



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Δημιουργία Android Εφαρμογής Διαδικτυακού Ραδιοφώνου



Του φοιτητή

Ηλιάδη Στέφανου

Αρ. Μητρώου: 144218

Επιβλέπων καθηγητής

Διαμαντάρας Κωνσταντίνος

Θεσσαλονίκη 2020

Τίτλος Δ.Ε.: Δημιουργία Android Εφαρμογής Διαδικτυακού Ραδιοφώνου

Κωδικός Δ.Ε. : 20175

Όνοματεπώνυμο φοιτητή: Ηλιάδης Στέφανος

Όνοματεπώνυμο εισηγητή: Κωνσταντίνος Διαμαντάρας

Ημερομηνία ανάληψης Δ.Ε.: 30/04/2020

Ημερομηνία περάτωσης Δ.Ε.: 17/06/2020

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Ηλιάδη Στέφανου που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

ΠΡΟΛΟΓΟΣ

Κάθε χρόνο, στο European School Radio – το πρώτο μαθητικό ραδιόφωνο, συμμετέχουν χιλιάδες μαθητές και δάσκαλοι - καθηγητές απ’ όλη την Ελλάδα και όχι μόνο. Έτσι, είναι απαραίτητο να υπάρχουν τα καλύτερα μέσα πρόσβασης στις λειτουργίες του μαθητικού ραδιοφώνου ώστε οι ακροατές να διατηρούν αμείωτο το ενδιαφέρον τους και να έχουν μια ευχάριστη εμπειρία χρήσης.

Η πτυχιακή αυτή, αφορά την δημιουργία μιας ολοκληρωμένης android εφαρμογής διαδικτυακού ραδιοφώνου, όπου ο χρήστης θα μπορεί να ακούει μέσω live streaming τις αγαπημένες του εκπομπές αλλά και να έχει πρόσβαση στο περιεχόμενο του European School Radio, μέσα από ένα εύχρηστο, μοντέρνο και φιλικό προς τον χρήστη περιβάλλον.

Η εφαρμογή αυτή έρχεται να καλύψει μια χρόνια ανάγκη των χρηστών για εύκολη πρόσβαση στο περιεχόμενο του ραδιοφώνου και στην αναπαραγωγή εκπομπών, δίνοντας λύση σε προβλήματα που έχει η υπάρχουσα ιστοσελίδα, ειδικά σε κινητές συσκευές.

ΠΕΡΙΛΗΨΗ

Στα κεφάλαια που ακολουθούν θα γίνει αναφορά στις μεθόδους και στις διαδικασίες με τις οποίες αναπτύχθηκε η εφαρμογή του European School Radio, όπως οι βιβλιοθήκες που χρησιμοποιήθηκαν και η λογική στην οποία στηρίχθηκε και αναπτύχθηκε ο αλγόριθμος της εφαρμογής. Θα γίνει μια εισαγωγή στον κόσμο του mobile app development, στα εργαλεία αλλά και στις γλώσσες προγραμματισμού που χρησιμοποιούνται για την ανάπτυξη εφαρμογών για κινητά, και φυσικά θα εξηγηθούν αναλυτικά τα χαρακτηριστικά και οι λειτουργίες της εφαρμογής του μαθητικού ραδιοφώνου, τα μέσα που χρησιμοποιήθηκαν αλλά και οτιδήποτε άλλο σχετικά μ’ αυτή.

Οι κύριες λειτουργίες της εφαρμογής που θα αναλυθούν εκτενέστερα παρακάτω είναι οι λειτουργίες «Ακούστε Ζωντανά», τα ραδιοφωνικά νέα, οι επόμενες εκπομπές, το ραδιοφωνικό πρόγραμμα και το αρχείο εκπομπών, που ανήκουν όλες οι παραγωγές του ραδιοφώνου.

Μερικές από τις πιο βασικές βιβλιοθήκες και εργαλεία που χρησιμοποιήθηκαν και παρουσιάζονται στα επόμενα κεφάλαια είναι η Exoplayer, για την αναπαραγωγή ήχου και βίντεο, η Jsoup, για την ανάλυση περιεχομένου html από ιστοσελίδες, το WP REST API v2.0, για την εξαγωγή πληροφοριών από ιστοσελίδες wordpress και πολλά ακόμα.

Επίσης, περιγράφεται ο τρόπος που λαμβάνονται όλα τα δεδομένα από την βάση δεδομένων του European School Radio αλλά και τα ερωτήματα που εκτελούνται για την λήψη αυτών.

Βασικός σκοπός της εφαρμογής είναι η ευκολότερη πρόσβαση των χρηστών κινητών τηλεφώνων του μαθητικού ραδιοφώνου στις δημοφιλέστερες λειτουργίες του European School Radio.

ABSTRACT

In the following chapters we will refer to the methods and procedures which the mobile application of European School Radio was developed, such as libraries used and logic on which the application algorithm was based and developed. The features and functions of the application will be explained in detail, and an introduction to the world of mobile app development will also be made.

The main activities of android app which will be analyzed more below, is activity " Listen Live", ESR News, the upcoming radio shows, radio schedule, and podcasts, where all radio productions of ESR belongs.

Some of the basic libraries and tools that are presented in the following chapters are Exoplayer, for playing audio and video, Jsoup, for displaying html content from websites, WP REST API v2.0, for exporting wordpress website data and much more.

Also, it describes how all data is received from the European School Radio database and the queries that are ran to take it.

The main purpose of the application is the easier access of smartphone users of the student radio to the most popular functions of the European School Radio.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω θερμά τα μέλη του European School Radio και ειδικά τους Γενιτζέ Αναγνώστη, Τούλιου Ευτυχία, Παπαδόπουλο Γεώργιο και Σιδέρη Συμεών για την πολύτιμη βοήθεια τους μέσω των ιδεών τους, των γνώσεων τους και όχι μόνο.

Τέλος, θα ήθελα να ευχαριστήσω τον κύριο Διαμαντάρα Κωνσταντίνο που μ' εμπιστεύτηκε στην υλοποίηση αυτής της πτυχιακής και ήταν τιμή μου που συνεργάστηκα μαζί του, μιας και συνεργαζόμαστε ήδη αρκετά χρόνια και για το εκπαιδευτικό πρόγραμμα του European School Radio στο ΔΙΠΑΕ.

ΠΕΡΙΕΧΟΜΕΝΑ

| | |
|---|----|
| ΠΡΟΛΟΓΟΣ | 3 |
| ΠΕΡΙΛΗΨΗ | 3 |
| ABSTRACT | 3 |
| ΕΥΧΑΡΙΣΤΙΕΣ..... | 4 |
| ΠΕΡΙΕΧΟΜΕΝΑ | 5 |
| Ευρετήριο εικόνων..... | 6 |
| ΚΕΦΑΛΑΙΟ 1 - ΕΙΣΑΓΩΓΗ..... | 8 |
| 1.1 European School Radio | 8 |
| 1.2 Ανάγκη για δημιουργία εφαρμογής κινητών τηλεφώνων..... | 8 |
| 1.3 Απαιτήσεις εφαρμογής | 9 |
| ΕΠΙΛΟΓΟΣ | 10 |
| ΚΕΦΑΛΑΙΟ 2 – Εισαγωγή στο λειτουργικό σύστημα Android | 11 |
| 2.1 Πληροφορίες για το λειτουργικό Android | 11 |
| 2.1.1 Αρχιτεκτονική | 11 |
| 2.1.2 Στοιχεία Android (Android Components) | 14 |
| 2.1.3 Ο κύκλος ζωής (Lifecycle) μιας εφαρμογής Android | 14 |
| 2.2 Τεχνολογίες που χρησιμοποιούνται για την ανάπτυξη mobile εφαρμογών..... | 15 |
| 2.2.1 Ανάπτυξη μιας Android εφαρμογής | 16 |
| 2.2.2 Ανάπτυξη μιας iOS εφαρμογής | 19 |
| 2.2.3 Ανάπτυξη μιας Cross-Platform εφαρμογής | 19 |
| ΕΠΙΛΟΓΟΣ | 21 |
| ΚΕΦΑΛΑΙΟ 3 – Android Εφαρμογή Ραδιοφώνου | 22 |
| 3.1 Γιατί επιλέχθηκε το λειτουργικό Android για την εφαρμογή..... | 22 |
| 3.2 Καταγραφή χαρακτηριστικών και απαιτήσεων εφαρμογής | 23 |
| 3.3 Λογισμικά-Εργαλεία ανάπτυξης της εφαρμογής..... | 24 |
| 3.4 Υλοποίηση εφαρμογής Android..... | 25 |
| 3.4.1 Άδειες-Permission | 25 |
| 3.4.2 Επικοινωνίας βάσης δεδομένων – εφαρμογής | 25 |
| 3.4.3 Αρχική οθόνη – Μενού Πλοήγησης | 26 |
| 3.4.4 Ακούστε Ζωντανά..... | 28 |
| 3.4.5 Νέα | 31 |
| 3.4.6 Επόμενες εκπομπές..... | 33 |

| | |
|---|----|
| 3.4.7 Ραδιοφωνικό Πρόγραμμα..... | 34 |
| 3.4.8 Παραγωγές..... | 39 |
| 3.4.9 Περισσότερα..... | 41 |
| 3.4.10 Μετάφραση Εφαρμογής..... | 42 |
| ΕΠΙΛΟΓΟΣ..... | 43 |
| ΚΕΦΑΛΑΙΟ 4 – Μελλοντικές επεκτάσεις..... | 44 |
| 4.1 Ειδοποιήσεις εφαρμογής..... | 44 |
| 4.2 Σύνδεση με social media..... | 44 |
| 4.3 Ανέβασμα ηχητικών αρχείων ή ηχογραφήσεων..... | 44 |
| 4.4 Δυναμική ενεργοποίηση/απενεργοποίηση μέρους της εφαρμογής..... | 44 |
| 4.5 Ψηφοφορία Διαγωνισμού..... | 45 |
| ΕΠΙΛΟΓΟΣ..... | 45 |
| ΣΥΜΠΕΡΑΣΜΑΤΑ..... | 46 |
| ΑΝΑΦΟΡΕΣ..... | 47 |
| ΒΙΒΛΙΟΓΡΑΦΙΑ..... | 48 |
| ΠΑΡΑΡΤΗΜΑΤΑ..... | 49 |
| ΠΑΡΑΡΤΗΜΑ Α - Ανάλυση json..... | 49 |
| ΠΑΡΑΡΤΗΜΑ Β – Αλγόριθμος υλοποίησης mp3 streaming με exoplayer..... | 51 |
| ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ..... | 52 |

Ευρετήριο εικόνων

| | |
|--|----|
| Εικόνα 1. Ενδεικτικά Μενού..... | 9 |
| Εικόνα 2. Αρχιτεκτονική Android..... | 12 |
| Εικόνα 3. Κύκλος ζωής εφαρμογής Android..... | 16 |
| Εικόνα 4. Εισαγωγή Android Studio..... | 17 |
| Εικόνα 5. Δημιουργία project..... | 18 |
| Εικόνα 6. Περιβάλλον Android Studio..... | 18 |
| Εικόνα 7. Δοκιμαστικό Μενού..... | 26 |
| Εικόνα 8. Τελικό Μενού Εφαρμογής..... | 27 |

| | |
|---|----|
| Εικόνα 9. Ακούστε Ζωντανά..... | 28 |
| Εικόνα 10. Ραδιοφωνικά Νέα..... | 31 |
| Εικόνα 11. Λεπτομέρειες Νέων..... | 32 |
| Εικόνα 12. Επόμενες Εκπομπές..... | 33 |
| Εικόνα 13. CalendarView..... | 35 |
| Εικόνα 14. Ραδιοφωνικό Πρόγραμμα..... | 38 |
| Εικόνα 15. Μορφή json για podcasts..... | 40 |
| Εικόνα 16. Εκπομπές..... | 41 |
| Εικόνα 17. Περισσότερα..... | 42 |
| Εικόνα 18. Μετάφραση Εφαρμογής..... | 43 |

ΚΕΦΑΛΑΙΟ 1 - ΕΙΣΑΓΩΓΗ

Σ αυτό το κεφάλαιο, θα γίνει μια μικρή αναφορά στο πρώτο μαθητικό ραδιόφωνο, το European School Radio. Θα παρουσιαστεί ο τρόπος και ο σκοπός λειτουργίας του, αλλά και κάποιες σύντομες απαιτήσεις της εφαρμογής του μαθητικού ραδιοφώνου για κινητά τηλέφωνα.

1.1 European School Radio

Το European School Radio, το πρώτο μαθητικό διαδικτυακό ραδιόφωνο στην Ελλάδα, ξεκίνησε την λειτουργία του το 2009. Από τότε, λειτουργεί 24 ώρες την ημέρα, 7 μέρες την εβδομάδα, δίνοντας την δυνατότητα σε χιλιάδες μαθητές πρωτοβάθμιας και δευτεροβάθμιας εκπαίδευσης αλλά και δασκάλους-καθηγητές, να εκφραστούν μέσω της φωνής τους, δημιουργώντας τις δικές τους ηχογραφημένες ή ζωντανές εκπομπές [1].

Στο European School Radio – Το Πρώτο Μαθητικό Ραδιόφωνο, συμμετέχουν κάθε χρόνο εκατοντάδες σχολεία πρωτοβάθμιας και δευτεροβάθμιας εκπαίδευσης με χιλιάδες μαθητές, δημιουργώντας τις δικές σου ραδιοφωνικές παραγωγές.

Υπεύθυνοι για την ομαλή λειτουργία του ραδιοφώνου και την διασφάλιση της ποιότητας του περιεχομένου που αναπαράγεται είναι δάσκαλοι, καθηγητές, φοιτητές αλλά και μαθητές.

Μέσα στην ιστοσελίδα του European School Radio, οι χρήστες μπορούν να ακούσουν τις ραδιοφωνικές παραγωγές των μαθητών ζωντανά, να ακούσουν σε δεύτερο χρόνο τις εκπομπές που έχουν ήδη παίξει μέσα από το αρχείο εκπομπών αλλά και να δει το ραδιοφωνικό πρόγραμμα, να διαβάσει τα νέα του μαθητικού ραδιοφώνου, να παρακολουθήσει μαθήματα ραδιοφώνου και πολλά ακόμα.

1.2 Ανάγκη για δημιουργία εφαρμογής κινητών τηλεφώνων

Τα τελευταία χρόνια, όλο και μεγαλύτερο είναι το ποσοστό των ανθρώπων που χρησιμοποιούν τα κινητά τους τηλέφωνα, όχι μόνο για να περιηγηθούν απλά στο διαδίκτυο, αλλά και για να ακούσουν το αγαπημένο τους τραγούδι ή να δουν το αγαπημένο τους βίντεο, ακόμα και για να κάνουν τις αγορές τους. Πιο συγκεκριμένα, έχει καταγραφεί πως πάνω από τους μισούς χρήστες που χρησιμοποιούν καθημερινά το διαδίκτυο, προτιμούν την κινητή τους συσκευή έναντι του υπολογιστή τους. Σε ορισμένες περιπτώσεις μάλιστα, όπως είναι το facebook ή το twitter, φαίνεται πως το ποσοστό αυτό ξεπερνάει ακόμα και το 90%!

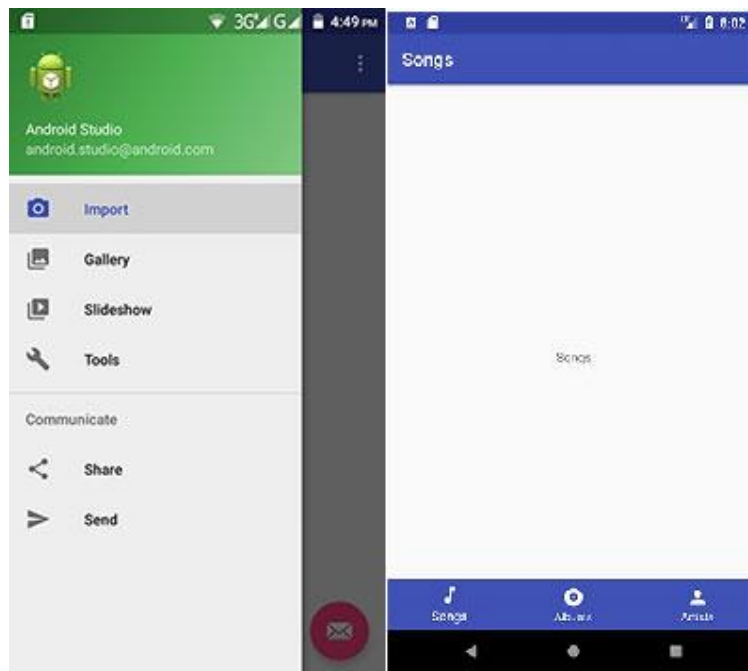
Όσο αφορά την ιστοσελίδα του European School Radio, σύμφωνα με τα επίσημα στατιστικά της ιστοσελίδας που αφορούν την επισκεψιμότητα, την φετινή χρονιά το 73% των χρηστών επισκέφθηκε την ιστοσελίδα του μαθητικού ραδιοφώνου από το κινητό τους τηλέφωνο, το 3% από tablet ενώ από υπολογιστή το 24%. Έτσι, είναι σαφές πως η εύκολη περιήγηση από κινητές συσκευές είναι πολύ σημαντική ώστε ο χρήστης να διατηρήσει αμείωτο το ενδιαφέρον του και να συνεχίσει την περιήγηση του στις λειτουργίες του ραδιοφώνου. Αρκετοί χρήστες διαμαρτυρήθηκαν πως πολλές φορές η περιήγηση τους ή η ακρόαση της αγαπημένης τους εκπομπής από το κινητό τους ήταν αρκετά δύσκολη, γεγονός που τους ανάγκαζε πολλές φορές στο να εγκαταλείψουν αυτή την προσπάθεια. Γι' αυτό τον λόγο κρίθηκε απαραίτητο να δημιουργηθεί μια εφαρμογή για κινητά τηλέφωνα, όπου οι ακροατές θα μπορούν εύκολα να έχουν πρόσβαση σε βασικές λειτουργίες του ραδιοφώνου όπως είναι η ακρόαση μιας εκπομπής, η παρακολούθηση των ραδιοφωνικών νέων, η

παρακολούθηση του προγράμματος σχετικά με τις εκπομπές που έπαιξαν ή που πρόκειται να παίξουν, η αναζήτηση και αναπαραγωγή των εκπομπών μέσα από το αρχείο εκπομπών και πολλά άλλα.

1.3 Απαιτήσεις εφαρμογής

Όπως αναφέρθηκε και παραπάνω, οι χρήστες που προτιμούν να χρησιμοποιούν τις κινητές τους συσκευές έναντι των υπολογιστών είναι πολλοί και αυξάνονται συνεχώς. Γι' αυτό τον λόγο, η δημιουργία μιας mobile εφαρμογής θα βοηθούσε τους ακροατές του μαθητικού ραδιοφώνου να ακούν τις αγαπημένες τους εκπομπές αλλά και να χρησιμοποιούν με ιδιαίτερη ευκολία πολλές από τις λειτουργίες του μαθητικού ραδιοφώνου.

Αρχικά, όσο αφορά το περιβάλλον (διεπαφή) της εφαρμογής, έγινε προσπάθεια ώστε να είναι όσο το δυνατόν πιο απλό αλλά ταυτόχρονα μοντέρνο και εύχρηστο. Γι' αυτό τον λόγο, προτιμήθηκε το μενού να γίνει custom, μιας και οι επιλογές που έδινε το πρόγραμμα (android studio) στο οποίο δημιουργήθηκε η εφαρμογή δεν ήταν ικανοποιητικές. Στην παρακάτω εικόνα (Εικόνα 1), φαίνονται δύο ενδεικτικά μενού που μπορείς να δημιουργήσεις μέσα από το android studio.



Εικόνα 1 "Ενδεικτικά Μενού"

Επιπλέον, είναι αναγκαίο η εφαρμογή να υποστηρίζει τις δημοφιλέστερες λειτουργίες του ραδιοφώνου, όπως είναι η ζωντανή μετάδοση της ραδιοφωνικής ροής αλλά και η χρήση του chat που υπάρχει και στην ιστοσελίδα, ώστε ο κάθε ακροατής να μπορεί να συμμετέχει και να αλληλεπιδρά με τους ραδιοφωνικούς παραγωγούς της κάθε εκπομπής, επικοινωνώντας μαζί τους με γραπτά μηνύματα.

Τέλος, είναι σημαντικό η εφαρμογή να υποστηρίζει παραπάνω από μια γλώσσες, ώστε να μπορούν χρήστες από όλες τις χώρες να την χρησιμοποιήσουν. Προς το παρόν, η εφαρμογή υποστηρίζει αγγλικά και ελληνικά, ενώ φυσικά υπάρχει δυνατότητα να μεταφραστεί και σε άλλες γλώσσες, εφόσον υπάρξει τέτοια ανάγκη.

Σε επόμενα κεφάλαια θα παρουσιαστούν αναλυτικότερα οι απαιτήσεις και οι λειτουργίες της εφαρμογής.

ΕΠΙΛΟΓΟΣ

Σ αυτό το κεφάλαιο έγινε μια πρώτη εισαγωγή στο European School Radio – το Πρώτο Μαθητικό Ραδιόφωνο, έγινε κατανοητό το πόσο σημαντική ήταν η δημιουργία μιας εφαρμογής για κινητά τηλέφωνα, ενώ παρουσιάστηκαν συνοπτικά οι απαιτήσεις της εφαρμογής του European School Radio.

Στο επόμενο κεφάλαιο θα γίνει μια εισαγωγή στο λειτουργικό σύστημα Android αλλά και γενικά στο mobile development.

ΚΕΦΑΛΑΙΟ 2 – Εισαγωγή στο λειτουργικό σύστημα Android

Σ αυτό το κεφάλαιο θα παρουσιαστούν κάποιες πληροφορίες σχετικά με το λειτουργικό σύστημα Android, όπως η αρχιτεκτονική του και ο τρόπος λειτουργίας του, ενώ θα αναφερθούν οι τεχνολογίες που χρησιμοποιούνται για ανάπτυξη εφαρμογών για κινητά τηλέφωνα σε διάφορα λειτουργικά συστήματα.

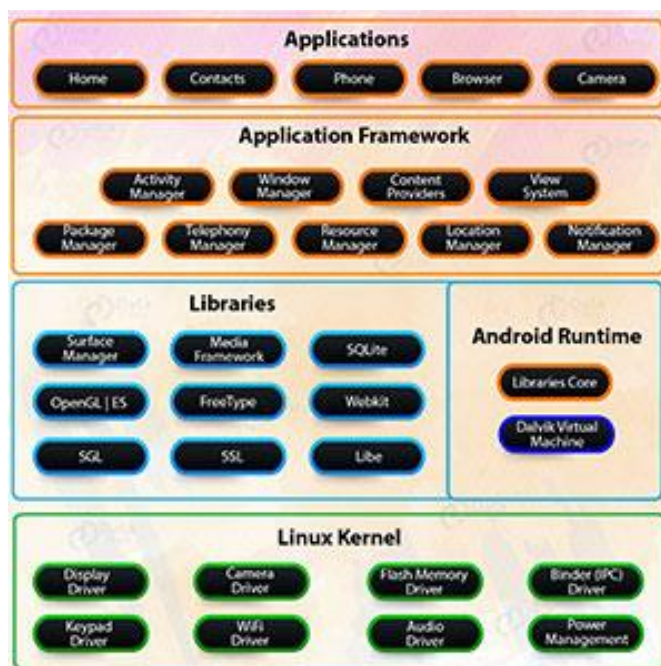
2.1 Πληροφορίες για το λειτουργικό Android

Το Android είναι ένα open-source λειτουργικό σύστημα βασισμένο στον πυρήνα του λειτουργικού Linux και είναι σχεδιασμένο κατά βάση για κινητές συσκευές με οθόνες αφής, όπως έξυπνα τηλέφωνα (smartphones) και tablets [2]. Είναι το πιο ευρέως διαδεδομένο λειτουργικό σύστημα στον κόσμο, με το μεγαλύτερο ποσοστό των κινητών συσκευών να το χρησιμοποιούν. Ο πηγαίος κώδικας του Android δίνεται από την Google με άδεια χρήσης Apache, μια άδεια που επιτρέπει στο λογισμικό να τροποποιείται ελεύθερα. Το Android παρουσιάστηκε το 2007, ταυτόχρονα με την ίδρυση του Open Handset Alliance, που ήταν μια κοινοπραξία εταιριών λογισμικού, υλικού και τηλεπικοινωνιών[3]. Η πρώτη Android συσκευή κυκλοφόρησε τον Σεπτέμβριο του 2008 και ήταν το HTC Dream[4].

2.1.1 Αρχιτεκτονική

Το Android είναι πολλά παραπάνω από ένα λειτουργικό σύστημα για κινητά τηλέφωνα. Είναι μια στοίβα λογισμικού που αποτελείται από το λειτουργικό σύστημα, τις υπηρεσίες διασύνδεσης με τις εφαρμογές αλλά και τις κύριες εφαρμογές όπως είναι οι επαφές, η εφαρμογή αποστολής SMS, το ημερολόγιο και πολλές ακόμα. Όπως φαίνεται και παρακάτω, η αρχιτεκτονική του λειτουργικού Android αποτελείται από 5 βασικά επίπεδα:

- Τον πυρήνα Linux (Linux Kernel)
- Τις βιβλιοθήκες (Libraries)
- Τον χρόνο εκτέλεσης (Android Runtime)
- Το πλαίσιο εφαρμογών (Application Framework)
- Τις εφαρμογές (Core Applications)



Εικόνα 2 "Αρχιτεκτονική Android"

Linux Kernel

Στο κατώτερο επίπεδο βρίσκεται ο πυρήνας του λειτουργικού Linux. Περιλαμβάνει και διαχειρίζεται όλα τα προγράμματα οδήγησης (drivers) όπως πρόγραμμα οδήγησης για την κάμερα, το πληκτρολόγιο, την οθόνη, τον ήχο, την μνήμη κ.τ.λ τα οποία είναι απαραίτητα για την εκτέλεση μιας εφαρμογής android. Επιπλέον, αυτό το επίπεδο είναι υπεύθυνο για την διαχείριση μνήμης, ισχύος, διεργασιών, ασφάλειας, πρόσβαση σε πόρους κ.ο.κ.

Libraries

Ακριβώς επάνω από τον πυρήνα του Linux υπάρχει ένα σύνολο βιβλιοθηκών όπως βιβλιοθήκες για αναπαραγωγή και εγγραφή ήχου, βιβλιοθήκες SSL που είναι υπεύθυνες για την ασφάλεια στο διαδίκτυο, βιβλιοθήκες για αποθήκευση και κοινή χρήση δεδομένων. Ενδεικτικά, παρακάτω αναφέρονται κάποιες βασικές βιβλιοθήκες:

Android.app: Παρέχει πρόσβαση στο μοντέλο της εφαρμογής και είναι απαραίτητη για όλες τις εφαρμογές Android.

Android.content: Διευκολύνει την πρόσβαση στο περιεχόμενο της εφαρμογής, στην ανταλλαγή μηνυμάτων μεταξύ των εφαρμογών και των στοιχείων (components) της κάθε εφαρμογής.

Android.database: Χρησιμοποιείται για την πρόσβαση στα δεδομένα της αποθήκευσης, για αποθήκευση δεδομένων της εφαρμογής κτλ μέσω της SQLite.

Android.hardware: Περιλαμβάνει ένα API που παρέχει πρόσβαση σε υλικό (hardware) όπως το επιταχυνσιόμετρο και ο αισθητήρας φωτός.

Android.opengl: Είναι μια διεπαφή για την απεικόνιση 2D και 3D γραφικών.

Android.widget : Περιλαμβάνει ένα μεγάλο σύνολο από στοιχεία διεπαφής όπως κουμπιά, ετικέτες κειμένου, λίστες, radio buttons και πολλά ακόμα στοιχεία (components).

Android.webkit: Περιλαμβάνει ένα σύνολο κλάσεων - εργαλείων που προορίζονται για πρόσβαση στο διαδίκτυο.

Android.media: Περιλαμβάνει κλάσεις για υποστήριξη αναπαραγωγής ήχου και βίντεο.

Android Runtime

Σε αυτό το επίπεδο, περιλαμβάνονται κάποιες βασικές βιβλιοθήκες πυρήνα και η εικονική μηχανή Dalvik (Dalvik virtual machine).

Οι βιβλιοθήκες επιτρέπουν στους προγραμματιστές εφαρμογών android να δημιουργούν εφαρμογές, ενώ η εικονική μηχανή Dalvik είναι ένα είδος JVM (Java Virtual Machine) ειδικά σχεδιασμένο και βελτιστοποιημένο για Android. Η Dalvik Virtual Machine χρησιμοποιεί βασικά χαρακτηριστικά του λειτουργικού συστήματος Linux όπως διαχείριση μνήμης και multi-threading, ενώ επιτρέπει σε κάθε εφαρμογή android να τρέχει με την δική της διεργασία, άρα σε κάθε συσκευή μπορούν να εκτελούνται πολλές εφαρμογές ταυτόχρονα.

Application Framework

Σ αυτό το επίπεδο περιλαμβάνονται πολλές υπηρεσίες (services) υψηλού επιπέδου με την μορφή κλάσεων και άλλων API, διαθέσιμες για τους προγραμματιστές εφαρμογών android.

Το Android Framework περιλαμβάνει τις ακόλουθες υπηρεσίες:

Activity Manager: Ελέγχει όλη την «διάρκεια ζωής» της εφαρμογής (activity lifecycle).

Content Provides: Επιτρέπει στις εφαρμογές να δημοσιεύουν και να μοιράζονται δεδομένα με άλλες εφαρμογές.

Resource Manager: Παρέχει πρόσβαση σε στοιχεία της εφαρμογής όπως είναι οι συμβολοσειρές, εικόνες, αρχεία xml κτλ, τα οποία τρέχουν κατά την εκτέλεση της εφαρμογής.

Στην παρούσα εφαρμογή για το ραδιόφωνο, αυτά τα στοιχεία είναι τα strings που εναλλάσσονται ανάλογα με την γλώσσα που έχει η εφαρμογή σε μια κινητή συσκευή, τα αρχεία xml που καθορίζουν την εμφάνιση διάφορων κουμπιών ή λιστών στην εφαρμογή αλλά και τα αρχεία xml που καθορίζουν την εμφάνιση των activities της εφαρμογής, οι εικόνες και γενικά οτιδήποτε βρίσκεται μέσα στον φάκελο Resources.

Notification Manager: Επιτρέπει στις εφαρμογές να εμφανίζει ειδοποιήσεις στους χρήστες.

View System: Είναι ένα σύνολο στοιχείων που χρησιμοποιούνται για την δημιουργία διεπαφών μιας εφαρμογής android, όπως είναι οι λίστες, το πλέγμα (GridView), κουμπιά κτλ.

Applications

Σ αυτό το επίπεδο υπάρχουν όλες οι βασικές εφαρμογές που χρησιμοποιούμε σαν χρήστες android. Βρίσκεται στο ανώτερο επίπεδο της αρχιτεκτονικής του λειτουργικού συστήματος android. Τέτοιες εφαρμογές είναι η εφαρμογή αποστολής sms, επαφών, ημερολογίου, χαρτών και πολλές ακόμα [5-7].

2.1.2 Στοιχεία Android (Android Components)

Τα στοιχεία μιας εφαρμογής Android είναι τα δομικά στοιχεία του Android. Κάθε στοιχείο έχει τον δικό του ρόλο και σκοπό στην εφαρμογή. Τα κύρια στοιχεία μιας εφαρμογής Android είναι:

- Δραστηριότητες (Activities)
- Υπηρεσίες (Services)
- Broadcast Receivers
- Πάροχοι Περιεχομένου (Content Providers)

Οι δραστηριότητες (**Activities**) αφορούν την διεπαφή και πιο συγκεκριμένα τις ενέργειες-αλληλεπιδράσεις του χρήστη στην οθόνη. Εκεί γράφεται μια νέα ενέργεια-λειτουργία της εφαρμογής. Σε κάθε εφαρμογή υπάρχει τουλάχιστον μια δραστηριότητα.

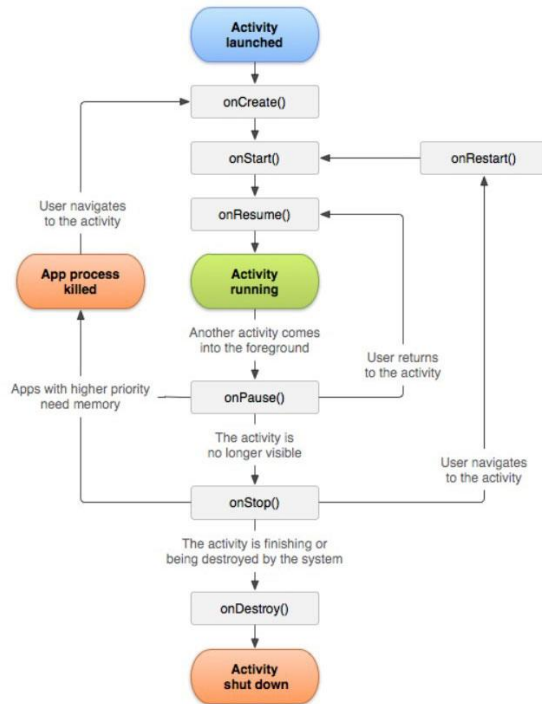
Οι υπηρεσίες (**Services**) είναι ενέργειες που εκτελούνται στο προσκήνιο (background) της εφαρμογής όπως αναπαραγωγή μουσικής της εφαρμογής όταν ο χρήστης περιηγείται σε άλλη εφαρμογή.

Οι **Broadcast Receivers** χρησιμοποιούνται για διαχείριση των μηνυμάτων από άλλες εφαρμογές ή από το σύστημα. Για παράδειγμα, όταν η μπαταρία του τηλεφώνου είναι χαμηλή, τότε το λειτουργικό Android ενεργοποιεί ένα μήνυμα για να ξεκινήσει η λειτουργία εξοικονόμησης ενέργειας.

Οι πάροχοι περιεχομένου (**Content Providers**) χρησιμοποιούνται για την μεταφορά και την αποθήκευση δεδομένων από μια εφαρμογή σε μια άλλη [8].

2.1.3 Ο κύκλος ζωής (Lifecycle) μιας εφαρμογής Android

Ο κύκλος ζωής μιας εφαρμογής φαίνεται στο παρακάτω σχήμα:



Εικόνα 3 "Κύκλος ζωής εφαρμογής Android"

Καταστάσεις (μέθοδοι) ενός κύκλου ζωής μιας εφαρμογής:

- `onCreate()`: Καλείται όταν το Activity δημιουργείται την πρώτη φορά
- `onStart()`: Καλείται όταν το Activity γίνεται ορατό στον χρήστη
- `onResume()`: Καλείται όταν το Activity αρχίζει να αλληλεπιδρά με τον χρήστη. Πάντα, ακολουθείτε από την `onPause()`.
- `onPause()`: Καλείται όταν το Activity μπαίνει στο προσκήνιο(background), αλλά δεν έχει τερματιστεί.
- `onStop`: Καλείται όταν το Activity δεν είναι πλέον ορατό στον χρήστη.
- `onRestart()`: Καλείται μετά την `onStop()`, δηλαδή όταν το Activity σταματήσει και ξεκινήσει ξανά.
- `onDestroy()`: Καλείται όταν το Activity κλείσει ή τερματιστεί [9].

2.2 Τεχνολογίες που χρησιμοποιούνται για την ανάπτυξη mobile εφαρμογών

Αρχικά, θα πρέπει να ξεκαθαριστεί πως υπάρχουν πολλοί τρόποι και τεχνολογίες που μπορεί κάποιος να δημιουργήσει μια εφαρμογή για κινητά τηλέφωνα. Σε πρώτη φάση, ο δημιουργός της εφαρμογής θα πρέπει να αποφασίσει για ποιο ή ποια λειτουργικά συστήματα προορίζεται. Τα πιο δημοφιλή λειτουργικά συστήματα για κινητά είναι 2: Το *Android* και το *iOS*. Αυτά τα 2 λειτουργικά πλέον χρησιμοποιούνται από το 99+% των χρηστών. Σήμερα, περίπου το 71% των χρηστών έξυπνων τηλεφώνων χρησιμοποιούν το λειτουργικό *Android*, ενώ το υπόλοιπο 29% το *iOS*. Έτσι, γίνεται σαφές πως ανάλογα με το λειτουργικό σύστημα που προορίζεται η εφαρμογή, αλλάζουν οι τεχνολογίες, η εφαρμογή ανάπτυξης της εφαρμογής (IDE) ακόμα και η γλώσσα προγραμματισμού

με την οποία θα γραφτεί. Παρακάτω θα γίνει μια μικρή αναφορά στις τεχνολογίες που χρησιμοποιούνται για την δημιουργία εφαρμογών σε iOS αλλά και σε cross-platform εφαρμογές, ενώ θα δοθεί έμφαση στις τεχνολογίες που χρησιμοποιούνται για το λειτουργικό Android, μιας και η εφαρμογή του European School Radio είναι σχεδιασμένη για κινητά Android.

2.2.1 Ανάπτυξη μιας Android εφαρμογής

Το λειτουργικό Android αποτελεί αδιαμφισβήτητα το πιο δημοφιλές λειτουργικό σύστημα για κινητά τηλέφωνα. Όπως αναφέρθηκε και παραπάνω, πάνω από 7 στους 10 χρήστες κινητών τηλεφώνων χρησιμοποιούν αυτό το λειτουργικό.

Η πιο γνωστή γλώσσα προγραμματισμού για την δημιουργία μιας android εφαρμογής είναι η Java, ενώ εδώ και λίγα χρόνια πολύ δημοφιλής έγινε και η Kotlin. Ταυτόχρονα, χρησιμοποιείται και η XML για την δομή των activities ενός project.

Τα πιο γνωστά περιβάλλοντα ανάπτυξης (IDE) για την ανάπτυξη είναι το Android Studio, το Eclipse και το Netbeans. Να σημειωθεί πως το Android Studio έχει ενσωματωμένο το Android SDK κατά την λήψη του, ενώ στο Eclipse και το NetBeans απαιτείται να γίνει ξεχωριστή λήψη και ενσωμάτωση στο πρόγραμμα.

2.2.1.1 Λίγες πληροφορίες για το Android SDK

Το Android SDK είναι ένα σύνολο εργαλείων και βιβλιοθηκών, απαραίτητων για την ανάπτυξη μιας Android εφαρμογής. Επιπλέον, παρέχονται APIs για την χρήση προγραμμάτων περιήγησης στο διαδίκτυο, εμφάνισης 2D και 3D γραφικών, χρήση τεχνολογιών Bluetooth, GPS, Wi-Fi, 3G/4G, πυξίδα, επιταχυνσιόμετρο, αποθήκευσης δεδομένων σε βάση δεδομένων, εμφάνιση πολυμέσων όπως ήχος ή βίντεο κτλ. Επιπλέον, περιέχει και έναν εξομοιωτή μιας εικονικής κινητής συσκευής (emulator), δίνοντας την δυνατότητα στον προγραμματιστή να τρέχει και να ελέγχει την εφαρμογή που φτιάχνει ανά πάσα στιγμή, χωρίς να απαιτείται μια πραγματική συσκευή Android. Αξίζει να σημειωθεί πως αυτός ο εξομοιωτής λειτουργεί ακριβώς όπως μια πραγματική συσκευή Android. Για παράδειγμα, έχει κανονικό μενού και πλήκτρα πλοήγησης, την εμφάνιση ειδοποιήσεων, την αναπαραγωγή ήχου και βίντεο, την προσομοίωση διακοπών από ένα SMS ή μια τηλεφωνική κλήση κτλ. Η εγκατάσταση του Android SDK είναι δωρεάν και πολύ απλή, ενώ σε IDE όπως το Android Studio περιέχεται ήδη κατά την εγκατάσταση του προγράμματος, επομένως δεν χρειάζεται να το εγκαταστήσουμε ξεχωριστά.

2.2.1.2 Λίγες πληροφορίες για το Android Studio

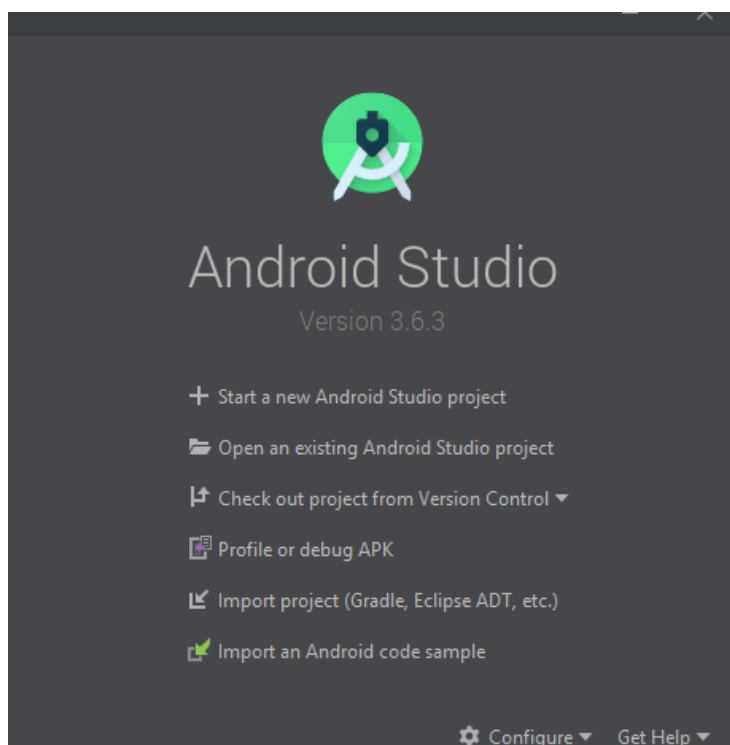
Αποτελεί αδιαμφισβήτητα το πιο γνωστό περιβάλλον ανάπτυξης μιας android εφαρμογής. Έχει σχεδιαστεί και αναπτυχθεί από την Google, είναι δωρεάν και περιέχει εργαλεία όπως το Android Virtual Device Manager (που επιτρέπει την δημιουργία μιας εικονικής συσκευής Android μέσω ενός emulator, όπου εκεί μπορεί ο προγραμματιστής να «τρέχει» και να δοκιμάζει την εφαρμογή που δημιουργεί). Το Android Studio μπορεί να τρέξει σε Windows, macOS και Linux, ενώ περιλαμβάνει χαρακτηριστικά όπως:

- Υποστήριξη Gradle και Maven
- Πλούσιο επεξεργαστή κειμένου και δυνατότητα για drag & drop στοιχείων στο UI, προεπισκόπηση σε πολλά μεγέθη και διαμορφώσεις οθονών κτλ.

- Ισχυρή επεξεργασία κώδικα (Powerful Code Editing), όπως έξυπνο σύστημα αυτοσυμπλήρωσης κώδικα, πλοήγηση από το ένα αρχείο σ' ένα άλλο αλληλένδετο, προχωρημένη και ασφαλή επεξεργασία (π.χ. μετονομασία ενός αρχείου και εφαρμογή του παντού μέσα στο project)
- Ανάλυση κώδικα κατά την επεξεργασία (On-the-fly Code Analysis), όπου εμφανίζονται προειδοποιήσεις και σφάλματα κατά την σύνταξη, καθώς και πιθανές λύσεις.
- Εργαλεία Lint (Lint Tools), τα οποία δίνουν την δυνατότητα καταγραφής της απόδοσης, της χρηστικότητας, της συμβατότητας μεταξύ των εκδόσεων Android ή πρόβλεψης σφαλμάτων [10].

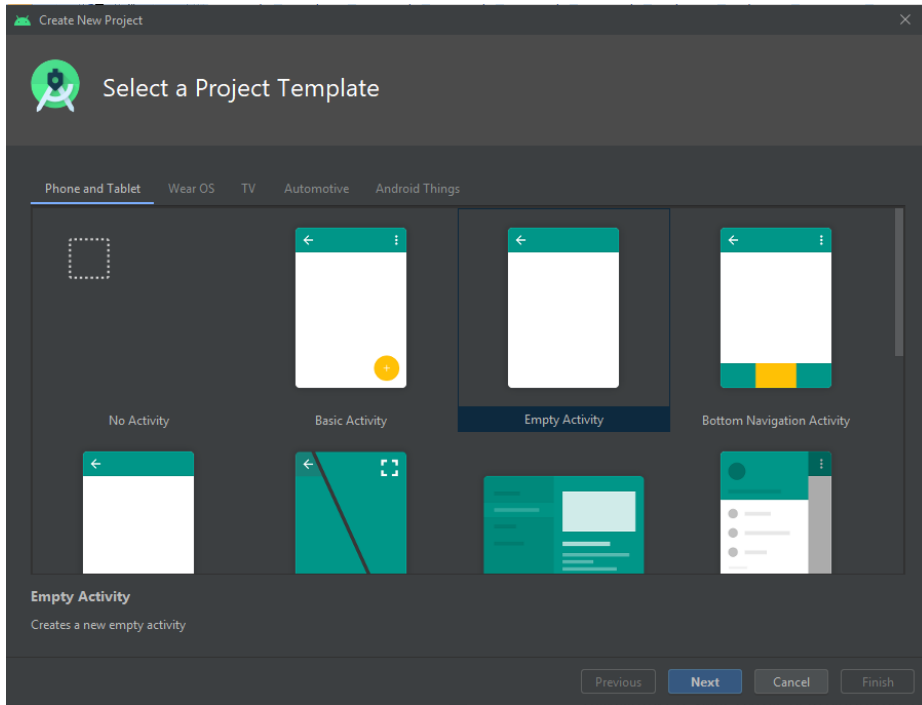
2.2.1.3 Εισαγωγή στο περιβάλλον του Android Studio

Μόλις ανοίξει το Android Studio, μας εμφανίζεται η επιλογή να προσθέσουμε ή να ανοίξουμε ένα υπάρχων project στο Android Studio.



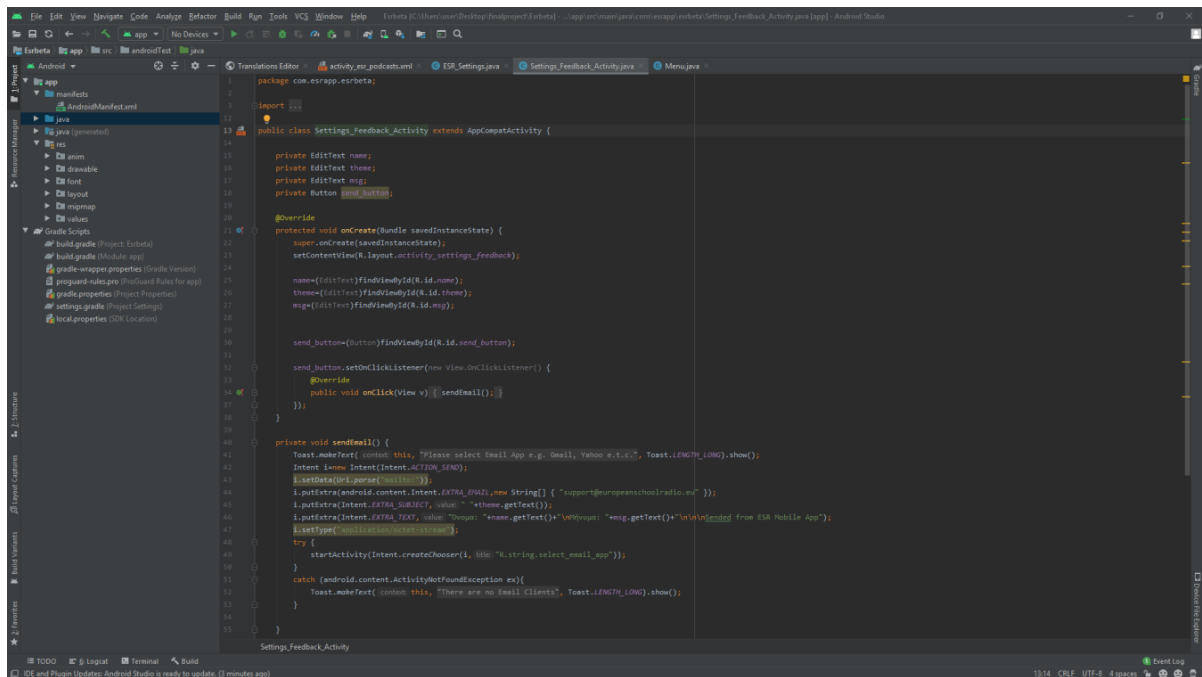
Εικόνα 4 "Εισαγωγή Android Studio"

Στην συνέχεια, πατώντας την επιλογή να δημιουργήσουμε ένα νέο Project, μας εμφανίζεται η παρακάτω οθόνη (εικόνα 5), όπου επιλέγουμε ανάλογα με το περιεχόμενο που θέλουμε να έχει η εφαρμογή. Έχει επιλογές για Empty Activity (όπου δημιουργείται ένα αρχείο με τον στοιχειώδη κώδικα και στοιχειώδη αρχεία που απαιτούνται για την δημιουργία μιας εφαρμογής), για Navigation Drawer (που παρέχει την δυνατότητα δημιουργίας μενού στην εφαρμογή) κτλ. Εγώ προτίμησα να δημιουργήσω μια κενή εφαρμογή (Empty Activity) και να δημιουργήσω μόνος μου τα στοιχεία της εφαρμογής.



Εικόνα 5 "Δημιουργία project"

Όταν δημιουργήσουμε μια εφαρμογή και ξεκινήσουμε να την σχεδιάζουμε, η οθόνη του Android Studio θα είναι κάπως έτσι (εικόνα 6):



Εικόνα 6 "Περιβάλλον Android Studio"

Στα αριστερά, βλέπουμε τα δομικά στοιχεία της εφαρμογής μας, όπως είναι ο φάκελος Manifest, ο φάκελος Java, ο φάκελος Resource, και τα αρχεία Gradle. Πιο αναλυτικά περιέχονται:

- **Φάκελος Manifest**, όπου περιέχει το αρχείο Manifest.xml, που είναι η ρίζα του συνόλου της εφαρμογής. Εδώ περιέχονται οι κλάσεις (Activities), οι άδειες (Permission) που απαιτούνται

για την εκτέλεση της εφαρμογής, το πακέτο (package), τις ειδικές κλάσεις όπως τα Services, broadcast receiver κτλ.

- **Φάκελος Java**, όπου περιέχονται όλα τα αρχεία java της εφαρμογής. Αυτά τα αρχεία περιέχουν τον αλγόριθμο που χρησιμοποιείται για την εκτέλεση των λειτουργιών της εφαρμογής.
- **Φάκελος Resource**, όπου περιέχονται όλα τα αρχεία που χρησιμοποιούνται στην εφαρμογή εκτός από τα αρχεία java. Για παράδειγμα, εδώ υπάρχουν τα αρχεία XML των Activities (φάκελος layout), οι εικόνες (φάκελος drawable), τα χρώματα της εφαρμογής (values=>colors.xml), οι συμβολοσειρές (strings) που χρησιμοποιούνται (φάκελος strings) κ.τ.λ.
- **Αρχεία Gradle**, όπου εδώ περιέχονται στοιχεία και ρυθμίσεις όπως η μικρότερη έκδοση λειτουργικού Android που θα τρέχει η εφαρμογή, το όνομα τωρινής έκδοσης της εφαρμογής, τις βιβλιοθήκες που έχουν προστεθεί από τον χρήστη και πολλά ακόμα.

2.2.2 Ανάπτυξη μιας iOS εφαρμογής

Για την δημιουργία μιας εφαρμογής iOS, η δημοφιλέστερη γλώσσα προγραμματισμού που χρησιμοποιείται είναι η *Swift*, αλλά και η *Objective-C*. Η *Objective-C* είναι ένα υπερσύνολο της γλώσσας C με δυνατότητες αντικειμενοστραφών χαρακτηριστικών, ενώ ήταν η βασική γλώσσα με την οποία ένας προγραμματιστής μπορούσε να δημιουργήσει μια εφαρμογή iOS, μέχρι που εμφανίστηκε η *Swift*. Πλέον, οι περισσότερες εφαρμογές δημιουργούνται με *Swift*, μιας και αυτή η γλώσσα έχει πολλά πλεονεκτήματα έναντι της *Objective-C* όπως είναι:

- Καλύτερα σχεδιασμένος κώδικας, άρα ευκολότερος στην ανάγνωση και στην συντήρηση
- Ασφαλέστερη και αποδοτικότερη
- Μεγαλύτερη υποστήριξη μεταξύ Apple συσκευών
- Αυτόματα διαχείριση μνήμης.

Όσο αφορά το IDE (Integrated Development Environment), δηλαδή το περιβάλλον που θα δημιουργηθεί η iOS εφαρμογή, το δημοφιλέστερο είναι το *Xcode*. Το *Xcode* είναι το μόνο IDE που δημιουργήθηκε και υποστηρίζεται επίσημα από την Apple, ενώ μπορεί να χρησιμοποιηθεί μόνο σε υπολογιστές με λειτουργικό MacOS. Εναλλακτικά, μπορεί να χρησιμοποιηθεί και το *AppCode*, το οποίο τρέχει κι αυτό μόνο σε υπολογιστές με λειτουργικό MacOS.

Απαραίτητο για την ανάπτυξη εφαρμογών είναι η εγκατάσταση του iOS SDK, το οποίο περιέχει ένα σύνολο εργαλείων και βιβλιοθηκών ανάπτυξης που επιτρέπουν στον προγραμματιστή να δημιουργήσει την εφαρμογή. Πολλές φορές, το SDK περιλαμβάνεται στο IDE που αναπτύσσεται η mobile εφαρμογή.

2.2.3 Ανάπτυξη μιας Cross-Platform εφαρμογής

Εκτός από την επιλογή δημιουργίας μιας iOS ή Android εφαρμογής, υπάρχουν εργαλεία με τα οποία ένας προγραμματιστής μπορεί να φτιάξει μια εφαρμογή, η οποία θα τρέχει στην πλειοψηφία των συσκευών με λειτουργικό Android και iOS ταυτόχρονα. Παρακάτω, θα δούμε τα πλεονεκτήματα και τα μειονεκτήματα της δημιουργίας μιας cross-platform εφαρμογής, ενώ θα δούμε επίσης τα δημοφιλέστερα εργαλεία και τις γλώσσες με τις οποίες κάποιος μπορεί να την δημιουργήσει.

Πλεονεκτήματα ανάπτυξης μιας cross-platform εφαρμογής

- Γρηγορότερη ανάπτυξη εφαρμογής, χαμηλότερο κόστος

Ο χρόνος δημιουργίας μιας cross-platform εφαρμογής είναι πολύ μικρότερος από την ξεχωριστή δημιουργία της εφαρμογής σε android και iOS, μιας και το project είναι ένα, ενώ ένα μεγάλο κομμάτι του κώδικα είναι κοινό και για τα δύο λειτουργικά συστήματα. Άρα, η διαδικασία δημιουργίας της εφαρμογής κοστίζει και λιγότερο.

- Προορίζεται για μεγαλύτερο εύρος χρηστών

Λόγω του ότι η εφαρμογή θα προορίζεται για συσκευές με λειτουργικό Android και iOS ταυτόχρονα, γίνεται σαφές πως θα λειτουργεί στην μεγαλύτερη πλειοψηφία των συσκευών. Μιας και δεν γίνεται διάκριση ανάμεσα σε Android ή iOS, οι εφαρμογές αυτές είναι διαθέσιμες προς όλους σχεδόν τους χρήστες.

- Συνοχή μεταξύ των λειτουργικών συστημάτων

Όταν δημιουργούνται 2 ξεχωριστές εφαρμογές για Android και iOS, το πιο πιθανόν είναι να υπάρχουν διαφορές μεταξύ τους, στην διεπαφή ή τις λειτουργίες, ακόμα κι αν δοθεί μεγάλη προσοχή στο κομμάτι αυτό. Αντίθετα, όταν μια εφαρμογή είναι cross-platform, λόγω του ότι το μεγαλύτερο μέρος του συνολικού κώδικα είναι κοινό, άρα οι 2 πλατφόρμες μοιράζονται κοινό κομμάτι κώδικα, μπορεί να επιτευχθεί μεγαλύτερη συνοχή μεταξύ των λειτουργικών με πολύ μικρότερη προσπάθεια.

Μειονεκτήματα ανάπτυξης μιας cross-platform εφαρμογής

- Χαμηλότερη απόδοση

Η απόδοση είναι ένα από τα πιο σημαντικά χαρακτηριστικά μιας εφαρμογής. Έχει παρατηρηθεί πως σε πολλές περιπτώσεις υπάρχει μια μικρή διαφορά στην ταχύτητα μιας εφαρμογής όταν αυτή είναι cross-platform, σε αντίθεση με την ίδια εφαρμογή αν δημιουργηθεί αποκλειστικά για ένα συγκεκριμένο λειτουργικό σύστημα.

- Δυσκολότερος σχεδιασμός της εφαρμογής και γράψιμο κώδικα

Όταν μια εφαρμογή σχεδιάζεται για πολλά λειτουργικά συστήματα, θα πρέπει να προσαρμοστεί όχι μόνο σε διαφορετικές συσκευές αλλά και σε διαφορετικές πλατφόρμες, οι οποίες είναι πολύ διαφορετικές μεταξύ τους. Έτσι, θα πρέπει ο προγραμματιστής της εφαρμογής να λάβει υπόψη του τις διαφορές μεταξύ των λειτουργικών, τις συγκεκριμένες εξαιρέσεις που θα πρέπει να χειριστεί, τα προβλήματα που μπορεί να υπάρχουν σε κάποια συγκεκριμένη πλατφόρμα και πολλά ακόμα.

- Περιορισμένες δυνατότητες

Οι δυνατότητες που μπορούν να προστεθούν σε μια cross-platform εφαρμογή είναι περιορισμένες σε σχέση με μια εφαρμογή που σχεδιάζεται για ένα συγκεκριμένο λειτουργικό σύστημα, μιας και δεν μπορεί να χρησιμοποιήσει όλες τις ενσωματωμένες λειτουργίες που παρέχονται από μια κινητή συσκευή. Αν λάβουμε υπόψη πως οι συσκευές έχουν και τις ιδιαιτερότητες τους, είναι αδύνατο να επιτευχθούν τα μέγιστα δυνατά αποτελέσματα και η μέγιστη δυνατή εμπειρία χρήσης [11-12].

Εργαλεία ανάπτυξης cross-platform εφαρμογών

Υπάρχουν αρκετά εργαλεία και περιβάλλοντα ανάπτυξης cross-platform εφαρμογών. Παρακάτω θα αναφερθούν τα πιο δημοφιλή.

Xamarin: Το Xamarin είναι μια open-source πλατφόρμα δημιουργίας εφαρμογών, σχεδιασμένη από την Microsoft. Είναι δωρεάν, ενώ υποστηρίζεται και από τα Windows και από τα MacOS. Η γλώσσα που χρησιμοποιείται για την δημιουργία εφαρμογών είναι η C#.

React Native: Το React Native είναι μια open source πλατφόρμα δημιουργίας cross-platform εφαρμογών. Είναι δωρεάν, ενώ μπορεί κάποιος να δημιουργήσει μια εφαρμογή χρησιμοποιώντας τις γλώσσες React και Javascript[13].

ΕΠΙΛΟΓΟΣ

Σ αυτό το κεφάλαιο παρουσιάστηκαν κάποιες πληροφορίες σχετικά με το λειτουργικό Android όπως η αρχιτεκτονική του και οι διαφορές που έχει σε σχέση με τα άλλα λειτουργικά συστήματα για κινητά τηλέφωνα, καθώς επίσης και οι τρόποι δημιουργίας εφαρμογών για διάφορα λειτουργικά συστήματα κινητών τηλεφώνων. Στο επόμενο κεφάλαιο θα αναφερθούν οι λόγοι που οδήγησαν στην επιλογή του λειτουργικού Android ως κατάλληλο για την ανάπτυξη της εφαρμογής. Επίσης, θα αναλυθούν οι τρόποι, τα μέσα, οι βιβλιοθήκες, οι διαδικασίες σχεδιασμού και υλοποίησης της εφαρμογής του μαθητικού ραδιοφώνου και οτιδήποτε άλλο έχει σχέση με αυτή.

ΚΕΦΑΛΑΙΟ 3 – Android Εφαρμογή Ραδιοφώνου

Σ αυτό το κεφάλαιο θα αναφερθούν οι λόγοι που οδήγησαν στην επιλογή του λειτουργικού Android για την ανάπτυξη της εφαρμογής, θα περιγραφούν τα χαρακτηριστικά και οι απαιτήσεις που πρέπει να έχει η εφαρμογή αλλά κυρίως θα αναλυθούν λεπτομερώς οι τρόποι, οι διαδικασίες, οι βιβλιοθήκες, τα εργαλεία αλλά και οτιδήποτε άλλο χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής.

3.1 Γιατί επιλέχθηκε το λειτουργικό Android για την εφαρμογή

Υπάρχουν αρκετοί λόγοι που οδήγησαν στην απόφαση να δημιουργηθεί η εφαρμογή του μαθητικού ραδιοφώνου για κινητά με λειτουργικό Android. Πρώτα απ' όλα, όπως αναφέρθηκε και παραπάνω, το Android καλύπτει πάνω από το 70% της αγοράς των έξυπνων τηλεφώνων [14]. Έτσι, γίνεται σαφές πως μια εφαρμογή Android θα καλύπτει πολύ μεγαλύτερο αριθμό χρηστών έξυπνων τηλεφώνων.

Όσο αφορά την δημιουργία ενός λογαριασμό προγραμματιστή (developer account), προκειμένου να είναι δυνατή η δημοσίευση μιας εφαρμογής στο Google Play για το Android και στο App store για το iOS αντίστοιχα, ισχύουν διαφορετικοί όροι, διαδικασίες αλλά και τιμές. Όσο αφορά το Android, το κόστος δημιουργίας ενός developer account είναι μόλις 25 δολάρια, τα οποία ισχύουν εφ' όρου ζωής. Άρα, με μόλις 25 ευρώ ένας προγραμματιστής μπορεί να ανεβάσει όσες εφαρμογές θέλει στο Play Store για πάντα[15]. Όμως, στο App Store της Apple τα πράγματα είναι διαφορετικά. Η απλή συνδρομή για την δημιουργία ενός developer account κοστίζει 99 δολάρια και ισχύει για ένα χρόνο. Έτσι, βλέπουμε ότι το κόστος είναι σαφώς πολύ μεγαλύτερο.

Επιπλέον, το Android είναι ένα ανοιχτό λειτουργικό, με πολύ λιγότερο αυστηρούς κανόνες και προδιαγραφές ασφαλείας. Επιτρέπει την εγκατάσταση εφαρμογών από άλλα markets εκτός του Google Play Store και γενικά από άγνωστες πηγές. Από την μια αυτό είναι σίγουρα λιγότερο ασφαλές, μιας και η εγκατάσταση άγνωστων εφαρμογών μπορεί να μην είναι τόσο ακίνδυνη, ωστόσο αυτό προσφέρει μεγάλη ελευθερία και ευελιξία στον χρήστη.

Αντίθετα, όσο αφορά το λειτουργικό iOS, είναι ένα κλειστό λειτουργικό σύστημα με ιδιαίτερα αυστηρές προδιαγραφές και κανόνες ασφαλείας. Δεν επιτρέπει την εγκατάσταση εφαρμογών από markets εκτός του App Store ή άγνωστων πηγών (τουλάχιστον με νόμιμο τρόπο). Άρα, η μόνη πηγή λήψης είναι το επίσημο market της Apple.

Άρα, θεωρητικά, μια εφαρμογή Android μπορεί να διαδοθεί και να γίνει διαθέσιμη προς τους χρήστες ακόμα και χωρίς χρήματα ή κάποια συνδρομή, ανεβάζοντας το σε ένα τρίτο android market ή διαμοιράζοντας το αρχείο apk (το αρχείο που χρειάζεται ένας χρήστης android για να κάνει εγκατάσταση μιας εφαρμογής). Αυτό όμως σίγουρα δεν ισχύει με μια εφαρμογή iOS, που απαιτείται η δημοσίευση της στο επίσημο market της Apple. Έτσι, βλέπουμε πως το κόστος αλλά και η αυστηρότητα των κανόνων ασφαλείας λειτουργεί ως περιοριστικός παράγοντας στην επιλογή ενός προγραμματιστή να δημιουργήσει μια εφαρμογή σε iOS.

3.2 Καταγραφή χαρακτηριστικών και απαιτήσεων εφαρμογής

Ξεκινώντας την διαδικασία σχεδιασμού και υλοποίησης της εφαρμογής του μαθητικού ραδιοφώνου, έπρεπε να καταγραφούν οι αρχικές απαιτήσεις και τα χαρακτηριστικά της εφαρμογής, με βάση τις ανάγκες και τις επιθυμίες των χρηστών, όπως αναφέρονται παρακάτω.

Μοντέρνα εμφάνιση και μενού περιήγησης – εύχρηστο περιβάλλον

Η εφαρμογή του ραδιοφώνου θα προορίζεται για χρήση από άτομα όλων των ηλικιών: από παιδιά νηπιαγωγείου μέχρι δασκάλους-καθηγητές. Έτσι, θα πρέπει το περιβάλλον της εφαρμογής να είναι απλό, μοντέρνο, αλλά και εύχρηστο, ώστε να μπορεί να χρησιμοποιείται ακόμα και από άτομα χωρίς ιδιαίτερες γνώσεις τεχνολογίας, παρέχοντας σε όλους τους χρήστες μια ευχάριστη εμπειρία χρήσης.

Ζωντανή ακρόαση του ραδιοφώνου – επικοινωνία ακροατών-παραγωγών

Το πιο σημαντικό κομμάτι της εφαρμογής είναι η ζωντανή ακρόαση του ραδιοφωνικού προγράμματος 24 ώρες την ημέρα και 7 μέρες την εβδομάδα. Επίσης, θα πρέπει να εμφανίζεται στην οθόνη η εκπομπή ή το τραγούδι που παίζει εκείνη την ώρα, καθώς και η ζωντανή συνομιλία που εμφανίζεται στην ιστοσελίδα, ώστε να μπορούν οι ακροατές να επικοινωνούν με τους παραγωγούς της εκπομπής αλλά και μεταξύ τους. Τέλος, μιας και υπάρχει η δυνατότητα από το ραδιόφωνο να γίνει live streaming σε 2 διαφορετικές ποιότητες (μια χαμηλή και μια υψηλή), θα ήταν χρήσιμο να υπάρχουν και οι 2 αυτές επιλογές στην εφαρμογή.

Ραδιοφωνικό Πρόγραμμα & Επόμενες Εκπομπές

Στην εφαρμογή θα πρέπει να εμφανίζονται με ωραίο και εύχρηστο τρόπο το ραδιοφωνικό πρόγραμμα και οι επόμενες εκπομπές που ακολουθούν. Όσο αφορά το ραδιοφωνικό πρόγραμμα, θα πρέπει ο χρήστης να μπορεί εύκολα, επιλέγοντας την μέρα ή τον μήνα που επιθυμεί, να δει τις εκπομπές που πρόκειται να παίξουν ή που έχουν παίζει ήδη, αλλά και αν επιθυμεί να βλέπει περισσότερες πληροφορίες για την εκπομπή που επιθυμεί, όπως ακριβώς γίνεται και στην ιστοσελίδα.

Νέα του ραδιοφώνου

Ακόμα μια σημαντική λειτουργία που θα πρέπει να διαθέτει η εφαρμογή είναι η εύκολη πρόσβαση στα νέα και τις ανακοινώσεις του ραδιοφώνου. Εκεί ανακοινώνονται όλα τα νέα για το πρόγραμμα του ραδιοφώνου, διαγωνισμοί και πολλά ακόμα.

Αναζήτηση και ακρόαση εκπομπών μέσα από το αρχείο του ραδιοφώνου

Ένα σημαντικό κομμάτι του ραδιοφώνου είναι το αρχείο εκπομπών. Εκεί, τα μέλη του ραδιοφώνου ανεβάζουν τις ζωντανές ή ηχογραφημένες εκπομπές τους ώστε να μπορούν οι ακροατές να τις ακούν όποτε επιθυμούν. Θα ήταν πολύ χρήσιμο να μπορεί κάποιος μέσα από την εφαρμογή να αναζητεί και να ακούει μια εκπομπή που επιθυμεί.

Πολύγλωσσο περιβάλλον

Στο European School Radio υπάρχουν χρήστες όχι μόνο από την Ελλάδα αλλά και από όλο τον κόσμο. Έτσι, προκειμένου να απευθύνεται σε μεγαλύτερο αριθμό χρηστών, θα ήταν καλό εκτός από Ελληνικά να υποστηρίζει κι άλλες γλώσσες. Τώρα, η εφαρμογή υποστηρίζει Αγγλικά και Ελληνικά, ενώ υπάρχει δυνατότητα μετάφρασης σε περισσότερες γλώσσες, εφόσον υπάρξει τέτοια ανάγκη.

3.3 Λογισμικά-Εργαλεία ανάπτυξης της εφαρμογής

Για την ανάπτυξη της εφαρμογής χρησιμοποιήθηκαν αρκετά προγράμματα, καθένα για διαφορετικό σκοπό. Τα βασικότερα προγράμματα που χρησιμοποιήθηκαν είναι:

➤ Android Studio

Το Android Studio είναι το πρόγραμμα ανάπτυξης της εφαρμογής. Προτιμήθηκε αντί άλλων όπως το Eclipse καθώς είναι το δημοφιλέστερο και πληρέστερο περιβάλλον ανάπτυξης Android εφαρμογών, σχεδιασμένο και υποστηριζόμενο από την Google. Παρέχει χαρακτηριστικά όπως:

- Μοντέρνα και σύγχρονη διεπαφή, με συχνές ενημερώσεις και βελτιώσεις
- Γρήγορο και πλούσιο σε λειτουργίες εξομοιωτή (emulator)
- Εργαλεία μεγιστοποίησης απόδοσης, χρηστικότητας και επίλυσης προβλημάτων συμβατότητας εκδόσεων Android
- Γρήγορη και ευέλικτη συγγραφή κώδικα [16]

➤ Atom

Το Atom είναι ένας δωρεάν, open-source επεξεργαστής κειμένου με πλούσιες λειτουργίες. Υποστηρίζει τις περισσότερες γλώσσες προγραμματισμού, ενώ έχει χαρακτηριστικά που κάνουν ευκολότερη την συγγραφή κώδικα καθώς και την απασφαλμάτωση. Χρησιμοποιήθηκε ιδιαίτερα για την δημιουργία ρηρ αρχείων για διασυνδέσεις της βάσης με την εφαρμογή. Περισσότερες λεπτομέρειες θα αναφερθούν παρακάτω.

➤ Postman

Το Postman είναι ένα εργαλείο δημιουργίας, τροποποίησης και ελέγχου APIs (Application Programming Interface). Χρησιμοποιήθηκε κατά την δημιουργία διασυνδέσεων μεταξύ της βάσης δεδομένων και την εφαρμογής για έλεγχο των αρχείων και των json. Περισσότερες λεπτομέρειες θα αναφερθούν παρακάτω.

➤ phpMyAdmin/DBeaver

Για την διαχείριση της βάσης δεδομένων, ώστε να δημιουργηθούν τα κατάλληλα SQL Queries με τα οποία θα παίρνονται δεδομένα για την εφαρμογή, χρησιμοποιήθηκαν το phpMyAdmin που είναι μια διαδικτυακή εφαρμογή διαχείρισης βάσης δεδομένων και το DBeaver, που είναι ένα open-source εργαλείο για τον ίδιο σκοπό.

➤ WP REST API v2.0

Για την επιλογή και διαχείριση δεδομένων της ιστοσελίδας του μαθητικού ραδιοφώνου, όπως είναι τα posts, χρησιμοποιήθηκε το WP REST API v2.0. Περισσότερες πληροφορίες γι' αυτό θα αναλυθούν παρακάτω.

➤ Filezilla

Για την μεταφόρτωση php αρχείων στον server, επιλέχτηκε ο FTP Client Filezilla.

➤ Google Chrome

Για αναζήτηση πληροφοριών, επίλυσης προβλημάτων αλλά και οτιδήποτε άλλο χρειάστηκε για αναζήτηση στο διαδίκτυο, προτιμήθηκε ο browser Google Chrome.

3.4 Υλοποίηση εφαρμογής Android

Παρακάτω θα αναλυθεί η διαδικασία, ο τρόπος αλλά και σημαντικά σημεία του αλγορίθμου που αναπτύχθηκε η εφαρμογή του μαθητικού ραδιοφώνου.

3.4.1 Άδειες-Permission

Κατά την ανάπτυξη μιας εφαρμογής Android, είναι πολύ πιθανόν να χρειάζεται να συμπεριληφθούν κάποιες άδειες στην εφαρμογή. Οι άδειες αυτές προστατεύουν τον χρήστη και τις απόρρητες πληροφορίες που μπορεί να έχει. Οι εφαρμογές, πρέπει να ζητήσουν άδεια για να αποκτήσουν πρόσβαση σε ευαίσθητα δεδομένα και λειτουργίες του χρήστη όπως είναι η πρόσβαση στις επαφές, SMS, κάμερα, διαδίκτυο κτλ. Ανάλογα με την λειτουργία, ενδέχεται να ζητηθεί από τον χρήστη να αποδεχτεί τις άδειες ή να εκχωρηθούν αυτόματα. Η μη αποδοχή των αδειών μπορεί να προκαλέσει την μη ορθή λειτουργία της εφαρμογής.

Ένα σημαντικό στοιχείο των προδιαγραφών ασφαλείας του Android είναι ότι καμία εφαρμογή από προεπιλογή δεν έχει άδεια να εκτελεί λειτουργίες που θα μπορούσαν να επηρεάσουν ή να βλάψουν άλλες εφαρμογές, το λειτουργικό σύστημα ή τον χρήστη, μέχρι να τις προσθέσει ο προγραμματιστής. Για παράδειγμα, η πρόσβαση και το δικαίωμα εγγραφής στις επαφές, στα μηνύματα ηλεκτρονικού ταχυδρομείου ή στην ανάγνωση άλλων αρχείων της συσκευής, είναι λειτουργίες που θα μπορούσαν να βλάψουν τον χρήστη, οπότε απαιτείται η ρητή συγκατάθεσή του[17].

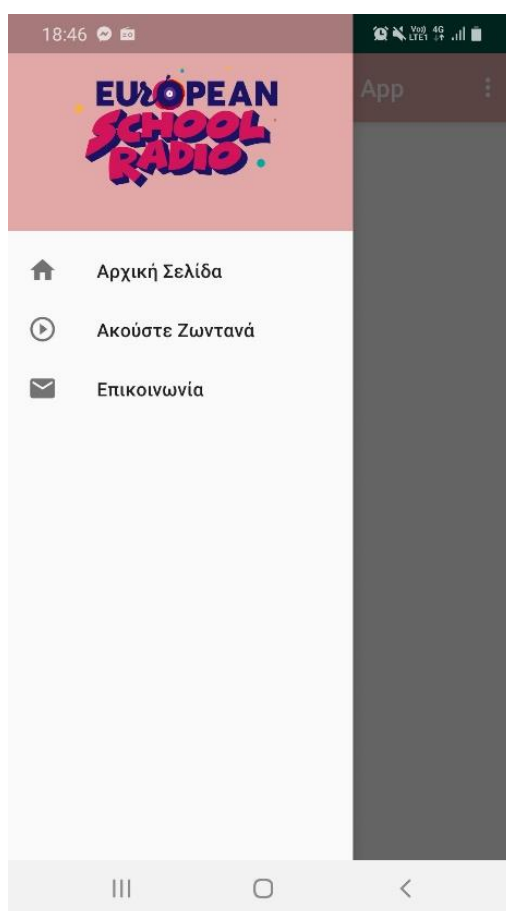
3.4.2 Επικοινωνίας βάσης δεδομένων - εφαρμογής

Για την λειτουργία της εφαρμογής, είναι απαραίτητη η λήψη πληροφοριών από την βάση δεδομένων του European School Radio. Συγκεκριμένα, χρειάστηκε η λήψη πληροφοριών όπως ημερομηνίες ακρόασης εκπομπών στο ραδιόφωνο, ονόματα και περιγραφές εκπομπών, ηχητικά αρχεία, εικόνες και πολλά ακόμα. Για την μεταφορά λοιπόν αυτών των δεδομένων δημιουργήθηκαν αρχεία php στον server του European School Radio, όπου μέσω ερωτημάτων SQL, εξάγουν συγκεκριμένες πληροφορίες σε μορφή json. Η εξαγωγή πληροφοριών σε json και η ενσωμάτωσή τους σε εφαρμογές android είναι ένας συνηθισμένος τρόπος λήψης δεδομένων από μια βάση δεδομένων. Συγκεκριμένα, εκτελείται ένα ερώτημα SQL, όπου τα αποτελέσματα του αποθηκεύονται σε έναν πίνακα ενώ στην συνέχεια γίνεται κωδικοποίηση σε json με την συνάρτηση json_encode(). Σε ορισμένες περιπτώσεις βέβαια, δεν αρκούσε να γίνει απλά μετατροπή των δεδομένων σε json.

Χρειαζόταν να γίνει επεξεργασία των δεδομένων, όπως στην περίπτωση του ραδιοφωνικού προγράμματος. Αναλυτικότερα θα περιγραφούν οι διαδικασίες αυτές παρακάτω.

3.4.3 Αρχική οθόνη – Μενού Πλοήγησης

Όσο αφορά την αρχική οθόνη - το μενού της εφαρμογής, δόθηκε ιδιαίτερη σημασία στην εμφάνιση αλλά και στην ευχρηστία. Η πρώτη εικόνα που βλέπει ο χρήστης μόλις μπαίνει στην εφαρμογή είναι σημαντική και αποτελεί καθοριστικό παράγοντα στην συνέχιση της πλοήγησης του σε αυτή. Οι επιλογές που υπήρχαν για την δημιουργία του μενού ήταν δυο: η δημιουργία ενός μενού με βάση τις επιλογές και δυνατότητες που δίνει το Android Studio και η δημιουργία ενός custom μενού. Παρακάτω (εικόνα 7) φαίνεται ένα δείγμα ενός μενού στα πρότυπα του Android, δημιουργημένο με εργαλεία του Android Studio. Συγκεκριμένα, είναι ένα Navigation Drawer, δηλαδή ένα μενού που ανοίγει και κλείνει στο αριστερό μέρος μιας οθόνης, παρέχοντας τις επιλογές-λειτουργίες μιας εφαρμογής.



Εικόνα 7 "Δοκιμαστικό Μενού"

Αυτό το πρότυπο μενού χρησιμοποιείται από πάρα πολλές mobile εφαρμογές και είναι αρκετά εύχρηστο και συνηθισμένο. Ωστόσο, στην εφαρμογή του European School radio, οι κύριες λειτουργίες που θα περιέχει το μενού δεν είναι πάρα πολλές, και υπήρχε φόβος αυτό το μενού να δείχνει "άδειο", αλλά κυρίως ένα τέτοιο μενού ίσως σε κάποιους μοιάζει απαρχαιωμένο. Έτσι, έγινε προσπάθεια να δημιουργηθεί ένα custom μενού, το οποίο θα είναι πιο μοντέρνο, καινοτόμο αλλά ταυτόχρονα και εύχρηστο για τον χρήστη. Μετά από πολλές δοκιμές και προσπάθειες, το μενού που

προτιμήθηκε φαίνεται παρακάτω (εικόνα 8) και περιλαμβάνει τις πιο συνηθισμένες λειτουργίες του ραδιοφώνου.



Εικόνα 8 "Τελικό Μενού Εφαρμογής"

Όσο αφορά τον τρόπο υλοποίησης του μενού, αρχικά φαίνεται το επίσημο λογότυπο του European School Radio (δημιουργημένο ως ImageView). Στην συνέχεια, εμφανίζονται οι επιλογές που έχει ο χρήστης, μέσω κάποιων «κουμπιών». Πιο συγκεκριμένα, δεν είναι απλά κουμπιά, αλλά Cardviews που περιέχονται μέσα σ' ένα GridLayout 2 στηλών. Το μόνο που απαιτείται για την δημιουργία ενός CardView σε μία διεπαφή, είναι να γίνει εισαγωγή στο αρχείο build.gradle, στα dependencies. Εσωτερικά έχουν ένα LinearLayout, όπου εσωτερικά του LinearLayout βρίσκεται το ImageView της κάθε λειτουργίας, δηλαδή η εικόνα, και το TextView, δηλαδή το κείμενο που υπάρχει από κάτω. Έτσι, όταν ο χρήστης πατήσει πάνω σ ένα LinearLayout (σ ένα κουμπί), ανοίγει το αντίστοιχο Activity, δηλαδή το περιεχόμενο της επιλογής που πάτησε. Για το άνοιγμα του νέου Activity, μόλις πατηθεί μια επιλογή καλείται η μέθοδος OnClickListener() και χρησιμοποιείται η κλάση Intent με την μέθοδος startActivity(), ενώ κατά την πρώτη είσοδο του χρήστη στην εφαρμογή εμφανίζεται ένα pop-up (AlertDialog για την ακρίβεια) που τον καλωσορίζει στη εφαρμογή. Γι' αυτό το AlertDialog, χρησιμοποιείται ένα flag όπου ελέγχεται εάν έχει ξαναεμφανιστεί στο παρελθόν. Εάν έχει εμφανιστεί, αυτή η ενέργεια αποθηκεύεται χρησιμοποιώντας την κλάση SharedPreferences και έτσι δεν ξαναεμφανίζεται.

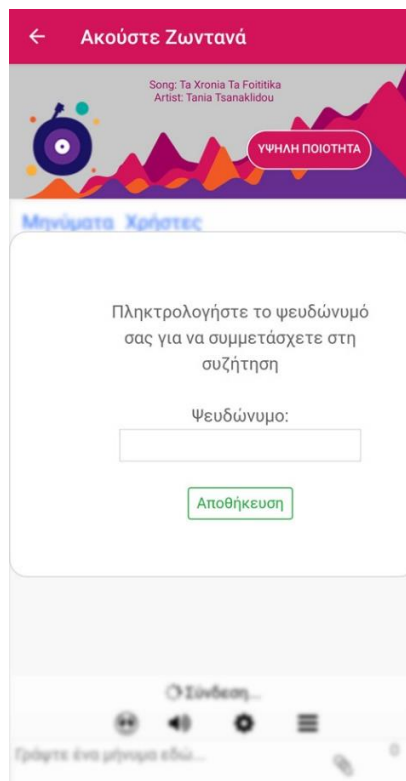
Επίσης, μιας και για οποιαδήποτε λειτουργία της εφαρμογής απαιτείται σύνδεση στο διαδίκτυο είτε μέσω Wi-Fi είτε μέσω δεδομένων κινητής τηλεφωνίας, υπάρχει ένας handler, όπου κάθε λίγα δευτερόλεπτα ελέγχει εάν υπάρχει ενεργή σύνδεση στο διαδίκτυο. Εάν δεν υπάρχει, εμφανίζεται κατάλληλο μήνυμα στον χρήστη και παράλληλα απενεργοποιούνται τα κουμπιά, ώστε να μην μπορεί να τα πατήσει μέχρι να αποκτήσει ενεργή σύνδεση.

Τέλος, είναι σημαντικό να τονιστεί ότι σε όλη την εφαρμογή τα χρώματα αλλά και η γραμματοσειρά που χρησιμοποιήθηκαν είναι σχετικά με τα χρώματα και τις γραμματοσειρές που υπάρχουν ήδη στην ιστοσελίδα του European School Radio.

3.4.4 Ακούστε Ζωντανά

Αυτό είναι ίσως το σημαντικότερο κομμάτι της εφαρμογής, ενώ προβλέπεται να είναι και το πιο δημοφιλές. Εδώ, οι ακροατές του μαθητικού ραδιοφώνου θα μπορούν να ακούν τις αγαπημένες τους εκπομπές 24 ώρες το 24ωρο. Γι' αυτό τον λόγο, έπρεπε να δοθεί και εδώ μεγάλη σημασία στην εμφάνιση αλλά και στην ευχρηστία αυτής της λειτουργίας.

Τα στοιχεία που περιέχει η λειτουργία «Ακούστε Ζωντανά» είναι η αναπαραγωγή/διακοπή του ήχου, η εμφάνιση της εκπομπής ή του τραγουδιού που παίζει τώρα, η ζωντανή συνομιλία που υπάρχει και στην ιστοσελίδα του ραδιοφώνου και ένα κουμπί αλλαγής ποιότητας, όπως φαίνονται και στην παρακάτω εικόνα (εικόνα 9).



Εικόνα 9 "Ακούστε Ζωντανά"

Όσο αφορά το κουμπί αναπαραγωγή/διακοπή, προτιμήθηκε να μην μπει το συνηθισμένο κουμπί με το σύμβολο της αναπαραγωγής και διακοπής, αλλά ένα βινύλιο όπου όταν το πατάει ο χρήστης θα ξεκινάει η αναπαραγωγή. Αυτό το βινύλιο είναι και το σήμα του European School Radio.

Επίσης, όταν παίζει (δηλαδή όταν είναι σε λειτουργία αναπαραγωγής), το βινύλιο περιστρέφεται συνεχώς, ώστε να δίνει την εικόνα ότι αναπαράγεται ήχος. Για την δημιουργία αυτού του κουμπιού, χρησιμοποιήθηκε ένα ImageButton, ενώ για την περιστροφή χρησιμοποιήθηκε η κλάση Animation και το αρχείο rotatebutton.xml, που περιλαμβάνει κάποιες παραμέτρους με τις οποίες θα γίνεται η περιστροφή, όπως οι μοίρες που θα περιστρέφεται, η διάρκεια περιστροφής κτλ.

Δεξιότερα του κουμπιού αναπαραγωγής/διακοπής, βρίσκεται το κουμπί αλλαγής ποιότητας. Πιο συγκεκριμένα, αυτό το κουμπί δίνει την δυνατότητα αναπαραγωγής της ραδιοφωνικής ροής σε υψηλότερη ή χαμηλότερη ποιότητα, με σκοπό την κατανάλωση μικρότερου όγκου δεδομένων που χρησιμοποιείται για την αναπαραγωγή. Ειδικά στην περίπτωση που ο χρήστης είναι συνδεδεμένος με δεδομένα κινητής τηλεφωνίας και όχι με Wi-fi, κάτι τέτοιο είναι ιδιαίτερα χρήσιμο, μιας και ο όγκος δεδομένων που έχει στην διάθεσή του για κατανάλωση δεν είναι απεριόριστος, άρα η αναπαραγωγή σε χαμηλότερη ποιότητα θα βοηθήσει πολύ στην εξοικονόμηση.

Μια έξυπνη λειτουργία που διαθέτει η εφαρμογή είναι ότι ανιχνεύει τον τρόπο με τον οποίο ο χρήστης είναι συνδεδεμένος (αν είναι συνδεδεμένος μέσω Wi-fi ή μέσω δεδομένων κινητής τηλεφωνίας). Όταν είναι συνδεδεμένος με Wi-fi, με το που μπει στην ενότητα «Ακούστε Ζωντανά» είναι επιλεγμένη η υψηλή ποιότητα. Όταν είναι συνδεδεμένος με δεδομένα κινητής τηλεφωνίας είναι επιλεγμένη η χαμηλή ποιότητα. Φυσικά, ανά πάσα στιγμή έχει την δυνατότητα αλλαγής της ποιότητας από υψηλή σε χαμηλή και το αντίστροφο πατώντας το κουμπί. Για τον έλεγχο του τρόπου σύνδεσης στο διαδίκτυο χρησιμοποιείται η κλάση ConnectivityManager και η android.net.NetworkInfo. Επίσης, έχει ελεγχθεί και προβλεφθεί η συμπεριφορά της λειτουργίας «Ακούστε Ζωντανά» σε καταστάσεις όπως η ξαφνική εισερχόμενη τηλεφωνική κλήση ενώ είναι σε εξέλιξη η αναπαραγωγή του ραδιοφώνου. Ανάλογα με την κατάσταση που είναι η τηλεφωνική κλήση, η εφαρμογή ενεργοποιεί ή απενεργοποιεί τον ήχο της εφαρμογής. Αυτός ο έλεγχος γίνεται με την κλάση PhoneStateListener.

Στο επάνω μέρος της σελίδας, φαίνεται το τραγούδι ή η εκπομπή που παίζει εκείνη την στιγμή. Πρόκειται για ένα απλό TextView, που όμως έχει υλοποιηθεί με html parser. Ο html parser που προτιμήθηκε είναι η βιβλιοθήκη **Jsoup**. Η Jsoup είναι μια βιβλιοθήκη java που χρησιμοποιείται για την επιλογή του περιεχομένου μιας σελίδας στο διαδίκτυο, για την επιλογή μιας συγκεκριμένης κλάσης μέσα σε μια ιστοσελίδα κτλ. Για την χρήση αυτή την βιβλιοθήκης, το μόνο που χρειάζεται είναι η δήλωση και η εκτέλεση της στο αρχείο build.gradle. Με αυτό τον τρόπο, επιλέχθηκε μέσα από ένα αρχείο που διαθέτει το European School Radio που αναφέρει τον καλλιτέχνη και το όνομα του τραγουδιού που παίζει εκείνη την στιγμή, το περιεχόμενο του και εμφανίζεται στο TextView, ενώ χρησιμοποιώντας έναν handler αλλάζει δυναμικά το περιεχόμενο ανά κάποια δευτερόλεπτα, εμφανίζοντας τις πιο πρόσφατες πληροφορίες.

Επίσης, σε περίπτωση που ξαφνικά διακοπεί η σύνδεση στο διαδίκτυο, απενεργοποιούνται τα κουμπιά αναπαραγωγής/διακοπής αλλά και της αλλαγής ποιότητας και βγαίνει ένα μήνυμα στον χρήστη ότι δεν έχει ενεργή σύνδεση στο διαδίκτυο. Μόλις αποκατασταθεί η σύνδεση, ενεργοποιούνται πάλι τα κουμπιά. Με την χρήση ενός handler ελέγχεται ανά κάποια δευτερόλεπτα η ύπαρξη ή όχι ενεργής σύνδεσης στο διαδίκτυο.

Όταν είναι σε εξέλιξη η αναπαραγωγή του ήχου στην εφαρμογή, πηγαίνοντας πίσω στο μενού δίνεται η επιλογή στον χρήστη να συνεχίσει να ακούει ή όχι, μέσω ενός pop-up που εμφανίζεται με τις επιλογές ναι και όχι (AlertDialog συγκεκριμένα). Εάν συνεχίσει να ακούει, εμφανίζεται ένα notification bar που περιέχει το όνομα του καλλιτέχνη και τον τίτλο του τραγουδιού (και ανανεώνεται αυτόματα), όπου εάν πατηθεί από τον χρήστη τον πηγαίνει στην εφαρμογή και συγκεκριμένα στην λειτουργία «Ακούστε Ζωντανά».

Τέλος, στο κάτω μέρος εμφανίζεται η ζωντανή συνομιλία που υπάρχει και στην ιστοσελίδα του ραδιοφώνου. Μιας και οτιδήποτε γράφεται είτε από την εφαρμογή είτε από έναν υπολογιστή στην συνομιλία θα πρέπει να εμφανίζεται ταυτόχρονα παντού (και στην συνομιλία της ιστοσελίδας και της εφαρμογής), ο μόνος τρόπος που μπορούσε να υλοποιηθεί είναι με WebView. Το WebView είναι ένα στοιχείο που σου επιτρέπει να εμφανίζεις και να χρησιμοποιείς το περιεχόμενο μιας ιστοσελίδας. Έτσι, στο WebView, δηλώνεται το URL μιας ιστοσελίδας και εκεί φορτώνεται το περιεχόμενό της.

Σ αυτό το σημείο αξίζει να αναφερθεί και η βιβλιοθήκη που χρησιμοποιήθηκε για το live streaming του ήχου. Αρχικά, χρησιμοποιήθηκε η βιβλιοθήκη MediaPlayer, γνωστή βιβλιοθήκη για αναπαραγωγή ήχου και βίντεο στις συσκευές Android. Είναι μια απλή βιβλιοθήκη και συστήνεται σε περιπτώσεις απλής αναπαραγωγής ήχου ή βίντεο μέσα από ένα αρχείο ή ένα url. Όμως, σε περιπτώσεις streaming, η πλειοψηφία των προγραμματιστών που είχαν δημιουργήσει εφαρμογές με παρόμοιες υπηρεσίες σύστηναν την βιβλιοθήκη Exoplayer.

Η **Exoplayer** είναι μια open-source βιβλιοθήκη για αναπαραγωγή ήχου ή βίντεο. Είναι μια εναλλακτική επιλογή του MediaPlayer API, με περισσότερες δυνατότητες, ενώ είναι εύκολη στην παραμετροποίηση. Πολύ γνωστές εφαρμογές πολυμέσων όπως είναι το YouTube App χρησιμοποιούν αυτή την βιβλιοθήκη. Η Exoplayer υποστηρίζει λειτουργίες όπως η Δυναμική προσαρμοστική ροή μέσω HTTP (Dynamic adaptive streaming over HTTP-DASH), SmoothStreaming και Common Encryption, οι οποίες δεν υποστηρίζονται από το MediaPlayer[18].

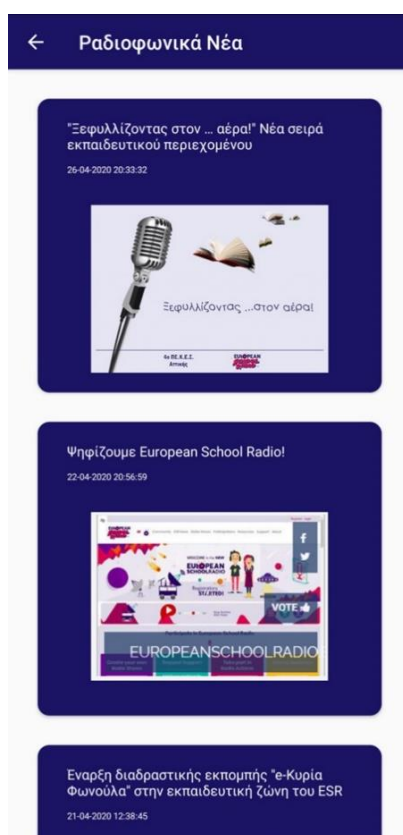
Οι κλάσεις που χρησιμοποιήθηκαν για την ζωντανή μετάδοση του ήχου ήταν η DefaultRenderersFactory, η DefaultTrackSelector, η ExoPlayerFactory, η DefaultDataSourceFactory, η DefaultExtractorsFactory, η ExtractorMediaSource κτλ. Αυτές οι κλάσεις που αφορούν την δημιουργία, την επεξεργασία και την αναπαραγωγή ηχητικών αρχείων από URL.

Ένα πρόβλημα που προέκυψε ήταν ότι σε καινούργιες συσκευές που είχαν τις τελευταίες εκδόσεις Android (από Android 9 και πάνω) δεν έπαιζε ο ήχος, ενώ όλα φαινόταν να είχαν γίνει κανονικά. Το πρόβλημα αυτό, οφειλόταν στο ότι οι συσκευές με API level 27 ή λιγότερο (Android 8.1 και κάτω), είχαν την ιδιότητα android:usesCleartextTraffic true, ενώ από την έκδοση 9, η προεπιλεγμένη τιμή γι' αυτή την ιδιότητα είναι false. Το android:usesCleartextTraffic υποδεικνύει εάν η εφαρμογή σκοπεύει να χρησιμοποιήσει ανταλλαγή δεδομένων, όπως πληροφορίες με http. Το πρόβλημα λύθηκε πηγαίνοντας στο AndroidManifest.xml και βάζοντας την ιδιότητα android:usesCleartextTraffic="true".

3.4.5 Νέα

Αυτή η λειτουργία παρέχει εύκολη πρόσβαση στα νέα του ραδιοφώνου. Έτσι, ο χρήστης μένει ενημερωμένος για οτιδήποτε τον αφορά, από εκδηλώσεις και εκπομπές μέχρι διαγωνισμούς κτλ.

Κάθε νέο περιλαμβάνεται σε ένα ξεχωριστό CardView. Κάθε CardView περιέχει τον τίτλο του νέου, την ημερομηνία δημοσίευσης και την εικόνα του, όπως φαίνεται και παρακάτω. Ο τίτλος και η ημερομηνία έχουν δημιουργηθεί με TextView ενώ η εικόνα είναι δημιουργημένη με την βιβλιοθήκη `pl.droidsonroids.gif.GifImageView`. Για την χρήση αυτής της βιβλιοθήκης απαιτείται απλά να δηλωθεί στο αρχείο `build.gradle`. Δεν μπορούσε να χρησιμοποιηθεί απλό `ImageView`, μιας και το απλό `ImageView` δίνει την δυνατότητα εμφάνισης μιας στατικής εικόνας, ενώ εδώ όταν ο χρήστης μπαίνει σ αυτή την λειτουργία, στην θέση των εικόνων εμφανίζεται ένα loading gif και στην συνέχεια αντικαθίσταται από την εικόνα.



Εικόνα 10 "Ραδιοφωνικά Νέα"

Στην συνέχεια, ο χρήστης μπορεί να πατήσει επάνω στο νέο που επιθυμεί για να δει περισσότερες πληροφορίες. Όταν πατήσει πάνω σε κάποιο νέο, η εικόνα που του εμφανίζεται είναι όπως στην εικόνα παρακάτω (εικόνα 11). Ο τίτλος καθώς και το κείμενο, έχουν δημιουργηθεί με `TextView`, η εικόνα με `ImageView` ενώ η μπάρα αναπαραγωγής του σποτ με `PlayerView`, ένα στοιχείο της βιβλιοθήκης `Exoplayer`. Τέλος, κάτω από κάθε νέο υπάρχει ένα button που δίνει την δυνατότητα στον χρήστη να ανοίξει το νέο που διαβάζει στον browser, και συγκεκριμένα στην αντίστοιχη σελίδα του European School Radio.



Εικόνα 11 "Λεπτομέρειες Νέων"

Για να ληφθούν τα δεδομένα που απαιτούνται (τίτλος νέου, περιεχόμενο νέου, εικόνα κτλ) χρησιμοποιήθηκε το **WP REST API v2.0**. Το WP REST API v2.0 παρέχει πρόσβαση στις δημοσιεύσεις, σελίδες, ετικέτες, κατηγορίες κτλ μιας wordpress ιστοσελίδας. Έτσι, πληκτρολογώντας ένα URL, ένας χρήστης μπορεί να αποκτήσει πρόσβαση στις πληροφορίες μιας wp ιστοσελίδας που επιθυμεί. Παρακάτω, φαίνονται κάποια παραδείγματα URL που χρησιμοποιούνται για πρόσβαση στις δημοσιεύσεις, στις σελίδες και στις κατηγορίες:

- domain-name.com/wp-json/wp/v2/posts
- domain-name.com/wp-json/wp/v2/pages
- domain-name.com/wp-json/wp/v2/categories

όπου domain-name.com το όνομα μιας ιστοσελίδας wordpress. Η μορφή που γίνονται εξαγωγή όλες αυτές οι πληροφορίες είναι σε json. Έτσι, μπορούμε να πάρουμε αυτές τις πληροφορίες και να τις χρησιμοποιήσουμε οπουδήποτε θέλουμε, αρκεί να κάνουμε ένα απλό parse το json. Επιπλέον, το WP REST API v2.0 επιτρέπει να φιλτράρεις τις πληροφορίες που επιλέγεις να πάρεις. Για παράδειγμα, εάν στο URL που αφορά τα posts προστεθεί το request `"?filter[orderby]=date&order=desc"`, γίνεται φθίνουσα ταξινόμηση με βάση την ημερομηνία δημοσίευσης. Επίσης, το URL `/wp-json/wp/v2/posts/<id>` όπου `<id>` το id της δημοσίευσης (post), εμφανίζει τις πληροφορίες της συγκεκριμένης δημοσίευσης.

Έτσι, για την λειτουργία «Νέα» έγιναν parse, δηλαδή λήψη και επεξεργασία των δεδομένων του json. Με τις κλάσεις JSONArray, JSONObject και χρησιμοποιώντας την βιβλιοθήκη Volley λαμβάνονται τα δεδομένα και εμφανίζονται στα σημεία που πρέπει. Για την εμφάνιση της εικόνας

χρησιμοποιήθηκε η βιβλιοθήκη Picasso. Η Picasso είναι μια βιβλιοθήκη εμφάνισης εικόνων από ένα URL.

Τέλος, αξίζει να αναφερθεί πως ανάλογα με την γλώσσα του κινητού του χρήστη εμφανίζονται τα νέα στην αντίστοιχη γλώσσα. Εάν η γλώσσα είναι η ελληνική, εμφανίζονται τα νέα που εμφανίζονται και στην ελληνική σελίδα του ραδιοφώνου. Σε διαφορετική περίπτωση, εμφανίζονται τα αγγλικά νέα που υπάρχουν στην αγγλική σελίδα του ραδιοφώνου.

3.4.6 Επόμενες εκπομπές

Στην λειτουργία «Επόμενες εκπομπές», ο χρήστης μπορεί να δει τις επόμενες εκπομπές που θα παίξουν στο ραδιόφωνο, όπως φαίνεται στην παρακάτω εικόνα (Εικόνα 12).



Εικόνα 12 "Επόμενες Εκπομπές"

Για να δημιουργηθεί αυτή η λειτουργία, αρχικά φτιάχτηκε μια ξεχωριστή απλή html σελίδα στο europeanschoolradio.eu, όπου έχει μόνο τις πληροφορίες που απαιτούνται να ληφθούν για αυτή την λειτουργία. Δηλαδή, για κάθε μία από τις επόμενες εκπομπές του ραδιοφώνου έχει τον τίτλο της εκπομπής, την ημέρα, την ημερομηνία και την ώρα που θα παίξει, το όνομα του σχολείου στο οποίο ανήκει η εκπομπή, το URL της εικόνας της, τον τύπο της εκπομπής (ζωντανή εκπομπή, εκπομπή 1 ώρας κτλ), το URL του σποτ της εκπομπής και το URL της παραγωγής όπου εκεί εμφανίζονται όλες οι πληροφορίες που αφορούν αυτή την εκπομπή.

Στην συνέχεια, χρησιμοποιώντας την βιβλιοθήκη Jsoup, επιλέγονται όλες οι πληροφορίες που εμφανίζονται στην διεπαφή του χρήστη. Πιο συγκεκριμένα, η κάθε εκπομπή εμφανίζεται μέσα σε ένα CardView όπου μέσα εμφανίζονται με TextView οι βασικές πληροφορίες όπως ο τίτλος και η ημερομηνία της εκπομπής, ενώ η εικόνα στην διεπαφή εμφανίζεται χρησιμοποιώντας την

βιβλιοθήκη **pl.droidsonroids.gif.GifImageView**. Όπως αναφέρθηκε και παραπάνω, αυτή η βιβλιοθήκη επιτρέπει την εμφάνιση ενός gif ή μιας εικόνας, σε αντίθεση με το απλό ImageView του android που εμφανίζει μόνο μια στατική εικόνα. Προτιμήθηκε αυτή η βιβλιοθήκη, μιας και με το που μπει ο χρήστης στις επόμενες εκπομπές, εμφανίζεται ένα loading gif, μέχρι να φορτωθεί και να εμφανιστεί η εικόνα. Για να εμφανιστεί η εικόνα χρησιμοποιήθηκε η βιβλιοθήκη Picasso.

Τέλος, όταν ο χρήστης πατήσει πάνω σε μια από τις επόμενες εκπομπές, ανοίγει ένα νέο Activity όπου εμφανίζονται περισσότερες πληροφορίες γι' αυτή την εκπομπή όπως η περιγραφή της, η συχνότητα που παίζει στο ραδιόφωνο, ο τύπος της, η θεματολογία της κτλ, χρησιμοποιώντας τα ίδια εργαλεία-βιβλιοθήκες με το παραπάνω Activity.

3.4.7 Ραδιοφωνικό Πρόγραμμα

Στην λειτουργία «Ραδιοφωνικό Πρόγραμμα», ο χρήστης επιλέγοντας μια συγκεκριμένη μέρα ή μήνα στο ημερολόγιο μπορεί να δει τις εκπομπές που πρόκειται να παίξουν ή που έπαιξαν. Κατά τον σχεδιασμό αυτής της λειτουργίας, παρουσιάστηκαν αρκετές δυσκολίες. Αρχικά, έπρεπε με κάποιο τρόπο να ληφθούν τα δεδομένα που έπρεπε. Πιο συγκεκριμένα, για κάθε εκπομπή που υπάρχει να ληφθούν οι ώρες (η ώρα έναρξης και η ώρα λήξης), οι ημερομηνίες που πρόκειται να παίξει ή που έπαιξε καθώς και λοιπά χρήσιμα στοιχεία.

Για να γίνει πιο κατανοητός ο τρόπος που αποθηκεύονται αυτές οι πληροφορίες στην βάση δεδομένων του European School Radio, θα εξηγηθεί η διαδικασία της κράτησης εκπομπής λίγο αναλυτικότερα. Όταν ένας χρήστης προγραμματίζει μια νέα εκπομπή στην πλατφόρμα του ραδιοφώνου, πέρα από το όνομα της εκπομπής, την περιγραφή κτλ, συμπληρώνει τον αριθμό των επεισοδίων (δηλαδή πόσες εκπομπές θα πραγματοποιήσει) και την συχνότητα (κάθε εβδομάδα, κάθε 2 ή 4 εβδομάδες). Αυτά, αποθηκεύονται στην βάση δεδομένων. Πιο συγκεκριμένα, αποθηκεύονται τα εξής πεδία που μας ενδιαφέρουν:

- production_id: Το id της παραγωγής που πραγματοποιήθηκε κράτηση
- booking_date: η ημερομηνία της πρώτης εκπομπής
- slots: Αν έχει την τιμή 1, η εκπομπή διαρκεί 30 λεπτά. Αν έχει την τιμή 2 διαρκεί 1 ώρα ενώ αν έχει την τιμή 4 διαρκεί 2 ώρες
- frequency: η συχνότητα της εκπομπής. Αν έχει την τιμή 1 είναι εβδομαδιαία, αν έχει την τιμή 2 παίζει ανά 2 εβδομάδες ενώ αν έχει την τιμή 4 παίζει ανά 4 εβδομάδες
- repeated_day: Δηλώνει ποια μέρα της εβδομάδας θα ξαναπαίξει
- repeated_times: Δηλώνει τις φορές που θα παίξει η εκπομπή, δηλαδή το πόσα επεισόδια θα πραγματοποιηθούν. Για παράδειγμα, αν έχει την τιμή 6, θα πραγματοποιηθούν 6 επεισόδια, με πρώτο την ημερομηνία που δηλώνεται στο booking_date

ενώ στον πίνακα i18n αποθηκεύονται οι μεταφράσεις των ονομάτων των εκπομπών, των περιγραφών κτλ, και παίρνουμε επιπλέον τα εξής πεδία:

- foreign_key: το id της παραγωγής
- locale: ο κωδικός της γλώσσας που είναι μεταφρασμένο π.χ. el για ελληνικά
- model: Τύπος εκπομπής, παίρνουμε την τιμή Productions

➤ field: Ο μεταφρασμένος τίτλος της παραγωγής

Γίνεται σαφές πως θα πρέπει να δημιουργηθεί ένα αρχείο rhr όπου θα λαμβάνει και θα κωδικοποιεί κάπως αυτές τις πληροφορίες ώστε να χρησιμοποιηθούν στην εφαρμογή. Το πιο εύκολο θα ήταν να λαμβάνονται αυτά τα δεδομένα με ένα sql query και να γίνονται κωδικοποίηση (encode) σε ένα αρχείο json. Η γλώσσα rhr διαθέτει έτοιμες συναρτήσεις όπου για κάθε σειρά που εμφανίζεται από το sql query, αποθηκεύονται σε έναν πίνακα αντικειμένων και στην συνέχεια αυτός ο πίνακας μπορεί να γίνει κωδικοποίηση σε json. Όμως, εδώ το πρόβλημα είναι ότι δεν μπορεί να γίνει τόσο απλά, μιας και δεν υπάρχει κάποιος πίνακας που να αποθηκεύει όλες τις εκπομπές στις συγκεκριμένες ημερομηνίες. Έτσι, αν επιλέγαμε να κάνουμε κωδικοποίηση σε json τις εκπομπές σύμφωνα με τις ημερομηνίες που φαίνονται στην βάση, θα παίρναμε μόνο το πρώτο επεισόδιο από κάθε παραγωγή. Για παράδειγμα, εάν ένας χρήστης προγραμματίσει 3 εβδομαδιαία επεισόδια, με το πρώτο να παίζει στις 1/1/2020, το 2^ο θα παίζει στις 8/1/2020 ενώ το τρίτο στις 15/1/2020. Όμως, στην βάση δεν υπάρχει γραμμή που να δείχνει ότι η εκπομπή αυτή θα παίζει για τις ημερομηνίες εκτός της πρώτης, δείχνει μόνο ότι θα παίξουν 3 επεισόδια, ανά 7 μέρες, με ημερομηνία έναρξης 1/1/2020. Έτσι, θα πρέπει στο αρχείο rhr που θα χρησιμοποιηθεί για την διασύνδεση της βάσης με την εφαρμογή να προστίθενται οι ημερομηνίες, αλλά και να γίνονται κάποιες λειτουργίες ακόμα όπως είναι ο υπολογισμός της ώρας λήξης της εκπομπής (μιας και στην βάση υπάρχει μόνο η ώρα έναρξης) κτλ.

Ξεκινώντας, θα πρέπει να σημειωθεί ο τρόπος που θα παίρνονται τα δεδομένα στην εφαρμογή, και την μορφή που θα έχει το rhr αρχείο. Στην εφαρμογή, το πρόγραμμα θα εμφανίζεται με ένα CalendarView, όπως φαίνεται παρακάτω (Εικόνα 13).

| Μάιος 2020 | | | | | | |
|------------|----|----|----|----|----|----|
| < | | | | | | > |
| Δ | Τ | Τ | Π | Π | Σ | Κ |
| | | | | 1 | 2 | 3 |
| 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 | 29 | 30 | 31 |

Εικόνα 13 "CalendarView"

Όταν ο χρήστης πατάει πάνω σε μία ημέρα, ενεργοποιείται η μέθοδος `onSelectedDayChange(CalendarView view, int year, int month, int dayOfMonth)`, που όπως φαίνεται παίρνει παραμέτρους την μέρα, τον μήνα και το έτος. Άρα, αν για παράδειγμα πατηθεί το 19 στον μήνα Μάιο που βρίσκεται το CalendarView, η μέθοδος θα πάρει τις εξής τιμές:

- dayOfMonth=19
- month=5
- year=2020.

Άρα, ένας πρακτικός τρόπος για να επιλέγονται οι εκπομπές μιας συγκεκριμένης ημερομηνίας είναι να περνιούνται με ένα GET Request στο αρχείο calendar.php αυτές οι τιμές, και έτσι το αρχείο να εμφανίζει με κάποιον τρόπο τις εκπομπές για την συγκεκριμένη ημερομηνία. Έτσι, το αρχείο θα έχει την εξής δομή: calendar.php?date=year-month-dayOfMonth. Άρα, αν ήταν για παράδειγμα calendar.php?date=2019-12-16, θα εμφάνιζε τις εκπομπές που θα παίξουν στις 16/12/2019.

Πηγαίνοντας ξανά πίσω στην υλοποίηση του php αρχείου, θα πρέπει να εξηγηθεί ο τρόπος με τον οποίο θα μπορούν να εμφανίζονται οι εκπομπές σωστά για όλες τις ημερομηνίες και γενικά η λογική με την οποία σχεδιάστηκε το αρχείο php. Αρχικά, το πεδίο repeated_times όπως προαναφέρθηκε μας δείχνει τον αριθμό των εκπομπών (επεισοδίων) που θα πραγματοποιηθούν. Έτσι, πρέπει στο ήδη υπάρχων επεισόδιο να προστεθούν κι άλλα επεισόδια, ίσα με την τιμή του πεδίου repeated_times και αφαιρώντας 1, που είναι η πρώτη εκπομπή που ήδη έχει καταγραφή σαν ημερομηνία στην βάση. Όμως, για να γίνει αυτό, θα πρέπει να ληφθεί υπόψιν και η τιμή του πεδίου frequency, ώστε στην ήδη υπάρχουσα ημερομηνία να προστεθούν επεισόδια κατά 7, 14 ή 28 μέρες. Αυτό γίνεται, με τον τρόπο που φαίνεται παρακάτω:

```
$reptime = date('Y-m-d', strtotime($dateshow.' + '.$dayscounter.' days'));
```

όπου *\$dateshow* η ημερομηνία του προηγούμενου επεισοδίου, ξεκινώντας από το πρώτο και *\$dayscounter* η τιμή 7,14 ή 28 ανάλογα με την συχνότητα που έχει επιλεγεί. Επίσης, επειδή η ώρα (starttime) που υπάρχει στην βάση είναι σε UTC, πρέπει να προστεθούν οι ώρες που πρέπει. Έτσι εξετάζεται αν η κράτηση έγινε πριν την αλλαγή της ώρας ή μετά, και έτσι προστίθενται οι κατάλληλες ώρες. Όμως, εκτός από την ημερομηνία έναρξης(starttime) πρέπει να υπολογιστεί και η ώρα λήξης της κάθε εκπομπής ώστε να φαίνεται στην εφαρμογή. Για να υπολογιστεί η ώρα λήξης, πρέπει να ληφθεί υπόψιν η τιμή του πεδίου slots. Αν έχει την τιμή 1, προστίθενται 30 λεπτά στο starttime, αν έχει την τιμή 2 προστίθεται 1 ώρα ενώ αν έχει την τιμή 4 προστίθενται 2 ώρες. Αλγοριθμικά, ο τρόπος που γίνεται φαίνεται παρακάτω:

```
if($row["slots"] == 1){  
  
    $timeend = date('H:i', strtotime($timestart)+1800);  
  
}  
  
if($row["slots"] == 2){  
  
    $timeend = date('H:i', strtotime($timestart)+3600);  
  
}  
  
if($row["slots"] == 4){  
  
    $timeend = date('H:i', strtotime($timestart)+7200);  
  
}
```

}

Ακόμα μια παράμετρος που θα πρέπει να εμφανίζεται στην εφαρμογή είναι το αν η εκπομπή είναι ζωντανή ή όχι, και στις περιπτώσεις που δεν είναι ζωντανή να ελέγχεται αν έχει ανέβει επεισόδιο, δηλαδή αν θα πραγματοποιηθεί για εκείνη την ημερομηνία. Έτσι, εμφανίζεται αν τελικά η εκπομπή είναι ζωντανή, αν έχει ανέβει επεισόδιο ή όχι. Για να γίνει αυτός ο έλεγχος, ελέγχεται αν υπάρχει το id του επεισοδίου στον πίνακα archives της βάσης για την συγκεκριμένη ημερομηνία.

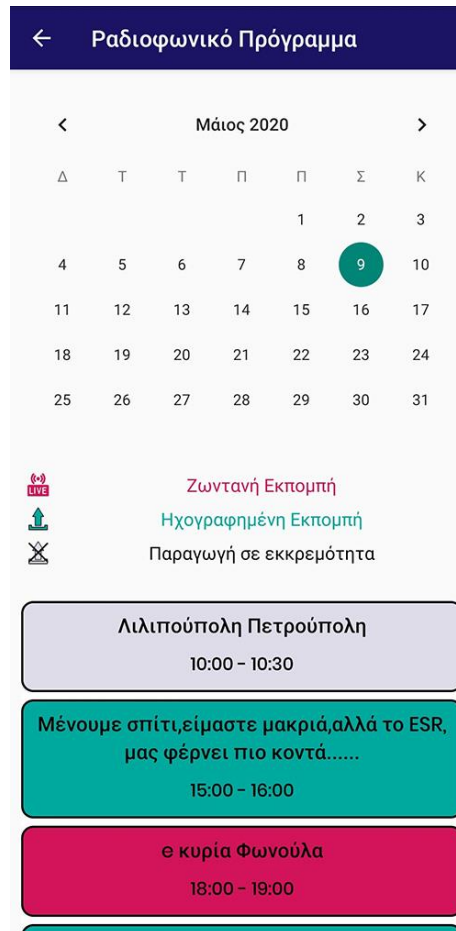
Τέλος, για να γίνει ταξινόμηση των εκπομπών για μια συγκεκριμένη μέρα με βάση την ώρα έναρξης χρησιμοποιείται η μέθοδος usort, αφού γίνει αποκωδικοποίηση του υπάρχοντος json και στην συνέχεια ξανά κωδικοποίηση, όπως φαίνεται παρακάτω.

```
$arr = json_decode($jsonbydate);  
  
usort($arr, function($item1, $item2){  
  
return $item1->timestart > $item2->timestart;  
  
});  
  
$myJSON = json_encode($arr);
```

Όλες αυτές οι πληροφορίες λοιπόν εμφανίζονται σε ένα json. Εκεί υπάρχουν πεδία όπως name, book_date, timestart, timeend, prod_type κτλ που δίνουν όλες τις πληροφορίες που απαιτούνται για το ραδιοφωνικό πρόγραμμα στην εφαρμογή. Επίσης, μιας και η εφαρμογή υποστηρίζει και αγγλικά και ελληνικά, έχει προβλεφθεί και η εμφάνιση των ονομάτων των εκπομπών και στα αγγλικά και στα ελληνικά. Εκτός από την παράμετρο date, προσθέτοντας την παράμετρο lang και ορίζοντάς της την τιμή gr, τα ονόματα είναι στα ελληνικά, ενώ σε οποιαδήποτε άλλη περίπτωση είναι με λατινικούς χαρακτήρες στα αγγλικά. Παρακάτω, φαίνεται το URL με το οποίο θα εμφανιζόντουσαν οι εκπομπές της ημέρας 9/5/2020 στα ελληνικά.

<http://europeanschoolradio.eu/mobile-app/calendar.php?date=2020-05-09&lang=gr>

Όσο αφορά το κομμάτι της εφαρμογής android, στην λειτουργία «Ραδιοφωνικό Πρόγραμμα» περιλαμβάνεται το ημερολόγιο μέσω ενός CalendarView και ένα ListView, όπου εκεί εμφανίζονται οι εκπομπές της επιλεγμένης ημερομηνίας (Εικόνα 14). Αρχικά, στο CalendarView με την μέθοδο onSelectedDayChange παίρνουν τιμές οι παράμετροι που αφορούν την μέρα, τον μήνα και το έτος, όταν πατηθεί μια συγκεκριμένη ημέρα από τον χρήστη. Στην συνέχεια γίνεται το parse, δηλαδή η ανάλυση των πληροφοριών του json, και για κάθε εκπομπή φορτώνεται μια ξεχωριστή γραμμή στο ListView.



Εικόνα 14 "Ραδιοφωνικό Πρόγραμμα"

Ο χρήστης μπορεί να επιλέξει αρχικά τον μήνα που τον ενδιαφέρει και στην συνέχεια την ημέρα. Μόλις πατήσει πάνω σε μια ημέρα, αλλάζει δυναμικά η λίστα και εμφανίζονται οι εκπομπές εκείνης της μέρας. Φυσικά, μόλις μπει στο ραδιοφωνικό πρόγραμμα είναι επιλεγμένη η σημερινή μέρα και στην λίστα φαίνονται οι εκπομπές εκείνης της μέρας. Επιπλέον, πατώντας πάνω σε μια εκπομπή της λίστας, του εμφανίζονται περισσότερες λεπτομέρειες για την εκπομπή όπως είναι η περιγραφή, η εικόνα της εκπομπής κτλ, ενώ έχει την δυνατότητα να ακούσει και επεισόδια της εκπομπής που έχουν ήδη ανέβει.

Για την δημιουργία και το γέμισμα της λίστας (του ListView) αρχικά δημιουργείται μια κλάση που θα περιέχει τον constructor και μέθοδοι get και set, που περιέχουν τις μεταβλητές που θα έχει κάθε στοιχείο της λίστας. Στην συνέχεια, δημιουργείται ένας adapter, που αποτελεί την «γέφυρα» μεταξύ της λίστας και της κλάσης που περιέχει τις μεταβλητές με τα στοιχεία που θα εμφανίζονται. Στον adapter (αρχείο scheduleadapter.java) δηλώνεται το xml που θα χρησιμοποιείται για την διάταξη των στοιχείων (όνομα εκπομπής, ώρα έναρξης) κάθε στοιχείου της λίστας, ενώ ελέγχεται εάν η εκπομπή είναι ζωντανή, ηχογραφημένη ή εάν δεν έχει ανέβει ακόμα επεισόδιο, και παίρνει το αντίστοιχο στυλ εμφάνισης. Για την εμφάνιση αυτών των στοιχείων της λίστας, υπεύθυνα είναι τα αρχεία live_schedule_item.xml, uploaded_schedule_item.xml και not_uploaded_schedule_item.xml που περιλαμβάνουν παραμέτρους όπως το μέγεθος της καμπυλότητας των γωνιών, το χρώμα φόντου καθώς και το χρώμα περιγράμματος.

3.4.8 Παραγωγές

Ένα σημαντικό κομμάτι του μαθητικού ραδιοφώνου είναι οι παραγωγές των μαθητών και το αρχείο εκπομπών του ραδιοφώνου. Στο αρχείο εκπομπών, οι ραδιοφωνικοί παραγωγοί ανεβάζουν τις εκπομπές που έχουν δημιουργήσει και αναπαράγει, με σκοπό να είναι διαθέσιμες στους χρήστες ώστε να μπορούν να τις ακούσουν όποτε αυτοί επιθυμούν. Σίγουρα μια τέτοια λειτουργία στην εφαρμογή θα είναι ιδιαίτερα χρήσιμη, ώστε οι χρήστες να μπορούν από το κινητό τους να ακούν τις εκπομπές που επιθυμούν.

Αρχικά, για να δημιουργηθεί αυτή η λειτουργία είναι απαραίτητο να ληφθούν όλες οι πληροφορίες που χρειάζονται από την βάση δεδομένων, όπως id επεισοδίου, όνομα επεισοδίου, περιγραφή, ημερομηνία μετάδοσης κτλ. Η λειτουργία του αρχείου εκπομπών θα έχει ως εξής: όταν ο χρήστης μπαίνει σ αυτή την λειτουργία, θα υπάρχουν τα πρόσφατα επεισόδια, δηλαδή τα επεισόδια που μεταφορτώθηκαν τελευταία. Επίσης, θα μπορεί κάνοντας αναζήτηση το όνομα μιας εκπομπής με ελληνικούς ή λατινικούς χαρακτήρες, να ακούσει τα επεισόδια που εμφανίστηκαν σύμφωνα με το αποτέλεσμα της αναζήτησης του.

Για να γίνουν όλα αυτά, είναι απαραίτητο να δημιουργηθεί ένα αρχείο php στον server του European School Radio, που θα λειτουργεί ως η διασύνδεση μεταξύ βάσης δεδομένων και εφαρμογής. Το αρχείο podcasts.php εμφανίζει τα δεδομένα σε μορφή json, και έτσι λαμβάνονται, επεξεργάζονται και εμφανίζονται στην εφαρμογή. Αυτό το αρχείο μπορεί να εμφανίσει δεδομένα με 2 τρόπους, ανάλογα με την επιλογή του χρήστη. Εάν δεν μπει κάποιο όρισμα στο αρχείο, δηλαδή αν έχει την μορφή <http://europeanschoolradio.eu/mobile-app/podcasts.php>, εμφανίζει τις τελευταίες εκπομπές (εικόνα 15), ενώ εάν μπει και το όρισμα name με μια τιμή, εμφανίζονται σε json οι εκπομπές που αντιστοιχούν ή περιέχουν την τιμή του ορίσματος name. Για παράδειγμα, αν το όρισμα έχει την τιμή “kara” ή “kara”, δηλαδή αν το url είναι <http://europeanschoolradio.eu/mobile-app/podcasts.php?name=kara>, εμφανίζει όλες τις εκπομπές που περιέχουν την λέξη “kara” ή “kara”. Επίσης, αν χρησιμοποιηθεί και το όρισμα lang=gr (με τον ίδιο τρόπο όπως το name), εμφανίζονται τα ονόματα των εκπομπών στα ελληνικά, αλλιώς εμφανίζονται στα αγγλικά. Αυτό είναι χρήσιμο, μιας και όπως έχει αναφερθεί η εφαρμογή υποστηρίζει και αγγλικά και ελληνικά, άρα στα ελληνικά το αρχείο php έχει και αυτό το όρισμα. Όπως φαίνεται και παραπάνω, εάν ο χρήστης πληκτρολογήσει το όνομα ενός επεισοδίου στα αγγλικά ή στα greeklish, η εφαρμογή είναι σε θέση να καταλάβει για ποιο επεισόδιο πρόκειται και το εμφανίζει στην αντίστοιχη γλώσσα που είναι και η εφαρμογή.

```

1  [
2  {
3    "mp3_id": "5691",
4    "production_id": "4210",
5    "name": "The evolution of the Universe from the perspective of Physics.",
6    "description": "The students present the evolution of the Universe from the perspective of Physics.",
7    "broadcast_day": "08/06/2020",
8    "start_time": "13:00",
9    "banner": "5691_LKB_Radio_Logo.png",
10   "grname": "Η εξέλιξη του Σύμπαντος από τη σκοπιά της Φυσικής.",
11   "grdesc": "Οι μαθητές παρουσιάζουν την εξέλιξη του Σύμπαντος από τη σκοπιά της Φυσικής."
12  },
13  {
14   "mp3_id": "5689",
15   "production_id": "3501",
16   "name": "Medecins Sans Frontieres ",
17   "description": "A very interesting interview concerning the mission of the Organisation #MedecinsSansFrontieres .Stay tuned to
18   "broadcast_day": "07/06/2020",
19   "start_time": "16:00",
20   "banner": "5689_msf_banner.PNG",
21   "grname": "Συνέντευξη στους \"/>

```

Εικόνα 15 "Μορφή json για podcasts"

Όσο αφορά την βάση δεδομένων, για να ληφθούν τα δεδομένα χρησιμοποιήθηκε ο πίνακας archives και ο πίνακας i18n, που περιέχει την ελληνική μετάφραση των στοιχείων των εκπομπών. Χρησιμοποιήθηκαν τα πεδία:

Από τον πίνακα archives:

- id, το id του επεισοδίου
- production_id, το id της παραγωγής
- name, το όνομα του επεισοδίου
- description, η περιγραφή του επεισοδίου
- broadcast_day, η ημερομηνία που έπαιξε
- start_time, η ώρα έναρξης
- banner, τον κωδικό (για τον συνδέσμο του banner)

Από τον πίνακα i18n:

- grn.content, το όνομα του επεισοδίου στα ελληνικά
- grd.content, η περιγραφή στα ελληνικά

Εάν δεν έχει δοθεί όρισμα name, εμφανίζονται τα στοιχεία των πινάκων που έχουν επιλεγεί από το sql query, ταξινομημένα από το πιο πρόσφατο στο παλαιότερο. Για την σωστή μετατροπή της ημερομηνίας και την σωστή ταξινόμηση χρησιμοποιήθηκε η συνάρτηση *STR_TO_DATE(archives.broadcast_day, '%d/%m/%Y')*.

Όσο αφορά την εφαρμογή, σ αυτή την λειτουργία περιέχεται ένα SearchView, δηλαδή ένα textbox που χρησιμοποιείται για την αναζήτηση των επεισοδίων και ένα ListView όπου εκεί φορτώνονται οι εκπομπές (εικόνα 16). Υπάρχει επιπλέον και ένα TextView που χρησιμοποιείται στην εμφάνιση των πρόσφατων εκπομπών και ένα GifImageView που εμφανίζει ένα loading gif όταν φορτώνει τα δεδομένα.

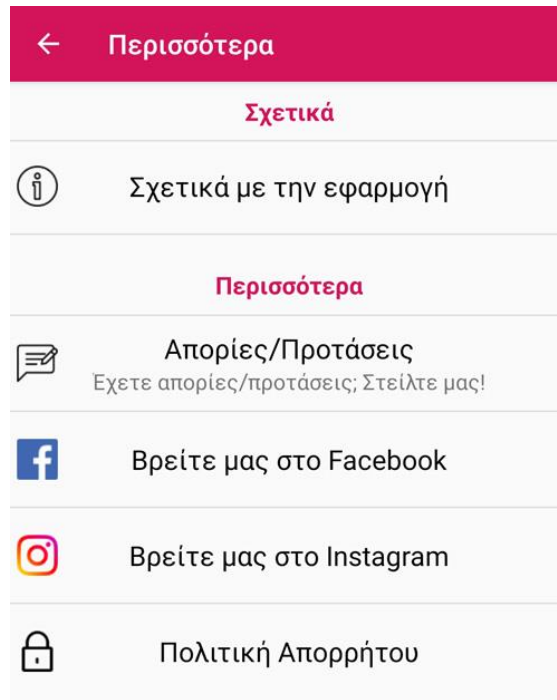


Εικόνα 16 "Εκπομπές"

Όταν μπαίνει την πρώτη φορά εμφανίζονται οι πρόσφατες εκπομπές καλώντας το αρχείο podcasts.php και κάνοντας parse το json, ενώ στην συνέχεια τοποθετεί τις πληροφορίες που παίρνει στο listview. Όταν κάνει αναζήτηση, εκτελείται η μέθοδος onQueryTextSubmit που παίρνει σαν παράμετρο το κείμενο που έγραψε ο χρήστης στην αναζήτηση, καλείται το αρχείο podcasts.php?name=apotelesma, όπου apotelesma το κείμενο στο SearchView.

3.4.9 Περισσότερα

Η τελευταία επιλογή στο μενού περιέχει την λειτουργία «Περισσότερα». Εδώ ο χρήστης, μπορεί να δει κάποιες πληροφορίες σχετικά με την εφαρμογή ή να διαβάσει κάποιες οδηγίες χρήσης της εφαρμογής, ενώ μέσα από την εφαρμογή μπορεί να στείλει ένα μήνυμα στο European School Radio, καθώς επίσης και να συνδεθεί με τα μέσα κοινωνικής δικτύωσης του ραδιοφώνου.



Εικόνα 17 "Περισσότερα"

Η παραπάνω λειτουργία είναι δημιουργημένη με ένα `RelativeLayout`, που μέσα περιλαμβάνει επιμέρους `RelativeLayout` που περιλαμβάνουν `ImageViews`, `TextViews` και `Views`. Κάθε ξεχωριστή επιλογή σ αυτή την λειτουργία είναι και ένα ξεχωριστό `RelativeLayout` που όταν πατηθεί, μεταφέρει τον χρήστη σε αντίστοιχο `Activity` ή σε κάποια ξεχωριστή εφαρμογή (όπως του facebook και του instagram).

3.4.10 Μετάφραση Εφαρμογής

Ένα από τα χαρακτηριστικά της εφαρμογής είναι η μετάφραση της ανάλογα με την γλώσσα που έχει ο χρήστης και στο κινητό του. Προς το παρόν, οι γλώσσες που υποστηρίζει η εφαρμογή είναι τα ελληνικά και τα αγγλικά. Έτσι, εάν στο κινητό του χρήστη είναι επιλεγμένη η ελληνική γλώσσα, το περιβάλλον της εφαρμογής είναι κι αυτό στα ελληνικά, ενώ εάν έχει κάποια άλλη γλώσσα εκτός των ελληνικών, η εφαρμογή είναι στα αγγλικά. Για να γίνει αυτό, έπρεπε να γίνει μετάφραση όλων των λέξεων-κειμένων της εφαρμογής, μέσα από τον `Translations Editor` του `Android Studio`, δηλαδή τον επεξεργαστή μεταφράσεων, όπως φαίνεται και στην παρακάτω εικόνα (Εικόνα 18), ενώ στα αρχεία `java` και `xml` δεν γράφεται η λέξη αυτολεξεί, αλλά γράφεται με το συγκεκριμένο `key` που δόθηκε κατά την μετάφραση του στον επεξεργαστή μεταφράσεων π.χ. `@string/string_for_trans` για τα αρχεία `xml` και `R.string.string_for_trans` για τα αρχεία `java`.

| Key | Resource Folder | Untranslatable | Default Value | English (en) | Greek (el) in Greece (GR) |
|--------------------------|------------------|--------------------------|------------------------------------|------------------------------------|-------------------------------------|
| settings | app/src/main/res | <input type="checkbox"/> | More | More | Περισσότερα |
| more | app/src/main/res | <input type="checkbox"/> | More | More | Περισσότερα |
| find_us_on_insta | app/src/main/res | <input type="checkbox"/> | Find us on Instagram | Find us on Instagram | Βρείτε μας στο Instagram |
| feedback | app/src/main/res | <input type="checkbox"/> | Send Feedback | Send Feedback | Απορίες/Προτάσεις |
| feedback_details | app/src/main/res | <input type="checkbox"/> | Got any questions? Contact us! | Got any questions? Contact us! | Έχετε απορίες/προτάσεις; Στείλτε μα |
| listen | app/src/main/res | <input type="checkbox"/> | Listen | Listen | Ακούστε |
| live | app/src/main/res | <input type="checkbox"/> | Live | Live | Ζωντανά |
| news | app/src/main/res | <input type="checkbox"/> | News | News | Νέα |
| schedule | app/src/main/res | <input type="checkbox"/> | Schedule | Schedule | Πρόγραμμα |
| settings_on_menu | app/src/main/res | <input type="checkbox"/> | More | More | Περισσότερα |
| podcasts | app/src/main/res | <input type="checkbox"/> | Podcasts | Podcasts | Παραγωγές |
| next | app/src/main/res | <input type="checkbox"/> | Next | Next | Επόμενες |
| shows | app/src/main/res | <input type="checkbox"/> | Shows | Shows | Εκπομπές |
| privacy | app/src/main/res | <input type="checkbox"/> | Privacy Policy | Privacy Policy | Πολιτική Απορρήτου |
| please_wait | app/src/main/res | <input type="checkbox"/> | Please Wait... | Please Wait... | Παρακαλώ Περιμένετε... |
| name_in_form | app/src/main/res | <input type="checkbox"/> | Name | Name | Όνομα |
| theme_in_form | app/src/main/res | <input type="checkbox"/> | Theme | Theme | Θέμα |
| message_in_form | app/src/main/res | <input type="checkbox"/> | Message | Message | Περιεχόμενο Μηνύματος |
| feedback_send_btn | app/src/main/res | <input type="checkbox"/> | Send | Send | Αποστολή |
| select_email_app | app/src/main/res | <input type="checkbox"/> | Please select Email App e.g. Gmail | Please select Email App e.g. Gmail | Παρακαλώ επιλέξτε εφαρμογή αποστο |
| no_email_client | app/src/main/res | <input type="checkbox"/> | There are no Email Clients | There are no Email Clients | Δεν έχετε εφαρμογή για αποστολή Επ |
| schedule_title | app/src/main/res | <input type="checkbox"/> | Radio Schedule | Radio Schedule | Ραδιοφωνικό Πρόγραμμα |
| next_shows_title | app/src/main/res | <input type="checkbox"/> | Upcoming Shows | Upcoming Shows | Επόμενες Εκπομπές |
| news_title | app/src/main/res | <input type="checkbox"/> | Radio News | Radio News | Ραδιοφωνικά Νέα |
| Listen_Live_title | app/src/main/res | <input type="checkbox"/> | Listen Live | Listen Live | Ακούστε Ζωντανά |
| no_internet_text | app/src/main/res | <input type="checkbox"/> | No Internet | No Internet | Δεν υπάρχει σύνδεση στο διαδίκτυο |
| connect_to_internet_text | app/src/main/res | <input type="checkbox"/> | Connect to Internet | Connect to Internet | Συνδεθείτε στο Διαδίκτυο |
| wifi_connected | app/src/main/res | <input type="checkbox"/> | You Are Connected to Wifi | You Are Connected to Wifi | Είστε συνδεδεμένοι μέσω Wifi |
| mobile_data_connected | app/src/main/res | <input type="checkbox"/> | You Are Connected to Mobile Data | You Are Connected to Mobile Data | Είστε συνδεδεμένοι μέσω δεδομένων |
| welcome_msg | app/src/main/res | <input type="checkbox"/> | Welcome! | Welcome! | Καλώς Ορίσατε! |
| welcome_msg_inside | app/src/main/res | <input type="checkbox"/> | Welcome to Official European Scho | Welcome to Official European Scho | Καλώς ορίσατε στην επίσημη εφαρμ |
| new_episodes | app/src/main/res | <input type="checkbox"/> | New Episodes | New Episodes | Νέα Προγράμματα |

Εικόνα 18 "Μετάφραση Εφαρμογής"

ΕΠΙΛΟΓΟΣ

Σ αυτό το κεφάλαιο αναλύθηκε στιδήποτε έχει σχέση με τον σχεδιασμό και την υλοποίηση της εφαρμογής του μαθητικού ραδιοφώνου. Αναλύθηκαν οι διαδικασίες, οι βιβλιοθήκες και η λογική πάνω στην οποία στηρίχθηκε η εφαρμογή. Στο επόμενο κεφάλαιο θα παρουσιαστούν κάποιες πιθανές μελλοντικές επεκτάσεις που θα μπορούσαν να γίνουν στην εφαρμογή.

ΚΕΦΑΛΑΙΟ 4 – Μελλοντικές επεκτάσεις

Η εφαρμογή του μαθητικού ραδιοφώνου δεν θα σταματήσει να αναπτύσσεται εδώ, αλλά θα προστίθενται συνεχώς νέες λειτουργίες, χαρακτηριστικά αλλά και βελτιώσεις στην ήδη υπάρχουσα εφαρμογή, σύμφωνα με τις επιθυμίες και τις ανάγκες των χρηστών. Παρακάτω, αναφέρονται κάποιες πιθανές βελτιώσεις που θα μπορούσαν να γίνουν.

4.1 Ειδοποιήσεις εφαρμογής

Μια χρήσιμη λειτουργία της εφαρμογής θα ήταν η ύπαρξη ειδοποιήσεων (notifications). Πιο συγκεκριμένα, να μπορεί ο χρήστης να επιλέγει μία ή περισσότερες αγαπημένες εκπομπές στο ραδιόφωνο, και όταν φτάνει η ώρα έναρξής τους να λαμβάνει ειδοποίηση ώστε να μπει να τις ακούσει. Επίσης, θα ήταν χρήσιμο να λαμβάνει ειδοποιήσεις και για άλλες λειτουργίες, όπως μια ανακοίνωση ή ένα σημαντικό νέο του ραδιοφώνου.

4.2 Σύνδεση με social media

Πλέον, σχεδόν όλοι οι χρήστες έξυπνων τηλεφώνων έχουν λογαριασμό σε κάποιο μέσο κοινωνικής δικτύωσης. Η σύνδεση της εφαρμογής με τα μέσα κοινωνικής δικτύωσης μπορεί να αυξήσει την ακροαματικότητα μιας εκπομπής αλλά και την δημοτικότητα του ραδιοφώνου. Θα μπορούσε ο χρήστης να κοινοποιεί την εκπομπή που ακούει εκείνη την στιγμή ή οτιδήποτε άλλο.

4.3 Ανέβασμα ηχητικών αρχείων ή ηχογραφήσεων

Ίσως μια καλή ιδέα θα ήταν να μπορούν οι χρήστες της εφαρμογής να ανεβάζουν με κάποιον εύκολο και εύχρηστο τρόπο ένα ηχητικό αρχείο ή να κάνουν μια ηχογράφηση κατευθείαν μέσα στην εφαρμογή και να την στείλουν στο European School Radio. Από κει και πέρα, αυτά τα ηχητικά θα μπορούσαν να συμπεριληφθούν στην καθημερινή ροή του ραδιοφώνου, ή ακόμα και να δημιουργηθεί μια σελίδα όπου θα φαίνονται όλα αυτά, μια σελίδα αφιερώσεων ή οτιδήποτε άλλο.

4.4 Δυναμική ενεργοποίηση/απενεργοποίηση μέρους της εφαρμογής

Μία ακόμα λειτουργία που θα μπορούσε να έχει η εφαρμογή είναι η δυναμική ενεργοποίηση ή απενεργοποίηση μέρους της εφαρμογής. Πιο συγκεκριμένα, σε σπάνιες περιπτώσεις όπου μπορεί να προκύψει κάποιο πρόβλημα στο European School Radio, όπως για παράδειγμα να έχει πέσει η βάση δεδομένων, αλλά να λειτουργεί ο icecast server (άρα να μπορεί να πραγματοποιείται μια εκπομπή και να μεταδίδεται στους ακροατές κανονικά), θα ήταν χρήσιμο να απενεργοποιούνται μέρη της εφαρμογής που θα λειτουργούσαν προβληματικά (όπως τα Νέα ή το Ραδιοφωνικό Πρόγραμμα), μιας και αυτές οι λειτουργίες λαμβάνουν και χρησιμοποιούν πληροφορίες από την βάση δεδομένων για να τις εμφανίσουν.

Ένας πιθανός τρόπος που θα μπορούσε να γίνει αυτό θα ήταν να δημιουργηθεί ένα αρχείο json με κάποια πεδία, όπου ανάλογα την τιμή των πεδίων θα γίνονται συγκεκριμένες ενέργειες. Έτσι, κάθε φορά που ο χρήστης θα μπαίνει στο μενού, θα γίνεται ανάλυση (parse) στο αρχείο json και θα ελέγχεται η τιμή ορισμένων πεδίων. Αν ένα ή περισσότερα πεδία έχουν συγκεκριμένη τιμή (την τιμή που θα οριστεί ότι θα παίρνει όταν παρουσιάζεται το πρόβλημα), θα απενεργοποιείται το πάτημα σε ορισμένες επιλογές. Να σημειωθεί πως μπορεί πολύ εύκολα μέσα από το σύστημα αρχείων του server να γίνει επεξεργασία σε κάποιο αρχείο. Άρα, όταν παρουσιαστεί κάποιο πρόβλημα θα επεξεργάζεται επιτόπου το αρχείο και δυναμικά θα ενεργοποιούνται και θα απενεργοποιούνται συγκεκριμένες λειτουργίες.

4.5 Ψηφοφορία Διαγωνισμού

Κάθε χρόνο, διοργανώνεται από το European School Radio ο διαγωνισμός «Καν' το ν' ακουστεί». Σε αυτόν τον διαγωνισμό, τα σχολεία ανεβάζουν τις δικές τους συμμετοχές, συγκεκριμένα ένα ραδιοφωνικό σποτ ή τραγούδι, και ψηφίζονται από όλους τους χρήστες του μαθητικού Ραδιοφώνου. Μια ακόμα μελλοντική αναβάθμιση της εφαρμογής θα μπορούσε να υποστηρίξει και αυτή την λειτουργία. Πιο συγκεκριμένα, μέσα από την εφαρμογή να μπορούν οι χρήστες να ψηφίζουν τις αγαπημένες τους συμμετοχές, κάτι που μέχρι τώρα γινόταν από την ιστοσελίδα.

ΕΠΙΛΟΓΟΣ

Οι λειτουργίες που συνεχώς προστίθενται όχι μόνο στην εφαρμογή, αλλά και γενικότερα στην ιστοσελίδα και πλατφόρμα του European School Radio, καθιστούν μεγάλη την ανάγκη για συνεχή αναβάθμιση της εφαρμογής και βελτίωσης των ήδη υπάρχοντων λειτουργιών. Μιας και το μεγαλύτερο μέρος των χρηστών του μαθητικού ραδιοφώνου χρησιμοποιούν το κινητό τους τηλέφωνο για να περιηγηθούν στις σελίδες του μαθητικού ραδιοφώνου, είναι πολύ σημαντικό να ανανεώνονται και να βελτιώνονται οι λειτουργίες που προσφέρονται μέσα από την εφαρμογή.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία δημιουργήθηκε μια εφαρμογή για κινητά τηλέφωνα με θέμα την ζωντανή μετάδοση ήχου του ραδιοφώνου αλλά και την πρόσβαση σε δημοφιλείς λειτουργίες του European School Radio. Μετά από αρκετή έρευνα και καταγραφή τόσο των αναγκών όσο και των επιθυμιών των χρηστών του μαθητικού ραδιοφώνου, υλοποιήθηκε η παρούσα εφαρμογή με βάση συγκεκριμένες προδιαγραφές και λειτουργίες του ραδιοφώνου. Η εφαρμογή ήδη δόθηκε για έναν πρώτο έλεγχο σε κάποια βασικά μέλη του European School Radio, αποσπώντας πολύ θετικές κριτικές.

Για την δημιουργία της εφαρμογής, απαιτήθηκε η πολύ καλή γνώση της γλώσσας προγραμματισμού java, του Android Studio αλλά και γενικά της φιλοσοφίας του λειτουργικού Android όσο αφορά τον τρόπο λειτουργίας του, τα στοιχεία που περιέχει, τις υπηρεσίες, τις βιβλιοθήκες του κτλ.

Επίσης, απαιτήθηκε η συνεχής επαφή και ανταλλαγή ιδεών με μέλη του ραδιοφώνου που γνωρίζουν καλά τις λειτουργίες και την φιλοσοφία του ραδιοφώνου, ώστε η εφαρμογή να παραμένει και να συνεχίζει να αναπτύσσεται προς την σωστή κατεύθυνση, με κύριο σκοπό την καλύτερη δυνατή εμφάνιση των λειτουργιών που αναπτύσσονται και την εύκολη χρήση τους από ολόκληρη την μερίδα χρηστών.

Πολύ μεγάλο μέρος της εφαρμογής απαιτούσε την λήψη πληροφοριών και στοιχείων από το European School Radio. Έτσι, ήταν απαραίτητη η μελέτη και η εκμάθηση του τρόπου σχεδιασμού της πλατφόρμας του ραδιοφώνου, της ιστοσελίδας, των αρχείων, των εργαλείων και της βάσης δεδομένων που χρησιμοποιείται.

Μελλοντικά, η εφαρμογή θα συνεχίσει να αναπτύσσεται και να βελτιώνεται σε όλους τους τομείς. Θα προστεθούν νέες λειτουργίες με στόχο την όσο το δυνατόν μεγαλύτερη κάλυψη των αναγκών των χρηστών, ώστε η εφαρμογή να φτάσει στο σημείο να μπορεί να υποστηρίξει σχεδόν όλες τις λειτουργίες του ραδιοφώνου, χωρίς να απαιτείται να μπαίνει κάποιος στην ιστοσελίδα για να τις κάνει. Παρά

Τέλος, ήδη η εφαρμογή αναμένεται από μεγάλη μερίδα χρηστών και εκτιμάται ότι θα έχει μεγάλη επιτυχία στους χρήστες του ραδιοφώνου.

ΑΝΑΦΟΡΕΣ

- [1] <http://europeanschoolradio.eu/>
- [2] <https://www.elprocus.com/>
- [3] <https://www.javatpoint.com/>
- [4] <https://www.androidauthority.com/>
- [5] <https://www.tutorialspoint.com/>
- [6] <https://www.tutlane.com/>
- [7] <https://www.techotopia.com/>
- [8] <https://developer.android.com/>
- [9] <https://www.geeksforgeeks.org/>
- [10] <https://developer.android.com/>
- [11] <https://www.rishabhsoft.com/>
- [12] <https://www.netguru.com/>
- [13] <https://hackernoon.com/>
- [14] <https://gs.statcounter.com/>
- [15] <https://support.google.com/>
- [16] <https://www.admecindia.co.in/>
- [17] <https://developer.android.com/>
- [18] <https://developer.android.com/>

ΒΙΒΛΙΟΓΡΑΦΙΑ

International Journal of Multimedia and Ubiquitous Engineering Vol.9, No.4 (2014), pp.187-198.
[Online]. Available: <http://dx.doi.org>

Mark Reynolds, “Xamarin Mobile Application Development for Android”, January 2014

<https://exoplayer.dev/>

<https://data-flair.training/>

<https://www.geeksforgeeks.org/>

<https://medium.com/>

<http://europeanschoolradio.eu/>

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

<https://jsoup.org/>

<https://square.github.io/picasso/>

<https://www.techotopia.com/>

<https://documenter.getpostman.com/>

<https://www.javatpoint.com/>

<https://www.edureka.co/>

<https://codinginfinite.com>

<https://code.tutsplus.com/>

ΠΑΡΑΡΤΗΜΑΤΑ

ΠΑΡΑΡΤΗΜΑ Α - Ανάλυση json

Κώδικας ανάλυσης αρχείων json και πέρασμα σε λίστα (ενδεικτικά, παρακάτω θα γίνει parse σε 3 πεδία, id, starttime, date). Η κλάση Details περιέχει τις πληροφορίες που θα γίνονται parse, δηλαδή σ αυτή την περίπτωση δηλώνονται οι μεταβλητές id, starttime και date και ακολουθεί ένας δομητής (constructor) και οι μέθοδοι get και set για κάθε μεταβλητή.

```
ArrayList<Details> details_of_json = new ArrayList<>();

private RequestQueue mQueue;

mQueue = Volley.newRequestQueue(this);

StringRequest response = new StringRequest(Request.Method.GET, json_podcasts_link, new
Response.Listener<String>() {

    @Override

    public void onResponse(String response) {

        int i;

        id.clear(); //δηλώνονται πιο πάνω σαν ArrayList<String>

        starttime.clear();

        date.clear();

        try {

            JSONArray jsonArray = new JSONArray(response);

            if (jsonArray.length() != 0) {

                for (i = 0; i < jsonArray.length(); i++) {

                    JSONObject obj = jsonArray.getJSONObject(i);

                    //Get the production's id

                    String mp3_id = obj.getString("mp3_id");

                    id.add(mp3_id);

                    String desc = obj.getString("description");

                    description.add(desc);

                    String d = obj.getString("broadcast_day");
```

```
        date.add(d);

        String t = obj.getString("start_time");

        time.add(t);

        Details details = new Details(id.get(i),date.get(i),time.get(i));

        details_of_json.add(details);

        Adapter adapter = new Adapter(con_this, details);

        list.setAdapter(adapter);

    }

    } else {

        list.setAdapter(null);

    }

} catch (JSONException e) {

    e.printStackTrace();

}

}

}, new Response.ErrorListener() {

@Override

public void onErrorResponse(VolleyError error) {

}

});

mQueue.add(response);
```

ΠΑΡΑΡΤΗΜΑ Β – Αλγόριθμος υλοποίησης mp3 streaming με exoplayer

```
private RenderersFactory renderersFactory;

private SimpleExoPlayer player;

private DataSource.Factory dataSourceFactory;

private ExtractorsFactory extractorsFactory;

private MediaSource mediaSource;

rendersFactory=new
DefaultRenderersFactory(getApplicationContext(),null,DefaultRenderersFactory.EXTENSION_REN
DERER_MODE_OFF);

    player = ExoPlayerFactory.newSimpleInstance(this,trackSelector);

    dataSourceFactory = new DefaultDataSourceFactory(getApplicationContext(), "Exoplayer");

    extractorsFactory = new DefaultExtractorsFactory();

    mediaSource = new ExtractorMediaSource(Uri.parse(streamUrl),

        dataSourceFactory,

        extractorsFactory,

        null, null);

    player.prepare(mediaSource);
```

ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ

Σ' αυτή την πτυχιακή εργασία σχεδιάστηκε και υλοποιήθηκε η επίσημη εφαρμογή του European School Radio. Πλέον, οι χρήστες του μαθητικού ραδιοφώνου αποκτούν ένα νέο μέσο πρόσβασης στις αγαπημένες λειτουργίες που συνήθιζαν να χρησιμοποιούν μέχρι σήμερα από την ιστοσελίδα. Μέσα από την εφαρμογή του European School Radio, οι χρήστες μπορούν να ακούσουν τις αγαπημένες τους εκπομπές αλλά και να περιηγηθούν στις πιο σημαντικές λειτουργίες του μαθητικού ραδιοφώνου. Η εφαρμογή θα γίνει διαθέσιμη και στο Play Store, και θα μπορεί να χρησιμοποιηθεί από όλους τους χρήστες έξυπνων τηλεφώνων με λειτουργικό Android.

Ξεκινώντας, όταν ο χρήστης μπαίνει στην εφαρμογή, εμφανίζεται το μενού με τις διαθέσιμες επιλογές, όπως φαίνεται παρακάτω.

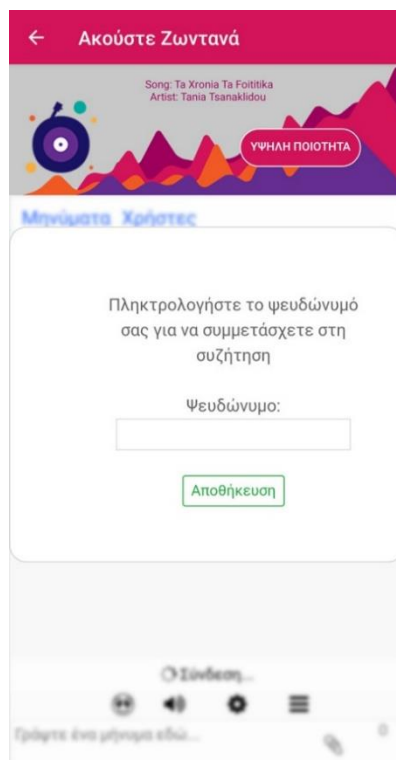


Εικόνα 1 " Μενού Εφαρμογής"

Οι διαθέσιμες επιλογές είναι οι εξής:

- Ακούστε Ζωντανά

Σε αυτή την λειτουργία, ο χρήστης μπορεί να ακούσει το καθημερινό ραδιοφωνικό πρόγραμμα 24 ώρες το 24ωρο, 7 ημέρες την εβδομάδα. Μπορεί επίσης να επικοινωνεί με τους άλλους ακροατές ή ραδιοφωνικούς παραγωγούς μέσα από την ζωντανή συνομιλία που έχει.



Εικόνα 2 " Ακούστε Ζωντανά"

Πιο συγκεκριμένα, μπορεί να πατήσει επάνω στο βινύλιο (πάνω αριστερά) για να ξεκινήσει η αναπαραγωγή του ήχου του European School Radio. Όταν ξεκινήσει η αναπαραγωγή του ήχου, το βινύλιο θα περιστρέφεται συνεχώς. Για διακοπή του ήχου, αρκεί να ξαναπατήσει επάνω. Επίσης, μπορεί να επιλέξει να ακούσει τον ήχο σε υψηλή ή χαμηλή ποιότητα. Η χαμηλή ποιότητα βοηθάει στην εξοικονόμηση όγκου δεδομένων που χρησιμοποιούνται κατά την λήψη και αναπαραγωγής της ζωντανής μετάδοσης του ήχου. Κατά την είσοδο σε αυτή την λειτουργία, όταν το κινητό είναι συνδεδεμένο με W-Fi είναι προεπιλεγμένη η υψηλή ποιότητα, ενώ όταν είναι συνδεδεμένο μέσω δεδομένων κινητής τηλεφωνίας είναι προεπιλεγμένη η χαμηλή ποιότητα. Φυσικά, μέσω του κουμπιού αλλαγής ποιότητας ο χρήστης μπορεί να αλλάξει την ποιότητα ανά πάσα στιγμή.

Επίσης, κάτω στην συνομιλία, πληκτρολογώντας απλά ένα ψευδώνυμο (username) μπορεί να επικοινωνεί με όλο τον κόσμο του μαθητικού ραδιοφώνου. Τέλος, όταν είναι σε εξέλιξη η αναπαραγωγή του ήχου στην εφαρμογή, πηγαίνοντας πίσω στο μενού δίνεται η επιλογή στον χρήστη να συνεχίσει να ακούει ή όχι, μέσω ενός pop-up παραθύρου που εμφανίζεται με τις επιλογές ναι και όχι (AlertDialog συγκεκριμένα). Εάν συνεχίσει να ακούει, εμφανίζεται ένα notification bar που περιέχει το όνομα του καλλιτέχνη και τον τίτλο του τραγουδιού (και ανανεώνεται αυτόματα), όπου εάν πατηθεί από τον χρήστη τον πηγαίνει στην εφαρμογή και συγκεκριμένα στην λειτουργία «Ακούστε Ζωντανά».

Τέλος, να σημειωθεί πως για την σωστή λειτουργία της εφαρμογής και ειδικά της λειτουργίας «Ακούστε Ζωντανά», είναι απαραίτητη η ενεργή σύνδεση στο διαδίκτυο είτε μέσω Wi-Fi είτε μέσω δεδομένων κινητής τηλεφωνίας.

➤ Νέα

Σ αυτή την λειτουργία, ο χρήστης μπορεί να διαβάσει όλα τα νέα του μαθητικού ραδιοφώνου. Εδώ ανακοινώνονται διαγωνισμοί, νέες εκπομπές αλλά και οτιδήποτε άλλο αφορά το μαθητικό ραδιόφωνο. Επίσης, πατώντας πάνω σε κάποιο ραδιοφωνικό νέο μπορεί να το διαβάσει αναλυτικά.



Εικόνα 3 " Ραδιοφωνικά νέα"

➤ Επόμενες Εκπομπές

Σ αυτή την λειτουργία, ο χρήστης μπορεί να δει τις επόμενες εκπομπές που πρόκειται να παίξουν στο ραδιόφωνο. Επίσης, πατώντας πάνω σε κάποια εκπομπή μπορεί να δει περισσότερες πληροφορίες γι' αυτή, όπως περιγραφή εκπομπής, τύπος εκπομπής κτλ. Αυτή η λειτουργία είναι διαφορετική από την λειτουργία του Ραδιοφωνικού Προγράμματος. Κατά την είσοδο στο Ραδιοφωνικό Πρόγραμμα, εμφανίζονται οι εκπομπές της σημερινής ημέρας, ενώ αν δεν υπάρχει εκπομπή εμφανίζεται κατάλληλο μήνυμα. Σ' αυτή την λειτουργία, εμφανίζονται οι επόμενες εκπομπές, ακόμα κι αν απέχουν μέρες μεταξύ τους.

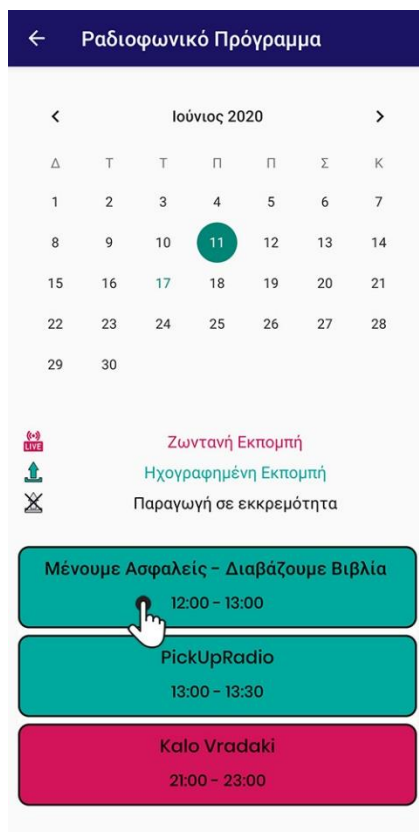


Εικόνα 4 " Επόμενες Εκπομπές"

➤ Πρόγραμμα

Εδώ, ο χρήστης μπορεί να δει τις εκπομπές που θα παίξουν μια συγκεκριμένη μέρα ή μήνα. Πατάει στα βελάκια δεξιά και αριστερά του μήνα εάν θέλει να αλλάξει μήνα και στην συνέχεια πάνω σε μία συγκεκριμένη ημέρα στο πρόγραμμα, και του εμφανίζονται σε λίστα οι εκπομπές. Έπειτα, όπως φαίνεται και στην εικόνα, μπορεί να πατήσει πάνω σε μία εκπομπή που επιθυμεί και να δει περισσότερες πληροφορίες, όπως την περιγραφή και την εικόνα της εκπομπής, να ακούσει το σποτ της, καθώς επίσης και να ακούσει όλα τα επεισόδια που έχουν ανέβει.

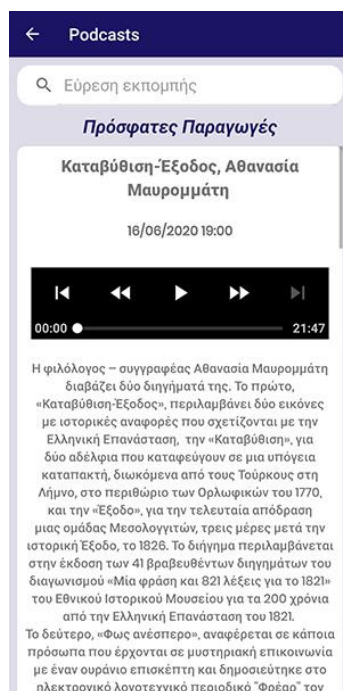
Όπως φαίνεται, οι εκπομπές χωρίζονται σε 3 μέρη. Σε ζωντανές εκπομπές, σε ηχογραφημένες, δηλαδή στις ηχογραφημένες εκπομπές που έχει ανέβει επεισόδιο και θα παίξει κανονικά, και στις παραγωγές σε εκκρεμότητα, που σημαίνει ότι ενώ έχει προγραμματιστεί επεισόδιο δεν έχει ανέβει επεισόδιο, επομένως δεν θα πραγματοποιηθεί εάν δεν ανέβει επεισόδιο μέχρι την ημέρα που έχει οριστεί.



Εικόνα 5 " Πρόγραμμα Εκπομπών"

➤ Παραγωγές

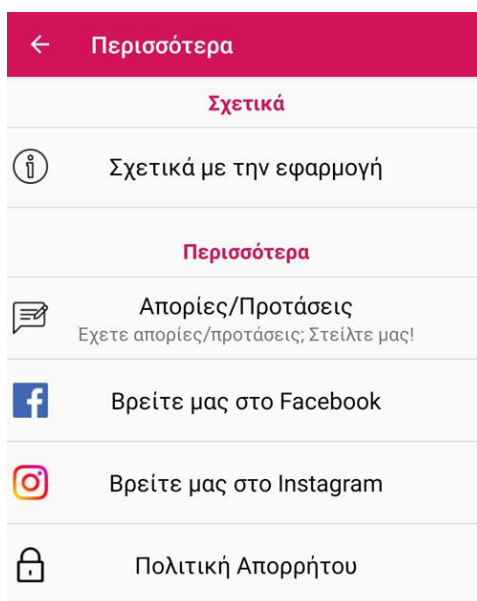
Στις παραγωγές, όταν μπαίνει την πρώτη φορά ο χρήστης, εμφανίζονται τα πρόσφατα επεισόδια που ανέβηκαν. Επίσης, μπορεί μέσω της αναζήτησης να αναζητήσει μία εκπομπή και να του εμφανιστούν όλα τα αποτελέσματα που ψάχνει.



Εικόνα 6 " Παραγωγές μαθητών"

➤ Περισσότερα

Σ' αυτή την λειτουργία, ο χρήστης αρχικά μπορεί να διαβάσει κάποιες πληροφορίες σχετικά με την εφαρμογή καθώς και κάποιες οδηγίες χρήσης της. Επίσης, μπορεί μέσω μιας φόρμας να επικοινωνήσει με το European School Radio για ό,τι θέμα επιθυμεί, γράφοντας το ονοματεπώνυμο του, το θέμα του μηνύματος και το περιεχόμενο. Τέλος, μπορεί να συνδεθεί με τους λογαριασμούς του μαθητικού ραδιοφώνου στα μέσα κοινωνικής δικτύωσης καθώς και να διαβάσει την πολιτική απορρήτου και το έγγραφο προστασίας δεδομένων των μελών του ραδιοφώνου.



Εικόνα 7 "Περισσότερα σχετικά με το Ραδιόφωνο "