

Εργασία μαθήματος
«ΚΑΤΑΝΕΜΗΜΕΝΑ ΣΥΣΤΗΜΑΤΑ»

Ομάδα **72**

ΠΟΥΛΗΜΕΝΟΣ ΓΕΩΡΓΙΟΣ , it2021123

ΜΑΖΙΤΖΗΣ ΚΩΝΣΤΑΝΤΙΝΟΣ , it22065

ΠΕΤΡΟΥΛΙΑΣ ΠΑΝΑΓΙΩΤΗΣ , it2021083

Δεκέμβριος 2023



Πίνακας Περιεχομένων

Πίνακας Περιεχομένων	2
Εισαγωγή	3
Αρχιτεκτονική	5
Use Case Διάγραμμα	6
Sequence Διαγράμματα	8
Class Διάγραμμα	15
Component Διάγραμμα	17
Use Case Περιγραφές	18



Εισαγωγή

Στόχος της εργασίας: Η δημιουργία των προδιαγραφών και των μοντέλο-διαγραμμάτων για ένα σύστημα δημιουργίας, έγκρισης/απόρριψης αιτήσεων δημιουργίας αγροτικών συνεταιρισμών. Το σύστημα έχει σκοπό να διευκολύνει τη διαδικασία δημιουργίας αγροτικών συνεταιρισμών.

Παραδοχές

Αν εγκριθεί κάποια αίτηση τότε ο συνεταιρισμός που θα δημιουργηθεί αποθηκεύεται σε κάποια άλλη εφαρμογή και εφόσον στην εκφώνηση δεν αναφέρεται τίποτα δεν μας αφορά.

Ο διαχειριστής δεν είναι χρήστης του συστήματος(απο εκφώνηση).

Ο διαχειριστής έχει άδεια για όλες τις λειτουργίες(και των δικών του και των χρηστών).

Ο διαχειριστής γνωρίζει πότε χρειάζεται να επέμβει στο σύστημα(για να δημιουργήσει, να αλλάξει ή να διαγράψει χρήστες) με τρόπο ανεξάρτητο και αδιάφορο του συστήματος.

Η ανάθεση των ρόλων γίνεται μαζί με τη δημιουργία νέου χρήστη από τον διαχειριστή.

Ο διαχειριστής έχει κάποιες συγκεκριμένες λειτουργίες που μπορεί να κάνει αποκλειστικά ο ίδιος.

Οι αγρότες είτε εγγράφονται μόνοι τους είτε απο τον διαχειριστή. Οι υπάλληλοι εγγράφονται αποκλειστικά από τον διαχειριστή.

Από τα παραπάνω συμπεραίνουμε 2 πράγματα για το σύστημα που φτιάχνουμε:

1. Έχουμε 2 μέρη. Το μέρος “λειτουργικότητα”, που αφορά την υλοποίηση των λειτουργιών που μπορούν να εκτελέσουν οι χρήστες(αγρότες και υπάλληλοι) και το μέρος “Διαχείριση Χρηστών” που αφορά αποκλειστικά τον ρόλο του διαχειριστή και τις λειτουργίες που εκτελεί.

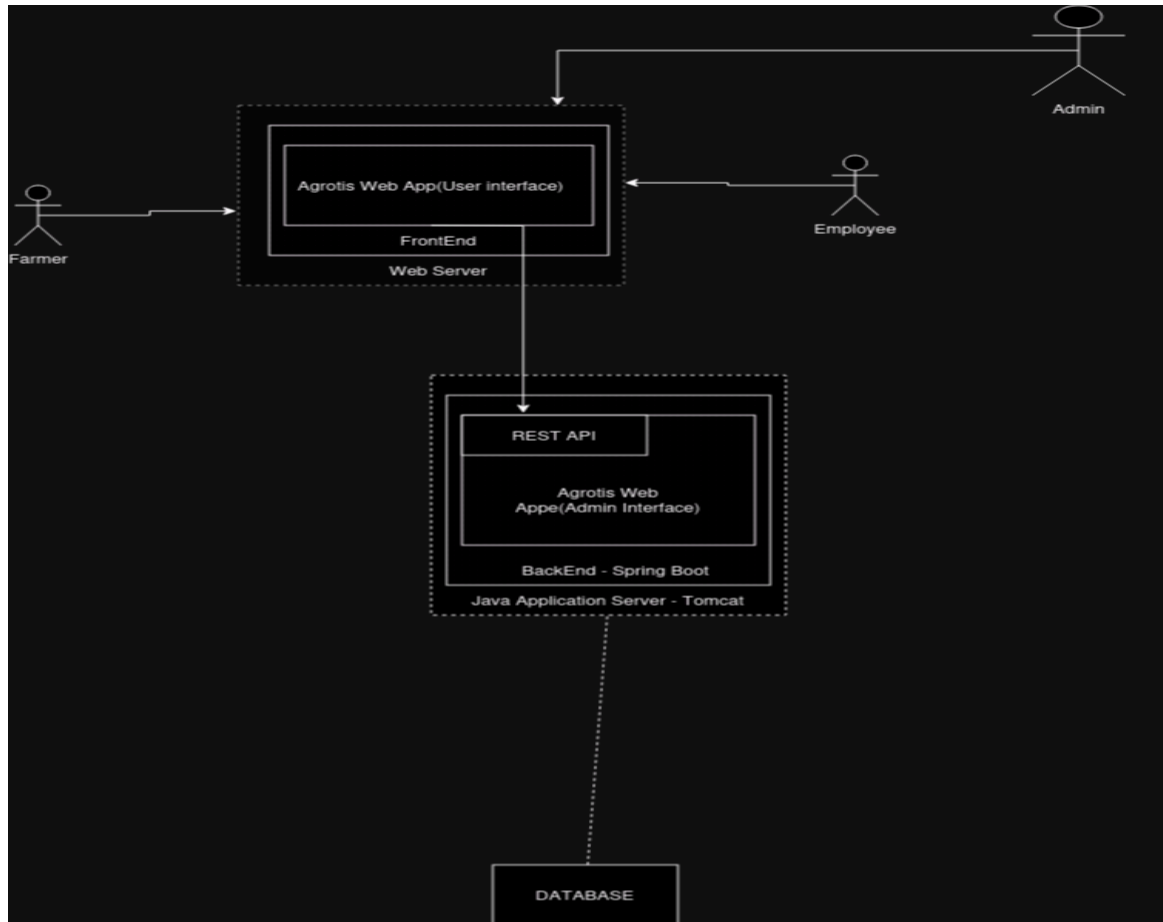
Επιλέχθηκε η παραπάνω υλοποίηση για τους παρακάτω λόγους:



1. Ασφάλεια: Απο τη στιγμή που ο διαχειριστής μπαίνει απο ξεχωριστή διεπαφή δεν υπάρχει κίνδυνος κάποιος χρήστης να έχει πρόσβαση σε λειτουργίες του διαχειριστή.
2. Εφόσον ο διαχειριστής δεν θεωρείται χρήστης και δεν έχει ενεργό ρόλο στη βασική λειτουργικότητα του συστήματος (βασική λειτουργικότητα: δημιουργία, έγκριση/απόρριψη αιτήσεων) τότε καλό είναι να μην εμπλέκονται μεταξύ τους σε επίπεδα που δεν είναι απαραίτητο. (περιορίζουμε την αλληλεπίδραση μεταξύ των 2 μερών του συστήματος σε επίπεδο Βάσης Δεδομένων)



Αρχιτεκτονική



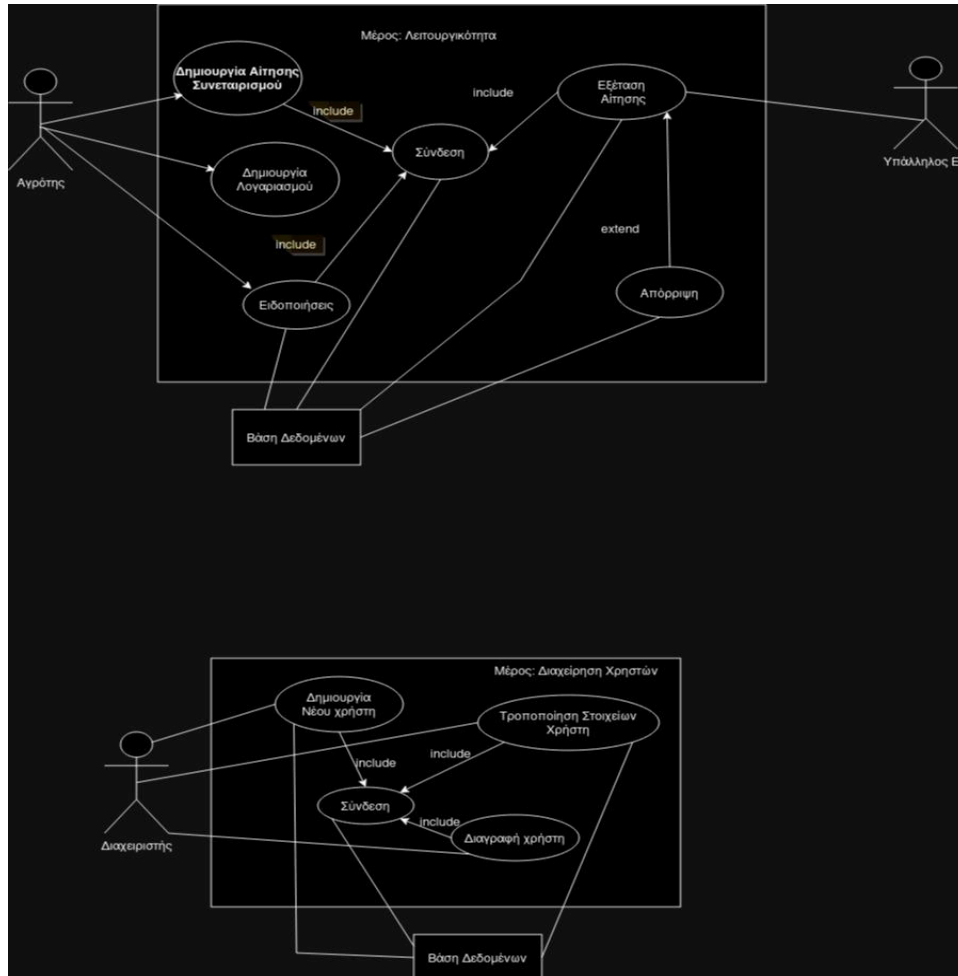
Στο διάγραμμα φαίνεται ξεκάθαρα πως οι χρήστες μπαίνουν με διαφορετικό τρόπο στο σύστημα απ'ότι ο διαχειριστής

Μπαίνουν διαφορετικά, διότι η εκφώνηση αναφέρει ότι ο διαχειριστής δεν είναι χρήστης και επιτελεί αποκλειστικά δικές του λειτουργίες. Οπότε, ο διαχειριστής εισέρχεται απευθείας στην back end app και αυτό θα είναι ουσιαστικά 2 μέρη. Το μέρος της "λειτουργικότητας" που επεκτείνεται ως τη front end app και είναι για τους χρήστες, και το μέρος της διαχείρισης χρηστών που υλοποιείται μόνο στην back end και αφορά αποκλειστικά και μόνο τον διαχειριστή.

Η αρχιτεκτονική όπως γίνεται ξεκάθαρα είναι Πελάτη-Εξυπηρετητή 3ων επιπέδων.



Use Case Διάγραμμα



Σημειώσεις:

Το βασικό διάγραμμα είναι αυτό για το μέρος “Λειτουργικότητα”.

Το διάγραμμα για το μέρος του συστήματος “Διαχείριση Χρηστών” είναι δευτερεύον και βοηθά απλά βοηθά να κατανοήσουμε όλες τις λειτουργίες που θα υλοποιηθούν στο backend.

Η λειτουργία “Σύνδεση” εμφανίζεται και στο βασικό διάγραμμα και στο διάγραμμα για τη “Διαχείριση Χρηστών” όμως είναι 2 διαφορετικές λειτουργίες.



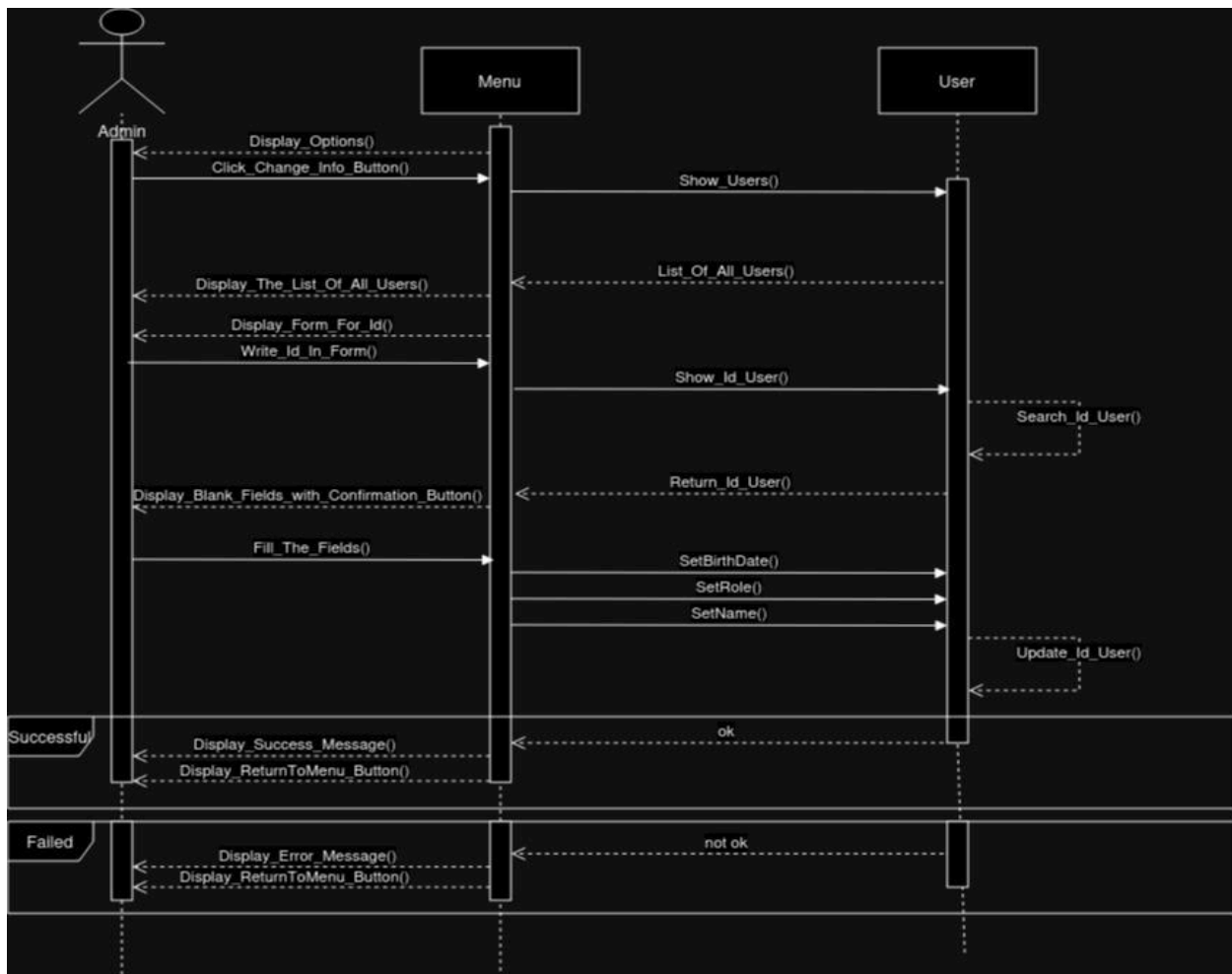
Η λειτουργία “Σύνδεση” εμφανίζεται κατά την είσοδο του χρήστη ή του διαχειριστή στο σύστημα και μετά εμφανίζεται το μενού με τις επιλογές αναλόγως τον ρόλο.(διάγραμμα ακολουθίας



Sequence Διαγράμματα

Με την σχεδίαση του front-end το return_to_Menu Button έχει αντικατασταθεί με το E-agrotis που υπάρχει στο header και εμφανίζεται συνέχεια σε όλες τις σελίδες. Οπότε στα παρακάτω διαγράμματα παραλείψτε το κουμπί αυτό. Επιπλέον υπάρχουν και άλλες αλλαγές στα sequence diagram, οι οποίες εμφανίζονται από κάτω από κάθε διάγραμμα.

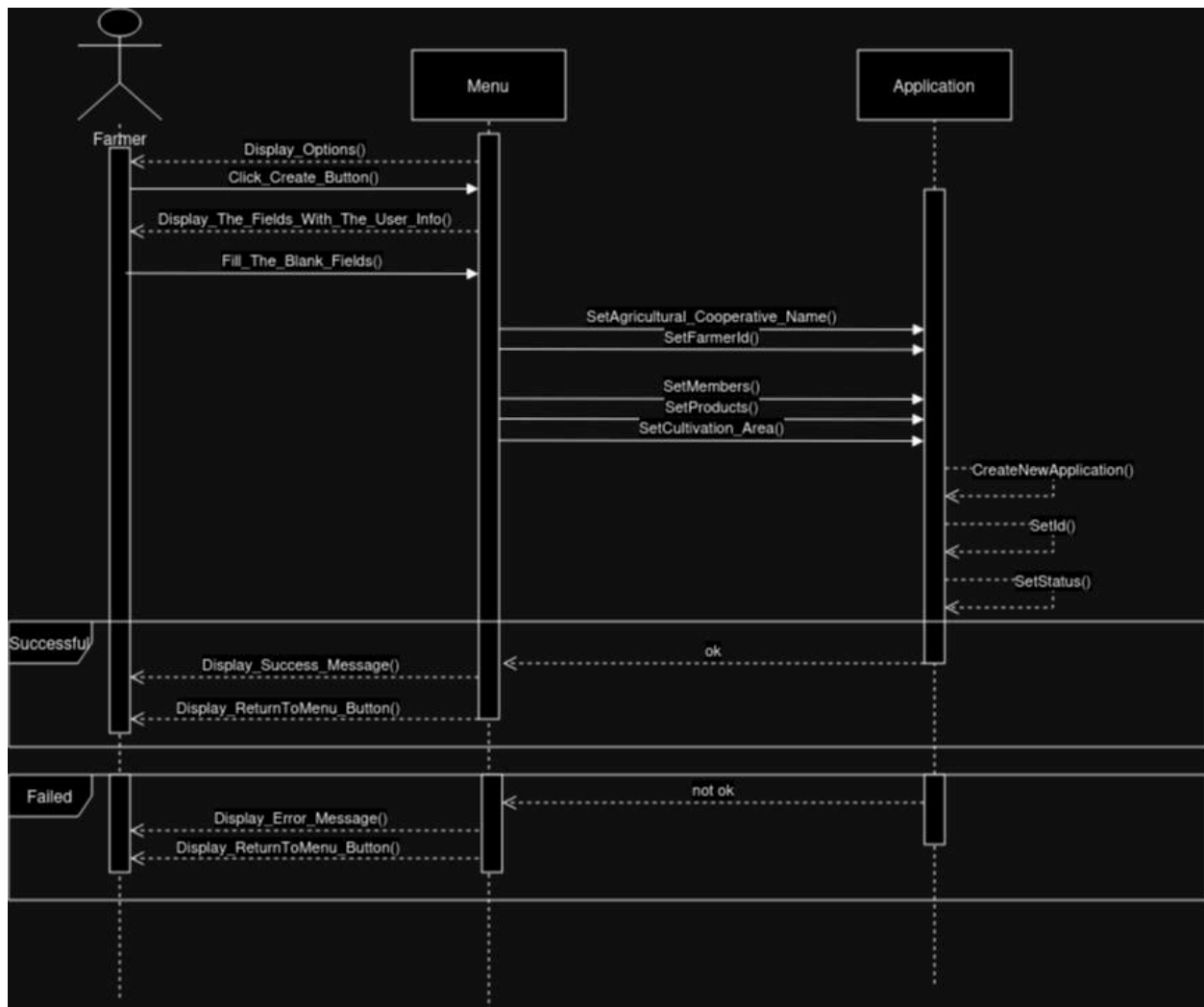
Τροποποίηση Στοιχείων Χρήστη





Σε περίπτωση επιτυχής τροποποίηση στοιχείων χρήστη επιστρέφει στην σελίδα με τον πίνακα με τα στοιχεία του χρήστη με τα αλλαγμένα στοιχεία,αντιθέτως σε περίπτωση σφαλματος εμφανίζεται μήνυμα σφάλματος.

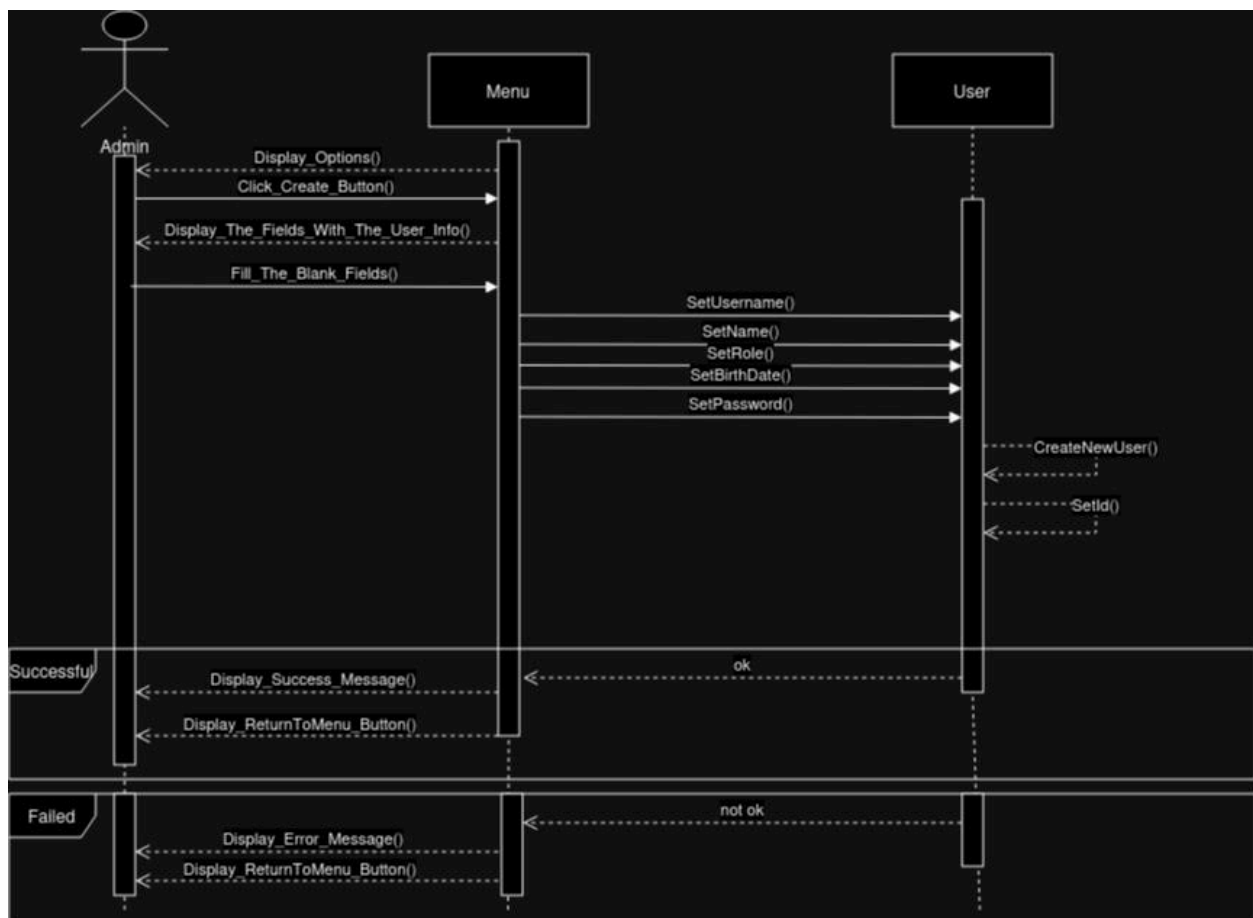
Δημιουργία Αίτησης Συνεταιρισμού





Σε περίπτωση επιτυχής δημιουργίας αίτησης επιστρέφει στο menu-home page,αντιθέτως σε περίπτωση σφαλματος εμφανίζεται μήνυμα σφάλματος.

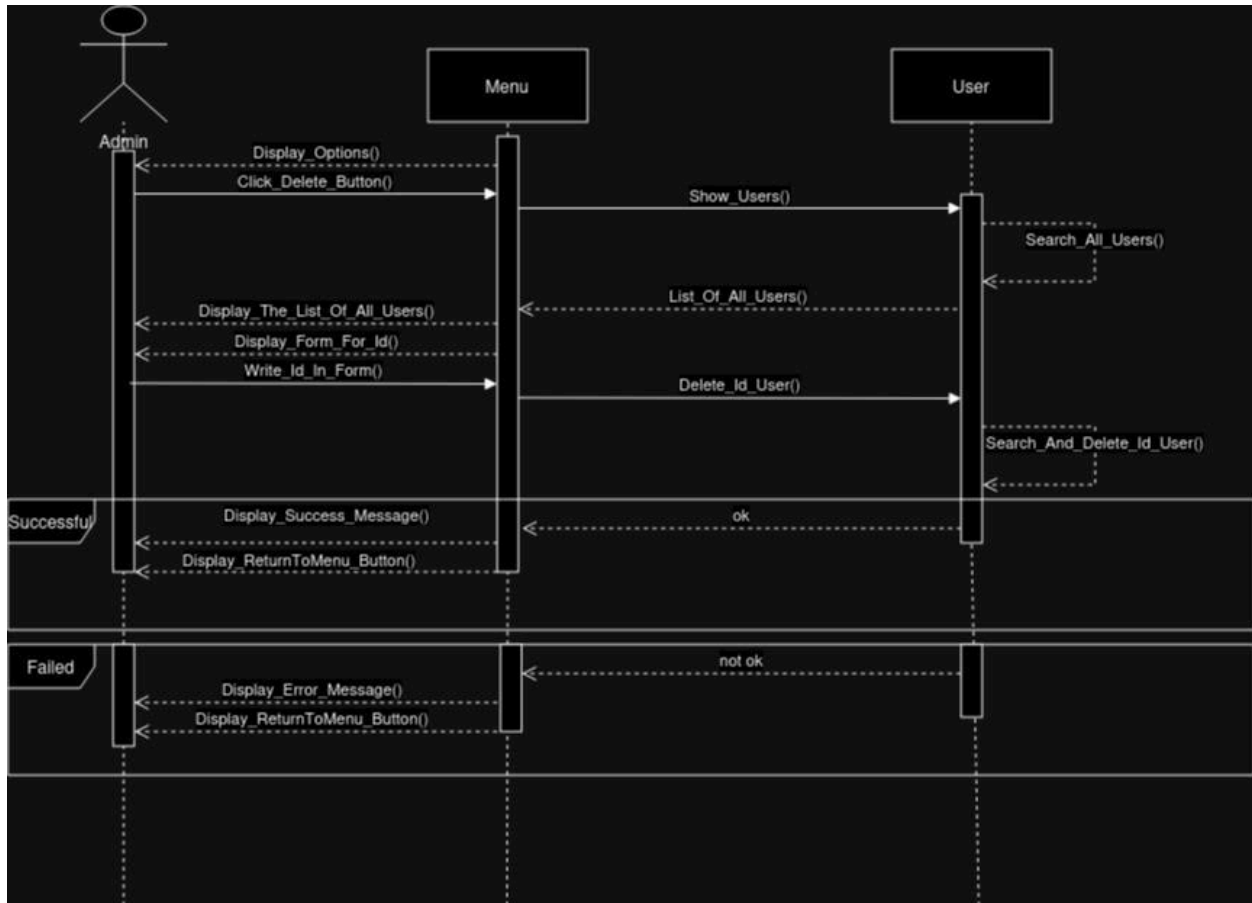
Δημιουργία Νέου Χρήστη



Σε περίπτωση επιτυχής δημιουργίας νέου χρήστη επιστρέφει στο menu-home page,αντιθέτως σε περίπτωση σφαλματος εμφανίζεται μήνυμα σφάλματος.

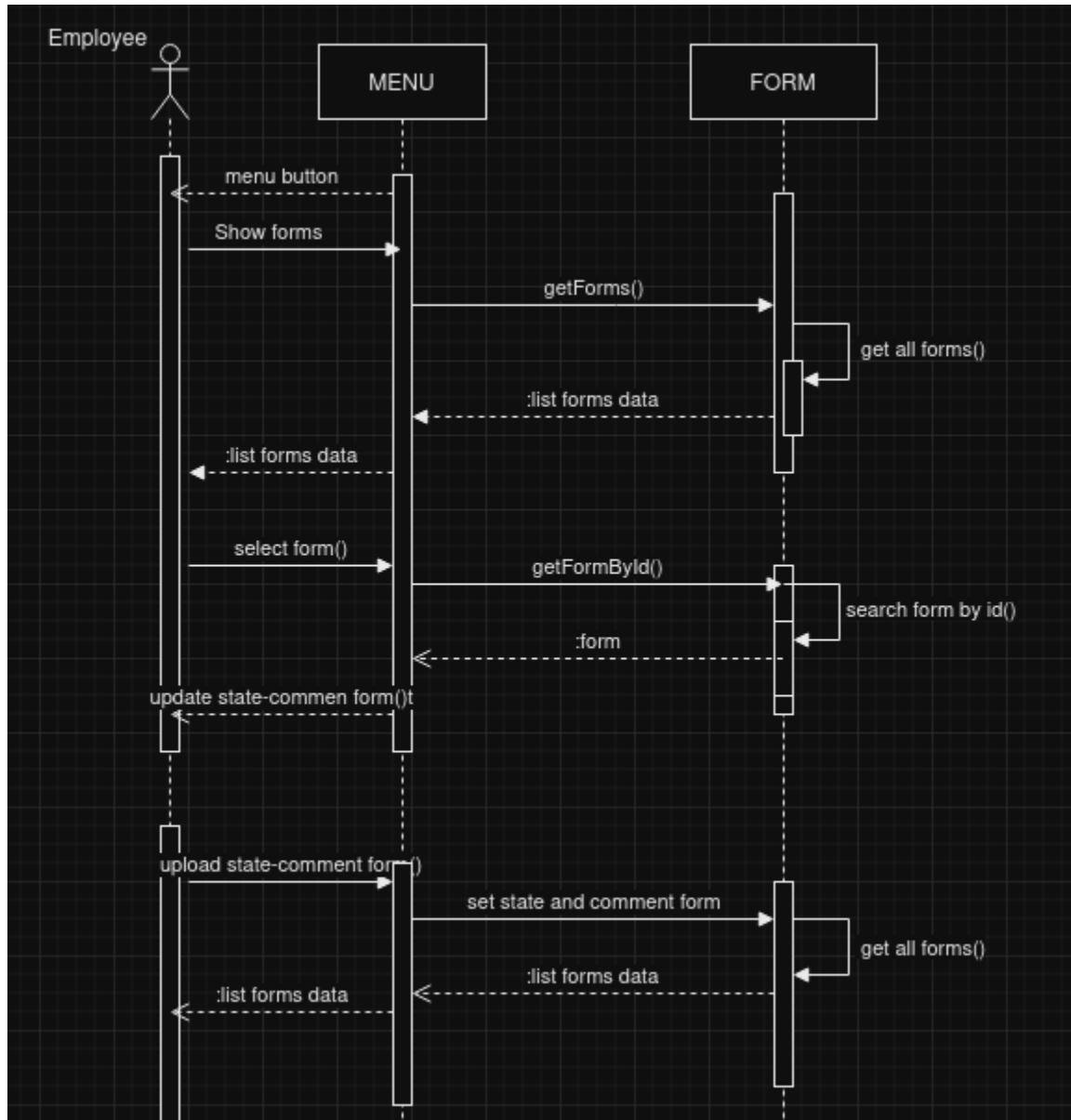


Διαγραφή Χρήστη



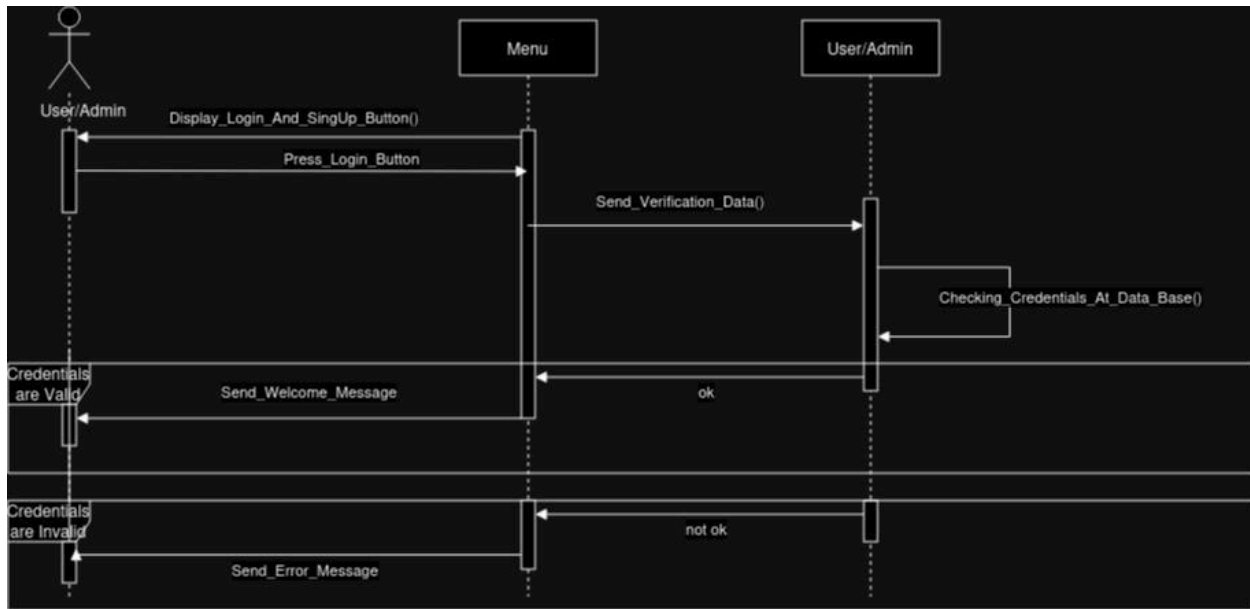
Σε περίπτωση επιτυχής διαγραφή χρήστη επιστρέφει στην σελίδα με τον πίνακα με τα στοιχεία χρήστη ,αντιθέτως σε περίπτωση σφαλματος εμφανίζεται μήνυμα σφάλματος.

Εξέταση και Απόρριψη Αίτησης

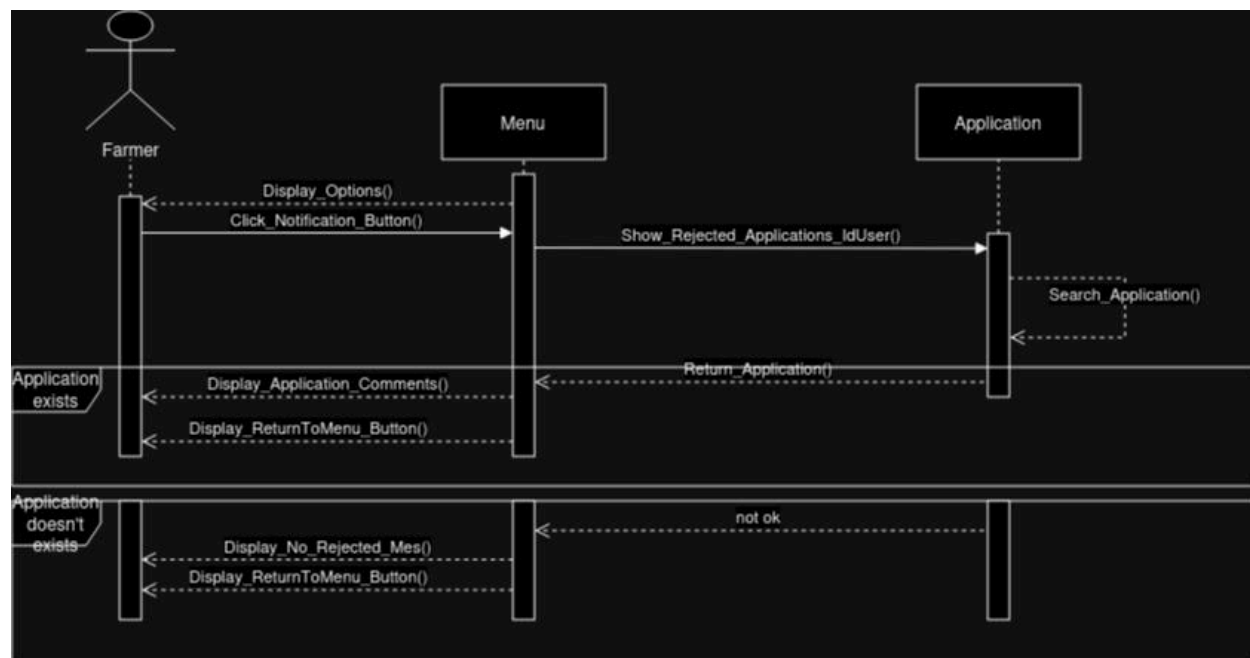




Σύνδεση

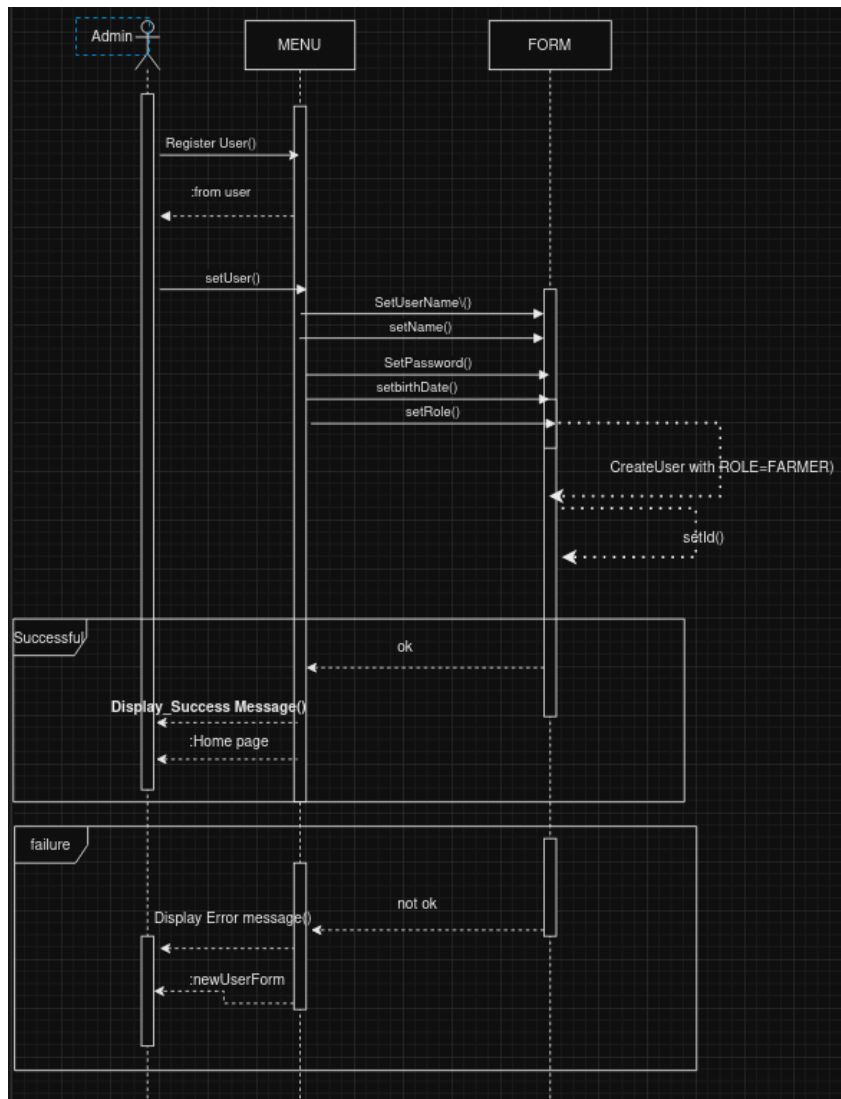


Ειδοποιήσεις



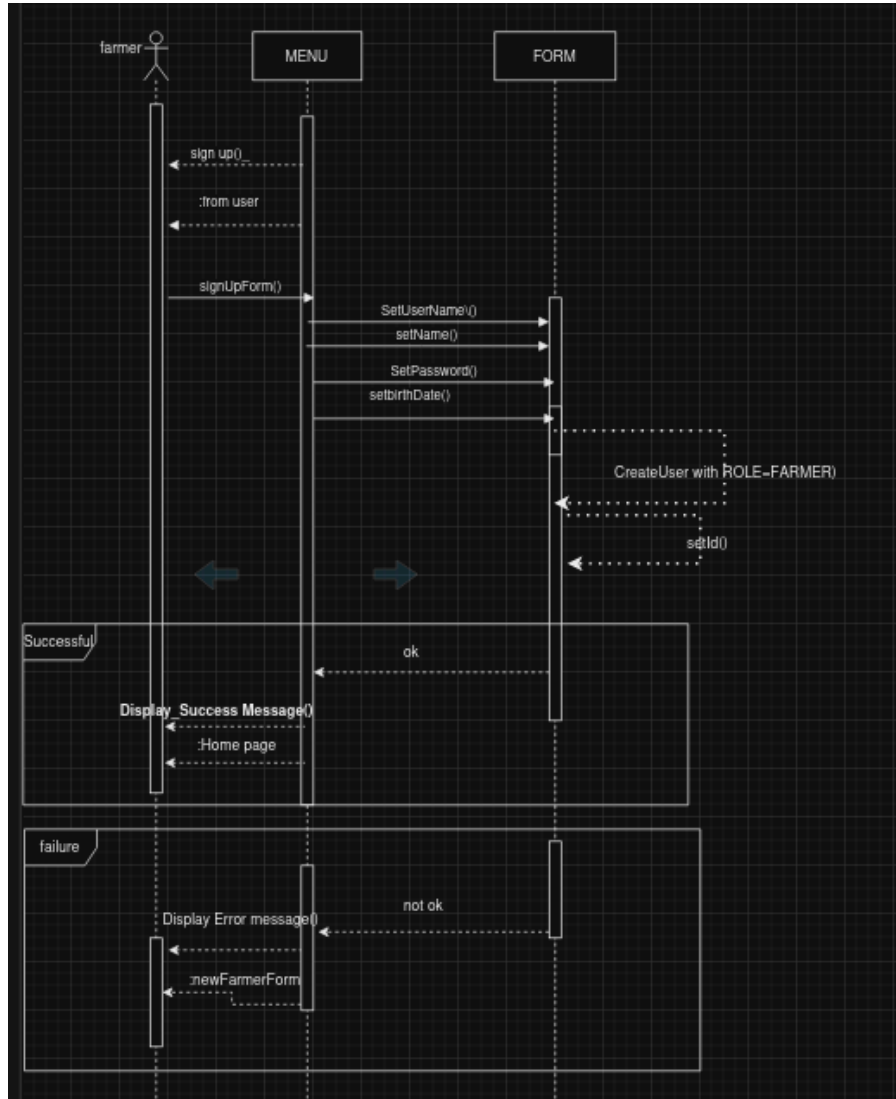


Δημιουργία Χρήστες





Εγγραφή Αγρότη στην Εφαρμογή



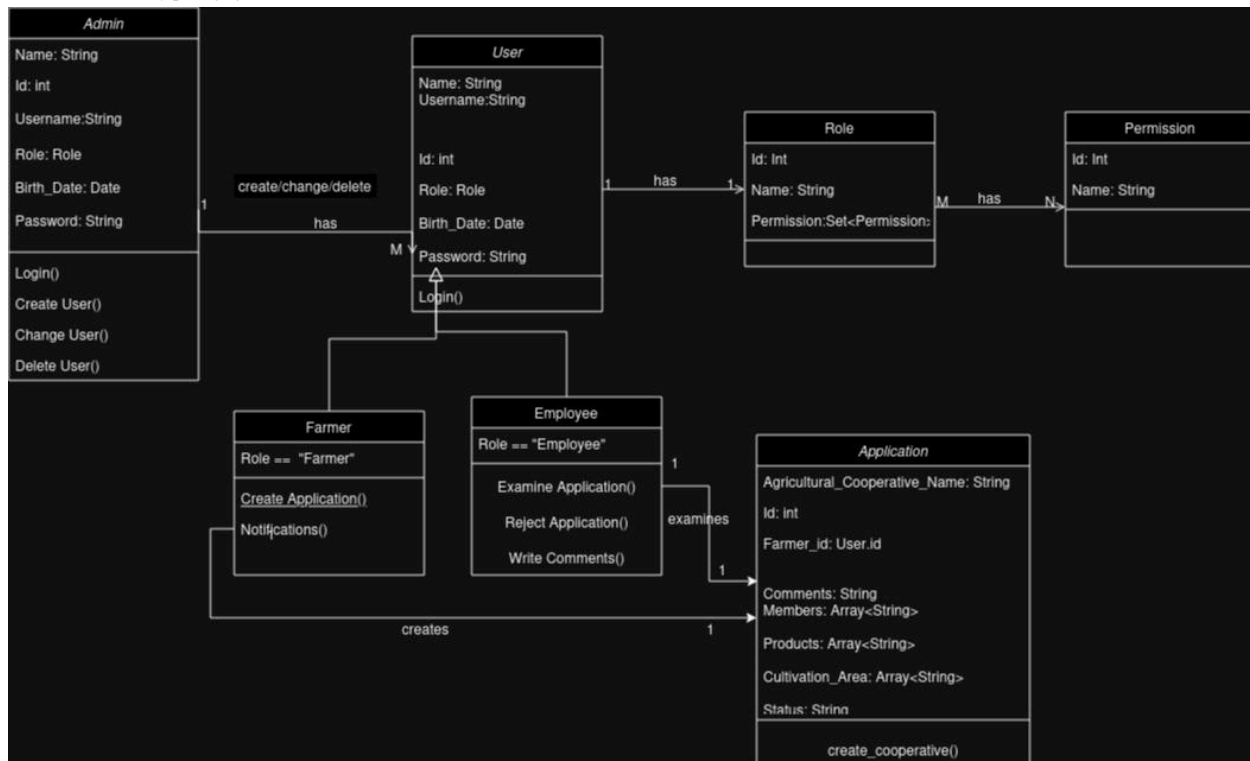
Σημειώσεις:

- Το πεδίο Id συμπληρώνεται απο το σύστημα στις λειτουργίες “Δημιουργία Λογαριασμού”, Δημιουργία Αίτησης Συνεταιρισμού” και “Δημιουργία Νέου Χρήστη”



- Τα `CreateNewUser()`, `CreateNewApplication()` και `setId()`, `setStatus()`, αναπαριστούν τους/αναφέρονται στους constructors για τη δημιουργία νέων αντικειμένων(`User`, `Application`).
- Για όλες τις λειτουργίες πέραν των “Σύνδεση” και “Δημιουργία Λογαριασμού” θεωρείται ότι ο χρήστης έχει κάνει ήδη login και στα διαγράμματα ξεκινάνε από το αμέσως επόμενο βήμα. (Το κάναμε για λόγους απλοποίησης των διαγραμμάτων. Να μην είναι υπερβολικά γεμάτα και “βαριά” στο μάτι)

Class Διάγραμμα



Παραδοχές-Σημειώσεις

Ο διαχειριστής(`Admin`) είναι ξεχωριστή class από τη κλάση `User` (έτσι ορίζεται στην εκφώνηση) αλλά στη Βάση Δεδομένων θα είναι στον ίδιο πίνακα `User` με τη χρήση του `Joined Table Inheritance` σε επίπεδο προγραμματισμού (το προσφέρει το `Spring`).



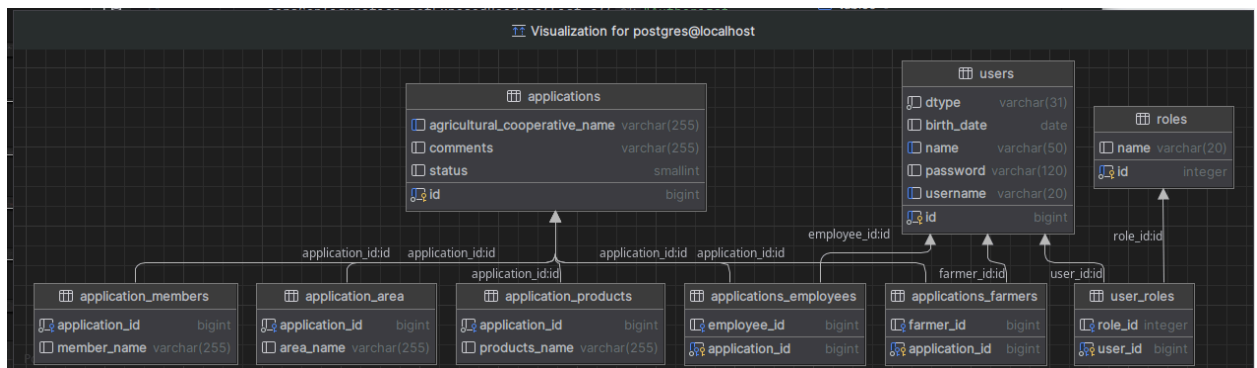
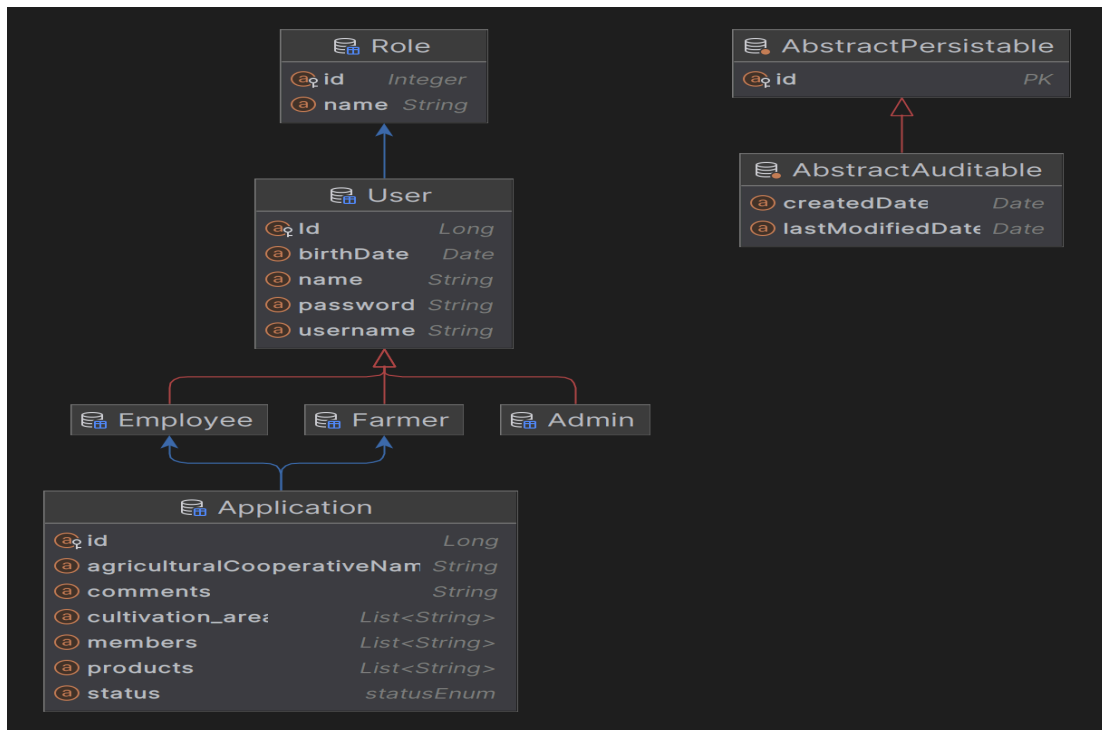
Πέρα από τον admin και τα subclasses του user όλα τα υπόλοιπα θα είναι πίνακες της Βάσης Δεδομένων μας.

Οι περιοχές καλλιέργειας του κάθε αγροτικού συναιτερισμού στην αίτηση δημιουργίας θα εμφανίζονται σε πίνακα με τα στοιχεία της κάθε περιοχής σε μορφή String.

Σε σημεία που σχετίζονται οι πίνακες όπως πχ, στη κλάση User, έχουμε το πεδίο Role, η υλοποίηση θα γίνει με ξένα κλειδιά στους αντίστοιχους πίνακες. Το id του πίνακα Role θα είναι ξένο κλειδί στον πίνακα User.

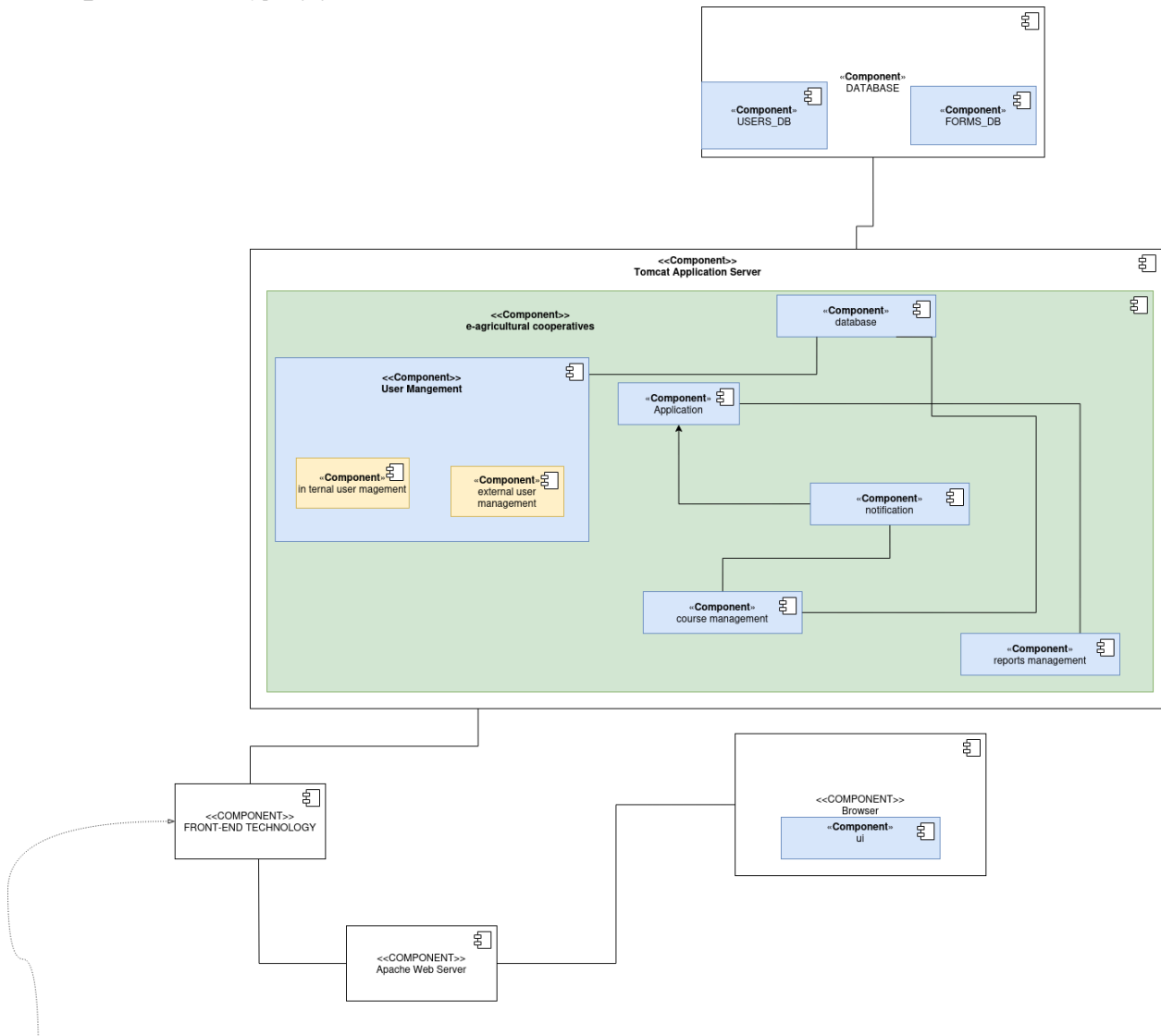
Οι λειτουργίες που αναφέρονται είναι ενδεικτικές.

- Αφαιρέσαμε τον πίνακα permissions απο τη βάση διότι ήταν πλεονασμός.(ο πίνακας των ρόλων είναι αρκετος με τη βοήθεια του spring security)Ακόμη, έγιναν μερικές μικρές αλλαγές στα attributes των classes οι οποίες φαίνονται στο παρακάτω E-R diagram:





Component Διάγραμμα



Στο front-end χρησιμοποιούμε το vue.js technology

Use Case Περιγραφές



Οι περιγραφές θα αφορούν μόνο τις λειτουργίες στο μέρος “Λειτουργικότητα” του συστήματος.

Τίτλος	Δημιουργία Αίτησης Συνεταιρισμού	
Σύντομη περιγραφή	Ο αγρότης συμπληρώνει μια φόρμα και κάνει αίτηση για τη δημιουργία ενός αγροτικού συνεταιρισμού	
Δρώντες/ Actors	Αγρότης	
Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1.Το σύστημα εμφανίζει φόρμα συμπλήρωσης στοιχείων(όνομα συνεταιρισμού, μέλη, προϊόντα, εκτάσεις καλλιέργειας , id αγρότη)	Application.name , Application.members , Application.products , Application.cult_areas, User.id
	2.Ο χρήστης συμπληρώνει τα στοιχεία.	Application.name , Application.members , Application.products , Application.cult_areas , User.id
	3.Ο χρήστης πατάει το κουμπί “Δημιουργία Αίτησης”.	
	4.Τα στοιχεία και η “Αίτηση” αποθηκεύονται στη Βάση Δεδομένων.	Application.name , Application.members , Application.products , Application.cult_areas, Application.id, User.id



	5. Το σύστημα επιστρέφει στην home page.	
Εναλλακτική ροή / βήματα	1. Αν είναι λανθασμένα συμπληρωμένα η φόρμα εμφανίζεται μήνυμα λάθους	error_mes

Τίτλος	Ειδοποιήσεις	
Σύντομη περιγραφή	Ο αγρότης βλέπει αν έχει απορριφθεί η αίτηση του και τα σχόλια απόρριψης που έχουν γίνει	
Δρώντες/ Actors	Αγρότης	
Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1. Το σύστημα αναζητά απορριφθείσες αιτήσεις του αγρότη στη Βάση Δεδομένων.	Application.id , application.state , User.id
	2. Το σύστημα βρίσκει και εμφανίζει τις αιτήσεις (όνομα συναιτερισμού)	Application.id , Application.comments



	,κατάσταση και τα σχόλια.	
Εναλλακτική ροή / βήματα	1.Το σύστημα δεν βρίσκει αίτηση.	
	2.Το σύστημα επιστρέφει μήνυμα λάθους	err_mesage

Τίτλος	Σύνδεση	
Σύντομη περιγραφή	Ο χρήστης κάνει είσοδο στη πλατφόρμα για να μπορέσει να αξιοποιήσει τις λειτουργίες της	
Δρώντες/ Actors	Αγρότης , Υπάλληλος Εποπτεύουσας Αρχής ,Admin	
Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1.Το σύστημα εμφανίζει φόρμα log in (username και κωδικό πρόσβασης) και ένα σύνδεσμο σε σελίδα δημιουργίας αγρότη χρήστη	User.username, User.password
	2.Ο χρήστης εισάγει τα στοιχεία του (username και κωδικό πρόσβασης).	User.username, User.password



	3. Έλεγχος αν ο συνδυασμός username και password υπάρχουν στην βάση δεδομένου.	User.username, User.password
	4. Ο χρήστης επιλέγει το κουμπί "Σύνδεση".	
Εναλλακτική ροή / βήματα	1. Αν δεν υπάρχει ο συνδυασμός(username, password) τότε επιστρέφεται μήνυμα λάθους.	error_mes
	2. Επιστρέφει τον χρήστη στο βήμα 2 της βασικής ροής.	

Τίτλος	Δημιουργία Λογαριασμού	
Σύντομη περιγραφή	Ο χρήστης δημιουργεί έναν λογαριασμό για το σύστημα	
Δρώντες/ Actors	Admin	
Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1. Το σύστημα εμφανίζει φόρμα συμπλήρωσης στοιχείων(όνομα, ρόλος, username, κωδικός, ημερομηνία γέννησης)	User.name , User.username , User.password , User.role , User.birthdate



	2.Ο χρήστης συμπληρώνει τα στοιχεία.	User.name , User.username , User.password , User.role , User.birthdate
	3.Ο χρήστης πατάει το κουμπί εγγραφή.	
	4.Τα στοιχεία και ο “Χρήστης” αποθηκεύονται στη Βάση Δεδομένων.	User.name , User.username , User.password , User.role , User.birthdate, User.id
	5.Το σύστημα εμφανίζει μήνυμα επιτυχούς εγγραφής και επιστρέφει στο μενού-home της εφαρμογής.	succ_mes
Εναλλακτική ροή / βήματα	1. Αν είναι λανθασμένα συμπληρωμένα η φόρμα εμφανίζεται μήνυμα λάθους	error_mes
	2.Επιστρέφει τον χρήστη στο βήμα 3	

Τίτλος	Εξέταση Αίτησης
Σύντομη περιγραφή	Ο υπάλληλος εξετάζει τις αιτήσεις και επιλέγει ποιές θα περάσουν και ποιές θα απορριφθούν.
Δρώντες/ Actors	Υπάλληλος Εποπτεύουσας Αρχής



Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1.Το σύστημα αναζητά όλες τις αιτήσεις στη βάση δεδομένων	
	2.Το σύστημα εμφανίζει όλες τις αιτήσεις στον υπάλληλο.	Application.id , Application.name , Application.state
	3.Ο υπάλληλος επιλέγει μια αίτηση.	
	4.Το σύστημα αναζητά την αίτηση στη Βάση Δεδομένων.	Application.id
	5. Το σύστημα επιστρέφει όλα τα στοιχεία της αίτησης και εμφανίζει τα κουμπιά “Disproved”, “Unseen” και “Approved”,και text για σχόλια	Application.name , Application.id , Application.members , Application.products , Application.cult_areas , Application.state,Application.comments User.id
	6. Ο υπάλληλος επιλέγει “Approved”.	



	7. Το σύστημα αλλάζει τη κατάσταση της αίτησης σε “Approved”.	Application.state
	8. Επιστροφή στο βήμα 1.	
Εναλλακτική ροή / βήματα	1. Ο χρήστης επιλέγει “”	
	2. Το σύστημα αλλάζει τη κατάσταση της αίτησης σε “Disproved”.	Application.state
	3. Επιστροφή στο βήμα 1.	

Τίτλος	Δημιουργία Λογαριασμού για Αγρότες
--------	------------------------------------



Σύντομη περιγραφή	Ο χρήστης δημιουργεί έναν λογαριασμό για το σύστημα	
Δρώντες/ Actors	Αγρότης	
Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1.Το σύστημα εμφανίζει φόρμα συμπλήρωσης στοιχείων(όνομα, ρόλος, username, κωδικός , ημερομηνία γέννησης)	User.name , User.username , User.password , User.role , User.birthdate
	2.Ο χρήστης συμπληρώνει τα στοιχεία.	User.name , User.username , User.password , User.role , User.birthdate
	3.Ο χρήστης πατάει το κουμπί εγγραφή.	
	4.Τα στοιχεία και ο “Χρήστης” αποθηκεύονται στη Βάση Δεδομένων.	User.name , User.username , User.password , User.role , User.birthdate, User.id
	5.Το σύστημα εμφανίζει μήνυμα επιτυχούς εγγραφής και ένα κουμπί για να πάει στο μενού -home της εφαρμογής.	succ_mes
Εναλλακτική ροή / βήματα	1. Αν είναι λανθασμένα συμπληρωμένα η φόρμα εμφανίζεται μήνυμα λάθους	error_mes



	2.Επιστρέφει τον χρήστη στο βήμα 3	
--	------------------------------------	--

Τίτλος	Εμφάνιση Στοιχειων Χρηστων-Αγροτων-Υπαλλήλων	
Σύντομη περιγραφή	Ο admin επιλέγει από την αρχική σελίδα ,αν θέλει να δει τα στοιχεία των χρηστών ή των αγροτών ή των υπαλλήλων.Και στην συνέχεια του δίνεται η επιλογή να επιλέξει αν θέλει να δει έναν συγκεκριμένο χρήστη.	
Δρώντες/ Actors	Διαχειριστής	
Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1.Το σύστημα εμφανίζει ένα μένου Users,Employees,Farmers	
	2.Επιλέγει ένα κουμπί από τα παραπάνω	
	3.Το σύστημα αναζητά τους χρήστες στη Βάση Δεδομένων.	



	4.Εμφανιση πίνακα με τους όλους τους χρήστες ή τους αγρότες ή τους υπαλλήλους .Επιπλέον ένα κουμπί διπλά από κάθε χρήστη	User.name , User.username , button:More Details
	5.Πατάει το κουμπί More Details	
	6.Το σύστημα αναζητά τον χρήστη με το συγκεκριμένο id στη Βάση Δεδομένων.	
	7.Το σύστημα εμφανίζει τα στοιχεία του επιλεγμένου χρήστη και δυο κουμπια για αλλαγή στοιχείων χρήστη ή διαγραφή του χρήστη	User.name , User.username , User.password , User.role , User.birthdate, User.id
Εναλλακτική ροή / βήματα	1. Αν δεν υπάρχουν χρήστες εμφανίζεται μήνυμα λάθους	error_mes

Τίτλος	Αλλαγή στοιχείων Χρήστη
Σύντομη περιγραφή	Ο admin επιλέγει να αλλάξει τα στοιχεία ενός χρήστη.



Δρώντες/ Actors	Διαχειριστής	
Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1.Το σύστημα εμφανίζει μια φόρμα για να συμπληρώσει τα στοιχεία και ένα κουμπί “Update User”	
	2.Συμπληρώνει τα στοιχεία	User.name , User.username , User.password , User.role , User.birthdate, User.id
	3.Πατάει το κουμπί Update User	
	4.Πατάει More Details	
	5.Τα στοιχεία και ο “Χρήστης” αποθηκεύονται στη Βάση Δεδομένων.	User.name , User.username , User.password , User.role , User.birthdate, User.id
	6.Επιστροφή στην σελίδα με τα στοιχεία του χρήστη που αλλάξαμε.	User.name , User.username , User.password , User.role , User.birthdate, User.id
Εναλλακτική ροή / βήματα	1. Αν υπαρξη καποιο λάθος εμφανίζεται μήνυμα λάθους	error_mes



Τίτλος	Διαγραφή Χρήστη	
Σύντομη περιγραφή	Ο admin επιλέγει να διαγράψει έναν χρήστη.	
Δρώντες/ Actors	Διαχειριστής	
Ροή / Βήματα	Βήματα / Διεργασίες(Tasks)	Δεδομένα
	1.Το σύστημα εμφανίζει ένα κουμπί Delete	
	2.Ο διαχειριστής το πατάει	
	3.Τα στοιχεία και ο “Χρήστης” διαγράφονται στη Βάση Δεδομένων.	
	4.Πατάει More Details	
	5.Επιστροφή στην σελίδα με το μενού.	



Εναλλακτική ροή / βήματα	1. Αν υπάρξει κάποιο λάθος εμφανίζεται μήνυμα λάθους	error_mes
-----------------------------	--	-----------

Σημείωση:

- Τα μηνύματα λάθους λαμβάνονται ως δεδομένα διότι περιέχουν μια μικρή περιγραφή του λάθους και έναν κωδικό λάθους.
- Οι λειτουργίες ξεκινάνε αφού ο χρήστης πατήσει το κουμπί της εκάστοτε λειτουργίας.

Παρακάτω βρίσκεται το link για να μπορέσετε να κατεβάσετε την εφαρμογή και τον κώδικα στο μηχανήμά σας:

To backend:

<https://github.com/it2021083/backend>



Στο link θα βρείτε μαζί με την εφαρμογή και κάποια αρχεία:

Στο Technologies.pdf θα βρίσκονται πληροφορίες σχετικές με τις τεχνολογίες που χρησιμοποιήθηκαν την ανάπτυξη του κώδικα.

Στο RestApi.pdf θα υπάρχουν πληροφορίες για το πώς να χρησιμοποιηθεί το REST API και τη λειτουργία του κάθε endpoint.

Στο notes.txt θα υπάρχουν αναφορές σε σημεία που δυσκολευτήκαμε ή που δεν καταφέραμε να επιτύχουμε και πράγματα που δεν λειτουργούν σωστά.

To frontend:

<https://github.com/it2021083/frontend>

Πηγές:

Μεγάλο μέρος του κώδικα χρησιμοποιήθηκε από τους δικούς σας κώδικες στο github.

Χρησιμοποιήθηκε και η βοήθεια του chatgpt.



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
HAROKOPIO UNIVERSITY

.dit/*

Informatics & Telematics