εύκολη να προστεθεί σε μεγαλύτερα έργα.

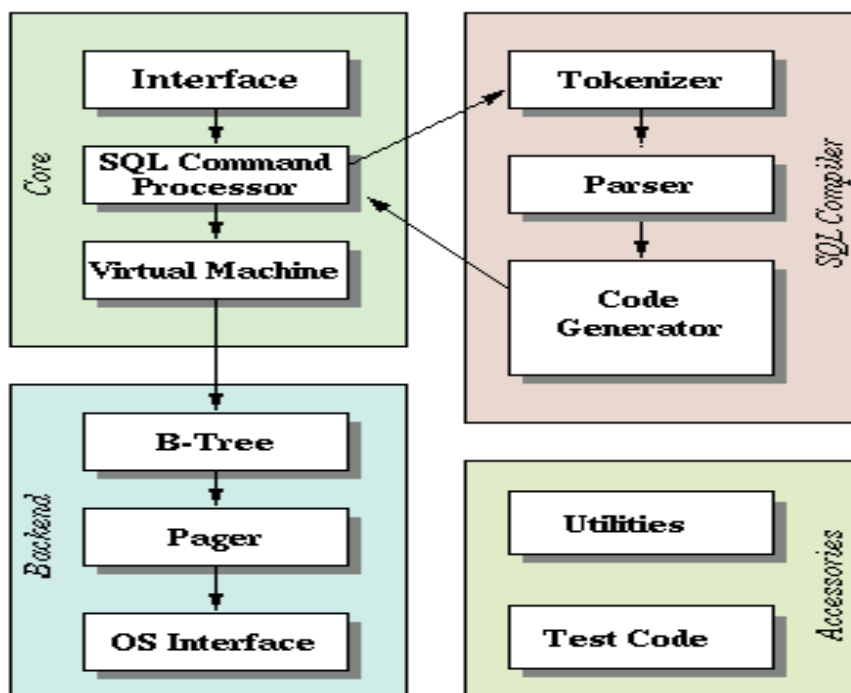
Παρόλα αυτά όπως ήδη αναφέραμε η SQLite ξεχωρίζει και για τα πλεονεκτήματα που παρουσιάζει σε σύγκριση και με τις συμβατικές βάσεις δεδομένων όπως για παράδειγμα την SQL ή την MySQL. Παρακάτω αναφέρονται επιγραμματικά ορισμένα από τα χαρακτηριστικά που την κάνουν ξεχωριστή έναντι αυτών.

- Ο SQL Server και η MySQL βασίζονται σε διακομιστές, αλλά η SQLite βασίζεται σε αρχεία.
- Η SQLite είναι ενσωματωμένο σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων ενώ η SQL είναι η γλώσσα ερωτήσεων και η MySQL είναι client Server σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων.
- Λόγο του ότι αποτελεί ένα ελαφρύ προϊόν (SQLite) αλλά και γιατί βασίζεται στο σχεσιακό μοντέλο διαχείρισης βάσεων δεδομένων μπορεί να χρησιμοποιηθεί σε εφαρμογές κινητών συσκευών.
- Η SQL βασίζεται στην σχεσιακή άλγεβρα. Η SQLite, η MySQL και η MSSQL είναι μέρος της SQL και όλες χρησιμοποιούν την σύνταξη της SQL παρόλα αυτά η κάθε μία έχει την δικές της συμβάσεις και τα δικά της χαρακτηριστικά.



- Δεδομένου ότι η SQLite είναι μια ελαφριά βάση δεδομένων αποτελείται από έναν απευθείας μηχανισμό διαχείρισης αρχείων που χρησιμοποιεί την σύνταξη που επιβάλλει η SQL. Επίσης η SQLite δεν απαιτεί κάποιο διακομιστή βάσεων δεδομένων οι οποιοδήποτε άλλη εξάρτηση.
- Τέλος οι συμβατικές βάσεις δεδομένων όπως SQL Server και η MySQL υποστηρίζουν αποθηκευτικές διαδικασίες (store procedures) ενώ η SQLite όχι.

Ενδιαφέρον ωστόσο παρουσιάζει η SQLite και από αρχιτεκτονικής άποψης. Πρόκειται για μια αρχιτεκτονική σχετικά απλή και μια σύντομη περιγραφή είναι απαραίτητη για κατανόηση κάποιος την λειτουργία και τις δυνατότητες που παρέχει στους χρήστες της. Ο σχεδιασμός της SQLite ακολουθεί την ιεραρχική σχεδιαστική σκέψη της μηχανικής λογισμικού, παίρνει αρθρωτό σχεδιασμό, για να διευκολύνει τις αναβαθμίσεις συντήρησης, αλλά και για ενσωματώνετε εύκολα σε διαφορετικές πλατφόρμες, είναι απαραίτητο να διαιρέσει ολόκληρο το σύστημα βάσεων δεδομένων σε διάφορες μονάδες διαφορετικού επιπέδου. Θα μπορούσαμε να πούμε ότι στο σύνολο της μπορεί να χωριστεί σε οκτώ βασικά υποσυστήματα: Interface, Tokenizer, Parser, Generator Code, Virtual Machine, B-Tree, Pager και OS.



ISSN 2320-088X SQLite: Light Database System S.T. Bhosale, Miss. Tejaswini Patil, Miss. Pooja Patil



Το βασικό τμήμα περιλαμβάνει τη διεπαφή χρήστη, τον επεξεργαστή εντολών SQL και το εικονικό μηχανήμα. Η διεπαφή χρήστη αποτελείται από μια βιβλιοθήκη λειτουργιών και δομών C, όπου ακόμα και αν χρησιμοποιείται διαφορετικό API γλώσσας, αυτές εξακολουθούν να εκτελούνται με βιβλιοθήκη C στο κάτω μέρος για να χειρίζονται λειτουργίες όπως η αρχικοποίηση βάσεων δεδομένων, η εκτέλεση ερωτημάτων και η εξέταση των αποτελεσμάτων. Οι κλήσεις λειτουργιών που εκτελούν ερωτήματα SQL χρησιμοποιούν τον επεξεργαστή εντολών SQL. Ο επεξεργαστής εντολών λειτουργεί ακριβώς όπως ένας μεταγλωττιστής. Οι εντολές SQL λαμβάνονται από την διεπαφή χρήστη και από εκεί πηγαινούν στο μεταγλωττιστή SQL και αποσυντίθενται σε διάφορα αναγνωριστικά με την βοήθεια του Tokenizer. Κατά την εκτέλεση ενός προγράμματος, η εικονική μηχανή κατευθύνει τη ροή ελέγχου μέσω μιας εναλλασσόμενης κατάστασης, η οποία μεταβαίνει σε ένα μπλοκ κώδικα που βασίζεται στον τρέχοντα κωδικό εκτέλεσης. Εν συνεχεία ο μεταγλωττιστής αποτελείται από τρία ανεξάρτητα στοιχεία, δηλαδή τον Tokenizer, τον Parser και τον Generator Code. Όταν πρόκειται να εκτελεστούν συμβολοσειρές που περιέχουν εντολές SQL, η διεπαφή μεταφέρει αυτές τις συμβολοσειρές στον tokenizer. Αυτός με την σειρά του σπάει την αρχική συμβολοσειρά σε tokens και τις περνάει μια προς μια στον Parser (Αναλυτή). Ο Tokenizer λοιπόν μαζί με τον Parser λαμβάνουν τις δηλώσεις SQL και σε συνεργασία τις μετατρέπουν σε δομές δεδομένων που μπορούν να γίνουν κατανοητές από το κατώτερο επίπεδο δηλαδή τον Code Generator. Συνήθως η δομή δεδομένων είναι δεντρική και μεταφέρεται από τον Parser στον Code Generator. Με την μεταφορά της πλέον κατανοητής για τον Code Generator δομής δεδομένων σε γλώσσα assembly αυτός με την σειρά του αφού την λάβει παράγει σε κώδικα εικονικής μηχανής τα SQL statements που ζητήθηκαν μέσω της δομής δεδομένων από τα πιο πάνω επίπεδα προκειμένου να είναι σε θέση να εκτελεστούν από την εικονική μηχανή. Η εικονική μηχανή είναι η καρδιά της εσωτερικής δομής του SQLite, ονομάζεται επίσης Virtual Engine Engineer (VDBE), ο οποίος είναι ένας μηχανισμός σχεδιασμένος για να ασχολείται με αρχεία βιβλιοθήκης. Το VDBE εκτελεί όχι μόνο όλες τις λειτουργίες που σχετίζονται με τη χειραγώγηση των δεδομένων αλλά θα μπορούσαμε να πούμε πώς αναλαμβάνει και το ρόλο του διαμεσολαβητή μεταξύ του πελάτη και του χώρου αποθήκευσης. Γενικότερα το εικονικό μηχανήμα έχει σχεδιαστεί κατά τέτοιο τρόπο ώστε να μπορεί να επεξεργαστεί τα δεδομένα που λαμβάνει από τον Code Generator. Με άλλα λόγια περιλαμβάνει ένα σύνολο εντολών τέτοιο ώστε να μπορεί να ολοκληρώσει ειδικές λειτουργίες βάσης δεδομένων όπως η εισαγωγή, η επερώτηση (query), η επεξεργασία συναλλαγών και ούτω καθεξής. Η εικονική μηχανή λοιπόν αναλαμβάνει δράση όπως ήδη αναφέραμε και πιο πάνω από την στιγμή που ο Code Generator της στείλει τα αναλυμένα SQL statements. Αυτή η ανάλυση που γίνεται από τον Code Generator δεν είναι τίποτε άλλο από ένα μίνι πρόγραμμα το οποίο αποτελείται από μια σειρά οδηγιών κατανοητές από την εικονική μηχανή. Έτσι λοιπόν το VDBE εκτελεί μια προς μία την κάθε εντολή αυτού του προγράμματος και το οποίο ολοκληρώνει την εκτέλεση του από το VDBE με την εκπλήρωση οπουδήποτε αιτήματος όριζε το SQL statement που ζητήθηκε εξ αρχής. Όσον αφορά τώρα το Backend περιλαμβάνει τρία μέρη: B-Tree, Pager και το OS. Μια βάση δεδομένων SQLite διατηρείται στο δίσκο χρησιμοποιώντας μια δομή δεδομένων B-Tree. Για κάθε πίνακα και ευρετήριο λοιπόν στην βάση αυτή χρησιμοποιείται και ένα ξεχωριστό B-Tree και όλα μαζί αποθηκεύονται στο ίδιο αρχείο δίσκου. Η χρήση της δεντρικής δομής B-Tree εξασφαλίζει πέρα από την γρήγορη αναζήτηση και αποθήκευση την διατήρηση των περίπλοκων σχέσεων μεταξύ των σελίδων δεδομένων αλλά και την σύνδεση μεταξύ αυτών, έτσι ώστε κάθε μέρος τους να είναι πολύ εύκολο να βρεθεί. Τώρα όσον αφορά τον Pager το καθήκον αυτού είναι η μεταφορά της διαχείρισης, η ανάγνωση των δεντρικών δομών, η εισαγωγή δεδομένων σε αυτές, η προσωρινή αποθήκευση, το κλείδωμα της βάσης δεδομένων και η ανάκτηση αυτής σε περιπτώσεις κατάρρευσης σε συνεργασία με το κατώτερο επίπεδο δηλαδή την διεπαφή του λειτουργικού συστήματος (OS). Η διεπαφή του λειτουργικού συστήματος από την άλλη δεν είναι τίποτε άλλο από ένα αφηρημένο interface που χρησιμοποιείται για να διευκολυνθεί η φορητότητα της SQLite από το ένα λειτουργικό σύστημα σε ένα άλλο αλλά και για προστατέψει ζωτικής σημασίας αρχεία της SQLite από τις διαφορές που παρουσιάζει το εκάστοτε λειτουργικό σύστημα που την





χρησιμοποιεί. Τελειώνοντας με την αρχιτεκτονική της SQLite συναντάμε το επίπεδο των Accessories που περιλαμβάνει τα Utilities και τον Test Code. Σε ότι αφορά τα Utilities πρόκειται για βοηθητικά προγράμματα που σκοπό τους έχουν όταν και εφόσον χρησιμοποιηθούν να εξαλείψουν κυρίως ορισμένα ζητήματα όπως η κατανομή μνήμης, η σύγκριση τύπων, η λεξική ανάλυση, η αποθήκευση του πίνακα συμβόλων στη γραμματική ανάλυση και η λειτουργία τυχαίου αριθμού. Τώρα όσον αφορά τον Test Code πρόκειται για σενάρια δοκιμών που προσφέρονται ώστε εάν είναι επιθυμητό να μπορεί να δοκιμαστούν και να ελεγχθούν περισσότερες από τις μισές συναρτήσεις που παρέχονται σε ένα χρήστη μέσω της SQLite.

Όπως ήδη έχει αναφερθεί η συγκεκριμένη πτυχιακή από την στιγμή που αυτό που διαπραγματεύεται είναι μια εφαρμογή για κινητές συσκευές σε λειτουργικό σύστημα Android κάνει χρήση της SQLite προκειμένου να διαχειριστή τα δεδομένα που απαιτούνται για να εξασφαλίσει την αναμενόμενη λειτουργικότητα. Παρόλα αυτά μέχρι στιγμής δεν έχει αναφερθεί με πιο τρόπο ακριβώς έχει γίνει χρήση της SQLite στην συγκεκριμένη εργασία. Για το λόγο αυτό παρακάτω η ανάλυση μας γίνεται πιο συγκεκριμένη σε ότι αφορά την χρήση της σε σχέση με την πτυχιακή εργασία. Η εφαρμογή λοιπόν προκειμένου να διαχειριστή τους χρήστες αλλά και τις σημειώσεις αυτών αποθηκεύει ορισμένα από τα δεδομένα της και με την χρήση της SQLite. Προκειμένου λοιπόν να το πετύχει αυτό με την εγκατάσταση της εφαρμογής των σημειώσεων στην εκάστοτε κινητή συσκευή που τρέχει δημιουργεί τους απαραίτητους πίνακες στους οποίους πραγματοποιούνται οι απαιτούμενες ενέργειες που του ορίζουν χρήστες της όπως η εισαγωγή, η διαγραφή και η αναζήτηση των σημειώσεων που επιθυμούν. Την δημιουργία αυτή την επιτυγχάνει εκτελώντας κατά την πρώτη εκτέλεση της εφαρμογής τα ακόλουθα query statements:

- **CREATE TABLE IF NOT EXISTS Assignments (title text NOT NULL,date text NULL,university text NOT NULL,department text NOT NULL,course text NOT NULL,notes text NULL,category text NULL,location text NULL,usrId text NOT NULL);**
- **CREATE TABLE IF NOT EXISTS Lessons(id integer PRIMARY KEY,name text NOT NULL,uni text NOT NULL)**

Έτσι λοιπόν με τα παραπάνω statement δημιουργούνται οι σχετικοί πίνακες που χρειάζεται η εφαρμογή για να κάνει χρήση της SQLite. Το πρώτο Statement είναι αυτό που δημιουργεί τον πίνακα με τις σημειώσεις των χρηστών της εφαρμογής. Αποτελεί έναν από τους βασικούς πίνακες της βάσης για την εφαρμογή καθώς είναι αυτός στον οποίο η εφαρμογή καταγράφει σε τοπικό επίπεδο δηλαδή όσον αφορά την εκάστοτε συσκευή τις σημειώσεις που επεξεργάζεται ο χρήστης. Όπως φαίνεται και στην παραπάνω SQL δήλωση ο πίνακας αυτός περιέχει ως πεδία του το title, το date, το university, το department, το course, το notes, το category, το location, το usrId πεδία τα οποία θα περιγράψουμε παρακάτω αναλυτικότερα.

Assignments	
Όνομα Πεδίου	Περιγραφή
title	Πεδίο τύπου text που δεν επιτρέπεται από την βάση να παραμείνει κενό. Σε αυτό αποθηκεύεται ο τίτλος της σημείωσης.



date	Πεδίο τύπου text που επιτρέπεται από την βάση να παραμείνει κενό. Σε αυτό καταχωρείται μια ημερομηνία δημιουργίας της σημείωσης από τον χρήστη εάν και εφόσον το επιθυμεί.
university	Πεδίο τύπου text που δεν επιτρέπεται από την βάση να παραμείνει κενό. Σε αυτό το πεδίο καταχωρείται το Πανεπιστήμιο στο οποίο διδάσκεται το μάθημα για το οποίο γράφτηκε η σημείωση.
department	Πεδίο τύπου text που δεν επιτρέπεται από την βάση να παραμείνει κενό. Είναι το σχετικό πεδίο στο οποίο καταγράφεται το τμήμα στο οποίο διδάσκεται το μάθημα για το οποίο αναφέρεται η τρέχουσα σημείωση.
course	Πεδίο τύπου text που δεν επιτρέπεται από την βάση να παραμείνει κενό. Το πεδίο αυτό αφορά το μάθημα για το οποίο γράφτηκε η σημείωση.
category	Πεδίο τύπου text που επιτρέπεται από την βάση να παραμείνει κενό. Το πεδίο αυτό καταγράφει μια κατηγορία που ο χρήστης έχει δημιουργήσει και θέλει να εντάξει την σημείωση που δημιούργησε για να μπορεί να την αναζήτηση πιο εύκολα.
location	Πεδίο τύπου text που επιτρέπεται από την βάση να παραμείνει κενό. Αφορά την σχετική τοποθεσία από την οποία αναρτήθηκε η σημείωση και ο χρήστης το ενεργοποιεί ώστε να καταγράψει εάν και εφόσον το επιθυμεί.
notes	Πεδίο τύπου text που επιτρέπεται από την βάση να παραμείνει κενό. Αποτελεί το πεδίο στο οποίο καταχωρείται το κείμενο της σχετικής σημείωσης.
usrId	Πεδίο τύπου text που δεν επιτρέπεται από την βάση να παραμείνει κενό. Στο πεδίο αυτό καταγράφεται το αναγνωριστικό id του χρήστη που έχει αποθηκεύσει στις σημειώσεις του την συγκεκριμένη σημείωση.

Ωστόσο στην εφαρμογή δημιουργείται άλλος ένας πίνακας σε SQLite τον οποίο δημιουργεί το δεύτερο SQL Statement που είδαμε παραπάνω κατά την πρώτη εκτέλεση της εφαρμογής. Ο πίνακας αυτός αφορά την κατηγοριοποίηση που μπορεί να κάνει εάν το επιθυμεί ένας χρήστης σε σχέση με τις σημειώσεις του ώστε να μπορεί να τις αναζητήσει πιο γρήγορα. Το δεύτερο statement λοιπόν όπως φαίνεται και πιο πάνω δημιουργεί όταν εκτελεστεί ένα πίνακα με την ονομασία Lessons ο οποίος περιλαμβάνει πεδία όπως το id, το name και το uni (university). Παρακάτω στον πίνακα που ακολουθεί επεξηγούμε τα πεδία του σχετικού πίνακα.



Lessons	
Όνομα Πεδίου	Περιγραφή
Id	Πεδίο τύπου integer το οποίο αποτελεί Primary Key. Σε αυτό αποθηκεύεται ένα μοναδικό αναγνωριστικό για την κατηγορία που δημιουργείται.
name	Πεδίο τύπου text που δεν επιτρέπεται από την βάση να παραμείνει κενό. Σε αυτό καταχωρείται η ονομασία της νέας κατηγορίας.
Uni	Πεδίο τύπου text που δεν επιτρέπεται από την βάση να παραμείνει κενό. Σε αυτό το πεδίο καταχωρείται το Πανεπιστήμιο στο οποίο έχουν καταγραφεί οι σημειώσεις τις οποίες κατηγοριοποιεί ο χρήστης.

## 5.6 FIREBASE ΤΕΧΝΟΛΟΓΙΕΣ

Στην εφαρμογή όμως όπως αναφέραμε και σε πιο πάνω σημείο της ενότητας αυτής πέρα από την βάση που χρησιμοποιεί «τοπικά» στην εκάστοτε συσκευή με την χρήση της SQLite που ήδη αναφέραμε χρησιμοποιείται και μία βάση στο cloud μέσω μιας τεχνολογία με την ονομασία firebase.



Η συγκεκριμένη τεχνολογία cloud ανήκει στην κατηγορία BaaS (Backed as a Service) αλλά διαφέρει σε σχέση με άλλες που ανήκουν στην ίδια. Η ειδοποιός διαφορά τους είναι στο τρόπο με τον οποίο διαχειρίζονται τα δεδομένα σε πραγματικό χρόνο με την τεχνολογία firebase να παρέχει ενημερώσεις απευθείας την ίδια στιγμή σε όλους τους «πελάτες» που περιμένουν κάποια ενημέρωση. Πρόκειται για μια τεχνολογία API που να παρέχει στους χρήστες της δυνατότητες όπως η βάση δεδομένων, το storage, το authentication, τα notification και άλλες οι οποίες μπορούν να χρησιμοποιηθούν σε οποιοδήποτε λειτουργικό σύστημα και εφαρμογή. Την τεχνολογία αυτή δημιούργησε η εταιρεία που είχαν ο Andrew Lee και ο James Tamplin το 2011 και εξαγοράστηκε στην συνέχεια τον Οκτώβριο του 2014 από την εταιρεία Google η οποία μέχρι σήμερα έχει ενσωματώσει επιπλέον και άλλες λειτουργικές δυνατότητες στην τεχνολογία Firebase. Όπως ήδη αναφέρθηκε η τεχνολογία firebase δεν διαπραγματεύεται μόνο την διαχείριση δεδομένων σε πραγματικό χρόνο από την πλευρά της βάσης δεδομένων αλλά αποτελείται από πολύ περισσότερες λειτουργικές δυνατότητες τις οποίες και σας παρουσιάζουμε στην λίστα παρακάτω:



- **Realtime Database & Cloud Firestore:**

Η βάση δεδομένων στο firebase είναι μια βάση στο cloud συνεπώς μπορεί να είναι προσβάσιμη από οποιοδήποτε μέρος και από οποιονδήποτε άνθρωπο. Πρόκειται για μια βάση NoSQL όπως άλλωστε και οι περισσότερες υπηρεσίες που έχουν να κάνουν με την διαχείριση δεδομένων και ανήκουν στην κατηγορία Backed as a Service. Αυτό με άλλα λόγια σημαίνει ότι όλα της τα αντικείμενα που αποθηκεύονται ή διαβάζονται είναι αντικείμενα JSON σε αυθαίρετη δομή. Πρόκειται για μία αξιόπιστη λύση η οποία είναι αρκετά γρήγορη καθώς διαχειρίζεται δεδομένα σε πραγματικό χρόνο κάτι που σημαίνει ότι τα δεδομένα ενημερώνονται και συγχρονίζονται σε ελάχιστο χρόνο και διατηρούνται αναλλοίωτα ακόμα και αν οι τελικοί παραλήπτες τους (χρήστες) δεν είναι συνδεδεμένη την στιγμή που πραγματοποιείται η ενημέρωση. Όσον αφορά τώρα τον τρόπο λειτουργία αυτής της βάσης αξίζει να αναφέρουμε μερικά πράγματα. Αρχικά η βάση αυτή όπως ήδη είπαμε και παραπάνω αποτελεί μια βάση NoSQL η οποία στην ουσία διαχειρίζεται τα δεδομένα της χρησιμοποιώντας αντικείμενα σε ακανόνιστη μορφή JSON. Οι τύποι δεδομένων που επιτρέπονται να αποθηκευτούν στην συγκεκριμένη δομή δεδομένων περιορίζεται στους εξής τύπους αντικειμένων:

1. String
2. Boolean
3. Long
4. Double
5. Map<String, Object>
6. List<Object>

Όσον αφορά τους τρόπους με του οποίους μπορεί να γίνει εισαγωγή δεδομένων στην βάση του Firebase είναι κάτι το σχετικά απλό καθώς προσφέρονται πάνω από μία μεθόδους εισαγωγής και συγκεκριμένα τέσσερις στον αριθμό. Αυτό που απαιτείται σαν βασική προϋπόθεση για την εισαγωγή δεδομένων μέσω αυτών των τεσσάρων συναρτήσεων είναι να οριστεί μια διαδρομή (path) που θα αποτελεί την διεύθυνση της βάσης δεδομένων αλλά και μια διαδρομή όπου τα δεδομένα εισαγωγής θα καταχωρηθούν. Οι συναρτήσεις που πραγματοποιούν καταχώρηση δεδομένων επιγραμματικά αναφέρονται παρακάτω:

1. setValue: Η συγκεκριμένη συνάρτηση γράφει ή αντικαθιστά όλα τα δεδομένα στην συγκεκριμένη διαδρομή που έχει οριστεί για καταχώρηση.
2. updateChild: Πραγματοποιεί ενημερώσεις στο path που έχει ζητηθεί χωρίς να αντικαθιστά όλα τα δεδομένα στην συγκεκριμένη διαδρομή.
3. Push: Πραγματοποιεί προσθήκη σε μια λίστα δεδομένων δημιουργώντας παράλληλα ένα μοναδικό αναγνωριστικό κωδικό που απευθύνεται στην εγγραφή αυτή.
4. runTransaction: Πραγματοποιεί καταχώρηση δεδομένων σε περιπτώσεις που υπάρχει κίνδυνος να υπάρξουν ταυτόχρονες τροποποιήσεις δεδομένων εξασφαλίζοντας ότι δεν θα υπάρξει αλλοίωση των δεδομένων.

Τώρα σε ότι αφορά την ανάκτηση και την ενημέρωση των δεδομένων στη βάση του Firebase προσφέρονται δύο events που πραγματοποιούν αυτές τις επιθυμητές ενέργειες.

1. ValueEventListener: Το event αυτό διαβάζει όλα τα δεδομένα στο path που έχει δοθεί και «ανταποκρίνεται» σε ενημερώσεις που γίνονται για κάθε ένα από αυτά στο συγκεκριμένο path.



2. ChildEventListener: Το event αυτό χρησιμοποιείται κυρίως όταν τα δεδομένα αποτελούνται από λίστες. Συγκεκριμένα εξετάζει αν κάποιος από τους κόμβους στο path που έχει δοθεί υπέστη κάποια ενέργεια. Σε αυτή την περίπτωση μια από τις παρακάτω μεθόδους ενεργοποιείται ανάλογα την ενέργεια που πραγματοποιείται. Οι μέθοδοι αυτές παρουσιάζονται στην παρακάτω λίστα:

- OnChildAdded: Ενεργοποιείται όταν πραγματοποιείται εισαγωγή ενός νέου κόμβου στην συγκεκριμένη διαδρομή.
- OnChildChanged: Ενεργοποιείται όταν πραγματοποιείται κάποια αλλαγή ενός κόμβου στην συγκεκριμένη διαδρομή.
- OnChildRemoved: Ενεργοποιείται όταν πραγματοποιείται αφαίρεση ενός κόμβου στην συγκεκριμένη διαδρομή.
- OnChildMoved: Ενεργοποιείται όταν πραγματοποιείται αναδιάταξη των κόμβων στην συγκεκριμένη διαδρομή.

Τελειώνοντας την περιγραφή στον τρόπο λειτουργία της βάσης στο Firebase αναφέρουμε όπως κάθε άλλη βάση έτσι και αυτή διαρθρώνεται από κανόνες τους οποίους μπορεί να ορίσει ο διαχειριστής της εκάστοτε εφαρμογής στην οποία χρησιμοποιείται η σχετική βάση. Αυτό σημαίνει με άλλα λόγια ότι μπορούν να οριστούν κανόνες σε ότι αφορά τόσο τους χρήστες που μπορούν να συνδεθούν και να εκτελέσουν κάποια ενέργεια στην βάση αυτή αλλά όσο και για τις ενεργείες που έχουν δικαίωμα να πραγματοποιήσουν σε αυτή.

```
{
  "rules": {
    ".read": true,
    ".write": true
  }
}
```

Overview of Backend as a Service platforms . Siim Plangi Institute of Computer Science University of Tartu Tartu, Estonia  
splangi@ut.ee .

Ορισμός δικαιωμάτων στην βάση στο Firebase.

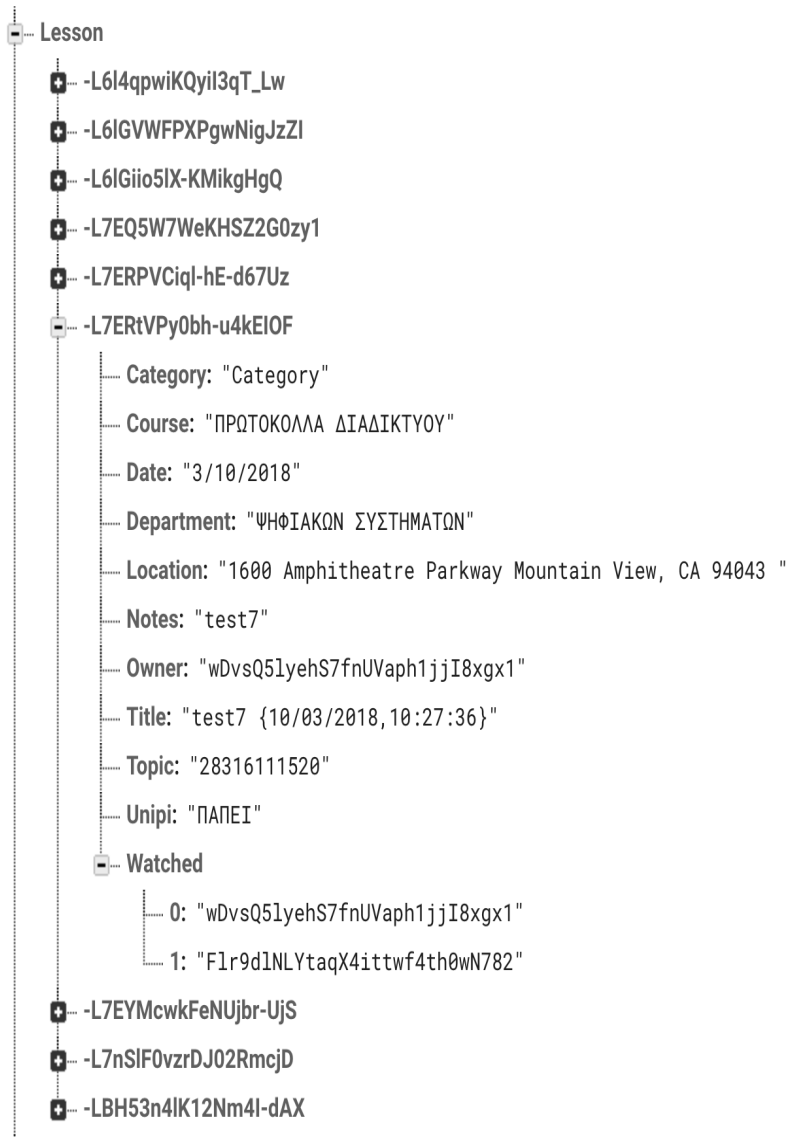


```
{
  "rules": {
    "users": {
      "$user_id": {
        ".write": "$user_id === auth.uid"
      }
    }
  }
}
```

Overview of Backend as a Service platforms . Siim Plangi Institute of Computer Science University of Tartu Tartu, Estonia  
splangi@ut.ee .

Κανόνες που εξασφαλίζουν το Authorization των χρηστών στην βάση στο Firebase.

Αφού ολοκληρώσαμε αναφέροντας επιγραμματικά κάποια πράγμα σχετικά με τον τρόπο λειτουργίας της συγκεκριμένη βάσης κάλο θα ήταν να αναφέρουμε και με ποιον τρόπο αυτές οι δυνατότητες αυτής της βάσης χρησιμοποιήθηκαν στο σύνολο της παρούσας πτυχιακής εργασίας. Γενικότερα η εφαρμογή που διαπραγματεύεται η σχετική πτυχιακή εργασία χρησιμοποιεί επί το πλείστον την τεχνολογία Firebase και ακόμα ειδικότερα την βάση δεδομένων αυτής. Ο λόγος για τον οποίο η εφαρμογή χρησιμοποιεί και την βάση του Firebase πέρα από αυτή που δημιουργεί σε SQLite όπως έχουμε αναφέρει και πιο πάνω είναι για να μπορεί ο οποιοσδήποτε χρήστης της εφαρμογής ανά πάσα στιγμή να μπορεί να ανατρέξει σε σημειώσεις είτε δικές του είτε άλλων και να μπορεί να τις επεξεργαστεί. Συνεπώς γίνεται κατανοητό πλέον για πιο λόγο οι σημειώσεις ενός χρήστη αποθηκεύονται πέρα από τοπικά και στην βάση του Firebase σχηματίζοντας αντίστοιχα JSON αντικείμενα σε αυτή τα οποία θα περιγράψουμε παρακάτω.



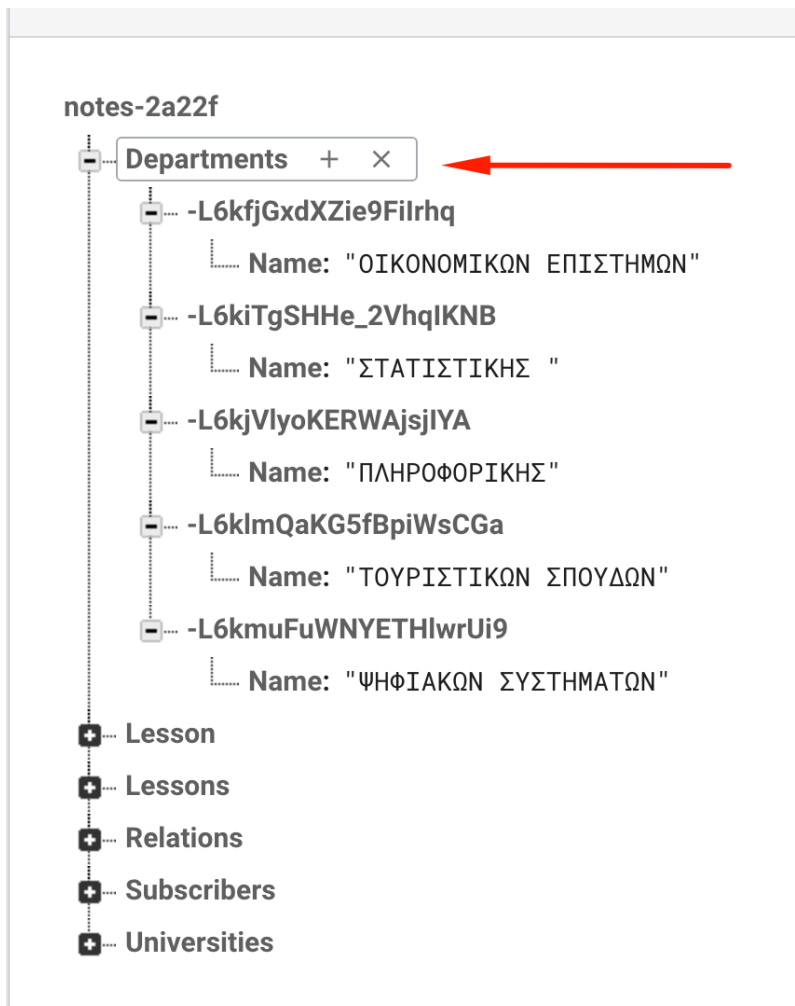
Στην παραπάνω εικόνα παρουσιάζεται η δομή του αντικειμένου JSON που σχηματίζεται κατά την εισαγωγή ή την επεξεργασία κάποιας υπάρχουσας σημείωσης. Το αντικείμενο αυτό όπως παρατηρούμε αποτελείται από τον βασικό κόμβο με την ονομασία Lesson. Εσωτερικά σε αυτών διακρίνονται κάποιοι ακόμα υπό κόμβοι οι οποίοι αντικατοπτρίζουν τις υπάρχουσες σημειώσεις της εφαρμογής. Κάθε ένας από αυτούς όπως παρατηρούμε έχει ένα αναγνωριστικό αριθμό το οποίο όπως ήδη αναφέραμε το κληροδοτήθηκε από την συνάρτηση push με την οποία πραγματοποιήθηκε η πρώτη καταχώρηση της κάθε σημείωσης. Εσωτερικά η κάθε εγγραφή δηλαδή η κάθε σημείωση περιέχει ένα σύνολο από πεδία. Τα πεδία αυτά είναι το Category, το Course, το Date, το Department, το Location, το Notes, το Owner, το Title, το Topic, το Unipi, το Watched και τα οποία θα εξηγήσουμε λίγο πιο αναλυτικά:



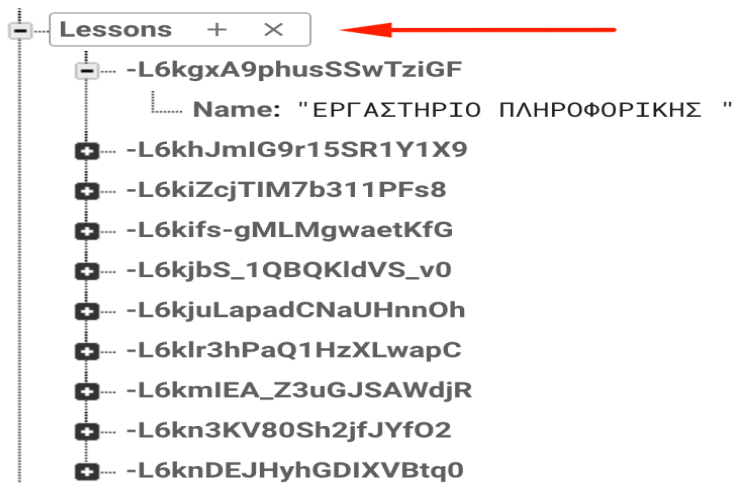
- **Category:** Αποτελεί το πεδίο στο οποίο αποθηκεύεται η κατηγορία στην οποία ο χρήστης της την έχει εντάξει προκειμένου να μπορεί να την αναζητεί πιο εύκολα.
- **Course:** Το πεδίο αναγράφει την ονομασία του μαθήματος με το οποίο σχετίζεται η σημείωση.
- **Date:** Στο πεδίο αυτό αναγράφει ο χρήστης εάν το επιθυμεί την ημερομηνία την οποία ο χρήστης θεωρεί ότι χρονολογικά ανήκει η σημείωση αυτή.
- **Department:** Στο πεδίο αυτό αναφέρεται το τμήμα στο οποίο το σχετιζόμενο με την σημείωση μάθημα διδάσκεται.
- **Location:** Σε αυτό το πεδίο εάν και εφόσον ο χρήστης το επιθυμεί καταγράφεται η τοποθεσία από την οποία πραγματοποιείται η ανάρτηση της σημείωσης.
- **Notes:** Αποτελεί το πεδίο στο οποίο αναγράφεται το κείμενο της σημείωσης.
- **Owner:** Το πεδίο αυτό δηλώνει ποιος είναι ο συντάκτης και ιδιοκτήτης της σημείωσης.
- **Title:** Στο πεδίο αυτό αναγράφεται ο τίτλος της σημείωσης σε συνδυασμό με την ημερομηνία κατά την οποία η σημείωση καταχωρήθηκε για πρώτη φορά στην εφαρμογή.
- **Topic:** Στο πεδίο αυτό αναγράφονται οι συνδυασμοί των κωδικών του Πανεπιστημίου, του Τμήματος και του Μαθήματος με τα οποία η σημείωση σχετίζεται έτσι ώστε να παρέχονται οι απαραίτητες ενημερώσεις στους ενδιαφερόμενους χρήστες.
- **Unipi:** Το πεδίο αυτό περιλαμβάνει την ονομασία του Πανεπιστημίου
- **Watched:** Το συγκεκριμένο πεδίο καταγράφει όλους τους χρήστες που έχουν κατεβάσει την σχετική σημείωση τοπικά στις συσκευές τους προκειμένου να τους παρέχει ενημερώσεις στην περίπτωση που η σημείωση υποστεί αλλαγές από τον ιδιοκτήτη της και αυτοί επιθυμούν να κατεβάσουν την νέα έκδοση της σημείωσης αυτής.

Παρόλα αυτά στην εφαρμογή παράγονται και αλλά αντικείμενα JSON που συμβάλουν στην σωστή λειτουργία της. Τα αντικείμενα αυτά προκειμένου να εξυπηρετήσουν ανάγκες της εφαρμογής περιγράφουν οντότητες όπως τα πανεπιστήμια, τα τμήματα, τα μαθήματα καθώς και τις μεταξύ τους σχέσεις αλλά και οντότητες όπως τους εγγεγραμμένους χρήστες σε κάποιο μάθημα. Παρακάτω παραθέτουμε και εξηγούμε αναλυτικότερα αυτές τις οντότητες.



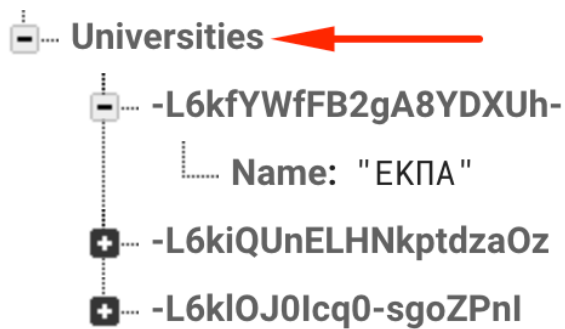


Η παραπάνω οντότητα με όνομα Departments περιγράφει τα τμήματα που είναι διαθέσιμα στην εφαρμογή. Κάθε κόμβος αυτής της οντότητας διακρίνεται από ένα μοναδικό αναγνωριστικό κωδικό και εσωτερικά μέσα στον καθένα από αυτούς συμπεριλαμβάνεται το πεδίο Name που προσδιορίζει την ονομασία του τμήματος που περιγράφεται. Αντίστοιχα παρόμοιες με την παραπάνω οντότητα είναι και αυτές που δημιουργούνται για να περιγράψουν ένα πανεπιστήμιο ή ένα μάθημα.

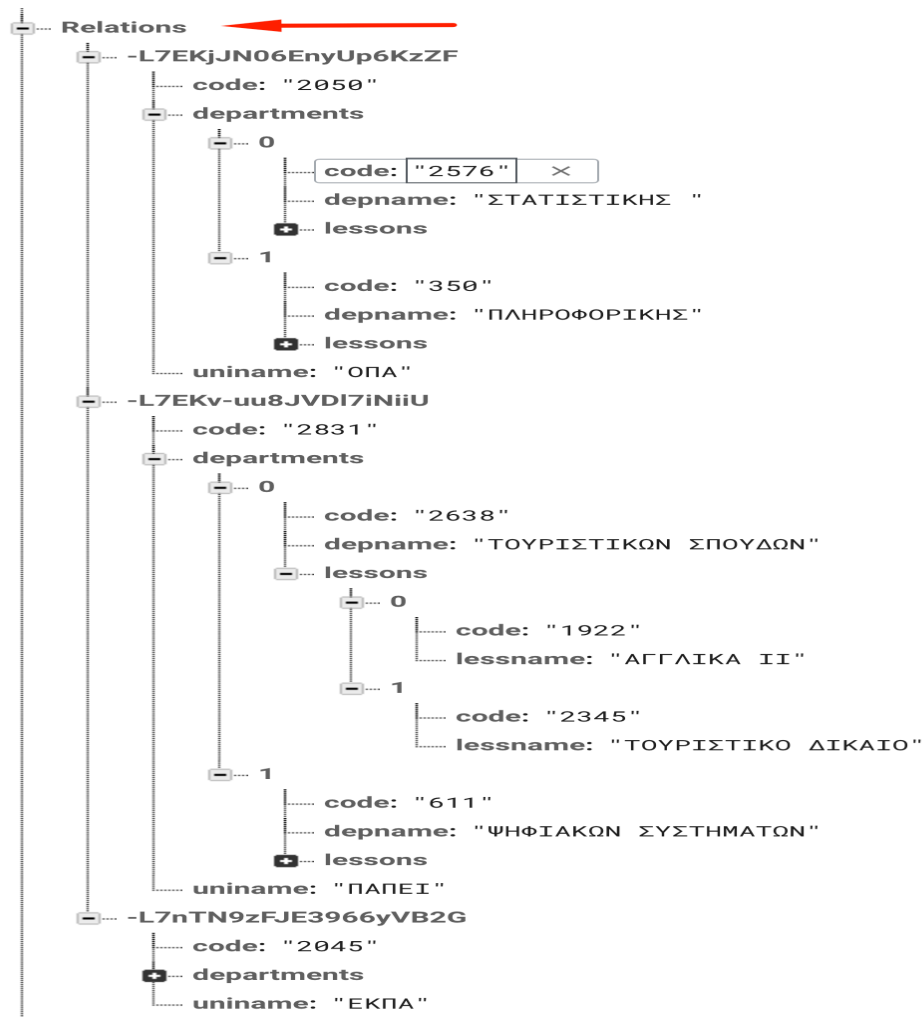


Όπως παρατηρούμε και σε αυτή την οντότητα κάθε κόμβος της διακρίνεται από ένα αναγνωριστικό Id. Επίσης μέσα του και σε αυτή την οντότητα κάθε κόμβος συμπεριλαμβάνει το πεδίο Name που δηλώνει την ονομασία αντίστοιχα του μαθήματος που περιγράφεται.

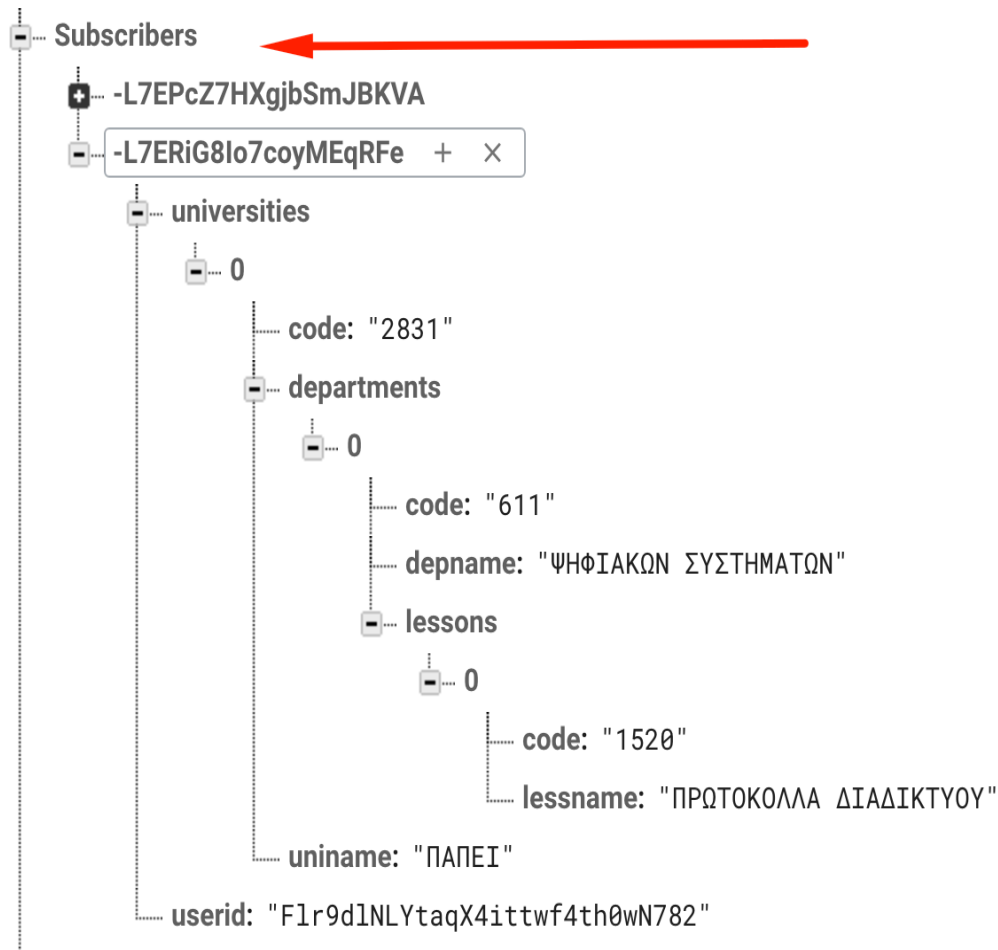
Ομοίως με την ίδια ακριβώς δομή όπως φαίνεται στην εικόνα παρακάτω δημιουργείται και η οντότητα που περιγράφει τα πανεπιστήμια.



Διαφορετική δομή παρουσιάζεται στην οντότητα που περιγράφει μια σχέση μεταξύ ενός Πανεπιστημίου, ενός Τμήματος και ενός Μαθήματος. Η δομή αυτή παρουσιάζεται παρακάτω.



Η οντότητα αυτή περιλαμβάνει τα Πανεπιστήμια και τις συσχετίσεις που έχουν δημιουργηθεί κάτω από αυτά μεταξύ των τμημάτων και των μαθημάτων. Συγκεκριμένα ξεκινώντας από το πιο πάνω επίπεδο ενός κόμβου αυτής την οντότητας βλέπουμε ότι περιγράφει ένα πανεπιστήμιο όπου στο πεδίο `uniname` διατηρεί το όνομα του πανεπιστημίου που περιγράφει και στο πεδίο `code` ένας μοναδικός κωδικός που απευθύνεται σε αυτό το πανεπιστήμιο. Εσωτερικά σε αυτό τον κόμβο παρουσιάζεται αντίστοιχα ένας άλλος κόμβος ο οποίος στο `depname` διατηρεί την ονομασία του τμήματος κάτω από το συγκεκριμένο πανεπιστήμιο καθώς και ένα μοναδικό κωδικό που απευθύνεται στο τμήμα αυτό. Κατά αντιστοιχία εσωτερικά σε κάθε κόμβο που αναφέρεται σε ένα τμήμα παρουσιάζονται άλλοι κόμβοι που με παρόμοιο τρόπο με τους γονείς κόμβους δηλαδή αυτόν του τμήματος και του πανεπιστημίου περιγράφουν τα μαθήματα που συσχετίζονται με το αντίστοιχο τμήμα και πανεπιστήμιο. Τελειώνοντας την περιγραφή των αντικειμένων JSON που παράγονται από την εφαρμογή προκειμένου να είναι λειτουργική συναντάμε ακόμα μία οντότητα αυτή των `Subscribers` η οποία περιγράφει τους ενδιαφερόμενους χρήστες σχετικά με κάποιο ή κάποια συγκεκριμένα μαθήματα.



Παρατείνοντας λοιπόν την οντότητα αυτή διακρίνουμε ότι κάθε κόμβος της αφορά και ένα χρήστη της εφαρμογής. Με αλλά λόγια την πληροφορία αυτή δηλώνει το πεδίο userid όπου και αναγράφεται ο χρήστης. Εσωτερικά σε αυτούς τους κόμβους αναφέρεται η πληροφορία σχετικά με το ποιο μάθημα ενδιαφέρει τον χρήστη αλλά και σε ποιο πανεπιστήμιο και τμήμα αυτό ανήκει.

- **Cloud Functions:**

Πρόκειται για μια λειτουργικότητα του Firebase η οποία επιτρέπει στους προγραμματιστές να ανεβάσουν κώδικα στο cloud του Firebase ο οποίος θα εκτελείτε αυτόματα είτε συνδέοντας τον με κάποιο υπάρχον event του Firebase όπως (Cloud Firestore Triggers, Realtime Database Triggers, Firebase Authentication Triggers, Google Analytics Triggers, Crashlytics Triggers, Cloud Storage Triggers, Cloud Pub/Sub Triggers) είτε συνδέοντας το ως response σε ένα HTTP request. Ο κώδικας αυτός αποθηκεύεται στο Cloud της Google και εκτελείται κάθε φορά που θα κληθεί με ένα από τους παραπάνω τρόπους που αναφέραμε χωρίς να υπάρχει απαίτηση για καμία διαχείριση από οποιοδήποτε άλλο server που χρησιμοποιείται.

- **Hosting:**

Η λειτουργικότητα αυτή αφορά την εγκατάσταση στατικών web εφαρμογών σε Server του Firebase παρέχοντας πολλά πλεονεκτήματα όπως για παράδειγμα την ταχύτητα, την



ασφάλεια και ούτω κάθε εξής που απαιτείται από μια εφαρμογή για να εγκατασταθεί αλλά και να λειτουργεί σε επόμενο στάδιο όπως αναμένεται σε ένα Web Server.

- **Performance Monitoring:**

Αυτή η λειτουργία παρέχει στους προγραμματιστές διαγνωστικό έλεγχο της απόδοσης της εφαρμογής απευθείας στην κονσόλα του Firebase. Με αυτοματοποιημένη ή και προσαρμοσμένη καταγραφή της απόδοσης τμημάτων της εφαρμογής η λειτουργικότητα αυτή παρέχει μετρήσεις σχετικές με την εφαρμογή πραγματοποιώντας διαγνωστικούς ελέγχους προσομοιώνοντας την λειτουργικότητα αυτής σαν να εκτελούνταν στην συσκευή του χρήστη.

- **Crashlytics:**

Με την λειτουργικότητα αυτή του firebase οι προγραμματιστές μπορούν να παρακολουθούν και να ενημερώνεται σε πραγματικό χρόνο για τα συμβάντα που προκύπτουν σε μία εφαρμογή με απώτερο σκοπό τους να παραμένει η εφαρμογή σε σταθερό επίπεδο αλλά και τυχόν προβλήματα που προκύπτουν να αντιμετωπίζονται άμεσα.

- **Authentication:**

Η λειτουργία αυτή του firebase δίνει την δυνατότητα αυθεντικοποίησης των χρηστών σε μία εφαρμογή. Με άλλα λόγια αυτό που προσφέρει είναι ότι επιτρέπει μόνο σε εξουσιοδοτημένους χρήστες να έχουν πρόσβαση και να εκτελούν λειτουργίες που προσφέρονται μέσω αυτής. Αυτό επιτυγχάνεται μέσω του Firebase Auth καθώς προσφέρει μία μεγάλη ποικιλία από ελέγχους ταυτοποίησης όπως αυτή του ηλεκτρονικού ταχυδρομείου Gmail, του Facebook, του Github, του Twitter αλλά επίσης προσφέρει τα μέσα και τις μεθόδους για την οικοδόμηση οποιοδήποτε τρόπου ταυτοποιήσεις ο προγραμματιστής επιθυμεί να κατασκευάσει για την εφαρμογή του. Μετά την επιτυχή είσοδο, παρέχετε η δυνατότητα πρόσβασης στις βασικές πληροφορίες του προφίλ του χρήστη και μπορεί να ελέγχετε η πρόσβαση του χρήστη σε δεδομένα που είναι αποθηκευμένα σε άλλα προϊόντα του Firebase. Όσον αφορά την παρούσα πτυχιακή εργασία η λειτουργία Authentication του Firebase χρησιμοποιείτε για την ταυτοποίηση του χρήστη κατά την είσοδο του στην εφαρμογή γενικότερα αλλά για την ταυτοποίηση του σε κάθε Activity που ο χρήστης εισέρχεται καθώς με αυτό τον τρόπο αποκτά πρόσβαση στα δεδομένα που τον αφορούν και είναι αποθηκευμένα στην βάση δεδομένων του Firebase.

- **Cloud Storage:**

Με την λειτουργία Storage που παρέχετε από το firebase δίνεται η δυνατότητα αποθήκευσης και κοινοποίησης multimedia αρχείων όπως εικόνας, ήχου και βίντεο που αποθηκεύονται ή λαμβάνονται από χρήστες της εφαρμογής προσφέροντας παράλληλα ασφάλεια αλλά και ποιότητα στις υπηρεσίες λήψης και αποθήκευσης των αρχείων αυτών που εξασφαλίζεται μέσω της εταιρείας Google. Η εφαρμογή λοιπόν αυτή των σημειώσεων με την οποία ασχολούμαστε στην παρούσα πτυχιακή εργασία κάνει χρήση της συγκεκριμένης λειτουργικότητας εκτελώντας ασύγχρονες κλήσεις καλώντας τις σχετικές μεθόδους που παρέχονται από το firebase όταν ο χρήστης επιθυμεί να καταχώρηση κάποια εικόνα σε κάποια σημείωση του ή να δει άλλες που εμπεριέχουν εικόνες.



Notes ▾

## Storage

FILES RULES USAGE

gs://notes-2a22f.appspot.com > Fbimages UPLOAD FILE + ⋮

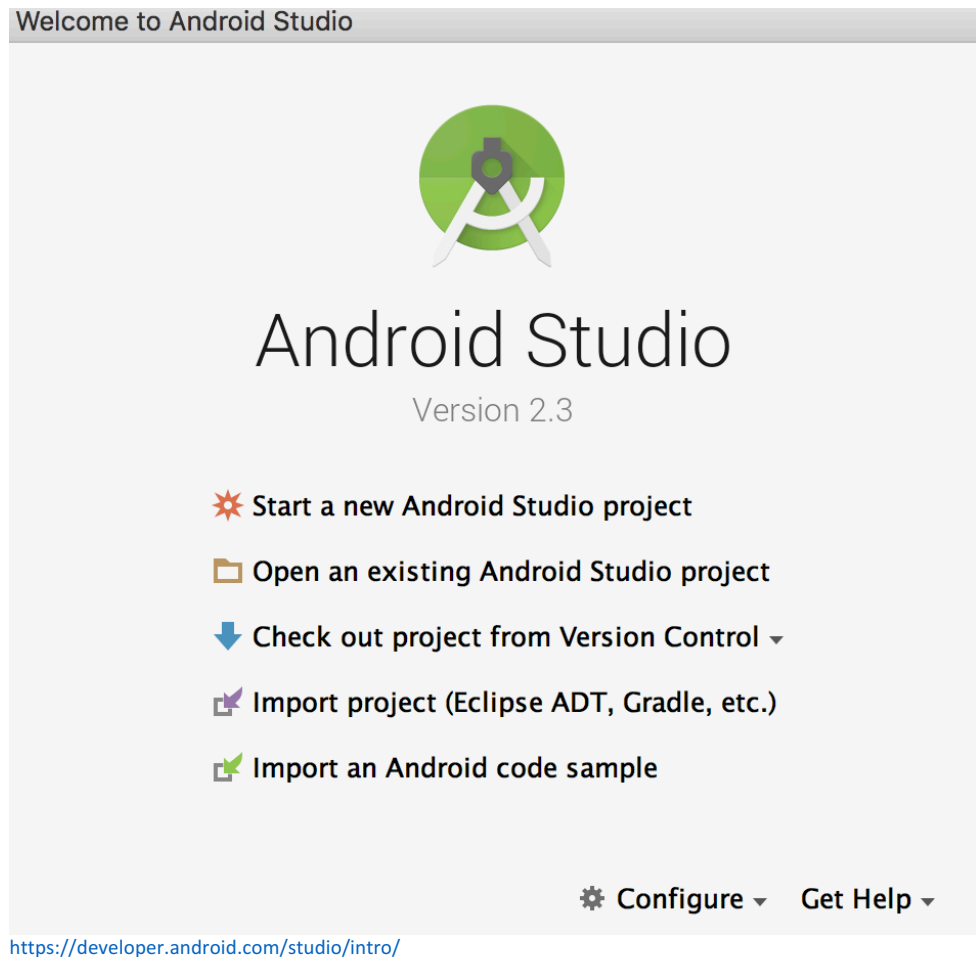
<input type="checkbox"/>	Name	Size	Type	Last modified
<input type="checkbox"/>	05dcbd5c-4d4a-460c-8522-f8221b5a7594.jpeg	4.84 KB	image/jpeg	Feb 18, 2018
<input type="checkbox"/>	0b707193-efee-491b-914c-af7f3a32e472.jpeg	10.35 KB	image/jpeg	Feb 18, 2018
<input type="checkbox"/>	121b7658-95f7-4dca-b55a-c268e19389af.jpeg	9.09 KB	image/jpeg	Mar 1, 2018
<input type="checkbox"/>	127b939b-7f59-4c12-a11f-32c1f1cac6fd.jpeg	7.89 KB	image/jpeg	Feb 18, 2018
<input type="checkbox"/>	14b67dd0-8493-4231-9f4e-ce9db6c4c555.jpeg	9.9 KB	image/jpeg	Mar 4, 2018
<input type="checkbox"/>	159b4d68-964c-4ac7-b7dc-048a28998581.jpeg	15.92 KB	image/jpeg	Mar 10, 2018
<input type="checkbox"/>	1c21c7ea-c7c8-47b6-860a-c3b6d76a6867.jpeg	4.7 KB	image/jpeg	Feb 18, 2018
<input type="checkbox"/>	24c0cc5a-be84-4c62-8d15-fce998d15ce8.jpeg	4.16 KB	image/jpeg	Feb 20, 2018
<input type="checkbox"/>	26e0be05-e7f0-4af3-a10b-d73aedb63a42.jpeg	4.97 KB	image/jpeg	Feb 18, 2018

- **Test Lab for Android:**

Η λειτουργία αυτή προσφέρει στους προγραμματιστές την δυνατότητα να δοκιμάζουν τις εφαρμογές τους τόσο σε φυσικές όσο και σε εικονικές συσκευές που προσφέρονται μέσω της Google παρέχοντας έτσι στον προγραμματιστή ένα περιβάλλον ανάπτυξης ιδανικό έτσι ώστε να προετοιμάσει την εφαρμογή του και να διορθώσει τυχόν σφάλματα που θα προκύψουν εξομοιώνοντας την λειτουργικότητα αυτής όσο τον δυνατόν καλύτερα σε ένα περιβάλλον ανάπτυξης με μεγάλη ποικιλία συσκευών προσομοίωσης.

## 5.7 ΕΡΓΑΛΕΙΟ ΑΝΑΠΤΥΞΗΣ ΚΩΔΙΚΑ ANDROID STUDIO

Το εργαλείο με το οποίο έγινε ανάπτυξη και εκτέλεση του κώδικα της εφαρμογής Android για σημείωσης φοιτητών που υλοποιήθηκε στην παρούσα πτυχιακή εργασία έχει την ονομασία Android Studio.



Το Android Studio είναι ένα ολοκληρωμένο αναπτυξιακό περιβάλλον που ειδικεύεται στην ανάπτυξη εφαρμογών για κινητές συσκευές σε λειτουργικό σύστημα Android και είναι βασισμένο στο IntelliJ IDEA. Πρόκειται για ένα εργαλείο ανάπτυξης με πολύ ισχυρές δυνατότητες όχι μόνο στην επεξεργασία και ανάπτυξη του κώδικα αλλά και πολλές άλλες που βελτιώνουν την παραγωγικότητα ενός προγραμματιστή κατά την κατασκευή εφαρμογών σε Android.

Ορισμένες από τις δυνατότητες αυτές αναφέρονται παρακάτω:

- Παρέχει ευέλικτο σύστημα κατασκευής βασισμένο στο Gradle.
- Προσφέρει ένα γρήγορο και αξιόπιστο προσομοιωτή κινητών συσκευών
- Παρέχει μεγάλη γκάμα συσκευών για προσομοίωση καθώς και μεγάλη ποικιλία όλων των λειτουργικών εκδόσεων του Android.
- Προσφέρει άμεση εκτέλεση του κώδικα χωρίς να απαιτείται η δημιουργία νέου APK.
- Υποστηρίζει πρότυπα κώδικα (Code Templates) αλλά και την λειτουργικότητα του GitHub προκειμένου να είναι εφικτή η χρήση κοινών λειτουργιών που έχουν αναπτυχθεί σε διαφορετικά project.
- Παρέχει εκτεταμένα εργαλεία για δοκιμές του κώδικα και υποστηρίζει διαφορετικά Frameworks.

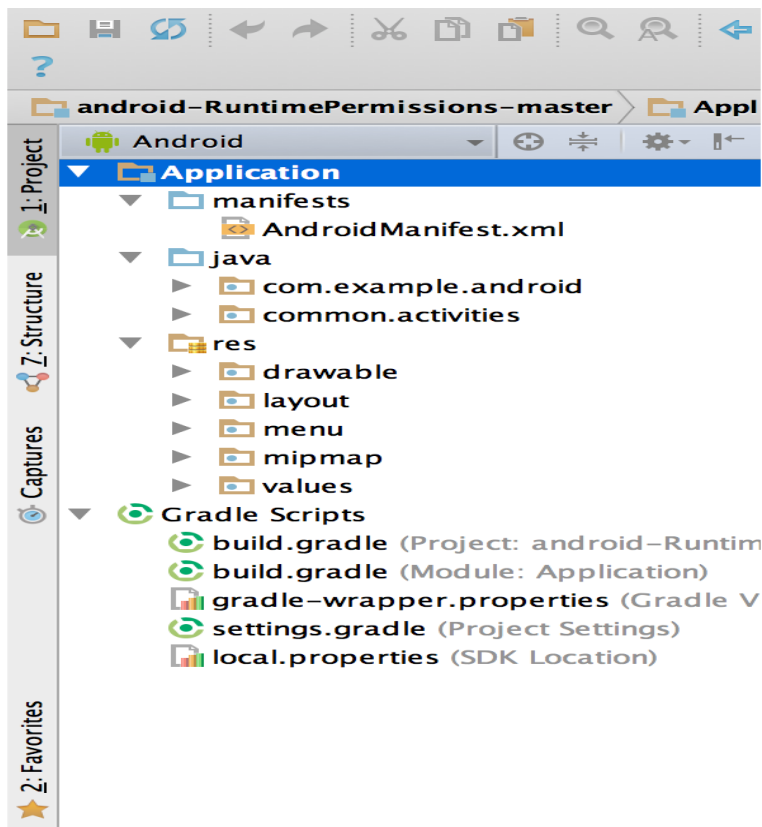




- Προσφέρει εργαλεία Lint προκειμένου να αντιμετωπίζονται προβλήματα απόδοσης, χρηστικότητας, συμβατότητας και άλλα.
- Υποστηρίζεται η γλώσσα προγραμματισμού c++ και σε c καθώς παρέχετε το NDK εργαλείο που βοηθάει στην ανάπτυξη εφαρμογών native code.
- Παρέχει ενσωματωμένη υποστήριξη για την πλατφόρμα Google Cloud.

## 5.8 ΔΟΜΗ ΕΝΟΣ ΕΡΓΟΥ ΣΕ ANDROID STUDIO

Τα έργα που παράγονται μέσω του εργαλείου ανάπτυξη κώδικα Android Studio διαρθρώνεται με μια προκαθορισμένη δομή αρχείων η οποία παρουσιάζεται στην παρακάτω εικόνα.



<https://developer.android.com/studio/intro/>

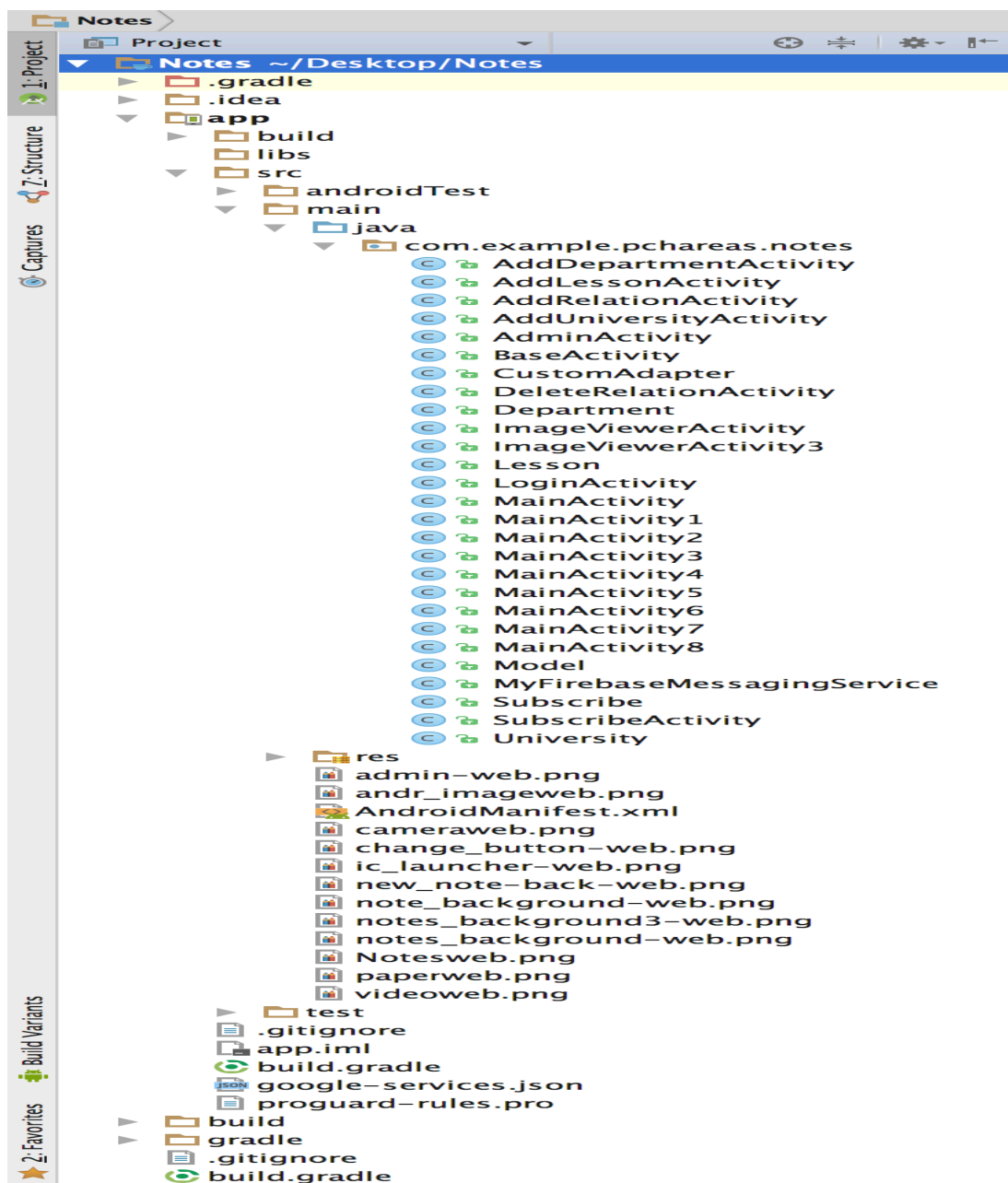
Όπως παρατηρούμε κάθε έργο που παράγεται μέσω του συγκεκριμένου εργαλείου αποτελείται από μια ή περισσότερες ενότητες αρχείων πηγαίου κώδικα ή πόρων τις οποίες θα μπορούσαμε να τις κατηγοριοποιήσουμε σε 3 μεγαλύτερες ενότητες αυτές των Android app module, Library modules και των Google App Engine modules. Πιο συγκεκριμένα βλέπουμε ότι όλα τα εκτελέσιμα αρχεία του έργου τοποθετούνται στο επάνω επίπεδο κάτω από τα Gradle Scripts και κάθε αρχείο περιλαμβάνει τους εξής φακέλους:

- manifests: Εδώ τοποθετείται το αρχείο AndroidManifest.xml που περιλαμβάνει γενικές ρυθμίσεις που αφορούν την εφαρμογή την οποία διαπραγματεύεται το εκάστοτε έργο.
















- java: Στον φάκελό αυτό περιλαμβάνεται όλος ο πηγαίος κώδικας σε Java που αφορά την λειτουργικότητα της εκάστοτε εφαρμογής.
- res: Περιλαμβάνει όλους του απαραίτητους πόρους τους οποίους απαιτεί η εφαρμογή όπως XML αρχεία που αφορούν το user interface, λεκτικά αυτής καθώς και εικόνες.

Έτσι λοιπόν εφόσον παρουσιάσαμε την γενική δομή των έργων που παράγονται μέσω του εργαλείου Android Studio παρακάτω σας παρουσιάζουμε πιο συγκεκριμένα την δομή που έχει η εφαρμογή των σημειώσεων που υλοποιήθηκε για την παρούσα πτυχιακή εργασία.





-  google-services.json
-  proguard-rules.pro
- ▶  build
- ▶  gradle
-  .gitignore
-  build.gradle
-  gradle.properties
-  gradlew
-  gradlew.bat
-  local.properties
-  Notes.iml
-  settings.gradle
- ▶  External Libraries



## 6. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Στο σημείο αυτό της ανάλυσης μας στην παρούσα διπλωματική εργασία θα σας παρουσιάσουμε τα συμπεράσματα που προέκυψαν κατά την σύνθεση της εφαρμογής που κατασκευάστηκε αλλά και τις μελλοντικές επεκτάσεις που θα μπορούσαν να υφίστανται ώστε να βελτιωθεί το εικαστικό αλλά και το λειτουργικό κομμάτι αυτής. Όσον αφορά τα συμπεράσματα που αποκομίσαμε από την σύνθεση της εφαρμογής είναι ότι οι ανάγκες της σύγχρονης εποχής που διανύουμε απαιτούν εφαρμογές οι οποίες πρέπει να είναι συμβατές με κινητές συσκευές οι οποίες όπως προμηνύεται επρόκειτο να βρίσκονται στο επίκεντρο των εξελίξεων για τα επόμενα χρόνια καθώς ήδη σε μεγάλο βαθμό έχουν αντικαταστήσει τους σταθερούς υπολογιστές και τα laptop τουλάχιστον στο σημείο να καλύπτουν τις καθημερινές ανάγκες ενός χρήστη. Επιπλέον η εφαρμογή αυτή των σημειώσεων που κατασκευάστηκε για τις ανάγκες της διπλωματικής βασίστηκε σε σύγχρονες τεχνολογίες και υπηρεσίες όπως αυτές του Cloud καθώς οι χρήστες πλέον απαιτούν εφαρμογές οι οποίες ανταποκρίνονται άμεσα στις ανάγκες και τα θέλω τους σε πραγματικό χρόνο χωρίς να αντιμετωπίζουν καθυστερήσεις. Επιπρόσθετα αναμένουν ότι πλέον στις μέρες μας μπορούν να χρησιμοποιούν αυτές τις εφαρμογές από οποιαδήποτε τοποθεσία και αν βρίσκονται καθώς και ότι ταυτόχρονα εξασφαλίζεται η ασφάλεια των δεδομένων τους παράμετροι που λήφθηκαν υπόψιν κατά την ανάπτυξη της παρούσας εφαρμογής. Σε ότι αφορά τώρα τις μελλοντικές επεκτάσεις που θα μπορούσαν να γίνουν στην εφαρμογή αναφέρουμε ότι μια από τις πρώτες επεκτάσεις που θα μπορούσαν να πραγματοποιηθούν αποτελεί αυτή της βελτίωσης των γραφικών στοιχείων που συνθέτουν τις οθόνες της. Η εφαρμογή σε αυτό το στάδιο υλοποιήσεις που πραγματοποιήθηκε δηλαδή για την εκπόνηση της διπλωματικής εργασίας επικεντρώθηκε κυρίως στο λειτουργικό κομμάτι της εφαρμογής ωστόσο πολλές βελτιώσεις μπορούν να εφαρμοστούν σε ότι αφορά το εικαστικό κομμάτι αυτής χρησιμοποιώντας εργαλεία σχεδίασης όπως το illustrator ή το Photoshop. Όσον αφορά το λειτουργικό κομμάτι αυτής θα μπορούσαν και εκεί να πραγματοποιηθούν μελλοντικές επεκτάσεις όπως η δυνατότητα προσθήκης πέρα από αρχείων εικόνας και αρχείων βίντεο στις σημειώσεις των χρηστών. Επίσης θα μπορούσε να ενσωματωθεί λειτουργικότητα καταχώρησης χειρόγραφης σημείωσης με την βοήθεια κάποια ειδικής γραφίδας αλλά και αυτή της καταγραφής σημείωσης μέσω φωνητικής εγγραφής. Επιπλέον θα μπορούσε να ενσωματωθεί η δυνατότητα σχολιασμού κάτω από κάθε σημείωση ώστε οι ενδιαφερόμενοι με την σημείωση χρήστες να μπορούν να ανταλλάσσουν απόψεις και σχόλια σχετικά με αυτή. Επιπρόσθετα μια ακόμα επέκταση που θα ήταν εφικτό σε μελλοντική έκδοση της εφαρμογής να υπάρχει είναι και η μετατροπή της σημείωσης σε αρχείο pdf. Τέλος θα μπορούσε να προστεθεί ως λειτουργικότητα αυτής η παροχή μίας πιο ισχυρής αναζήτησης των σημειώσεων της εφαρμογής μέσω ενός πεδίου που θα ψάχνει τόσο στους τίτλους και την θεματολογία των σημειώσεων όσο και στο περιεχόμενο αυτών κάνοντας χρήση ευρετηρίου και κάποιου σχετικού αλγόριθμου αναζήτησης όπως ο PageRank.



## 7. ΒΙΒΛΙΟΓΡΑΦΙΑ

- International Journal of Innovative Research in Computer and Communication Engineering (*An ISO 3297: 2007 Certified Organization*) Vol. 4, Issue 9, September 2016 Study of Google Firebase API for Android
- Proceedings of 2014 RA ECS UIET Panjab University Chandigarh, 06 – 08 March, 2014 Google Android a Mobile Platform: A Review
- 2017 14th International Conference on Engineering of Modern Electric Systems (EMES) Comparative Study of Google Android, Apple iOS and Microsoft Windows Phone Mobile Operating Systems
- *Core Java 1.1: Volume 1 - Fundamentals* by Cay S. Horstmann and Gary Cornell, Pentice-Hall. (This will be the textbook for the Spring 1998 Java class.) *Java in a Nutshell, 2nd Edition* by David Flanagan. O'Reilly. (A very good reference!)
- “History of Java Programming Language” Free Java Guide & Tutorials
- Overview of Backend as a Service platforms Report Siim Plangi Institute of Computer Science University of Tartu Tartu, Estonia splangi@ut.ee
- Research and Application of SQLite Embedded Database Technology CHUNYUE BI School of Computer Science and Information Technology Zhejiang Wanli University No.8, South Qian Hu Road Ningbo, Zhejiang P.R.CHINA
- Mobile Data Store Platforms: Test Case based Performance Evaluation Karim Kussainov and Bolatzhan Kumalakov *School of Science and Technology, Nazarbayev University, Kabanbay batyr 53, Astana, Republic of Kazakhstan*
- S.T. Bhosale et al, International Journal of Computer Science and Mobile Computing, Vol.4 Issue.4, April- 2015, pg. 882-885
- <https://developer.android.com/studio/intro/>
- <https://zapier.com/blog/best-note-taking-apps/>
- <https://firebase.google.com/products/>
- [https://en.wikipedia.org/wiki/Programming\\_language](https://en.wikipedia.org/wiki/Programming_language)