



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 «διαβάσει» χρησιμοποιώντας τη μηχανή ομιλίας του κινητού τηλεφώνου] - Ακαδημαϊκό έτος: 2024-2025

ΟΜΑΔΑ	ΓΙΩΡΓΟΣ ΧΡΙΣΤΟΠΟΥΛΟΣ-Π20206
	ΚΥΡΙΑΚΟΣ ΚΙΧΗΣ – Π20013
	ΧΡΙΣΤΟΣ ΞΥΔΗΣ -Π20008
ΥΠΕΥΘΥΝΟΣ ΚΑΘΗΓΗΤΗΣ	ΕΥΘΥΜΙΟΣ ΑΛΕΠΗΣ

## Περιεχόμενα

<b>1.Εισαγωγή .....</b>	<b>2</b>
<b>2.Περιγραφή της εφαρμογής .....</b>	<b>2</b>
<b>3.Αρχιτεκτονική .....</b>	<b>2</b>
Δομή εφαρμογής .....	2
Χρήση Βάσεων Δεδομένων .....	2
Γλωσσική υποστήριξη .....	3
<b>4.Λειτουργικότητα – Διεπαφή χρήστη .....</b>	<b>4</b>
Log in – Register .....	4
Επιλογή ιστορίας .....	4
Εμφάνιση στατιστικών .....	5
SlidePanel .....	6
<b>5.Τεχνικές υλοποίησης .....</b>	<b>6</b>
Κώδικας και βασικά modules .....	6
Διαχείριση βάσης δεδομένων .....	8
Χρήση API Text-to-Speech .....	9
<b>6.Github Repository .....</b>	<b>9</b>

## 1.Εισαγωγή

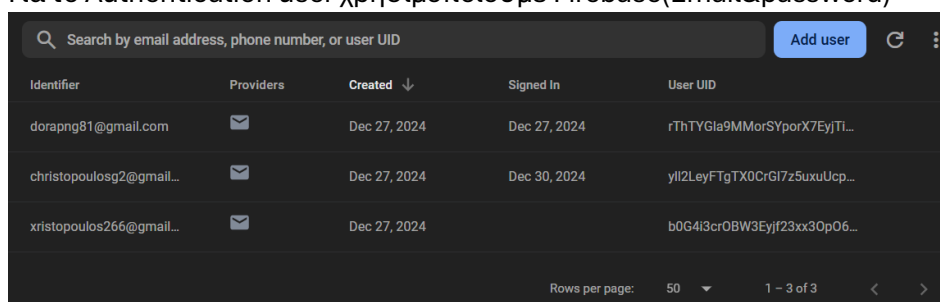
- ✓ **Σκοπός της εφαρμογής** είναι η υλοποίηση μιας Android εφαρμογής για κινητά, όπου ο χρήστης θα μπορεί να διαλέξει ανάμεσα σε παιδικές ιστορίες που βρίσκονται σε Firebase Cloud Remote Database, χρησιμοποιώντας τη μηχανή ομιλίας του κινητού τηλεφώνου.
- ✓ Απευθύνεται κυρίως σε παιδιά και γονείς.

## 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 πλοηγηθεί στις αγαπημένες του ιστορίες)
- ✓ Search Fragment(Δεν έχει ολοκληρωθεί).

## 3.Αρχιτεκτονική

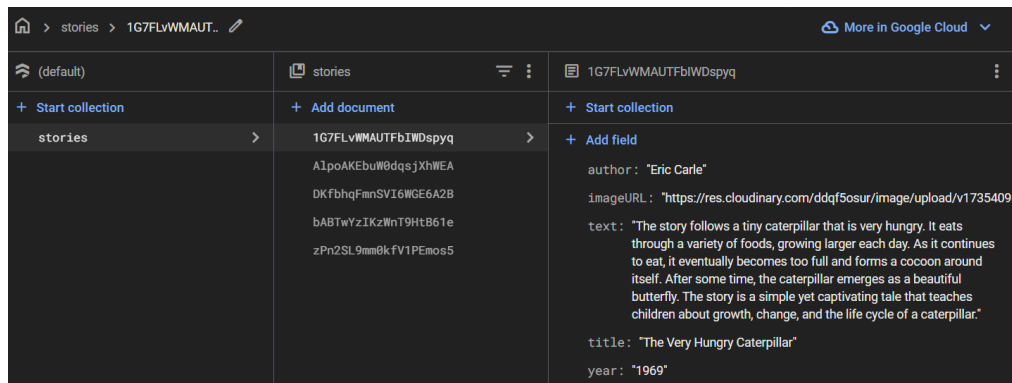
- Δομή εφαρμογής:
  - Το Main Activity είναι ο κορμός, εδώ βρίσκονται η Bottom Navigation και η εικόνα χρήστη πάνω δεξιά. Τα υπόλοιπα είναι Fragments(Home, Search, Library, SlidePanel & Player)
  - Για το Authentication user έχουμε τις κλάσεις Login.java, Register.java .
  - Statistics activity που φορτώνει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



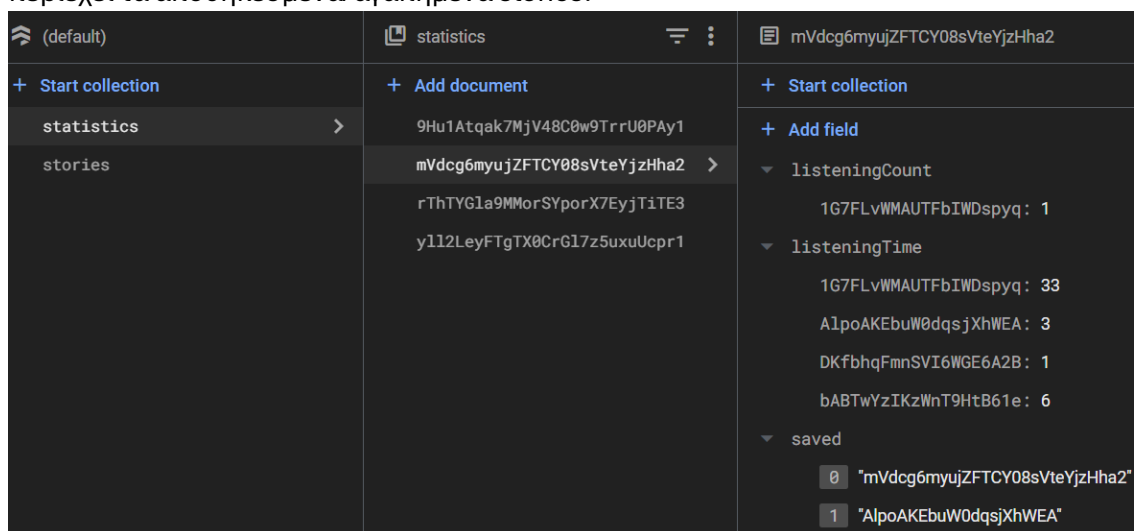
Identifier	Providers	Created ↓	Signed In	User UID
dorapng81@gmail.com	📧	Dec 27, 2024	Dec 27, 2024	rThTYGla9MMorSYporX7EjTl...
christopoulosg2@gmail...	📧	Dec 27, 2024	Dec 30, 2024	ylI2LeyFTgTX0CrGi7z5uxuUcp...
xristopoulos266@gmail...	📧	Dec 27, 2024		b0G4i3cr0BW3Eylf23xx30p06...

Rows per page: 50 1 - 3 of 3

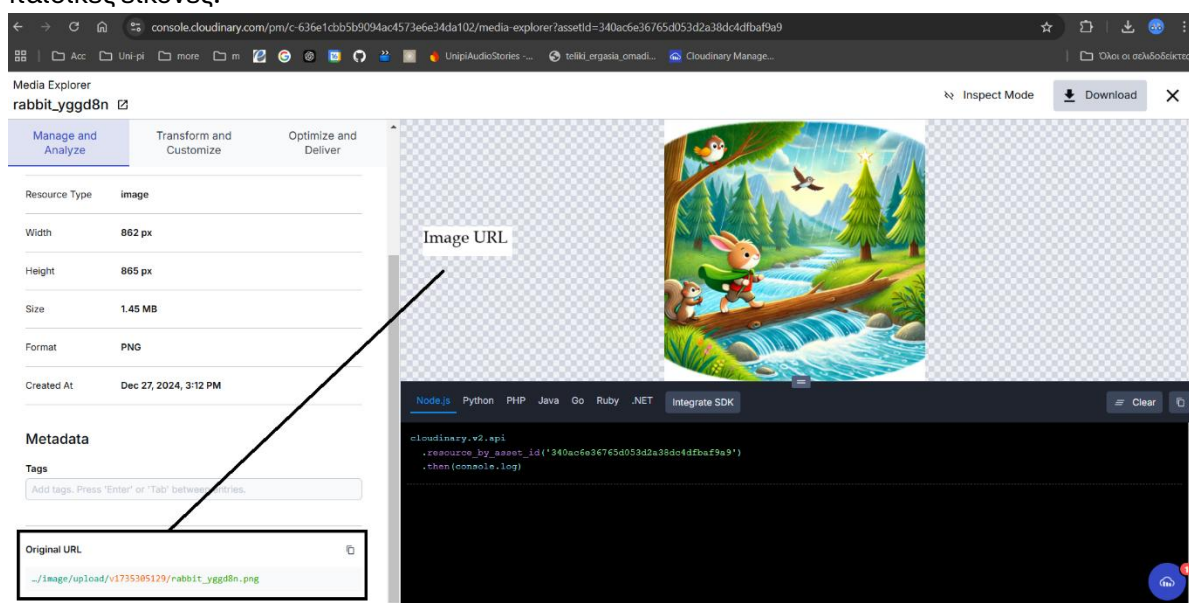
- Για την αποθήκευση των ιστοριών έχουμε ένα collection “Stories” και ένα document για κάθε ιστορία με 5 πεδία(title, author, year, text, imageURL).



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

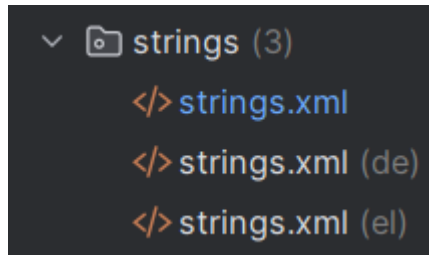


- Χρησιμοποιήσαμε το cloudinary.com για να ανακτά από εκεί η εφαρμογή μας τις παιδικές εικόνες.



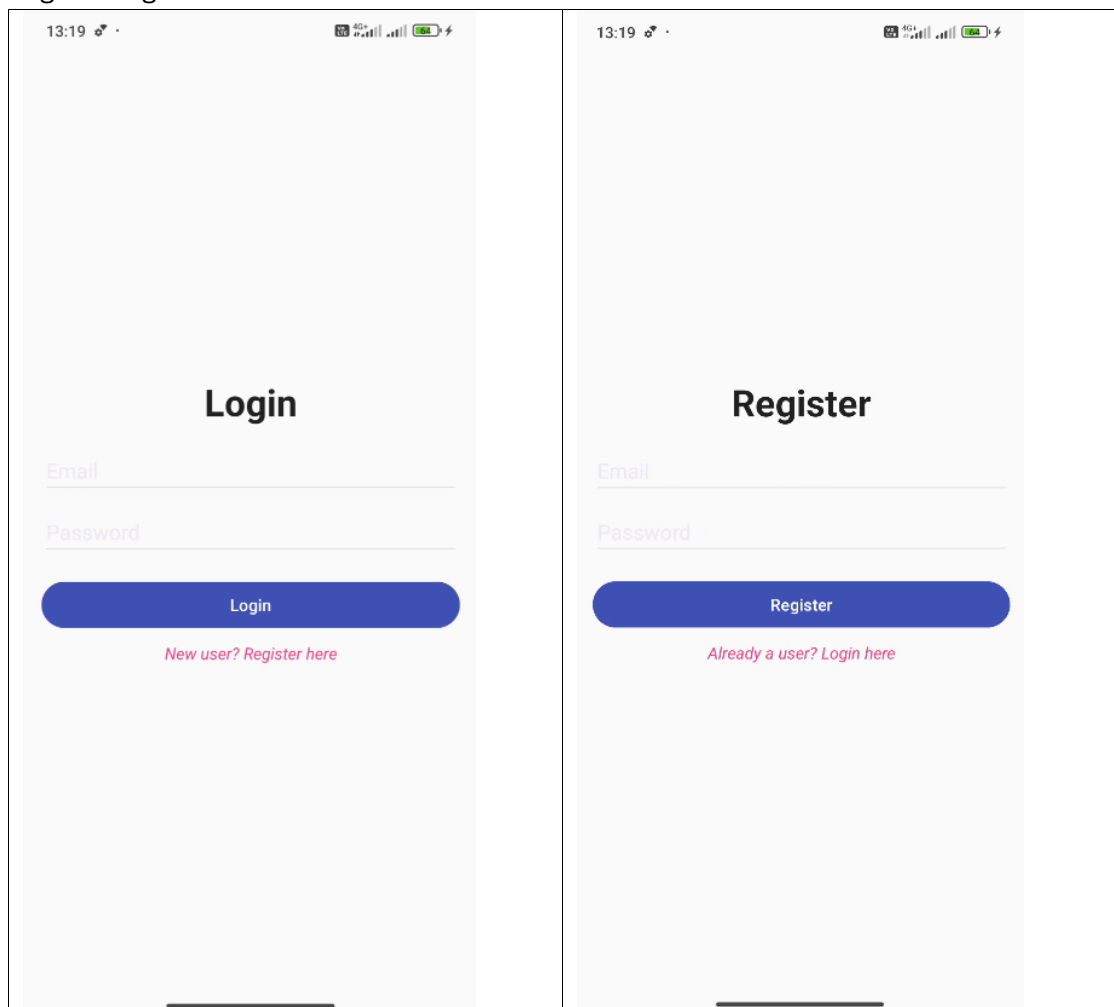
➤ Γλωσσική υποστήριξη:

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

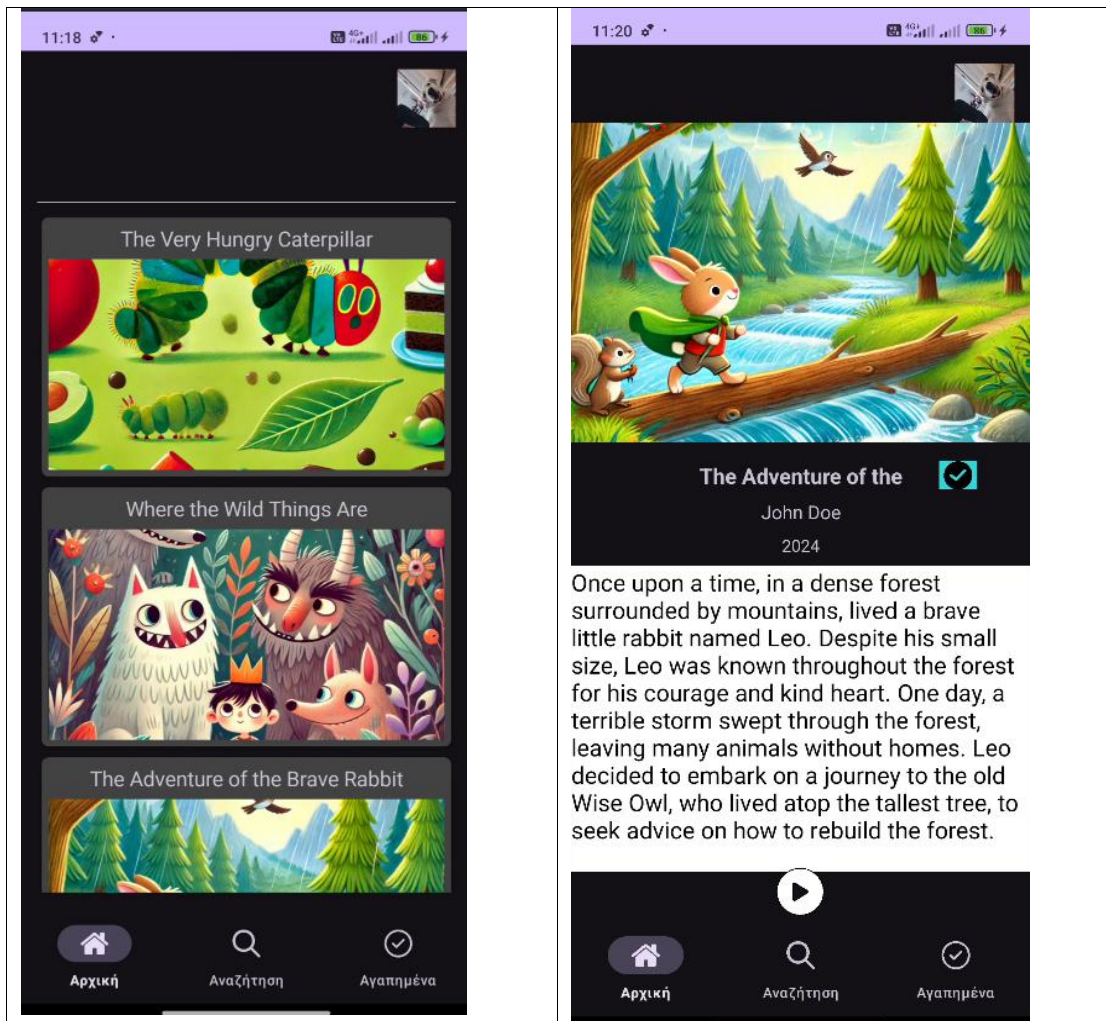


#### 4.Λειτουργικότητα – Διεπαφή χρήστη

- Log in – Register :



- Επιλογή ιστορίας. Όταν 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()

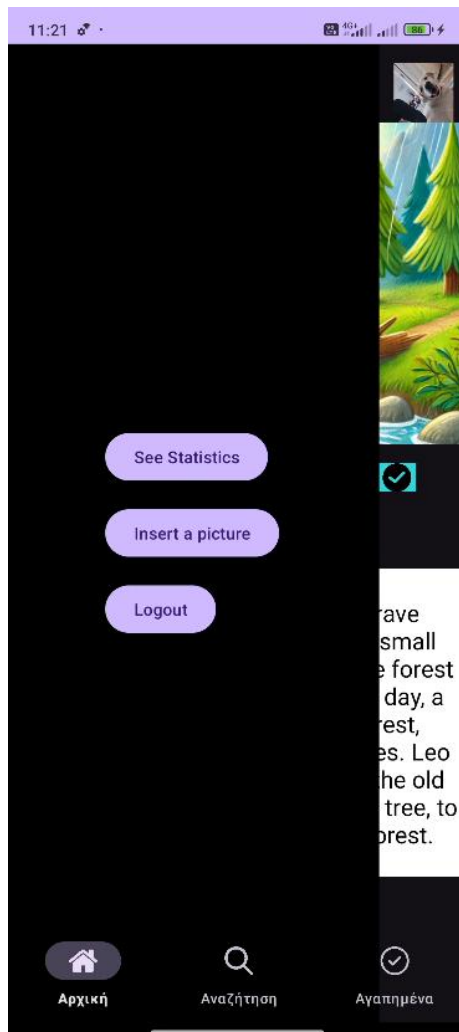
```
private void pauseSpeaking() {
    if (isPlaying) {
        long currentTime = System.currentTimeMillis();
        long elapsedTime = (currentTime - startTime) / 1000;
        totalListeningTime += elapsedTime;
        startTime = 0;

        recordListeningTime(user.getUid(), documentId, elapsedTime);

        isPlaying = false;
        myTts.stopSpeaking();
        toggleIcons();
    } else {
        Toast.makeText(getContext(), text: "Nothing is being spoken", Toast.LENGTH_SHORT).show();
    }
}
```

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

➤ SlidePanel :



## 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):

```
auth = FirebaseAuth.getInstance();
user = auth.getCurrentUser();
if (user == null) {
    Intent intent = new Intent( packageContext: this, Login.class);
    startActivity(intent);
    return;
}
```

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

```
getSupportFragmentManager().beginTransaction()
    .replace(R.id.container, new HomeFragment()) |
    .commit();
```

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

```
BottomNavigationView bottomNavigationView = findViewById(R.id.bottom_navigation);
bottomNavigationView.setOnNavigationItemSelectedListener(item -> {
    int itemId = item.getItemId();

    if (itemId == R.id.nav_home) {
        getSupportFragmentManager().beginTransaction()
            .replace(R.id.container, new HomeFragment())
            .commit();
        return true;
    } else if (itemId == R.id.nav_search) {
        getSupportFragmentManager().beginTransaction()
            .replace(R.id.container, new SearchFragment())
            .commit();
        return true;
    } else if (itemId == R.id.nav_library) {
        getSupportFragmentManager().beginTransaction()
            .replace(R.id.container, new LibraryFragment())
            .commit();
        return true;
    } else {
        return false;
    }
});
```

### HomeFragment.java:

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



```

private void loadAllDocuments() {
    db.collection( collectionPath: "stories") CollectionReference
        .get() Task<QuerySnapshot>
        .addOnCompleteListener(task -> {
            if (task.isSuccessful()) {
                for (QueryDocumentSnapshot document : task.getResult()) {
                    String title = document.getString( field: "title");
                    String imageUrl = document.getString( field: "imageUrl");
                    text = document.getString( field: "text");
                    author = document.getString( field: "author");
                    year = document.getString( field: "year");
                    String documentId = document.getId();

                    addDataToView(title, imageUrl, documentId);

                }
            } else {
                Log.w(TAG, msg: "Error", task.getException());
            }
        });
}

```

Κάθε ιστορία που ανακτά η loadAllDocuments() την στέλνει στην addDataToView(String title, String imageUrl, String documentId) για να την εμφανίσει 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.

```

imageView.setOnClickListener(v -> {
    PlayerFragment playerFragment = PlayerFragment.newInstance(imageUrl, text, title, author, year, documentId);
    requireActivity().getSupportFragmentManager() FragmentManager
        .beginTransaction() FragmentTransaction
        .replace(R.id.container, playerFragment) // Αντικατάσταση του fragment container
        .addToBackStack( name: null) // Προσθήκη στο backstack
        .commit();
});

```

- 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](#)

- 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](#).