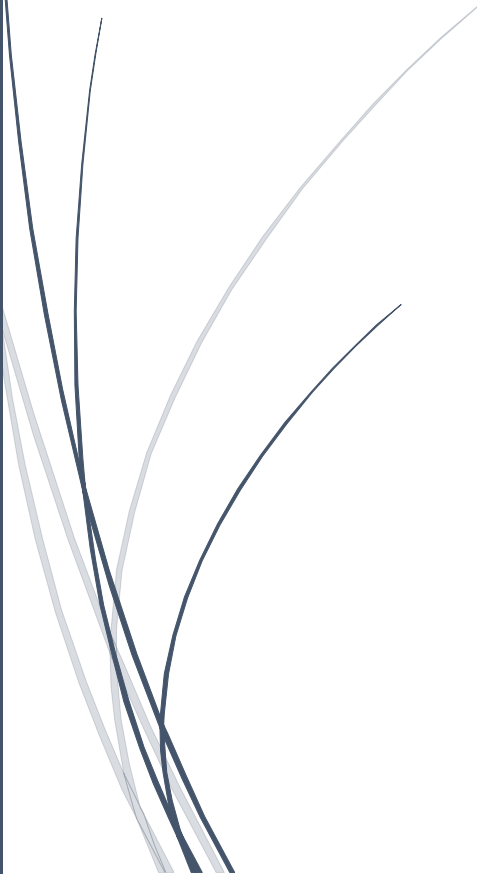




Documentation

Πτυχιακή Εργασία

Πέτρος-Ευστάθιος Φατούρος Π18164



Περιεχόμενα

Περίληψη.....	1
Εισαγωγή.....	2
Ανασκόπηση πεδίου.....	2
Παρουσίαση και χρήση εφαρμογής.....	2
Web εφαρμογή δημιουργίας μενού.....	2
Back-end API.....	8
Android εφαρμογή.....	9
Admin.....	12
Manager.....	19
Employee.....	30
Αρχιτεκτονική συστήματος.....	43
Web εφαρμογή δημιουργίας μενού.....	43
Back-end API.....	43
Android εφαρμογή.....	44
Συμπεράσματα και μελλοντικές επεκτάσεις.....	45
Βιβλιογραφία.....	45

Περίληψη

Η παρών πτυχιακή πραγματεύεται την υλοποίηση ενός personal digital assistant (PDA). Η γλώσσα υλοποίησης είναι εξ' ολοκλήρου java. Χρησιμοποιήθηκαν τα εξής IDE: IntelliJ IDEA και Android Studio. Το back-end είναι ένα API υλοποιημένο σε java spring boot, ενώ το front-end είναι μία android εφαρμογή που έχει υλοποιηθεί σε java. Ιδιαίτερη σημασία δίδεται στο security του back-end το οποίο χρησιμοποιεί json web tokens (JWTs) για το authentication και το authorization.

Εισαγωγή

Το θέμα της συγκεκριμένης πτυχιακής εργασίας είναι η υλοποίηση ενός personal digital assistant (PDA) για καφετέριες, μπαρ, εστιατόρια και γενικότερα για οποιαδήποτε επιχείρηση που λειτουργεί με παρόμοιο τρόπο. Συνολικά εμπεριέχει μία web εφαρμογή για τη δημιουργία προσαρμοσμένων μενού, μία εφαρμογή Android που αποτελεί το user interface (UI) και ένα back-end API που διαχειρίζεται HTTP αιτήματα από την κύρια εφαρμογή.

Η πτυχιακή εξυπηρετεί, σε βασικό επίπεδο, τις ίδιες ανάγκες που εξυπηρετεί ένα PDA. Ουσιαστικά αποτελεί μία μινιμαλιστική προσέγγιση των σύγχρονων PDA.

Ανασκόπηση πεδίου

Μερικές αντίστοιχες επαγγελματικές εφαρμογές (λογισμικά παραγγελιοληψίας) που εξυπηρετούν παρόμοιους σκοπούς με το παρών project είναι οι εξής:

- **Xenios** από την εταιρία **Portable Solutions**:
<https://www.portable.gr/efarmoges/efarmogi-paraggeliolipsias-xenios.html>
- **Rodon** από την εταιρία **eLINK**:
<https://www.elink.com.gr/product/rodon-%ce%bb%ce%bf%ce%b3%ce%b9%cf%83%ce%bc%ce%b9%ce%ba%cf%8c-%cf%80%ce%b1%cf%81%ce%b1%ce%b3%ce%b3%ce%b5%ce%bb%ce%b9%ce%bf%ce%bb%ce%b7%cf%88%ce%af%ce%b1%cf%82>
- **Mobile Anima.NET** από την εταιρία **Poseidon Software**:
<http://www.poseidon.gr/mobile.htm>

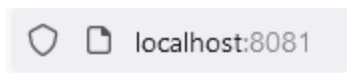
Παρουσίαση και χρήση εφαρμογής

Ακολουθεί το user's manual της πτυχιακής. Για ευκολία ανάγνωσης, η συγκεκριμένη ενότητα χωρίζεται σε υποενότητες που αντιστοιχούν στις 3 εφαρμογές από τις οποίες απαρτίζεται η πτυχιακή.

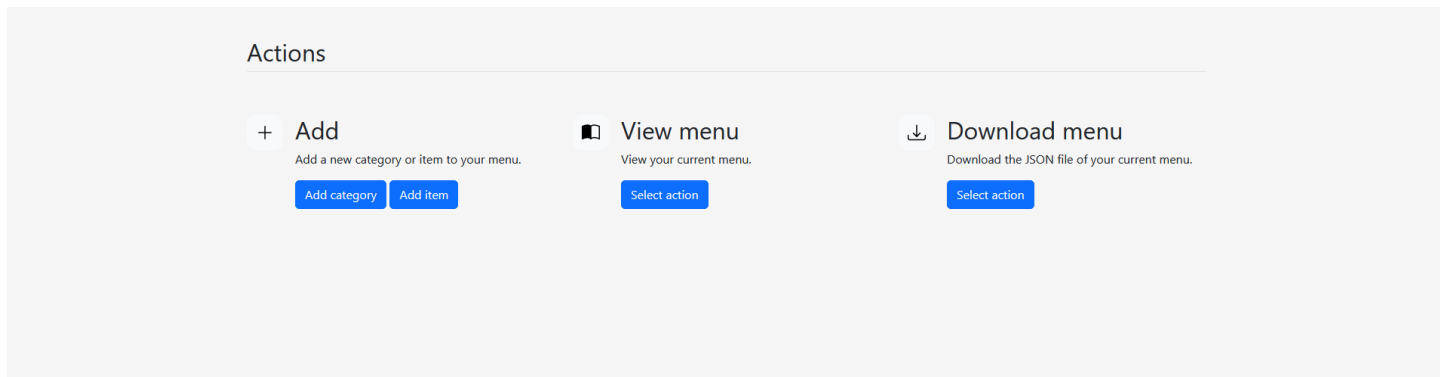
Web εφαρμογή δημιουργίας μενού

Για τη σωστή λειτουργία του συστήματος παραγγελιών είναι απαραίτητο ένα μενού. Για το σκοπό αυτό υλοποιήθηκε η αντίστοιχη web εφαρμογή. Για την εκτέλεση της, αρκεί να ανοίξετε το project στο φάκελο **Spring_Boot_Menu_Creator** στο IntelliJ IDEA και να τρέξετε την εφαρμογή.


Πλοηγηθείτε στη διεύθυνση IP του localhost στην πόρτα 8081 (θα πρέπει να μην χρησιμοποιείται ήδη από άλλο πρόγραμμα ή εφαρμογή).



Θα εμφανιστεί η εξής οθόνη:



Με τη δράση **Add** μπορείτε να προσθέσετε κατηγορίες και αντικείμενα στο τρέχων μενού σας.
Για την προσθήκη κατηγορίας αρκεί να δώσετε το όνομα αυτής.



eAssist


Add category

Category name

Add category

[Return to homepage](#)

Για την προσθήκη αντικειμένου, εκτός του ονόματός του, πρέπει να επιλέξετε μία ήδη υπάρχων κατηγορία, την τιμή του και την αρχική ποσότητα στην οποία θα είναι διαθέσιμη.


eAssist

Add item

Item name

Martini

Category

Cocktails

Price

10

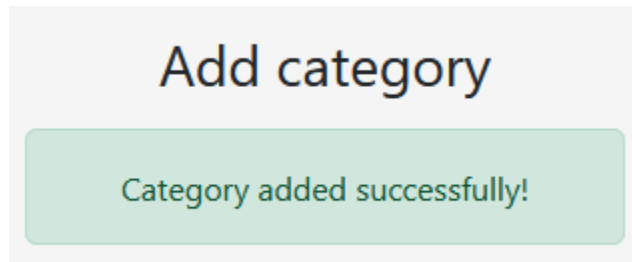
Quantity

10

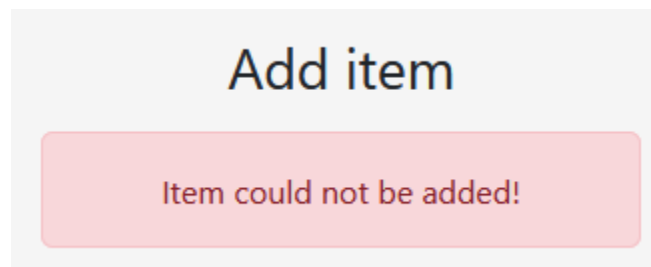
Add item

[Return to homepage](#)

Όταν η προσθήκη μίας κατηγορίας ή ενός αντικειμένου είναι επιτυχής, τότε εμφανίζεται σχετική ειδοποίηση.



Αντίστοιχα, σε περίπτωση διπλότυπων ονομάτων, η προσθήκη είναι ανεπιτυχής και εμφανίζεται σχετικό μήνυμα λάθους.

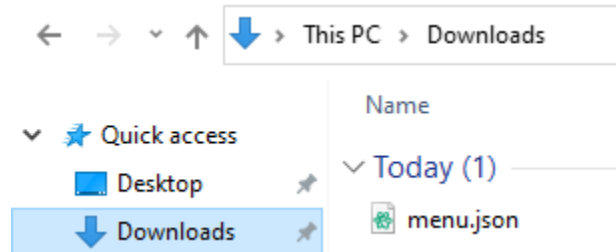


Χάρη παραδείγματος εκτέλεσης, δημιουργούμε 2 κατηγορίες **Cocktails** και **Snacks** και προσθέτουμε 1 αντικείμενο στην κάθε μία. Επιλέγοντας τη δράση **View menu** εμφανίζεται το τρέχων μενού το οποίο μπορούμε επίσης να τροποποιήσουμε.

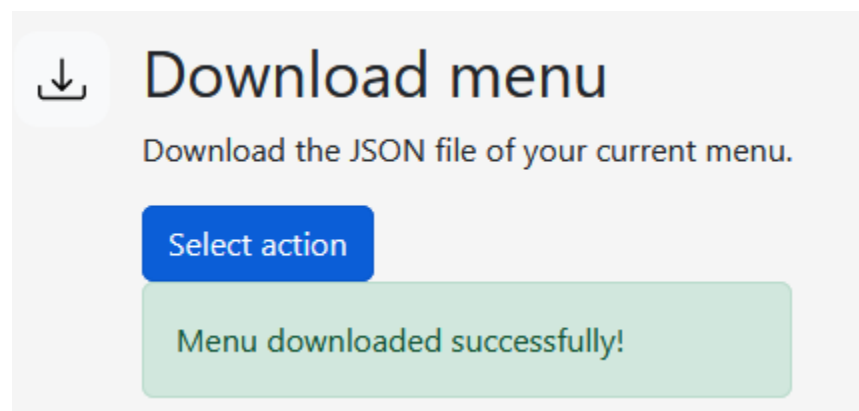
Current menu				
The lists below contain all the current categories and items of your menu				
Back to homepage				
Category name			Action	
Cocktails			Delete	
Snacks			Delete	
Item name	Category	Price	Quantity	Action
Martini	Cocktails	10.0	10	Delete
Vegan Pancakes	Snacks	13.0	10	Delete

Με την δράση **Delete** δίδεται η δυνατότητα να αφαιρέσουμε κατηγορίες ή/και αντικείμενα από το μενού.

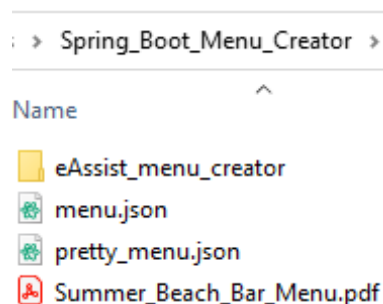
Αφού ολοκληρωθεί η δημιουργία του μενού, επιλέγουμε τη δράση **Download menu** η οποία αποθηκεύει το τρέχων μενού στις λήψεις σε μορφή αρχείου json.



Όταν η αποθήκευση του μενού ολοκληρώνεται, εμφανίζεται σχετικό μήνυμα.



Σημείωση: το ολοκληρωμένο μενού που χρησιμοποιείται από την android εφαρμογή βρίσκεται σε μορφή αρχείου JSON στον ίδιο φάκελο που βρίσκεται και το συγκεκριμένο project. Για λόγους ευκολότερης ανάγνωσης και παρουσίασης το ίδιο μενού υπάρχει επίσης σε ξεχωριστό αρχείο JSON σε prettified μορφή καθώς και σε μορφή PDF (<https://www.canva.com/>).

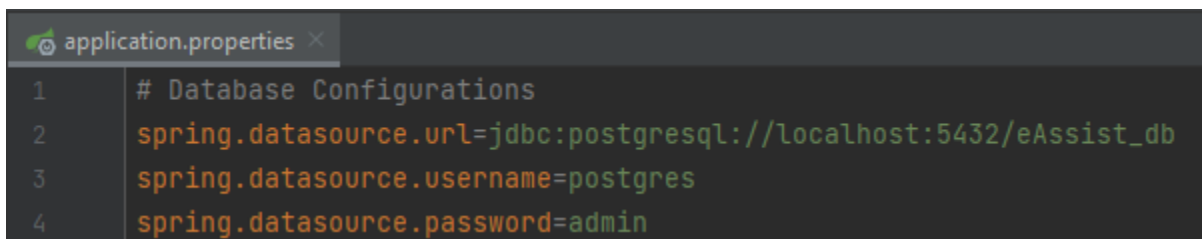


Back-end API

Για τη διαχείριση αιτημάτων έχει υλοποιηθεί αντίστοιχο API. Για την εκτέλεση της, αρκεί να ανοίξετε το project στο φάκελο **Spring_Boot_Backend** στο IntelliJ IDEA και να τρέξετε την εφαρμογή. Η εφαρμογή τρέχει στη διεύθυνση IP του localhost στην πόρτα 8080 (θα πρέπει να μην χρησιμοποιείται ήδη από άλλο πρόγραμμα ή εφαρμογή).

Το προκατασκευασμένο μενού υπάρχει ήδη μέσα στο φάκελο του project. Αν θέλετε να χρησιμοποιήσετε το δικό σας, απλά αντικαταστήστε το αρχείο **menu.json** (το όνομα του αρχείου πρέπει να παραμείνει ίδιο).

Πριν τρέξετε στην εφαρμογή θα πρέπει να έχετε δημιουργήσει μία βάση δεδομένων στη PostgreSQL που να συμφωνεί με τα παρακάτω στοιχεία (ή να προσαρμόσετε το αρχείο **application.properties** ανάλογα με τη δική σας βάση δεδομένων και στοιχεία χρήστη).



```
application.properties x
1 # Database Configurations
2 spring.datasource.url=jdbc:postgresql://localhost:5432/eAssist_db
3 spring.datasource.username=postgres
4 spring.datasource.password=admin
```

- **eAssist_db**: όνομα της βάσης δεδομένων
- **postgres**: όνομα χρήστη στον οποίο ανήκει η βάση
- **admin**: κωδικός χρήστη

Κατά την εκκίνηση του back-end δημιουργούνται μαζί με τη βάση δεδομένων και κάποιες dummy εγγραφές για λόγους παρουσίασης. Συγκεκριμένα:

- 3 διαφορετικοί ρόλοι (employee, manager, admin)
- 4 χρήστες με διαφορετικούς ρόλους ο καθένας (**peter**-admin, **jim**-manager, **athena**-employee, **timmy**-employee, manager). Χάρη του παραδείγματος εκτέλεσης, ο κωδικός για όλους τους χρήστες είναι “123456”.
- 10 τραπέζια (με διαφορετικά ονόματα)
- Όλα τα αντικείμενα/ κατηγορίες από το προκατασκευασμένο μενού
- 8 διαφορετικές παραγγελίες που αντιστοιχούν σε 8 διαφορετικά τραπέζια, οι οποίες συνολικά χρησιμοποιούν 16 διαφορετικά αντικείμενα σε διαφορετικές ποσότητες

Δεν χρειάζεται να πραγματοποιήσετε κάποια άλλη ενέργεια σχετικά με το back-end. Το επόμενο βήμα είναι να τρέξετε την κύρια εφαρμογή.

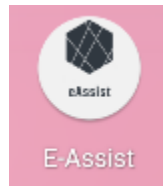
Στο φάκελο **Spring_Boot_Backend** στο εκτός του project βρίσκονται επίσης:

- Οι dummy εγγραφές σε μορφή αρχείου sql (**insert.sql**)
- Εντολές select για τους πίνακες της βάσης δεδομένων (**select.sql**)
- Εντολές create table της βάσης δεδομένων (**create_table.sql**)
- Όλα τα endpoints του API σε postman requests (**postman_requests.json**). Άπλα κάνετε import το αρχείο στο postman και τρέχετε τα requests (θα πρέπει να συμπληρωθούν τα authorization headers με τα αντίστοιχα tokens. Περισσότερες πληροφορίες σχετικά με τα json web tokens στην ενότητα “*Αρχιτεκτονική συστήματος*”)

Android εφαρμογή

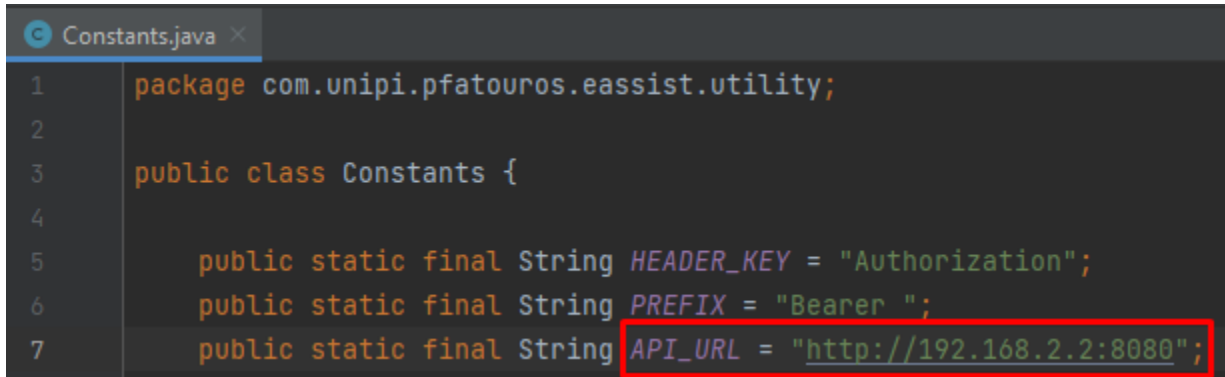
Αποτελεί front-end/ user interface της πτυχιακής. Στο παρών documentation αναφέρεται και ως “*κύρια εφαρμογή*”. Μέσω αυτής ο χρήστης επικοινωνεί/ στέλνει αιτήματα στο back-end. Για την εκτέλεση της, αρκεί να ανοίξετε το project στο φάκελο **Android_UI** στο Android Studio και να τρέξετε την εφαρμογή σε κάποιο ενσωματωμένο emulator ή εξωτερική συσκευή.

Αν δεν τρέξει αυτόματα, εκκινήστε την εφαρμογή πατώντας στο εικονίδιο της. Η εφαρμογή φαίνεται ως εξής στην android συσκευή:



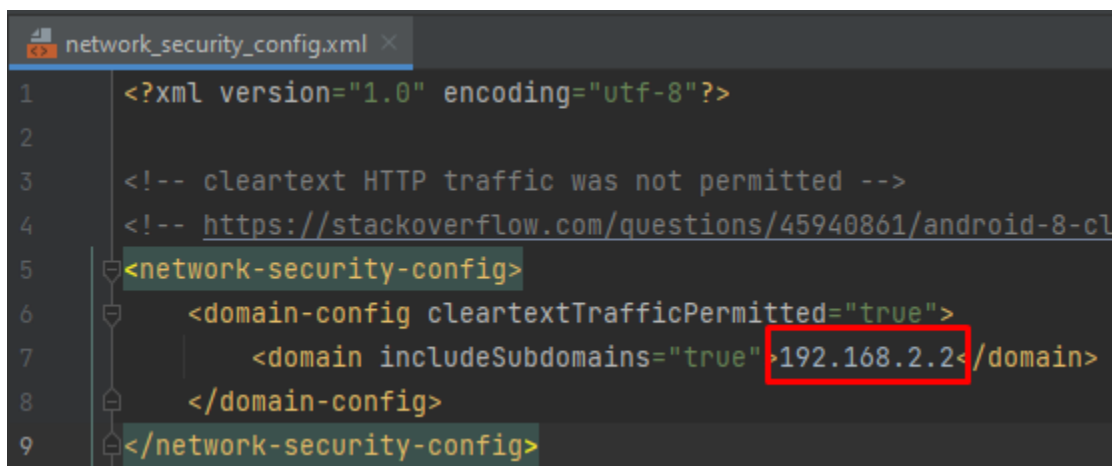
Πριν τρέξετε την εφαρμογή θα πρέπει να τροποποιήσετε 2 αρχεία του project. Συγκεκριμένα:

- Στο αρχείο **Constants.java** στο package **utility** αλλάξτε την σταθερά **API_URL** με την αντίστοιχη IPv4 address του υπολογιστή σας



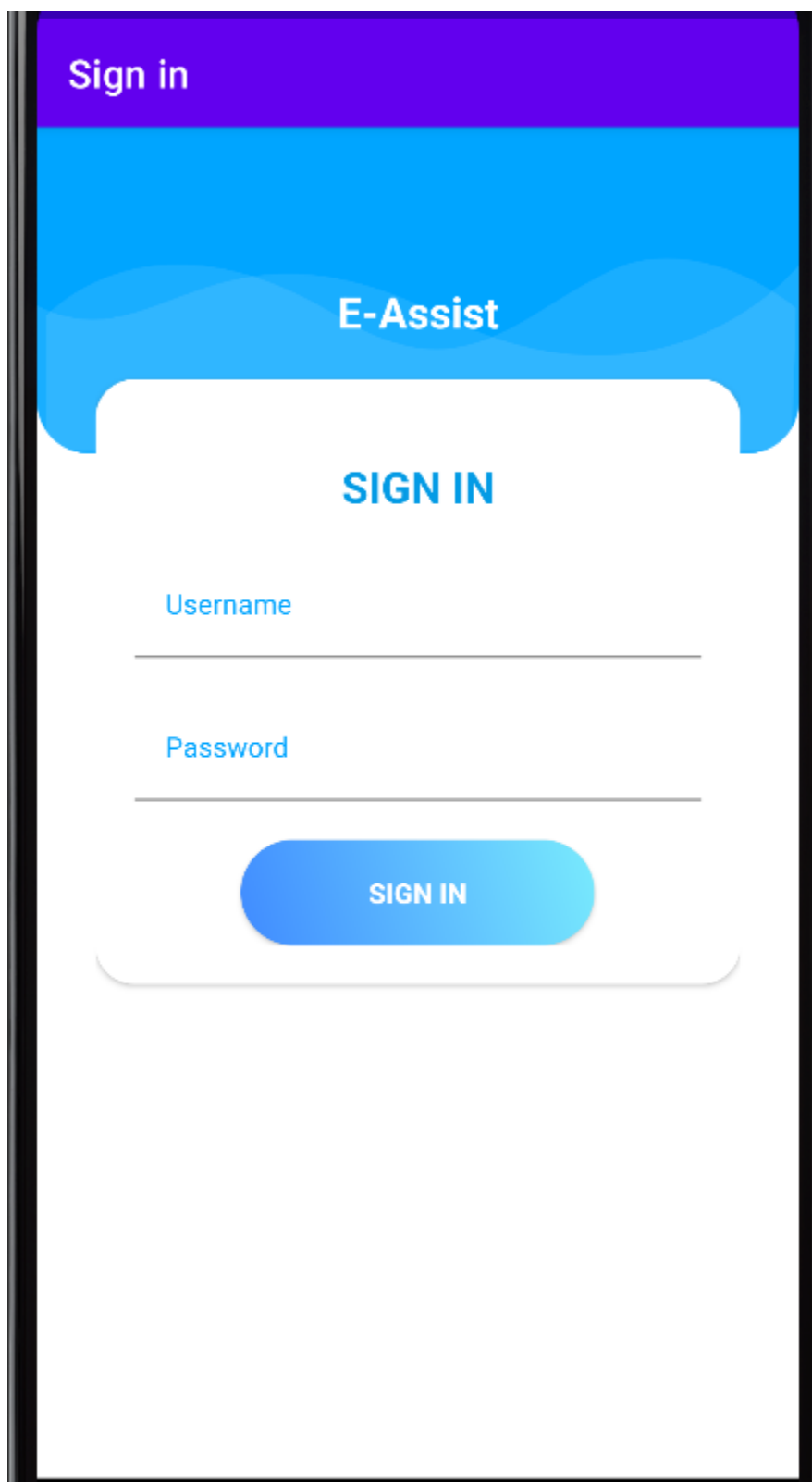
```
1 package com.unipi.pfatouros.eassist.utility;
2
3 public class Constants {
4
5     public static final String HEADER_KEY = "Authorization";
6     public static final String PREFIX = "Bearer ";
7     public static final String API_URL = "http://192.168.2.2:8080";
```

- Στο αρχείο **network_security_config.xml** στο directory **xml** αλλάξτε το domain με την αντίστοιχη IPv4 address του υπολογιστή σας



```
1 <?xml version="1.0" encoding="utf-8"?>
2
3 <!-- cleartext HTTP traffic was not permitted -->
4 <!-- https://stackoverflow.com/questions/45940861/android-8-cl
5 <network-security-config>
6     <domain-config cleartextTrafficPermitted="true">
7         <domain includeSubdomains="true">192.168.2.2</domain>
8     </domain-config>
9 </network-security-config>
```

Αφού ανοίξετε την εφαρμογή θα εμφανιστεί η εξής οθόνη:



The image shows a mobile application interface for signing in. At the top, there is a purple header bar with the text "Sign in" in white. Below this is a blue banner with the text "E-Assist" in white. The main content area is white and contains a "SIGN IN" title in blue. Underneath the title are two input fields: "Username" and "Password", both in blue text. Below the password field is a blue button with the text "SIGN IN" in white. The entire interface is framed by a black border.

Εδώ πραγματοποιείται το authentication. Αυτή τη στιγμή στη βάση δεδομένων υπάρχουν 4 χρήστες που μπορούν να συνδεθούν, οι οποίοι συνολικά καλύπτουν όλους τους διαθέσιμους ρόλους. Σε περίπτωση αποτυχημένης σύνδεσης, εμφανίζεται ενημερωτικό μήνυμα.

Σημείωση: για να λειτουργήσει το authentication, και γενικότερα οποιοδήποτε άλλη λειτουργία της android εφαρμογής, απαραίτητη προϋπόθεση αποτελεί να τρέχει το back-end API.

Η επόμενη οθόνη κάνει prompt το χρήστη να επιλέξει σε τι είδους δράσεις θέλει να αποκτήσει πρόσβαση. Εδώ πραγματοποιείται το authorization. Αν ο χρήστης επιλέξει ρόλο τον οποίο δεν έχει τότε δεν επιτρέπεται η πρόσβαση στις δράσεις του επιλεγμένου ρόλου και εμφανίζεται σχετικό μήνυμα.

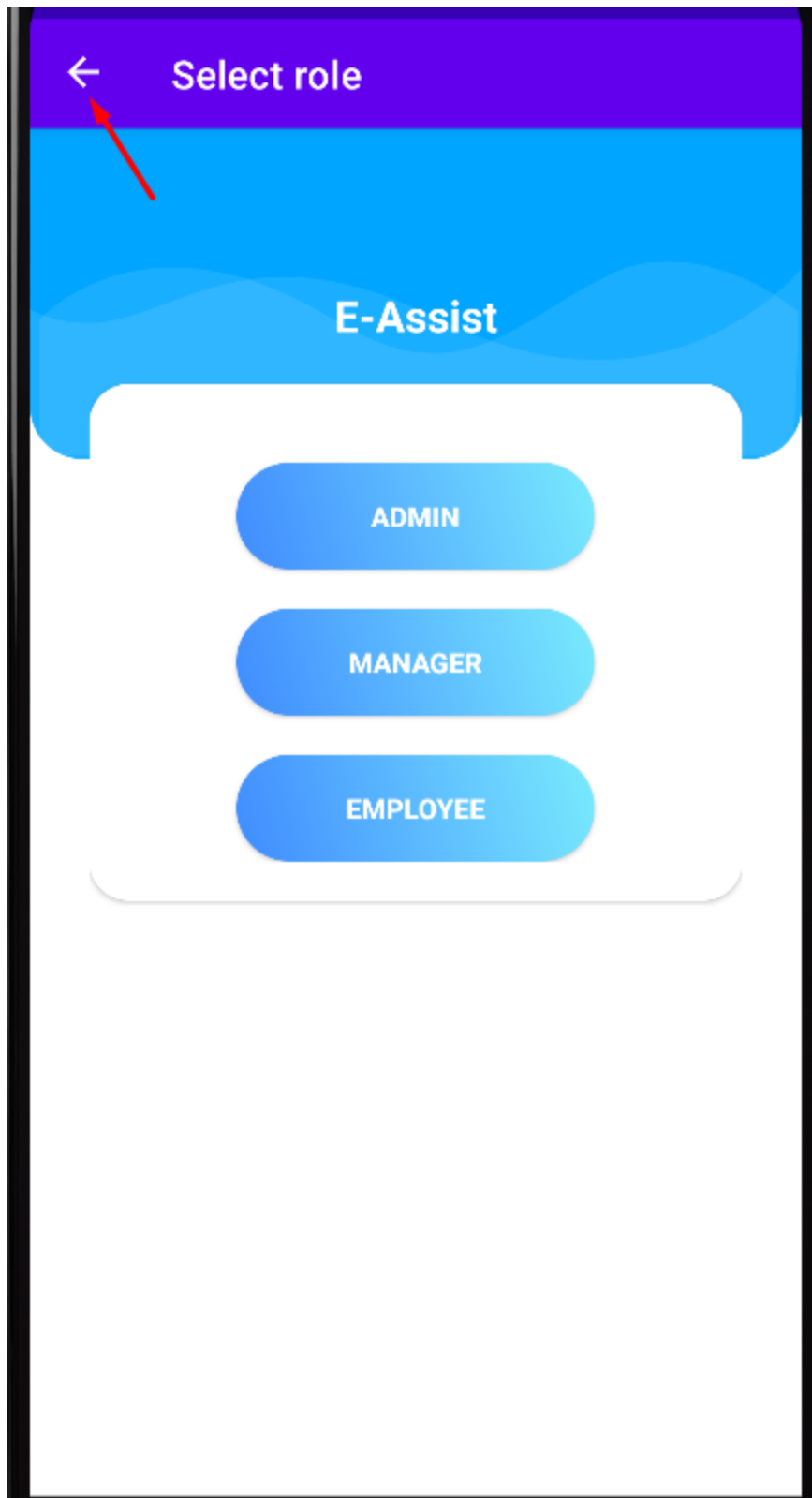
Πατώντας το βελάκι πάνω αριστερά, ο χρήστης επιστρέφει στην αρχική οθόνη και του ζητείται να ξανασυνδεθεί.

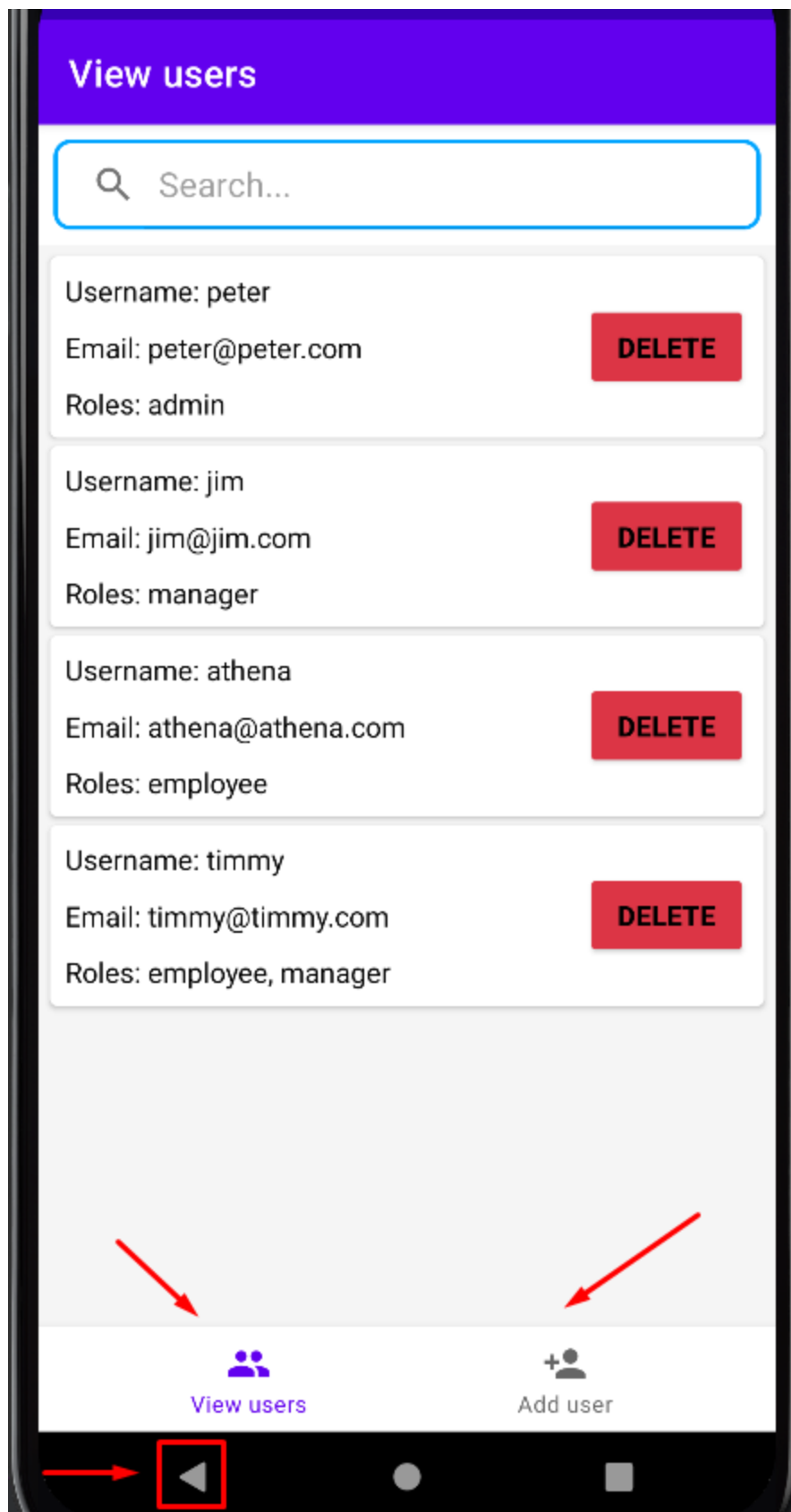
Θα ακολουθήσουν παραδείγματα όλων των δράσεων για καθένα από τους 3 ρόλους, ξεκινώντας από το ρόλο **admin**.

Admin

Ο διαχειριστής έχει τις εξής δράσεις:

- Προβολή και αναζήτηση χρηστών
- Διαγραφή χρηστών
- Δημιουργία νέων χρηστών





Στην πρώτη οθόνη ο διαχειριστής έχει την δυνατότητα να δει τα στοιχεία όλων των χρηστών, να αναζητήσει χρήστες με βάση το username τους, το email τους και τους ρόλους τους, καθώς και να διαγράψει οποιονδήποτε χρήστη επιθυμεί πατώντας το κουμπί **DELETE** που βρίσκεται δεξιά από τα στοιχεία κάθε χρήστη.

Ενδεικτικά ακολουθούν μερικά screenshots από αναζητήσεις χρηστών:

A screenshot of a web application showing search results for the term "manager". At the top is a search bar with a magnifying glass icon on the left and a close "X" icon on the right. Below the search bar, two user profiles are listed. Each profile contains the username, email, roles, and a red "DELETE" button.

Username	Email	Roles	Action
jim	jim@jim.com	manager	DELETE
timmy	timmy@timmy.com	employee, manager	DELETE

A screenshot of the user search interface with the email "peter@peter.com" entered in the search bar. One user profile is displayed with the details: Username: peter, Email: peter@peter.com, Roles: admin, and a red "DELETE" button.

Username	Email	Roles	Action
peter	peter@peter.com	admin	DELETE

A screenshot of the user search interface with the username "timmy" entered in the search bar. One user profile is displayed with the details: Username: timmy, Email: timmy@timmy.com, Roles: employee, manager, and a red "DELETE" button.

Username	Email	Roles	Action
timmy	timmy@timmy.com	employee, manager	DELETE

Χρησιμοποιώντας το μενού στο κάτω μέρος της οθόνης, ο χρήστης μπορεί να πλοηγηθεί στην εφαρμογή. Με έντονο μωβ φαίνεται η τρέχων οθόνη, ενώ με γκρι χρώμα όλες οι άλλες επιλογές. Πατώντας στο αντίστοιχο εικονίδιο/ περιοχή της οθόνης ο χρήστης οδηγείται σε μια άλλη οθόνη με διαφορετική(ές) δράση(εις). Παρόμοια μενού υπάρχουν σε όλη την έκταση της εφαρμογής και λειτουργούν με το ίδιο τρόπο.

Πατώντας το βελάκι του κινητού για να πάει “πίσω”, ο χρήστης επιστρέφει στην οθόνη που πραγματοποιείται το authorization. Ανεξάρτητα του ρόλου που έχει επιλέξει ο χρήστης (εφόσον έχει επιλέξει ρόλο και έχει μεταβεί στις αντίστοιχες λειτουργίες της εφαρμογής), αν πατήσει το βελάκι του κινητού για να πάει “πίσω”, πάντα θα επιστρέφει στην ίδια οθόνη και θα του ζητείται να επιλέξει ρόλο.

Επιλέγοντας την επιλογή **Add user**, ο διαχειριστής έχει τη δυνατότητα να εισάγει ένα νέο χρήστη στην εφαρμογή. Οι πληροφορίες του νέου χρήστη πρέπει να πληρούν τις εξής προϋποθέσεις:

- Το username πρέπει να αποτελείται τουλάχιστον από 3 χαρακτήρες και να είναι μοναδικό.
- Το email πρέπει να μοναδικό
- Ο κωδικός πρέπει να αποτελείται τουλάχιστον από 6 χαρακτήρες
- Πρέπει να ανατεθεί στον χρήστη τουλάχιστον ένας ρόλος

Σε οποιαδήποτε άλλη περίπτωση εμφανίζεται ενημερωτικό μήνυμα.

Ακολουθεί στιγμιότυπο από την οθόνη μέσω της οποίας δημιουργείτε νέος χρήστης:

Add user

Username

Email

Password

Employee ☐

Manager ☐

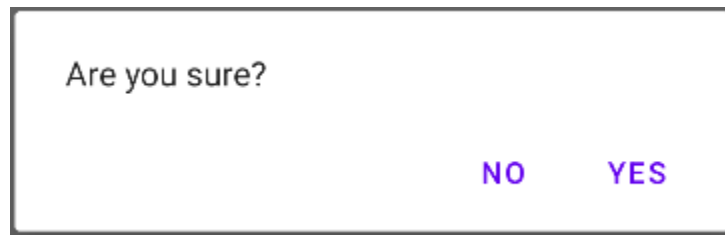
Admin ☐

ADD USER

View users

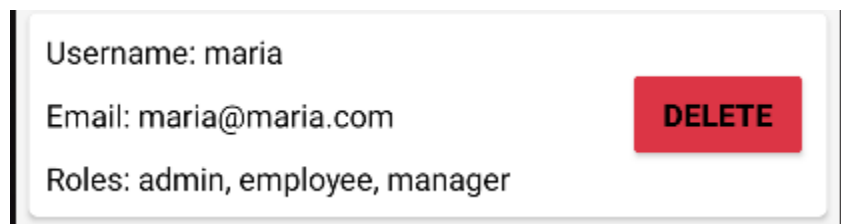
Add user

Ενδεικτικά δημιουργούμε ένα νέο χρήστη. Πριν την ολοκλήρωση της εγγραφής, ο διαχειριστής ρωτάτε από την εφαρμογή αν είναι σίγουρος και μόνο αν πατήσει την επιλογή “yes” συνεχίζεται η διαδικασία, αλλιώς ακυρώνεται.

A white rectangular dialog box with a thin grey border. Inside, the text "Are you sure?" is centered at the top. At the bottom right, there are two buttons: "NO" and "YES", both in a purple color.

Σημείωση: το ίδιο μήνυμα επιβεβαίωσης εμφανίζεται επίσης όταν ο διαχειριστής επιλέξει να διαγράψει κάποιον χρήστη.

Ο νέος χρήστης δημιουργήθηκε. Μπορούμε να τον δούμε αν επιστρέψουμε στην προηγούμενη οθόνη μέσω του μενού.

A white rectangular card with a thin grey border, flanked by vertical black bars on the left and right. It contains three lines of text: "Username: maria", "Email: maria@maria.com", and "Roles: admin, employee, manager". On the right side of the card, there is a red button with the word "DELETE" in white capital letters.


Ο διαχειριστής μπορεί να επιλέξει να διαγράψει τον εαυτό του αν θελήσει αλλά αυτό θα έχει ως αποτέλεσμα να μην μπορεί να χρησιμοποιήσει πλέον την εφαρμογή και να αναγκαστεί να επιστρέψει στην αρχική οθόνη για εκ νέου σύνδεση (authentication).

Εδώ ολοκληρώνεται η παρουσίαση των δράσεων του ρόλου **admin**. Ακολουθεί παρουσίαση των δράσεων του ρόλου **manager**.

Manager

Η πρώτη οθόνη που εμφανίζεται στον μάνατζερ είναι η εξής:

Inventory

 Search...

Item: Martini

Category: Cocktails

Price: 10

Quantity: 10

EDIT

Item: Brooklyn

Category: Cocktails

Price: 14

Quantity: 10

EDIT

Item: Margarita

Category: Cocktails

Price: 15

Quantity: 10

EDIT

Item: Irish Coffee


Category: Cocktails


Price: 13

Quantity: 10

EDIT

Item: Mai Tai


Inventory


Statistics

Ο μάνατζερ έχει τις εξής δράσεις:

- Προβολή και αναζήτηση των αντικείμενων του μενού
- Προσαρμογή των τιμών και των διαθέσιμων ποσοτήτων των αντικείμενων του μενού
- Προβολή στατιστικών σχετικά με τις καταχωρημένες παραγγελίες του συστήματος

Στην πρώτη οθόνη ο μάνατζερ έχει την δυνατότητα να δει τα στοιχεία όλων των αντικείμενων του μενού, να αναζητήσει αντικείμενα με βάση το όνομά τους και την κατηγορία τους, καθώς και να προσαρμόσει οποιαδήποτε αντικείμενα του μενού επιθυμεί πατώντας το κουμπί **EDIT** που βρίσκεται δεξιά από τα στοιχεία κάθε αντικειμένου.

Ενδεικτικά ακολουθούν μερικά screenshots από αναζητήσεις αντικείμενων του μενού

beers

Item: **Pisco Sour**

Category: Beers

Price: 9

Quantity: 10

EDIT

Item: **Cosmopolitan**

Category: Beers

Price: 7

Quantity: 10

EDIT

Item: **Las Word**

Category: Beers

Price: 10

Quantity: 10

EDIT

Item: **Negroni**

Category: Beers

Price: 12

Quantity: 10

EDIT

Item: **Jackrose**

Inventory

Statistics

Q martini

X

Item: Martini

Category: Cocktails

Price: 10

Quantity: 10

EDIT

Item: French Martini

Category: Mocktails

Price: 10

Quantity: 10

EDIT

Πατώντας το κουμπί **EDIT** που βρίσκεται δεξιά των πληροφοριών κάθε αντικειμένου, ο μάνατζερ μπορεί να προσαρμόσει την τιμή και την διαθέσιμη ποσότητα οποιουδήποτε αντικειμένου του μενού.

Edit quantity

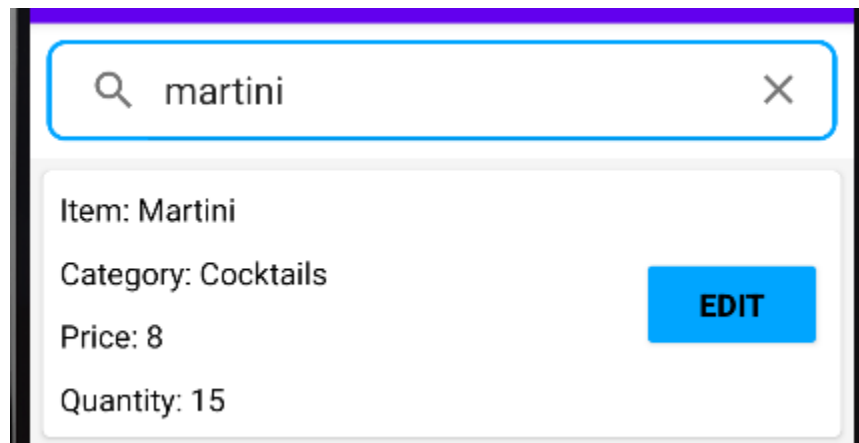
New quantity

Edit price

New price

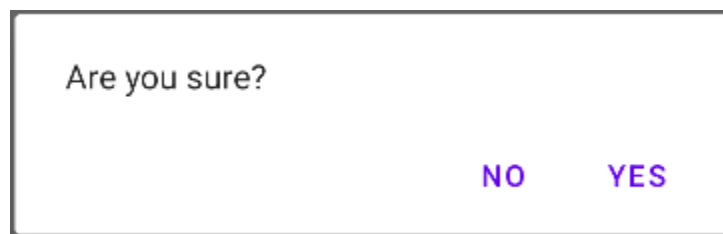
EDIT

Στο συγκεκριμένο παράδειγμα θα αλλάξουμε την τιμή και την διαθέσιμη ποσότητα του “*Martini*”.



A screenshot of a search results modal. At the top is a search bar with a magnifying glass icon on the left, the text 'martini', and a close 'X' icon on the right. Below the search bar, the modal displays the following information: 'Item: Martini', 'Category: Cocktails', 'Price: 8', and 'Quantity: 15'. To the right of this information is a blue button with the text 'EDIT' in white capital letters.

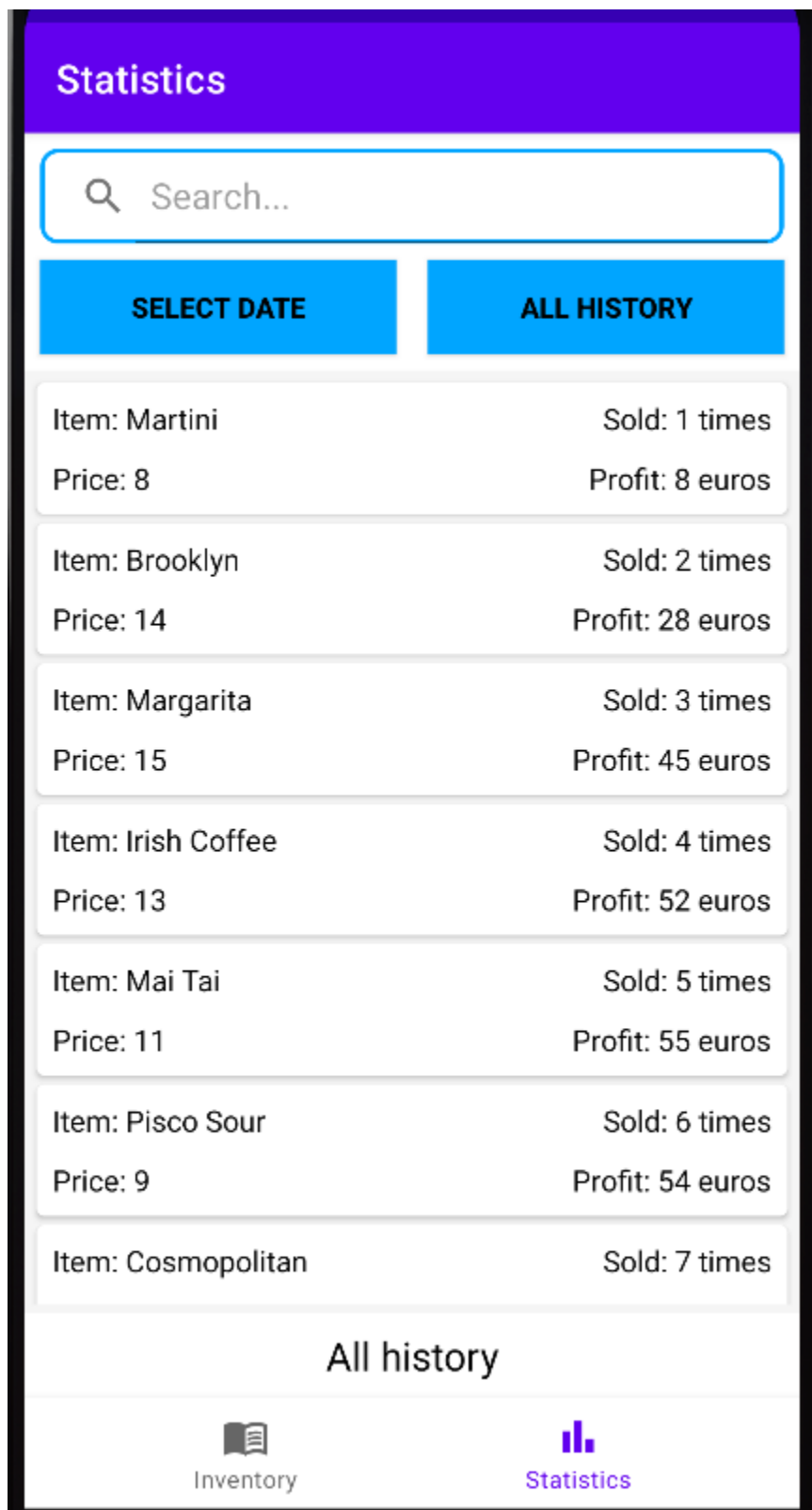
Πριν την ολοκληρωθεί οποιαδήποτε αλλαγή ο μάνατζερ ρωτάτε από την εφαρμογή αν είναι σίγουρος και μόνο αν πατήσει την επιλογή “yes” συνεχίζεται η διαδικασία, αλλιώς ακυρώνεται.



A confirmation dialog box with a thin gray border. It contains the text 'Are you sure?' in the center. At the bottom right, there are two buttons: 'NO' and 'YES', both in purple capital letters.

Σημείωση: μέγιστη τιμή οποιουδήποτε αντικειμένου είναι 1000 € και μέγιστη διαθέσιμη ποσότητα είναι 10000.

Όσον αφορά τα στατιστικά, πατώντας πάνω στο εικονίδιο **Statistics**, η οθόνη που εμφανίζεται είναι η εξής:

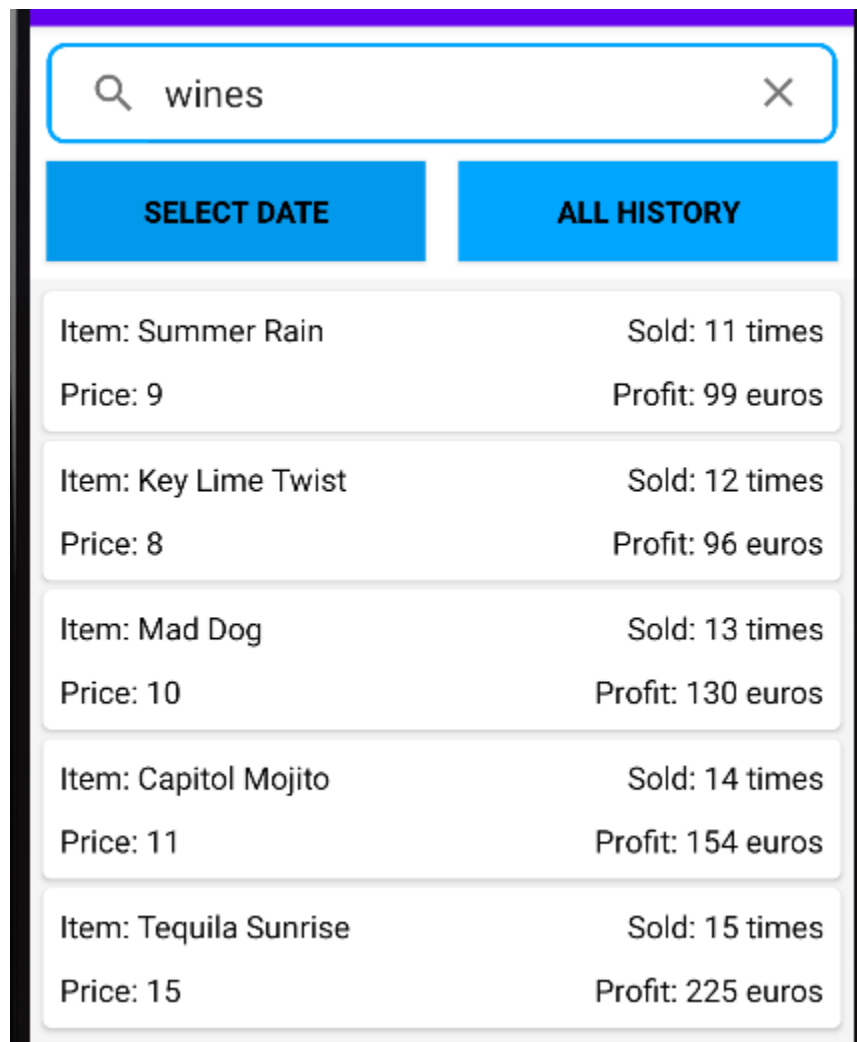


Εδώ εμφανίζονται στατιστικά σχετικά με τις καταχωρημένες παραγγελίες του συστήματος. Συγκεκριμένα, σε κάθε εγγραφή εμφανίζεται:

- Το όνομα του αντικειμένου
- Η τιμή του αντικειμένου
- Το πλήθος των φορών που πουλήθηκε
- Το συνολικό κέρδος από τις πωλήσεις του

Ο μάνατζερ μπορεί να κάνει αναζήτηση με βάση το όνομα του προϊόντος, τη κατηγορία του προϊόντος, τη δημοσιότητα του (πλήθος φορών που πουλήθηκε) και το συνολικό κέρδος που παρήγαγε.

Ακολουθούν ενδεικτικά screenshots αναζητήσεων:



The screenshot shows a web interface for wine sales. At the top, there is a search bar with the text 'wines' and a magnifying glass icon on the left, and a close 'X' icon on the right. Below the search bar are two blue buttons: 'SELECT DATE' and 'ALL HISTORY'. The main content area displays a list of wine items with their respective sales and profit data. Each item is shown in a light gray box with a white background, containing the item name, price, quantity sold, and profit in euros.

Item	Price	Sold	Profit
Summer Rain	9	11 times	99 euros
Key Lime Twist	8	12 times	96 euros
Mad Dog	10	13 times	130 euros
Capitol Mojito	11	14 times	154 euros
Tequila Sunrise	15	15 times	225 euros

tequila sunrise

×

SELECT DATE

ALL HISTORY

Item: Tequila Sunrise

Sold: 15 times

Price: 15

Profit: 225 euros

sold

×

SELECT DATE

ALL HISTORY

Item: Martini

Sold: 1 times

Price: 8

Profit: 8 euros

Item: Brooklyn

Sold: 2 times

Price: 14

Profit: 28 euros

Item: Margarita

Sold: 3 times

Price: 15

Profit: 45 euros

Item: Irish Coffee

Sold: 4 times

Price: 13

Profit: 52 euros

Item: Mai Tai

Sold: 5 times

Price: 11

Profit: 55 euros

Item: Pisco Sour

Sold: 6 times

Price: 9

Profit: 54 euros

Item: Cosmopolitan

Sold: 7 times

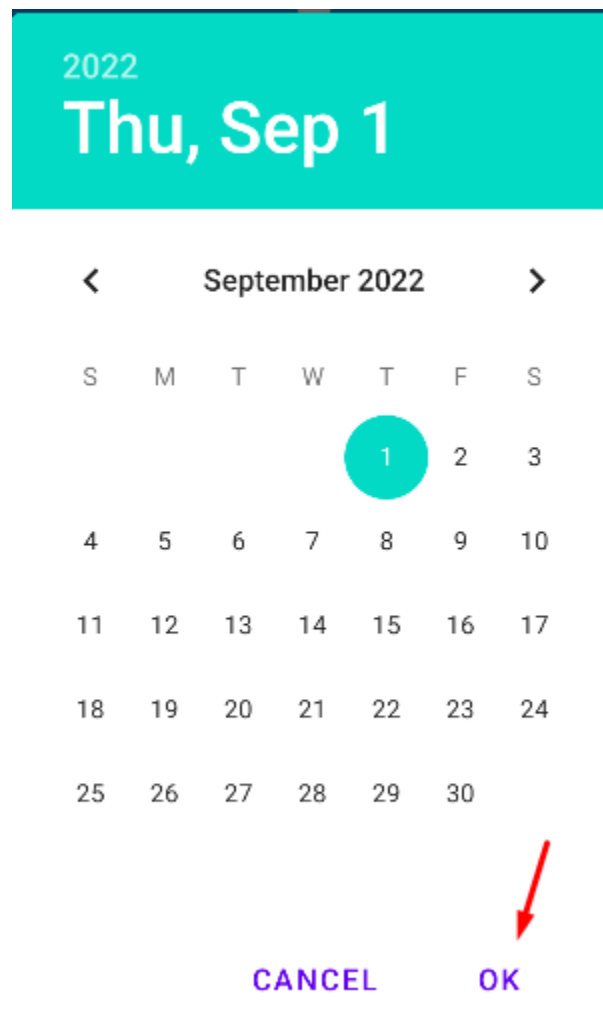
All history

Πέρα από την επιλογή **ALL HISTORY** που δείχνει στατιστικά με βάση όλες τις καταχωρημένες παραγγελίες στη βάση δεδομένων, υπάρχει και η επιλογή **SELECT DATE** στην οποία ο μάνατζερ μπορεί να επιλέξει μία συγκεκριμένη ημερομηνία και να δει μεμονωμένα στατιστικά μόνο για αυτή.



Αν η συγκεκριμένη ημερομηνία δεν έχει καταχωρημένες παραγγελίες δεν θα εμφανιστούν στατιστικά. Από τα dummy δεδομένα έχουμε 4 παραγγελίες που αντιστοιχούν στην σημερινή ημερομηνία (την εκάστοτε σημερινή ημερομηνία την οποία έγινε η εκκίνηση του back-end API) και άλλες 4 παραγγελίες που αντιστοιχούν στην 1^η Σεπτεμβρίου του 2022.

Αφού πατήσουμε το κουμπί **SELECT DATE** εμφανίζεται ένα ημερολόγιο, σε αυτό για χάρη παραδείγματος την 1^η Σεπτεμβρίου και πατάμε “OK”.



Ακολουθεί στιγμιότυπο από τα στατιστικά της 1^{ης} Σεπτεμβρίου:

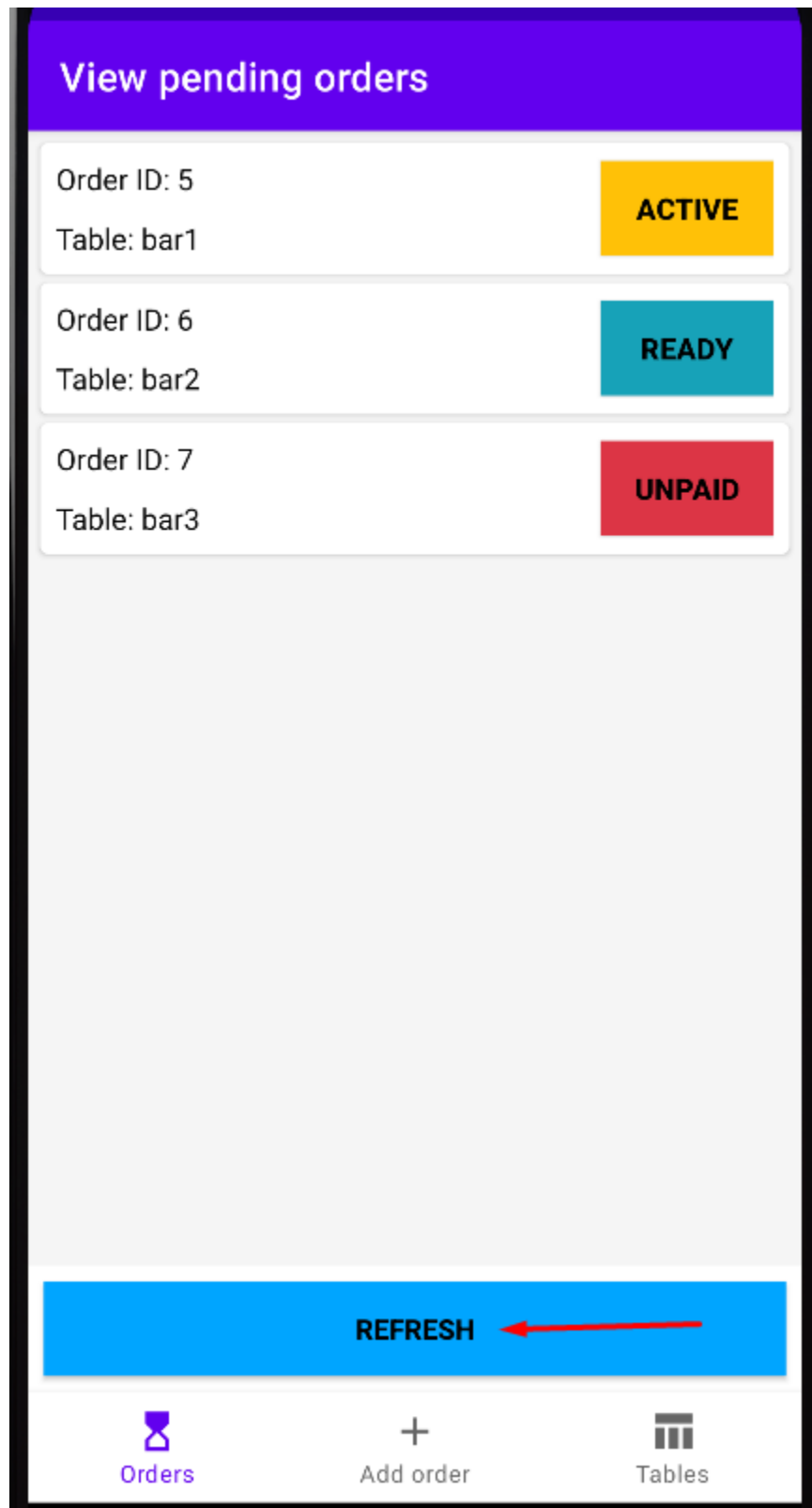
<input type="text" value="Search..."/>	
<div><div>SELECT DATE</div><div>ALL HISTORY</div></div>	
Item: Martini	Sold: 1 times
Price: 8	Profit: 8 euros
Item: Brooklyn	Sold: 2 times
Price: 14	Profit: 28 euros
Item: Margarita	Sold: 3 times
Price: 15	Profit: 45 euros
Item: Irish Coffee	Sold: 4 times
Price: 13	Profit: 52 euros
Item: Mai Tai	Sold: 5 times
Price: 11	Profit: 55 euros
Item: Pisco Sour	Sold: 6 times
Price: 9	Profit: 54 euros
Item: Cosmopolitan	Sold: 7 times
Date: 2022-09-01	

Σημείωση: η αναζήτηση και ταξινόμηση λειτουργεί κανονικά με τον ίδιο τρόπο όπως και με την επιλογή **ALL HISTORY**.

Εδώ ολοκληρώνεται η παρουσίαση των δράσεων του ρόλου **manager**. Ακολουθεί παρουσίαση των δράσεων του ρόλου **employee**.

Employee

Η πρώτη οθόνη που εμφανίζεται στον εργαζόμενο είναι η εξής:



Στην συγκεκριμένη οθόνη φαίνονται οι σημερινές (εκάστοτε σημερινής ημέρας, ανάλογα με την τρέχων ημερομηνία) τρέχων παραγγελίες. Κάθε παραγγελία ξεχωρίζει με ένα μοναδικό ID. Κάθε παραγγελία αντιστοιχεί σε κάποιο τραπέζι. Μπορεί πολλές παραγγελίες να αντιστοιχούν στο ίδιο τραπέζι αλλά το αντίστροφο δεν ισχύει, δηλαδή μόνο ένα τραπέζι αντιστοιχεί σε κάθε παραγγελία. Επιπλέον, κάθε παραγγελία διακρίνεται και από μία κατάσταση (status). Πατώντας πάνω στο κουμπί με το status της παραγγελίας, αλλάζει το status της στην επόμενη κατάσταση.

Υπάρχουν 4 καταστάσεις που χρησιμοποιούνται στην εφαρμογή:

- **ACTIVE:** μόλις η παραγγελία εισαχθεί στο σύστημα, αυτόματα μπαίνει σε αυτή την κατάσταση
- **READY:** η επόμενη κατάσταση μιας παραγγελιάς, υποδηλώνει πως η παραγγελία είναι έτοιμη για παράδοση
- **UNPAID:** η επόμενη κατάσταση μιας έτοιμης παραγγελιάς, υποδηλώνει πως η παραγγελία έχει παραδοθεί αλλά δεν έχει εξοφληθεί
- **COMPLETE:** η τελική κατάσταση μιας παραγγελιάς, υποδηλώνει πως η συγκεκριμένη παραγγελία έχει ολοκληρωθεί επιτυχώς

Στην οθόνη **View pending orders** εμφανίζονται όλες οι σημερινές (εκάστοτε σημερινές, ανάλογα με την τρέχων ημερομηνία) παραγγελίες που δεν έχουν ολοκληρωθεί. Στο παράδειγμα εκτέλεσης υπάρχουν 4 παραγγελίες, μία για κάθε διαφορετικό status.

Αλλάζουμε το status την παραγγελίας με ID το νούμερο 5 πατώντας στο αντίστοιχο κουμπί. Το status θα αλλάξει σε **READY**. Ακολουθεί ενδεικτικό screenshot:

Order ID: 5 Table: bar1	READY
Order ID: 6 Table: bar2	READY
Order ID: 7 Table: bar3	UNPAID

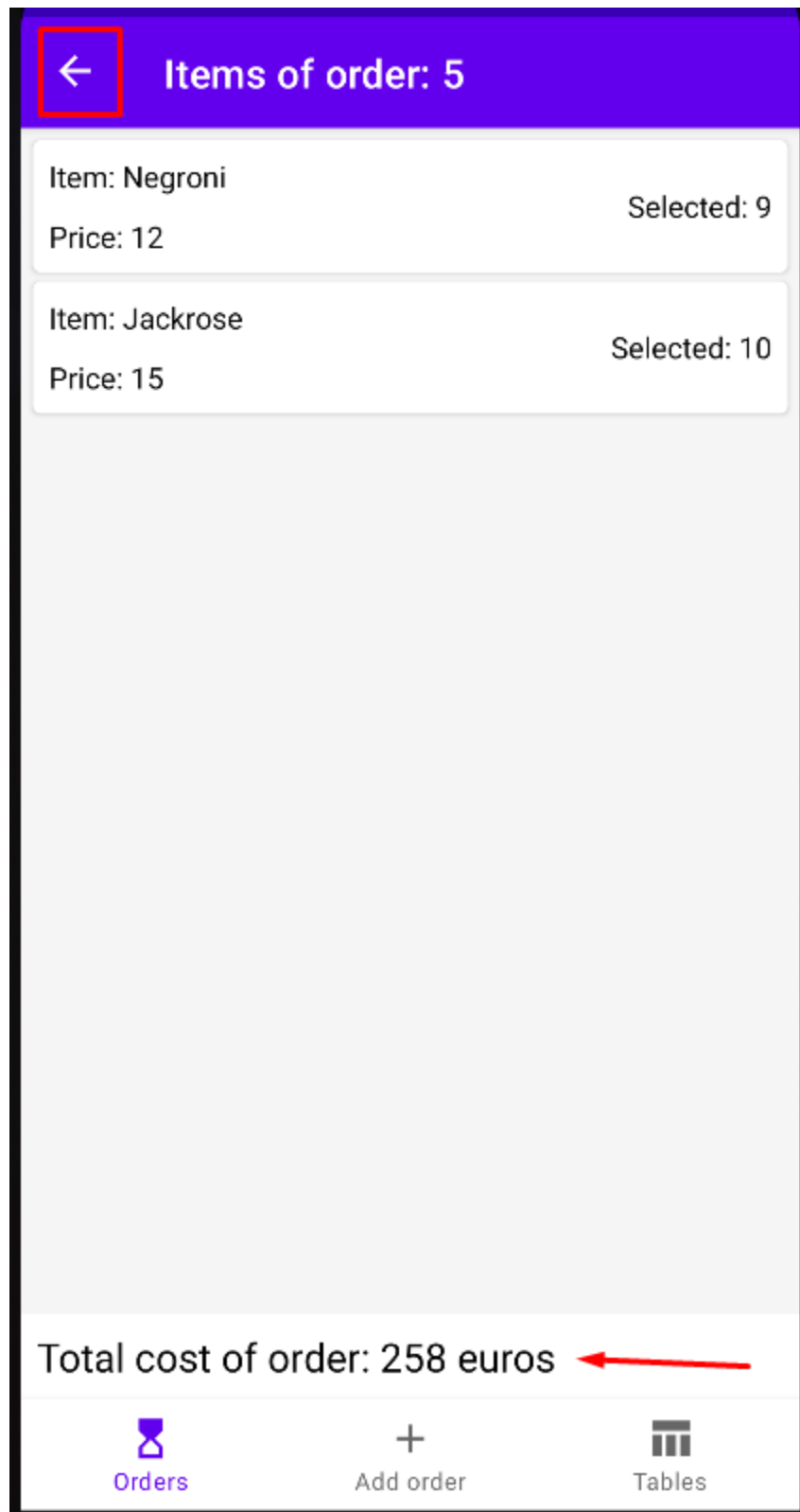
Παραγγελίες μπορούν να εισάγονται και να ενημερώνονται οποιαδήποτε στιγμή αφού η εφαρμογή έχει υλοποιηθεί με σκοπό να χρησιμοποιείται από πολλούς χρήστες ταυτόχρονα. Ο εργαζόμενος μπορεί να ανανεώσει την οθόνη με τις τρέχων παραγγελίες πατώντας το κουμπί **REFRESH**.

Πατώντας πάνω στο σώμα της παραγγελίας (αλλά όχι πάνω στο κουμπί status), εμφανίζονται περισσότερες πληροφορίες σχετικά με την παραγγελία.



Συγκεκριμένα, εμφανίζονται τα επιλεγμένα προϊόντα, η ποσότητα στην οποία ζητήθηκαν, το κόστος τους, καθώς και το συνολικό κόστος της παραγγελίας (στο κάτω μέρος της οθόνης).

Ακολουθεί ενδεικτικό στιγμιότυπο οθόνης:



Σημείωση: πατώντας το βελάκι πάνω αριστερά, ο εργαζόμενος επιστρέφει στις τρέχων παραγγελίες.

Επιλέγοντας το εικονίδιο **Add order** ο εργαζόμενος, ακολουθώντας τις οδηγίες, μπορεί να εισάγει παραγγελίες στο σύστημα.

Αρχικά ο εργαζόμενος επιλέγει τραπέζι. Στη λίστα εμφανίζονται μόνο τα διαθέσιμα τραπέζια.

Select table

Table: beach1

Table: beach2

Table: beach3

Table: beach4

Table: beach5

Table: bar4

Table: bar5



Orders



Add order



Tables

Έστω ότι η νέα παρέα που κατέφθασε στο μαγαζί επιλέγει να κάτσει στο τραπέζι με όνομα **beach1**.

← Order for table: beach1

🔍 Search...

Item: Brooklyn

Price: 14

Quantity: 10

Selected: 0

Item: Margarita

Price: 15

Quantity: 10

Selected: 0

Item: Irish Coffee

Price: 13

Quantity: 10

Selected: 0

Item: Mai Tai

Price: 11

Quantity: 10

Selected: 0

Item: Pisco Sour

Price: 9

Quantity: 10

Selected: 0

REVIEW ORDER

🕒 Orders

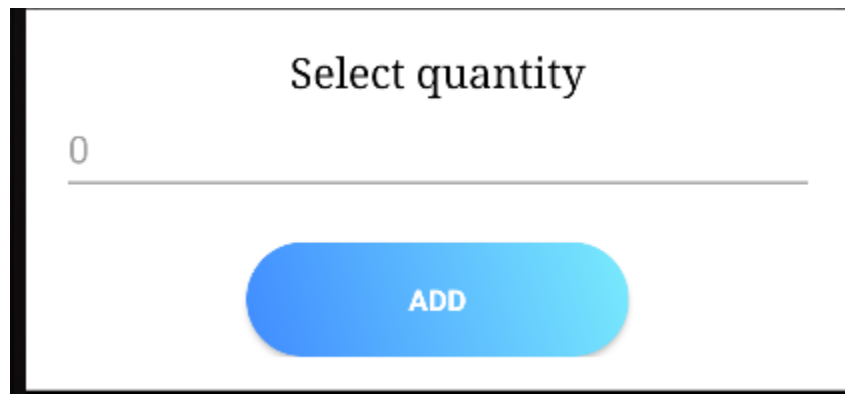
+

Add order

🏠 Tables

Ακριβώς όπως ο μάνατζερ αναζητά αντικείμενα από το μενού, έτσι και ο εργαζόμενος έχει την ίδια δυνατότητα αναζήτησης αντικειμένων με βάση το όνομα και με βάση την κατηγορία τους.

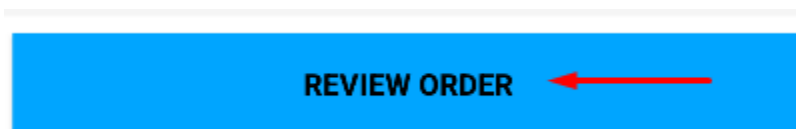
Πατώντας στο σώμα του αντικειμένου που θέλει να προσθέσει στην παραγγελία, εμφανίζεται το εξής:



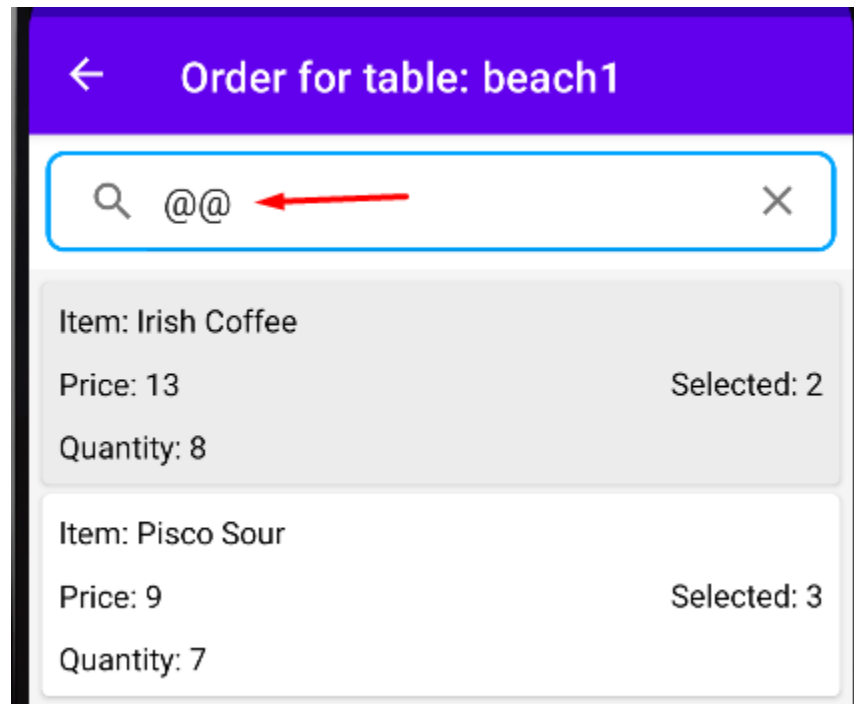
The image shows a rectangular box with a black border. Inside, the text "Select quantity" is centered at the top. Below it is a text input field with a horizontal line and the number "0" on the left. At the bottom center is a blue rounded rectangular button with the word "ADD" in white capital letters.

Εδώ ο εργαζόμενος συμπληρώνει την επιλεγμένη από τον πελάτη ποσότητα. Η μέγιστη επιλεγμένη ποσότητα δεν πρέπει να ξεπερνά τα 100 κομμάτια (σε περίπτωση που πάνω από 100 είναι διαθέσιμα) ή αλλιώς δεν πρέπει να ξεπερνά την διαθέσιμη ποσότητα. Σε τέτοιες περιπτώσεις, εμφανίζεται σχετικό μήνυμα λάθους.

Έστω πως ο εργαζόμενος προσθέτει τα ζητούμενα από τον πελάτη προϊόντα στην παραγγελία. Όταν είναι έτοιμος πατάει το κουμπί **REVIEW ORDER** ώστε να επανεξετάσει την παραγγελία.



Στην αναζήτηση ο εργαζόμενος μπορεί να γράψει τον ειδικό κωδικό “@@” ο οποίος εμφανίζει τα τρέχων επιλεγμένα προϊόντα και τις ποσότητες τους (πρέπει να έχουν επιλεγεί τουλάχιστον μία φορά).



← Order for table: beach1

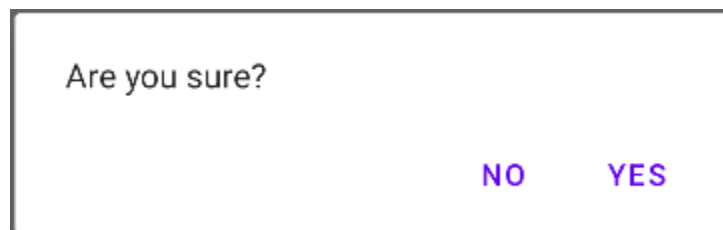
🔍 @@ ➔ ✕

Item: Irish Coffee	
Price: 13	Selected: 2
Quantity: 8	

Item: Pisco Sour	
Price: 9	Selected: 3
Quantity: 7	

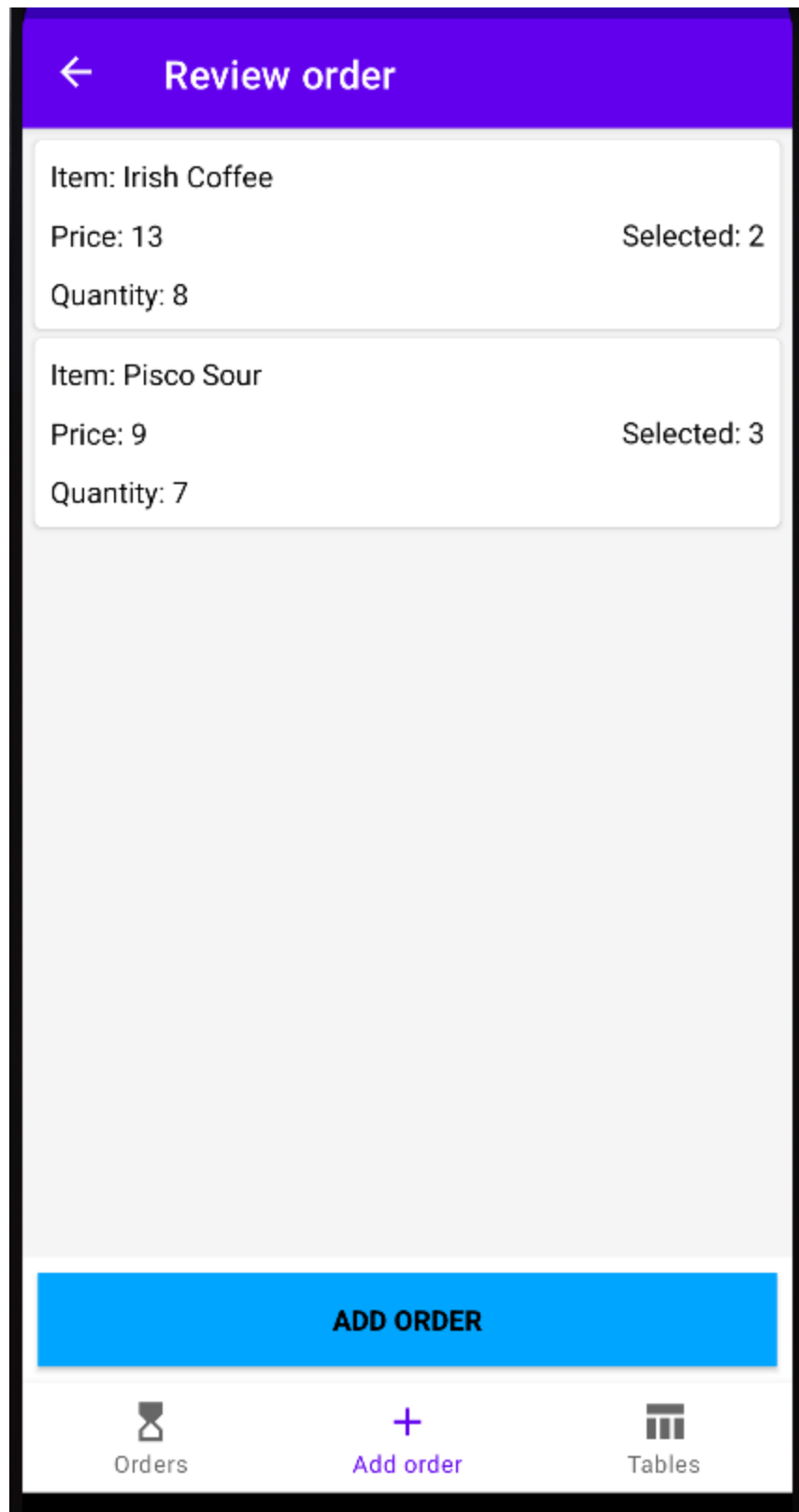
Αν κρίνει πως όλα είναι εντάξει πατάει το κουμπί **ADD ORDER** και η εισαγωγή της παραγγελίας στη βάση δεδομένων ολοκληρώνεται.

Πριν την ολοκληρωθεί η παραγγελία ο εργαζόμενος ρωτάτε από την εφαρμογή αν είναι σίγουρος και μόνο αν πατήσει την επιλογή “yes” συνεχίζεται η διαδικασία, αλλιώς ακυρώνεται.



Are you sure?

NO YES



Σημείωση: πατώντας το βελάκι πάνω αριστερά της οθόνης ανακατευθύνει το χρήστη στην εκάστοτε προηγούμενη οθόνη. Υπενθυμίζεται πως αν ο χρήστης πατήσει το back button του κινητού του θα μεταβεί στην οθόνη επιλογής ρόλων.

Πλέον στις τρέχων παραγγελίες βρίσκεται και η παραγγελία που μόλις δημιουργήθηκε.

Order ID: 9

Table: beach1

ACTIVE

← Items of order: 9

Item: Irish Coffee

Price: 13

Selected: 2

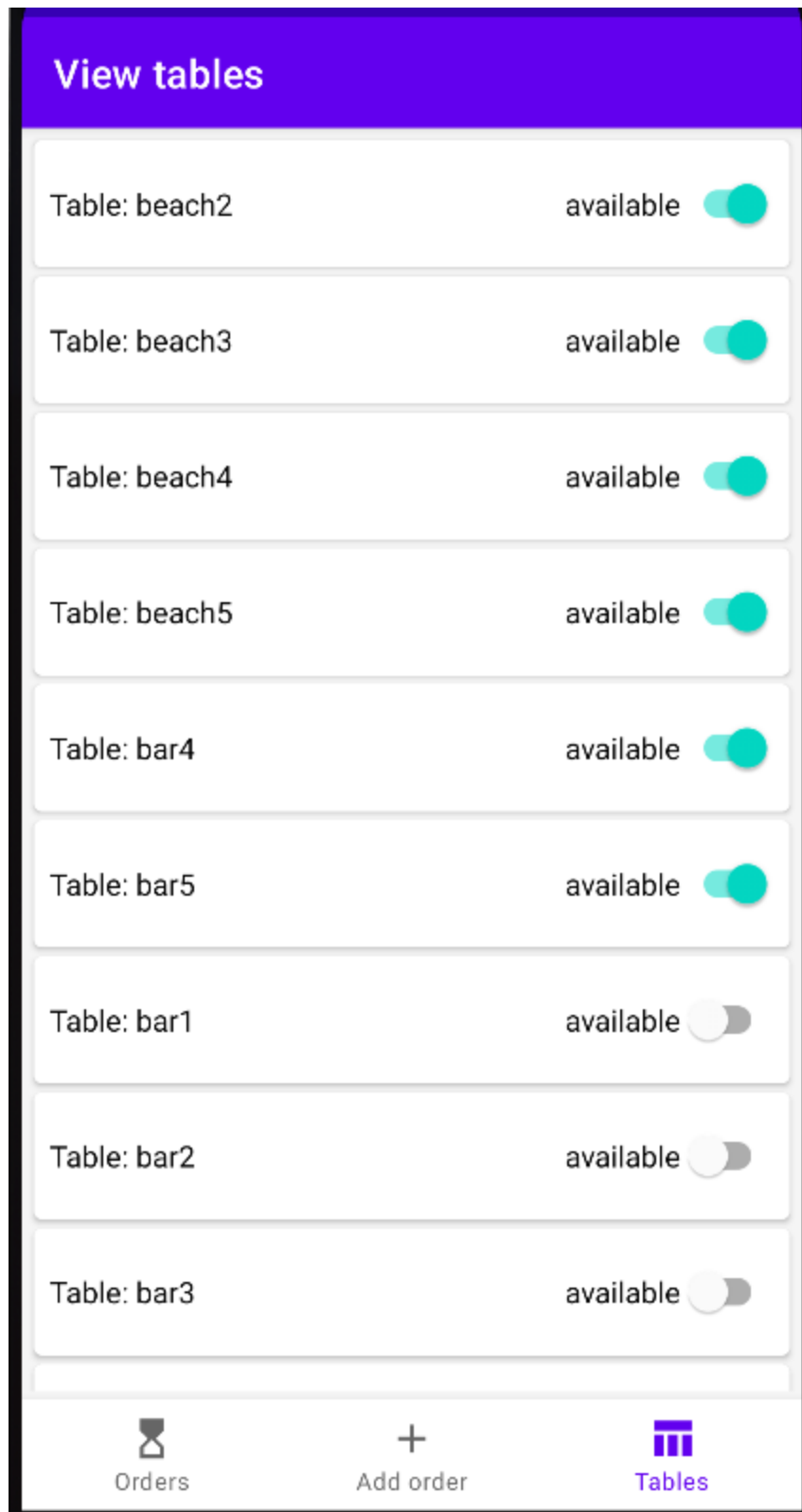
Item: Pisco Sour

Price: 9

Selected: 3

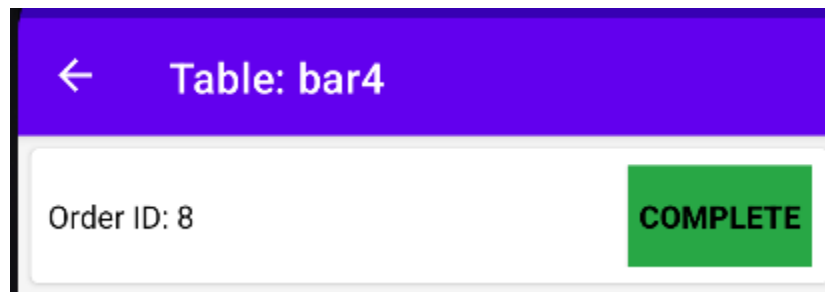
Τελευταία επιλογή του μενού είναι η επιλογή **Tables**. Η οθόνη που εμφανίζεται περιέχει μία λίστα με τα ονόματα των τραπεζιών, καθώς και το ποια τραπέζια είναι διαθέσιμα.

Πατώντας πάνω στο εκάστοτε switch που γράφει **available**, ο εργαζόμενος μπορεί να αλλάξει την διαθεσιμότητα οποιουδήποτε τραπεζιού.



Πατώντας πάνω στο σώμα μιας εγγραφής της λίστας (επιλέγοντας δηλαδή ένα τραπέζι), ο εργαζόμενος μπορεί να δει όλες τις σημερινές (εκάστοτε σημερινής μέρας) παραγγελίες ανεξαρτήτως του status.

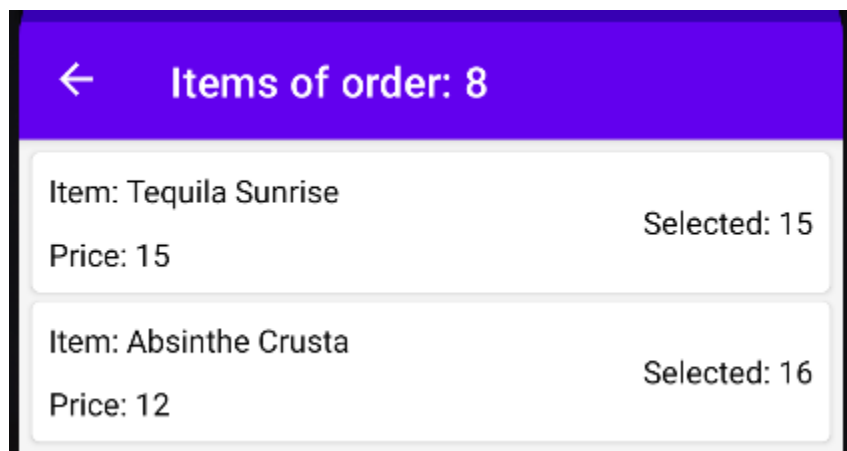
Για παράδειγμα επιλέγοντας το τραπέζι **bar4**, εμφανίζεται η εξής παραγγελία:



Ο εργαζόμενος αν επιθυμεί μπορεί να αλλάξει το status της παραγγελίας (μπορεί για παράδειγμα εξαρχής να το άλλαξε κατά λάθος και να θέλει να επαναφέρει την παραγγελία στο σωστό status).

Όπως ακριβώς και στις τρέχων παραγγελίες, ο εργαζόμενος έχει τη δυνατότητα πατώντας πάνω της να δει τα αντικείμενα του μενού που συσχετίζονται με την εξής παραγγελία.

Για παράδειγμα στην παραγγελία με ID το νούμερο 8:



Σημείωση: πατώντας το βελάκι πάνω αριστερά της οθόνης ανακατευθύνει το χρήστη στην εκάστοτε προηγούμενη οθόνη. Υπενθυμίζεται πως αν ο χρήστης πατήσει το back button του κινητού του θα μεταβεί στην οθόνη επιλογής ρόλων.

Εδώ ολοκληρώνεται η παρουσίαση των δράσεων του ρόλου **employee**.

Αρχιτεκτονική συστήματος

Ακολουθεί η τεχνική επεξήγηση της πτυχιακής. Για ευκολία ανάγνωσης, η συγκεκριμένη ενότητα χωρίζεται σε υποενότητες που αντιστοιχούν στις 3 εφαρμογές από τις οποίες απαρτίζεται η πτυχιακή.

Web εφαρμογή δημιουργίας μενού

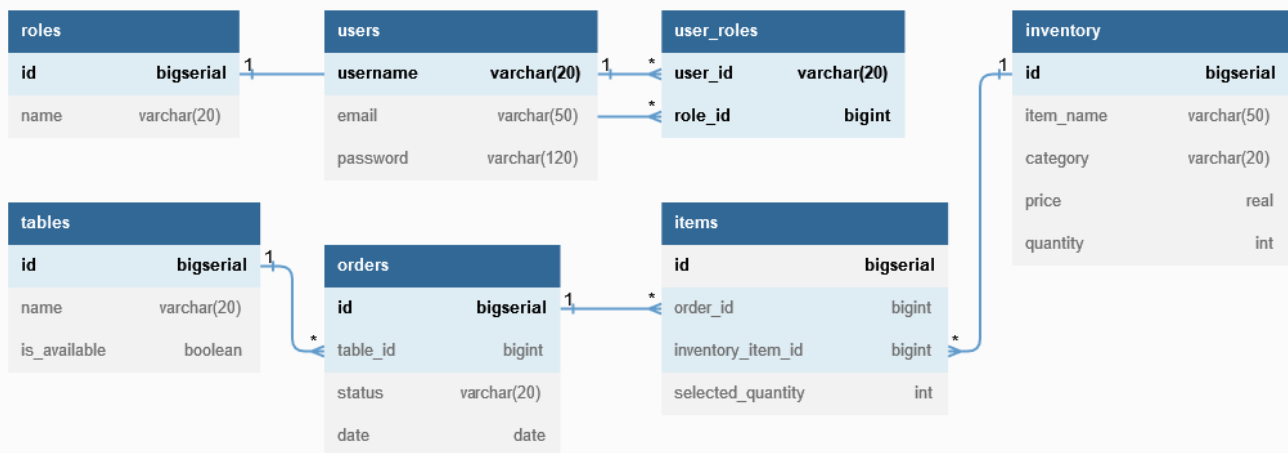
Η εφαρμογή υλοποιήθηκε σε γλώσσα Java σε περιβάλλον IntelliJ IDEA. Για τη μεταφορά πληροφοριών μεταξύ των σελίδων HTML και των controllers (και το αντίστροφο) χρησιμοποιείται το template engine Thymeleaf μέσω του αντίστοιχου dependency. Με τη χρήση τοπικών μεταβλητών αποθηκεύονται οι επιλογές του χρήστη σχετικά με το μενού και στο τέλος αποθηκεύονται σε μορφή αρχείου json.

Για περισσότερες πληροφορίες σχετικά με την εφαρμογή υπάρχουν πυκνά σχόλια στον κώδικα.

Back-end API

Η εφαρμογή υλοποιήθηκε σε γλώσσα Java σε περιβάλλον IntelliJ IDEA. Αποτελεί το back-end της πτυχιακής.

Η βάση δεδομένων με την οποία αλληλοεπιδρά το API είναι σε PostgreSQL, η οποία δημιουργείτε εσωτερικά της εφαρμογής μέσω οντοτήτων χρησιμοποιώντας το dependency Spring Data JPA.



Κάθε έγκυρο HTTP αίτημα που δέχεται το API λαμβάνεται από τον κατάλληλο controller, ο οποίος καλεί τις αντίστοιχες μεθόδους (service), που με τη σειρά τους αλληλοεπιδρούν με τη βάση δεδομένων. Τελικά, παράγεται ένα HTTP response το οποίο στέλνεται πίσω στον client.

Όσον αφορά το security της εφαρμογής, το authentication και το authorization γίνεται μέσω JWTs. Ο client χρειάζεται να στείλει τα credentials του μόνο μία φορά στο API το οποίο, αν είναι έγκυρα, του επιστρέφει ένα json web token (JWT). Το token είναι έγκυρο για συγκεκριμένο χρονικό διάστημα (στην παρούσα πτυχιακή είναι έγκυρο για 12 ώρες). Μετά το πέρας του χρονικού ορίου ο client πρέπει να ξανακάνει αίτημα για νέο JWT (δηλαδή να ξανασυνδεθεί στην εφαρμογή). Όσο ο client έχει στη κατοχή του ένα έγκυρο JWT, δεν χρειάζεται να ξαναστείλει τα στοιχεία του στο API. Σε κάθε αίτημα προς τον server στέλνεται μέσω header και το JWT. Το token περιέχει κωδικοποιημένα τα στοιχεία του client και μέσω αυτού γίνεται και το authorization (μιας και το token στέλνεται σε κάθε HTTP αίτημα). Για την υλοποίηση του security χρησιμοποιήθηκε το Spring Security dependency.

Μέσα στο φάκελο του project, στο αρχείο **logs.txt** αποθηκεύονται τα logs της εφαρμογής σε πραγματικό χρόνο (όπως για παράδειγμα μηνύματα **INFO**, **WARN** ή **ERROR**)

Για περισσότερες πληροφορίες σχετικά με την εφαρμογή υπάρχουν πυκνά σχόλια στον κώδικα.

Android εφαρμογή

Η εφαρμογή υλοποιήθηκε σε γλώσσα Java σε περιβάλλον Android Studio. Αποτελεί το user interface της πτυχιακής. Επικοινωνεί με το API μέσω HTTP αιτημάτων. Τα αιτήματα στέλνονται μέσω Retrofit REST client. Για την πλοήγηση στην εφαρμογή έχει γίνει χρήση του navigation component σε συνδυασμό με bottom navigation menus. Η εφαρμογή αποτελείται κυρίως από fragments (13 fragments) και μόλις 4 activities. Για την παρουσίαση των δεδομένων χρησιμοποιούνται κυρίως recycle views μερικά από τα οποία ενημερώνονται χρησιμοποιώντας DiffUtil. Επίσης γίνεται χρήση διαλόγων όπως τα DialogInterface και BottomSheetDialog.

Για περισσότερες πληροφορίες σχετικά με την εφαρμογή υπάρχουν πυκνά σχόλια στον κώδικα.

Συμπεράσματα και μελλοντικές επεκτάσεις

Το παρόν project είναι μια προσέγγιση των σύγχρονων λογισμικών παραγγελιοληψίας. Υποστηρίζει τη λήψη, τροποποίηση και παρακολούθηση των παραγγελιών για κάθε διαφορετικό τραπέζι της επιχείρησης σε πραγματικό χρόνο. Επιπλέον, δίνει τη δυνατότητα για άμεση τροποποίηση και γενική διαχείριση του τιμοκαταλόγου και της αποθήκης προϊόντων της επιχείρησης. Πέρα των βασικών λειτουργιών, η εφαρμογή παράγει και στατιστικά για τον τζίρο του μαγαζιού.

Σίγουρα υπάρχουν όμως και περιθώρια βελτίωσης. Μερικές μελλοντικές επεκτάσεις θα μπορούσαν να είναι οι εξής:

- Deploy του API στο internet (είτε σε προσωπικό server, είτε στο cloud) ώστε να είναι δυνατή η χρήση του PDA όχι μόνο σε τοπικό επίπεδο.
- Δυνατότητα υποστήριξης πολλαπλών καταστημάτων.
- Βελτίωση του τρόπου λήψης παραγγελιών ώστε ανεξάρτητα από το πόσο παράξενη είναι η παραγγελία του πελάτη να μπορεί να καταγραφεί.
- Δυνατότητα εξαγωγής και αποθήκευσης στατιστικών στοιχείων και δεδομένων.

Βιβλιογραφία

- Amigoscode, Spring Boot Tutorial:

<https://www.youtube.com/watch?v=9SGDpanrc8U>

- Amigoscode, Spring Security:

<https://github.com/amigoscode/spring-boot-security-course>

https://www.youtube.com/watch?v=her_7pa0vrg

<https://docs.spring.io/spring-security/reference/>

- GetArrays, Spring Boot and Spring Security with JWT including Access and Refresh Tokens:

<https://www.youtube.com/watch?v=VVn9OG9nfH0>

- Coding in Flow, RecyclerView + CardView – Android Studio Tutorial:

<https://www.youtube.com/playlist?list=PLrnPJCHvNZuBtTYUuc5Pyo4V7xZ2HNtf4>

- Coding in Flow, Date & Time – Android Programming:

<https://www.youtube.com/playlist?list=PLrnPJCHvNZuAWIa1JCho4PGgNTNK0dfJ>

- Coding in Flow, Retrofit Tutorial for Beginners – Android Programming:
<https://www.youtube.com/playlist?list=PLrnPJCHvNZuCbUD3xpfKzQWOj3AXybSaM>
- Coding in Flow, Bottom Sheets – Android Programming:
https://www.youtube.com/playlist?list=PLrnPJCHvNZuCH99R_1qBLtsNGZJag1xy5
- Coding in Flow, BottomNavigationView - Getting Started With Navigation Component:
<https://www.youtube.com/watch?v=llWsm9Pjkpc>
- Coding in Flow, How to Filter a RecyclerView with SearchView - Android Studio Tutorial:
<https://www.youtube.com/watch?v=sJ-Z9G0SDhc>
- PRABEESH R K, Smart way to update RecyclerView using Diffutil:
<https://www.youtube.com/watch?v=i1AjlGywGFU>
- Stack Overflow, How to display Yes/No dialog box on Android:
<https://stackoverflow.com/questions/2478517/how-to-display-a-yes-no-dialog-box-on-android>
- Stack Overflow, Cleartext HTTP traffic not permitted:
<https://stackoverflow.com/questions/45940861/android-8-cleartext-http-traffic-not-permitted>
- Mayank, How to put android layout files in subfolders:
<https://medium.com/mindorks/how-to-put-android-layout-files-in-subfolders-1f7cf07ff48f>
- Ben O'Brien, SharedPreferences - How to Save & Retrieve Data Android Studio:
<https://www.youtube.com/watch?v=jiD2fxn8iKA>
- CodeWithMazn, Navigation Component in Android Studio:
https://www.youtube.com/watch?v=Qe_6OmmFaR8
- yoursTRULY, Multi view RecyclerView | getItemViewType() | Android:
<https://www.youtube.com/watch?v=VtnLpHUu2U0>
- Android Login / Registration Design:
<https://appsnipp.com/free-login-registration-xml-design-for-android/>
- Penguin Coders, How to Change App Icon in Android Studio:
<https://www.youtube.com/watch?v=DQ8HDL-X9rM>
- Δημιουργία icon για την android εφαρμογή: <https://editor.freelogodesign.org/>