**ΑΤΟΜΙΚΗ ΕΡΓΑΣΙΑ**

**ΠΡΟΗΓΜΕΝΑ ΘΕΜΑΤΑ ΑΛΛΗΛΕΠΙΔΡΑΣΗΣ**



*Εικόνα 1: Logo της εφαρμογής*

XSystems

Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων, ΔΙ.ΠΑ.Ε.

**Πολυζωίδης Γιώργος**

AM: 134050

georgepol01@yahoo.com

**ΕΙΣΑΓΩΓΗ**

Το **Xsystems** είναι μια **Android** εφαρμογή διαχείρησης Βάσεων Δεδομένων για ένα **eshop** κατάστημα που εμπορεύεται και πουλάει hardware υλικό. Τα δεδομένα αποθηκεύονται σε δύο ξεχωριστές Βάσεις Δεδομένων, μια τοπική, κάνοντας χρήση του **Room API** και μια απομακρυσμένη κάνcντας χρήση του **Firebase Cloud Firestore**. Υπάρχει δυνατότητα προβολής, προσθήκης, αφαίρεσης και επεξεργασίας δεδομένων μέσω κατάλληλης διεπαφής χρήστη, η οποία αφορά τον υπάλληλο διαχείρησης των βάσεων (**DB Admin**).Για την δημιουργία της εφαργμογής χρησιμοποιήθηκε το Android Studio και ως προγραμματιστική γλώσσα η **Java**, παρακάτω γίνεται αναλυτική αναφορά για την υλοποίηση της.

**ΤΟΠΙΚΗ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ**

**ROOM API**

Η βάση αποτελείται από τους εξής 3 πίνακες **Product**, **Supplier**, **Supplies.**

**Πίνακες:**

1. **Product:** Αποθηκεύει όλα τα προιόντα που εμπορευέται το κατάστημα, με χαρακτηριστικά (id, name, stock, price) κωδικός, όνομα, απόθεμα και τιμή αντίστοιχα. Κύριο κλειδί του πίνακα είναι το id.
2. **Supplier:** Αποθηκεύει όλους τους προμηθευτές που προμηθεύουν το κατάστημα με χαρακτηριστικά (id, name, address, phone, email) κωδικός, όνομα, διεύθυνση, τηλέφωνο και email αντίστοιχα. Κύριο κλειδί του πίνακα είναι το id.
3. **Supplies:** Αποθηκεύει τις σχέσεις μεταξύ προιόντων και προμηθευτών με χαρακτηριστικά (id, idSupplier, idProduct) κωδικός προμήθειας , κωδικός προμηθευτή και κωδικός προιόντος αντίστοιχα. Κύριο κλειδί του πίνακα είναι ο συνδιασμός τους. Επίσης τα πεδία idSupplier και idSupplies είναι αναφορές σε ξένα κλειδία των αντίστοιχων πινάκων.

**ΑΠΟΜΑΚΡΥΣΜΕΝΗ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ**

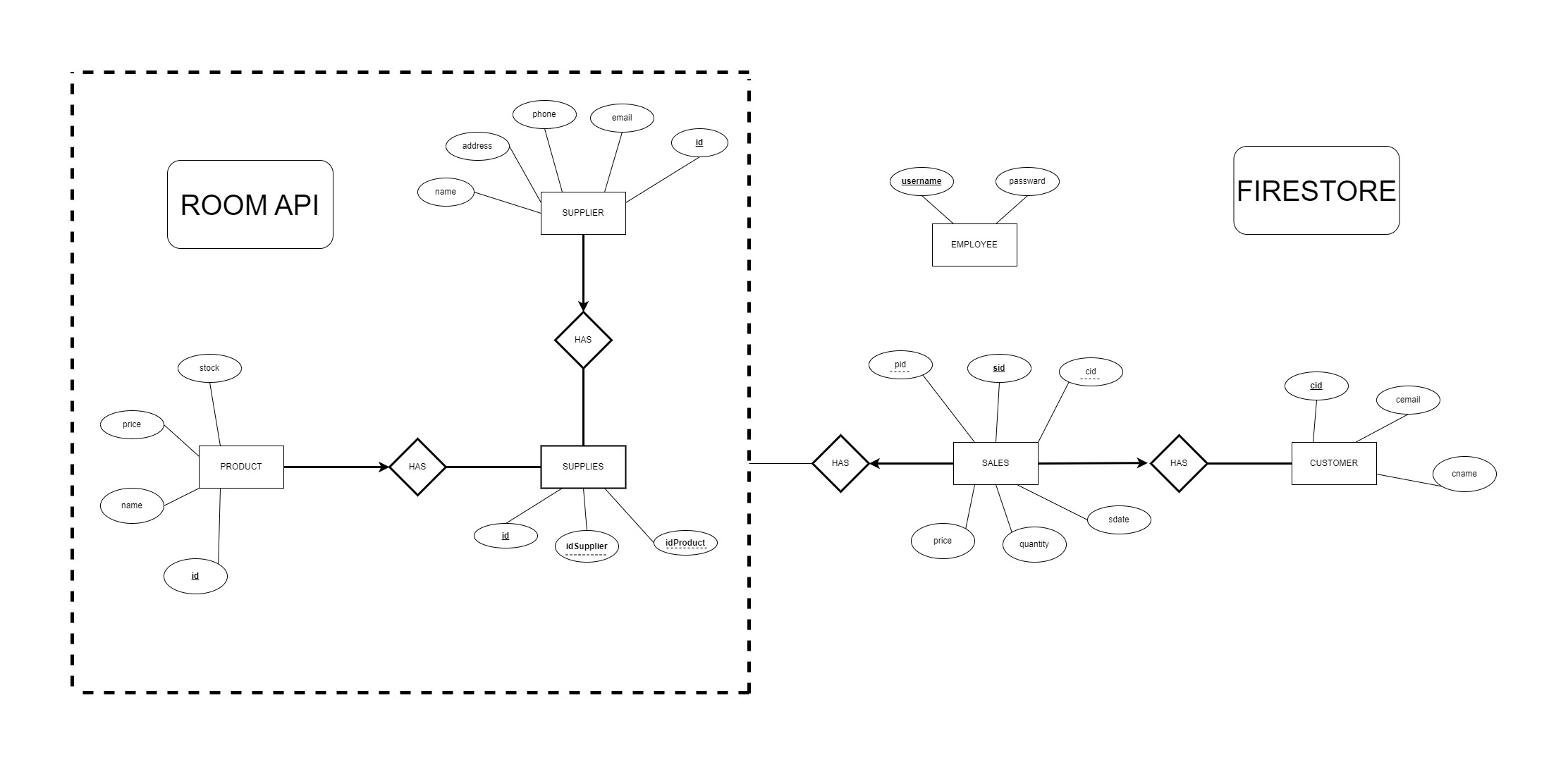
**FIREBASE CLOUD FIRESTORE**

Η βάση αποτελείται από τα εξής 3 Collections **Employee**, **Customer**, **Sales.** Επίσηςη βάση αυτή επειδή είναι NoSQl, έχει γίνει μια παραμοίωση-αντιστοιχηση σε σχεσιακή μορφή και ο ορισμός των κλειδιών είναι καθαρά για λόγους ευκολότερης κατανόησης.

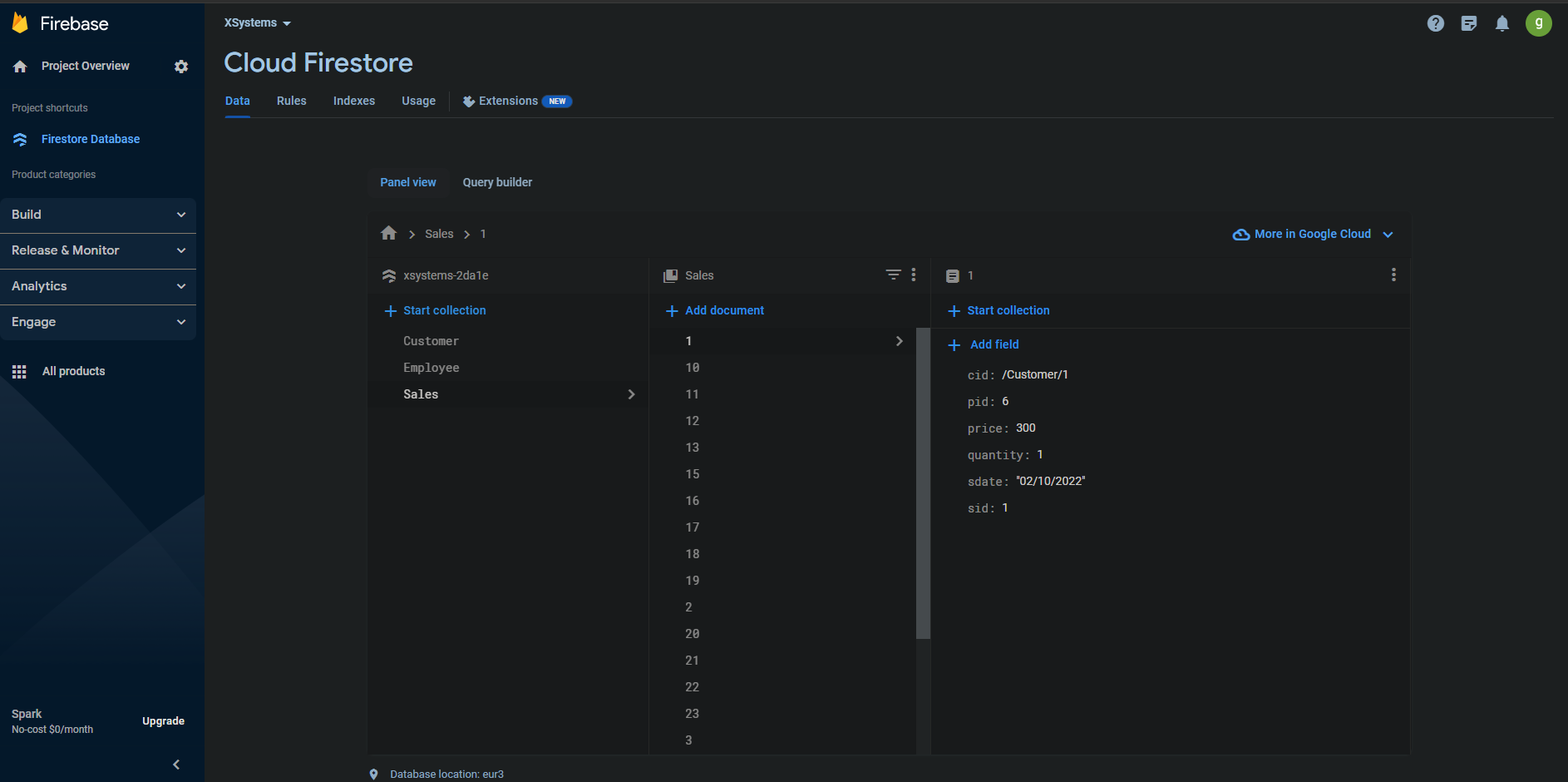
**Collections:**

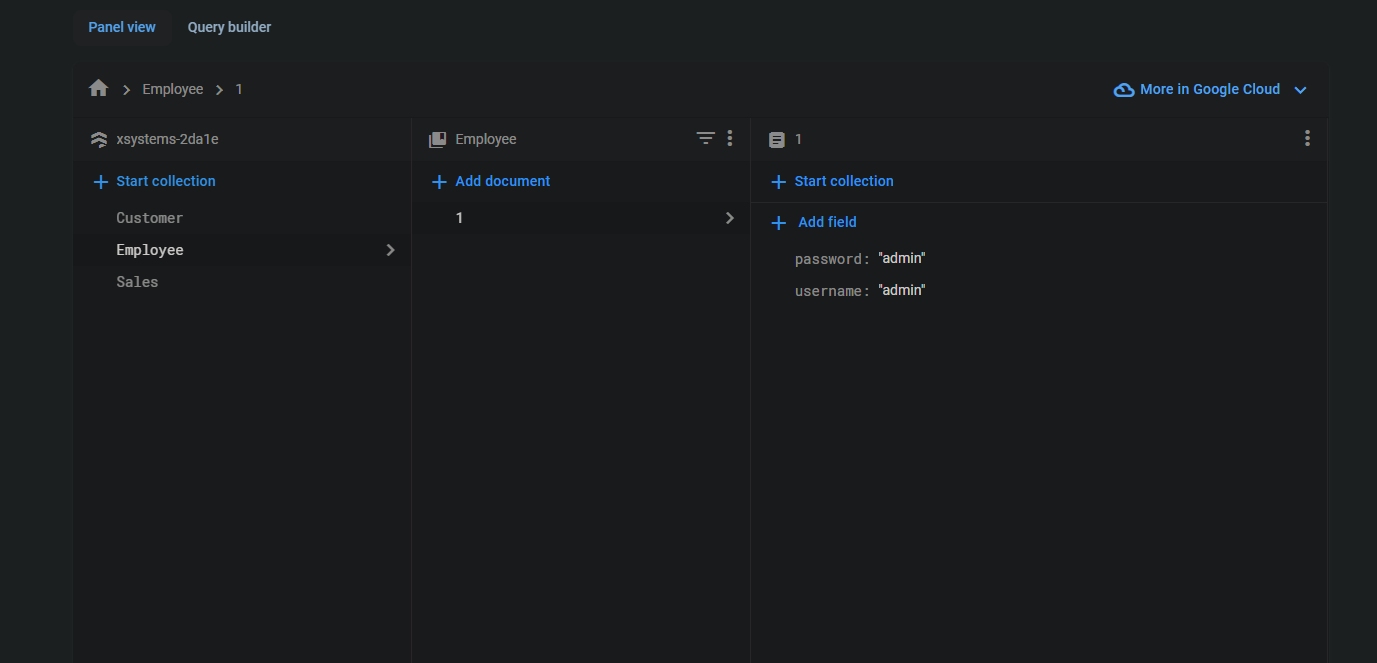
1. **Employee:** Αποθηκεύει τους λογαριασμούς αυτών που έχουν πρόσβαση στην εφαρμογή, με χαρακτηριστικά (username, password) όνομα και κωδικός αντίστοιχα. Κύριο κλειδί του πίνακα είναι το username.
2. **Customer:** Αποθηκεύει όλους τους πελάτες που έχουν κάνει αγορά από το κατάστημα με χαρακτηριστικά (cid, cname, cemail) κωδικός, όνομα και email αντίστοιχα. Κύριο κλειδί του πίνακα είναι το cid.
3. **Sales:** Αποθηκεύει όλες τις πωλήσεις που έχουν γίνει στο κατάστημα με χαρακτηριστικά (sid, cid, pid, price, quantity, sdate) κωδικός πώλησης , κωδικός πελάτη κωδικός προμηθευτή, τιμή πώλησης, ποσότητα προιόντος και ημερομηνία αντίστοιχα. Κύριο κλειδί του πίνακα είναι το sid. Επίσης τα πεδία cid και pid είναι αναφορές σε ξένα κλειδία των αντίστοιχων πινάκων.

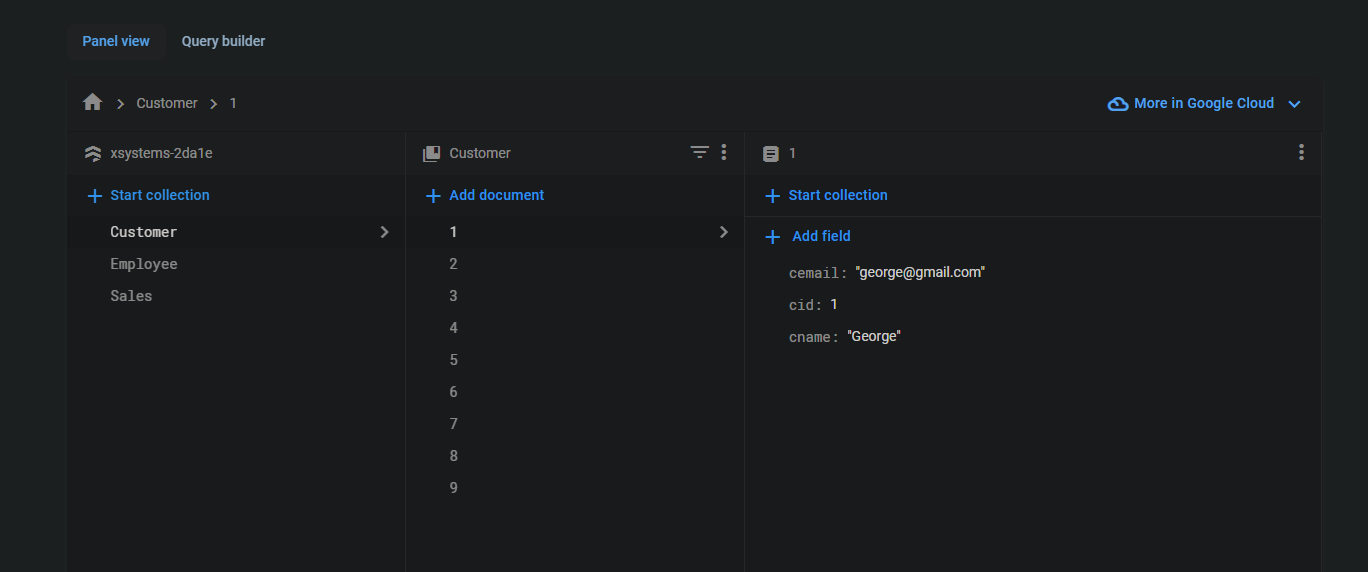
**ER Diagramm**:

 *Εικόνα 2.1: Αρχική σχεδίαση διαγράμματος ER.*

**Cloud FIrestore**:

 *Εικόνα 2.2: Firestore Sales Collection*

 *Εικόνα 2.3: Firestore Employee Collection*

 *Εικόνα 2.4: Firestore Customer Collection*

**Login Main Activity**

Η οθόνη Login είναι η πρώτη οθόνη που εμφανίζεται και χρησιμοποιείται για την σύνδεση του χρήστη με τον λογαριασμό του μέσω Firestore.

**ELEMENTS**

1) Constraint Layout: Το οποιό περιέχει τα παρακάτω.

2) TextView: Κείμενο “Log in”.

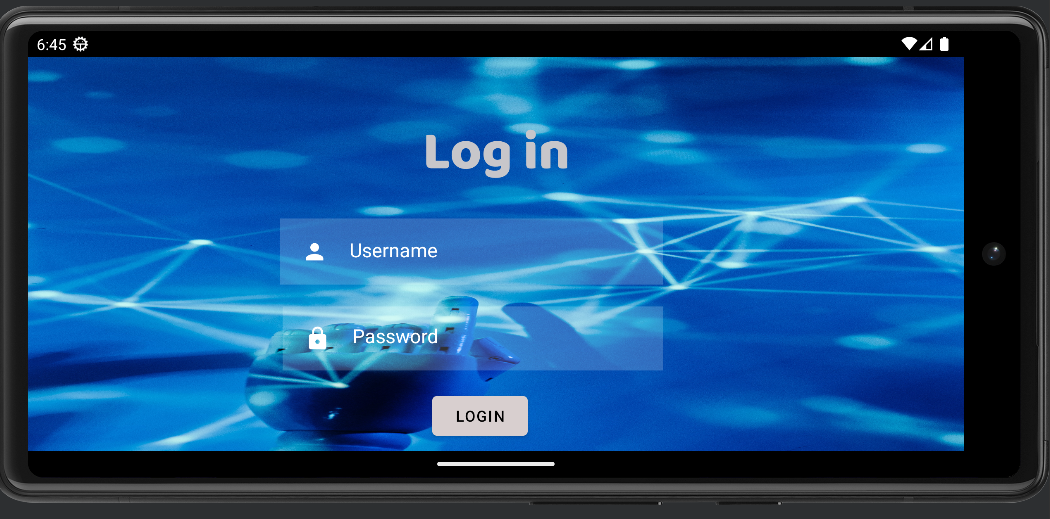
3) EditText: input με κείμενο “Username:” , όπου ο χρήστης γράφει το username του.

4) EditText: input με κείμενο “Password:” , όπου ο χρήστης γράφει το password του.

5) Button: Κουμπί, το οποίο ελέγχει αν τα στοιχεία ταιριάζουν σε κάποιον χρήστη. Αν ταιριάζουν ανοίγει το Menu Activity. Αν δεν ταιριάζουν τα στοιχεία εμφανίζει ένα toast message, με κείμενο “Invalid credentials”.



*Εικόνα 3.1: Login Frame (Portrait) της εφαρμογής.*

**

*Εικόνα 3.2: Login Frame (Landscape) της εφαρμογής.*

**Menu Activity**

Το Menu παράθυρο εμφανίζεται με την χρήση του login Button του Login activity. Αυτή η οθόνη χρησιμοποιείται για την μετακίνηση του χρήστη στις διάφορες λειτουργίες της εφαρμογής. Η μετακίνηση από οθόνη σε οθόνη γίνεται με την χρήση των activities και των fragments.

**ELEMENTS**

1) Drawer menu Layout: Το οποιό περιέχει τα παρακάτω.

2) Linear Layout: Το οποιό περιέχει τα παρακάτω.

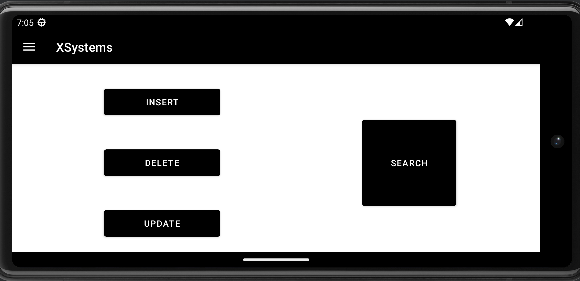
3) Custom Toolbar: Είναι η μπάρα εργαλείων στο πανώ μέρος της οθόνης η οποία περιέχει ένα burger button icon που συνδεέται με το nav view για να ανοίγει το Drawer menu.

4) Fragment container: Είναι ένα constraint layout στο οποίο γίνεται η εμφάνιση των fragment.

5) Navigation view: Για την περιήγηση στην εφάρμογη από το Drawer menu.



*Εικόνα 4.1: Menu Frame (Portrait) της εφαρμογής.*



*Εικόνα 4.2: Menu Frame (Landscape) της εφαρμογής.*

**Drawer Menu**

Το Drawer menu παράθυρο εμφανίζεται με την χρήση του burger button icon του toolbar που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την περιήγηση του χρήστη στις διαθέσιμες λειτουργίες της εφαρμογής και είναι προσβάσιμη από κάθε οθόνη όπως και το toolbar.

**ELEMENTS**

1) Menu: Το οποιό περιέχει τα παρακάτω.

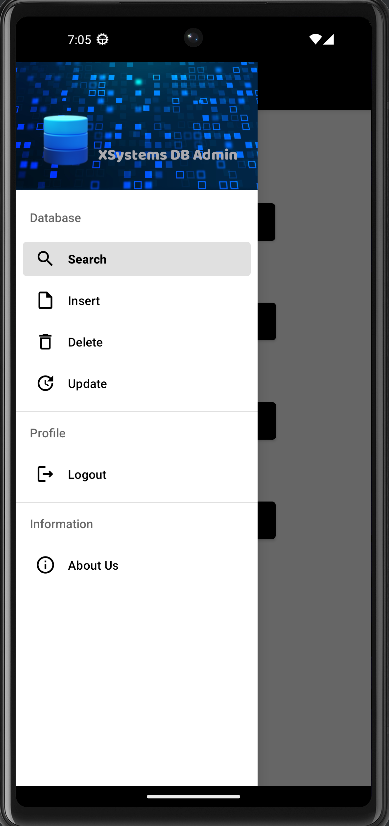
2) Item: Τίτλος item group “Database”.

3) Menu: Το οποιό περιέχει τα παρακάτω.

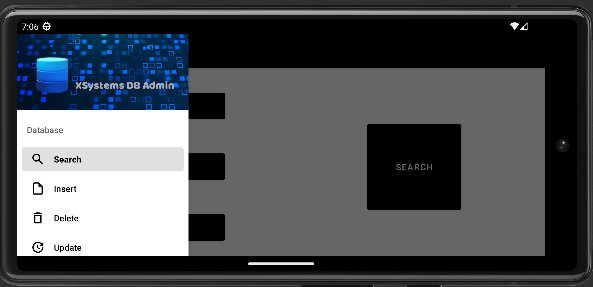
4) Group: To οποίο περιέχει τα items search, insert, delete, update με τα αντίστοιχα icons.

Με την ιδία λογική είναι και το Profile που περιέχει το logout το οποίο όταν πατηθεί μας γυρνάει στο login screen, παρομοίως και το Information που περιέχει το about us με το πατήμα του φορτώνει το αντίστοιχο fragment.

Επίσης, το Drawer menu περιέχει ξεχωριστό xml για το header του, το οποίο περιέχει το background image μαζι με το logo και ενα textview.



*Εικόνα 5.1: Drawer Menu (Portrait) της εφαρμογής.*



*Εικόνα 5.2: Drawer Menu (Landscape) της εφαρμογής.*

**Search Fragment**

Το Search παράθυρο εμφανίζεται με την χρήση του search Button του main fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την αναζήτηση του χρήστη στις εγγραφές που περιέχονται στις βάσεις της εφαρμογής, αλλά και χρήση ερωτημάτων.

**ELEMENTS**

1) Linear Layout: Το οποιό περιέχει τα παρακάτω.

2) TextView: Κείμενο “Query Result”.

3) Spinner: Το οποιό περιέχει 11 επιλογές από queries προς τις βάσεις.

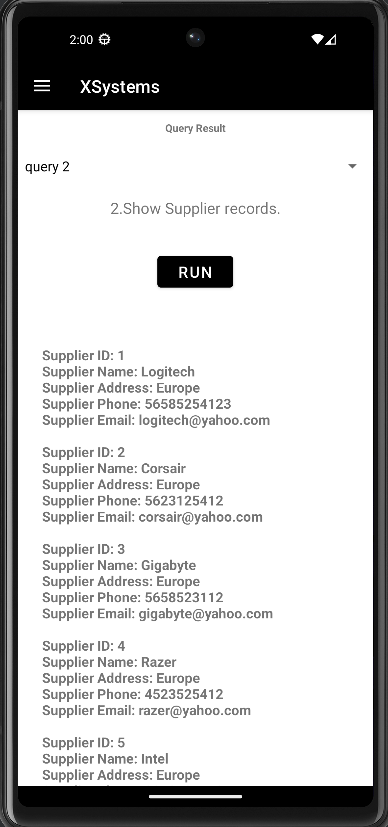
4) TextView: Κείμενο “Query Description”, στο οποίο εμφανίζεται η επιλογή που έχει γίνει από το spinner.

5) Button: Κουμπί, το οποίο εμφανίζει τα αποτελέσματα της επιλογής στο textview.

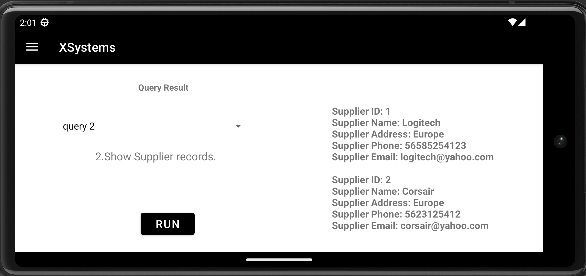
6) Linear Layout: Το οποιό περιέχει τα παρακάτω.

7) ScrollView: Για να μπορεί να γίνει scroll στα αποτελέσματα που βριίσκονται μέσα στο textview.

8) TextView: Το οποιό περιέχει τα αποτελέσματα από την επιλογή που έχει γίνει από το spinner.



*Εικόνα 6.1: Search Frame (Portrait) της εφαρμογής.*



*Εικόνα 6.2: Search Frame (Landscape) της εφαρμογής.*

**Insert-Delete-Update Fragments**

Τα Insert-Delete-Update παράθυρα εμφανίζονται με την χρήση του αντίστοιχου Button του main fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την πλοήγηση του χρήστη στις λειτουργίες insert-delete-update της εφαρμογής.

**ELEMENTS**

1) Constraint Layout: Το οποιό περιέχει τα παρακάτω.

2) Button: Κουμπί, το οποίο εμφανίζει την οθόνη για το insert του αντίστοιχου πίνακα (Supplies, Supplier, Product, Sales, Customer) .

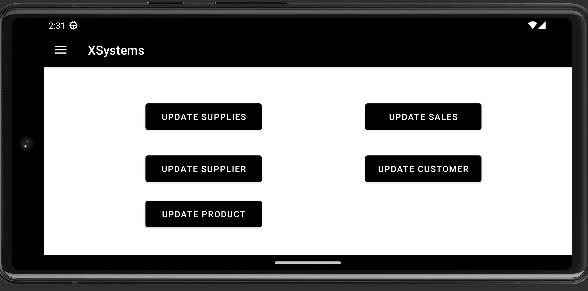
3) Button: Κουμπί, το οποίο εμφανίζει την οθόνη για το delete του αντίστοιχου πίνακα (Supplies, Supplier, Product, Sales, Customer) .

4) Button: Κουμπί, το οποίο εμφανίζει την οθόνη για το update του αντίστοιχου πίνακα (Supplies, Supplier, Product, Sales, Customer) .

Όλες οι οθόνες έχουν ακριβώς το ίδιο layout και λειτουργίες, (**είκονα 7.1 και 7.2**).



*Εικόνα 7.1: Insert Frame (Portrait) της εφαρμογής.*



*Εικόνα 7.2: Update Frame (Landscape) της εφαρμογής.*

**Insert Supplies Fragment**

Το Insert Supplies παράθυρο εμφανίζεται με την χρήση του αντίστοιχου Button του insert fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την εισαγωγή των εγγραφών που έχουν σχέση με τις προμήθειες και θα αποθηκευτούν στην τοπική βάση της εφαρμογής.

**ELEMENTS**

1) Linear Layout: Το οποιό περιέχει τα παρακάτω.

2) TextView: Τίτλος “INSERT SUPPLIES”.

3) EditText: input με κείμενο “insert supplies id”.

4) EditText: input με κείμενο “insert supplier id”.

5) EditText: input με κείμενο “insert product id”.

6) Button: Κουμπί με κείμενο “INSERT”, το οποίο πατώντας το γίνεται η καταχώρηση της εγγραφής στη βάση μετά από κατάλληλους ελέγχους, όπου ανάλογα με το αποτέλεσμα θα εμφανιστεί κατάλληλο μήνυμα στο κάτω μέρος της οθόνης.



*Εικόνα 8.1: Insert Supplies Frame (Portrait) της εφαρμογής.*



*Εικόνα 8.2: Insert Supplies (Landscape) της εφαρμογής.*

**Insert Supplier Fragment**

Το Insert Supplier παράθυρο εμφανίζεται με την χρήση του αντίστοιχου Button του insert fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την εισαγωγή των εγγραφών που έχουν σχέση με τους προμηθευτές και θα αποθηκευτούν στην τοπική βάση της εφαρμογής.

**ELEMENTS**

1) Linear Layout: Το οποιό περιέχει τα παρακάτω.

2) TextView: Τίτλος “INSERT SUPPLIER”.

3) EditText: input με κείμενο “insert id”.

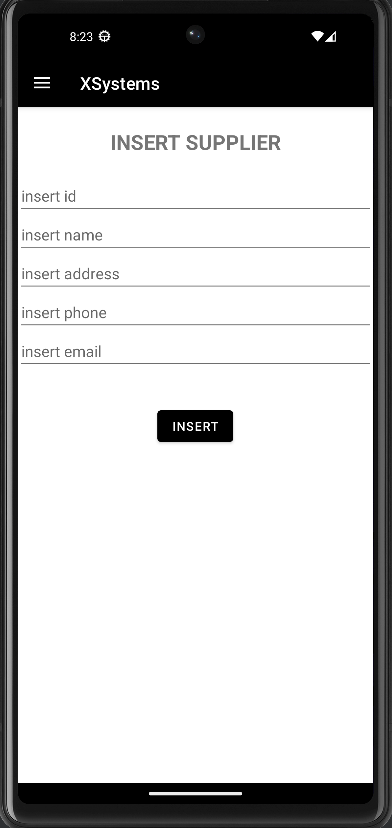
4) EditText: input με κείμενο “insert name”.

5) EditText: input με κείμενο “insert address”.

5) EditText: input με κείμενο “insert phone”.

6) EditText: input με κείμενο “insert email”.

7) Button: Κουμπί με κείμενο “INSERT”, το οποίο πατώντας το γίνεται η καταχώρηση της εγγραφής στη βάση μετά από κατάλληλους ελέγχους, όπου ανάλογα με το αποτέλεσμα θα εμφανιστεί κατάλληλο μήνυμα στο κάτω μέρος της οθόνης.



*Εικόνα 9.1: Insert Supplier Frame (Portrait) της εφαρμογής.*



*Εικόνα 9.2: Insert Supplier (Landscape) της εφαρμογής.*

**Insert Product Fragment**

Το Insert Product παράθυρο εμφανίζεται με την χρήση του αντίστοιχου Button του insert fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την εισαγωγή των εγγραφών που έχουν σχέση με τα προιόντα και θα αποθηκευτούν στην τοπική βάση της εφαρμογής.

**ELEMENTS**

1) Linear Layout: Το οποιό περιέχει τα παρακάτω.

2) TextView: Τίτλος “INSERT PRODUCT”.

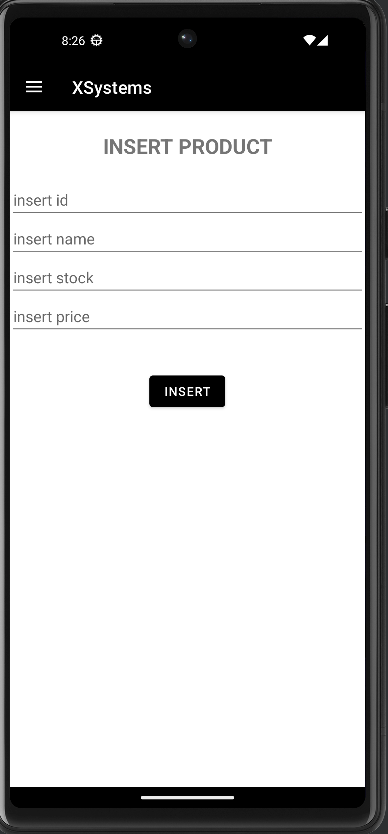
3) EditText: input με κείμενο “insert id”.

4) EditText: input με κείμενο “insert name”.

5) EditText: input με κείμενο “insert stock”.

6) EditText: input με κείμενο “insert price”.

7) Button: Κουμπί με κείμενο “INSERT”, το οποίο πατώντας το γίνεται η καταχώρηση της εγγραφής στη βάση μετά από κατάλληλους ελέγχους, όπου ανάλογα με το αποτέλεσμα θα εμφανιστεί κατάλληλο μήνυμα στο κάτω μέρος της οθόνης.



*Εικόνα 10.1: Insert Product Frame (Portrait) της εφαρμογής.*



*Εικόνα 10.2: Insert Product (Landscape) της εφαρμογής.*

**Insert Sales Fragment**

Το Insert Sales παράθυρο εμφανίζεται με την χρήση του αντίστοιχου Button του insert fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την εισαγωγή των εγγραφών που έχουν σχέση με τις πωλήσεις και θα αποθηκευτούν στην απομακρυσμένη βάση της εφαρμογής.

**ELEMENTS**

1) Linear Layout: Το οποιό περιέχει τα παρακάτω.

2) TextView: Τίτλος “INSERT SALES”.

3) EditText: input με κείμενο “insert id”.

4) EditText: input με κείμενο “insert cid”.

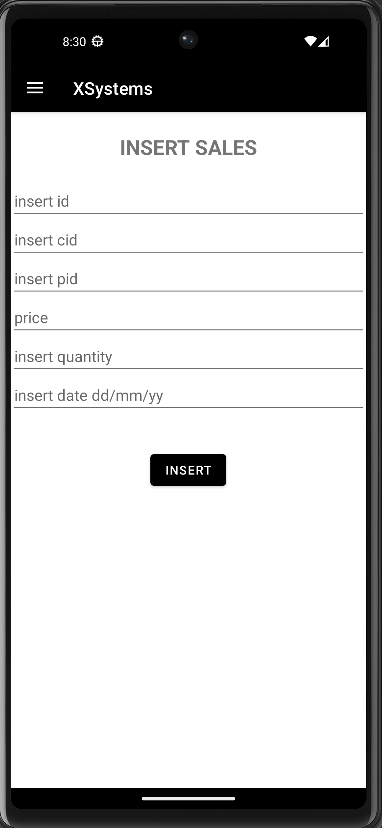
5) EditText: non editable με κείμενο “price”.

6) EditText: input με κείμενο “insert quantity”.

7) EditText: input με κείμενο “insert date dd/mm/yy”.

8) Button: Κουμπί με κείμενο “INSERT”, το οποίο πατώντας το γίνεται η καταχώρηση της εγγραφής στη βάση μετά από κατάλληλους ελέγχους, όπου ανάλογα με το αποτέλεσμα θα εμφανιστεί κατάλληλο μήνυμα στο κάτω μέρος της οθόνης.

Επιπλέον όταν συμπληρωθούν κατάλληλα τα πεδία pid και quantity γίνεται αυτόματος υπολογισμός της τιμής στο πεδίο price και ενημέρωση του Product με το καινούργιο quantity.



*Εικόνα 11.1: Insert Sales Frame (Portrait) της εφαρμογής.*



*Εικόνα 11.2: Insert Sales (Landscape) της εφαρμογής*

**Insert Customer Fragment**

Το Insert Customer παράθυρο εμφανίζεται με την χρήση του αντίστοιχου Button του insert fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την εισαγωγή των εγγραφών που έχουν σχέση με τους πελάτες και θα αποθηκευτούν στην απομακρυσμένη βάση της εφαρμογής.

**ELEMENTS**

1) Linear Layout: Το οποιό περιέχει τα παρακάτω.

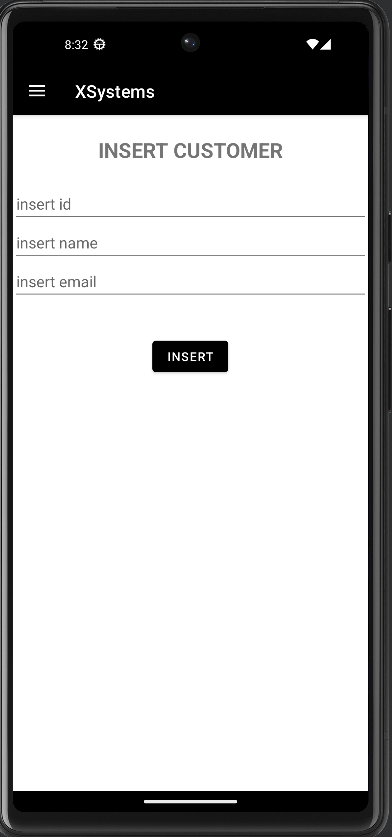
2) TextView: Τίτλος “INSERT CUSTOMER”.

3) EditText: input με κείμενο “insert id”.

4) EditText: input με κείμενο “insert name”.

5) EditText: input με κείμενο “insert email”.

6) Button: Κουμπί με κείμενο “INSERT”, το οποίο πατώντας το γίνεται η καταχώρηση της εγγραφής στη βάση μετά από κατάλληλους ελέγχους, όπου ανάλογα με το αποτέλεσμα θα εμφανιστεί κατάλληλο μήνυμα στο κάτω μέρος της οθόνης.



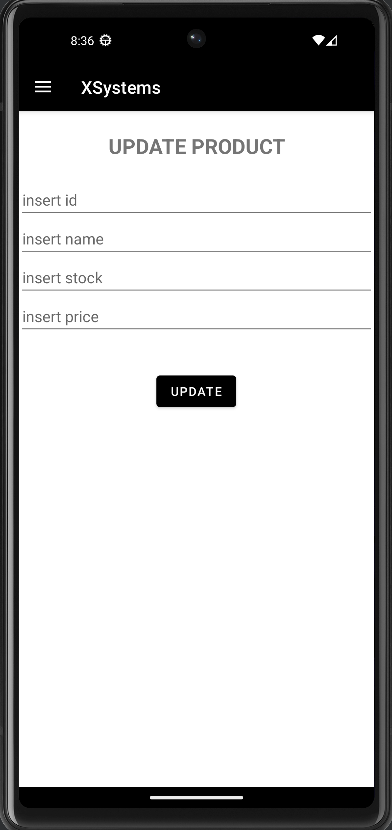
*Εικόνα 11.1: Insert Customer Frame (Portrait) της εφαρμογής.*



*Εικόνα 11.2: Insert Customer (Landscape) της εφαρμογής*

**Update Fragments**

Τα Update παράθυρα εμφανίζονται με την χρήση του update Button του main fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την επεξεργασία των αντίστοιχων των εγγραφών που περιέχονται στις βάσεις της εφαρμογής και έχουν ακριβώς ίδια δομή και λειτουργία με τα Insert fragments, με μόνη διαφόρα οτι οι έλεχγοι είναι πιο σύνθετοι. Επίσης μόλις συμπληρωθεί κάποιο υπάρχων id και χαθεί το focus από το edittext, αυτόματα γέμιζουν τα υπόλοιπα πεδία με τα κατάλληλα στοιχεία που αντιστοιχούν στο id.Παράδειγμα (**εικόνες 12.1 και 12.2**).



*Εικόνα 12.1: Update Product Frame (Portrait) της εφαρμογής.*



*Εικόνα 12.2: Update Product (Landscape) της εφαρμογής*

**Delete Fragments**

Τα Delete παράθυρα εμφανίζονται με την χρήση του delete Button του main fragment που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την διαγραφή των αντίστοιων εγγραφών που περιέχονται στις βάσεις της εφαρμογής και έχουν όλες την ίδια δομή και λειτουργία. Παρακάτω φαίνεται το παράδειγμα του delete supplies (**εικόνες 13.1 και 13.2**).

**ELEMENTS**

1) Linear Layout: Το οποιό περιέχει τα παρακάτω.

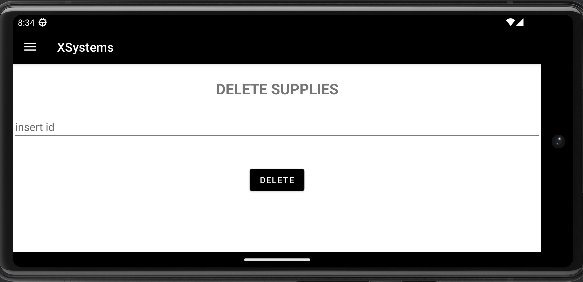
2) TextView: Κείμενο “DELETE SUPPLIES”.

3) EditText: input με κείμενο “insert id”.

4) Button: Κουμπί με κείμενο “DELETE”, το οποίο πατώντας το γίνεται η διαγραφή της εγγραφής από τη βάση μετά από κατάλληλους ελέγχους, όπου ανάλογα με το αποτέλεσμα θα εμφανιστεί κατάλληλο μήνυμα στο κάτω μέρος της οθόνης.



*Εικόνα 13.1: Delete Supplies Frame (Portrait) της εφαρμογής.*



*Εικόνα 13.2: Delete Supplies (Landscape) της εφαρμογής*

**About us Fragment**

Το About us παράθυρο εμφανίζεται με την χρήση του about us option του Drawer menu που βρίσκεται μέσα στο menu activity. Αυτή η οθόνη χρησιμοποιείται για την εμφάνιση πληροφοριών σχετικά με την εφαρμογή.

**ELEMENTS**

1) Constraint Layout: Το οποιό περιέχει τα παρακάτω.

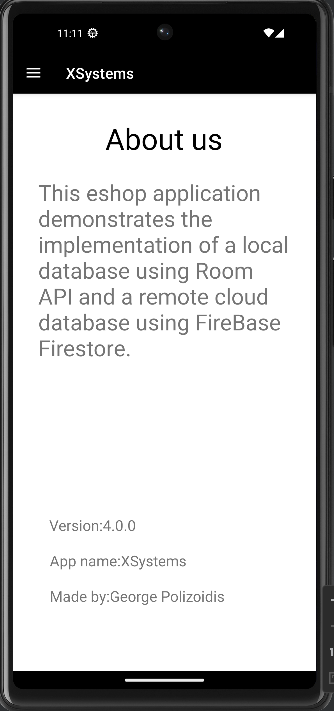
2) TextView: Τίτλος “About us”.

3) TextView: με το κεντρικό κείμενο.

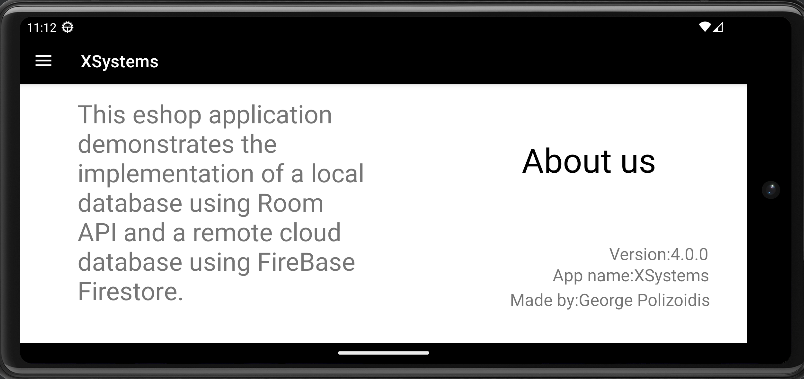
4) TextView: με κείμενο “Version 4.0.0”.

5) TextView: με κείμενο “App name: XSystems”.

6) TextView: με κείμενο “Made by: George Polizoidis”.



*Εικόνα 14.1: About Us Frame (Portrait) της εφαρμογής.*



*Εικόνα 14.2: About Us (Landscape) της εφαρμογής*

**ΕΡΓΑΛΕΙΑ**

Παρακάτω εμφανίζονται όλα τα εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση της εργασίας.

Στα αρχικά στάδια χρησιμοποιήθηκε το **Diagram Editor** για την σχεδίαση του **ER**. <https://app.diagrams.net>

Στο περιβάλλον του **Android Studio**, με την χρήση **Java** δημιουργήθηκε η εφαρμογή. <https://developer.android.com/studio>

Η διαχείρηση για την τοπική βάση δεδομένων έγινε με χρήση του **Room API** μέσω του **Android Studio** και για την απομακρυσμένη, χρήση του **Firebase Cloud Firestore**. <https://firebase.google.com/docs/firestore>

Η συγγραφή της έκθεσης έγινε με το **Microsoft Office Word**.

Τα icons που χρησιμοποιήθηκαν προήλθαν από τα παρακάτω site, <https://icons8.com/icons/set/android> και οι εικόνες <https://www.pexels.com>