D F M

ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΔΗΛΩΣΕΩΝ ΦΟΡΟΥ ΜΕΤΑΒΙΒΑΣΗΣ



Συνεργατική Υλοποίηση

Θεοχάρης Γεώργιος (it21623)

Καραμάλης Γεώργιος (it21630)

Τάχος Αθανάσιος (it21692)

Αθήνα, 20/02/2023

# **Εισαγωγή**

Το DFM είναι ένα πληροφοριακό σύστημα διαχείρισης δηλώσεων φόρου μεταβίβασης. Δημιουργήθηκε με γνώμονα την απλότητα και την ευχρηστία χωρίς να γίνουν «εκπτώσεις» σε θέματα ασφάλειας και αποτελεσματικότητας.

**Περιεχόμενα**

[**Εισαγωγή** 1](#_Toc127824187)

[**Χρήστες και Ρόλοι** 2](#_Toc127824188)

[**Σενάριο Εργασίας** 3](#_Toc127824189)

[**Υλοποίηση** 5](#_Toc127824190)

[**Εγχειρίδιο Χρήσης** 9](#_Toc127824191)

# **Χρήστες και Ρόλοι**

Το σύστημα είναι προσβάσιμο μόνο σε πιστοποιημένους χρήστες καθώς οι πληροφορίες που διαχειρίζεται είναι ευαίσθητες. Για το λόγο αυτό προσδιορίστηκαν οι παρακάτω ρόλοι:

* Διαχειριστής (Administrator)
* Συμβολαιογράφος (Notary)
* Πωλητής (Seller)
* Αγοραστής (Buyer)

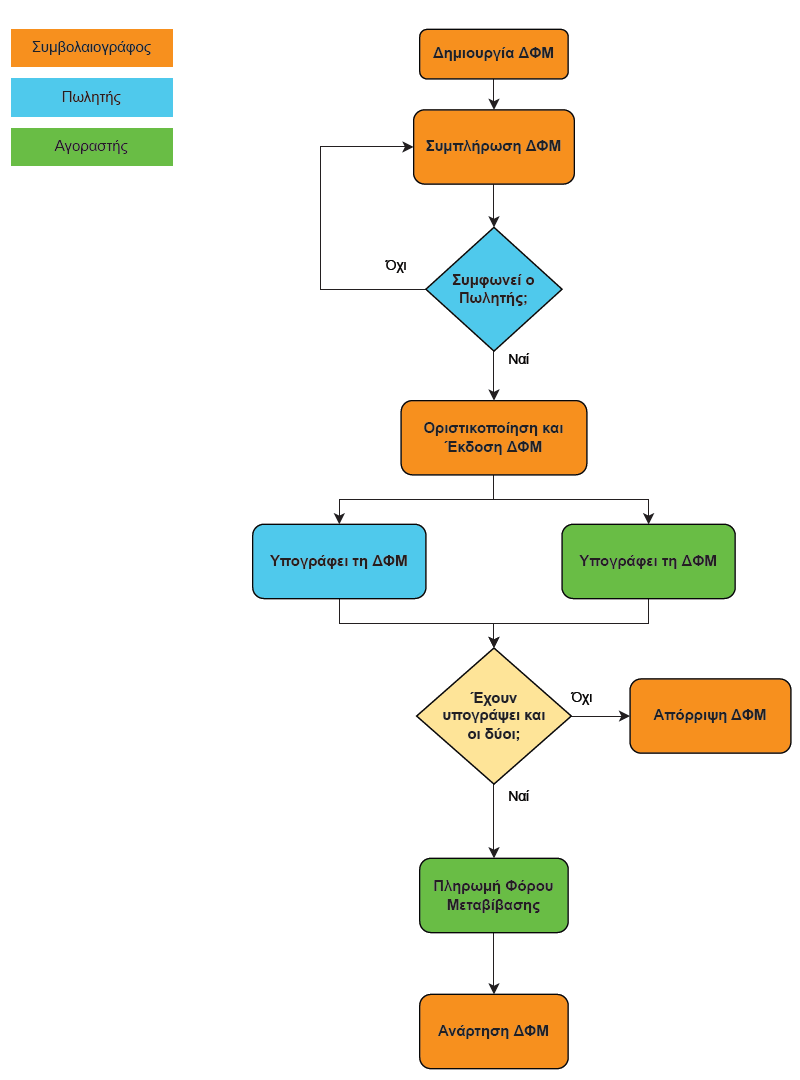
Κάθε ρόλος φέρει και διαφορετικά δικαιώματα που βάση αυτών δημιουργούνται λειτουργικές περιοχές πρόσβασης. Συγκεκριμένα τα δικαιώματα ανά ρόλο:

* Διαχειριστής
  + Μπορεί να δει και να επεξεργαστεί οποιαδήποτε πληροφορία του συστήματος.
* Συμβολαιογράφος
  + Μπορεί να δει τις πληροφορίες των πωλητών, των αγοραστών και των ακινήτων χωρίς να μπορεί να τις επεξεργαστεί.
  + Μπορεί να δει και να επεξεργαστεί τις πληροφορίες των δηλώσεων
* Πωλητής
  + Μπορεί να δει τις πληροφορίες των συμβολαιογράφων, των αγοραστών και των δηλώσεων που συσχετίζονται με αυτόν.
  + Μπορεί να δει και να επεξεργαστεί τις πληροφορίες των ακινήτων.
* Αγοραστής
  + Μπορεί να δει τις πληροφορίες των συμβολαιογράφων, των πωλητών, των ακινήτων και των δηλώσεων που συσχετίζονται με αυτόν.

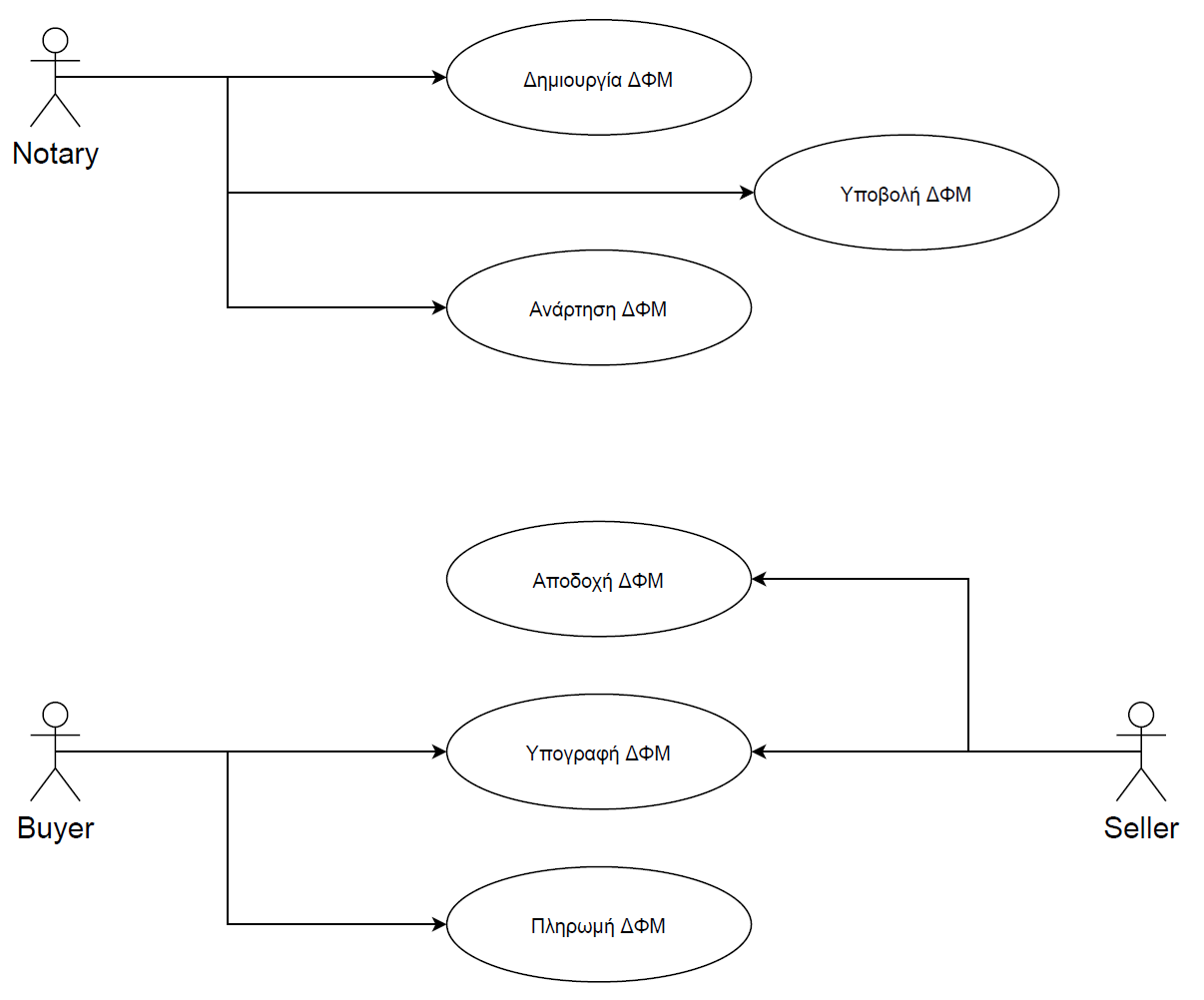
Κατά την πρώτη είσοδο του χρήστη στο σύστημα προσδιορίζεται και ο ρόλος που επιθυμεί να έχει, μεταξύ των επιλογών Buyer (Αγοραστής), Seller(Πωλητής) και Notary (Συμβολαιογράφος). Δεν μπορεί να γίνει επιλογή ρόλου Διαχειριστή.

# **Σενάριο Εργασίας**

Το βασικό σενάριο εργασίας είναι η σύναψη της δήλωσης φόρου μεταβίβασης. Για την διεκπεραίωση του εν λόγο σεναρίου αναπτύχθηκε η παρακάτω ροή εργασίας:



Οι ενέργειες που πρέπει να ακολουθήσει ο κάθε χρήστης για την διεκπεραίωση του πρωθύστερου σεναρίου αναλύονται στο παρακάτω διάγραμμα:



# **Υλοποίηση**

Για την ανάπτυξη του DFM χρησιμοποιήθηκε Java έκδοση 19.0.1 κάνοντας χρήση του Spring Boot Framework με τα παρακάτω εργαλείων:

* postgressql

Βιβλιοθήκη πρόσβασης στη Βάση Δεδομένων

* spring-boot-starter-data-jpa

Μέσο επικοινωνίας με τη Βάση Δεδομένων χρήση του Hibernate ORM

* spring-boot-starter-security

Μηχανισμός διαχείρισης χρηστών και ρόλων

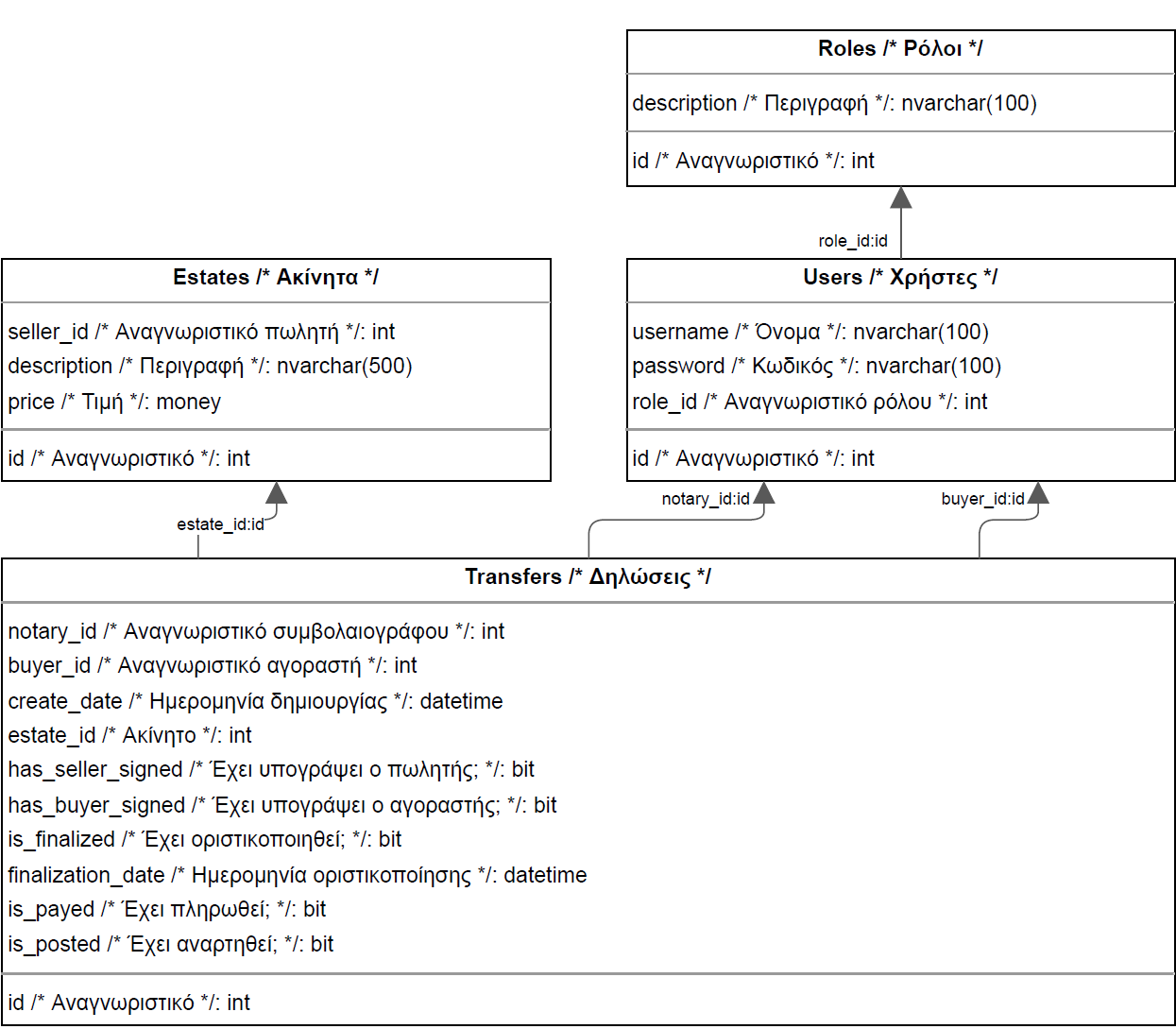
* spring-boot-starter-web

Τρόπος δημιουργίας διαδικτυακής εφαρμογής

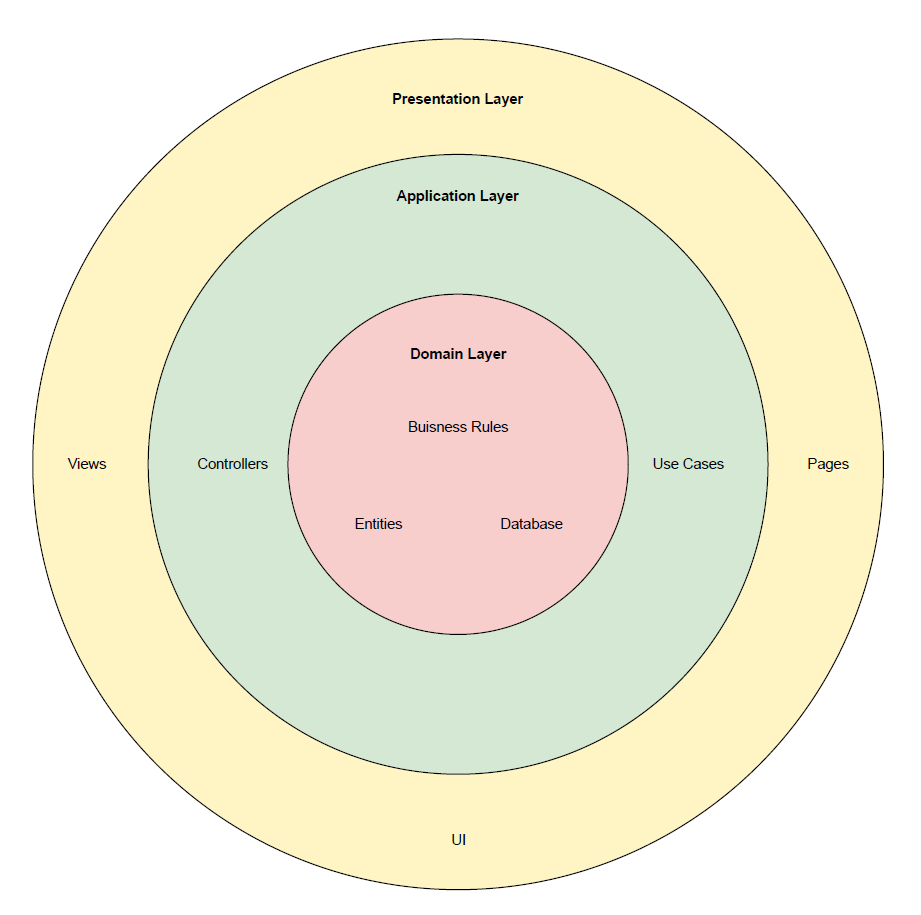
* modelmapper

Βιβλιοθήκη αντιστοίχισης μοντέλων βάσης με μοντέλα από τα request

Η Βάση Δεδομένων που επιλέχθηκε είναι η postgresSQL. Η υλοποίηση έγινε κάνοντας χρήση ενός σχήματος (scheme) με τους απαραίτητους πίνακες για τη διατήρηση των δεδομένων. Ενδεικτικά παρατίθεται το σχήμα:



Η αρχιτεκτονική που στήθηκε η εφαρμογή είναι βασισμένη σε επίπεδα (layers) και λέγεται Clean Architecture. Το κάθε επίπεδο έχει διακριτές δυνατότητες και πρόσβαση σε λειτουργίες.



Πέρα από την Clean Architecture χρησιμοποιήθηκε και το Model View Controller Design Pattern.

Με βάση τα παραπάνω ο κώδικας που αφορά το κάθε επίπεδο διαχωρίζεται σε:

* Presentation Layer
  + Διαθέτει τον κώδικα που τρέχει στον πελάτη (client) δηλαδή στο πρόγραμμα περιήγησης (browser). Πιο συγκεκριμένα είναι το User Interface που παρέχεται μέσα από τις Pages που φορτώνουν Views για την εμφάνιση αυτού.
* Application Layer
  + Έρχεται σε επαφή με το Presentation Layer μέσω των Endpoints που ορίζονται στους Controllers. Οι Controllers «ακούν» σε request που γίνονται από τον client και επιστρέφουν πίσω τα απαραίτητα δεδομένα που χρειάζονται τα Views για να «ζωγραφίσουν» το Interface.
  + Επίσης, έρχεται σε επαφή και με το Domain Layer για την άντληση των δεδομένων μέσω των Repositories που καλούν τα Services. Τα Services έχουν τα Use Cases.
* Domain Layer
  + Έχει την σύνδεση με τη Βάση Δεδομένων μέσω των Repositories, τα οποία έχουν όλες τις απαραίτητες μεθόδους για την άντληση και επεξεργασία των δεδομένων. Επίσης, φέρουν και όλα τα Business Rules που πρέπει το σύστημα και τα δεδομένα να «υπακούν» προστατεύοντας με αυτόν τον τρόπο την ακεραιότητα και την κανονικότητα των δεδομένων.
  + Για την επίτευξη των παραπάνω υπάρχουν τα μοντέλα (models) των πινάκων της Βάσης Δεδομένων που τα Repositories φορτώνουν μέσα τους τα δεδομένα.

# **Εγχειρίδιο Χρήσης**

Η δομή των αρχείων έχει ως εξής:

1. Application: Backend springboot Web API
2. Database: Αρχεία Βάσης Δεδομένων
3. Documentation Αρχεία Ανάλυσης
4. Frontend Frontend, Blazor Web Assembly
5. Pgadmin Αρχεία Εργαλείου Δχσης Βάσης Δεδομένων

Για την εγκατάσταση της εφαρμογής, απαραίτητη προϋπόθεση είναι η ύπαρξη των παρακάτω:

* WSL
* Docker Desktop
* JDK
* Visual studio code με τα παρακάτω extensions
  + C#
  + Extension Pack for Java
  + IntelliCode
  + Springboot Extension Pack

Κατόπιν αυτών, η πρώτη επέμβαση πρέπει να γίνει στο αρχείο hosts στον φάκελο C:\Windows\System32\drivers\etc και να προστεθεί η παρακάτω εγγραφή:

127.0.0.1 dfm.com

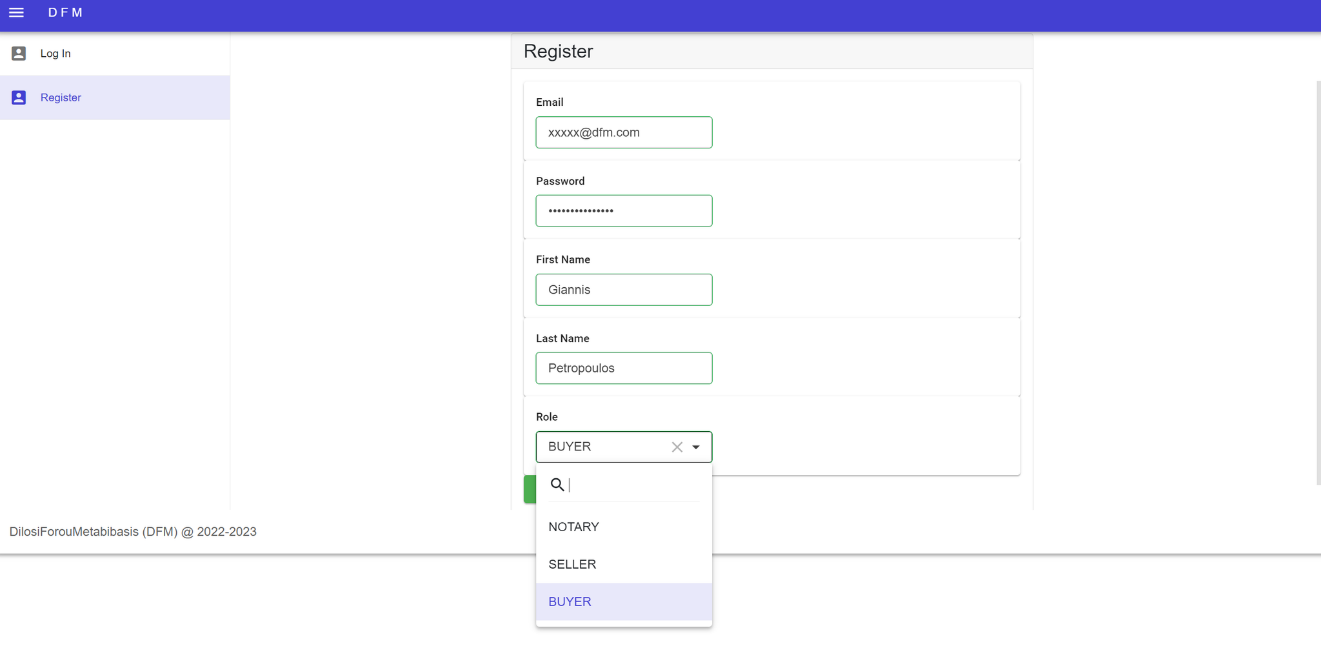
Επόμενο βήμα είναι να ανοίξουμε ένα τερματικό και να πλοηγηθούμε στον root folder του project μας. Εκεί, θα εκτελέσουμε την εντολή docker-compose up –d . Με αυτή την εντολή θα γίνει η αρχικοποίηση της βάσης δεδομένων, καθώς και η εκκίνηση του pgadmin που μας δίνει την δυνατότητα μέσω της διεύθυνσης http://dfm.com:5050 να πλοηγηθούμε σε εργαλείο διαχείρισης της βάσης δεδομένων.

Επόμενο βήμα θα είναι να ανοίξουμε με το VS Code τον φάκελο Application, καθώς και σε ένα δεύτερο παράθυρο τον φάκελο FrontEnd. Τρέχουμε και τις δύο εφαρμογές. Στη συνέχεια μπορούμε να πλοηγηθούμε στην εφαρμογή μας ακολουθώντας τον παρακάτω σύνδεσμο http://dfm.com:51337

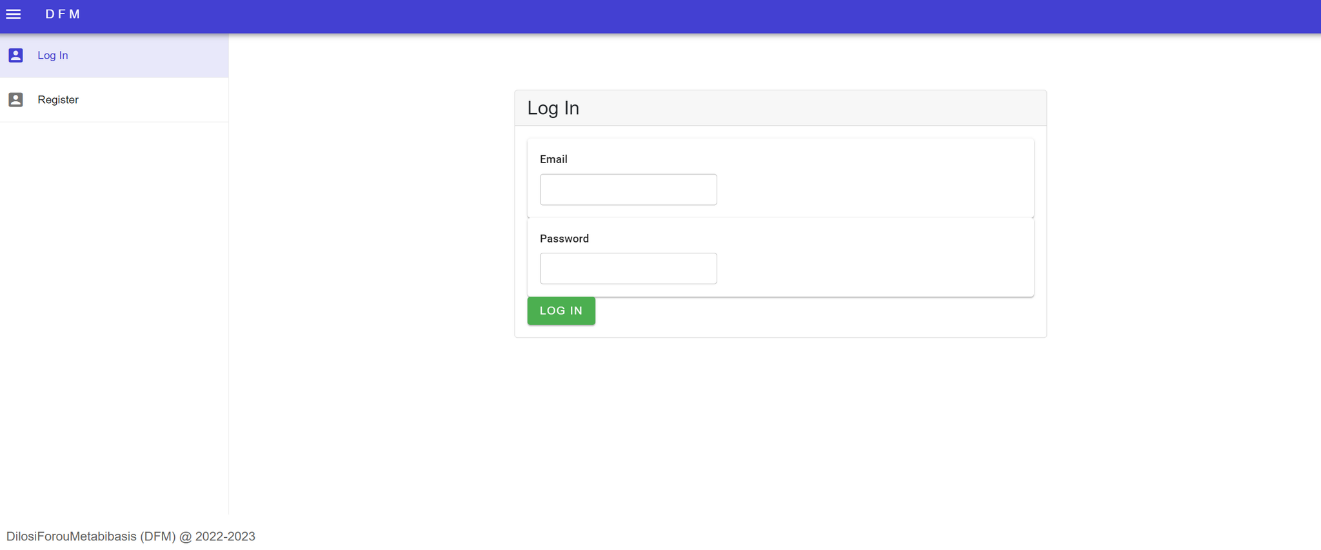
Κατά την πρώτη είσοδο ενός unregistered χρήστη, η οθόνη υποδοχής έχει όπως στην παρακάτω εικόνα:



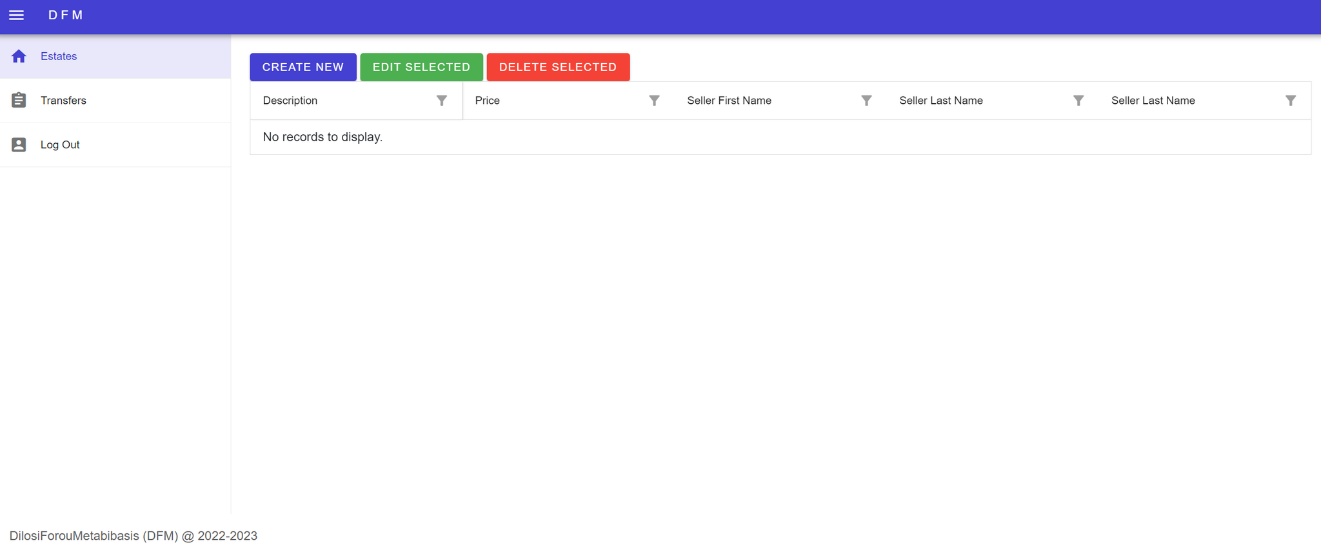
Εκεί, παρέχονται οι δυνατότητες Login ή Register για τον χρήστη. Με την επιλογή του Register, ο νέος χρήστης συμπληρώνει τα πεδία Email, Password, First Name, Last Name και τέλος επιλέγει τον ρόλο-ιδιότητα του από το drop-down menu , όπως φαίνεται στην παρακάτω εικόνα.



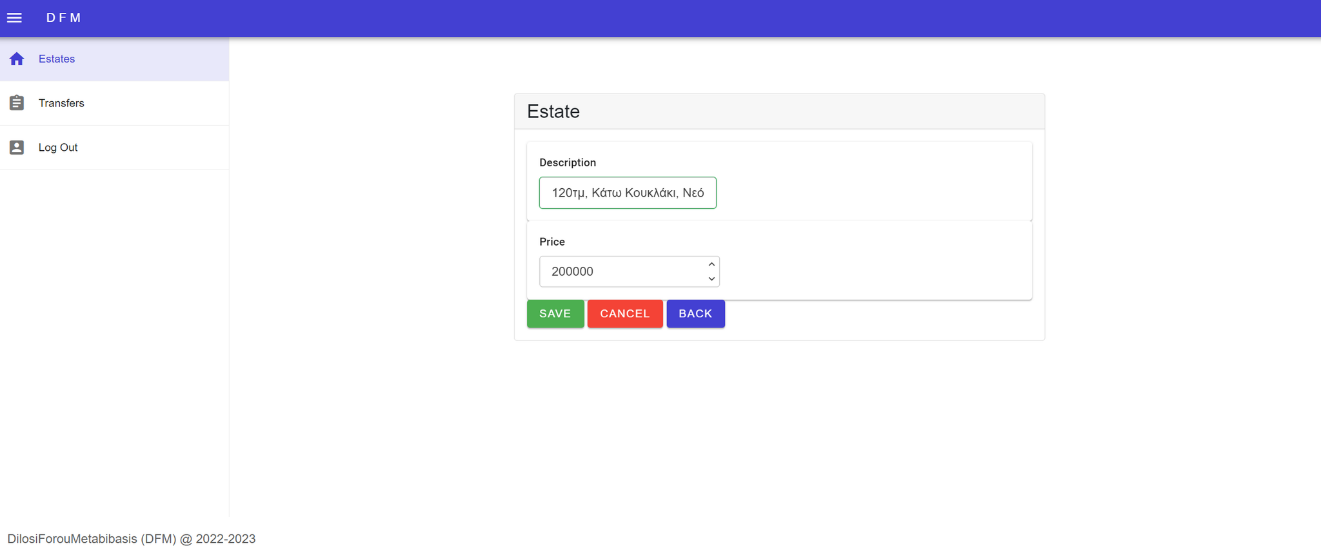
Αν ο χρήστης έχει κάνει ήδη register σε προηγούμενο χρόνο, τότε με την επιλογή Log In θα μας ανακατευθύνει στην επόμενη οθόνη όπου με τα credential μας (email και password) θα συνδεθούμε στην εφαρμογή.



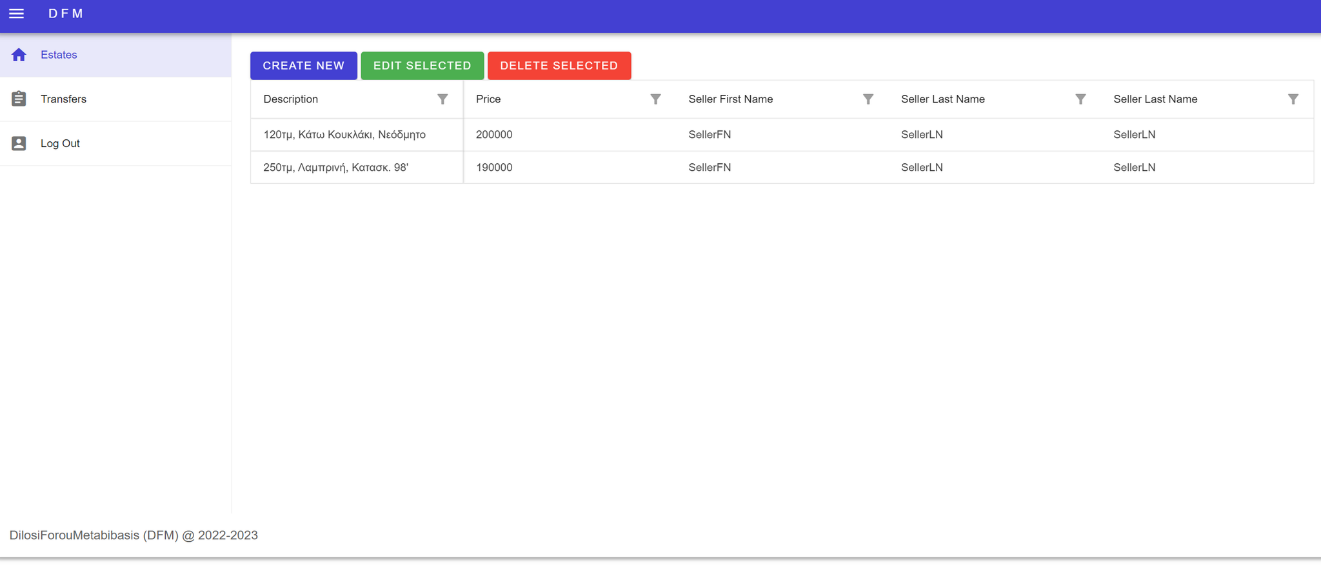
Με την είσοδο στην εφαρμογή, εμφανίζεται η παρακάτω οθόνη



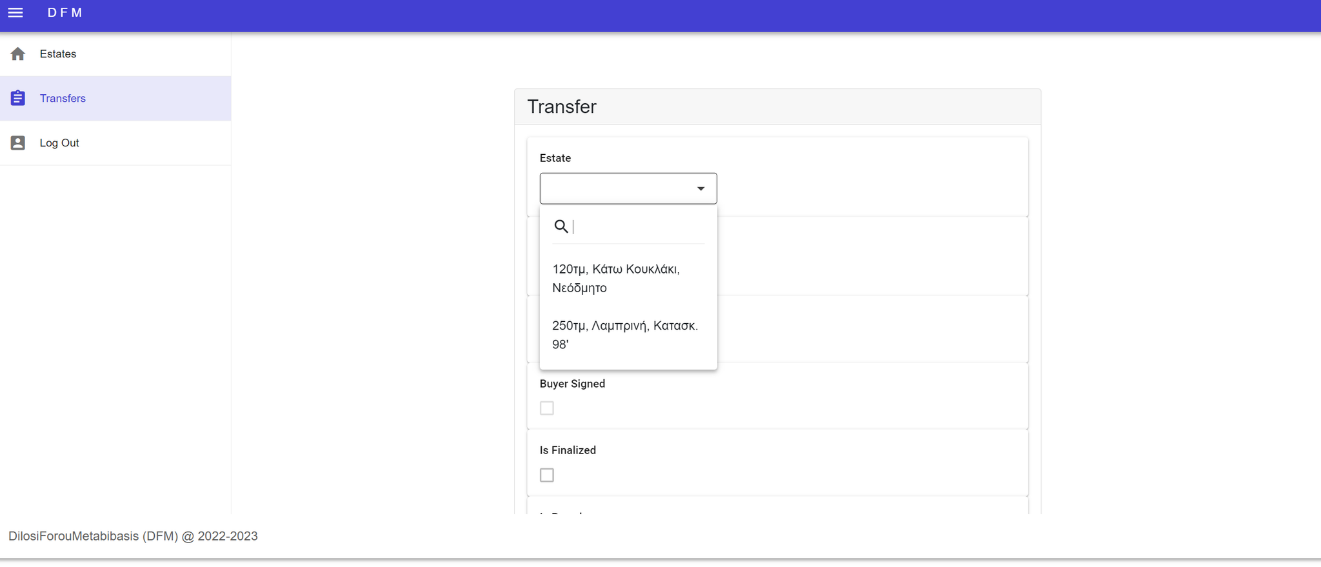
Επιλέγοντας Create New, ένας Seller δύναται να προσθέσει ακίνητα στη λίστα, όπως παρακάτω:



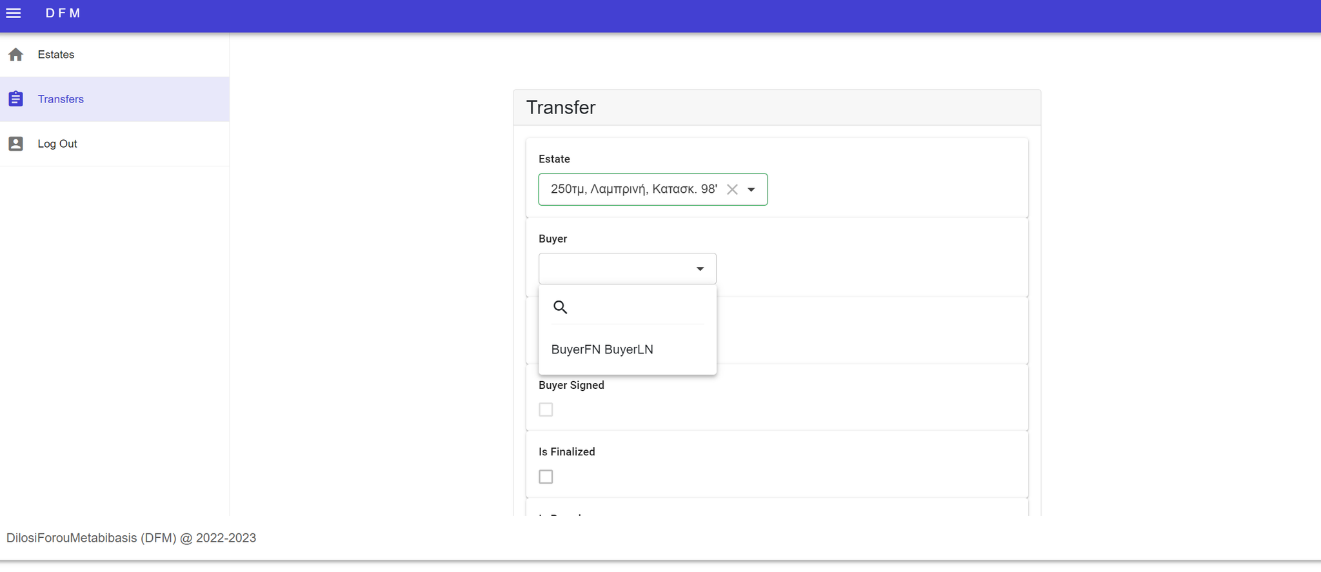
Η λίστα των προς δήλωση ακινήτων έχει την παρακάτω μορφή:

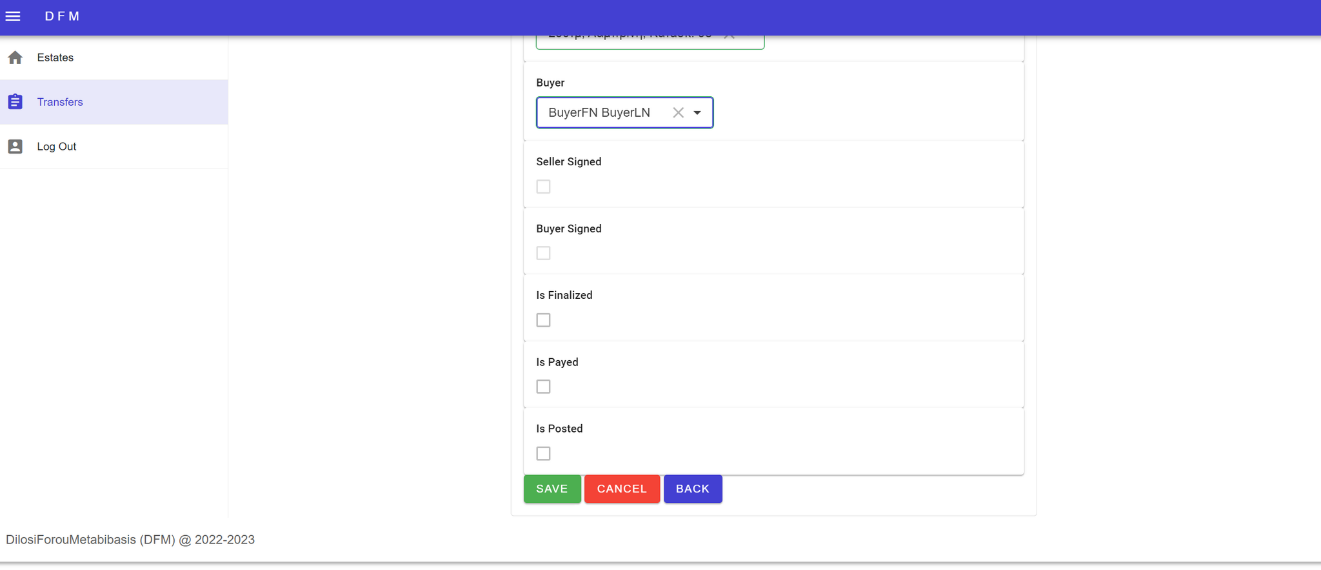


Κατ’αντιστοιχία, με την είσοδο ενός Συμβολαιογράφου στην εφαρμογή, αυτός μπορεί πλέον να μεταβεί στο πεδίο Transfers, όπου και θα μπορεί να αναζητήσει τα διαθέσιμα ακίνητα που θα έχουν καταχωρήσει οι διάφοροι πωλητές(παρουσιάζονται με τη μορφή drop-down menu). Η όψη της σελίδας των Transfers θα έχει την παρακάτω μορφή:

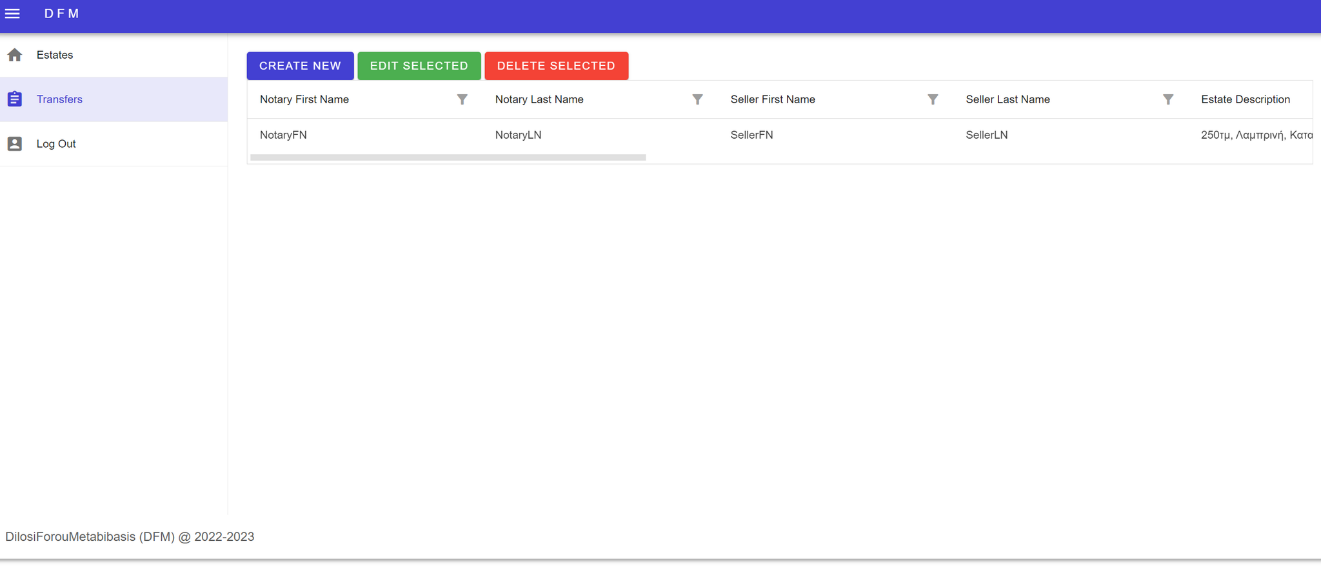
****

Αφού επιλεχθεί το ακίνητο που μας ενδιαφέρει, επιλέγεται ο αγοραστής, από την λίστα των διαθέσιμων αγοραστών. (παρουσιάζονται με τη μορφή drop-down menu)

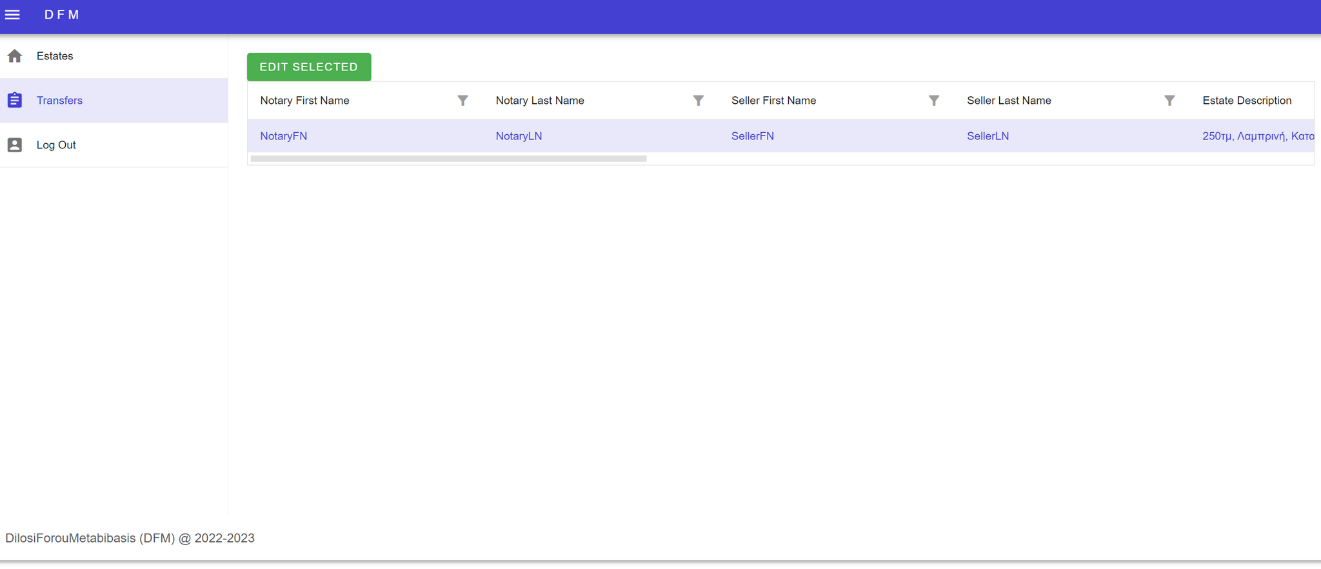


Έχοντας ολοκληρώσει αυτά τα βήματα, πλέον ο συμβολαιογράφος μπορεί να επιλέξει μεταξύ των επιλογών Save, Cancel και Back.

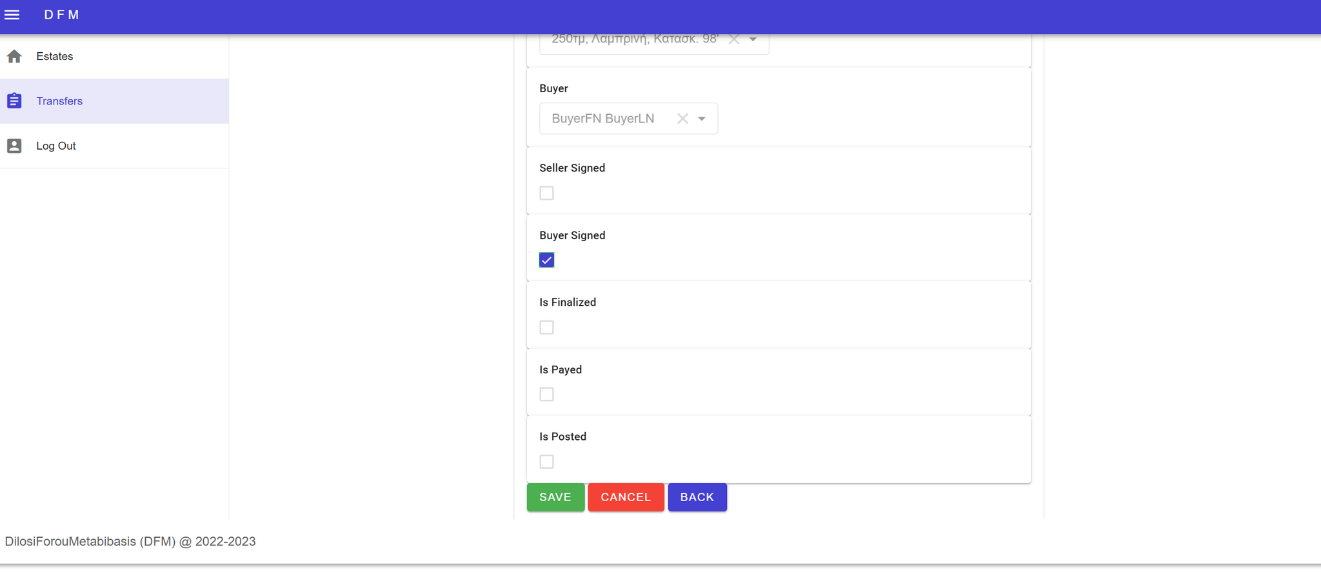
Με το που ολοκληρωθεί η παραπάνω διαδικασία, η μεταβίβαση του ακινήτου εμφανίζεται στην παρακάτω λίστα:

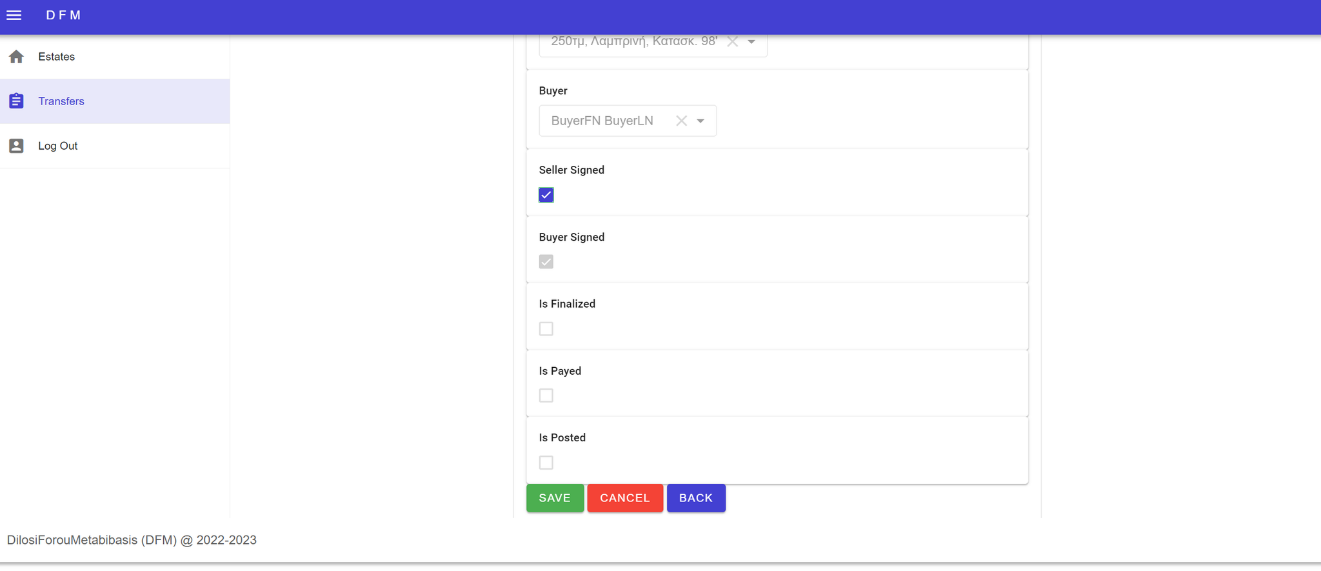


Ένας χρήστης με την ιδιότητα-ρόλο του Αγοραστή(Buyer), μπάινοντας στην εφαρμογή και επιλέγοντας Transfers, βλέπει τις δηλώσεις που τον αφορούν και θα πρέπει να πατήσει το Edit για αυτή που επιθυμεί.



Εκτελώντας την παραπάνω ενέργεια, εισέρχεται στο μενού επιλογών της παρακάτω εικόνας, όπου και θα συμπληρώσει το πεδίο Buyer Signed. Πλέον ο Αγοραστής έχει «εγκρίνει» την μεταβίβαση του ακινήτου.



Τα ίδια βήματα ακολουθεί και ο Πωλητής, και συνυπογράφει την μεταβίβαση, συμπληρώνοντας το αντίστοιχο πεδίο που του αναλογεί.  


Το επόμενο βήμα για την ολοκλήρωση της διαδικασίας μεταβίβασης, περιλαμβάνει τον συμβολαιογράφο, ο οποίος και θα μεταβεί στα Transfers, θα εντοπίσει την δήλωση και θα προχωρήσει στην συμπλήρωση των πεδίων i) Is Finalized ii) Is Payed και iii) Is Posted όταν και μόνο όταν πραγματοποιηθούν αυτές οι ενέργειες μεταξύ των συναλλασόμενων, διαδοχικά.