



**“Ανάπτυξη εφαρμογής Android για εκμάθηση  
προγραμματιστικού περιβάλλοντος Γ’ Λυκείου.”**

**“Creation of android application for learning programming  
language at greek high school.”**

Διπλωματική Εργασία του Αντωνιάδη Εμμανουήλ.

Diploma thesis,  
Antoniadis Emmanouhl

## **Ευχαριστίες**

Με την περάτωση της παρούσας εργασίας θα ήθελα να ευχαριστήσω θερμά τους επιβλέποντες της Διπλωματικής εργασίας κ. Τσουκαλά Ελευθέριο και την κ. Τσαλαπάτα Χαρίκλεια για την άριστη συνεργασία, τις ουσιώδης υποδείξεις και παρεμβάσεις, που διευκόλυναν την εκπόνηση της Διπλωματικής εργασίας.

Επίσης, οφείλω ένα μεγάλο και θερμό ευχαριστώ στην οικογένεια μου και στους φίλους μου, Στέφανο και Πολύκαρπο, για υποστήριξη και την βοήθεια τους που με παρείχαν τόσα χρόνια.

Βόλος, Φεβρουάριος 20



**“Ανάπτυξη εφαρμογής Android για εκμάθηση προγραμματιστικού περιβάλλοντος Γ’ Λυκείου.”**

**“Creation of android application for learning programming language at greek high school.”**

Διπλωματική Εργασία του Αντωνιάδη Εμμανουήλ.

Επιβλέποντες:

<b>Τσουκαλάς Ελευθέριος</b>	<b>Τσαλαπάτα Χαρίκλεια</b>
Καθηγητής, Πανεπιστήμιο Θεσσαλίας	Ε.ΔΙ.Π, Πανεπιστήμιο Θεσσαλίας

## Περίληψη

Ο σκοπός αυτής της διπλωματικής εργασίας είναι η σχεδίαση και η υλοποίηση μιας εφαρμογής σε λειτουργικό περιβάλλον Android που θα έχει ως στόχο την εκμάθηση προγραμματιστικού περιβάλλοντος Γ' λυκείου.(ΑΕΠΠ).

Συγκεκριμένα ο χρήστης-μαθητής μπορεί να αποκομίσει γνώσεις και να εξετάσει τις γνώσεις του μέσα από μια σειρά ερωτήσεων όπως, ερωτήσεις Σωστό-Λάθος, ερωτήσεις συμπλήρωσης κώδικα κ.α..

Ο σχεδιασμός και η υλοποίηση αυτής της εφαρμογής έγινε σε λειτουργικό περιβάλλον Android με την χρήση του εργαλείου Android studio 3.1 .

## Abstract

The purpose of this diploma thesis is to design and implement an application in an Android operating environment that will aim at learning a high school programming environment (AEPP).

Specifically, the user-learner can gain knowledge and examine his / her knowledge through a series of questions such as, True-False questions, fill-in questions, etc.

The design and implementation of this app took place in an Android operating environment using the Android Studio 3.1 tool.

## Περιεχόμενα

Εισαγωγή.....	7
1 Η υπάρχουσα κατάσταση στα σχολεία σχετικά με τη χρήση τεχνολογίας στην Εκπαίδευση.....	8
1.2 Προκλήσεις για τη χρήση τεχνολογίας στην εκπαίδευση.....	9
1.3 Εφαρμογές για φορητές συσκευές που υπάρχουν για ανάπτυξη προγραμματιστικού περιβάλλοντος Γ' λυκείου.....	9
1.4 Πλεονεκτήματα και μειονεκτήματα των υπάρχων εφαρμογών και σύγκριση με την δικιά μου εφαρμογή.....	14
2. Έξυπνα τηλέφωνα(smartphones).....	16
2.1 Λειτουργικό σύστημα.....	16
2.2 Λειτουργικά συστήματα για έξυπνα τηλέφωνα.....	17
2.3 Λειτουργικό σύστημα Android.....	18
2.4 Εφαρμογές Android.....	18
2.5 Ιστορία εκδόσεων του Android.....	19
2.6 Αρχιτεκτονική Android.....	20
2.7 Android Studio.....	24
3.Βάσεις δεδομένων που χρησιμοποιεί η εφαρμογή και τρόπος επικοινωνίας της εφαρμογής με την βάση.....	29
3.1 Περιεχόμενα της βάσης (βάσεις και πίνακες).....	30
3.2 Επικοινωνία εφαρμογής Android με xampp και τις βάσεις δεδομένων.....	31
3.3 Αρχεία PHP που δημιουργήθηκαν στο xampp.....	32
4. Η εφαρμογή ProGlykeiou.....	35
4.1 Το ProGlykeiou ως εκπαιδευτικό λογισμικό.....	36
4.2 Υλοποίηση της εφαρμογής ProGlykeiou.....	36
4.2.1 Δομή της εφαρμογής ProGlykeiou.....	36
4.2.2 Περιγραφή των βασικών παραθύρων(Activities).....	38
4.3 Αλγόριθμοι και τεχνικές προγραμματισμού.....	51
4.3.1 Γραφικό περιβάλλον με χρήση της γλώσσας xml.....	51
4.3.2 Επικοινωνία με την βάση και επεξεργασία δεδομένων.....	53
4.3.3 Οι βασικοί αλγόριθμοι της εφαρμογής.....	55
4.4 Στατιστικά στοιχεία για την εφαρμογή.....	67
Παράρτημα.....	68
Βιβλιογραφία.....	68



## **Εισαγωγή**

Ζούμε σε μια εποχή που τα πάντα γύρω μας αλλάζουν. Αντιλαμβανόμαστε ότι αυτό έχει να κάνει με τον αυξανόμενο ρυθμό της τεχνολογίας, τις νέες εφευρέσεις, τις νέες επινοήσεις. Συνέπεια όλων αυτών είναι η αλλαγή στον τρόπο με τον οποίο ο άνθρωπος σκέφτεται και πράττει. Έτσι ενεργεί με μεγαλύτερη αυτονομία, πιο γρήγορα και με νέα θεμέλια αποδοτικότητας. Όλοι οι τομείς της ζωής είναι επηρεασμένοι απ' αυτό το πνεύμα της αλλαγής.

Όλο και περισσότερο παρατηρούμε την μεγάλη επιρροή που ασκούν οι Νέες Τεχνολογίες στον τομέα της εκπαίδευσης. Οι υπολογιστές, το Διαδίκτυο (World Wide Web) και η διεισδυτικότητα του, δεν αφήνουν ανεπηρέαστο τον τομέα της παιδείας. Ο εκπαιδευτικός πρέπει να είναι ενήμερος γι' αυτά τα μοναδικά παιδαγωγικά και εκπαιδευτικά πλεονεκτήματα που προσφέρουν οι Νέες Τεχνολογίες πριν αποφασίσει να τις χρησιμοποιήσει. Να μην καταλήξει σ' αυτές μόνο εξαιτίας της άμετρης αγάπης για την τεχνολογία ή απλώς επειδή σήμερα η τεχνολογία έχει μπει στην ζωή μας.

Πραγματικά όμως στον τομέα αυτό, η τεχνολογία έχει πολλά να προσφέρει προς όφελος, των μαθητών, των εκπαιδευτικών και της κοινωνίας. Οι μέσοι μαθητές θα αποκομίσουν πολλά οφέλη, ενώ οι ιδιοφυίες δεν θα περιορίζονται μόνο στην συγκεκριμένη διδακτέα ύλη, ούτε θα χρειάζεται να ακολουθούν αργούς ρυθμούς προς χάριν μερικών άλλων μαθητών. Στα παιδιά με ειδικές ανάγκες ανοίγονται νέοι δρόμοι, ενώ ο αναλφαβητισμός σε ορισμένες περιοχές θα βρει ένα πολύτιμο σύμμαχο.

## 1 Η Υπάρχουσα Κατάσταση στα Σχολεία σχετικά με τη Χρήση Τεχνολογίας στην Εκπαίδευση

Τα περισσότερα σχολεία της Ελλάδος στις μέρες μας έχουν στην κατοχή τους τουλάχιστον τα εξής πράγματα:

- Σύνδεση με το διαδίκτυο(broadband)
- 1 υπολογιστή
- Διαδραστικό πίνακα
- Οργανωμένο εργαστήριο υπολογιστών.

Αυτό σημαίνει ότι οι υποδομές στα σχολεία έχουν βελτιωθεί και αναπτυχθεί αρκετά.

Επίσης μεγάλη ανάπτυξη έγινε το 2000-2005 μέσω του Πανελλήνιου σχολικού Δικτύου. Το Πανελλήνιο Σχολικό Δίκτυο ΠΣΔ ([www.sch.gr](http://www.sch.gr)) είναι το εθνικό δίκτυο του Υπουργείου Παιδείας, Έρευνας και Θρησκευμάτων (ΥΠΠΕΘ), το οποίο διασυνδέει ηλεκτρονικά με ασφάλεια όλα τα σχολεία της πρωτοβάθμιας και της δευτεροβάθμιας εκπαίδευσης, περιλαμβανομένων και των μονάδων της αλλοδαπής, τις υπηρεσίες και τους εποπτευόμενους φορείς του ΥΠΠΕΘ σε κεντρικό και περιφερειακό επίπεδο, τους φορείς παροχής υπηρεσιών δια βίου μάθησης, τους μαθητές, τα στελέχη της εκπαίδευσης και τους λοιπούς εκπαιδευτικούς και φορείς του ΥΠΠΕΘ .

Μέσω του ΠΣΔ, το ΥΠΠΕΘ παρέχει στην εκπαιδευτική κοινότητα ειδικά σχεδιασμένες ψηφιακές υπηρεσίες μάθησης, επικοινωνίας και συνεργασίας, διακυβέρνησης, καθώς και υπηρεσίες υποστήριξης και αρωγής των χρηστών σε όλα τα σχολεία της επικράτειας, ακόμα και στα πλέον απομακρυσμένα.

Ένα από τα βασικότερα ερωτήματα που προκύπτει, λοιπόν, μέσα στα πλαίσια της αξιοποίησης των ψηφιακών μέσων στην εκπαίδευση, είναι με ποιόν τρόπο θα μπορούσε το σύγχρονο σχολείο να ανταποκριθεί στο ρόλο που έχουν τα σύγχρονα αυτά μέσα στη ζωή και την καθημερινότητα των παιδιών. Θα έπρεπε να τα αγνοήσει, πρακτική που ακολουθείται ως επί το πλείστον στις μέρες μας; θα έπρεπε να τα αξιοποιήσει ως εργαλεία για την επίτευξη των στόχων του Αναλυτικού Προγράμματος σπουδών, ή, τελικά, να ανακαλύψει τρόπους να τα εντάξει με περισσότερο κριτικό και δημιουργικό τρόπο;



## 1.2 Προκλήσεις για τη Χρήση Τεχνολογίας στην Εκπαίδευση

Μια από τις βασικότερες προκλήσεις για την χρήση τεχνολογίας στην εκπαίδευση αποτελεί η ανάπτυξη των δεξιοτήτων των εκπαιδευτικών, καθώς πολλές φορές είναι περιορισμένη. Επίσης οι εκπαιδευτικοί είναι πολλές φορές προσκολλημένοι στις παλιές κλασσικές μεθόδους και δεν θέλουν να δοκιμάσουν κάτι καινοτόμο.

Άλλη μια πρόκληση αποτελεί η τεχνική υποστήριξη των σχολείων και η διαθεσιμότητα εξειδικευμένου εκπαιδευτικού λογισμικού σε ελεύθερη χρήση (freeware, open source).

Όλα τα παραπάνω περιορίζουν και δυσκολεύουν τη χρήση Τεχνολογίας υπολογιστών μέσα στην τάξη.

## 1.3 Εφαρμογές για φορητές συσκευές που υπάρχουν για ανάπτυξη προγραμματιστικού περιβάλλοντος Γ' λυκείου.

Οι εφαρμογές για φορητές συσκευές είναι ως επί το πλείστον εφαρμογές που εκτελούνται σε περιβάλλον Android ή IOS και εγκαθίστανται σε κινητά τηλέφωνα και tablet.

Σε αυτό το σημείο θα αναφέρω μερικές εφαρμογές που υπάρχουν διαθέσιμες για Android στο google play και έχουν παρόμοιο στόχο με την δική μου εφαρμογή.

### **ΑΕΠΠ TEST4U**

Το ΑΕΠΠ TEST4U είναι ένα πρόγραμμα το οποίο σε βοηθά να προετοιμαστείς για το μάθημα της ΑΕΠΠ (Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον). Περιέχει 365 ερωτήσεις διαφόρων τύπων. Π.χ. πολλαπλών επιλογών, εισαγωγής κειμένου, σταυρόλεξο κ.α.

**TEST4U**  
www.test4u.eu

Επιστροφή Επόμενο

Κατηγορίες: Πείτε κλικ στις 66/256 ερωτήσεις

- 01 Κεφάλαιο - Ανάλυση προβλημάτων 40 κριτήρια 2.7%
- 02 Κεφάλαιο - Βασικές έννοιες αλγορίθμων 1 36 κριτήρια
- 02 Κεφάλαιο - Βασικές έννοιες αλγορίθμων 2 36 κριτήρια
- 02 Κεφάλαιο - Βασικές έννοιες αλγορίθμων 3 36 κριτήρια
- 02 Κεφάλαιο - Βασικές έννοιες αλγορίθμων 4 36 κριτήρια
- 03 Κεφάλαιο - Δομές δεδομένων και Αλγόριθμοι 1 31 κριτήρια
- 03 Κεφάλαιο - Δομές δεδομένων και Αλγόριθμοι 2 27 κριτήρια

Επιλέξτε 1 από τις 4 απαντήσεις

```

A ← 0
B ← 15
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
ΔΙΑΒΑΣΕ C
ΑΝ C > 0 ΤΟΤΕ
    A ← A + 1
ΑΝ C = 5 ΤΟΤΕ
    A ← A * 2
    B ← B - 3
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΑΝ
    
```

Σε ποιον αλγόριθμο αντιστοιχεί το διάγραμμα ροής της φωτογραφίας;

Ερώτηση: 7/50  
ID: 12450

Υποβολή Ανάρεση

00:51 37:30

Αγνόηση Τερματισμός

**TEST4U**  
Το αυτόματουαξιόμεινα τεστ που βαθμολογεί αυτόματα την απάντησή του υπογραφή.

Κάθετα

2 Τύπος δεδομένων για την αποθήκευση ακέραιων αριθμών

5 Εντολή εισόδου

Συμπληρώστε το σταυρόλεξο

Ερώτηση: 18/50  
ID: 12366

Υποβολή Ανάρεση

01:52 37:30

Αγνόηση Τερματισμός

**TEST4U**  
Το αυτόματουαξιόμεινα τεστ που βαθμολογεί αυτόματα την απάντησή του υπογραφή.

Πληκτρολογήστε στο παρακάτω πεδίο την απάντησή σας.

Τα σύμβολα που χρησιμοποιούνται για να αναπαροστήσουμε διάφορες πράξεις ονομάζονται...

Ερώτηση: 5/50  
ID: 12156

Υποβολή Ανάρεση

00:21 37:30

Αγνόηση Τερματισμός

**TEST4U**  
Το αυτόματουαξιόμεινα τεστ που βαθμολογεί αυτόματα την απάντησή του υπογραφή.

## Σ/Λ Ανάπτυξη Εφαρμογών

Quiz με θέματα θεωρίας Σ/Λ περασμένων Πανελληνίων Εξετάσεων για το μάθημα της Γ' Λυκείου Ανάπτυξη Εφαρμογών σε Προγραμματιστικό Περιβάλλον (ΑΕΠΠ)



## ΑΕΠΠ - The Hero Quest

Σε αυτήν την εφαρμογή δεν είσαι απλά ένας μαθητής.. Είσαι ένας υπερήρωας του Προγραμματισμού!

Η περιπέτεια ξεκινά και καθώς μαθαίνεις νέες γνώσεις, συγκεντρώνεις πόντους, ξεκλειδώνεις πίστες και κερδίζεις διακριτικά προόδου!

Η εφαρμογή περιέχει όλη τη Θεωρία, σύμφωνα με την ύλη των εξετάσεων, Μεθοδολογίες, που θα σε βοηθήσουν να λύσεις Ασκήσεις και να γράφεις τα δικά σου Προγράμματα, πάνω από 500 Λυμένες Ασκήσεις και περισσότερες από 1000 Ασκήσεις και Ερωτήσεις για να λύσεις!

5.2 Διοδιάστατοι Πίνακες

1. Κατά Σειρές

Τμή 1	Τμή 2	Τμή 3
Τμή 4	Τμή 5	Τμή 6
Τμή 7	Τμή 8	Τμή 9

```

...
) Προσέλαση Κατά Σειρές
ΓΙΑ Ι ΑΠΟ 1 ΜΕΧΡΙ Μ
  ΓΙΑ J ΑΠΟ 1 ΜΕΧΡΙ Ν
    Δουλειά
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
...

```

**ΑΕΠΠ** THE HERO QUEST

**Θεωρία**  
 Όλες οι πληροφορίες που χρειάζεσαι για να λύσεις τα προβλήματα της Θεωρίας.

Σε αυτό το τμήμα θα ασχοληθείς με τη Θεωρία. Η Θεωρία δίνεται με απλό και κατανοητό τρόπο και είναι θύρα για με την όλη των εξετάσεων! Τα κεφάλαια χωρίζονται με διαφορετικό τρόπο από εκείνον του σχολικού βιβλίου, έτσι ώστε η κατανόηση των διαφόρων εννοιών με τις οποίες θα απασχοληθείς, να γίνει ευκολότερη.

**Μεθοδολογία**  
 Όλες οι πληροφορίες που χρειάζεσαι για να λύσεις τα προβλήματα της Μεθοδολογίας.

Αυτό το τμήμα είναι ίσως το πιο σημαντικό! Είναι η Μεθοδολογία.



## LySTe - Τεστ Λυκείου

Η LySTe - Τεστ Λυκείου, όπως προσδιορίζεται κι από το όνομά της, Lyceum SOS Tests, είναι μια εφαρμογή ερωτήσεων Σωστού/Λάθους και Πολλαπλής Επιλογής οργανωμένων ανά κεφάλαιο για κινητά Android. Σκοπός της εφαρμογής είναι η ευκολότερη εκμάθηση και επανάληψη της θεωρίας των μαθημάτων όλων των τάξεων της Γ' Λυκείου, Ενιαίου και ΕΠΑΛ. Οι ερωτήσεις έχουν την επιμέλεια καθηγητών με πολυετή εμπειρία. Ο μαθητής έχει τη δυνατότητα να κάνει δυο ειδών τεστ σε κάθε μάθημα. Το πρώτο είδος είναι τεστ των 10 ερωτήσεων. Το δεύτερο είδος είναι τεστ που ολοκληρώνονται σε συγκεκριμένο χρόνο (ο χρόνος κυλάει αντίστροφα) και με δυνατότητα συγκεκριμένων λαθών (οι καρδιές μειώνονται) και ενεργοποιούνται από τις ρυθμίσεις. Το μεγάλο πλεονέκτημα αυτής της εφαρμογής είναι ότι, μετά την ολοκλήρωση κάθε τεστ, ο μαθητής μπορεί να δει σε ποιες ερωτήσεις έκανε λάθος και ποιες είναι οι σωστές απαντήσεις. Επιπλέον, όλα τα ολοκληρωμένα τεστ αποθηκεύονται για να μπορεί ο μαθητής να επανέλθει οποιαδήποτε στιγμή και να δει τα αποτελέσματα.



#### **1.4 Πλεονεκτήματα και μειονεκτήματα των υπαρχών εφαρμογών και σύγκριση με την δικιά μου εφαρμογή**

Ύστερα από έρευνα που έγινε πάνω στις υπάρχουσες εφαρμογές για εκμάθηση προγραμματιστικού περιβάλλοντος Γ' λυκείου, διαπιστώθηκε ότι σχεδόν όλες, ήταν ελλιπής ως προς το είδος των ερωτήσεων και την εξάσκηση πάνω στον κώδικα. Συγκεκριμένα σχεδόν όλες έχουν μόνο εξάσκηση της θεωρίας σε ερωτήσεις τύπου Σωστό-Λάθος ή πολλαπλής επιλογής, έτσι ο χρήστης δεν μπορεί να έρθει σε επαφή καθόλου με την δημιουργία κώδικα και την λογική της εκτέλεσης προγράμματος σε ΓΛΩΣΣΑ (Ονομασία της γλώσσας προγραμματισμού που χρησιμοποιείτε στο μάθημα).

Η μόνη εφαρμογή που έχει κώδικα είναι η "Α.Ε.Π.Π TEST 4U", με μικρό αριθμό ερωτήσεων, στην οποία δίνεται στον χρήστη ένα ολόκληρο πρόγραμμα όπου πρέπει να συμπληρώσει ή ένα μεγάλο τμήμα ψευδοκώδικα, πράγμα που κατά την άποψη μου έχει δύο μειονεκτήματα. Πρώτον, ύστερα από προσπάθειες που έκανα δεν κατάφερα να συμπληρώσω σωστά τον κώδικα διότι δεν γίνονται σαφές οι κανόνες που γράφεται ο κώδικας όπως π.χ αν οι δεσμευμένες λέξεις μπορούν να γραφούν με μικρά γράμματα ή εάν θέλουν τόνους κ.α. Οπότε δεν το θεωρώ πολύ φιλικό και εύχρηστο στον χρήστη. Δεύτερον, κατά την άποψη μου, δύσκολα κάποιος θα θέλει να γράψει ένα ολόκληρο πρόγραμμα από μια εφαρμογή σε ένα κινητό ή tablet με χρήση του πληκτρολογίου (keyboard) αφής.

Έτσι λοιπόν στην δικιά μου εφαρμογή περιορίστηκα στην συμπλήρωση μιας γραμμής κώδικα. Οι ερωτήσεις και τα τμήματα κώδικα δεν είναι πολύ σύνθετα και αποτελούν κώδικα αρκετά στοχευμένο για την εκμάθηση/έλεγχο γνώσεων πολύ απαραίτητων για την επιτυχία στις πανελλαδικές εξετάσεις. Αυτός ο τρόπος αποτελεί περισσότερο φιλικός προς τον χρήστη. Επίσης στην δικιά μου εφαρμογή υπάρχει η καρτέλα “Εμφάνιση τιμών ανά εντολή”, που βάζει τον χρήστη στην διαδικασία στο να εκτελέσει ένα κομμάτι κώδικα γραμμή κατά γραμμή και να σημειώσει τις αλλαγές των μεταβλητών του προγράμματος, πράγμα αρκετά χρήσιμο για την κατανόηση της εκτέλεσης ενός προγράμματος και την λειτουργία ενός Η/Υ. Η συγκεκριμένη άσκηση αποτελεί σχεδόν πάντα θέμα θεωρίας στις πανελλαδικές εξετάσεις, οπότε η αξία της ύπαρξής της αυξάνεται.

## 2. Έξυπνα τηλέφωνα(smartphones).

Το **smartphone**, ή με τον ελληνικό όρο **έξυπνο τηλέφωνο**, είναι ένα κινητό τηλέφωνο βασισμένο σε ένα λειτουργικό σύστημα κινητής τηλεφωνίας με περισσότερο προηγμένη υπολογιστική ικανότητα και συνδεσιμότητα σε σχέση με ένα συμβατικό κινητό τηλέφωνο. Τα πρώτα smartphones συνδύαζαν τις λειτουργίες ενός προσωπικού ψηφιακού βοηθού (PDA) και ενός κινητού τηλεφώνου. Σε μεταγενέστερα μοντέλα προστέθηκαν οι λειτουργίες των φορητών media players, low-end compact ψηφιακές φωτογραφικές μηχανές, βιντεοκάμερες τσέπης, καθώς και μονάδες πλοήγησης GPS με αποτέλεσμα την διαμόρφωση μιας πολυχρηστικής συσκευής. Πολλά σύγχρονα smartphones περιλαμβάνουν επίσης οθόνες αφής υψηλής ανάλυσης και web browsers που εμφανίζουν τυποποιημένες ιστοσελίδες, καθώς και βελτιστοποιημένες ιστοσελίδες για κινητά.



Εικόνα 1: smartphones

### 2.1 Λειτουργικό σύστημα

Λειτουργικό σύστημα ή ΛΣ ονομάζεται στην επιστήμη της πληροφορικής το λογισμικό του υπολογιστή που είναι υπεύθυνο για τη διαχείριση και τον συντονισμό των εργασιών, καθώς και την κατανομή των διαθέσιμων πόρων. Το λειτουργικό σύστημα παρέχει ένα θεμέλιο, ένα μεσολαβητικό επίπεδο λογικής διασύνδεσης μεταξύ λογισμικού και υλικού, διαμέσου του οποίου οι εφαρμογές αντιλαμβάνονται εμμέσως τον υπολογιστή. Μια από τις κεντρικές αρμοδιότητες του λειτουργικού συστήματος είναι η διαχείριση του υλικού, απαλλάσσοντας



έτσι το λογισμικό του χρήστη από τον άμεσο και επίπονο χειρισμό του υπολογιστή και καθιστώντας ευκολότερο τον προγραμματισμό τους.



Εικόνα 2 : Τυπική θέση του λειτουργικού συστήματος σε ένα υπολογιστικό σύστημα.

## 2.2 Λειτουργικά συστήματα για έξυπνα τηλέφωνα

Τα λειτουργικά συστήματα(OS) των κινητών τηλεφώνων που χρησιμοποιούνται από τα σύγχρονα smartphones περιλαμβάνουν, μεταξύ άλλων, το Android της Google, το iOS της Apple, το Symbian της Nokia , το BlackBerry OS της RIM, το Bada της Samsung, τα Windows Phone της Microsoft, το webOS της Hewlett-Packard, καθώς και ενσωματωμένες διανομές Linux όπως το Maemo και το MeeGo. Τέτοιου είδους λειτουργικά συστήματα μπορούν να εγκατασταθούν σε πολλά διαφορετικά μοντέλα κινητών τηλεφώνων και συνήθως κάθε συσκευή μπορεί να λάβει πολλές ενημερωμένες εκδόσεις λογισμικού λειτουργικού συστήματος κατά τη διάρκεια ζωής της. Μερικά άλλα επερχόμενα λειτουργικά συστήματα είναι το Firefox OS της Mozilla, το Ubuntu Phone της Canonical Ltd's και το Tizen.

## 2.3 Λειτουργικό σύστημα Android

Το Android είναι λειτουργικό σύστημα για συσκευές κινητής τηλεφωνίας το οποίο τρέχει τον πυρήνα του λειτουργικού Linux. Αρχικά αναπτύχθηκε από την Google και αργότερα από την Open Handset Alliance. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας την συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από την Google. Το Android είναι κατά κύριο λόγο σχεδιασμένο για συσκευές με οθόνη αφής, όπως τα έξυπνα τηλέφωνα και τα τάμπλετ με διαφορετικό περιβάλλον χρήσης για τηλεοράσεις (Android Tv), αυτοκίνητα (Android Auto) και ρολόγια χειρός (Android Wear). Παρόλο που έχει αναπτυχθεί για συσκευές με οθόνη αφής, έχει χρησιμοποιηθεί σε κονσόλες παιχνιδιών, ψηφιακές φωτογραφικές μηχανές, συνηθισμένους Η/Υ (π.χ. το HP Slate 21) και σε άλλες ηλεκτρονικές συσκευές.

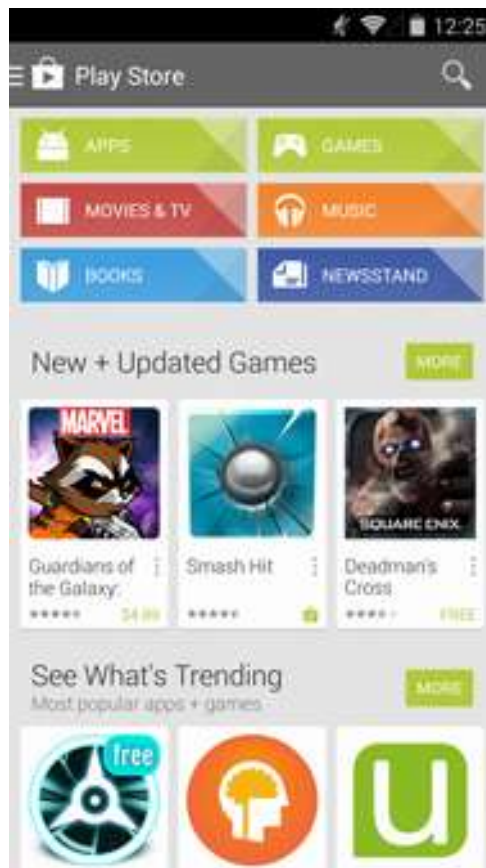


Εικόνα 3: Εικονίδιο Android

## 2.4 Εφαρμογές Android

Το Android έχει μια μεγάλη κοινότητα προγραμματιστών που γράφουν εφαρμογές, οι οποίες επεκτείνουν τη λειτουργικότητα των συσκευών. Οι εφαρμογές γράφονται σε μια προσαρμοσμένη έκδοση της JAVA και μπορεί κανείς να τις κατεβάσει από το online κατάστημα Google Play (πρώην Android Market) της Google όπως και από άλλα sites. Το Android Market, είναι ένα online κατάστημα της Google, προσφέρει σε κάθε χρήστη εφαρμογές για το κινητό του που είναι συμβατό με το λειτουργικό της Google. Το συντριπτικό ποσοστό των εφαρμογών είναι δωρεάν ενώ πλέον οι πληρωμένες εφαρμογές είναι διαθέσιμες και στο ελληνικό κοινό. Μέχρι τον Ιούνιο του 2014 περισσότερες από 1,2 εκατομμύρια εφαρμογές ήταν διαθέσιμες για Android

ενώ ο αριθμός των downloads από το Google Play μέχρι το Ιούλιο του 2013 είχε υπερβεί τα 50 δισεκατομμύρια.



Εικόνα 4: Εφαρμογές από google play

## 2.5 Ιστορία εκδόσεων του Android

Η ιστορία εκδόσεων του Android του λειτουργικού συστήματος των κινητών ξεκίνησε με την κυκλοφορία του Android beta το Νοέμβριο του 2007. Η πρώτη εμπορική έκδοση ήταν το Android 1.0 που κυκλοφόρησε το Σεπτέμβριο του 2008.

Το Android είναι υπό συνεχή ανάπτυξη από την Google και την Open Handset Alliance(OHA), και έχουν γίνει μια σειρά από ενημερώσεις στην λειτουργία του συστήματος από την αρχική κυκλοφορία του.



Εικόνα 4: εκδόσεις του Android

## 2.6 Αρχιτεκτονική Android

Το Android αποτελείτο από ορισμένες συνιστώσες λογισμικού οι οποίες συνθέτουν ένα ενιαίο και ολοκληρωμένο σύστημα. Έτσι, το σύστημα αυτό μπορεί να παρέχει τα μέσα που απαιτούνται για την χρήση νέων εφαρμογών όπως άλλωστε συμβαίνει και με τα λειτουργικά συστήματα των ηλεκτρονικών υπολογιστών. Όπως φαίνεται στην εικόνα 5, το Android αποτελείτο από 4 επίπεδα και από 5 ομάδες συνιστωσών τα οποία περιγράφονται παρακάτω ξεκινώντας από τα χαμηλότερα προς τα υψηλότερα επίπεδα:

### Linux Kernel

Το Android βασίζεται στον πυρήνα του Linux για βασικές λειτουργίες όπως η διαχείριση των drivers της συσκευής, διαχείριση μνήμης, διαχείριση διεργασιών καθώς και δικτύωσης που συνεπάγεται την διαχείριση των διεπαφών δικτύου που διαθέτει κάθε συσκευή(π.χ GSM, HSDPA, WiFi, Bluetooth Κτλ).

## Native Libraries

Οι βιβλιοθήκες του Android είναι γραμμένες στις γλώσσες C και C++ και μπορούν να χρησιμοποιηθούν μέσω κατάλληλου interface της Java. Μερικές από τις κυριότερες είναι

1. η βιβλιοθήκη Surface Manager για την δημιουργία παραθύρων καθώς και δισδιάστατων και τρισδιάστατων γραφικών
2. η βιβλιοθήκη Media Framework που περιέχει αποκωδικοποιητές για αναπαραγωγή αρχείων πολυμέσων όπως MPEG, MP3 κτλ
3. η βιβλιοθήκη SQLite για την υποστήριξη της βάσης δεδομένων
4. η βιβλιοθήκη WebKit για την υποστήριξη των φυλλομετρητών (browsers).

## Android Runtime

Όπως φαίνεται και στην παρακάτω εικόνα η συνιστώσα του android runtime αποτελείται από:

- Βασικές βιβλιοθήκες για την διεπαφή των εφαρμογών Java με το περιβάλλον της συσκευής στην οποία εκτελούνται.
- Τη Dalvik Virtual Machine η οποία είναι υπεύθυνη για την δημιουργία των εκτελέσιμων αρχείων των εφαρμογών προκειμένου να τα “τρέξει” το λειτουργικό σύστημα.

Κάθε εφαρμογή του Android είναι γραμμένη κυρίως σε γλώσσα Java την οποία το λειτουργικό σύστημα δεν την αντιλαμβάνεται απευθείας. Για τον λόγο αυτό η Dalvik Virtual Machine αναλαμβάνει τη δημιουργία των εκτελέσιμων αρχείων \*.dex (Dalvik Executable) τα οποία εκτελούνται από την δική τους Virtual Machine, ακόμα και όταν εκτελούνται παράλληλα, με αποτέλεσμα τα διαφορετικά προγράμματα να μην επηρεάζονται το ένα από το άλλο και σε περίπτωση που προκύψει σφάλμα σε κάποιο από αυτά να μην προκαλέσει πρόβλημα στα υπόλοιπα.

## Application Framework

Εφόσον το Android προσφέρει μια ανοικτή πλατφόρμα ανάπτυξης εφαρμογών είναι επόμενο ορισμένες από τις εφαρμογές να είναι αρκετά προχωρημένες και καινοτόμες. Οι εφαρμογές έχουν πρόσβαση στις βασικές βιβλιοθήκες του λειτουργικού συστήματος, μέσω κατάλληλων διεπαφών και μέσω του Application Framework μπορούν με τη σειρά τους να παρέχουν επιπρόσθετες λειτουργίες-υπηρεσίες προς άλλες εφαρμογές, εφόσον κάτι τέτοιο φυσικά δεν περιορίζεται από τις

πολιτικές ασφάλειας του Application Framework.Μερικές από τις βασικές οντότητες που περιλαμβάνονται στο πλαίσιο του Application Framework είναι :

- **View System:** Επιτέπει την χρήση πλαισίων,πεδίων κειμένου,κουμπιών κλπ.
- **Content Provides:** Επιτρέπει στις εφαρμογές την πρόσβαση σε δεδομένα άλλων εφαρμογών ή τον διαμοιρασμό των δικών τους δεδομένων,όπως οι επαφές.
- **Resource Manager:** Παρέχει την πρόσβαση σε πόρους όπως γραφικά και σε αρχεία σχετικά με την διάταξη των στοιχείων του γραφικού περιβάλλοντος. Απλούστερα,ότι δεν είναι κώδικας.
- **Notification Manager :** Διαχειρίζεται τα μηνύματα των εφαρμογών που εμφανίζονται στην status bar,όπως εισερχόμενα μηνύματα,ραντεβού κτλ.
- **Activity Manager :** Διαχειρίζεται τον κύκλο ζωής των εφαρμογών και παρέχει την δυνατότητα μετάβασης στις προγενέστερες καταστάσεις τους.

## Applications

Στην ομάδα των Application βρίσκονται οι εφαρμογές που θα χρησιμοποιούν τελικά οι χρήστες με διαφάνεια ως προς το τι συμβαίνει πίσω από αυτές ή το τι απαιτείται για την εκτέλεση τους από το λειτουργικό σύστημα.Μερικές από τις πιο γνωστές εφαρμογές είναι ο browser,email,client,αποστολή και λήψη SMS,προβολή χαρτών σε συνδυασμό με το στίγμα της συσκευής εάν διαθέτει δέκτη GPS,ημερολόγιο,διαχείριση επαφών,παιχνίδια,RSS readers και πολλές άλλες. Όλες οι εφαρμογές όπως έχει ήδη αναφερθεί μπορούν να τρέχουν πολλές παράλληλα χωρίς να επηρεάζει η μια την άλλη.



**Εικόνα 5 : Διάγραμμα αρχιτεκτονικής του Android**

## 2.7 Android Studio

Το Android Studio είναι ένα ολοκληρωμένο προγραμματιστικό περιβάλλον (IDE) για ανάπτυξη εφαρμογών στην πλατφόρμα Android. Ανακοινώθηκε στις 16 Μαΐου 2013 στο συνέδριο Google από την Google Product Manager, Katherine Chou. Το Android Studio είναι διαθέσιμο ελεύθερα με την άδεια Apache License 2.0.

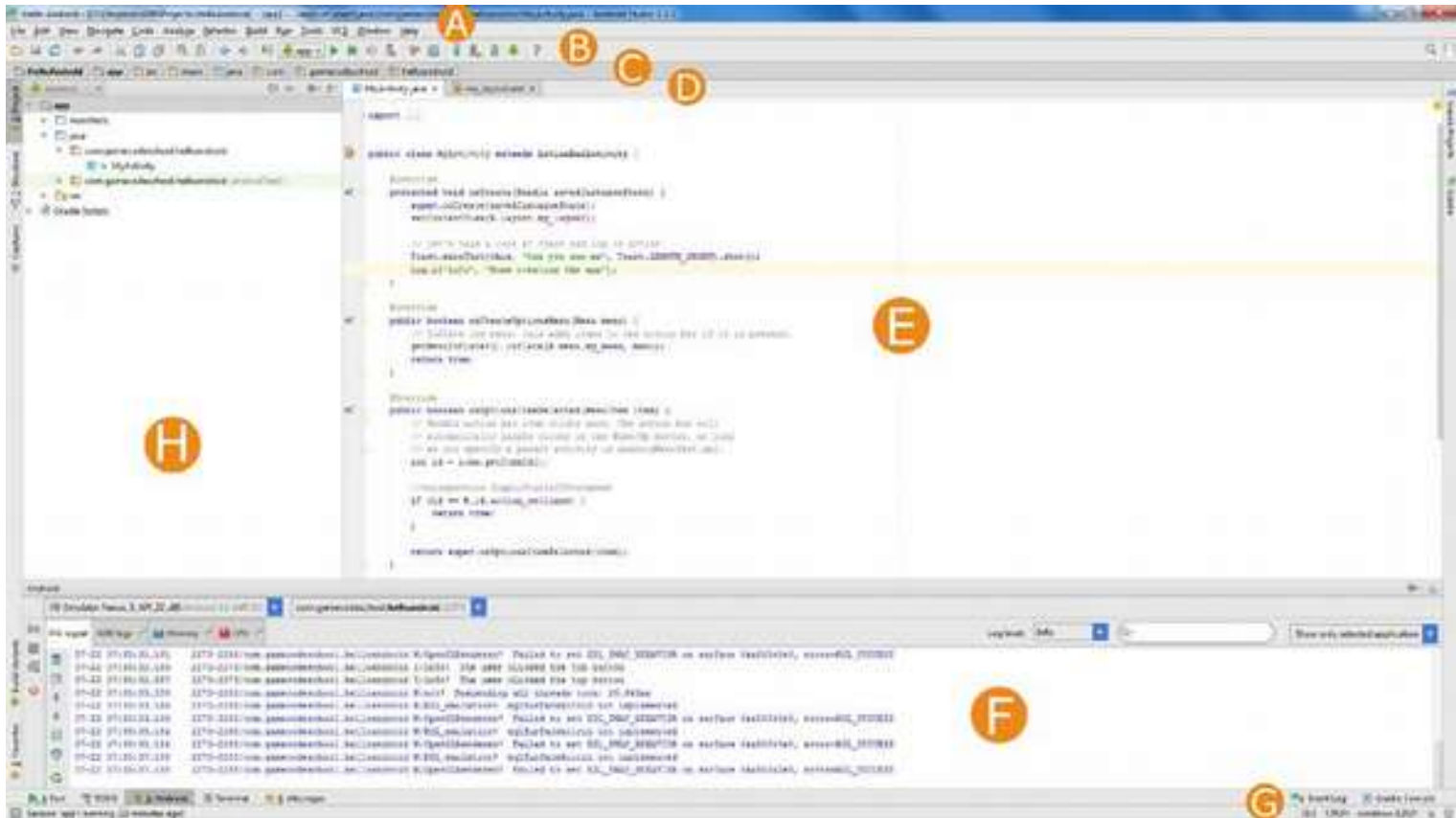
Το Android Studio ήταν διαθέσιμο σε πρώιμο στάδιο για προεπισκόπηση ξεκινώντας από την έκδοση 0.1 τον Μάιο του 2013, έπειτα ξεκίνησε το δοκιμαστικό στάδιο από την έκδοση 0.8 που βγήκε τον Ιούνιο του 2014. Η πρώτη σταθερή έκδοση βγήκε το Δεκέμβριο του 2014, με την έκδοση 1.0.

Βασισμένο στο λογισμικό της JetBrains' IntelliJ IDEA, το Android Studio σχεδιάστηκε αποκλειστικά για προγραμματισμό Android . Είναι διαθέσιμο για Windows, Mac OS και Linux και αντικατέστησε τα Eclipse Android Development Tools (ADT) ως το κύριο IDE της Google για ανάπτυξη εφαρμογών Android.

### User interface

Όπως φαίνεται και στην παρακάτω εικόνα, το κεντρικό παράθυρο του Android Studio αποτελείται από τα παρακάτω επιμέρους λογικά τμήματα.





**Εικόνα 6 : Android studio**

- A) Αυτή είναι η μπάρα menu(menu bar).Όπως και στην περισσότερες εφαρμογές μπορούμε να κάνουμε τις περισσότερες επιλογές από εδώ.
- B) Αυτή είναι μπάρα εργαλείων.Από εδώ μπορούμε να έχουμε πρόσβαση στα περισσότερα quick launch icons.Από εδώ οι πιο σύνηθες επιλογές γίνονται εύκολα με ένα click .
- C) Αυτή είναι μπάρα πλοήγησης(navigation bar).Μας δείχνει την θέση του αρχείου που είναι ανοιχτό στον editor windows και μας επιτρέπει να πλοηγηθούμε ανάμεσα στα αρχεία και φακέλους του project.
- D)Αυτή είναι μπάρα editor tabs. Μπορούμε να κλικάρουμε σε ένα tab και να δούμε τα περιεχόμενα του editor.Εδώ μπορούμε να μπισθέσουμε αρχεία πατώντας διπλό κλικ στο project explorer.
- E) Αυτός είναι ο editor. Εδώ θα ξοδέψουμε τον περισσότερο χρόνο μας.Εδώ βλέπουμε και επεξεργαζόμαστε τον κώδικα που έχουν τα αρχεία.
- F) Αυτό είναι η κονσόλα.Εδώ μπορεί να δει κανείς το logcat.Επίσης μπορεί να δει τα runtime errors καθώς τρέχει το πρόγραμμα.
- H) Αυτό το μέρος αποτελεί το project explorer.Εδώ μπορεί ο χρήστης να περιηγηθεί ανάμεσα στα αρχεία και τους φακέλους του project και να ανοίξει ένα οποιοδήποτε αρχείο.

- G) Αυτό το μέρος αποτελεί το status bar.Εδώ μπορεί να ενημερωθεί ο χρήστης αν το Android Studio κάνει κάποια δουλειά για εμάς.

Ο χρήστης έχει τη δυνατότητα να οργανώσει όπως θέλει την εμφάνιση του κεντρικού παραθύρου του Android Studio κρύβοντας ή μετακινώντας γραμμές εργαλείων και παράθυρα.Μπορεί επίσης να χρησιμοποιήσει συντομεύσεις πληκτρολογίου για άμεση πρόσβαση σε πολλά χαρακτηριστικά του IDE.Επίσης ο χρήστης μπορεί ανά πάσα στιγμή, να πραγματοποιήσει αναζήτηση σε όλο τον πηγαίο κώδικα τις βάσεις δεδομένων,τα στοιχεία της διεπαφής χρήστη(user interface),και ούτω καθεξής,με διπλό πάτημα του πλήκρου shift ή κάνοντας κλικ στο μεγενθυτικό φακό στην επάνω δεξιά γωνία παραθύρου του Android studio.

Η δομή των αρχείων ενός project στο Android Studio και η παρουσίαση τους στο

## User Interface

Από προεπιλογή το Android Studio εμφανίζει παρουσιάζει τα αρχεία του project σε Android view. Αυτό δεν αντικατοπτρίζει την πραγματική ιεραρχία των

φακέλων στο σκληρό δίσκο, αλλά είναι οργανωμένο με βάση τα επιμέρους modules και τους διαφορετικούς τύπους αρχείων με σκοπό την απλοποίηση της πλοήγησης του χρήστη μεταξύ σημαντικών αρχείων κλειδιών του project, κρύβοντας ορισμένα

αρχεία ή φακέλους που συνήθως δεν χρησιμοποιούνται Ο χρήστης, έχοντας επιλέξει το Android View, μπορεί να βρει τα αρχεία ρυθμίσεων για τη διαδικασία



Εικόνα 7: Android View

“χτισίματος”(build) της εφαρμογής, τα manifest αρχεία για κάθε module, όπως και όλα τα εναλλακτικά resources με βάση το είδος τους, ομαδοποιημένα στο ίδιο μέρος και όχι σε διαφορετικούς φακέλους όπως αυτά βρίσκονται πραγματικά στον σκληρό δίσκο.

Μέσα σε κάθε Android app module, τα αρχεία βρίσκονται ομαδοποιημένα στις ομάδες “manifests”, “java” και “res”. Η ομάδα “manifests” περιέχει το αρχείο AndroidManifest.xml, η ομάδα “java” περιέχει τα αρχεία πηγαίου κώδικα Java, διαχωρισμένα με βάση το package name τους και η ομάδα “res” περιέχει όλους τους πόρους που δεν έχουν μορφή κώδικα, όπως είναι τα XML layouts, User interface strings, εικόνες bitmap κ.α., διαχωρισμένους σε κατάλληλες υποομάδες.

Για να δει ο χρήστης την πραγματική δομή των αρχείων του project, όπως αυτά βρίσκονται στο σκληρό, συμπεριλαμβανομένων και κάποιων αρχείων που αποκρύπτονται στο Android view, πρέπει να επιλέξει το “Project” στο παράθυρο του project, εκεί που ήταν προεπιλεγμένο το “Android” (είχαμε δηλαδή το Android view ως προεπιλογή). Με την επιλογή του Project view ο χρήστης έχει τη δυνατότητα να δει πολλά περισσότερα αρχεία και φακέλους, από τα οποία τα πιο σημαντικά είναι:

## **modulename/**

### **build/**

Περιέχει διαφορετικά build options.

### **libs/**

Περιέχει ιδιωτικές βιβλιοθήκες της εφαρμογής.

### **src/**

Περιέχει όλα τα αρχεία κώδικα και πόρων για το συγκεκριμένο module στους εξής υποφακέλους:

#### **androidTest/**

Περιέχει κώδικα για δοκιμές της εφαρμογή σε μια συσκευή Android.

### **main/**

Περιέχει τα “κύρια” αρχεία: τον κώδικα και τους πόρους που είναι κοινά για όλα τα διαφορετικά builds.

AndroidManifest.xml

Περιγράφει τη φύση της εφαρμογής και κάθε ένα από τα συστατικά της μέρη.

**Java/**

Περιέχει τα αρχεία πηγαίου κώδικα.

**Jni/**

Περιέχει εγγενή κώδικα χρησιμοποιώντας το java Native interface(JNI).

**gen/**

Περιέχει τα αρχεία Java που δημιουργούνται από το Android Studio, όπως το αρχείο R.Java.

**Res/**

Περιέχει τους πόρους της εφαρμογής όπως τα drawable αρχεία, τα layout αρχεία, τα User Interface strings κ.α.

**assets/**

Περιέχει αρχεία τα οποία θα πρέπει να μεταγλωττιστούν σε αρχείο .apk όπως είναι, χωρίς να υποστούν αλλαγές.

**Test/**

Περιέχει κώδικα για τις τοπικές δοκιμές που εκτελούνται σε τοπική JVM.

**Build.grande(module)**

Αυτό καθορίζει το build configuration(διαμόρφωση κατασκευής) για το συγκεκριμένο Module.

**Build.grande(project)**

Αυτό καθορίζει το build configuration(διαμόρφωση κατασκευής) για όλα τα modules. Αυτό το αρχείο είναι αναπόσπαστο μέρος του project και θα πρέπει να ελέγχεται συχνά για τυχόν αναθεωρήσεις όπως και όλος ο πηγαίος κώδικας της εφαρμογής. Το grande είναι ένα εργαλείο το οποίο αυτοματοποιεί την διαδικασία μεταγλώττισης και της δημιουργίας του εκτελέσιμου αρχείου .apk .

### 3. Βάσεις δεδομένων που χρησιμοποιεί η εφαρμογή και τρόπος επικοινωνίας της εφαρμογής με την βάση.

Οι βάσεις δεδομένων είναι χτισμένες πάνω στην εφαρμογή xampp.

Το **XAMPP** είναι ένα πακέτο προγραμμάτων [ελεύθερου λογισμικού](#), [λογισμικού ανοικτού κώδικα](#) και [ανεξαρτήτου πλατφόρμας](#) το οποίο περιέχει το [εξυπηρετητή ιστοσελίδων http Apache](#), την βάση δεδομένων [MySQL](#) και ένα [διερμηνέα](#) για κώδικα γραμμένο σε [γλώσσες προγραμματισμού PHP](#) και [Perl](#).

Ο **Apache HTTP** γνωστός και απλά σαν Apache είναι ένας [εξυπηρετητής του παγκόσμιου ιστού](#) (web). Όποτε ένας χρήστης επισκέπτεται ένα [ιστότοπο](#) το πρόγραμμα πλοήγησης (browser) επικοινωνεί με έναν [διακομιστή](#) (server) μέσω του πρωτοκόλλου [HTTP](#), ο οποίος παράγει τις ιστοσελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης. Ο Apache είναι ένας από τους δημοφιλέστερους [εξυπηρετητές ιστού](#), εν μέρει γιατί λειτουργεί σε διάφορες πλατφόρμες όπως τα [Windows](#), το [Linux](#), το [Unix](#) και το [Mac OS X](#). Κυκλοφόρησε υπό την άδεια λογισμικού Apache και είναι [λογισμικό ανοικτού κώδικα](#). Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation). Ο Apache χρησιμοποιείται και σε τοπικά δίκτυα σαν [διακομιστής συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων](#) π.χ. [Oracle](#), [MySQL](#).

Η **MySQL** είναι ένα σύστημα διαχείρισης [σχεσιακών βάσεων δεδομένων](#) που μετρά περισσότερες από 11 εκατομμύρια εγκαταστάσεις. Έλαβε το όνομά της από την κόρη του Μόντυ Βιντένιους, τη Μάι ([αγγλ. My](#)). Το πρόγραμμα τρέχει έναν [εξυπηρετητή](#) (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο [βάσεων δεδομένων](#).

Η **PHP** (PHP: Hypertext Preprocessor) είναι μια [γλώσσα προγραμματισμού](#) για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό [διακομιστή](#) του Παγκόσμιου Ιστού (π.χ. [Apache](#)), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που είτε θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα [HTML](#) ή θα επεξεργασθεί τις εισόδους δίχως να προβάλλει την έξοδο στο χρήστη, αλλά θα τις μεταβιβάσει σε κάποιο άλλο PHP script.



**XAMPP** Εικόνα 8: xampp logo

### 3.1 Περιεχόμενα της βάσης (βάσεις και πίνακες).

Η εφαρμογή Android χρησιμοποιεί δέκα βάσεις δεδομένων. Οι οκτώ από αυτές αποτελούν τα οκτώ κεφάλαια του μαθήματος (ΑΕΠΠ), τα ονόματά τους είναι “kefalaio\_1”, “kefalaio\_2”.. και αντιστοιχούν δηλαδή με τον αριθμό των κεφαλαίων που είναι εντός ύλης. Αυτές οι βάσεις περιέχουν στην πλειοψηφία τους το σύνολο των πινάκων:

- erwt\_sl
- emfanish\_timwm\_ana\_entolh
- erwt\_smp\_kenou
- erwt\_smp\_kwdika

Ο πίνακας “erwt\_sl” περιέχει τα πεδία (στήλες), ΑΕ, Erwthsh, Apanthsh που είναι όλα τύπου text. Αυτός ο πίνακας χρησιμοποιείται για να καλύψει τις ανάγκες της εφαρμογής για ερωτήσεις τύπου “ΣΩΣΤΟ-ΛΑΘΟΣ”. Το πεδίο “ΑΕ” χρησιμοποιείται για το δηλώσει τον αριθμό της ερώτησης, το πεδίο “Erwthsh” για την ερώτηση και το πεδίο “Apanthsh” έχει μόνο δύο τιμές το “0” και το “1”, όπου το ένα σημαίνει ότι η ερώτηση έχει για απάντηση “Σωστό” ενώ το μηδέν έχει για απάντηση το Λάθος.

Ο πίνακας “emfanish\_timwm\_ana\_entolh” περιέχει τα πεδία (στήλες), ΑΕ, Erwthsh, value\_1, value\_2, value\_3, value\_4, numofvalues που είναι όλα τύπου text. Αυτός ο πίνακας χρησιμοποιείται για να καλύψει τις ανάγκες της εφαρμογής για ερωτήσεις τύπου “εμφάνιση τιμών ανά εντολή” όπως θα επεξηγηθεί και στο παρακάτω κεφάλαιο. Το πεδίο “ΑΕ” χρησιμοποιείται για το δηλώσει τον αριθμό της ερώτησης, το πεδίο “Erwthsh” για την ερώτηση, το “code” περιέχει τμήμα κώδικα ή ολόκληρο πρόγραμμα ή αλγόριθμο, τα values\_1 έως values\_4 τις αλλαγές των μεταβλητών που εξετάζονται καθώς τρέχει το πρόγραμμα, και το numofvalues πόσες μεταβλητές εξετάζονται ανάλογα με το ερώτημα.

Ο πίνακας “erwt\_smp\_keno” περιέχει τα πεδία (στήλες), ΑΕ, Erwthsh, Apanthsh που είναι όλα τύπου text. Αυτός ο πίνακας χρησιμοποιείται για να καλύψει τις ανάγκες της εφαρμογής για ερωτήσεις τύπου “ερωτήσεις συμπλήρωσης κενού”. Το πεδίο “ΑΕ” χρησιμοποιείται για το δηλώσει τον αριθμό της ερώτησης, το πεδίο “Erwthsh” για την ερώτηση και το πεδίο “Apanthsh” περιέχει την λέξη ή λέξεις που λείπουν.

Ο πίνακας “erwt\_smp\_kwdika” περιέχει τα πεδία (στήλες), ΑΕ, Erwthsh, Apanthsh, code που είναι όλα τύπου text. Αυτός ο πίνακας χρησιμοποιείται για να καλύψει τις ανάγκες της εφαρμογής για ερωτήσεις τύπου “ερωτήσεις συμπλήρωσης κώδικα” όπου ο χρήστης συμπληρώνει μια γραμμή κώδικα από ένα πρόγραμμα που δίνεται. Το πεδίο “ΑΕ” χρησιμοποιείται για το δηλώσει τον αριθμό της ερώτησης, το πεδίο “Erwthsh” για την ερώτηση, το πεδίο “Apanthsh” περιέχει την γραμμή κώδικα που λείπει και το “code” το τμήμα κώδικα.

Υπάρχει άλλη μια βάση με το όνομα “arotimhsh” και περιέχει τον πίνακα expr\_table με τα πεδία AE,Erwthsh,Aranthsh που είναι όλα τύπου text..

Η τελευταία βάση είναι “user\_db” που περιέχει τον πίνακα user\_info με πεδία name,email,user\_name,password που είναι όλα τύπου text.Χρησιμοποιείται για log in και register στην εφαρμογή Android.

### 3.2 Επικοινωνία εφαρμογής Android με xamarin και τις βάσεις δεδομένων.

Η σύνδεση της εφαρμογής με το xamarin γίνεται με την βοήθεια και την χρήση της βιβλιοθήκης **Volley Android**, η οποία υπάρχει διαθέσιμη στο Android studio.Μπορεί να προστεθεί στο project πολύ εύκολα με μία εντολή στο αρχείο build.gradle της εφαρμογής :

```
dependencies {
...
compile 'com.android.volley:volley:1.1.0'
}
```

Η βιβλιοθήκη volley είναι μια βιβλιοθήκη HTTP, η οποία απλοποιεί και επιταχύνει την διαδικασία σύνδεσης σε δικτύο.Μερικά από τα πλεονεκτήματα της είναι:

- Αυτόματος προγραμματισμός αιτημάτων δικτύου.
- Πολλαπλές ταυτόχρονες συνδέσεις δικτύου.
- Διαφανής αποθήκευση απόκρισης δίσκου και μνήμης με συνηθισμένη συνοχή της κρυφής μνήμης HTTP.
- Ευκολία προσαρμογής, για παράδειγμα, για επανάληψη και αναστροφή.

Τα δεδομένα στέλνονται στον server σε μορφή αλφαριθμητικών (strings) και λαμβάνονται επίσης στην ίδια μορφή.Αυτό γίνεται με την βοήθεια του αντικειμένου (Object) StringRequest το οποίο ανήκει στην βιβλιοθήκη Volley.Τα δεδομένα που λαμβάνονται από τον server έχουν την μορφή JSON .

Η **JavaScript Notation Object** ή το **JSON** είναι μια μορφή αρχείου ανοικτού προτύπου που χρησιμοποιεί κείμενο αναγνώσιμο από άνθρωπο για τη μετάδοση αντικειμένων δεδομένων που αποτελούνται από ζεύγη χαρακτηριστικών-τιμών και τύπους δεδομένων συστοιχιών (ή οποιαδήποτε άλλη σειριοποιήσιμη τιμή). Είναι μια πολύ κοινή μορφή δεδομένων που χρησιμοποιείται για την ασύγχρονη επικοινωνία προγράμματος περιήγησης-διακομιστή, συμπεριλαμβανομένης της αντικατάστασης της XML σε ορισμένα συστήματα τύπου AJAX.

Το JSON είναι μια μορφή δεδομένων ανεξάρτητη από τη γλώσσα. Προέρχεται από

τη JavaScript, αλλά από το 2017 πολλές γλώσσες προγραμματισμού περιλαμβάνουν κώδικα για τη δημιουργία και την ανάλυση δεδομένων JSON. Ο επίσημος τύπος μέσου διαδικτύου για το JSON είναι η εφαρμογή / json. Τα ονόματα αρχείων JSON χρησιμοποιούν την επέκταση .json.

Έτσι όταν λαμβάνεται η απάντηση από τον server έρχεται σε μορφή JSON και χρησιμοποιώ τις μεθόδους JSONArray και JSONObject της βιβλιοθήκης json για την προεπεξεργασία της απάντησης.

### 3.3 Αρχεία PHP που δημιουργήθηκαν στο xampp

Στα δημόσια έγγραφα του εξυπηρετητή xampp υπάρχουν αρχεία σε γλώσσα PHP τα οποία βοηθάνε στην επικοινωνία της εφαρμογής με τις βάσεις. Στα αρχεία αυτά υπάρχει κώδικας για την σύνδεση της εφαρμογής με τις βάσεις και SQL queries για την επεξεργασία των πινάκων και την ανάκτηση των δεδομένων από τους πίνακες των βάσεων. Ένα παράδειγμα είναι το αρχείο **init.php** το οποίο περιέχει τον κώδικα:

```
<?php
$db_name = "user_db";
$mysql_user = "root";
$mysql_pass = "root";

$server_name = "localhost";

$con = mysqli_connect($server_name,$mysql_user,$mysql_pass,$db_name);
if (!$con){
//echo "Connection Error.. " .mysqli_connect_error();
}
else {
//echo "<h3>Database connection Succes... </h3>";
}

?>
```



Το παραπάνω αρχείο υπάρχει για την δημιουργία μιας σύνδεσης sql με την βάση “user\_db”. Χρησιμοποιείται από τα αρχεία login\_json.php και register.php. Το αρχείο [login\\_json.php](#) περιέχει τον κώδικα:

```
<?php

require "init.php";
$user_name = $_POST["user_name"];
$password = $_POST["password"];

$sql = "select name,email from user_info where user_name like '". $user_name.'" and
password like '". $password.'"";

$result = mysqli_query($con,$sql);
$response = array();

if(mysqli_num_rows($result)>0)
{
    $row = mysqli_fetch_row($result);
    $name = $row[0];
    $email = $row[1];
    $code = "login success";

    array_push($response,array("code"=>$code,"name"=>$name,"email"=>$email));
    echo json_encode($response);
}
else
{
    $code = "login failed";
    $message = "User not found...Please try again...";
    array_push($response,array("code"=>$code,"message"=>$message));
    echo json_encode($response);
}

mysqli_close($con);

?>
```

Το αρχείο αυτό ( `login_json.php` ) χρησιμοποιείται από την εφαρμογή Android για την είσοδο του χρήστη στην εφαρμογή. Ο τρόπος λειτουργίας είναι ο εξής:

Αρχικά στην εφαρμογή Android δημιουργείται μια σύνδεση με τον server με την κλήση του αντικειμένου `StringRequest`. Μία από τις παραμέτρους αυτού του αντικειμένου είναι ένα URL. Έτσι για το αρχείο `login_json.php` δίνεται το αντίστοιχο URL, π.χ `String login_url="http://192.168.1.7/manos/login_json.php";` Επίσης δίνεται στο αντικείμενο `StringRequest` ως παράμετρος η μέθοδος που στέλνονται τα δεδομένα και στην συγκεκριμένη περίπτωση είναι η μέθοδος `POST`.

Κώδικας java:

```
StringRequest stringRequest = new StringRequest(Request.Method.POST,
login_url,
...

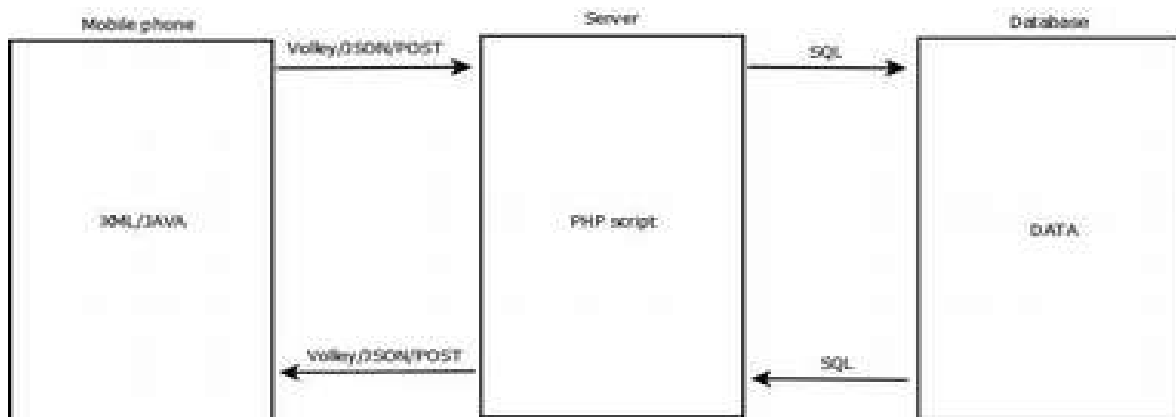
```

Έτσι μετά την κλήση, “τρέχει” ο κώδικας του αρχείου στον server.

Αφού αρχίσει να τρέχει ο κώδικας στο PHP αρχείο και γίνει η σύνδεση στην βάση με την βοήθεια του αρχείου `init.php`, γίνονται `POST` το `user_name` και το `password` από την εφαρμογή android στον κώδικα PHP. Ο κώδικας PHP συνεχίζει και ελέγχει αν υπάρχουν στον πίνακα `user_info` το `user_name` και το `password` που έδωσε ο χρήστης, με την χρήση `sql queries`. Έτσι αν υπάρχει εγγραφή στον πίνακα που να αντιστοιχούν, επιστρέφει το αντίστοιχο μήνυμα σε μορφή `JSON encode`. Τέλος το αρχείο PHP τελειώνει με τον τερματισμό της σύνδεσης στην βάση.

Αυτός είναι ο τρόπος επικοινωνίας με το `xampp` και συνολικά έχουν συγγραφεί 6 αρχεία PHP για την εξυπηρέτηση της εφαρμογής τα οποία είναι:

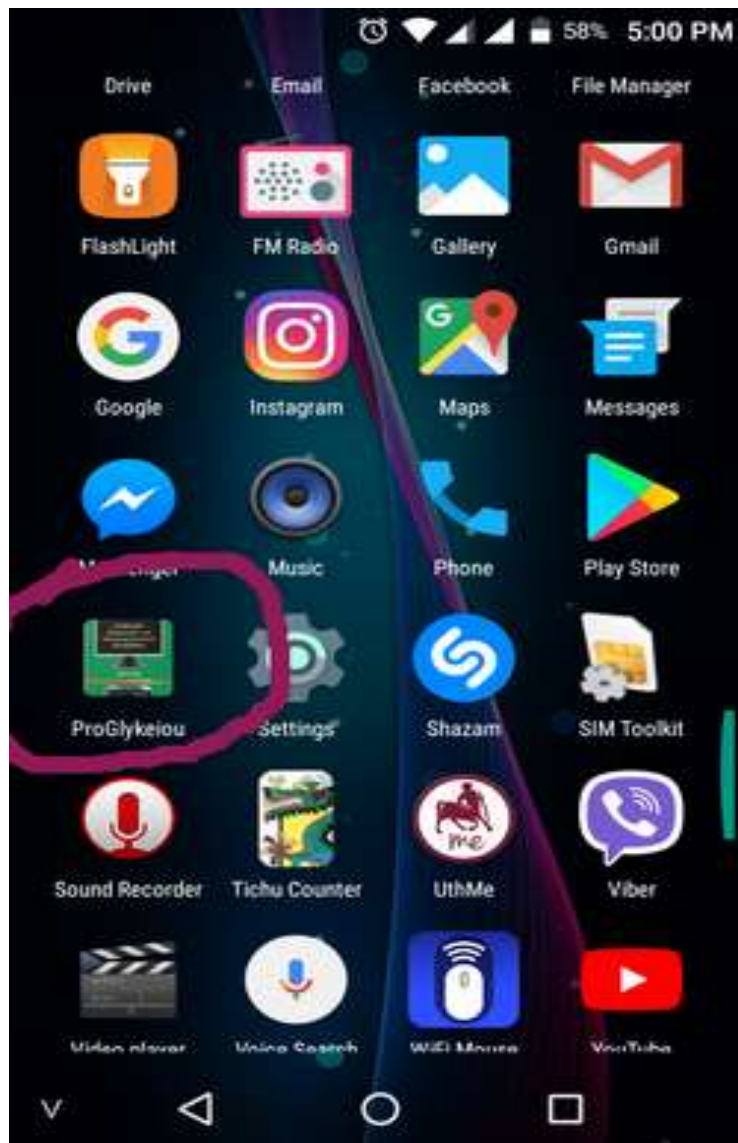
- `init.php`
- `json_get_data.php`
- `json_get_data_2.php`
- `json_get_data_3.php`
- `login_json.php`
- `register.php`



Εικόνα 9: Περιγραφή σύνδεσης με την βάση δεδομένων.

#### 4. Η εφαρμογή ProGlykeiou

Αφού μελέτησα την υπάρχουσα κατάσταση στον τομέα της εκπαίδευσης λογισμικού και θέλοντας να εισάγω σε αυτόν μια εφαρμογή βοήθημα για την εκμάθηση προγραμματισμού Γ' λυκείου, απεφάσισα να υλοποιήσω αυτήν την εφαρμογή. Το όνομα της ProGlykeiou προέρχεται ετυμολογικά από τα συνθετικά Prog + Glykeiou, δηλαδή προγραμματισμός Γ' λυκείου. Στην παρακάτω εικόνα φαίνεται το λογότυπο της εφαρμογής.



Εικόνα 10 : ProGlykeiou logo

## 4.1 Το ProGlykeiou ως εκπαιδευτικό λογισμικό

Το ProGlykeiou είναι μια εφαρμογή που απευθύνεται σε παιδιά λυκείου. Η εφαρμογή αυτή προσφέρει την δυνατότητα στους μαθητές Γ' λυκείου να εξασκηθούν στο μάθημα του προγραμματισμού πάνω σε διάφορες καρτέλες ερωτήσεων. Υπάρχει καρτέλα που προσφέρει εξάσκηση πάνω σε ερωτήσεις Σωστού-Λάθους, καρτέλα συμπλήρωσης κενού, καρτέλα συμπλήρωσης κώδικα, καρτέλα συμπλήρωσης πίνακα με τις αλλαγές των τιμών των μεταβλητών καθώς εκτελείτε ένα τμήμα κώδικα, καρτέλα με την θεωρία-SOS του μαθήματος και καρτέλα για εξάσκηση στην αποτίμηση λογικών εκφράσεων (Αληθής-Ψευδής). Δίνεται η δυνατότητα προβολής της ορθής απάντησης, ώστε ο χρήστης να πάρει την απαραίτητη γνώση άμεσα και να "προβληματιστεί" με την απάντηση.

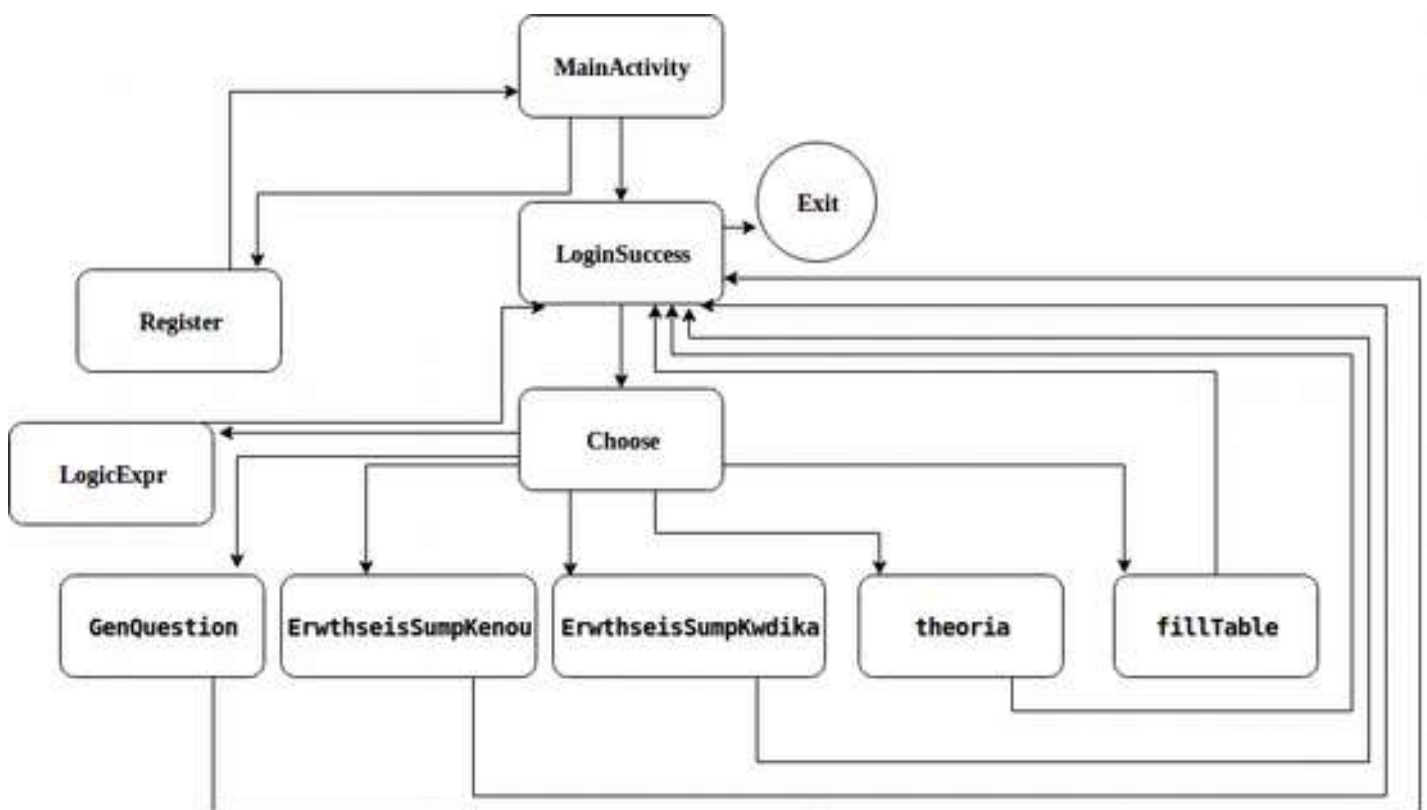
## 4.2 Υλοποίηση της εφαρμογής ProGlykeiou

Όπως έχει αναφερθεί και στο κεφάλαιο ένα το βασικό εργαλείο που έχει χρησιμοποιηθεί για την υλοποίηση της εφαρμογής είναι το Android studio 3.1. Όσον αφορά το γραφικό κομμάτι, η γλώσσα προγραμματισμού που έχει χρησιμοποιηθεί είναι η xml, μια δυνατότητα που προσφέρει το Android studio. Όσον αφορά τις υπόλοιπες λειτουργίες της εφαρμογής και το αλγοριθμικό κομμάτι, η γλώσσα προγραμματισμού που έχει χρησιμοποιηθεί είναι η Java. Εκτός από το περιβάλλον του Android studio για την υλοποίηση της εφαρμογής χρειάστηκε να χρησιμοποιηθεί το εργαλείο xampp που παρέχει δυνατότητες server και php\_my\_admin για τις βάσεις. Επίσης έχουν χρησιμοποιηθεί οι βιβλιοθήκες JSON και VOLLEY για την επικοινωνία όπως έχει αναφερθεί και στο κεφάλαιο τρία και αρχεία με κώδικα php και sql.

### 4.2.1 Δομή της εφαρμογής ProGlykeiou

Σε αυτό το σημείο θα παρουσιαστεί αναλυτικά, ξεκινώντας από το αρχικό παράθυρο, η ροή εκτέλεσης της εφαρμογής. Κάθε εφαρμογή Android, όπως αναφέρθηκε και πιο πάνω, αποτελείται από το γραφικό κομμάτι το οποίο είναι γραμμένο σε κώδικα xml και το αλγοριθμικό κομμάτι το οποίο είναι γραμμένο σε κώδικα Java. Ο προγραμματιστικός όρος για τον συνδυασμό αυτών των δύο ονομάζεται Activity, και στην ουσία πρόκειται για ένα παράθυρο που εμφανίζεται στην συσκευή του κινητού μας τηλεφώνου. Στο διάγραμμα ροής που ακολουθεί (εικόνα 10) φαίνεται αναλυτικά η ροή εκτέλεσης από ένα Activity στο επόμενο για όλη τη διάρκεια ζωής της εφαρμογής ξεκινώντας από το

MainActivity, το οποίο είναι το πρώτο Activity που εκτελείτε όταν ο χρήστης εκκινεί την εφαρμογή.

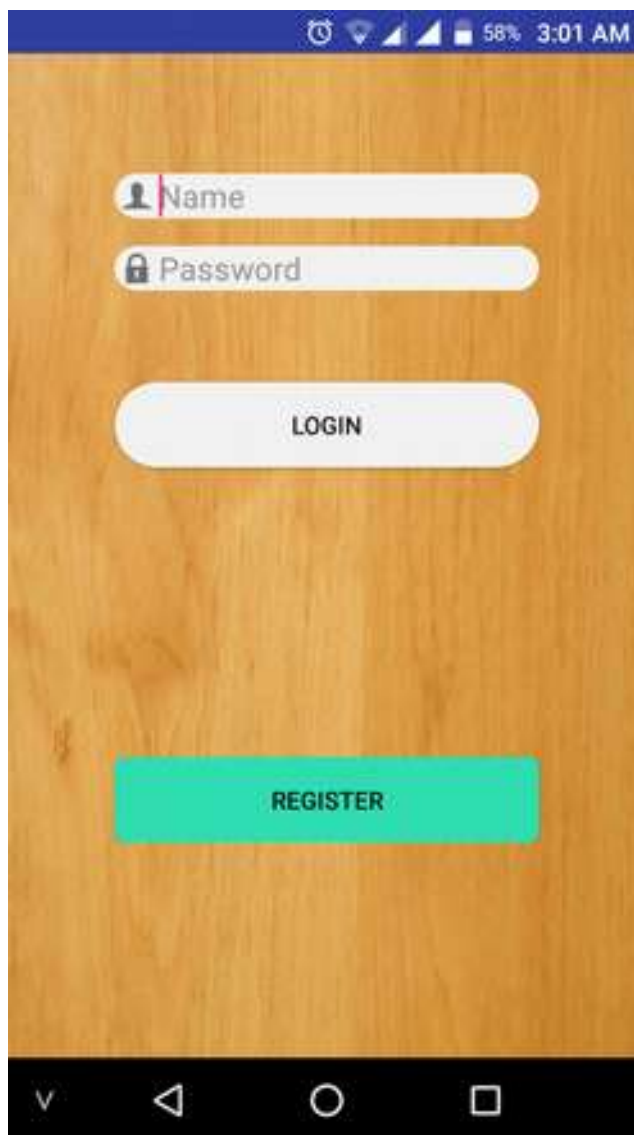


Εικόνα 11: Διάγραμμα ροής

## 4.2.2 Περιγραφή των βασικών παραθύρων(Activities)

### MainActivity

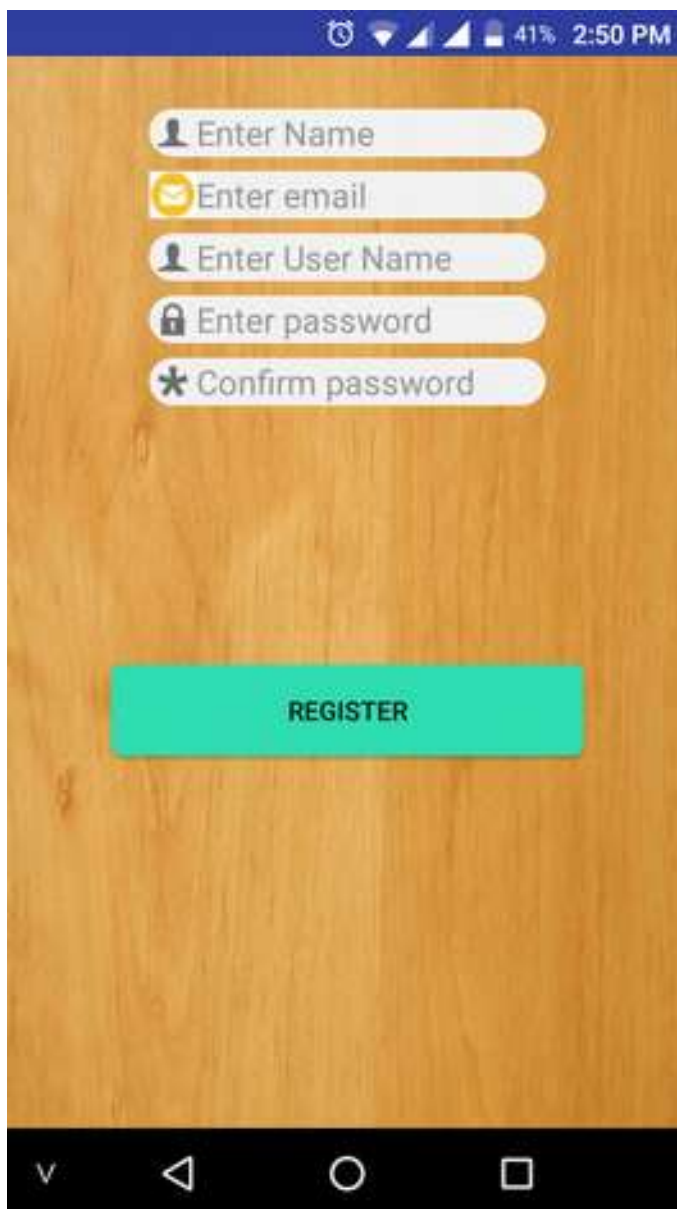
Είναι το πρώτο Activity που εκτελείτε κατά την έναρξη της εφαρμογής. Σε αυτό το παράθυρο ο χρήστης μπορεί να δώσει τα στοιχεία του λογαριασμού του για να συνδεθεί στην εφαρμογή. Όταν ξεκινάει η εφαρμογή ανοίγει μια καρτέλα για είσοδο(log in),αλλιώς μπορείς να κάνεις εγγραφή πατώντας το κουμπί register και σε στέλνει σε νέα καρτέλα. Η εφαρμογή συνδέεται σε μία βάση δεδομένων σε ένα υπολογιστή(server) που τρέχει το xamarin όπου υπάρχει το table με τους Users. Εάν ο χρήστης δώσει εσφαλμένα στοιχεία εμφανίζεται αντίστοιχο μήνυμα.



Εικόνα 12: MainActivity

## Register

Σε αυτό το Activity ο χρήστης μπορεί να κάνει εγγραφή(register) στην εφαρμογή δίνοντας τα στοιχεία του δηλαδή όνομα(name), διεύθυνση ηλεκτρονικού ταχυδρομίου (mail),όνομα χρήστη(user name),και κωδικό χρήστη(password).Δεν επιτρέπεται η κενή εγγραφή σε κάποιο πεδίο.Αφού γίνει σωστά η διαδικασία και ο χρήστης πατήσει το κουμπί register,επιστρέφει στην καρτέλα MainActivity όπου ο χρήστης μπορεί πλέον να κάνει είσοδο.



Εικόνα 13: Register activity

## LoginSuccess

Σε αυτό το Activity πηγαίνει ο χρήστης μετά την είσοδό του στην εφαρμογή. Αυτή η καρτέλα έχει ένα μήνυμα που καλωσορίζει τον χρήστη και δύο κουμπιά όπου ο χρήστης μπορεί να κάνει εκκίνηση της εφαρμογής ή να την τερματίσει.



Εικόνα 14: LoginSuccess activity

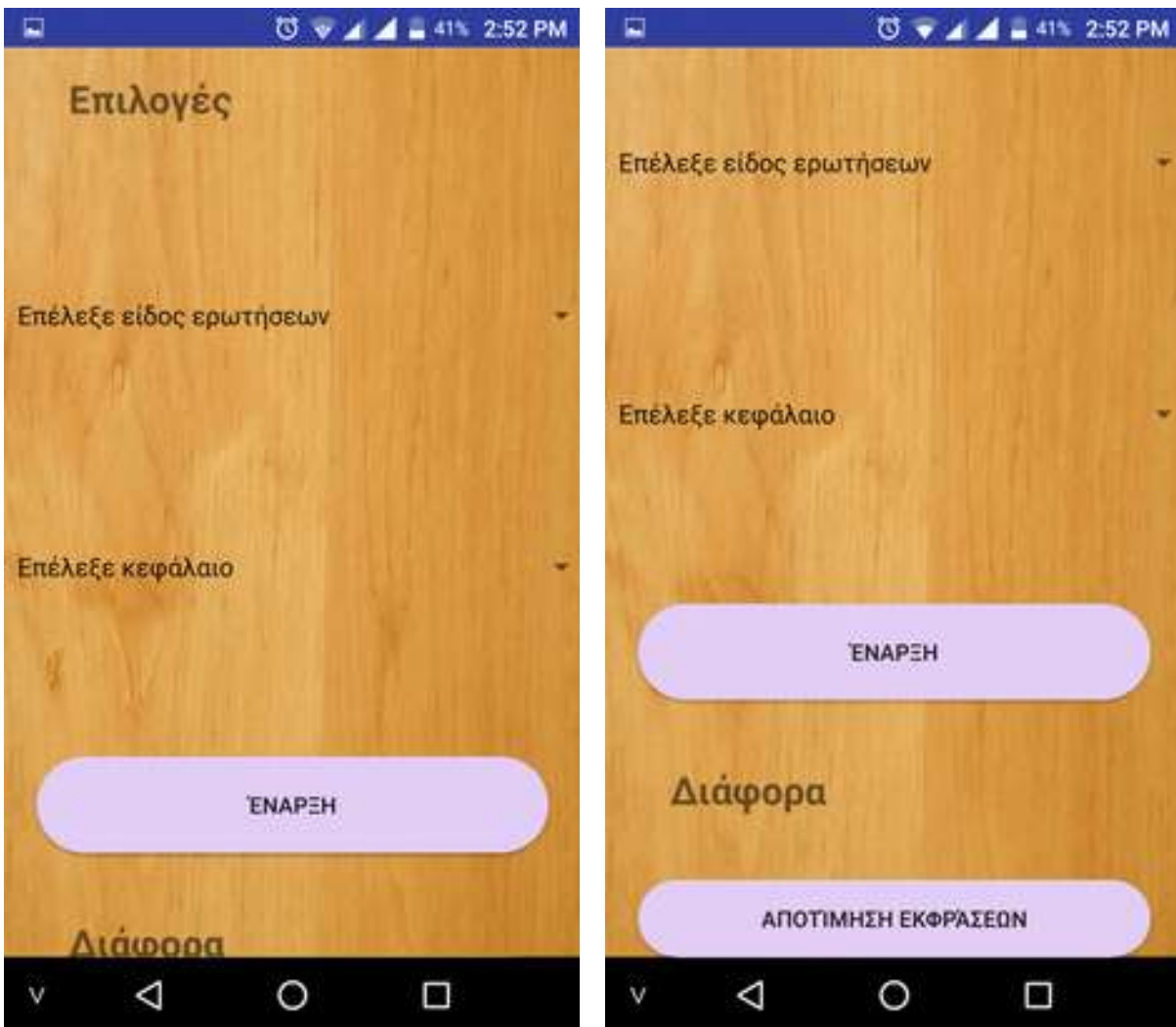
## Choose

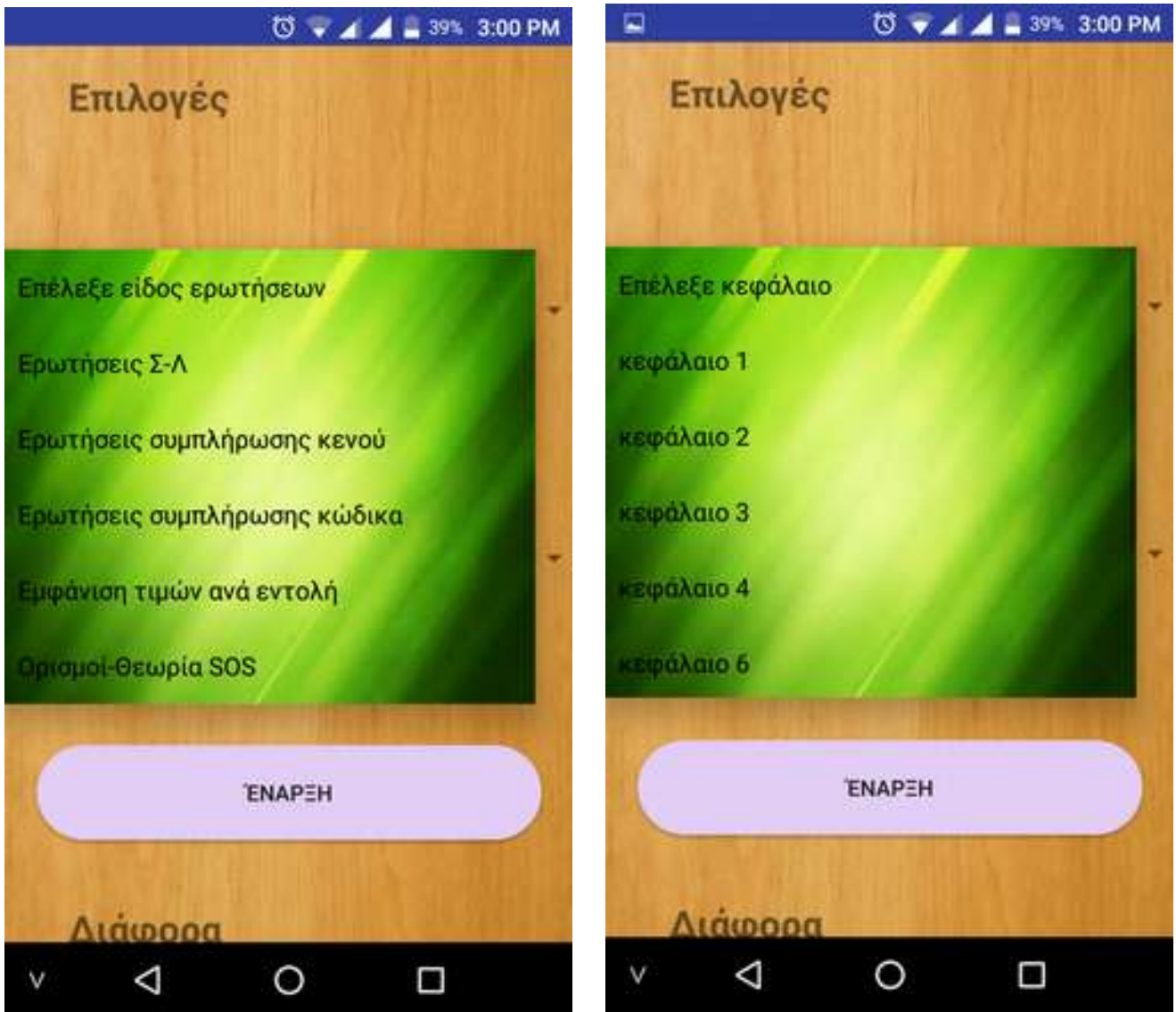
Σε αυτό το Activity ο χρήστης επιλέγει την κατηγορία των ερωτήσεων στην οποία θέλει να εξασκηθεί. Υπάρχουν δύο μπάρες επιλογών όπου ο χρήστης μπορεί να



επιλέξει ανάμεσα σε κάποιες επιλογές. Στην πρώτη μπάρα ο χρήστης μπορεί να επιλέξει το είδος των ερωτήσεων ενώ στην δεύτερη επιλέγει το κεφάλαιο με την

αντίστοιχη ύλη που υπάρχει στο βιβλίο του μαθήματος. Επίσης μπορεί να επιλέξει την καρτέλα αποτίμηση εκφράσεων. Αφού επιλέξει ο χρήστης κάποια επιλογή και πατηθεί το κουμπί έναρξη η εφαρμογή κατευθύνεται στην αντίστοιχη καρτέλα ανάλογα με το είδος.





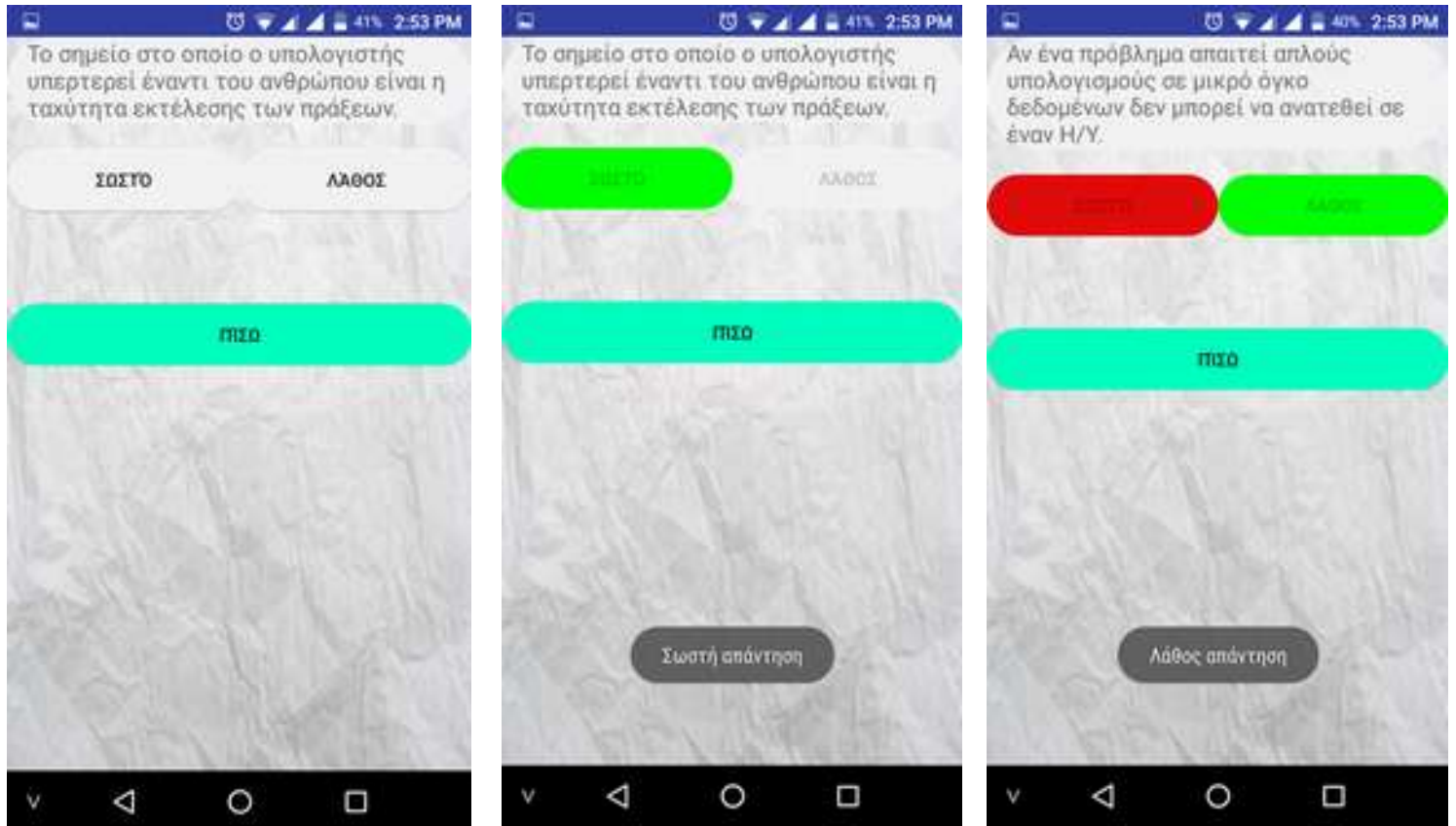
Εικόνα 15: Choose activity

## GenQuestion

Αυτό το Activity είναι η καρτέλα ερωτήσεων “Σωστο-Λαθος”. Η καρτέλα περιλαμβάνει:

- Ένα κείμενο(text) που δίνεται η ερώτηση στον χρήστη.
- Δύο κουμπια για την επιλογή Σωστό ή Λάθος.
- Ένα κουμπί “Πίσω” όπου το πρόγραμμα επιστρέφει στο LoginSuccess Activity.

Αφού ο χρήστης πατήσει ένα από τα 2 κουμπιά, Σωστό ή Λάθος, γίνεται το κουμπί της επιλογής του πορτοκαλί χρώμα και μετά από 1 δευτερόλεπτο γίνεται κόκκινο ή πράσινο αναλόγως την ορθότητα της απάντησης. Η επόμενη ερώτηση εμφανίζεται σε 2 δευτερόλεπτα αυτόματα και επιλέγεται τυχαία από την βάση.



Εικόνα16: GenQuestion activity

## ErwthseisSumpKenou

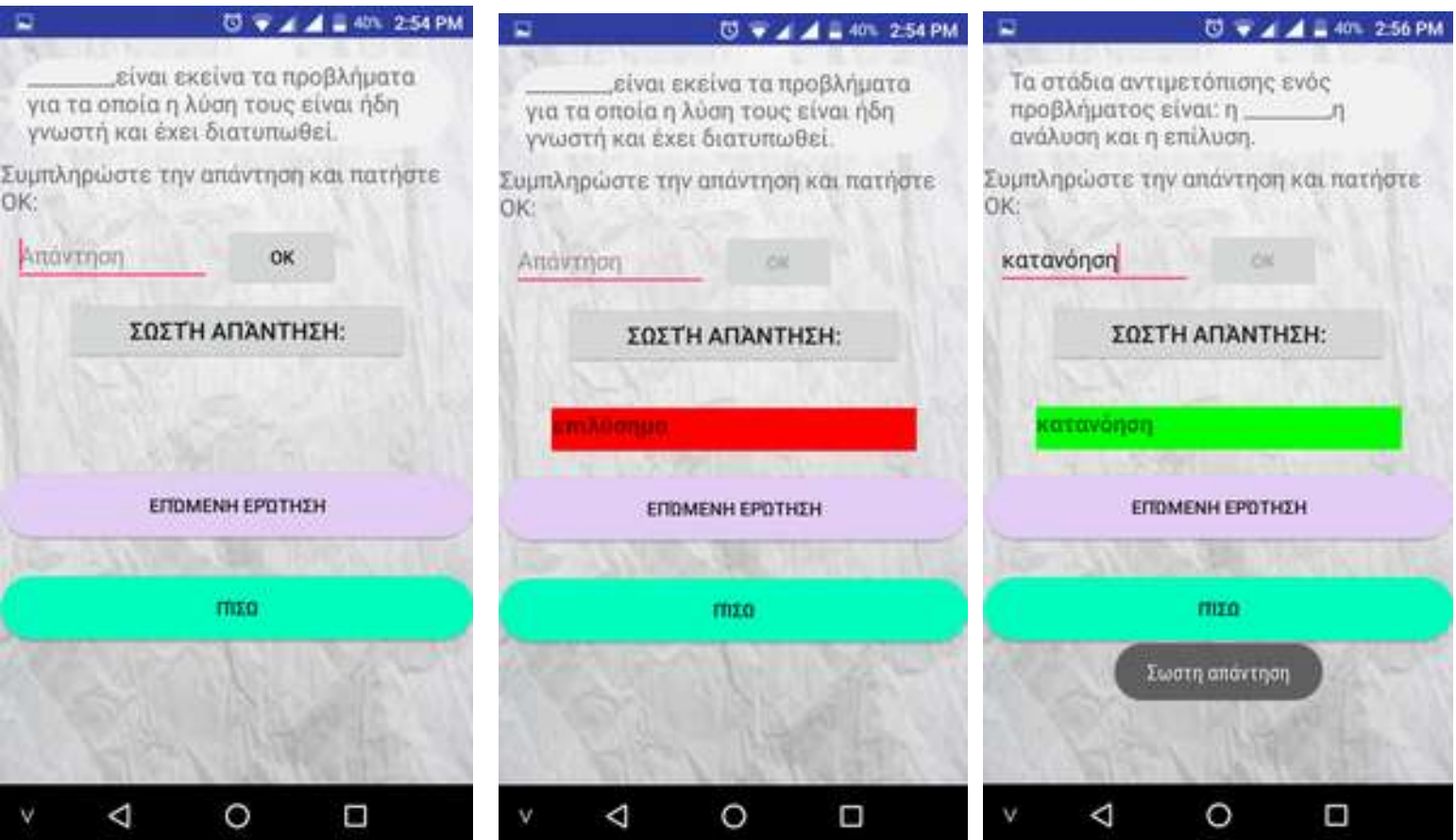
Αυτό το Activity είναι η καρτέλα ερωτήσεων συμπλήρωσης κενού. Η καρτέλα περιλαμβάνει:

- Ένα κείμενο(text) που δίνεται η ερώτηση στον χρήστη.
- Ένα πεδίο συμπλήρωσης κειμένου(text).
- Ένα κουμπί “OK”για να δώσει την απάντηση ο χρήστης.
- Ένα κουμπί για να εμφανιστεί η σωστή απάντηση στον χρήστη.
- Ένα κουμπί για συνέχεια στην επόμενη ερώτηση.
- Ένα κουμπί “Πίσω” όπου το πρόγραμμα επιστρέφει στο LoginSuccess Activity.

Ο χρήστης πρέπει να συμπληρώσει την κατάλληλη λέξη που λείπει στην πρόταση. Αφού συμπληρώσει την λέξη στο πεδίο, πρέπει να πατήσει το κουμπί OK για να ελέγξει την ορθότητα της απάντησης. Εάν η λέξη που συμπλήρωσε είναι η σωστή εμφανίζεται το μήνυμα “Σωστή απάντηση” και η λέξη εμφανίζεται πράσινη κάτω από το κουμπί ΣΩΣΤΗ ΑΠΑΝΤΗΣΗ. Εάν η λέξη είναι λάθος εμφανίζεται το μήνυμα “Λάθος απάντηση” και η λέξη εμφανίζεται κάτω από το κουμπί ΣΩΣΤΗ ΑΠΑΝΤΗΣΗ κόκκινη.

Αν ο χρήστης δεν γνωρίζει την απάντηση και δεν θέλει να συμπληρώσει τίποτα στο πεδίο, μπορεί να πατήσει απευθείας το κουμπί ΣΩΣΤΗ ΑΠΑΝΤΗΣΗ για να εμφανιστεί η λέξη που λείπει.

Το κουμπί “επόμενη ερώτηση” εμφανίζει άλλη ερώτηση τυχαία και το κουμπί “πίσω” επιστρέφει στην αρχική καρτέλα όπου υπάρχει η είσοδος και η έξοδος από την εφαρμογή.



Εικόνα16: ErwthseisSumpKenou activity

## ErwthseisSumpKwdika

Αυτό το Activity αποτελεί την καρτέλα ερωτήσεων συμπλήρωσης κώδικα. Η καρτέλα περιλαμβάνει:

- Δύο κείμενα(text) που βοηθάνε τον χρήστη στην κατανόηση της καρτέλας.
- Ένα κείμενο(text) για τον κώδικα και για την εκφώνηση της άσκησης.
- Ένα κουμπί “OK” για να δώσει την απάντηση ο χρήστης.
- Ένα κουμπί για να εμφανιστεί η σωστή απάντηση στον χρήστη.
- Ένα κουμπί για συνέχεια στην επόμενη ερώτηση.
- Ένα κουμπί “Πίσω” όπου το πρόγραμμα επιστρέφει στο LoginSuccess Activity.

Η λειτουργία της καρτέλας είναι η εξής:

Ο χρήστης πρέπει να συμπληρώσει την κατάλληλη εντολή που λείπει στον κώδικα που εμφανίζεται στην καρτέλα σύμφωνα με την εκφώνηση που εμφανίζετε. Αφού συμπληρώσει την εντολή στο πεδίο, πρέπει να πατήσει το κουμπί OK για να ελέγξει την ορθότητα της απάντησης. Εάν η εντολή που συμπλήρωσε είναι η σωστή εμφανίζεται το μήνυμα “Σωστή απάντηση” και η εντολή εμφανίζεται πράσινη κάτω από το κουμπί ΕΜΦΑΝΙΣΗ ΑΠΑΝΤΗΣΗΣ. Αν η εντολή είναι λάθος εμφανίζεται το μήνυμα “Λάθος απάντηση” και η λέξη εμφανίζεται κάτω από το κουμπί ΕΜΦΑΝΙΣΗ ΑΠΑΝΤΗΣΗΣ κόκκινη. Εάν ο χρήστης δεν γνωρίζει την απάντηση και δεν θέλει να συμπληρώσει τίποτα στο πεδίο, μπορεί να πατήσει απευθείας το κουμπί ΕΜΦΑΝΙΣΗ ΑΠΑΝΤΗΣΗΣ για να εμφανιστεί η εντολή που λείπει. Το κουμπί “επόμενη ερώτηση” εμφανίζει άλλη ερώτηση τυχαία και το κουμπί “πίσω” επιστρέφει στην αρχική καρτέλα όπου υπάρχει η είσοδος και η έξοδος από την εφαρμογή.



Εικόνα17: ErwthseisSumpKwdika activity

## filltable

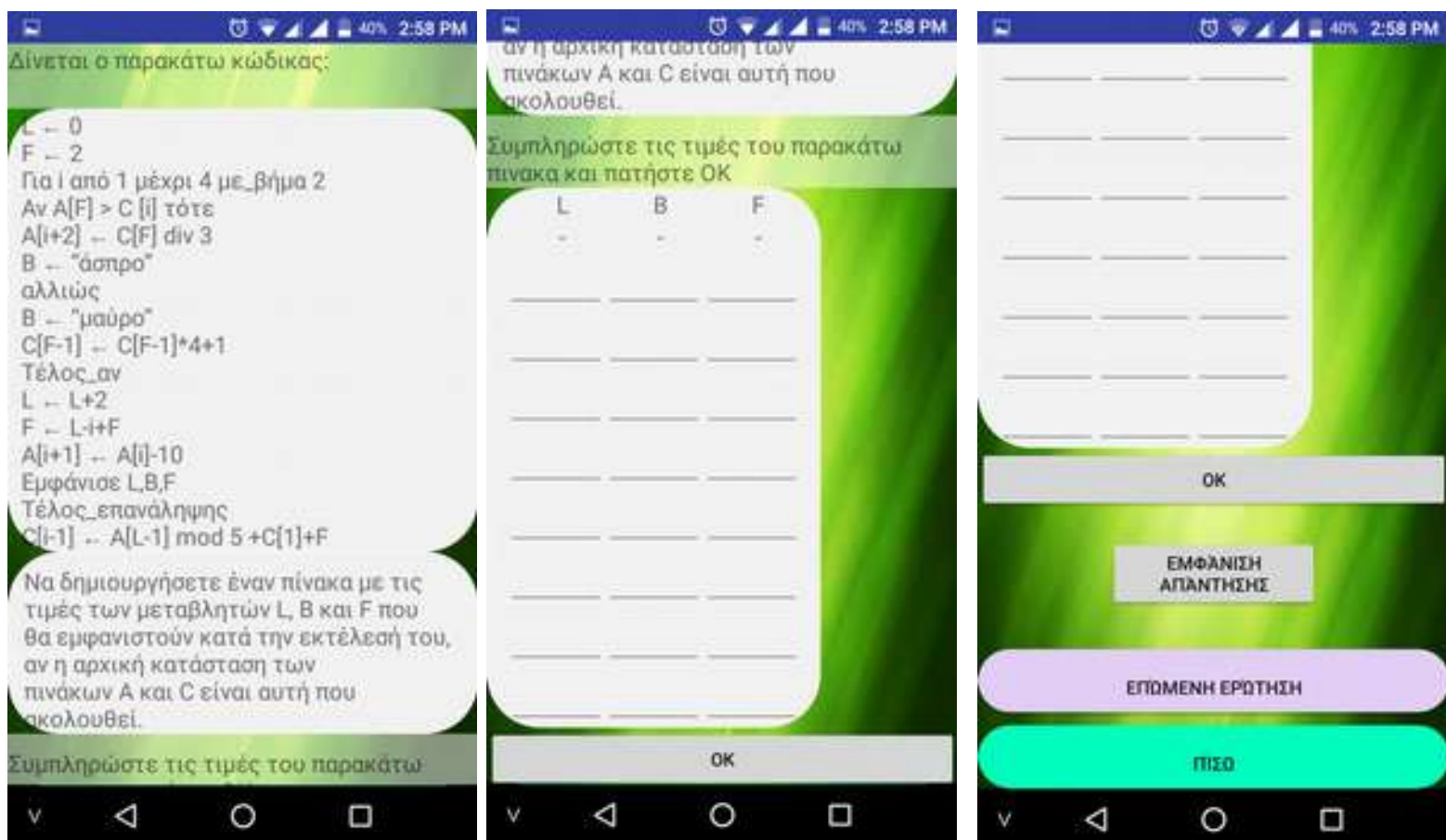
Αυτό το Activity αποτελεί την καρτέλα ερωτήσεων εμφάνιση τιμών ανά εντολή. Αυτή η καρτέλα περιλαμβάνει:

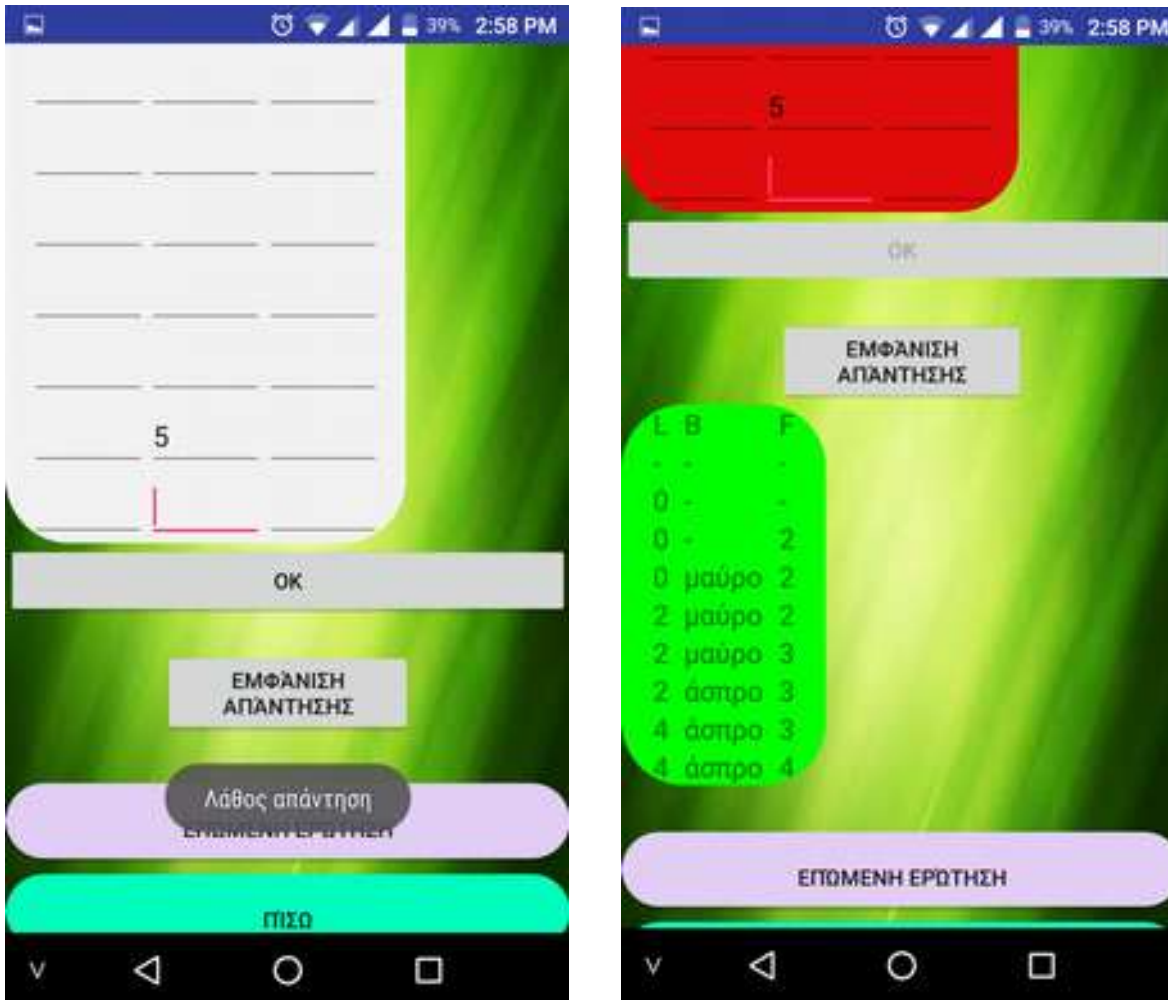
- Ένα κομμάτι κώδικα
- Εκφώνηση
- Πίνακα με πεδία για την απάντηση
- κουμπί OK
- κουμπί ΕΜΦΑΝΙΣΗ ΑΠΑΝΤΗΣΗΣ
- Πίνακας που εμφανίζει την σωστή απάντηση
- κουμπί επόμενη ερώτηση.
- Κουμπί που επιστρέφει στην αρχική καρτέλα.

Κάθε γραμμή του πίνακα αντιστοιχεί σε αλλαγή τιμής, σε μία από τις μεταβλητές του φαίνονται στον πίνακα καθώς τρέχει το πρόγραμμα από την αρχή μέχρι το τέλος και μόνο σε αυτές αυστηρά. Οπότε εάν αλλάξει η τιμή μιας μεταβλητής του

προγράμματος αλλά δεν ανήκει στον πίνακα δεν αναγράφεται και εάν ανήκει στο πίνακα ο χρήστης πρέπει να καταγράψει την αλλαγή σε μια νέα γραμμή του πίνακα. Εάν ο χρήστης συμπληρώσει τον πίνακα σωστά εμφανίζεται το μήνυμα “Σωστή απάντηση” και εμφανίζεται ο πίνακας με τις σωστές απαντήσεις κάτω από το κουμπί ΕΜΦΑΝΙΣΗ ΑΠΑΝΤΗΣΗΣ πράσινος. Εάν ο χρήστης συμπληρώσει λάθος την απάντηση εμφανίζεται το μήνυμα “Λάθος απάντηση”, αλλά δίνεται η δυνατότητα στον χρήστη να αλλάξει την απάντηση και να προσπαθήσει ξανά. Εάν ο χρήστης δεν γνωρίζει την απάντηση ή δεν μπορεί να την βρει, μπορεί να πατήσει το κουμπί ΕΜΦΑΝΙΣΗ ΑΠΑΝΤΗΣΗΣ και να εμφανιστεί η απάντηση.

Το κουμπί “επόμενη ερώτηση” εμφανίζει άλλη ερώτηση τυχαία και το κουμπί “πίσω” επιστρέφει στην αρχική καρτέλα όπου υπάρχει η είσοδος και η έξοδος από την εφαρμογή.





Εικόνα17: filltable activity

## theoria

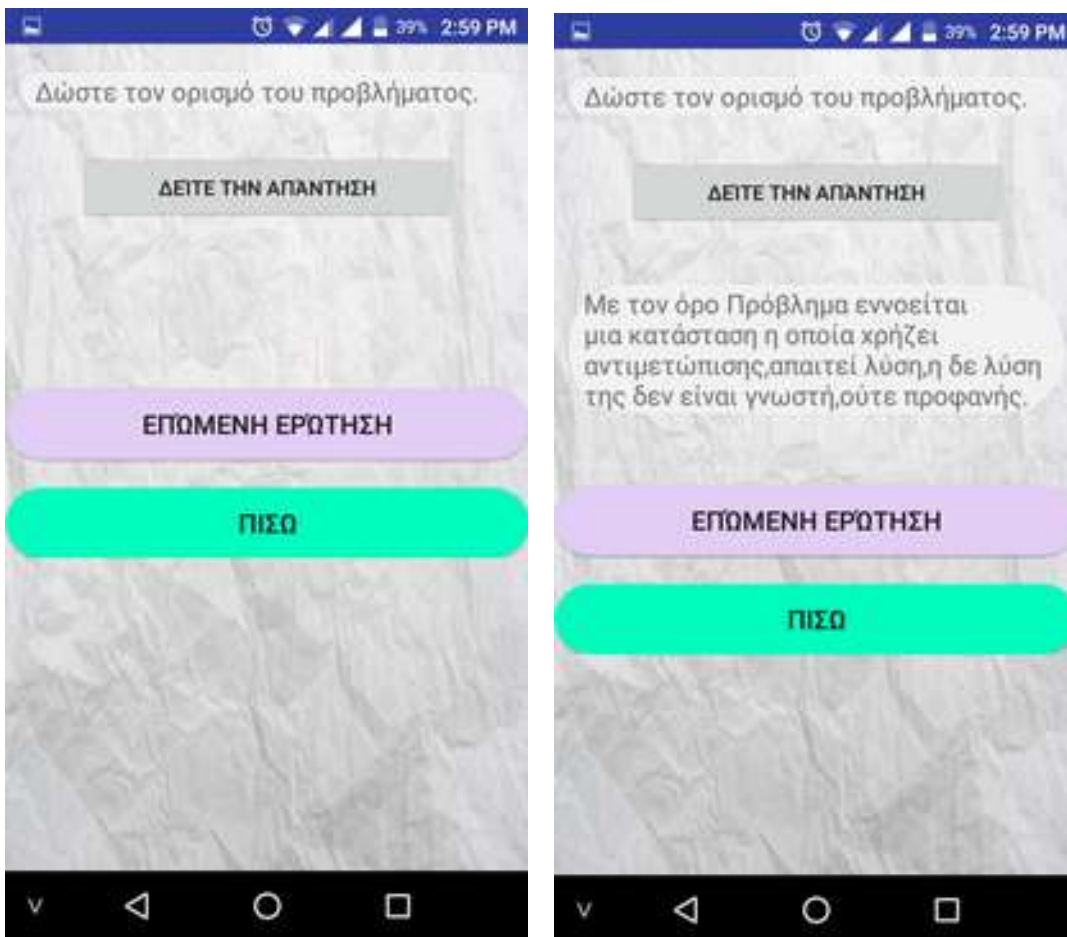
Αυτό το Activity αποτελεί την καρτέλα Θεωρία-SOS. Αυτή η καρτέλα περιλαμβάνει:

- Δυο κείμενα(text) ένα για την ερώτηση και ένα για την απάντηση.
- Ένα κουμπί, δείτε την απάντηση για να εμφανίζεται η απάντηση στον χρήστη.
- Ένα κουμπί για συνέχεια στην επόμενη ερώτηση.
- Ένα κουμπί “Πίσω” όπου το πρόγραμμα επιστρέφει στο LoginSuccess Activity.



Σε αυτήν την καρτέλα ο χρήστης δεν συμπληρώνει κάτι. Υπάρχει μόνο για εξάσκηση και για έναν έλεγχο του χρήστη εάν γνωρίζει την θεωρία και τους ορισμούς. Μπορεί να φανεί αρκετά χρήσιμη το τελευταίο διάστημα πριν την εξέταση.

Σε αυτήν την καρτέλα τα δεδομένα βρίσκονται σε εσωτερικά αρχεία της εφαρμογής και δεν “τραβιούνται” από βάση που υπάρχει στο internet. Αυτό έγινε για τον λόγο ότι το μέγεθος της θεωρίας παραμένει σταθερό και για λόγους εξάσκησης με αρχεία σε android.



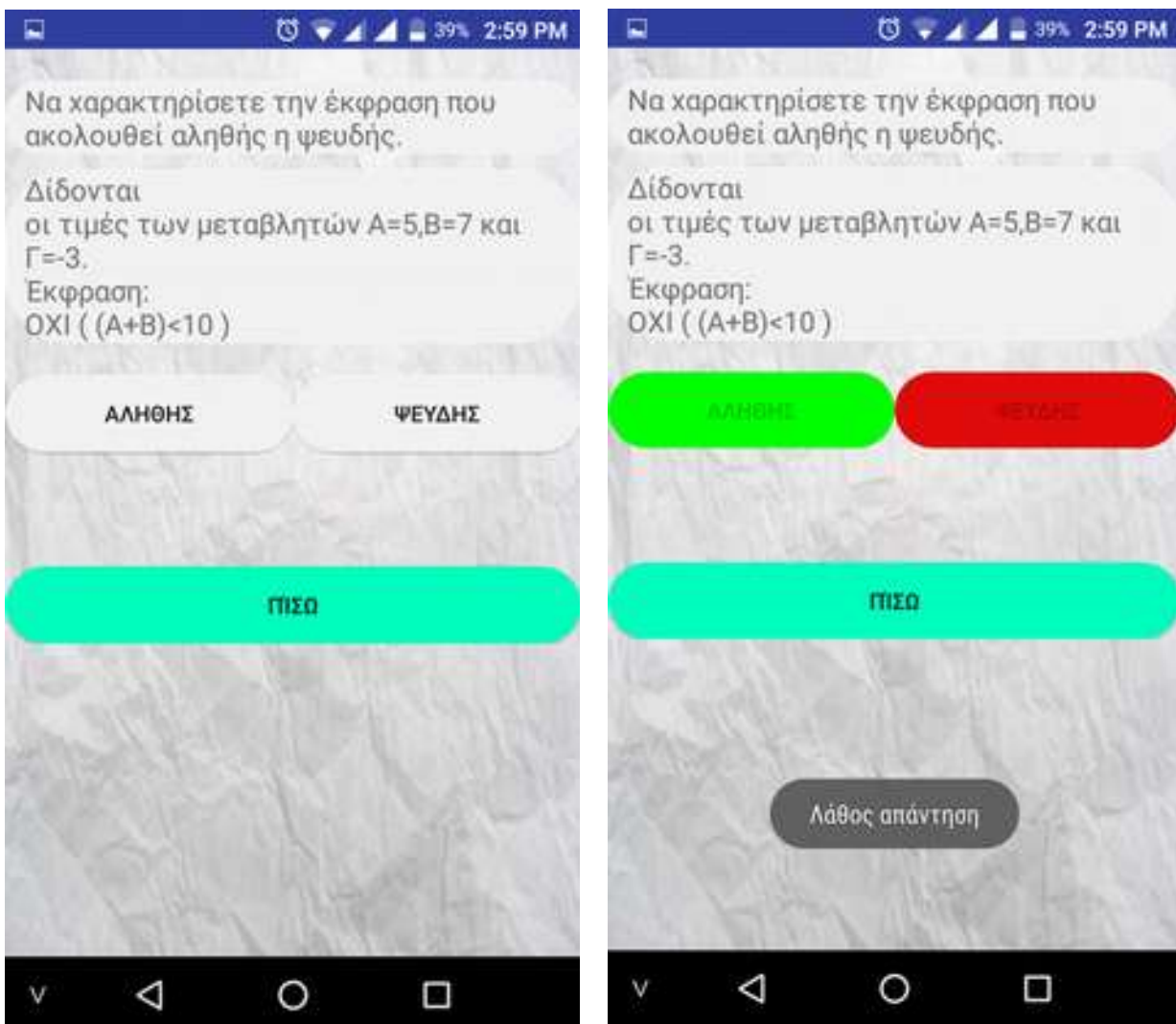
Εικόνα18: teoria activity

## logicExpr

Αυτό το Activity αποτελεί την καρτέλα αποτίμηση εκφράσεων. Αυτή η καρτέλα περιλαμβάνει:

- Δύο κείμενα, ένα για την εκφώνηση και ένα για τις λογικές εκφράσεις.
- Δύο κουμπιά για την επιλογή Αληθής ή Ψευδής.
- Ένα κουμπί “Πίσω” όπου το πρόγραμμα επιστρέφει στο LoginSuccess Activity.

Αυτή η καρτέλα δουλεύει όπως η καρτέλα “Σ-Λ”. Ο χρήστης όμως σε αυτήν την καρτέλα καλείται να αποτιμήσει μια λογική έκφραση εάν είναι αληθής ή ψευδής. Η εξάσκηση σε αυτό το είδος ερωτήσεων είναι αρκετά χρήσιμη για τον λόγο ότι μπορεί να μπει σαν θέμα θεωρίας στις πανελλήνιες (ΘΕΜΑ Α) αλλά και γιατί βοηθάει στην σωστή συγγραφή κώδικα.



Εικόνα19: logicExpr activity

## 4.3 Αλγόριθμοι και τεχνικές προγραμματισμού

### 4.3.1 Γραφικό περιβάλλον με χρήση της γλώσσας xml.

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

Η XML σχεδιάστηκε δίνοντας έμφαση στην απλότητα, τη γενικότητα και τη χρησιμότητα στο Διαδίκτυο . Είναι μία μορφοποίηση δεδομένων κειμένου, με ισχυρή υποστήριξη Unicode για όλες τις γλώσσες του κόσμου. Αν και η σχεδίαση της XML εστιάζει στα κείμενα, χρησιμοποιείται ευρέως για την αναπαράσταση αυθαίρετων δομών δεδομένων, που προκύπτουν για παράδειγμα στις υπηρεσίες ιστού.

Όλα τα layout από όλες τις καρτέλες έχουν γίνει στην γλώσσα xml με την βοήθεια του Android studio. Ένα παράδειγμα xml είναι η καρτέλα choose με layout το αρχείο activity\_choose.xml.

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

    android:layout_width="match_parent"
    android:background="@drawable/rsz_lightwoodtexture2"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:fillViewport="false">
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="41dp"
        android:layout_marginTop="16dp"
        android:text="@string/Epilogos"
        android:layout_marginLeft="40dp"
        android:layout_marginRight="40dp"
        android:textSize="24sp"
        android:textStyle="bold" />
    <Spinner
        android:id="@+id/eidos"
        android:layout_width="match_parent"
        android:layout_height="77dp"
        android:layout_marginTop="72dp" />
    <Spinner
        android:id="@+id/kef"
        android:layout_width="match_parent"
        android:layout_height="77dp"
        android:layout_marginTop="80dp" />
    <Button
        android:id="@+id/Start_b"
        android:layout_width="match_parent"
```



```

        android:layout_height="60dp"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"

android:layout_marginTop="80dp"

        android:background="@drawable/next_bn"
        android:text="@string/start" />
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/diaf"
        android:textSize="24sp"
        android:textStyle="bold"
        android:text="@string/diaf"
        android:layout_marginTop="40dp"
        android:layout_marginRight="40dp"
        android:layout_marginLeft="40dp"/>
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="40dp"
        android:text="Αποτίμηση εκφράσεων"
        android:background="@drawable/next_bn"
        android:layout_marginRight="20dp"
        android:id="@+id/apotimhsh"/>
</LinearLayout>
</ScrollView>

```

Μπορεί να παρατηρήσει κανείς ότι η xml έχει δενδρική δομή δηλαδή το αντικείμενο scrollView έχει παιδί το LinearLayout και το αντικείμενο LinearLayout έχει τα Spinners,Buttons,TextViews.Επίσης κάθε αντικείμενο έχει ένα ID (android:id="xxx" ) το οποίο πρέπει να είναι και μοναδικό. Αυτό το ID χρησιμοποιείται από τον κώδικα Java,εάν χρειάζεται να γίνει έλεγχος η αλλαγή στα δεδομένα που περιέχει.Ένα παράδειγμα σε αυτήν καρτέλα είναι το κουμπί “Εναρξη” το οποίο έχει ID:Start\_b. Στον κώδικα Java της καρτέλα choose.java όπου αντιστοιχεί το layout υπάρχει ο κώδικας:

```

Button button_st = (Button) findViewById(R.id.Start_b);
button_st.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        //Οποιαδήποτε ενέργεια
    }
}

```

Έτσι δημιουργείται ένα αντικείμενο τύπου Button που αντιστοιχεί στο κουμπί Start\_b και μπορείς να καλέσεις οποιαδήποτε μέθοδο ανήκει σε αυτήν την κλάση. Μια μέθοδος είναι η `setOnClickListener` η οποία όταν ο χρήστης πατήσει το κουμπί μπορείς να εκτελέσεις μια οποιαδήποτε ενέργεια.

### 4.3.2 Επικοινωνία με την βάση και επεξεργασία δεδομένων

Όπως έχει αναφερθεί και στο κεφάλαιο 3, η σύνδεση με την βάση γίνεται με την χρήση της βιβλιοθήκης Volley. Για να εξηγηθεί καλύτερα η διαδικασία θα χρησιμοποιηθεί κώδικας από το Intent register

```
StringRequest stringRequest = new StringRequest(Request.Method.POST,
reg_url,
    new Response.Listener<String>() {
        @Override
        public void onResponse(String response) {
            try {
                JSONArray jsonArray = new JSONArray(response);
                JSONObject jsonObject = jsonArray.getJSONObject(0);
                String code = jsonObject.getString("code");
                String message = jsonObject.getString("message");
                builder.setTitle("Server response...");
                builder.setMessage(message);
                displayAlert(code);
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }
    }, new Response.ErrorListener() {
        @Override
        public void onErrorResponse(VolleyError error) {
            Toast.makeText(getApplicationContext(), "error VOLLEY
"+error.getMessage(), Toast.LENGTH_LONG).show();
        }
    }) {
        @Override
        protected Map<String,String> getParams() throws AuthFailureError
        {
            Map<String,String> params = new HashMap<String,String>();
            params.put("name", name);
            params.put("email", email);
            params.put("user_name", username);
            params.put("password", password);
            return params;
        }
    };
MySingleton.getInstance(Register.this).addToRequestQueue(stringRequest
);
```

Με την μέθοδο `StringRequest` στέλνουμε και λαμβάνουμε τα δεδομένα. Στην `StringRequest` έχουμε δύο `Response Listeners`. Ο πρώτος (`Response.Listener`) χρησιμοποιείται όταν η διαδικασία σύνδεσης είναι ομαλή και ο δεύτερος (`Response.ErrorListener()`) όταν συμβαίνει κάποιο σφάλμα στην διαδικασία. Στην περίπτωση του πρώτου επειδή η απάντηση είναι κωδικοποιημένη σε μορφή JSON αλλά επιστρέφεται από τον server ως αλφαριθμητικό(`string`) χρησιμοποιείται ο κώδικας:

```
JSONArray jsonArray = new JSONArray(response);
JSONObject jsonObject = jsonArray.getJSONObject(0);
```

Ύστερα για να “πάρουμε” τα αντικείμενα που υπάρχουν στην μορφή `Json` χρησιμοποιούμε την συνάρτηση

```
String code = jsonObject.getString("xxx");
```

Στην περίπτωση του `Response.ErrorListener`, εμφανίζεται ένα αντίστοιχο μήνυμα του σφάλματος.

Τα δεδομένα στέλνονται στην βάση με την μέθοδο `POST` και ο κώδικας που χρησιμοποιείται για αυτό βρίσκεται μετά τα ορίσματα της κλήσης του αντικειμένου `StringRequest`. Συγκεκριμένα στο κομμάτι κώδικα:

```
{
    @Override
    protected Map<String,String> getParams() throws AuthFailureError {
        Map<String,String> params = new HashMap<String,String>();
        params.put("name", name);
        params.put("email", email);
        params.put("user_name", username);
        params.put("password", password);
        return params;
    }
}
```

Με την χρήση της μεθόδου `getParams()` και `params.put("xxx", xxx);`

### 4.3.3 Οι βασικοί αλγόριθμοι της εφαρμογής

Στην εφαρμογή εκτελούνται μια σειρά από αλγορίθμους. Οι πιο βασικοί από αυτούς είναι οι εξής: ο αλγόριθμος επιλογής κεφαλαίου και είδος ερωτήσεων από τον χρήστη και η κατεύθυνση στην αντίστοιχη καρτέλα, ο αλγόριθμος υπολογισμού σωστής απάντησης ερωτήσεων Σωστό-Λάθος, ο αλγόριθμος υπολογισμού σωστής απάντησης ερωτημάτων συμπλήρωσης κώδικα, ο αλγόριθμος δημιουργίας του πίνακα για της απάντησης του χρήστη στην καρτέλα “ΕΜΦΑΝΙΣΗ ΤΙΜΩΝ ΑΝΑ ΕΝΤΟΛΗ” και ο έλεγχος της ορθότητας της απάντησης και ο αλγόριθμος ανάγνωσης αρχείων για την καρτέλα “ΟΡΙΣΜΟΙ-ΘΕΩΡΙΑ SOS”.

#### Αλγόριθμος επιλογής κεφαλαίου και είδος ερωτήσεων από τον χρήστη και η κατεύθυνση στην αντίστοιχη καρτέλα

Ο χρήστης επιλέγει το είδος και το κεφάλαιο που θέλει να εξασκηθεί από μια μπάρα επιλογής. Αυτό γίνεται με την βοήθεια ενός αντικειμένου που ονομάζεται spinner. Η java μπορεί να λάβει την επιλογή του χρήστη με την χρήση μίας μεθόδου

```
spinner.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener()
```

Και της μεθόδου

```
public void onItemClick(AdapterView<?> parent, View view, int
pos, long id).
```

Κάθε φορά που ο χρήστης κάνει μια επιλογή από μια λίστα μπορεί να επιστραφεί η επιλογή του με την βοήθεια της παραμέτρου int pos και μπορεί να αποθηκευτεί σε ένα αλφαριθμητικό π.χ στο αλφαριθμητικό με το όνομα msg:

```
msg = (String) parent.getItemAtPosition(pos);
```

Παρακάτω φαίνεται ο κώδικας που χρησιμοποιείται για την επιλογή του είδους ερωτήσεων.

```

spinner_eidos.setOnItemSelectedListener(new
AdapterView.OnItemSelectedListener() {
    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int
position, long id) {
        msg1 = (String) parent.getItemAtPosition(position);
        checkeidos =1;
        if ( parent.getItemAtPosition(position).equals("Ερωτήσεις Σ-
Λ") ) {
            Toast.makeText (getBaseContext(), "Επιλέχθηκε Ερωτήσεις
Σ-Λ ", Toast.LENGTH_SHORT).show();
        }
        else if (parent.getItemAtPosition(position).equals("Ερωτήσεις
πολλαπλής επιλογής")) {
            Toast.makeText (getBaseContext(), "Επιλέχθηκε Ερωτήσεις
πολλαπλής επιλογής", Toast.LENGTH_SHORT).show();
        }
        else if (parent.getItemAtPosition(position).equals("Ερωτήσεις
συμπλήρωσης κενού")) {
            Toast.makeText (getBaseContext(), "Επιλέχθηκε Ερωτήσεις
συμπλήρωσης κενού", Toast.LENGTH_SHORT).show();
        }
        else if (parent.getItemAtPosition(position).equals("Ερωτήσεις
συμπλήρωσης κώδικα")) {
            Toast.makeText (getBaseContext(), "Επιλέχθηκε Ερωτήσεις
συμπλήρωσης κώδικα", Toast.LENGTH_SHORT).show();
        }
        else if
(parent.getItemAtPosition(position).equals("Δημιουργία διαγραμμάτων
ροής")) {
            Toast.makeText (getBaseContext(), "Επιλέχθηκε Δημιουργία
διαγραμμάτων ροής", Toast.LENGTH_SHORT).show();
        }
        else if
(parent.getItemAtPosition(position).equals("Εμφανίσεις τιμών ανά
εντολή")) {
            Toast.makeText (getBaseContext(), "Επιλέχθηκε Εμφανίσεις
τιμών ανά εντολή", Toast.LENGTH_SHORT).show();
        }
        else if (parent.getItemAtPosition(position).equals("Ορισμοί-
Θεωρία SOS")) {
            Toast.makeText (getBaseContext(), "Επιλέχθηκε Ορισμοί-
Θεωρία SOS", Toast.LENGTH_SHORT).show();
        }
        else {
            checkeidos=0;
        }
    }
}
@Override
public void onNothingSelected(AdapterView<?> parent) {

```



```
57     }  
});
```

Εάν ο χρήστης δεν κάνει κάποια επιλογή υπάρχει μια μεταβλητή που παίρνει τιμή “0”(checkidos=0;) όπου χρησιμοποιείται παρακάτω στο πρόγραμμα.

Εάν ύστερα ο χρήστης πατήσει το κουμπί “έναρξη” υπάρχει κώδικας που κατευθύνει το πρόγραμμα στην σωστή καρτέλα(Intent).

```
button.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        if ( checkidos==0 || checkKefalaio==0 ){  
            Toast.makeText (getBaseContext(), "Επιλέξτε κεφάλαιο και  
είδος απο την μπάρα επιλογής", Toast.LENGTH_LONG).show();  
        }  
        else {  
            Bundle bundle = new Bundle();  
            bundle.putString("message", msg);  
            bundle.putString("eidos", msg1);  
            if ( msg1.equals("Ερωτήσεις Σ-Λ") ) {  
                Intent intent = new Intent(Choose.this,  
GenQuestion.class);  
                intent.putExtras(bundle);  
                startActivity(intent);  
            }  
            else if (msg1.equals("Ερωτήσεις συμπλήρωσης κενού")){  
                Intent intent_smp_kenou = new Intent(Choose.this,  
ErwthseisSumpKenou.class);  
                intent_smp_kenou.putExtras(bundle);  
                startActivity(intent_smp_kenou);  
            }  
            else if (msg1.equals("Δημιουργία διαγραμμάτων ροής")){  
                Intent intent_smp_kenou = new Intent(Choose.this,  
Diagram.class);  
                intent_smp_kenou.putExtras(bundle);  
                startActivity(intent_smp_kenou);  
            }  
            else if (msg1.equals("Εμφανίσεις τιμών ανά εντολή")){  
                if (msg.equals("κεφάλαιο 1") || msg.equals("κεφάλαιο  
4") ||msg.equals("κεφάλαιο 6")) {  
                    Toast.makeText (getBaseContext(), "Δεν υπάρχουν  
"+msg1 +" σε αυτό το κεφάλαιο", Toast.LENGTH_LONG).show();  
                }  
            }  
        }  
    }  
});
```

```

        else {
            Intent intent_smp_kenou = new Intent(Choose.this,
fillTable.class);
            intent_smp_kenou.putExtras(bundle);
58             startActivity(intent_smp_kenou);
        }
    }
    else if (msg1.equals("Ορισμοί-Θεωρία SOS")){
        Intent intent_smp_kenou = new Intent(Choose.this,
theoria.class);
        intent_smp_kenou.putExtras(bundle);
        startActivity(intent_smp_kenou);
    }
    else {
        if (msg.equals("κεφάλαιο 1") || msg.equals("κεφάλαιο
4") || msg.equals("κεφάλαιο 6")) {
            Toast.makeText (getBaseContext(), "Δεν υπάρχουν
"+msg1 +" σε αυτό το κεφάλαιο", Toast.LENGTH_LONG).show();
        }
        else {
            Intent intent_smp_kenou = new Intent(Choose.this,
ErwthseisSumpKwdika.class);
            intent_smp_kenou.putExtras(bundle);
            startActivity(intent_smp_kenou);
        }
    }
}
});

```

## Αλγόριθμος υπολογισμού σωστής απάντησης ερωτήσεων Σωστό-Λάθος

```

False.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        False.setEnabled(false);
        False.setBackgroundResource(R.drawable.rounded_orange);
        True.setEnabled(false);
        randomNumber= rn.nextInt(max) + 1;
        Runnable clickButton = new Runnable() {
            @Override
            public void run() {
                // whatever you would like to implement when or after
clicking button
                checkQ(0);
            }
        };
        clickButton.run();
    }
});

```

```

        Runnable clickButton1 = new Runnable() {
            @Override
            public void run() {
59         True.setBackgroundResource(R.drawable.rounded_white);
False.setBackgroundResource(R.drawable.rounded_white);
                StringRequest jsonObjectRequest = new
StringRequest(Request.Method.POST, url,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String
response) {
                        try {
                            True.setEnabled(true);
                            False.setEnabled(true);
                            JSONObject jsonObject = new
JSONObject(response); //jsonArray.getJSONObject(0);
                            String Quest1 =
jsonObject.getString("Erwthsh");
                            Quest.setText(Quest1);
                            answer =
jsonObject.getString("Apanthsh");
                        } catch (JSONException e) {
                            e.printStackTrace();
                        }
                    }, new Response.ErrorListener() {
                        @Override
                        public void onErrorResponse(VolleyError
error) {
                            Toast.makeText(GenQuestion.this, "Error",
Toast.LENGTH_LONG).show();
                            error.printStackTrace();
                        }
                    }) {
                        @Override
                        protected Map<String, String> getParams()
throws AuthFailureError {
                            Map<String, String> params = new
HashMap<String, String>();
                            params.put("db_name", db_name);
                            params.put("ae_num",String.valueOf(randomNumber));
                            params.put("tableName", table_name);
                            return params;
                        }
                    };

                MySingleton.getInstance(GenQuestion.this).addToRequestque(jsonObject
Request);
            }
        };

```

```

        False.postDelayed(clickButton1, 2000);
    }
};
False.postDelayed(clickButton, 2000);
60 }
});

```

```

public void checkQ (int i) {
    if (answer.equals("1")){
        True.setBackgroundResource(R.drawable.rounded_green);
        if (i==0){
            False.setBackgroundResource(R.drawable.rounded_red);
            Toast.makeText (getBaseContext(), "Λάθος απάντηση",
Toast.LENGTH_SHORT).show();
        }
        else{
            Toast.makeText (getBaseContext(), "Σωστή απάντηση",
Toast.LENGTH_SHORT).show();
        }
    }
    else{
        False.setBackgroundResource(R.drawable.rounded_green);
        if(i==1){
            True.setBackgroundResource(R.drawable.rounded_red);
            Toast.makeText (getBaseContext(), "Λάθος απάντηση",
Toast.LENGTH_SHORT).show();
        }
        else{
            Toast.makeText (getBaseContext(), "Σωστή απάντηση",
Toast.LENGTH_SHORT).show();
        }
    }
}
}

```

Ο παραπάνω κώδικας χρησιμοποιείται όταν ο χρήστης πατήσει το κουμπί “Λάθος”. Τα κουμπιά Σωστό και λάθος απενεργοποιούνται και “παγώνουν” για δύο δευτερόλεπτα και το κουμπί Λάθος γίνεται πορτοκαλί. Ύστερα καλείται η συνάρτηση checkQ(int i) με όρισμα “0” που δηλώνει ότι προέρχεται από επιλογή του κουμπιού “Λαθος”. Η συνάρτηση checkQ ελέγχει την ορθότητα της απάντησης του χρήστη και εάν ο χρήστης απάντησε σωστά η επιλογή του γίνεται πράσινη, εάν απάντησε λάθος η επιλογή του γίνεται κόκκινη. Επίσης εμφανίζεται κατάλληλο μήνυμα για την ορθότητα της απάντησης. Μετά άλλα δύο δευτερόλεπτα το πρόγραμμα προχωράει στην επόμενη ερώτηση.

Ο ίδιος κώδικας χρησιμοποιείται και για την επιλογή του κουμπιού Σωστό με την διαφορά ότι η συνάρτηση `checkQ` καλείται με όρισμα “1” που δηλώνει ότι ο χρήστης επέλεξε για απάντηση “Σωστό”.

### **Αλγόριθμος υπολογισμού σωστής απάντησης ερωτημάτων συμπλήρωσης κώδικα.**

Όταν ο χρήστης συμπληρώσει την γραμμή κώδικα που λείπει και πατήσει OK, γίνεται έλεγχος για την ορθότητα της απάντησης. Η απάντηση είναι σωστή και στην περίπτωση που ο χρήστης χρησιμοποιήσει κεφαλαία ή μικρά γράμματα. Επίσης τα κενά διαστήματα περιλείπονται και δεν προσμετρούνται. Η εντολή εκχώρησης τιμής μπορεί να γραφτεί από το χρήστη και με το σύμβολο “#” για διευκόλυνση του. Εάν η απάντηση δεν έχει ίδια τιμή με αυτήν που υπάρχει στην βάση ως σωστή απάντηση, η απάντηση του χρήστη πάνεται ως λάθος και αυτό καταγράφεται σε μια μεταβλητή με όνομα “ch”. Αφού γίνει ο έλεγχος εμφανίζεται κατάλληλο μήνυμα για την ορθότητα της απάντησης.

```

ok_bn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String ch_an = UserAnswer.getText().toString().toLowerCase();
        String ans = answer.toLowerCase();
        if (ch_an.equals("")) {
            Toast.makeText(ErwthseisSumpKwdika.this, "Δωστε
απάντηση", Toast.LENGTH_LONG).show();
        } else {
            ok_bn.setEnabled(false);
            AnswerField.setText(answer);
            int index = ch_an.indexOf("<-");
            if (index > 0) {
                ch_an = ch_an.replace("<-", "# ");
            }
            int index2 = ans.indexOf("<-");
            if (index2 > 0) {
                ans = ans.replace("<-", "# ");
            }
            ch_an = ch_an.replaceAll("\\s+", "");
            ans = ans.replaceAll("\\s+", "");
            boolean ch;
            if (ch_an.equals(ans)) {
                ch = true;
            } else {
                ch = false;
            }
            if (ch) {
                Toast.makeText(ErwthseisSumpKwdika.this, "Σωστη
απάντηση", Toast.LENGTH_SHORT).show();
                AnswerField.setBackgroundColor(0xFF00FF00);
            } else {
                Toast.makeText(ErwthseisSumpKwdika.this, "Λάθος
απάντηση", Toast.LENGTH_SHORT).show();
                AnswerField.setBackgroundColor(0xFFFF0000);
            }
        }
    }
});

```

## Αλγόριθμος δημιουργίας του πίνακα για της απάντησης του χρήστη στην καρτέλα “ΕΜΦΑΝΙΣΗ ΤΙΜΩΝ ΑΝΑ ΕΝΤΟΛΗ” και ο έλεγχος της ορθότητας της απάντησης

Η δημιουργία του πίνακα γίνεται δυναμικά και το μέγεθος του πίνακα είναι ανάλογο με τις μεταβλητές του ελέγχει για αλλαγές τιμής ο χρήστης(δύο έως τέσσερεις) και με το πλήθος των γραμμών της απάντησης. Έτσι έχουμε τον παρακάτω κώδικα:

```

int numOfValues= Integer.parseInt(numOfval);
String [] valuesFirstChar = new String[4 ];
valuesFirstChar[0] = (String) value_1.substring(0,1);
valuesFirstChar[1]= (String) value_2.substring(0,1);
valuesFirstChar[2] = (String) value_3.substring(0,1);
valuesFirstChar[3]= (String) value_4.substring(0,1);
String[] variable_names = new String[4];
StringTokenizer [] strtokens = new StringTokenizer[4];
strtokens[0] = new StringTokenizer(value_1);
strtokens[1] = new StringTokenizer(value_2);
strtokens[2] = new StringTokenizer(value_3);
strtokens[3] = new StringTokenizer(value_4);
variable_names[0]=strtokens[0].nextToken();
variable_names[1]=strtokens[1].nextToken();
variable_names[2]=strtokens[2].nextToken();
variable_names[3]=strtokens[3].nextToken();
TextView[] textViews = new TextView[numOfValues*2];
for (i=0; i<numOfValues; i++) {
    textViews[i] = new TextView(fillTable.this);
    textViews[i+numOfValues] = new TextView(fillTable.this);
    textViews[i].setText( variable_names[i] );
    variable_names[i]=strtokens[i].nextToken();
    textViews[i+numOfValues].setText(variable_names[i] );
    textViews[i].setWidth(200);
    textViews[i+numOfValues].setWidth(200);
    textViews[i].setTextSize(18);
    textViews[i+numOfValues].setTextSize(18);
    textViews[i].setGravity(Gravity.CENTER_HORIZONTAL);
    textViews[i+numOfValues].setGravity(Gravity.CENTER_HORIZONTAL);
    textViews[i].setPadding(20,0,0,0);
    textViews[i+numOfValues].setPadding(20,0,0,0);
    tableRow1.addView(textViews[i]);
    tableRow2.addView(textViews[i+numOfValues]);
}
StringTokenizer st = new StringTokenizer(value_1);
int tb_lenght =st.countTokens();
// int numOfValues =3;
tableRows = new TableRow[tb_lenght-2];
editTexts = new EditText[(tb_lenght-2)*numOfValues];
//int i=0, j=0, k=0;

```

```

k=0;
64for (i=0; i<tb_lenght-2; i++){
    tableRows[i] = new TableRow(fillTable.this);
    tableRows[i].setLayoutParams(params);
    tableLayout.addView(tableRows[i]);
    for (j=0; j<numOfValues; j++) {
        TableRow.LayoutParams params2 = new
        TableRow.LayoutParams(TableRow.LayoutParams.WRAP_CONTENT
            , TableRow.LayoutParams.WRAP_CONTENT);
        editTexts[k]= new EditText(fillTable.this);
        editTexts[k].setLayoutParams(params2);
        editTexts[k].setTextSize(18);
        editTexts[k].setInputType(InputType.TYPE_CLASS_TEXT);
        tableRows[i].addView(editTexts[k]);
        k++;
    }
}

```

Δύο μεταβλητές η “numOfValues” που μας δείχνει τον αριθμό των μεταβλητών και η “tb\_lenght” που μας δείχνει τον αριθμό των γραμμών του πίνακα χρησιμοποιούνται από δύο βρόχους για να μπορέσει να δημιουργηθεί ο πίνακας.

Όταν ο χρήστης δώσει την απάντηση και πατήσει το κουμπί OK ελέγχονται όλα τα πεδία (EditText)για την ορθότητα και εμφανίζεται το κατάλληλο μήνυμα. Αυτό φαίνεται στον παρακάτω κώδικα:



```

st_value1 = new StringTokenizer(value_1);
st_value2 = new StringTokenizer(value_2);
st_value3 = new StringTokenizer(value_3);
st_value4 = new StringTokenizer(value_4);
int nm = Integer.parseInt(numOfval);
st_value1.nextToken();
st_value1.nextToken();
st_value2.nextToken();
st_value2.nextToken();
if (nm==3){
    st_value3.nextToken();
    st_value3.nextToken();
}
else if (nm==4) {
    st_value3.nextToken();
    st_value3.nextToken();
    st_value4.nextToken();
    st_value4.nextToken();
}
StringTokenizer st = new StringTokenizer(value_1);
int tb_lenght =st.countTokens();
String
correct_value1,correct_value3,correct_value2,correct_value4,userTbA
65
nswer1,userTbAnswer2,userTbAnswer3,userTbAnswer4;
boolean checkForCorrectAnswer=true;
j=0;
for (i=0; i<tb_lenght-2; i++) {
    correct_value1=st_value1.nextToken();
    correct_value2=st_value2.nextToken();
    userTbAnswer1=
editTexts[i*nm].getText().toString().toLowerCase();
    userTbAnswer2=
editTexts[i*nm+1].getText().toString().toLowerCase();
    if ( !( ( userTbAnswer1.equals( correct_value1 ) )&&
( userTbAnswer2.equals( correct_value2) ) ) ) {
        checkForCorrectAnswer =false;
        break;
    }
    if (nm==3){
        correct_value3=st_value3.nextToken();
        userTbAnswer3= editTexts[i*nm +2
].getText().toString().toLowerCase();
        if ( !( userTbAnswer3.equals( correct_value3 ) ) ){
            checkForCorrectAnswer =false;
            break;
        }
    }
}
else if (nm==4) {
    correct_value3=st_value3.nextToken();
    correct_value4=st_value4.nextToken();
    userTbAnswer4= editTexts[i*nm +3

```

```

66].getText().toString().toLowerCase();
    userTbAnswer3= editTexts[i*nm
+2].getText().toString().toLowerCase();
    if ( !( ( userTbAnswer3.equals( correct_value3 ) )&&
( userTbAnswer4.equals( correct_value4) ) ) ) {
        checkForCorrectAnswer =false;
        break;
    }
}
}
}
if (checkForCorrectAnswer){
    Toast.makeText(fillTable.this,"Σωστη
απάντηση",Toast.LENGTH_SHORT).show();
}
else {
    Toast.makeText(fillTable.this,"Λάθος
απάντηση",Toast.LENGTH_SHORT).show();
}
}

```

Μια μεταβλητή `checkForCorrectAnswer` γίνεται `false` όταν κάποιο από τα πεδία δεν είναι ίσο με την απάντηση που υπάρχει στην βάση.

## Αλγόριθμος ανάγνωσης αρχείων για την καρτέλα “ΟΡΙΣΜΟΙ-ΘΕΩΡΙΑ SOS”.

Η θεωρία βρίσκεται σε εσωτερικά αρχεία της εφαρμογής και όχι στην βάση. Οι ερωτήσεις και οι απαντήσεις βρίσκονται μέσα στα αρχεία τα οποία είναι διαχωρισμένα ανάλογα με το κεφάλαιο. Κάθε φορά που ο χρήστης επιλέγει ένα κεφάλαιο για εξάσκηση γίνεται άνοιγμα του αντίστοιχου αρχείου και οι ερωτήσεις και απαντήσεις του κεφαλαίου αποθηκεύονται σε ένα στατικό πίνακα. Ο διαχωρισμός της ερώτησης και της απάντησης δίνεται με την λέξη “.end.” όταν βρίσκεται μόνη της σε κενή γραμμή.

```

try {
    reader = new BufferedReader(
        new InputStreamReader(getAssets().open(fileName)));
    int j=0;
    mLine="0";
    while ( mLine!=null){
        do {
            mLine = reader.readLine();
            if (mLine==null){
                break;
            }
        }
    }
}

```

```
    }
    if (! mLine.equals(".end.)){
        Qs = Qs +"\n"+ mLine;
    }
}
while (! (mLine).equals(".end." ));
tb_str[j]=Qs +"\n";
j++;
Qs="";
}
reader.close();
textView1.setText(tb_str[0]);
} catch (IOException e) {
    //log the exception
    e.printStackTrace();
}
```

#### 4.4 Στατιστικά στοιχεία για την εφαρμογή.

Για την υλοποίηση της εφαρμογής χρειάστηκε η παραγωγή κώδικα XML, JAVA, PHP, SQL.

Χρειάστηκε να προγραμματιστούν 11 διαφορετικά αρχεία XML με συνολικό όγκο 956 γραμμές κώδικα και 34kb.

Χρειάστηκε να προγραμματιστούν 11 διαφορετικά αρχεία JAVA με συνολικό όγκο 2267 γραμμές κώδικα και 110kb.

Χρειάστηκε να προγραμματιστούν 6 διαφορετικά αρχεία PHP στα οποία εκτελείτε και κώδικας SQL με συνολικό όγκο 179 γραμμές κώδικα και 4kb.

## Παράρτημα

### Βιβλιογραφία

<https://play.google.com/store/apps?hl=el>

<https://el.wikipedia.org/wiki/Smartphone>

<https://el.wikipedia.org/wiki/%CE%9B%CE%B5%CE%B9%CF%84%CE%BF%CF%85%CF%81%CE%B3%CE%B9%CE%BA%CF%8C%CF%83%CF%8D%CF%83%CF%84%CE%B7%CE%BC%CE%B1>

<https://el.wikipedia.org/wiki/Android>

[https://el.wikipedia.org/wiki/Android\\_Studio](https://el.wikipedia.org/wiki/Android_Studio)

<https://el.wikipedia.org/wiki/XAMPP>

[https://el.wikipedia.org/wiki/Apache\\_HTTP\\_%CE%B5%CE%BE%CF%85%CF%80%CE%B7%CF%81%CE%B5%CF%84%CE%B7%CF%84%CE%AE%CF%82](https://el.wikipedia.org/wiki/Apache_HTTP_%CE%B5%CE%BE%CF%85%CF%80%CE%B7%CF%81%CE%B5%CF%84%CE%B7%CF%84%CE%AE%CF%82)

<https://el.wikipedia.org/wiki/MySQL>

<https://el.wikipedia.org/wiki/PHP>

<https://developer.android.com/training/volley/index.html>

<https://developer.android.com/index.html>

<https://stackoverflow.com/>

<https://www.youtube.com/user/TICOONTECHNOLOGIES/featured>