



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ - ΣΧΟΛΗ
ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΣΥΓΧΡΟΝΑ ΘΕΜΑΤΑ ΤΕΧΝΟΛΟΓΙΑΣ ΛΟΓΙΣΜΙΚΟΥ

**Android εφαρμογή, προβολής, αγοράς και άμεσων
ειδοποιήσεων για προϊόντα της αγοράς (εάν το
κατάστημα βρίσκεται κοντά στον χρήστη)**

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

Περιεχόμενα

1.Εισαγωγή.....	2
2.Περιγραφή της εφαρμογής	2
3.Αρχιτεκτονική	2
Δομή εφαρμογής.....	2
Χρήση Βάσεων Δεδομένων	2
Γλωσσική υποστήριξη	4
4.Λειτουργικότητα – Διεπαφή χρήστη	4
Log in – Register	4
Προβολή προϊόντων	5
Αγορά προϊόντος	6
5.Τεχνικές υλοποίησης	7
Κώδικας και βασικά modules	7
Διαχείριση βάσης δεδομένων	9
Χρήση API Text-to-Speech	9
Light/Dark Theme.....	9
Permissions.....	10
6.Github Repository	10

Εισαγωγή

- ✓ **Σκοπός της εφαρμογής** είναι η υλοποίηση μιας Android εφαρμογής για κινητά, όπου ο χρήστης θα μπορεί να διαλέξει ανάμεσα σε προϊόντα που βρίσκονται σε Firebase Cloud Remote Database, χρησιμοποιώντας τη μηχανή ομιλίας του κινητού τηλεφώνου.
- ✓ Απευθύνεται σε όλους και σε ανθρώπους με προβλήματα όρασης.

2.Περιγραφή της εφαρμογής

- ✓ Log in & Register & Log out (Authentication user with Firebase[Email & password]).
- ✓ Υπάρχουν τουλάχιστον 10 αποθηκευμένα προϊόντα στο Firestore(Κωδικός, Τίτλος, Περιγραφή, Ημερομηνία κυκλοφορίας, Τιμή, Τοποθεσία μαγαζιού)
- ✓ Χρήση TextToSpeech, με λειτουργία start/pause για να ακούει την περιγραφή του προϊόντος ο χρήστης.
- ✓ Υποστήριξη τριών γλωσσών: Ελληνικά, Αγγλικά & Γερμανικά(Η εφαρμογή μας ακολουθεί την γλώσσα που ακολουθεί το εκάστοτε κινητό).
- ✓ Ο χρήστης μπορεί να πλοηγηθεί ανάμεσα σε 10 προϊόντα (HomeFragment)
- ✓ Cart Fragment(Ο χρήστης μπορεί να πλοηγηθεί στα αγαπημένα του προϊόντα)
- ✓ Notification Fragment(Δεν έχει ολοκληρωθεί)
- ✓ ProfileFragment(settings profile & logout)

3.Αρχιτεκτονική



- Δομή εφαρμογής:
 - Το Main Activity είναι ο κορμός, εδώ βρίσκεται η Bottom Navigation. Τα υπόλοιπα είναι Fragments(Home, Cart, Notification, Profile).
 - Για το Authentication user έχουμε τις κλάσεις Login.java, Register.java . Επιπλέον χρειάστηκαν οι κλάσεις :
 - MyTts για την εκφώνηση περιγραφής.
 - PreferencesManager για την καλύτερη οργάνωση αποθήκευσης και ανάκτησης των Settings δεδομένων,
 - NotificationHelper για να δημιουργεί και να στέλνει ειδοποιήσεις.

Χρήση Βάσεων Δεδομένων:

- Για το Authentication user χρησιμοποιούμε Firebase(Email&password)

Search by email address, phone number, or user UID

Add user

Identifier	Providers	Created ↓	Signed In	User UID
georgechrp2@gmail.com		Jan 6, 2025	Jan 6, 2025	waKhb08hQUTz62S4Yn6DsaT...
christopoulosg2@gmail...		Jan 3, 2025	Jan 5, 2025	uqfiV9jgc6ZMueHuGmfOVt3v...

- Για την αποθήκευση των προϊόντων έχουμε ένα collection “products” και ένα document για κάθε ένα με 5 πεδία(code, description, imageURL, location, name, price, release_date).

products > JLe6H208D6R0rqaAoZih		
(default)	products	JLe6H208D6R0rqaAoZih
+ Start collection	+ Add document	+ Start collection
orders	CJdaF7q85qVF3CUWwPkv	+ Add field
products >	JLe6H208D6R0rqaAoZih >	code: "54851624"
users_data	Mk9s1Pr1Wh0SovA614yK	description: "Χρόνος Λειτουργίας 7 ώρες Λειτουργία με Θήκη Φόρτισης 30 ώρες Active Noise Cancellation Αυτόματη Προσαρμογή Θορύβου Bluetooth 5.4 Φωτισμός Blade"
	NbX9pt1FqFLNAujbx2BH	imageURL: "https://res.cloudinary.com/ddqf5osur/image/upload/v1735994"
	a7mWsuTeuhL0qPVyydSb	location: [36.94493° N, 21.66536° E]
	bW1pq3lUdXtu4r1G9Mgu	name: "Samsung Galaxy Buds3 Pro Bluetooth Handsfree Ακουστικά με Αντοχή στον Ιδρώτα και Θήκη Φόρτισης Ασημί"
	dM2xt0zTUorznd1AJZqr	price: "188,00 €"
	fgx4YjVmYH2z09P2jApL	release_date: December 17, 2024 at 6:47:01 PM UTC+2
	mKb5hCv9kEhGFXLPjZiy	
	uA8SfQIA9dEIi6jq1AId	

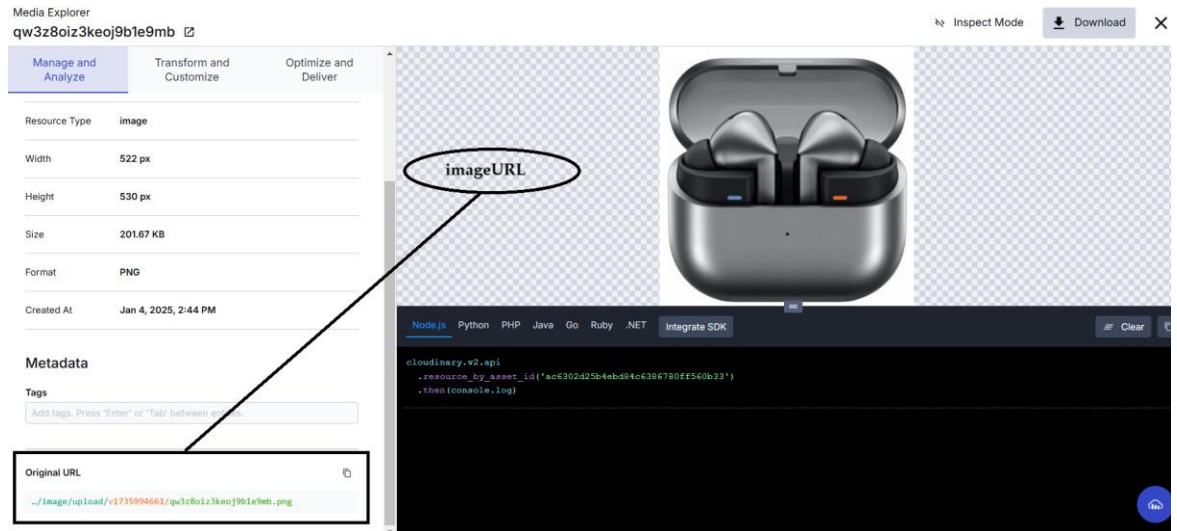
- Στο collection “users_data” κάθε document έχει το documentID του εκάστοτε user με 1 πεδίο array. Ο πίνακας favorites περιέχει τα id των αγαπημένων ιστοριών του κάθε χρήστη

users_data > uqfiV9jgc6ZMue		
(default)	users_data	uqfiV9jgc6ZMueHuGmfOVt3va4t2
+ Start collection	+ Add document	+ Start collection
orders	uqfiV9jgc6ZMueHuGmfOVt3va4t2 >	+ Add field
products	waKhb08hQUTz62S4Yn6DsaTSXTC2	favorites
users_data >		0 "Mk9s1Pr1Wh0SovA614yK"
		1 "CJdaF7q85qVF3CUWwPkv"
		2 "bW1pq3lUdXtu4r1G9Mgu"
		3 "dM2xt0zTUorznd1AJZqr"

- Στο collection “orders” αποθηκεύονται οι παραγγελίες μόλις ο χρήστης πατήσει “ΑΓΟΡΑ” με τα εξής πεδία

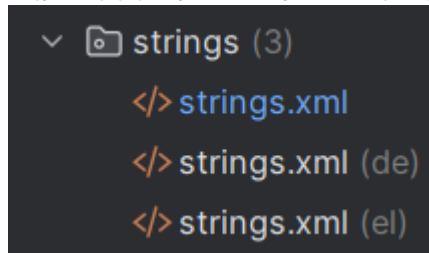
+ Start collection	+ Add document	+ Start collection
orders >	ERoXnmvjUsHdgtLAWjY7 >	+ Add field
products	EV6mQkHG6SQoxQp0cBC0	customerName: "Φοίβος Δεληβοριάς"
users_data	F6EKHKDHLTWvPzqVoBJR	documentId: "mKb5hCv9kEhGFXLPjZiy"
	GTu1b91Y58Utp0wgazaV	productCode: "57221691"
	JOIEVot6EqLTFG4w013q	timestamp: January 6, 2025 at 6:43:27 PM UTC+2

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



➤ Γλωσσική υποστήριξη:

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

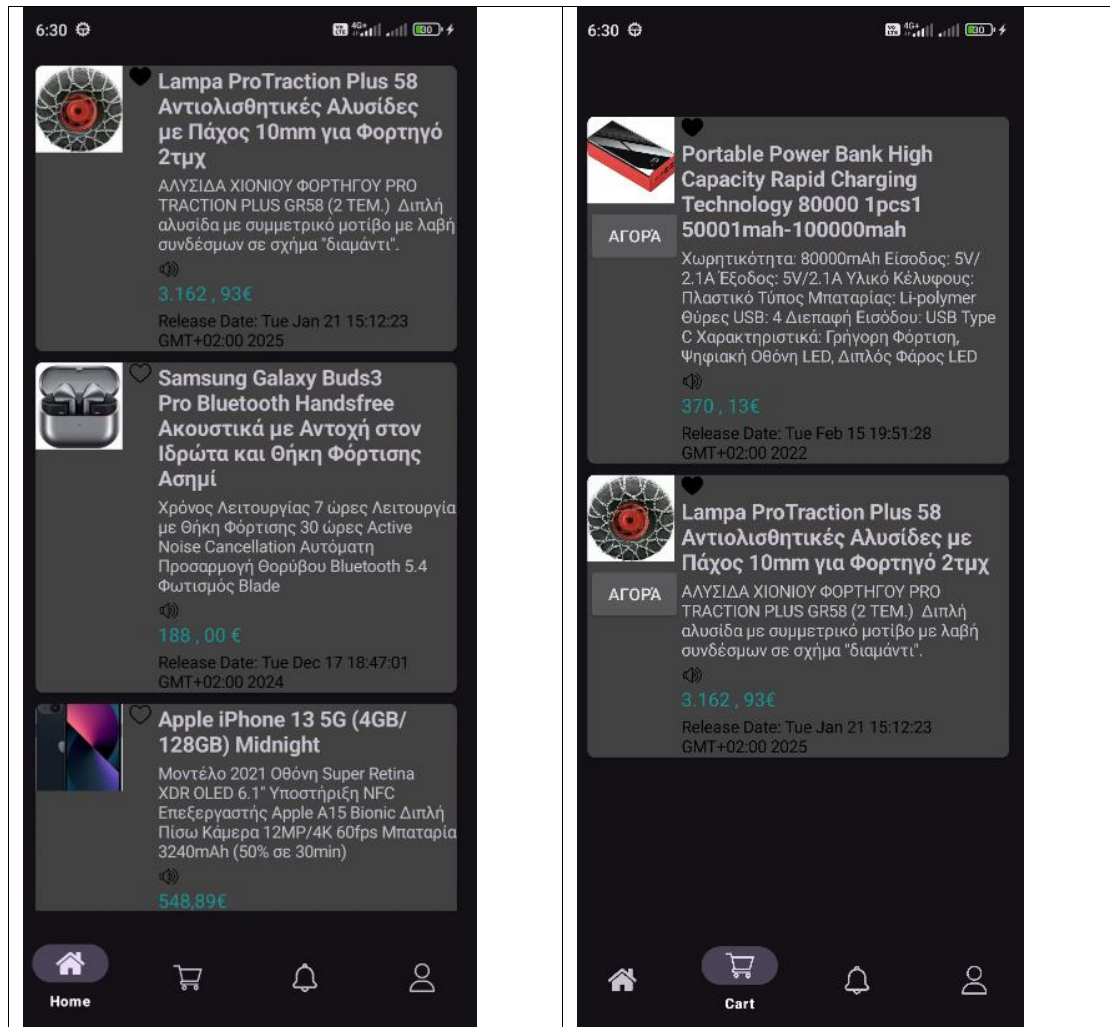


4.Λειτουργικότητα – Διεπαφή χρήστη

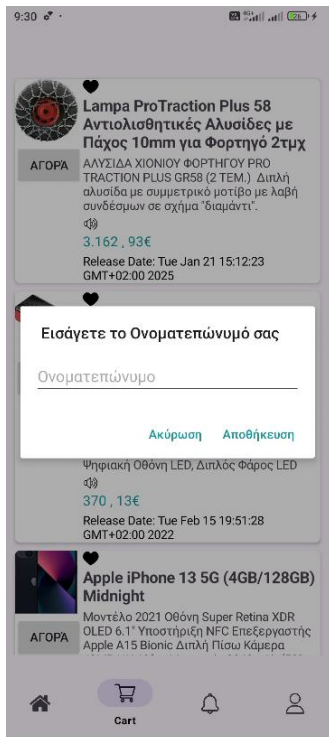
➤ Log in – Register :

The image displays two side-by-side mobile application screens. The left screen is titled 'Login' and features two input fields labeled 'Email' and 'Password'. Below these fields is a prominent blue button labeled 'Login'. At the bottom of the screen, there is a red link that reads 'New user? Register here'. The right screen is titled 'Register' and also features 'Email' and 'Password' input fields. Below these is a blue button labeled 'Register'. At the bottom, a red link reads 'Already a user? Login here'. Both screens have a status bar at the top showing the time as 13:19 and various system icons.

- Προβολή προϊόντων(αριστερά). Προβολή αγαπημένων προιόντων(δεξιά)



- **Αγορά Προϊόντος:**
Εάν ο χρήστης δεν έχει συμπληρώσει το Ονοματεπώνυμό του στο ProfileFragment(Settings) για να ανακτηθούν από τα SharedPreferences, τότε στην πρώτη παραγγελία, του ζητείται και βλέπουμε το παρακάτω.



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.setOnNavigationItemSelectedListener(item -> {  
    int itemId = item.getItemId();  
  
    // Εναλλαγή μεταξύ των fragment ανάλογα με το επιλεγμένο tab  
    if (itemId == R.id.nav_home) {  
        getSupportFragmentManager().beginTransaction()  
            .replace(R.id.container, new HomeFragment())  
            .commit();  
        return true;  
    } else if (itemId == R.id.nav_cart) {  
        getSupportFragmentManager().beginTransaction()  
            .replace(R.id.container, new CartFragment())  
            .commit();  
        return true;  
    } else if (itemId == R.id.nav_notifications) {  
        getSupportFragmentManager().beginTransaction()  
            .replace(R.id.container, new NotificationsFragment())  
            .commit();  
        return true;  
    } else if (itemId == R.id.nav_profile) {  
        getSupportFragmentManager().beginTransaction()  
            .replace(R.id.container, new ProfileFragment())  
            .commit();  
        return true;  
    } else {  
        return false;  
    }  
});
```

Αυτό το επιτυγχάνουμε χάρη στο αρχείο menu/**bottom_nav_menu.xml**, όπου και είναι τα 4 items Home, Cart, Notification & Profile

HomeFragment.java:

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

```
db.collection( collectionPath: "products").get().addOnCompleteListener(productTask -> {
    if (productTask.isSuccessful()) {
        for (QueryDocumentSnapshot document : productTask.getResult()) {
            String documentId = document.getId();
            String code = document.getString( field: "code");
            String description = document.getString( field: "description");
            String imageURL = document.getString( field: "imageURL");
            GeoPoint locationShop = document.getGeoPoint( field: "location");
            double latitude = 0;
            double longitude = 0;
            if (locationShop != null) {
                latitude = locationShop.getLatitude();
                longitude = locationShop.getLongitude();
            }
            String name = document.getString( field: "name");
            String price = document.getString( field: "price");
            Timestamp timestamp = document.getTimestamp( field: "release_date");
            String releaseDate = (timestamp != null) ? timestamp.toDate().toString() : "N/A";

            boolean isFavorite = favorites != null && favorites.contains(documentId);

            addDataToView(documentId, code, description, imageURL, latitude, longitude, name, price, releaseDate, isFavorite);
        }
    } else {
        Log.w(TAG, msg: "Error retrieving products", productTask.getException());
    }
});
```

Κάθε προϊόν που ανακτά η `loadAllDocuments()` την στέλνει στην `addDataToView` (String documentId, String code, String description, String imageURL, double latitude, double longitude, String name, String price, String releaseDate, boolean isFavorite) για να την εμφανίσει μέσα στο container(ScrollView) με την μορφή CardView.

Βιβλιοθήκη Picasso: Για ανάκτηση εικόνων μέσω url, πχ.

```
Picasso.get().Picasso
    .load(imageURL) RequestCreator
    .placeholder(R.drawable.placeholder)
    .error(R.drawable.errorimage)
    .into(imageView);
```

- Διαχείριση βάσης δεδομένων (εισαγωγή/ανάκτηση δεδομένων προϊόντων): Η εισαγωγή δεδομένων έγινε απευθείας από το firebase console. Η ανάκτηση δεδομένων έγινε από ένα αντικείμενο `Firestore db`.
- Για την τοποθεσία χρειαστηκε η βιβλιοθήκη `com.google.android.gms.location` που την συναντάμε στο Notification Fragment στην μεταβλητή **fusedLocationClient** :

```
fusedLocationClient = LocationServices.getFusedLocationProviderClient(requireContext());
```

- Χρήση API Text-to-Speech: Για καλύτερη οργάνωση δημιουργήσαμε το `MyTts.java` που περιλαμβάνει τις μεθόδους:
 - speakOrPause: Ξεκινά ή σταματά την ανάγνωση του κειμένου ανάλογα με το αν το TTS είναι ήδη ενεργό. Αν η υπηρεσία TTS δεν έχει αρχικοποιηθεί, την αρχικοποιεί και διαβάζει το κείμενο.
 - startSpeaking: Διαβάζει το κείμενο
 - shutdown: Τερματίζει το tts, απελευθερώνει τους πόρους, τερματίζει εάν υπάρχει κάποια ανάγνωση.

- Light/Dark Theme: Για αυτή την λειτουργία χρειάστηκαν τα αρχεία themes.xml & themes.xml(night)

```
private void loadThemePreference() {
    // Update RadioGroup based on saved theme
    if (isDarkThemeSelected) {
        rgColors.check(R.id.rb_dark);
    } else {
        rgColors.check(R.id.rb_light);
    }
}

1 usage
private void applyTheme() {
    // Apply selected theme
    if (isDarkThemeSelected) {
        AppCompatActivity.setDefaultNightMode(AppCompatActivity.MODE_NIGHT_YES);
    } else {
        AppCompatActivity.setDefaultNightMode(AppCompatActivity.MODE_NIGHT_NO);
    }
}
```

- **Permissions** που χρειάστηκαν(AndroidManifest.xml):

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.READ_MEDIA_IMAGES" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.POST_NOTIFICATIONS" />
```

6.Github Repository

Πατήστε [εδώ](#).