|  |  |
| --- | --- |
| ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ [University project – 7th semester] | **[Η εφαρμογή διαθέτει ένα σύνολο από παιδικές ιστορίες (ή παραμύθια), τις οποίες ο χρήστης μπορεί να «διαβάζει» χρησιμοποιώντας τη μηχανή ομιλίας του κινητού τηλεφώνου] - Ακαδημαϊκό έτος: 2024-2025** |

Περιεχόμενα

[1.Εισαγωγή2](#a1)

[2.Περιγραφή της εφαρμογής 2](#a2)

[3.Αρχιτεκτονική 2](#a3)

[Δομή εφαρμογής2](#a3_1)

[Χρήση Βάσεων Δεδομένων2](#a3_2)

[Γλωσσική υποστήριξη 3](#a3_3)

[**4.Λειτουργικότητα – Διεπαφή χρήστη** **4**](#a4)

[Log in – Register 4](#a4_1)

[Επιλογή ιστορίας 4](#a4_2)

[Εμφάνιση στατιστικών 5](#a4_4)

[SlidePanel 6](#a4_5)

[**5.Τεχνικές υλοποίησης6**](#a5)

[Κώδικας και βασικά modules 6](#a5_1)

[Διαχείριση βάσης δεδομένων 8](#a5_2)

[Χρήση API Text-to-Speech 9](#a5_3)

[6.Github Repository 9](#github)

**1.****Εισαγωγή**

* **Σκοπός της εφαρμογής** είναι η υλοποίηση μιας Android εφαρμογής για κινητά, όπου ο χρήστης θα μπορεί να διαλέξει ανάμεσα σε παιδικές ιστορίες που βρίσκονται σε Firebase Cloud Remote Database, χρησιμοποιώντας τη μηχανή ομιλίας του κινητού τηλεφώνου.
* Απευθύνεται κυρίως σε παιδιά και γονείς.

**2.****Περιγραφή της εφαρμογής**

* Log in & Register & Log out (Authentication user with Firebase[Email & password]).
* Υπάρχουν τουλάχιστον 5 αποθηκευμένες ιστορίες στο Firestore(Τίτλος, εικόνα, κείμενο, συγγραφέας & έτος συγγραφής)
* Χρήση TextToSpeech, με λειτουργία start/pause για να ακούει την ιστορία ο χρήστης.
* Υποστήριξη τριών γλωσσών: Ελληνικά, Αγγλικά & Γερμανικά(Η εφαρμογή μας ακολουθεί την γλώσσα που ακολουθεί το εκάστοτε κινητό).
* Προβολή Στατιστικών(Αγαπημένες ιστορίες ανά χρήστη και πόση ώρα έπαιξε κάθε ιστορία).
* Υποστήριξη εισαγωγής εικόνας από τοπική συλλογή για εικόνα Προφίλ.
* SlidePanel(Fragment), που ανοίγει με το πάτημα της εικόνας χρήστη πάνω δεξιά.
* Library Fragment(Ο χρήστης μπορεί να πλογηθεί στις αγαπημένες του ιστορίες)
* Search Fragment(Δεν έχει ολοκληρωθεί).

**3.****Αρχιτεκτονική**

* Δομή εφαρμογής:
* Το Main Activity είναι ο κορμός, εδώ βρίσκονται η Bottom Navigation και η εικόνα χρήστη πάνω δεξιά. Τα υπόλοιπα είναι Fragments(Home, Search, Library, SlidePanel & Player)
* Για το Authentication user έχουμε τις κλάσεις Login.java, Register.java .
* Statistics activity που φορτώνει τα δεδομένα από την βάση με την βοήθεια της statisticsItem και StatisticsAdapter.
* Χρήση Βάσεων Δεδομένων:
* Για το Authentication user χρησιμοποιούμε Firebase(Email&password)

A screenshot of a computer

Description automatically generated

* Για την αποθήκευση των ιστοριών έχουμε ένα collection “Stories” και ένα document για κάθε ιστορία με 5 πεδία(title, author, year, text, imageURL).

**A screenshot of a computer

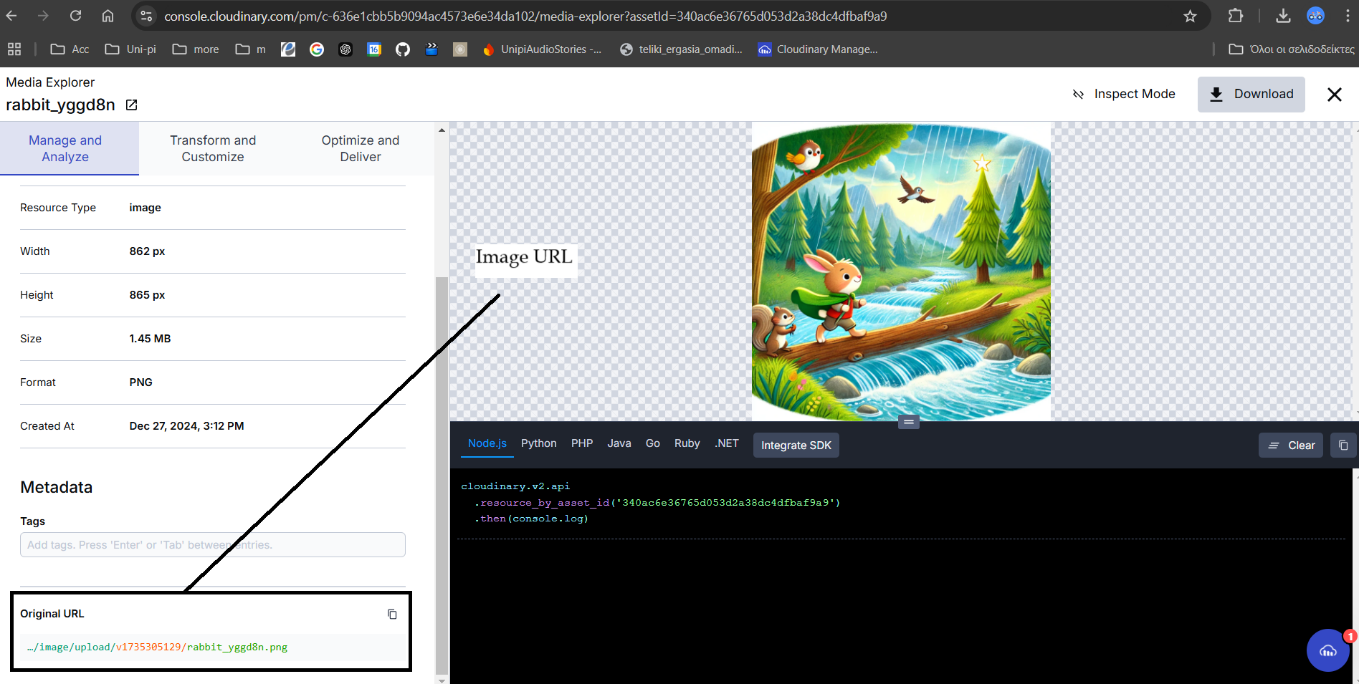
Description automatically generated**

* Στο collection “statistics” κάθε document έχει το documentID του εκάστοτε user με 3 πίνακες.Ο listatingCount πόσες φορές έχει ακουστει η κάθε ιστορία, ο listeningTime αποθηκεύει τον χρόνο ακρόασης κάθε ιστορίας και ο πίνακας saved περιέχει τα αποθηκευμένα/αγαπημένα stories.

**A screenshot of a computer

Description automatically generated**

* Χρησιμοποιήσαμε το cloudinary.com για να ανακτά απο εκεί η εφαρμογή μας τις παιδικές εικόνες.



* Γλωσσική υποστήριξη:
* Εκτός του βασικού αρχείου strings.xml που δημιουργεί το Android Studio, δημιουργήσαμε 2 ακόμα, ένα για γερμανικά και ένα για ελληνικά.

A screenshot of a computer

Description automatically generated

**4.****Λειτουργικότητα – Διεπαφή χρήστη**

* Log in – Register :

|  |  |
| --- | --- |
|  |  |

* Επιλογή ιστορίας. Οταν ο χρήστης διαλέξει μία από τις διαθέσιμες ιστορίες(αριστερά) ανοίγει το PlayerFragment και βλέπει τις δυνατοτητες να ακούσει ή να αποθηκεύσει την ιστορία(δεξιά)

|  |  |
| --- | --- |
|  |  |

* Εμφάνιση στατιστικών:

Στο PlayerFragment.java κάθε φορά που ο χρήστης ξεκινά ή σταματά μία ιστορία καταγράφονται τα milliseconds που άκουσε με την βοήθεια της μεθόδου currentTimeMillis()

A screen shot of a computer program

Description automatically generated

Στην Statistics θα φορτωθούν τα δεδομένα αυτα για να προβληθούν στην xml.

* SlidePanel :

A screenshot of a video game

Description automatically generated

**5.****Τεχνικές υλοποίησης**

* Κώδικας και βασικά modules:

**MainActivity.java:**

Κατά το άνοιγμα της Main Activity ελέγχεται εάν υπάρχει συνδεδεμένος χρήστης(εάν όχι θα πρέπει να συνδεθεί πρώτα):

A screen shot of a computer

Description automatically generated

Αμέσως μετά θα αντικατασταθείτο FrameLayout της Mainactivity για να μπει το HomeFragment μέσα στο container:

A screen shot of a computer

Description automatically generated

Στη συνέχεια βλέπουμε ότι με βάση το itemId, βρίσκουμε ποιο fragment θέλει ο χρήστης:

A computer screen shot of a program code

Description automatically generated

**HomeFragment.java:**

Κατά της φόρτωση της κλάσης αυτής καλείται η loadAllDocuments(), που είναι υπεύθυνη να τραβήξει τα δεδομένα από την Firebase.

A screen shot of a computer program

Description automatically generated  
Κάθε ιστορία που ανακτά η loadAllDocuments() την στέλνει στην addDataToView(String title, String imageUrl, String documentId) για να την εμφανισει μέσα στο container(Scroll) με την μορφή CardView.

Αξίζει να σημειωθεί και ότι με την βοήθεια του SetOnClickListener, όταν ο χρήστης επιλέξει την ιστορία που τον ενδιαφέρει, καλούμε την *newInstance*(imageUrl, text, title, author, year, documentId) της PlayerFragment που με τη σειρά της μπαίνει στο container.

A screenshot of a computer program

Description automatically generated

* Διαχείριση βάσης δεδομένων (εισαγωγή/ανάκτηση δεδομένων ιστοριών):  
  Η εισαγωγή δεδομένων έγινε απευθείας από το firebase console.

Η ανάκτηση δεδομένων έγινε από ένα αντικείμενο FirebaseFirestore ‘db’ , όπως φαίνεται [εδώ](#loadAllDocuments)

* Χρήση API Text-to-Speech: Για καλύτερη οργάνωση δημιουργήσαμε το MyTts.java που περιλαμβάνει τις μεθόδους:
* MyTts constructor: Αρχικοποιεί τη μηχανή Text-to-Speech (TTS) και ορίζει την προεπιλεγμένη γλώσσα σε Αγγλικά.
* speak(String message): Μετατρέπει το κείμενο σε ομιλία και το προσθέτει στην ουρά αναπαραγωγής.
* stopSpeaking(): Σταματά αμέσως την τρέχουσα ομιλία και καθαρίζει την ουρά αναπαραγωγής.
* isSpeaking(): Ελέγχει αν η μηχανή TTS βρίσκεται σε διαδικασία ομιλίας και επιστρέφει true ή false.

**6.****Github Repository**

Πατήστε [εδώ](https://github.com/Georgechrp/Unipi-Audio-Stories).