

# **Μια Πλατφόρμα Κοινωνικής Δικτύωσης για Φιλοζωικές Ενέργειες**

**Παναγιώτης Παπαϊωάννου**

**Διπλωματική Εργασία**

**Επιβλέπων: Απόστολος Ζάρρας**

**Ιωάννινα, Ιούλιος, 2022**



**ΤΜΗΜΑ ΜΗΧ. Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ**

---

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
UNIVERSITY OF IOANNINA**



# Ευχαριστίες

Για την επιτυχή ολοκλήρωση αυτής της διπλωματικής εργασίας θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου, κ. Απόστολο Ζάρρα, για την βοήθεια και την καθοδήγησή του σε όλη τη διάρκειά της. Επίσης θέλω να ευχαριστήσω τους φίλους και την οικογένειά μου, για την στήριξή τους και για τη βοήθειά τους στην αξιολόγηση του τελικού αποτελέσματος.

Πέμπτη, 7 Ιουλίου, 2022

Παναγιώτης Παπαϊωάννου

# Περίληψη

Σήμερα, οι φιλόζωοι εθελοντές και φιλοζωικές οργανώσεις, δρουν με τη βοήθεια των κοινωνικών δικτύων (π.χ. Facebook, Instagram) και με βάση την ανταπόκριση άλλων χρηστών οργανώνουν τη δράση τους, σχηματίζοντας κοινότητες. Βασικό όμως μειονέκτημα αυτής της τεχνικής είναι ο τεράστιος όγκος πληροφορίας που δέχονται οι χρήστες των κοινωνικών δικτύων. Επιπλέον, ο χώρος των ιστοσελίδων/εφαρμογών αυτών δεν είναι αφιερωμένος αποκλειστικά στη φιλοζωία. Αναρτήσεις μπορούν να χαθούν εύκολα ή να προωθηθούν παραπάνω από όσο θα χρειαζόταν, δεν υπάρχει δικαιοσύνη αλλά ούτε και επιλογή από τον χρήστη για το ποιες αναρτήσεις θα δει. Επηρεαζόμενοι από το δύσκολο έργο των φιλόζωων και την μη επαρκή βοήθεια που προσφέρουν τα κοινωνικά δίκτυα, δημιουργήσαμε μία εφαρμογή με στόχο να το διευκολύνει και να αναβαθμίσει τις ζωές των αδέσποτων ζώων. Δημιουργήσαμε μία εφαρμογή τύπου κοινωνικό δίκτυο, ενώνοντας όλους τους ανθρώπους που αγαπούν τα ζώα, καθώς αυτή αφορά μόνο φιλοζωικά ζητήματα. Σε αυτή, μπορεί να εγγραφεί οποιοσδήποτε που τον αφορούν αυτά τα ζητήματα (φιλοζωικές οργανώσεις, εθελοντές και απλοί φιλόζωοι) και να αναρτήσει δημοσιεύσεις σχετικές με την υιοθεσία κάποιου ζώου, τον χαμό του κατοικίδιού τους και την φροντίδα ενός αδέσποτου ζώου. Ακόμα μπορεί να κάνει ανάρτηση ενός κειμένου στο οποίο μπορεί να ζητά συμβουλές, να λέει τη γνώμη του ή και να ενημερώνει τους άλλους χρήστες για τοπικά ζητήματα όπως την ύπαρξη ενός ζώου που χρειάζεται βοήθεια στην περιοχή του. Όλοι οι τύποι αναρτήσεων μπορούν να συνοδευτούν από ένα σημείο στον χάρτη που υποδεικνύει την τοποθεσία του εκάστοτε ζώου. Φυσικά, οι υπόλοιποι χρήστες μπορούν να σχολιάσουν τις αναρτήσεις αυτές, για να βοηθήσουν ή να δείξουν ενδιαφέρον. Τέλος, υπάρχει η δυνατότητα δημιουργίας «εκδήλωσης» στην οποία περιγράφεται ένα μελλοντικό γεγονός φιλοζωικού σκοπού, στο οποίο οι χρήστες μπορούν να δηλώσουν ότι επιθυμούν να παρευρεθούν.

**Λέξεις Κλειδιά:** Φιλοζωία, Κοινωνικά Δίκτυα, Διαδικτυακή Εφαρμογή

# Abstract

Today, animal welfare organizations and volunteers, act with the help of social networking platforms (e.g., Facebook, Instagram) and based on the response of other users the organize their activities, forming communities. Main disadvantage of this technique is the huge volume of information that the users receive. Moreover, the space of these websites/applications is not dedicated to animal welfare. Posts can easily get lost, or even get promoted unevenly and unfairly, the user has no control on what posts they can view. Influenced from the challenging work of volunteers and the subpar assistance from social media, we created an application with the purpose to facilitate that work and enhance the lives of stray animals. We developed a social networking platform, to connect all the animal lovers, as it only concerns animal issues. Anyone that relates to these issues can register on this application (animal welfare organizations, volunteers, and simple individuals) and publish animal adoption posts, missing posts for lost pets and the wellness of stray animals. They can also create posts asking for advice and help, as well as post with their opinion on certain subjects and additionally posts that inform other users about issues locally such as the existence of an animal in need of assistance in their area. All kinds of posts are accompanied with a map location that indicates the geographic location of each animal. Naturally, other users can have discussions on these posts to offer help or show interest. Finally, there is the possibility of creating an "event" which describes a future animal welfare activity, in which users can state their wish to participate.

Keywords: Animal Welfare, Social Networks, Web App

# Πίνακας Περιεχομένων

<b>Κεφάλαιο 1. Εισαγωγή.....</b>	<b>1</b>
1.1 Πηγή Έμπνευσης και Αναγκαιότητα.....	1
1.2 Αντικείμενο Διπλωματικής .....	1
1.3 Οργάνωση Κεφαλαίων .....	2
<b>Κεφάλαιο 2. Σχετική Δουλειά.....</b>	<b>3</b>
2.1 Μέσα Κοινωνικής Δικτύωσης .....	3
2.2 Εφαρμογές Αποκλειστικά Φιλοζωικού Περιεχομένου .....	5
2.2.1 <i>Dogs' Voice</i> .....	5
2.2.2 <i>Mytracknet</i> .....	6
2.2.3 <i>Spot a Stray</i> .....	7
2.2.4 <i>Your Pet</i> .....	8
2.3 Rescue.....	9
<b>Κεφάλαιο 3. Ανάλυση Απαιτήσεων.....</b>	<b>11</b>
3.1 Ιστορίες Χρήστη.....	11
3.1.1 Διαχείριση Λογαριασμού .....	11
3.1.2 Διαχείριση Δημοσιεύσεων.....	12
3.1.3 Κοινότητα.....	15
3.2 Περιπτώσεις Χρήσης.....	16
3.2.1 Διαχείριση Λογαριασμού .....	16
3.2.2 Διαχείριση Δημοσιεύσεων.....	21
3.2.3 Κοινότητα.....	34
<b>Κεφάλαιο 4. Σχεδίαση και Αρχιτεκτονική Λογισμικού .....</b>	<b>42</b>
4.1 Σχεδίαση Πρωτοτύπου .....	42
4.1.1 Σύνδεση και Εγγραφή.....	42
4.1.2 Κυρίως Εφαρμογή.....	44
4.1.3 Κοινότητα.....	48
4.1.4 Προσωπικό Προφίλ και Λογαριασμός .....	49
4.2 Αρχιτεκτονική Συστήματος .....	50
4.2.1 Πακέτα Συστήματος <i>Back End</i> .....	51

4.2.2	<i>Model</i> .....	52
4.2.3	<i>Repository</i> .....	55
4.2.4	<i>Service</i> .....	56
4.2.5	<i>Controller</i> .....	57
4.2.6	<i>Συμπληρωματικές Κλάσεις</i> .....	63
4.2.7	<i>Front End</i> .....	64
<b>Κεφάλαιο 5.</b>	<b>Έλεγχος</b> .....	<b>67</b>
5.1	Έλεγχος Μεθόδων Πακέτου repository .....	67
5.2	Έλεγχος Μεθόδων Πακέτου service .....	69
5.3	Έλεγχος Μεθόδων Πακέτου controller .....	72
<b>Κεφάλαιο 6.</b>	<b>Αξιολόγηση της Εφαρμογής από τους Χρήστες</b> .....	<b>74</b>
6.1	Προφίλ των Χρηστών .....	74
6.2	Ερωτήσεις Αλληλεπίδρασης με την Εφαρμογή .....	76
6.3	Προαιρετικές Ερωτήσεις Ανάπτυξης .....	79
<b>Κεφάλαιο 7.</b>	<b>Επίλογος</b> .....	<b>81</b>
7.1	Σύνοψη και Συμπεράσματα .....	81
7.2	Μελλοντικές Επεκτάσεις και Βελτιώσεις .....	81

# Κεφάλαιο 1. Εισαγωγή

## 1.1 Πηγή Έμπνευσης και Αναγκαιότητα

Μπαίνοντας σιγά σιγά τα τελευταία χρόνια στον χώρο της φιλοζωίας και του εθελοντισμού, παρατηρούμε πως ο τρόπος που οι ίδιοι οι εθελοντές και οι φιλοζωικές οργανώσεις δραστηριοποιούνται κατά ένα μεγάλο ποσοστό είναι στα κοινωνικά δίκτυα. Μέσω αυτών (δείχνοντας μεγαλύτερη προτίμηση στο Instagram και στο Facebook) προωθούν όχι μόνο τη φιλοζωική τους δράση, αλλά και τις αγγελίες κατοικίδιων προς υιοθεσία, εκκλήσεις οικονομικής βοήθειας και άλλα. Ταυτόχρονα, αν κάποιος εθελοντής και μη χάσει το κατοικίδιό του η πρώτη του πράξη είναι να κάνει ανάρτηση στα μέσα κοινωνικής δικτύωσης, με μία φωτογραφία.

Καθώς κάθε πλατφόρμα έχει διαφορετικό κοινό, αλλά και εντός αυτών υπάρχουν εκατοντάδες διαφορετικές ομάδες/κοινότητες που είναι αφιερωμένες στις παραπάνω πράξεις, τα άτομα κατακλύζονται από τέτοιες δημοσιεύσεις. Στον αντίποδα, άτομα τα οποία μπορεί να ενδιαφέρονταν για παράδειγμα να υιοθετήσουν κάποιο κατοικίδιο δεν μπορούν να δουν με κάποιο δομημένο τρόπο τα διαθέσιμα προς υιοθεσία ζώα.

Αυτά κάνουν να σκεφτούμε, αν θα μπορούσε να υπάρξει κάποιος πιο εύκολος και δομημένος τρόπος να γίνει η ενημέρωση του κοινού για τη δράση, τις υιοθεσίες, τα χαμένα κατοικίδια, τα αδέσποτα ζώα των οποίων την φροντίδα έχουν αναλάβει εθελοντές, αλλά και έκτακτα περιστατικά.

## 1.2 Αντικείμενο Διπλωματικής

Σκοπός αυτής της εργασίας είναι να δημιουργηθεί ένα εργαλείο για την αποδοτικότερη δουλειά των φιλόζων αλλά και το οποίο να δίνει μεγαλύτερη πιθανότητα στις επανασυνδέσεις ιδιοκτήτη – κατοικίδιου. Η κατασκευή δηλαδή μίας εφαρμογής τύπου κοινωνικό δίκτυο, που να ενώνει τα ενδιαφερόμενα και τα εμπλεκόμενα άτομα ανάλογα



την εκάστοτε ανάγκη ή ενδιαφέρον. Η εφαρμογή αυτή έχει ως βασικό στόχο την δημιουργία μίας φιλοζωικής κοινότητας, η οποία έχει ως κύριο συστατικό την αλληλοβοήθεια μεταξύ των φιλόζων και την προστασία των αδέσποτων ζώων. Με τη δημιουργία αυτής της εφαρμογής έχουμε ως σκοπό την συγκέντρωση των εκατοντάδων αναρτήσεων και την κατηγοριοποίησή τους ανάλογα με την κατηγορία στην οποία ανήκουν, ώστε να υπάρχει έμπρακτη δράση και αποτέλεσμα. Η εργασία στοχεύει επίσης, στην ενημέρωση του εκάστοτε χρήστη μόνο με αναρτήσεις που βρίσκονται κοντά του. Ο χρήστης δηλαδή, να μπορεί να οριοθετεί την δράση του και να μην χάνει τον χρόνο του από τον βομβαρδισμό με αναρτήσεις που αφορούν ζώα που βρίσκονται σε άλλη πόλη.

## **1.3 Οργάνωση Κεφαλαίων**

Το υπόλοιπο της διπλωματικής εργασίας περιλαμβάνει τα εξής κεφάλαια: Αρχικά, στο Κεφάλαιο 2 γίνεται ανάλυση στον τρόπο χρήσης των μέσων κοινωνικής δικτύωσης από φιλόζωνους, όπως και αναφορές σε άλλες παρόμοιες εφαρμογές από τις οποίες μπορούμε να εμπνευστούμε. Στο Κεφάλαιο 3, περιέχονται οι ιστορίες χρήστη καθώς και οι περιπτώσεις χρήσεων της εφαρμογής. Στο Κεφάλαιο 4 περιλαμβάνεται η σχεδίαση της εφαρμογής με τη χρήση πρωτοτύπων, αλλά και η αρχιτεκτονική του λογισμικού, που αφορά κυρίως λεπτομέρειες για τα πακέτα και τις κλάσεις. Στο Κεφάλαιο 5 γίνεται αναφορά σε όλους τους ελέγχους που υποβλήθηκε η εφαρμογή. Στο Κεφάλαιο 6 υπάρχει η αξιολόγηση της εφαρμογής η οποία είναι βασισμένη σε ερωτήσεις που έγιναν σε πραγματικούς χρήστες. Τέλος, το Κεφάλαιο 7 αποτελεί τον επίλογο και περιλαμβάνει σχολιασμό για βελτιώσεις και επεκτάσεις καθώς και τα τελικά συμπεράσματα.

## Κεφάλαιο 2. Σχετική Δουλειά

Στο κεφάλαιο αυτό θα γίνει περιγραφή του τρόπου χρήσης των κοινωνικών δικτύων για την εκπλήρωση φιλοζωικού έργου και ταυτόχρονα θα αναλυθούν οι παρόμοιες εφαρμογές από τις οποίες μπορούμε να εμπνευστούμε για την εφαρμογή μας.

### 2.1 Μέσα Κοινωνικής Δικτύωσης

Σήμερα, οι φιλόζωοι εθελοντές χρησιμοποιούν σχεδόν αποκλειστικά τα μέσα κοινωνικής δικτύωσης για την διευκόλυνση του έργου τους. Καθημερινά αναρτώνται αμέτρητες δημοσιεύσεις που αφορούν υιοθεσίες κατοικίδιων, εκκλήσεις οικονομικής βοήθειας για αναγκαίες χειρουργικές επεμβάσεις, συμβουλές για την καλύτερη συμβίωση με τα κατοικίδια μας κ.α. Ακόμα, τα μέσα κοινωνικής δικτύωσης είναι η πρώτη επιλογή στην οποία στρέφονται οι ιδιοκτήτες κατοικίδιων όταν αυτό χάνεται. Η αναζήτηση βοήθειας πλέον επεκτείνεται και στο διαδίκτυο, δεν περιορίζεται στην αφισκόλληση στη γειτονιά.

Αναρτήσεις φιλοζωικού περιεχομένου γίνονται είτε από προσωπικά προφίλ είτε από φιλοζωικές σελίδες. Μάλιστα, υπάρχουν ειδικά διαμορφωμένες κοινότητες στα κοινωνικά δίκτυα οι οποίες απευθύνονται μόνο σε φιλοζωικό κοινό. Έχουν δημιουργηθεί ακόμα και πιο συγκεκριμένες ομάδες που αφορούν τις υιοθεσίες σκύλων (ή ακόμα πιο συγκεκριμένες – π.χ. υιοθεσίες μικρόσωμων σκύλων), φροντιστές νεογέννητων, βοήθεια των αδέσποτων, συμβουλές περί προβλημάτων συμπεριφοράς των κατοικίδιων και πολλές άλλες. Φιλόζωοι, είτε ανήκουν σε κάποια οργάνωση, είτε όχι ενώνονται για την καλύτερη ζωή των αδέσποτων και των κατοικίδιων ζώων.

#### Τρόπος λειτουργίας και χαρακτηριστικά παραδείγματα αναρτήσεων

Μια πολύ συχνή και απλή ανάρτηση στα κοινωνικά δίκτυα είναι η δήλωση χαμένου ζώου. Ο ιδιοκτήτης του χαμένου κατοικίδιου αναζητά τη βοήθεια του κόσμου για την ανεύρεσή του. Αυτό, το κάνει δημιουργώντας και «ανεβάζοντας» μία ανάρτηση η οποία

έχει (συνήθως) τα εξής χαρακτηριστικά: όνομα, τελευταία γνωστή τοποθεσία, ημερομηνία, τηλέφωνο και φωτογραφία του χαμένου ζώου. Ακόμα, σε αυτή συμπληρώνει κάποιο κείμενο που καταγράφει κάποιες επιπλέον πληροφορίες τις οποίες θεωρεί σημαντικές, όπως για παράδειγμα ότι φοβάται τους ξένους ή ότι φορά περιλαίμιο χρώματος μπλε. Κάτω από αυτή την ανάρτηση, τα μέλη της κοινότητας στην οποία έγινε ή οι (εικονικοί) φίλοι του ατόμου που έκανε την ανάρτηση απαντούν αν το είδαν, μπορούν να αντιδράσουν αλλά και να τη μοιραστούν με άλλα άτομα και σε άλλες κοινότητες.

Άλλη μια συνήθης ανάρτηση είναι αυτή που αφορά την υιοθεσία κάποιου ζώου. Μία τέτοια ανάρτηση γίνεται συνήθως από προφίλ φιλοζωικών οργανώσεων και φιλόζωων εθελοντών. Κατά κανόνα περιέχουν το όνομα του ζώου, το αν είναι στερημένο, έχει εμβολιαστεί, αρκετές φωτογραφίες και συχνά περιέχουν και βίντεο. Σε αυτή τη δημοσίευση, οι ενδιαφερόμενοι μπορούν να σχολιάσουν και οι υπόλοιποι χρήστες να βοηθήσουν με την αναδημοσίευσή της (share) στα προφίλ τους και σε κοινότητες.

Υπάρχουν κι άλλων ειδών αναρτήσεις από τις οποίες αξίζει όμως να αναφέρουμε τη δημοσίευση έκκλησης οικονομικής βοήθειας. Αφορά κάποιο συγκεκριμένο ζώο το οποίο κατά κανόνα χρειάζεται κάποιο σημαντικό ακριβό χειρουργείο. Τέτοιες αναρτήσεις γίνονται στην πλειοψηφία τους από εθελοντές φιλόζωους οι οποίοι έχουν μεγάλο αριθμό αδέσποτων/προς υιοθεσία ζώων και αδυνατούν να καλύψουν μόνοι τους τα έξοδα. Αποτελούνται συνήθως από το όνομα του ζώου, μία φωτογραφία, μία περιγραφή της κατάστασης, το ποσό που χρειάζεται να συγκεντρωθεί και τον τρόπο με τον οποίο οι υπόλοιποι χρήστες μπορούν να αποστείλουν χρήματα. Αναρτώνται συνήθως σε διάφορες φιλοζωικές κοινότητες/σελίδες και στα προσωπικά τους προφίλ όπου οι χρήστες μπορούν να βοηθήσουν επιπλέον των χρημάτων, μοιράζοντας εκ νέου (share) τη δημοσίευση και σχολιάζοντας για να την προωθήσουν.

Ακόμα μία ενέργεια που αναρτάται στα μέσα κοινωνικής δικτύωσης, είναι η «εκδήλωση». Η εκδήλωση είναι ουσιαστικά μία σελίδα αφιερωμένη σε ένα γεγονός το οποίο έχει τόπο, ώρα και κάποιον διοργανωτή. Ο διοργανωτής είναι αυτός που δημιούργησε την εκδήλωση και αυτός είναι σχεδόν πάντα ένα προφίλ φιλοζωικής οργάνωσης. Στις εκδηλώσεις μπορούν οι χρήστες να δηλώσουν ότι ενδιαφέρονται ή ότι θα πάνε. Τέτοιες εκδηλώσεις αφορούν παζάρια με είδη κατοικίδιων, ημέρες υιοθεσίας αλλά και ομαδική φροντίδα και σίτιση ζώων.

## Σχολιασμός

Η χρήση των κοινωνικών δικτύων έχει αποδειχτεί πολύ χρήσιμη και προσφέρει την ευκαιρία στον οποιοδήποτε με ένα απλό προφίλ να βοηθήσει την φιλοζωική κοινότητα.

Πρόκειται για μία μεγάλη κοινότητα με χρήστες από ολόκληρο τον κόσμο. Τα νέα μαθαίνονται γρήγορα και πολλά μάτια βλέπουν τις αναρτήσεις καθημερινά, καθώς πλέον ο μέσος άνθρωπος δεν μπορεί να πάρει τα μάτια του πάνω από τα μέσα κοινωνικής δικτύωσης. Όμως, ένα βασικό μειονέκτημα στις αναρτήσεις φιλοζωικών θεμάτων στα κοινωνικά δίκτυα είναι ο κατακλυσμός των χρηστών από πληροφορίες οι οποίες μπορεί να είναι αδιάφορες προς τον χρήστη καθώς για παράδειγμα να αφορούν ζωά άλλων μακρινών πόλεων. Ακόμα, οι χρήστες που είναι εγγεγραμμένοι σε πολλές φιλοζωικές κοινότητες βλέπουν συνεχώς τις ίδιες αναρτήσεις. Το σημαντικότερο όμως μειονέκτημα της χρήσης των μέσων κοινωνικής δικτύωσης για φιλοζωικούς σκοπούς είναι πως αυτά πλέον είναι πάρα πολλά. Για να μεγιστοποιήσει κάποιος της πιθανότητες ανεύρεσης υιοθεσίας για παράδειγμα πρέπει να κάνει ανάρτηση σε όλα τα κοινωνικά δίκτυα που στα οποία έχει δημιουργήσει προφίλ. Αυτό φυσικά προκαλεί κούραση αλλά και συμφόρηση, καθώς θα λαμβάνει ειδοποιήσεις από όλα.

## **2.2 Εφαρμογές Αποκλειστικά Φιλοζωικού Περιεχομένου**

Παρόλο που τα μέσα κοινωνικής δικτύωσης έχουν αρκετά πλεονεκτήματα, υπάρχουν αρκετοί άνθρωποι που θέλησαν να μειώσουν τα μειονεκτήματά τους, δημιουργώντας εφαρμογές αποκλειστικά φιλοζωικού περιεχομένου. Παρακάτω θα παραθέσουμε και θα δώσουμε τον τρόπο λειτουργίας μερικών τέτοιων εφαρμογών και θα αναλύσουμε τα πλεονεκτήματα και μειονεκτήματά τους, προσπαθώντας να αντλήσουμε ιδέες για τη δική μας εφαρμογή αλλά παράλληλα να προσέξουμε τι να αποφύγουμε.

### **2.2.1 Dogs' Voice**

Το Dogs' Voice[1] είναι ένας μη κερδοσκοπικός οργανισμός που ξεκίνησε τη λειτουργία του τον Ιούνιο του 2015. Στόχος του είναι να αναδείξει το έργο των φιλοζωικών οργανώσεων που δραστηριοποιούνται ενεργά σε όλη την Ελλάδα, να προσελκύσει οργανωμένα και αξιόπιστα εθελοντές και οικονομική ενίσχυση για τις δράσεις τους και να αποτελέσει την ψηφιακή πλατφόρμα αποτελεσματικής προώθησης αδέσποτων σκύλων σε υπεύθυνες υιοθεσίες.

Το Dogs' Voice δίνει τη δυνατότητα σε φιλοζωικούς οργανισμούς να δημιουργήσουν δικό τους προφίλ και λογαριασμό μέσω του οποίου κάθε οργανισμός θα μπορεί να:

- Καταχωρεί τους σκύλους που προωθεί σε υπεύθυνες υιοθεσίες με πλήρη στοιχεία.
- Αναγράφει τις ανάγκες του σε είδη, τρόφιμα, φάρμακα.

- Ενημερώνει για τις δράσεις των εθελοντών του οργανισμού και το έργο που παράγουν.
- Παρέχει τα στοιχεία επικοινωνίας του οργανισμού.
- Δέχεται δωρεές από ιδιώτες και εταιρείες για την ενίσχυση του φιλοζωικού έργου.

Παράλληλα και ιδιώτες μπορούν να δημιουργήσουν λογαριασμό με τον οποίο μπορούν και αυτοί να καταχωρήσουν σκύλους προς υιοθεσία αλλά και επίσης μπορούν να βρουν εύκολα και γρήγορα, αξιοποιώντας την αναζήτηση τεσσάρων φίλτρων (α. Περιοχή β. Ηλικία γ. Μέγεθος δ. Φύλο) τους σκύλους που προωθούνται σε υιοθεσίες από άλλους χρήστες και οι οποίοι πληρούν τα κριτήρια που ταιριάζουν στον τρόπο ζωής τους. Ο χρήστης έχει διαθέσιμα όλα τα στοιχεία που αφορούν τον σκύλο που τον ενδιαφέρει και τους τρόπους με τους οποίους μπορεί να επικοινωνήσει με την αρμόδια φιλοζωική ή τον ιδιώτη που έκανε την καταχώρηση.

### Σχολιασμός

Το Dogs' Voice αποτελεί ίσως τη πιο γνωστή διαδικτυακή εφαρμογή στην Ελλάδα με αποκλειστικά φιλοζωικό περιεχόμενο. Μέλη της είναι 112 φιλοζωικές οργανώσεις από όλη την Ελλάδα και καθημερινά έχει εκατοντάδες ενεργές αγγελίες υιοθεσίας, από τις οποίες συνολικά από την αρχή της πλατφόρμας έχουν ολοκληρωθεί επιτυχώς 3.192. Στα πλεονεκτήματα βρίσκεται η άμεση και απλή αναζήτηση αγγελιών υιοθεσίας καθώς και η εύκολη εύρεση φιλοζωικών οργανισμών με βάση τη περιοχή που βρίσκονται. Οι φιλοζωικοί οργανισμοί που επιλέγουν να χρησιμοποιήσουν την πλατφόρμα Dogs' Voice για τις ανάγκες τους έχουν προσωπικές σελίδες, στις οποίες μπορούν να παραθέτουν τις πληροφορίες επικοινωνίας τους και παράλληλα τις ανάγκες τους.

Στα αρνητικά της πλατφόρμας είναι η μικρή ποικιλία από είδη αγγελιών, καθώς όλες αφορούν υιοθεσία και δεν υπάρχει υποστήριξη για άλλα είδη αγγελιών. Επίσης άλλη μια προφανής έλλειψη της εφαρμογής είναι η μη ύπαρξη περιεχομένου που να αφορά γάτες ή και άλλα είδη κατοικίδιων. Τέλος, αξίζει να σημειωθεί ως αρνητικό η έλλειψη οποιασδήποτε ειδοποίησης και ενημέρωσης των χρηστών για νέες εξελίξεις στην πλατφόρμα.

### **2.2.2 Mytracknet**

Η Mytracknet[2] είναι μια Startup με φιλοζωικό χαρακτήρα που προσπαθεί να βοηθήσει στο πρόβλημα των χαμένων κατοικίδιων, βασισμένη στην αλληλοβοήθεια και την αλληλεγγύη των τοπικών κοινωνιών. Προσπαθεί να φτιάξει ένα δίκτυο εύρεσης, που ενώνει τους χρήστες για καλό σκοπό και ταυτόχρονα τους ανταμείβει με πόντους που αποτυπώνουν τη συνεισφορά τους στο δίκτυο.

Το δίκτυο λειτουργεί με δυο τρόπους:

- Χειροκίνητες αναζητήσεις: κοινοποιώντας μια φωτογραφία και την τοποθεσία που χάθηκε το κατοικίδιο. Η Mytracknet ειδοποιεί κάθε χρήστη της εφαρμογής που βρίσκεται σε κοντινή περιοχή.
- Με οποιοδήποτε Bluetooth Tracker, που μπορεί να φοράει το κατοικίδιο στο κολάρο του, και έτσι η διαδικασία γίνεται πια αυτόματα με τα κινητά των χρηστών να αναζητούν τον χαμένο κατοικίδιο λαμβάνοντας το σήμα Bluetooth.

### Σχολιασμός

Η εφαρμογή Mytracknet αποτελεί μια αξιόλογη προσπάθεια επίλυσης του προβλήματος χαμένων κατοικίδιων και επιχειρεί να ενημερώσει περισσότερο κόσμο πιο εύκολα αξιοποιώντας μια αποτελεσματικότερη λύση από την ενημέρωση της γειτονίας με αφισκοκόλληση. Στα πλεονεκτήματα ανήκει η ιδιότητα ενημέρωσης των πλησιέστερων χρηστών, γεγονός που ενισχύει τους κόπους εύρεσης χαμένων κατοικίδιων. Μια πολύ ξεχωριστή δυνατότητα της εφαρμογής που αποτελεί σημαντικό πλεονέκτημα είναι η χρήση Bluetooth Tracker για την αυτόματη ανανέωση τοποθεσίας του χαμένου κατοικίδιου.

Ως αρνητικά της εφαρμογής μπορούμε να θεωρήσουμε ελλείψεις σε διαφορετικές ενέργειες φιλοζωικού σκοπού, όπως για παράδειγμα δημοσιεύσεις με σκοπό τη βοήθεια αδέσποτων. Επίσης ως αρνητικό μπορούμε να θεωρήσουμε την απουσία φιλοζωικών οργανισμών από την εφαρμογή και γενικότερη έλλειψη αισθήματος κοινότητας. Τελικά η υπερβολική ώθηση προς gamification και συστήματος ανταμοιβών και πόντων καθιστά την εφαρμογή περίπλοκη και θολώνει τον φιλοζωικό σκοπό.

### **2.2.3 Spot a Stray**

Ο κύριος στόχος της εφαρμογής Spot a Stray[3] είναι η καταγραφή όλων των αδέσποτων σκύλων στην Ελλάδα που θα οδηγήσει μέσω της κατάλληλης διαδικασίας στην οριστική απομάκρυνσή τους από το δρόμο. Σκοπεύει να συνεισφέρει στον αγώνα που κάνουν Μ.Κ.Ο., φιλοζωικές οργανώσεις και κτηνίατροι.

Η εφαρμογή επιτρέπει στον χρήστη να:

- Δημοσιεύσει τη φωτογραφία ενός αδέσποτου σκύλου που είδε στο δρόμο, να προσθέσει τα χαρακτηριστικά του καθώς και να επικοινωνήσει με άλλους χρήστες μέσω σχολίων.
- Περιηγηθεί στο δυναμικό χάρτη του Spot a Stray για να βρει αδέσποτα στην περιοχή του (ή και σε οποιοδήποτε μέρος της Ελλάδας) μέσω διαφόρων φίλτρων (μέγεθος, ράτσα, χρώμα, φύλο).
- Έχει άμεση πρόσβαση στα στοιχεία επικοινωνίας των κοντινότερων κλινικών ζώων, κτηνιατρείων, φιλοζωικών οργανώσεων και αρμόδιων υπηρεσιών των Δήμων.
- Λαμβάνει ειδοποιήσεις για αδέσποτα σκυλιά στην περιοχή του.

## Σχολιασμός

Η εφαρμογή Spot a Stray επικεντρώνεται στην επίλυση του προβλήματος των αδέσποτων σκυλιών, μέσω της συγκεκριμενοποίησης της τοποθεσίας τους και παράλληλα ενημέρωσης των αρμόδιων φορέων. Βασιζόμενη στη συνεχώς επεκτεινόμενη κοινότητά της, καλύπτει μεγάλο έδαφος στις δημοσιεύσεις αδέσποτων σε μεγάλο τμήμα της Ελλάδας. Έχει καταφέρει να δημιουργήσει ένα πολύ εύκολο τρόπο δημιουργίας δημοσίευσης, με κομψή εμφάνιση που επιτρέπει στους χρήστες να δημοσιεύουν τα αδέσποτα που βρίσκουν γρήγορα και χωρίς καμία δυσκολία. Αξίζει να σημειωθεί ακόμα η αποτελεσματική χρήση του χάρτη εντός της εφαρμογής που επιτρέπει στον χρήστη να δει ποιες περιοχές έχουν αδέσποτα και χρειάζονται τη βοήθεια του. Τέλος στα προτερήματα εντάσσεται και η δυνατότητα λήψης ειδοποιήσεων για αδέσποτα κοντά στην περιοχή του χρήστη.

Όσον αφορά τα αρνητικά, μπορούμε να κάνουμε λόγο για την κακή απόδοση της εφαρμογής και τους αργούς χρόνους φόρτωσης. Επίσης, σημαντικό μειονέκτημα είναι η έλλειψη υποστήριξης άλλων ζώων καθώς οι δημιουργοί της εφαρμογής θέτουν ξεκάθαρα πως προορίζεται μόνο για αδέσποτους σκύλους. Τέλος, έλλειψη της εφαρμογής μπορεί να θεωρηθεί η αποκλειστικότητα δημοσιεύσεων αδέσποτων ζώων γεγονός που περιορίζει τη χρησιμότητα της.

### **2.2.4 Your Pet**

Η εφαρμογή Your Pet[4] απευθύνεται σε όλη τη φιλοζωική κοινότητα και αποσκοπεί στην διευκόλυνση του έργου κάθε ανθρώπου που είναι μέλος της, ώστε η ποιότητα ζωής των κατοικίδιων και αδέσποτων να βελτιωθεί αποτελεσματικά. Οι χρήστες της εφαρμογής μπορούν να:

- Ενημερώνονται από ειδικούς για θέματα που αφορούν στα ζώα
- Αναζητούν ζώα για υιοθεσία από καταφύγια ή ιδιώτες
- Διευκολύνουν τη διαδικασία αναζήτησης εξαφανισμένου ζώου με άμεση ειδοποίηση των άλλων χρηστών της εφαρμογής
- Βρίσκουν κτηνίατρους, κλινικές και pet shops κοντά τους, όπου κι αν βρίσκονται
- Προμηθεύονται μέσω e-shop προϊόντα φιλικά προς το περιβάλλον, με μέρος των εσόδων να διατίθεται σε φιλοζωικές δράσεις.

## Σχολιασμός

Η συγκεκριμένη εφαρμογή προσεγγίζει αρκετά τον στόχο της δικιάς μας εφαρμογής και αποτελεί τη καλύτερη πηγή έμπνευσης και παράδειγμα. Αξιοσημείωτο είναι πως συνδυάζει δημοσιεύσεις και για υιοθεσίες αλλά και για αναζήτηση εξαφανισμένων

ζώων, γεγονός που την καθιστά πολλαπλών χρήσεων και αυξάνει το κοινό που μπορεί να προσεγγίσει. Αξιοποιεί ειδοποιήσεις για την ενημέρωση των πλησιέστερων χρηστών και ο χρήστης μπορεί να επιλέξει την απόσταση που επιθυμεί να λαμβάνει ειδοποιήσεις. Ακόμα αξίζει να ειπωθεί πως επιτρέπει στους χρήστες να χρησιμοποιήσουν φίλτρα στην αναζήτησή τους για το ζώο που θέλουν υιοθετήσουν. Καταληκτικά, σημαντικό πλεονέκτημά είναι η ύπαρξη σελίδας αφιερωμένη σε δημοσιεύσεις που έχει πραγματοποιήσει ο χρήστης, όπου και μπορεί να ανατρέξει σε όποια επιθυμεί άμεσα και εύκολα.

Το σημαντικότερο αρνητικό, από το οποίο πηγάζουν και όλα τα υπόλοιπα, είναι πως δεν υπάρχουν σελίδες προφίλ άλλων χρηστών. Λόγο της έλλειψης προφίλ, είναι άγνωστο με ποιον επικοινωνείς ανά πάσα στιγμή, κάτι το οποίο μπορεί να θεωρηθεί επικίνδυνο. Ακόμα δεν προωθούνται φιλοζωικές ενέργειες μέσα από την εφαρμογή, γεγονός που αποδίδεται στην έλλειψη προφίλ φιλοζωικών οργανισμών. Συνολικά από την εφαρμογή λείπει η αίσθηση κοινότητας.

## 2.3 Rescue

Το Rescue, το οποίο αποτελεί και το αντικείμενο της διπλωματικής, είναι μια εφαρμογή που θα προσπαθήσει να ενώσει το φιλοζωικό κοινό δημιουργώντας μια οικουμενική πλατφόρμα που θα προωθεί το φιλοζωικό έργο. Για να πετύχει το στόχο της η εφαρμογή πρέπει να καλύπτει τις ανάγκες κάθε πιθανού χρήστη, δηλαδή να παρέχει σε κάθε ενδιαφερόμενο φιλόζωο, εθελοντή, και οργανισμό τη δυνατότητα πραγματοποίησης των σκοπών του. Όπως είδαμε στις παρόμοιες εφαρμογές που εξετάσαμε, παρατηρήσαμε πως συνήθως επικεντρώνονται στην επίλυση ενός πολύ συγκεκριμένου προβλήματος και συχνά αγνοούσαν υπόλοιπες φιλοζωικές ενέργειες, διαιρώντας έτσι τη φιλοζωική κοινότητα σε μικρότερα μέρη. Πρωταρχικό μέλημα της εφαρμογής λοιπόν, είναι να ενώσει όλες τις πράξεις φιλοζωίας σε ένα μέρος. Για αρχή πρέπει να παρέχει λειτουργίες για τα διαφορετικά είδη χρήστη, ιδιώτη και φιλοζωικό οργανισμό έτσι ώστε να μπορούν να πραγματοποιήσουν το έργο τους.

Στην πλατφόρμα θα υποστηρίζονται κάθε είδος πιθανής δημοσίευσης, από εγκλήσεις βοήθειας για χαμένα ζώα, σε αγγελίες υιοθεσίας, ενημερώσεις για αδέσποτα και ακόμα δημοσιεύσεις γενικού σκοπού με στόχο την κάλυψη όλων των πιθανών σεναρίων χρήσης. Ακόμα, για την αποτελεσματικότερη διεκπεραίωση του φιλοζωικού έργου των οργανισμών θα δίνεται η δυνατότητα δημιουργίας εκδηλώσεων με στόχο την ενημέρωση και συσπείρωση της κοινότητας. Λαμβάνοντας υπόψη τα χαρακτηριστικά των παραπάνω εφαρμογών μπορούμε να καταλήξουμε στο συμπέρασμα πως για την



εφαρμογή μας είναι απαραίτητες λειτουργίες χάρτη με γεωγραφικές τοποθεσίες για τις δημοσιεύσεις καθώς και το φιλτράρισμα βάση κριτηρίων των δημοσιεύσεων. Ακόμα, σημαντικό ρόλο παίζουν οι ειδοποιήσεις καθώς όπως είδαμε δεν απουσιάζουν σχεδόν από καμία εφαρμογή. Τελικό χαρακτηριστικό της εφαρμογής θα είναι η ύπαρξη κοινότητας και η δυνατότητα σύνδεσης μεταξύ των μελών.

## Κεφάλαιο 3. Ανάλυση Απαιτήσεων

Στο κεφάλαιο αυτό θα δοθούν οι ιστορίες χρήστη που αφορούν την εφαρμογή σε μορφή πινάκων καθώς και οι περιπτώσεις χρήσης της. Θα γίνει διαχωρισμός μεταξύ των ενεργειών που αφορούν το προφίλ του χρήστη, τις δημοσιεύσεις της εφαρμογής και την κοινότητα που αποτελείται από τους υπολοίπους χρήστες.

### 3.1 Ιστορίες Χρήστη

Οι ιστορίες χρήστη αποτελούν άτυπες περιγραφές των χαρακτηριστικών της εφαρμογής και των δυνατοτήτων της σε φυσική γλώσσα. Γράφονται από την πλευρά του χρήστη της εφαρμογής σε μορφή καρτών.

Κλίμακα Σπουδαιότητας:

1. Απαιτήσεις που σε κάθε περίπτωση πρέπει να τηρηθούν
2. Απαιτήσεις που είναι εξαιρετικά επιθυμητές αλλά όχι αναγκαίες

#### 3.1.1 Διαχείριση Λογαριασμού

Ιστορία Χρήστη	Ως [τύπος χρήστη]	θέλω να [ενέργεια]	ώστε να [στόχος].	Σπουδαιότητα
1.1	απλός χρήστης / φιλοζωικός οργανισμός	δημιουργήσω προσωπικό λογαριασμό στην εφαρμογή	χρησιμοποιώ την εφαρμογή με τα δικά μου στοιχεία	1
1.2	απλός χρήστης / φιλοζωικός οργανισμός	συνδεθώ στην εφαρμογή με τον λογαριασμό που δημιούργησα	έχω πρόσβαση σε όλες τις λειτουργίες της εφαρμογής	1

1.3	απλός χρήστης / φιλοζωικός οργανισμός	λαμβάνω ειδοποιήσεις για οποιαδήποτε νέα δραστηριότητα στην εφαρμογή	δράσω όσο το δυνατόν νωρίτερα	1
1.4	φιλοζωικός οργανισμός	μπορώ να αλλάξω τα στοιχεία του προφίλ μου	ενημερώνω τους τρόπους επικοινωνίας καθώς και τις ανάγκες του οργανισμού	2
1.5	απλός χρήστης / φιλοζωικός οργανισμός	ενημερώνομαι απλά για νέες ειδοποιήσεις	μην χρειάζεται να ψάχνω για αυτές	2
1.6	απλός χρήστης / φιλοζωικός οργανισμός	δείξω ότι διάβασα την ειδοποίηση που έλαβα	μην ενημερωθώ ξανά για αυτήν	2
1.7	απλός χρήστης / φιλοζωικός οργανισμός	διαγράψω τις ειδοποιήσεις που έλαβα	για να οργανώσω καλύτερα τις ειδοποιήσεις μου	2
1.8	απλός χρήστης / φιλοζωικός οργανισμός	μην χρειάζεται να δημιουργήσω λογαριασμό	μπορώ να δω το περιεχόμενο της εφαρμογής	2

### 3.1.2 Διαχείριση Δημοσιεύσεων

Ιστορία Χρήστη	Ως [τύπος χρήστη]	θέλω να [ενέργεια]	ώστε να [στόχος].	Σπουδαιότητα
2.1	απλός χρήστης	δημοσιεύσω ότι χάθηκε το	ζητήσω βοήθεια και έτσι να	1

		κατοικίδιο μου	αυξήσω τις πιθανότητες να βρεθεί	
2.2	απλός χρήστης	δημοσιεύσω ότι βρήκα κάποιο ζώο στο δρόμο	ενημερώσω τους υπολοίπους χρήστες και φιλοζωικές	1
2.3	φιλοζωικός οργανισμός	δημοσιεύσω ζώα που βρίσκονται στο καταφύγιο μου και ψάχνουν για σπίτι	ενημερώσω όποιον σκέφτεται να υιοθετήσει	1
2.4	φιλοζωικός οργανισμός	οργανώσω ενέργειες εθελοντισμού	ενισχύσω την ανταπόκριση σε περιπτώσεις ανάγκης	1
2.5	απλός χρήστης / φιλοζωικός οργανισμός	αναρτήσω δημοσιεύσεις χωρίς κατηγορία	επικοινωνήσω με την κοινότητα για γενικούς λόγους	1
2.6	απλός χρήστης / φιλοζωικός οργανισμός	αναρτήσω φωτογραφίες στις δημοσιεύσεις που θα κάνω	βοηθήσω να καταλάβουν καλύτερα για τι ακριβώς πρόκειται η δημοσίευση	1
2.7	απλός χρήστης / φιλοζωικός οργανισμός	επιλέξω την τοποθεσία της δημοσίευσης που θα κάνω	δηλώσω την πραγματική τοποθεσία που αφορά η δημοσίευση	1
2.8	απλός χρήστης	επιλέγω την περιοχή στην οποία βρίσκομαι	βλέπω μόνο τις αναρτήσεις που είναι κοντά μου.	1

2.9	απλός χρήστης	φιλτράρω τις πληροφορίες των δημοσιεύσεων	για να δω μόνο αυτές που τηρούν ορισμένα κριτήρια μου	1
2.10	απλός χρήστης	ταξινομήσω τις δημοσιεύσεις με βάση την απόσταση	βλέπω πρώτες αυτές που είναι πιο κοντά σε εμένα	1
2.11	απλός χρήστης	επικοινωνήσω με κάποιον άλλον χρήστη	μιλήσουμε σχετικά με κάποια δημοσίευση του	1
2.12	απλός χρήστης / φιλοζωικός οργανισμός	δω μαζεμένες όλες τις δημοσιεύσεις μου	ανατρέξω σε όποια δημοσίευση επιθυμώ	1
2.13	απλός χρήστης / φιλοζωικός οργανισμός	διαγράψω κάποια δημοσίευση μου	μην την βλέπουν οι υπόλοιποι χρήστες	1
2.14	απλός χρήστης	αναφέρω τι ενέργειες έκανα με το αδέσποτο που βρήκα στον δρόμο	ενημερώσω τους υπόλοιπους χρήστες για την κατάσταση του ζώου	2
2.15	απλός χρήστης / φιλοζωικός οργανισμός	δω μια προεπισκόπηση για κάθε ανάρτηση	αποφασίσω αν με ενδιαφέρει να μάθω περισσότερα	2
2.16	απλός χρήστης / φιλοζωικός οργανισμός	μπορώ να επιλέξω αν θα παραβρεθώ σε κάποια εκδήλωση	ενημερώσω τους διοργανωτές	2

2.17	φιλοζωικός οργανισμός	δω ποιοι θα παραβρεθούν στην εκδήλωση που διοργάνωσα	γνωρίζω πόσοι θα συμμετέχουν	2
------	-----------------------	--	------------------------------	---

### 3.1.3 Κοινότητα

Ιστορία Χρήστη	Ως [τύπος χρήστη]	θέλω να [ενέργεια]	ώστε να [στόχος].	Σπουδαιότητα
3.1	απλός χρήστης	επισκεφτώ την σελίδα προφίλ κάποιου άλλου χρήστη	μάθω περισσότερες πληροφορίες για αυτόν	1
3.2	απλός χρήστης	επισκεφτώ την σελίδα προφίλ κάποιου άλλου χρήστη	δω όλες τις δημοσιεύσεις που έχει κάνει	1
3.3	απλός χρήστης	επισκεφτώ την σελίδα προφίλ κάποιου φιλοζωικού οργανισμού	δω όλα τα ζώα που έχει διαθέσιμα για υιοθεσία	1
3.4	απλός χρήστης	συνδεθώ με άλλους χρήστες	ειδοποιούμε όταν έχουν νέες δημοσιεύσεις	1
3.5	απλός χρήστης	επιλέξω αν θέλω να συνδεθώ με κάποιον άλλο χρήστη	συνδεθώ μόνο με αυτούς που με ενδιαφέρει	1
3.6	απλός χρήστης	διαγράψω την σύνδεση μου με κάποιον άλλο χρήστη	σταματήσω να είμαι συνδεδεμένος με αυτόν οπότε επιθυμώ	1
3.7	απλός χρήστης	προσκαλέσω	μεγαλώσει η	1

	/ φιλοζωικός οργανισμός	γνωστούς και φίλους μου στην εφαρμογή	φιλοζωική κοινότητα	
3.8	απλός χρήστης  / φιλοζωικός οργανισμός	δω μαζεμένους όλους τους χρήστες που έχω συνδεθεί	ανατρέξω σε οποίο λογαριασμό επιθυμώ	1
3.9	απλός χρήστης	δω όλους του υπόλοιπους χρήστες	γνωρίζω ποια είναι τα μέλη της κοινότητας	2
3.10	απλός χρήστης	ψάξω ονομαστικά ανάμεσα σε όλους τους χρήστες	βρω τον χρήστη που ψάχνω	2

## 3.2 Περιπτώσεις Χρήσης

Οι Περιπτώσεις Χρήσης, αφορούν σύνολα διαδοχικών ενεργειών που προσδιορίζουν τη συμπεριφορά του συστήματος και τις λειτουργικές του απαιτήσεις. Αποτελούν μια πιο λεπτομερή προσέγγιση των ιστοριών χρήστη. Κάθε περίπτωση χρήσης πρέπει να διαθέτει τουλάχιστον έναν Δράστη, δηλαδή κάποιον που παίζει ένα ρόλο και αλληλοεπιδρά με την εφαρμογή με τον τρόπο που ορίζει το περιεχόμενο της περίπτωσης χρήστης.

### 3.2.1 Διαχείριση Λογαριασμού

#### ΔΗΜΙΟΥΡΓΙΑ ΠΡΟΣΩΠΙΚΟΥ ΛΟΓΑΡΙΑΣΜΟΥ

##### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 1

##### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Δημιουργία Προσωπικού Λογαριασμού' επιτρέπει στον χρήστη να δημιουργήσει προσωπικό λογαριασμό στην εφαρμογή τον οποίο θα μπορεί να χρησιμοποιεί για να πραγματοποιήσει τις περισσότερες λειτουργίες μέσα στην εφαρμογή.

##### ΔΡΑΣΤΗΣ

Ο χρήστης.

## ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Register'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης μπορεί να επιλέξει το είδος του λογαριασμού που επιθυμεί να δημιουργήσει.
  - 2.1. Εάν ο λογαριασμός πρόκειται για ιδιώτη τότε χρειάζεται να πατήσει το κουμπί 'Individual', εάν πρόκειται για φιλοζωικό οργανισμό τότε το κουμπί 'Animal Welfare Organization'.
  - 2.2. Ύστερα θα μεταφερθεί σε μια σελίδα όπου θα του ζητηθεί να συμπληρώσει τα στοιχεία του.
3. Αφού έχει συμπληρώσει όλα τα απαραίτητα στοιχεία μπορεί να πατήσει το κουμπί 'Sign Up' και να ολοκληρώσει την εγγραφή.

## ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Σε περίπτωση που επιλέξει να δημιουργήσει λογαριασμό για φιλοζωικό οργανισμό τότε κατά την συμπλήρωση στοιχείων θα μπορεί να συμπληρώσει το λογότυπο του οργανισμού καθώς και την τοποθεσία του οργανισμού.

Εάν δεν συμπληρωθούν όλα τα απαραίτητα πεδία τότε ο χρήστης θα πρέπει να οδηγείται σε σελίδα όπου του φανερώνεται το σφάλμα του.

## ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Ο λογαριασμός έχει δημιουργηθεί και έχει σταλεί email με σύνδεσμο για την ενεργοποίηση του λογαριασμού.

## ΣΥΝΔΕΣΗ ΣΤΗΝ ΕΦΑΡΜΟΓΗ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 1.2

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Σύνδεση Στην Εφαρμογή' επιτρέπει στον χρήστη συνδεθεί στην εφαρμογή με τον λογαριασμό του και να έχει πρόσβαση σε όλες τις λειτουργίες της.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να έχει δημιουργήσει λογαριασμό στην εφαρμογή και να τον έχει ενεργοποιήσει.

## ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ



1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Login'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης πρέπει να συμπληρώσει τα στοιχεία 'Email' και 'Password'.
3. Αφού έχει συμπληρώσει τα πεδία μπορεί να πατήσει το κουμπί 'Sign In' και να πραγματοποιήσει την σύνδεση.

### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν είναι λανθασμένος ο συνδυασμός 'Email' και 'Password' ή δεν συμπληρωθούν και τα δυο πεδία τότε ο χρήστης θα πρέπει να οδηγείται σε σελίδα όπου του φανερώνεται το σφάλμα του.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Ο χρήστης έχει τώρα συνδεθεί στην εφαρμογή και έχει πρόσβαση σε όλες τις λειτουργίες της.

---

## ΛΗΨΗ ΕΙΔΟΠΟΙΗΣΕΩΝ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 1.3

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Λήψη Ειδοποιήσεων' επιτρέπει στον χρήστη να λαμβάνει ειδοποιήσεις σχετικές με εξελίξεις μέσα στην εφαρμογή.

#### ΔΡΑΣΤΗΣ

Η εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις συμβεί κάποια ενέργεια μέσα στην εφαρμογή από την οποία αποστέλλονται ειδοποιήσεις.
2. Ειδοποιήσεις στέλνονται από αιτήματα σύνδεσης, τις απαντήσεις σε αυτά τα αιτήματα και την εξέλιξη δημιουργίας λογαριασμού από πρόσκληση σε αυτούς που αφορούν.
3. Ειδοποιήσεις στέλνονται ακόμα όταν αναρτιούνται νέες δημοσιεύσεις στις συνδέσεις του χρήστη που ανάρτησε τη δημοσίευση και σε χρήστες που έχουν επιλέξει την τοποθεσία τους και η δημοσίευση βρίσκεται μέσα στην εμβέλεια που έχουν ορίσει.
4. Ειδοποιήσεις στέλνονται επίσης στον ιδιοκτήτη της δημοσίευσης όταν υπάρχουν νέα σχόλια σε αυτή καθώς και στους χρήστες που έχουν ήδη σχολιάσει για να ειδοποιηθούν για νέες εξελίξεις.

## ΑΛΛΑΓΗ ΣΤΟΙΧΕΙΩΝ ΠΡΟΣΩΠΙΚΟΥ ΠΡΟΦΙΛ

---

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 1.4

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Αλλαγή Στοιχείων Προσωπικού Προφίλ' επιτρέπει στον χρήστη να επεξεργαστεί τα στοιχεία που φαίνονται στην σελίδα προφίλ του.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης βρεθεί στην σελίδα προφίλ του.
  - 1.1. Αρχικά πρέπει να πατήσει στο κουμπί 'Account'.
  - 1.2. Από το μενού που θα εμφανιστεί πρέπει να πατήσει στην κατηγορία 'Profile'.
2. Η σελίδα που θα εμφανιστεί αποτελεί το προφίλ του χρήστη.
3. Τώρα ο χρήστης μπορεί να επεξεργαστεί τα επιθυμητά πεδία του λογαριασμού του.
4. Μόλις ολοκληρώσει την επιθυμητή επεξεργασία πρέπει να πατήσει το κουμπί 'Update' για να οριστικοποιήσει τις αλλαγές.

#### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Το προφίλ του χρήστη έχει ανανεωθεί με τα νέα στοιχεία.

## ΕΝΗΜΕΡΩΣΗ ΝΕΩΝ ΕΙΔΟΠΟΙΗΣΕΩΝ

---

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 1.5

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Ενημέρωση Νέων Ειδοποιήσεων' επιτρέπει στον χρήστη να ενημερωθεί με εύκολο τρόπο για το εάν έχει νέες ειδοποιήσεις.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

Πρέπει να έχει νέες, μη διαβασμένες ειδοποιήσεις.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Account'
2. Στο μενού που θα εμφανιστεί υπάρχει η επιλογή 'Notifications'
3. Εάν υπάρχουν νέες ειδοποιήσεις τότε στην επιλογή 'Notifications' θα υπάρχει μια μικρή κόκκινη σήμανση.

### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν δεν υπάρχουν αδιάβαστες ειδοποιήσεις τότε δεν χρειάζεται να φαίνεται η κόκκινη σήμανση.

---

## ΔΙΑΒΑΣΜΑ ΕΙΔΟΠΟΙΗΣΗΣ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 1.6

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Διάβασμα Ειδοποίησης' επιτρέπει στον χρήστη να ορίσει μια οποιαδήποτε ειδοποίηση ως διαβασμένη.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης βρεθεί στην σελίδα των ειδοποιήσεων του.
  - 1.1. Αρχικά πρέπει να πατήσει στο κουμπί 'Account'.
  - 1.2. Από το μενού που θα εμφανιστεί πρέπει να πατήσει στην κατηγορία 'Notifications'.
2. Στη σελίδα που θα εμφανιστεί θα υπάρχουν όλες οι ειδοποιήσεις του χρήστη.
3. Μπορεί να πατήσει το κουμπί 'Read' που αντιστοιχεί σε μια ειδοποίηση για να την ορίσει ως διαβασμένη.

### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Οι ειδοποιήσεις που δεν έχουν διαβαστεί θα ξεχωρίζονται από μια μπλε σήμανση.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η ειδοποίηση έχει διαβαστεί και πλέον δεν φέρει τη μπλε σήμανση.

---

## ΔΙΑΓΡΑΦΗ ΕΙΔΟΠΟΙΗΣΗΣ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 1.7

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Διαγραφή Ειδοποίησης' επιτρέπει στον χρήστη να διαγράψει μια οποιαδήποτε ειδοποίηση που έχει λάβει.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης βρεθεί στην σελίδα των ειδοποιήσεων του.
  - 1.1. Αρχικά πρέπει να πατήσει στο κουμπί 'Account'.
  - 1.2. Από το μενού που θα εμφανιστεί πρέπει να πατήσει στην κατηγορία 'Notifications'.
2. Στη σελίδα που θα εμφανιστεί θα υπάρχουν όλες οι ειδοποιήσεις του χρήστη.
3. Μπορεί να πατήσει το κουμπί 'Delete' που αντιστοιχεί σε μια ειδοποίηση για να την διαγράψει.

#### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η ειδοποίηση έχει διαγραφεί και δεν εμφανίζεται πια στη σελίδα των ειδοποιήσεων.

### **3.2.2 Διαχείριση Δημοσιεύσεων**

---

## ΑΝΑΡΤΗΣΗ ΧΑΜΕΝΟΥ ΖΩΟΥ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.1

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Ανάρτηση Χαμένου Ζώου' επιτρέπει στον χρήστη να δημιουργήσει μια δημοσίευση στην οποία θα υπάρχει πληροφορία σχετική με ένα ζώο που είναι χαμένο.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Add Post'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης πρέπει να επιλέξει την κατηγορία 'Missing Post'.
3. Στην σελίδα που θα εμφανιστεί ο χρήστης μπορεί να συμπληρώσει όλα τα πεδία που επιθυμεί.
4. Αφού έχει συμπληρώσει όλα τα επιθυμητά στοιχεία μπορεί να πατήσει το κουμπί 'Post' και να αναρτήσει τη δημοσίευση.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν δεν συμπληρωθούν όλα τα απαραίτητα πεδία τότε ο χρήστης θα πρέπει να οδηγείται σε σελίδα όπου του φανερώνεται το σφάλμα του.

#### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η δημοσίευση έχει αναρτηθεί και τώρα μπορούν οι υπόλοιποι χρήστες να την δουν.

## ΑΝΑΡΤΗΣΗ ΑΔΕΣΠΟΤΟΥ ΖΩΟΥ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.2

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Ανάρτηση Χαμένου Ζώου' επιτρέπει στον χρήστη να δημιουργήσει μια δημοσίευση στην οποία θα υπάρχει πληροφορία σχετική με ένα ζώο που είδε στο δρόμο.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Add Post'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης πρέπει να επιλέξει την κατηγορία 'Stray Post'.
3. Στην σελίδα που θα εμφανιστεί ο χρήστης μπορεί να συμπληρώσει όλα τα πεδία που επιθυμεί.
4. Αφού έχει συμπληρώσει όλα τα απαραίτητα στοιχεία μπορεί να πατήσει το κουμπί 'Post' και να αναρτήσει τη δημοσίευση.

### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν δεν συμπληρωθούν όλα τα απαραίτητα πεδία τότε ο χρήστης θα πρέπει να οδηγείται σε σελίδα όπου του φανερώνεται το σφάλμα του.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η δημοσίευση έχει αναρτηθεί και τώρα μπορούν οι υπόλοιποι χρήστες να την δουν.

---

## ΑΝΑΡΤΗΣΗ ΖΩΟΥ ΠΡΟΣ ΥΙΟΘΕΣΙΑ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.3

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Ανάρτηση Ζώου Προς Υιοθεσία' επιτρέπει στον χρήστη να δημιουργήσει μια δημοσίευση στην οποία θα υπάρχει πληροφορία σχετική με ένα ζώο που είναι διαθέσιμο για υιοθεσία.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Add Post'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης πρέπει να επιλέξει την κατηγορία 'Adoption Post'.
3. Στην σελίδα που θα εμφανιστεί ο χρήστης μπορεί να συμπληρώσει όλα τα πεδία που επιθυμεί.

4. Αφού έχει συμπληρώσει όλα τα απαραίτητα στοιχεία μπορεί να πατήσει το κουμπί 'Post' και να αναρτήσει τη δημοσίευση.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν δεν συμπληρωθούν όλα τα απαραίτητα πεδία τότε ο χρήστης θα πρέπει να οδηγείται σε σελίδα όπου του φανερώνεται το σφάλμα του.

#### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η δημοσίευση έχει αναρτηθεί και τώρα μπορούν οι υπόλοιποι χρήστες να την δουν.

---

## ΑΝΑΡΤΗΣΗ ΕΚΔΗΛΩΣΗΣ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.4

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Ανάρτηση Εκδήλωσης' επιτρέπει στον χρήστη να δημιουργήσει μια δημοσίευση στην οποία θα υπάρχει πληροφορία σχετική με μια εκδήλωση που διοργανώνει.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει από την αρχική σελίδα της εφαρμογής.
  - 1.1. Ο χρήστης πρέπει να επιλέξει την καρτέλα 'Events'
  - 1.2. Από τη σελίδα που θα εμφανιστεί πρέπει να πατήσει το κουμπί 'Create New Event'
2. Στην σελίδα που θα εμφανιστεί ο χρήστης μπορεί να συμπληρώσει όλα τα πεδία που επιθυμεί.
3. Αφού έχει συμπληρώσει όλα τα απαραίτητα στοιχεία μπορεί να πατήσει το κουμπί 'Post' και να αναρτήσει την εκδήλωση.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν δεν συμπληρωθούν όλα τα απαραίτητα πεδία τότε ο χρήστης θα πρέπει να οδηγείται σε σελίδα όπου του φανερώνεται το σφάλμα του.

## ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η εκδήλωση έχει αναρτηθεί και τώρα μπορούν οι υπόλοιποι χρήστες να την δουν.

---

## ΑΝΑΡΤΗΣΗ ΓΕΝΙΚΟΥ ΣΚΟΠΟΥ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.5

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Ανάρτηση Γενικού Σκοπού' επιτρέπει στον χρήστη να δημιουργήσει μια δημοσίευση στην οποία θα μπορεί να μιλήσει για ότι θέμα επιθυμεί.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Add Post'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης πρέπει να επιλέξει την κατηγορία 'Simple Post'.
3. Στην σελίδα που θα εμφανιστεί ο χρήστης μπορεί να συμπληρώσει όλα τα πεδία που επιθυμεί.
4. Αφού έχει συμπληρώσει όλα τα απαραίτητα στοιχεία μπορεί να πατήσει το κουμπί 'Post' και να αναρτήσει τη δημοσίευση.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν δεν συμπληρωθούν όλα τα απαραίτητα πεδία τότε ο χρήστης θα πρέπει να οδηγείται σε σελίδα όπου του φανερώνεται το σφάλμα του.

## ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η δημοσίευση έχει αναρτηθεί και τώρα μπορούν οι υπόλοιποι χρήστες να την δουν.

---

## ΑΝΑΡΤΗΣΗ ΦΩΤΟΓΡΑΦΙΩΝ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.6

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ



Το use case 'Ανάρτηση Φωτογραφιών' επιτρέπει στον χρήστη να προσθέτει φωτογραφίες στις δημοσιεύσεις που μπορεί να κάνει.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει κατά τη διάρκεια δημιουργίας δημοσίευσης οποιασδήποτε κατηγορίας.
2. Στη σελίδα όπου θα συμπληρώνει τα πεδία της δημοσίευσης, θα έχει τη δυνατότητα να προσθέσει και φωτογραφίες πατώντας το κουμπί 'Browse'.
  - 2.1. Πατώντας το κουμπί 'Browse' θα παρουσιαστεί ένα παράθυρο file explorer.
  - 2.2. Ο χρήστης θα μπορεί να επιλέξει τις φωτογραφίες που επιθυμεί ψάχνοντας μέσα στο σύστημα του.
3. Αφού επιλέξει τις φωτογραφίες που επιθυμεί, θα του εμφανίζεται προεπισκόπηση για κάθε φωτογραφία.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Στην κατηγορία εκδηλώσεων ο χρήστης θα έχει τη δυνατότητα να προσθέσει μια μόνο φωτογραφία.

#### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Οι δημοσιεύσεις θα περιλαμβάνουν και τις φωτογραφίες που επέλεξε ο κάθε χρήστης.

---

## ΑΝΑΡΤΗΣΗ ΤΟΠΟΘΕΣΙΑΣ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.7

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Ανάρτηση Τοποθεσίας' επιτρέπει στον χρήστη να επιλέξει τη γεωγραφική τοποθεσία που αφορά τη δημοσίευση του.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει κατά τη διάρκεια δημιουργίας δημοσίευσης οποιασδήποτε κατηγορίας.
2. Στη σελίδα όπου θα συμπληρώνει τα πεδία της δημοσίευσης, θα υπάρχει ένας χάρτης με ένα πεδίο αναζήτησης.
3. Ο χρήστης μπορεί να επιλέξει την τοποθεσία που επιθυμεί αναζητώντας την ονομασία στο πεδίο αναζήτησης.
4. Μπορεί να οριστικοποιήσει την τοποθεσία που επέλεξε πατώντας το κουμπί 'Set Location'.
5. Η τοποθεσία που επιλέχθηκε θα εμφανίζεται κάτω από τον χάρτη σε μορφή διεύθυνσης.

### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Ο χρήστης θα μπορεί να επιλέξει και τοποθεσία πατώντας πάνω στον χάρτη.

Οι δημοσιεύσεις γενικής κατηγορίας δεν θα υποστηρίζουν τοποθεσία.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η δημοσίευση τώρα περιλαμβάνει και γεωγραφική τοποθεσία

## ΕΠΙΛΟΓΗ ΤΟΠΟΘΕΣΙΑΣ ΧΡΗΣΤΗ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.8

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Επιλογή Τοποθεσίας Χρήστη' επιτρέπει στον χρήστη να επιλέξει την τοποθεσία καθώς και μια εμβέλεια μέσα της οποίας θα βρίσκονται οι δημοσιεύσεις που θα βλέπει και θα λαμβάνει ειδοποιήσεις.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Near Me'.

2. Στη σελίδα που θα εμφανιστεί θα υπάρχει ένας χάρτης με ένα πεδίο αναζήτησης.
3. Ο χρήστης μπορεί να επιλέξει την τοποθεσία που επιθυμεί αναζητώντας την ονομασία στο πεδίο αναζήτησης.
4. Μπορεί να οριστικοποιήσει την τοποθεσία που επέλεξε πατώντας το κουμπί 'Set Location'.
5. Η τοποθεσία που επιλεγθεί θα εμφανίζεται κάτω από τον χάρτη.
6. Θα υπάρχει ακόμα μια μπάρα όπου ο χρήστης θα μπορεί να επιλέξει την εμβέλεια σε μέτρα.
7. Αφού κάνει τις επιλογές που επιθυμεί μπορεί να οριστικοποιήσει την τοποθεσία του πατώντας το κουμπί 'Update'.

### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Ο χρήστης θα μπορεί να επιλέξει και τοποθεσία πατώντας πάνω στον χάρτη.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Έχει οριστεί η τοποθεσία του χρήστη και τώρα θα μπορεί να δει στην αρχική σελίδα μόνο τις κοντινές δημοσιεύσεις, δηλαδή αυτές που τηρούν το κριτήριο εμβέλειας που έθεσε.

### ΕΙΔΙΚΕΣ ΑΠΑΙΤΗΣΕΙΣ, ΘΕΜΑΤΑ, ΚΙΝΔΥΝΟΙ ΚΑΙ ΑΛΛΑ ΣΧΟΛΙΑ:

Μόνο οι δημοσιεύσεις που έχουν τοποθεσία επηρεάζονται από την εμβέλεια, οι δημοσιεύσεις γενικού σκοπού θα εμφανίζονται ανεξαρτήτως στην αρχική σελίδα.

## ΦΙΛΤΡΑΡΙΣΜΑ ΔΗΜΟΣΙΕΥΣΕΩΝ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.9

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Φιλτράρισμα Δημοσιεύσεων' επιτρέπει στον χρήστη να θέσει κριτήρια έτσι ώστε να βλέπει στην αρχική του σελίδα μόνο συγκεκριμένες δημοσιεύσεις.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Filter'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης μπορεί να επιλέξει τα κριτήρια που επιθυμεί.
3. Αφού κάνει την επιλογή του, μπορεί να πατήσει το κουμπί 'Update' και να οριστικοποιήσει την επιλογή του.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η αρχική σελίδα έχει τώρα ανανεωθεί, και εμφανίζονται μόνο οι δημοσιεύσεις που τηρούν τα κριτήρια που τέθηκαν.

## ΤΑΞΙΝΟΜΗΣΗ ΔΗΜΟΣΙΕΥΣΕΩΝ

---

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.10

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Ταξινόμηση Δημοσιεύσεων' επιτρέπει στον χρήστη να ταξινομήσει τις δημοσιεύσεις έτσι ώστε να εμφανιστούν πρώτες οι πιο πρόσφατες δημοσιεύσεις ή οι πιο κοντινές.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Filter'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης πρέπει να επιλέξει την ταξινόμηση που επιθυμεί μεταβάλλοντας τη τιμή του πεδίου 'Sort By'.
3. Αφού κάνει την επιλογή του, μπορεί να πατήσει το κουμπί 'Update' και να οριστικοποιήσει την επιλογή του.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Η προκαθορισμένη ταξινόμηση είναι η ημερομηνία, 'Date (recent first)'.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Οι δημοσιεύσεις θα εμφανίζονται ταξινομημένες στην αρχική σελίδα με βάση το κριτήριο που έθεσε ο χρήστης.

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.11

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Επικοινωνία Με Χρήστη' επιτρέπει στον χρήστη να συνομιλήσει με κάποιον άλλο χρήστη σχετικά με μια δημοσίευση.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει από την σελίδα μιας οποιαδήποτε δημοσίευσης.
  - 1.1. Ο χρήστης πρέπει να πατήσει το κουμπί 'Discussion' για να οδηγηθεί στην σελίδα συνομιλίας για αυτή τη δημοσίευση.
  - 1.2. Από εκεί πρέπει να πατήσει το κουμπί 'Add Comment'.
2. Στη σελίδα που θα εμφανιστεί θα υπάρχει ένα πεδίο υποδοχής κειμένου.
3. Αφού γράψει αυτό που επιθυμεί μπορεί να πατήσει το κουμπί 'Post' για να δημοσιεύσει το κείμενο του.

#### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Το κείμενο έχει προστεθεί στη σελίδα της συζήτησης για τη δημοσίευση.

## ΠΡΟΒΟΛΗ ΠΡΟΣΩΠΙΚΩΝ ΔΗΜΟΣΙΕΥΣΕΩΝ

---

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.12

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Προβολή Προσωπικών Δημοσιεύσεων' επιτρέπει στον χρήστη να δει μαζεμένες τις δημοσιεύσεις του.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

## ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

## ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης βρεθεί στην σελίδα των δημοσιεύσεων του.
  - 1.1. Αρχικά πρέπει να πατήσει στο κουμπί 'Account'.
  - 1.2. Από το μενού που θα εμφανιστεί πρέπει να πατήσει στην κατηγορία 'My Posts'.
2. Η σελίδα που θα εμφανιστεί περιέχει όλες τις δημοσιεύσεις του χρήστη.

## ΔΙΑΓΡΑΦΗ ΔΗΜΟΣΙΕΥΣΗΣ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.13

## ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Διαγραφή Δημοσίευσης' επιτρέπει στον χρήστη να διαγράψει μια οποιαδήποτε δημοσίευση του.

## ΔΡΑΣΤΗΣ

Ο χρήστης.

## ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

Ο χρήστης πρέπει να έχει τουλάχιστον μια δημοσίευση.

## ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης βρεθεί στην σελίδα των δημοσιεύσεων του.
  - 1.1. Αρχικά πρέπει να πατήσει στο κουμπί 'Account'.
  - 1.2. Από το μενού που θα εμφανιστεί πρέπει να πατήσει στην κατηγορία 'My Posts'.
2. Η σελίδα που θα εμφανιστεί περιέχει όλες τις δημοσιεύσεις του χρήστη.
3. Μπορεί να πατήσει το κουμπί 'Delete' που έχει η κάθε δημοσίευση για να διαγράψει την αντίστοιχη δημοσίευση.

## ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Η δημοσίευση έχει πλέον διαγραφτεί και δεν εμφανίζεται στην αρχική σελίδα ούτε στις δημοσιεύσεις του χρήστη.

## ΑΝΑΦΟΡΑ ΕΝΕΡΓΕΙΑΣ ΠΟΥ ΑΦΟΡΑ ΑΔΕΣΠΟΤΟ

---

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.14

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Αναφορά Ενέργειας Που Αφορά Αδέσποτο' επιτρέπει στον χρήστη να αναφέρει τις ενέργειες που έκανε όσον αφορά τη βοήθεια του αδέσποτου ζώου σε αντίστοιχη δημοσίευση.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει κατά τη διάρκεια δημοσίευσης αδέσποτου ζώου.
2. Στη σελίδα αυτή υπάρχει πεδίο με κατηγορία 'Actions Taken'
3. Ο χρήστης μπορεί να επιλέξει ποια ενέργεια ταιριάζει με την ενέργεια που έλαβε.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν δεν ταιριάζει καμία προκαθορισμένη ενέργεια με αυτή που λήφθηκε, θα υπάρχει και πεδίο συμπλήρωσης κειμένου έτσι ώστε ο χρήστης να την περιγράψει.

#### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Οι δημοσιεύσεις αδέσποτου ζώου περιέχουν ενέργειες βοήθειας.

## ΠΡΟΒΟΛΗ ΠΡΟΕΠΙΣΚΟΠΗΣΗΣ ΔΗΜΟΣΙΕΥΣΗΣ

---

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.15

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Προβολή Προεπισκόπησης Δημοσίευσης' επιτρέπει στον χρήστη να δει μερικές από τις πληροφορίες των δημοσιεύσεων στην αρχική σελίδα.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει από την αρχική σελίδα της εφαρμογής.
2. Οι δημοσιεύσεις θα εμφανίζονται η μια κάτω από την άλλη σε μορφή προεπισκόπησης, όπου θα φαίνονται μερικές από τις πληροφορίες.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Κάνοντας click στον τίτλο της δημοσίευσης ή πατώντας στο κουμπί 'Read More' ο χρήστης θα οδηγείται στην σελίδα της δημοσίευσης όπου θα φαίνονται όλες οι πληροφορίες.

---

### ΑΠΑΝΤΗΣΗ ΣΥΜΜΕΤΟΧΗΣ ΕΚΔΗΛΩΣΗΣ

#### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.16

##### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Απάντηση Συμμετοχής Εκδήλωσης' επιτρέπει στον χρήστη να δηλώσει σε δημοσιεύσεις εκδηλώσεων εάν θα παραβρεθεί

##### ΔΡΑΣΤΗΣ

Ο χρήστης.

##### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

##### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης βρεθεί σε σελίδα δημοσίευσης εκδήλωσης.
2. Ο χρήστης μπορεί να πατήσει το κουμπί 'Will Attend' για να δείξει το ενδιαφέρον του.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν ο χρήστης έχει απαντήσει ήδη στην εκδήλωση ότι θα συμμετέχει τότε το κουμπί μετατρέπεται σε 'Will Not Attend' για να αλλάξει εύκολα την απάντηση του.

##### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Ο χρήστης έχει δηλώσει ότι θα παραβρεθεί στην εκδήλωση.

---

### ΠΡΟΒΟΛΗ ΣΥΜΜΕΤΕΧΟΝΤΩΝ ΕΚΔΗΛΩΣΗΣ

#### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 2.17



### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Προβολή Συμμετεχόντων Εκδήλωσης' επιτρέπει στον χρήστη να δει όλους τους χρήστες που έχουν δηλώσει πως θα παραβρεθούν σε μια εκδήλωση.

### ΔΡΑΣΤΗΣ

Ο χρήστης.

### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης βρεθεί σε σελίδα δημοσίευσης εκδήλωσης.
2. Ο χρήστης μπορεί να πατήσει πάνω στη καρτέλα 'See who is attending'.
3. Τώρα μπορεί να δει όσους χρήστες έχουν δηλώσει πως θα παραβρεθούν στην εκδήλωση.

## **3.2.3 Κοινότητα**

### ΕΠΙΣΚΕΨΗ ΣΕΛΙΔΑΣ ΧΡΗΣΤΗ ΚΑΙ ΠΡΟΒΟΛΗ ΠΛΗΡΟΦΟΡΙΩΝ

#### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.1

### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Επίσκεψη Σελίδας Χρήστη Και Προβολή Πληροφοριών' επιτρέπει στον χρήστη να επισκεφτεί τη σελίδα προφίλ κάποιου άλλου χρήστη και να δει τις πληροφορίες που είναι διαθέσιμες.

### ΔΡΑΣΤΗΣ

Ο χρήστης.

### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει τον υπερσύνδεσμο με το όνομα του χρήστη που επιθυμεί να επισκεφτεί το προφίλ.
2. Η σελίδα που θα εμφανιστεί αποτελεί το προφίλ του χρήστη.
3. Ο χρήστης τώρα μπορεί να δει οποιαδήποτε πληροφορία έχει κάνει διαθέσιμη ο άλλος χρήστης.

## ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν η σελίδα προφίλ ανήκει σε φιλοζωικό οργανισμό τότε θα υπάρχουν ακόμα πληροφορίες σχετικές με το φιλοζωικό οργανισμό.

## ΕΠΙΣΚΕΨΗ ΣΕΛΙΔΑΣ ΧΡΗΣΤΗ ΚΑΙ ΠΡΟΒΟΛΗ ΔΗΜΟΣΙΕΥΣΕΩΝ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.2

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Επίσκεψη Σελίδας Χρήστη Και Προβολή Δημοσιεύσεων' επιτρέπει στον χρήστη να επισκεφτεί τη σελίδα προφίλ κάποιου άλλου χρήστη και να δει όλες τις δημοσιεύσεις του.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει τον υπερσύνδεσμο με το όνομα του χρήστη που επιθυμεί να επισκεφτεί το προφίλ.
2. Η σελίδα που θα εμφανιστεί αποτελεί το προφίλ του χρήστη.
3. Κάτω από τις πληροφορίες του χρήστη θα υπάρχει μια ενότητα με δημοσιεύσεις που θα ονομάζεται 'Posts'.
4. Εκεί ο χρήστης μπορεί να δει όλες τις δημοσιεύσεις που έχει κάνει ο άλλος χρήστης.

## ΕΠΙΣΚΕΨΗ ΣΕΛΙΔΑΣ ΟΡΓΑΝΙΣΜΟΥ ΚΑΙ ΠΡΟΒΟΛΗ ΔΗΜΟΣΙΕΥΣΕΩΝ ΥΙΟΘΕΣΙΑΣ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.3

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Επίσκεψη Σελίδας Οργανισμού Και Προβολή Δημοσιεύσεων Υιοθεσίας' επιτρέπει στον χρήστη να επισκεφτεί τη σελίδα προφίλ κάποιου φιλοζωικού οργανισμού και να δει όλες τις δημοσιεύσεις που αφορούν ζώα προς υιοθεσία.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει τον υπερσύνδεσμο με το όνομα του χρήστη που επιθυμεί να επισκεφτεί το προφίλ.
2. Η σελίδα που θα εμφανιστεί αποτελεί το προφίλ του φιλοζωικού οργανισμού.
3. Κάτω από τις πληροφορίες του χρήστη θα υπάρχει μια ενότητα με δημοσιεύσεις που θα ονομάζεται 'Fosters'.
4. Εκεί ο χρήστης μπορεί να δει όλες τις δημοσιεύσεις που αφορούν ζώα προς υιοθεσία που έχει κάνει ο οργανισμός.

---

## ΣΥΝΔΕΣΗ ΜΕΤΑΞΥ ΧΡΗΣΤΩΝ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.4

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Σύνδεση Μεταξύ Χρηστών' επιτρέπει στον χρήστη να στείλει αίτημα φιλίας σε άλλους χρήστες και να ακολουθήσει οργανισμούς με σκοπό να ενημερώνεται για τις νέες δημοσιεύσεις τους.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

Οι χρήστες δεν πρέπει να είναι ήδη συνδεδεμένοι μεταξύ τους.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης επισκεφτεί τη σελίδα προφίλ κάποια άλλου χρήστη.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης μπορεί να πατήσει το κουμπί 'Add Friend' για να στείλει αίτημα φιλίας στον άλλο ιδιώτη.

### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν η σελίδα προφίλ ανήκει σε φιλοζωικό οργανισμό τότε το κουμπί θα λέει 'Follow Organization'.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Έχει γίνει αίτημα φιλίας και έχει σταλεί ειδοποίηση στον χρήστη που αφορά.

### ΕΙΔΙΚΕΣ ΑΠΑΙΤΗΣΕΙΣ, ΘΕΜΑΤΑ, ΚΙΝΔΥΝΟΙ ΚΑΙ ΑΛΛΑ ΣΧΟΛΙΑ:

Οι λογαριασμοί οργανισμών δεν μπορούν να πραγματοποιήσουν αιτήματα σύνδεσης.

## ΑΠΑΝΤΗΣΗ ΑΙΤΗΜΑΤΟΣ ΣΥΝΔΕΣΗΣ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.5

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Απάντηση Αιτήματος Σύνδεσης' επιτρέπει στον χρήστη να επιλέξει εάν επιθυμεί να συνδεθεί με τον χρήστη που του έστειλε αίτημα φιλίας.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει ο χρήστης βρεθεί στην σελίδα συνδέσεων του.
  - 1.1. Αρχικά πρέπει να πατήσει στο κουμπί 'Account'.
  - 1.2. Από το μενού που θα εμφανιστεί πρέπει να πατήσει στην κατηγορία 'Profile'.
  - 1.3. Στην σελίδα που θα εμφανιστεί θα υπάρχει η ενότητα 'Connections' που μπορεί να πατηθεί
2. Στη σελίδα αυτή υπάρχει η ενότητα Friend Requests.
3. Εκεί για κάθε ενεργό αίτημα φιλίας ο χρήστης μπορεί να επιλέξει την επιλογή 'Accept'

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν δεν επιθυμεί να συνδεθεί μαζί με τον χρήστη τότε μπορεί να επιλέξει 'Decline', που θα διαγράψει το αίτημα φιλίας.

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Οι χρήστες είναι τώρα συνδεδεμένοι και μπορούν να λαμβάνουν ειδοποιήσεις ο ένας από τον άλλον.

## ΔΙΑΓΡΑΦΗ ΣΥΝΔΕΣΗΣ

---

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.6

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Διαγραφή Σύνδεσης' επιτρέπει στον χρήστη να διαγράψει τη σύνδεση που έχει με έναν άλλο χρήστη.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

Οι χρήστες πρέπει να είναι συνδεδεμένοι.

#### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης επισκεφτεί τη σελίδα προφίλ κάποιου άλλου χρήστη με τον οποίο είναι συνδεδεμένοι.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης μπορεί να πατήσει το κουμπί 'Remove Friend' για να διαγράψει τη σύνδεση.

#### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Εάν η σελίδα προφίλ ανήκει σε φιλοζωικό οργανισμό τότε το κουμπί θα λέει 'Unfollow Organization'.

#### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Οι χρήστες δεν είναι πλέον συνδεδεμένοι.

## ΠΡΟΣΚΛΗΣΗ ΝΕΟΥ ΧΡΗΣΤΗ

---

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.7

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Πρόσκληση Νέου Χρήστη' επιτρέπει στον χρήστη να προσκαλέσει κάποιον φίλο του στην εφαρμογή.

## ΔΡΑΣΤΗΣ

Ο χρήστης.

## ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

## ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Community'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης πρέπει να πατήσει το κουμπί 'invite your friends'
  - 2.1. Θα εμφανιστεί ένα modal με ένα πεδίο υποδοχής κειμένου με τίτλο email.
  - 2.2. Αφού συμπληρώσει το email του φίλου που επιθυμεί να καλέσει θα πατήσει το κουμπί 'Send Invitation' για να στείλει την πρόσκληση.
3. Ο παραλήπτης του μηνύματος μπορεί να ακολουθήσει τον υπερσύνδεσμο που θα τον οδηγήσει στην σελίδα εγγραφής.

## ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Μόλις ολοκλήρωση την εγγραφή του και ενεργοποιήσει τον λογαριασμό του στην εφαρμογή τότε οι δυο χρήστες θα είναι ήδη συνδεδεμένοι.

## ΕΙΔΙΚΕΣ ΑΠΑΙΤΗΣΕΙΣ, ΘΕΜΑΤΑ, ΚΙΝΔΥΝΟΙ ΚΑΙ ΑΛΛΑ ΣΧΟΛΙΑ:

Κάθε χρήστης κατέχει ένα ξεχωριστό κωδικό 'Referral Token', αυτός ο κωδικός περιέχεται μέσα στον υπερσύνδεσμο της πρόσκλησης για την εγγραφή. Έτσι μπορούν οι δυο χρήστες να συνδεθούν μόλις ολοκληρωθεί η εγγραφή.

## ΠΡΟΒΟΛΗ ΣΥΝΔΕΣΕΩΝ

---

## ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.8

## ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Προβολή Συνδέσεων' επιτρέπει στον χρήστη να δει μαζεμένες τις συνδέσεις τους.

## ΔΡΑΣΤΗΣ

Ο χρήστης.

## ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει ο χρήστης βρεθεί στην σελίδα συνδέσεων του.
  - 1.1. Αρχικά πρέπει να πατήσει στο κουμπί 'Account'.
  - 1.2. Από το μενού που θα εμφανιστεί πρέπει να πατήσει στην κατηγορία 'Profile'.
  - 1.3. Στην σελίδα που θα εμφανιστεί θα υπάρχει η ενότητα 'Connections' που μπορεί να πατηθεί
2. Στη σελίδα που θα εμφανιστεί υπάρχει η υποενότητα 'Friends' και 'Organizations' οι οποίες περιέχουν όλες τις συνδέσεις του χρήστη.

### ΕΠΕΚΤΑΣΕΙΣ / ΕΝΑΛΛΑΓΕΣ

Οι οργανισμοί αντί για τις υποενότητες 'Friends' και 'Organizations' έχουν υποενότητα με τίτλο 'Followers'.

---

## ΠΡΟΒΟΛΗ ΧΡΗΣΤΩΝ ΕΦΑΡΜΟΓΗΣ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.9

#### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Προβολή Χρηστών Εφαρμογής' επιτρέπει στον χρήστη να δει όλους τους χρήστες που έχουν λογαριασμό στην εφαρμογή.

#### ΔΡΑΣΤΗΣ

Ο χρήστης.

#### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει μόλις ο χρήστης πατήσει το κουμπί 'Community'.
2. Στη σελίδα που θα εμφανιστεί ο χρήστης μπορεί να δει μερικούς από τους χρήστες της εφαρμογής, χωρισμένους σε ιδιώτες και φιλοζωικούς οργανισμούς.
3. Πατώντας το κουμπί 'See All' θα δει όλους τους χρήστες της κατηγορίας που ήθελε.

---

## ΑΝΑΖΗΤΗΣΗ ΧΡΗΣΤΗ

### ΑΝΑΓΝΩΡΙΣΤΙΚΟ: ΠΧ 3.10

### ΠΕΡΙΓΡΑΦΗ ΚΑΙ ΣΤΟΧΟΣ

Το use case 'Αναζήτηση Χρήστη' επιτρέπει στον χρήστη να αναζητήσει ονομαστικά κάποιον χρήστη της εφαρμογής.

### ΔΡΑΣΤΗΣ

Ο χρήστης.

### ΠΡΟΫΠΟΘΕΣΕΙΣ

Ο χρήστης πρέπει να είναι συνδεδεμένος με τον λογαριασμό του στην εφαρμογή.

### ΒΑΣΙΚΗ ΡΟΗ ΓΕΓΟΝΟΤΩΝ

1. Το use case ξεκινάει από τη σελίδα όλων των χρηστών στην εφαρμογή, είτε ιδιώτες είτε οργανισμούς.
2. Στο πάνω μέρος της σελίδας θα υπάρχει ένα πεδίο υποδοχής κειμένου με την ένδειξη 'Search'.
3. Ο χρήστης μπορεί να γράψει το όνομα που επιθυμεί και να πατήσει το κουμπί 'Search'

### ΜΕΤΑ-ΣΥΝΘΗΚΕΣ

Θα εμφανιστούν τα πλησιέστερα αποτελέσματα.



## **Κεφάλαιο 4. Σχεδίαση και**

# **Αρχιτεκτονική**

# **Λογισμικού**

Σε αυτό το κεφάλαιο θα περιγραφεί η διαδικασία της σχεδίασης του συστήματος. Αρχικά θα γίνει αναφορά στο σχεδιασμό των βασικών οθονών της εφαρμογής και ύστερα θα αναλυθούν τα βασικά συστατικά του συστήματος, τόσο του Back End όσο και του Front End, και οι σχέσεις που τα συνδέουν μεταξύ τους.

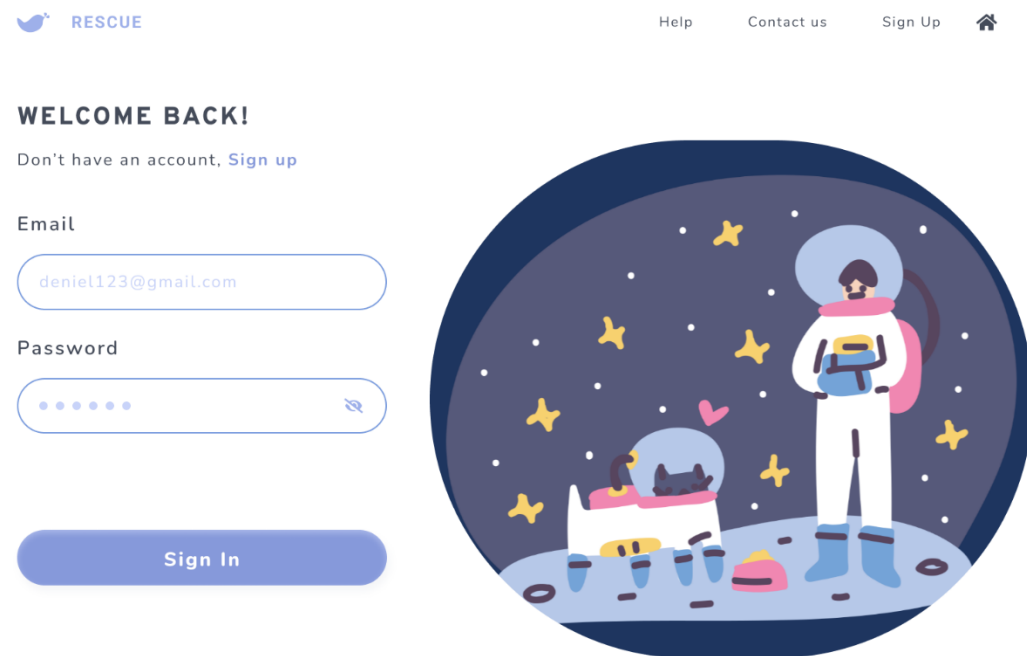
### **4.1 Σχεδίαση Πρωτοτύπου**

Βασικό κομμάτι αυτής της εργασίας ήταν να σχεδιαστούν όλες οι οθόνες της εφαρμογής, για τον σχεδιασμό τους χρησιμοποιήσαμε το λογισμικό Figma[5]. Αυτές σχεδιάστηκαν για την αποδοτικότερη και ευκολότερη διεκπεραίωση της ίδιας της εφαρμογής, καθώς οπτικοποίησαν τις ιστορίες χρήστη και μαζί με αυτό έδωσαν μορφή στο σχήμα της βάσης δεδομένων. Όλες οι οθόνες σχεδιάστηκαν ώστε να είναι λιτές και εύκολα κατανοητές από όλους τους χρήστες, με απαλά χρώματα, ιδανικές γραμματοσειρές και προσεκτικά διαλεγμένα εικονίδια. Στη παρούσα ενότητα θα αναλύσουμε τον σκοπό και τη σημασία κάποιων βασικών αντιπροσωπευτικών οθονών που σχεδιάστηκαν.

#### **4.1.1 Σύνδεση και Εγγραφή**

Το παρακάτω σύνολο οθονών περιγράφει τη διαδικασία σύνδεσης και εγγραφής στην εφαρμογή.

Για τη σύνδεση στην εφαρμογή, ο χρήστης πρέπει να έχει δημιουργήσει ήδη λογαριασμό και να τον έχει ενεργοποιήσει. Ύστερα στην σελίδα σύνδεσης χρειάζεται να συμπληρώσει το email με το οποίο πραγματοποίησε την εγγραφή του και τον κωδικό που επέλεξε. Παρακάτω φαίνεται η σελίδα σύνδεσης στην εφαρμογή.



Εικόνα 4.1 Οθόνη Σύνδεσης

Από την άλλη, για να εγγραφεί στην εφαρμογή, πρέπει αρχικά να επιλέξει το είδος του λογαριασμού που επιθυμεί να φτιάξει και αντιστοιχεί στις ανάγκες του, ιδιώτης ή φιλοζωικός οργανισμός. Έπειτα, ανάλογα με την επιλογή του χρειάζεται να συμπληρώσει την φόρμα εγγραφής, η οποία διαφέρει ανάλογα το είδος του λογαριασμού. Τέλος, μετά την επιτυχή εγγραφή του θα εμφανιστεί η σελίδα επιτυχούς εγγραφής και θα ενημερωθεί για τη διαδικασία ενεργοποίησης του λογαριασμού του.

Παρακάτω φαίνεται η οθόνη εγγραφής ενός φιλοζωικού οργανισμού.

## WELCOME TO RESCUE!

A place to connect, ask and provide help for anything pet related!  
If you already have an account, [click here](#)

Please provide your Organization's information

Name of Animal Welfare Organization

The Invaluables

Email

deniel123@gmail.com

Password

• • • • •



Logo

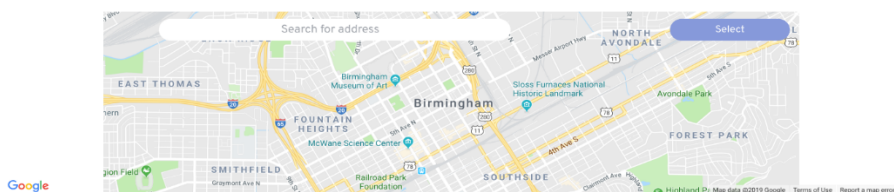
Drag and Drop here

Select file

Description

San

Select Organization Location



Contact Phone

San

Contact email

San

Web Site URL

San

Facebook page link

San

Needs

San

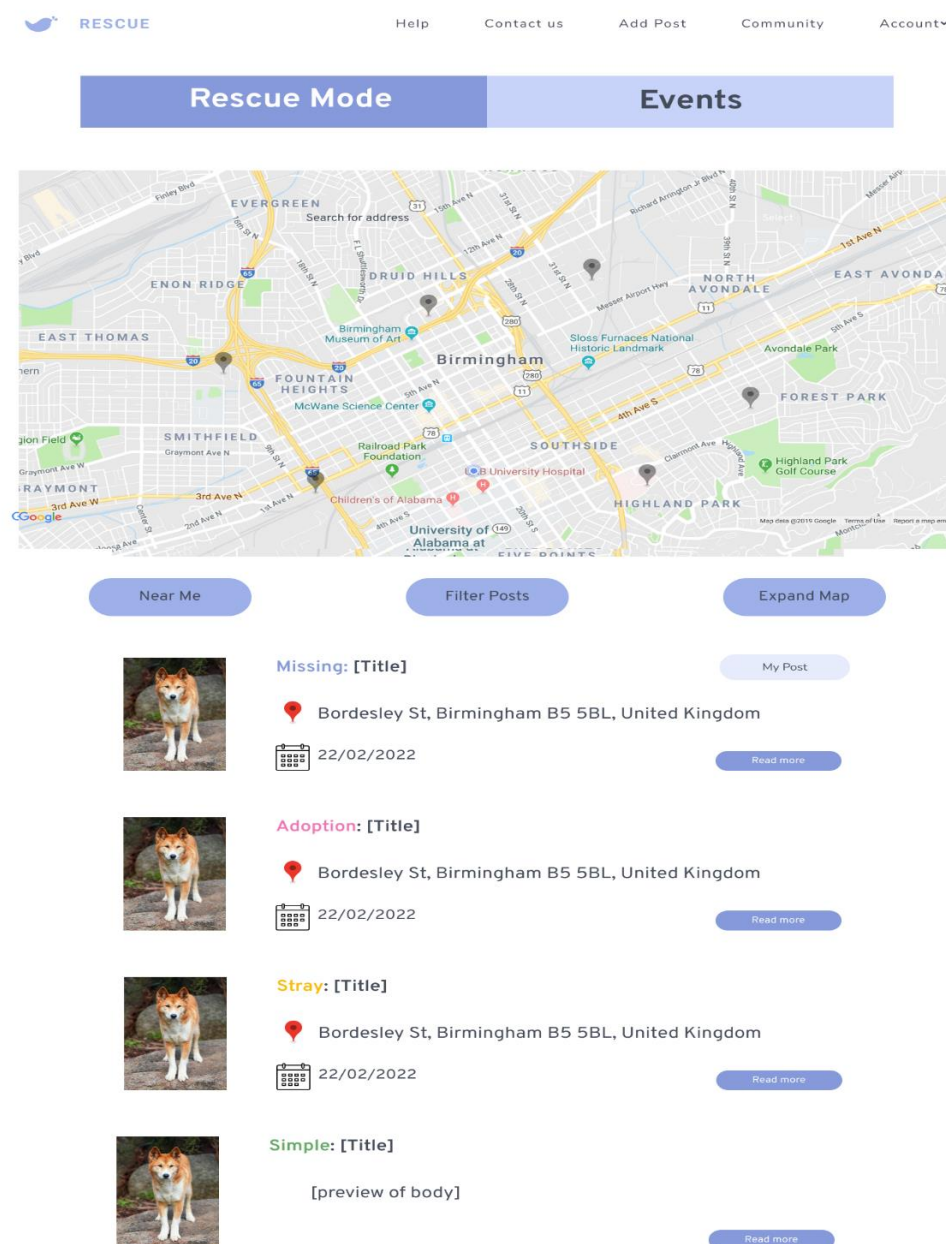
Sign Up

Εικόνα 4.2 Οθόνη Εγγραφής Φιλοζωικού Οργανισμού

### 4.1.2 Κυρίως Εφαρμογή

Το παρακάτω σύνολο οθονών περιγράφει τη διαδικασία χρήσης της εφαρμογής και την αλληλεπίδραση με τις δημοσιεύσεις.

Η κεντρική σελίδα της εφαρμογής χωρίζεται σε δυο μέρη, στο 'Rescue Mode' και στο 'Events'. Στο πρώτο βρίσκονται οι δημοσιεύσεις που αφορούν ζώα που μπορεί να χρειάζονται βοήθεια καθώς και τις δημοσιεύσεις γενικού σκοπού, ενώ στο άλλο βρίσκονται οι προσεχείς εκδηλώσεις. Ο χρήστης μπορεί να περιηγηθεί και να δει τις προεπισκοπήσεις των δημοσιεύσεων και των εκδηλώσεων. Από αυτήν τη σελίδα ο χρήστης μπορεί να επισκεφτεί μια οποιαδήποτε δημοσίευση και έτσι να διαβάσει όλες τις λεπτομέρειες της. Ακόμα από τη σελίδα μιας συγκεκριμένης δημοσίευσης ο χρήστης μπορεί να περιηγηθεί στη σελίδα συζήτησης, μέσα στην οποία μπορεί να δει τα σχόλια που υπάρχουν ήδη αλλά και να προσθέσει ένα δικό του. Παρακάτω φαίνεται η κεντρική οθόνη της εφαρμογής.



Εικόνα 4.3 Κεντρική Οθόνη Εφαρμογής

Από την κεντρική σελίδα της εφαρμογής, ο χρήστης έχει πρόσβαση σε 2 λειτουργίες, 'Near Me' και 'Filter Posts', που του επιτρέπουν να τροποποιήσει την αρχική του σελίδα. Επιλέγοντας συγκεκριμένα κριτήρια για τις δημοσιεύσεις που επιθυμεί να προβάλλονται και καθορίζοντας την απόσταση, από τον ίδιο, που θα βρίσκονται οι δημοσιεύσεις οι οποίες θα του εμφανίζονται. Παρακάτω φαίνονται οι βοηθητικές οθόνες 'Near Me' και 'Filter Posts'

Near Me

Set your location

Search for address

Select

Google

Distance from me

Update

Filter Posts

Sort by: Distance (nearest first)

Date (recent first)

☒ Show post only from friends or following

☒ Hide my posts

Show only

<input type="checkbox"/> Missing	11
<input checked="" type="checkbox"/> Adoption	23
<input type="checkbox"/> Stray	12
<input type="checkbox"/> Other/Uncategorized	12

Posts about

☒ Dogs

☐ Cats

☐ Any

Size

☒ Large

☐ Medium

☐ Small

Gender

☒ Male

☐ Female

Age

☒ <6 months

☐ 6 months to 1 year

☐ 1 year to 2 years


☐ 2 years to 5 years


☐ 5+ years

Εικόνα 4.4 Οθόνες 'Near Me' και 'Filter Posts'

Έπειτα, σειρά έχει η διαδικασία δημιουργίας δημοσίευσης. Η διαδικασία ξεκινάει από την επιλογή 'Add Post' που φαίνεται στο πάνω μέρος της σελίδας, ύστερα ο χρήστης οδηγείται στη σελίδα όπου μπορεί να επιλέξει το είδος της δημοσίευσης που θέλει να αναρτήσει, κάθε είδος συνοδεύεται και από μια σύντομη περιγραφή. Αφού γίνει η

επιλογή, ο χρήστης καλείται να συμπληρώσει μια φόρμα με τις λεπτομέρειες που είναι σημαντικές για κάθε είδος δημοσίευσης. Η παρακάτω οθόνη αφορά τις δημοσιεύσεις που αφορούν αδέσποτα ζώα και βρίσκονται στο δρόμο.

 RESCUE


[Help](#) [Contact us](#) [Add Post](#) [Community](#) [Account](#) 

### New Stray Post



**This is a post about a...**  
☒ Dog ☐ Cat ☐ Other

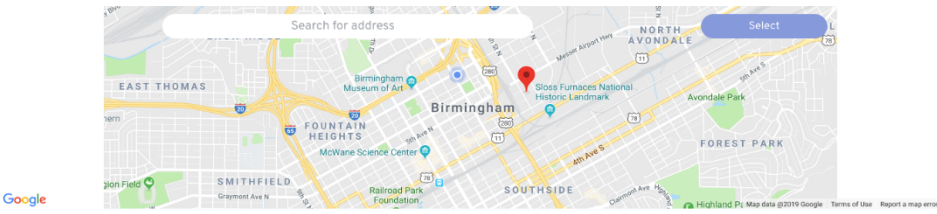
**Title**

**Upload Images**  

 Drag and Drop here

Select file

 drag-drop.jpg (50.7kb) 

**Select Location**  


**Select Date**  

Date  
Oct 12, 2018

October 2021

M	T	W	T	F	S	S
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

**Characteristics**  

Breed: half-bred

**Gender**  
☒ Male  
☐ Female  
☐ Unknown

**Color**  
☐ Black  
☒ Brown  
☐ White

**Size**  
☒ Small  
☐ Medium  
☐ Large

**Actions Taken**  

**Actions**  
☒ No actions taken ☐ Brought it to an animal shelter ☐ Informed the competent municipality / local police ☐ Brought it to a vet or an animal clinic  
☐ Adopted ☐ Other

**Additional Info**

Back

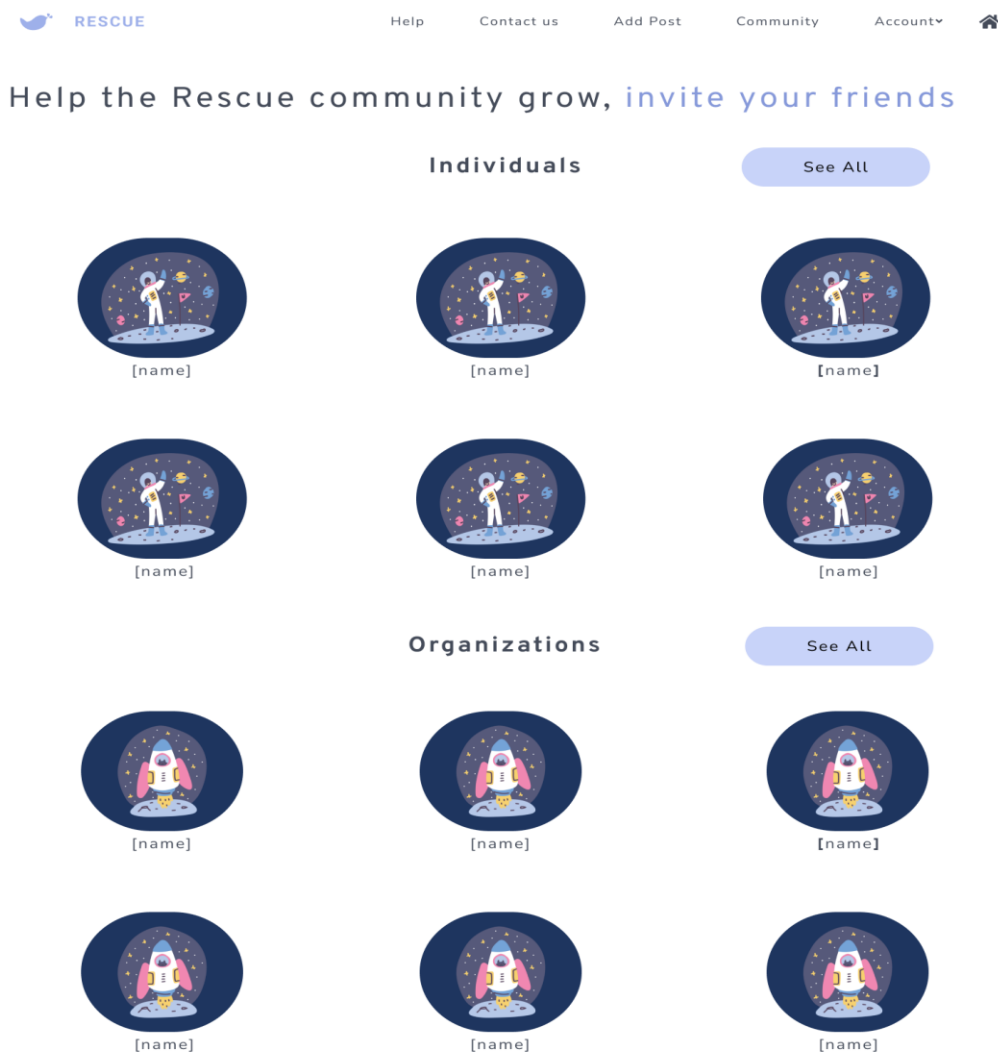
Post

Εικόνα 4.5 Οθόνη Δημιουργίας Δημοσίευσης Αδέσποτου.

### 4.1.3 Κοινότητα

Ένα ακόμα σημαντικό κομμάτι της εφαρμογής, είναι η κοινότητα και παρακάτω φαίνονται οι κύριες οθόνες που σχετίζονται με τις ενέργειες που μπορούν να γίνουν και αφορούν τους υπόλοιπους χρήστες.

Ο χρήστης μπορεί να περιηγηθεί στην κεντρική σελίδα της κοινότητας από την επιλογή 'Community' που φαίνεται στο πάνω μέρος της σελίδας. Η σελίδα αυτή περιέχει μια προεπισκόπηση με μερικούς από τους χρήστες της εφαρμογής, ιδιωτών και φιλοζωικών οργανισμών. Με την επιλογή 'See All' μπορεί να περιηγηθεί σε όλους τους εγγεγραμμένους χρήστες της εφαρμογής και να αναζητήσει ονομαστικά κάποιον χρήστη. Από αυτή την οθόνη επίσης, μπορεί να πραγματοποιηθεί η πρόσκληση φίλου στην εφαρμογή που ξεκινάει τη διαδικασία αποστολής email. Παρακάτω φαίνεται η οθόνη της κοινότητας της εφαρμογής.



Εικόνα 4.6 Οθόνη Κοινότητας Εφαρμογής

Από αυτό το σημείο ο χρήστης μπορεί, επιλέγοντας το όνομα οποιουδήποτε χρήστη επιθυμεί, να επισκεφτεί τη σελίδα προφίλ του εκάστοτε χρήστη. Στη σελίδα προφίλ του κάθε χρήστη βρίσκονται οι πληροφορίες που έχει κάνει διαθέσιμες και ταυτόχρονα υπάρχουν μαζεμένες όλες οι δημοσιεύσεις που έχει κάνει ο χρήστης αυτός. Επιπρόσθετα από αυτή τη σελίδα μπορεί να γίνει αίτημα σύνδεσης μεταξύ των χρηστών. Παρακάτω φαίνεται η οθόνη προφίλ ενός φιλοζωικού οργανισμού στην εφαρμογή.



Εικόνα 4.7 Οθόνη Προφίλ Φιλοζωικού Οργανισμού

#### 4.1.4 Προσωπικό Προφίλ και Λογαριασμός

Το τελευταίο σύνολο από οθόνες που θα περιγράψουμε, είναι οι οθόνες που αφορούν το προσωπικό προφίλ του χρήστη και τη διαχείριση του.

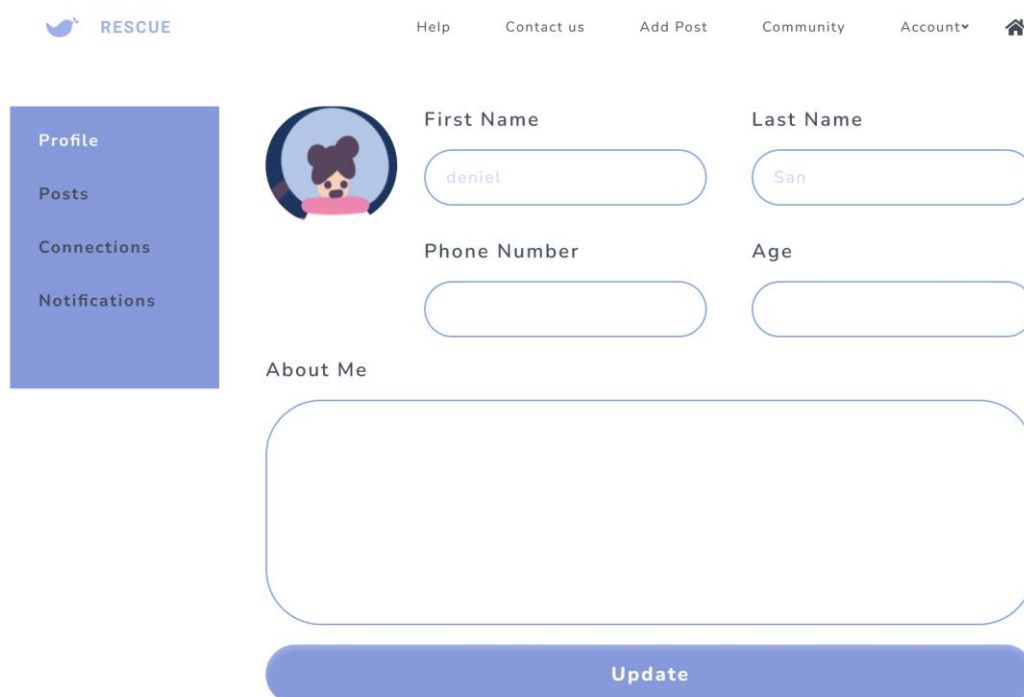
Οι ενέργειες που μπορεί να πραγματοποιήσει ο χρήστης όσον αφορά το προφίλ του είναι οι εξής:

- Προβολή και τροποποίηση των στοιχείων του.
- Προβολή και διαχείριση των δημοσιεύσεων του.
- Προβολή και διαχείριση των συνδέσεων του.
- Προβολή και διαχείριση των ειδοποιήσεων του.

Η παρακάτω οθόνη ανήκει σε λογαριασμό χρήστη που είναι ιδιώτης και αφορά τη σελίδα προβολής και τροποποίησης των στοιχείων του. Από αυτή τη σελίδα μπορεί να



περιηγηθεί και στις υπόλοιπες σελίδες που αφορούν τον λογαριασμό του: 'Posts', 'Connections' και 'Notifications'.



Εικόνα 4.8 Οθόνη Προσωπικής Σελίδας Προφίλ

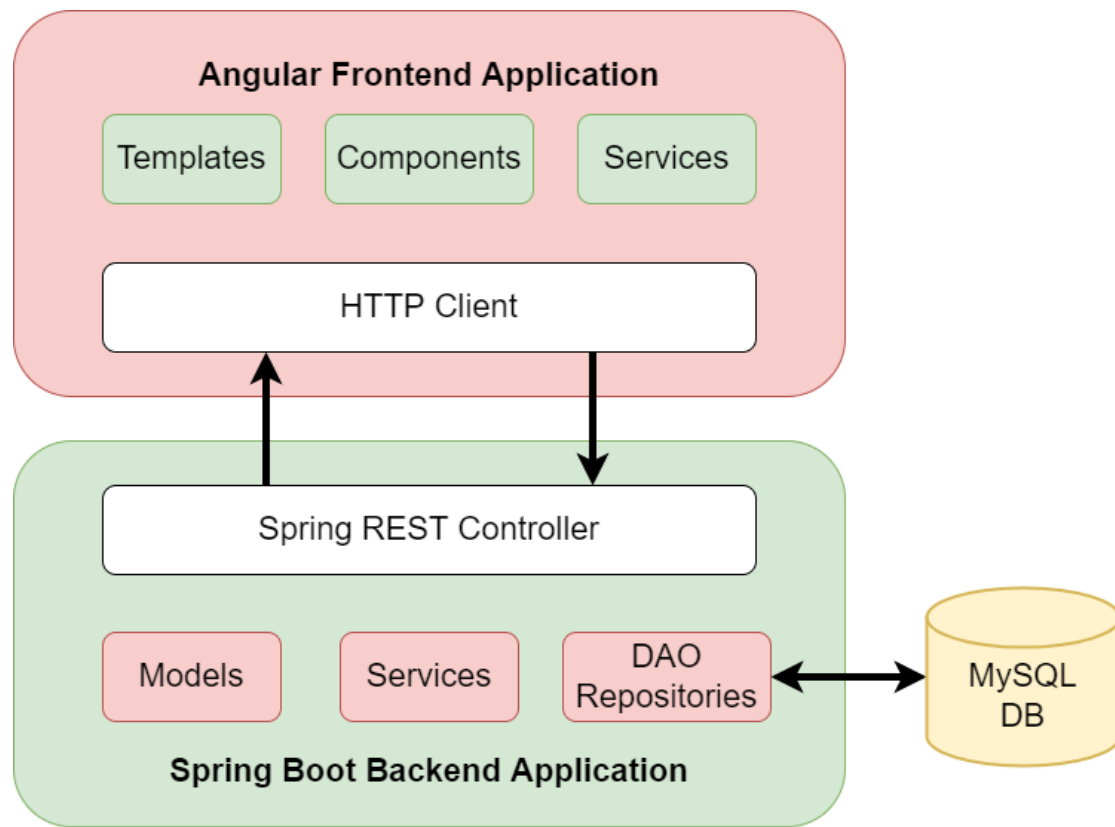
## 4.2 Αρχιτεκτονική Συστήματος

Η ανάπτυξη της εφαρμογής έχει γίνει σε δυο μέρη, Back End και Front End, τα οποία είναι διαθέσιμα ως λογισμικά ανοικτού κώδικα στο GitHub: [Back End](#), [Front End](#).

Το Back End μέρος της εφαρμογής είναι γραμμένο σε Java με τη βοήθεια του πλαισίου Spring Boot[6]. Η αρχιτεκτονική του λογισμικού βασίζεται σε πολυεπίπεδο αρχιτεκτονικό στύλ και στο μοτίβο Model View Controller (MCV)[7] που βασίζεται η Spring. Έτσι όχι μόνο διευκολύνεται η μελλοντική συντήρηση και επέκταση της εφαρμογής, αλλά γίνεται και ξεκάθαρος διαχωρισμός των ευθυνών των διαφορετικών συστατικών που συνεργάζονται για τη λειτουργία της εφαρμογής.

Το Front End μέρος της εφαρμογής είναι γραμμένο σε Typescript με τη βοήθεια του εργαλείου Angular[8]. Κατά τη διάρκεια ανάπτυξης αυτού του μέρους ακολουθήθηκαν συμβάσεις με σκοπό την καλύτερη κατανόηση και ευκολότερη συντήρηση της εφαρμογής. Κυριότερη των οποίων είναι η single responsibility principle[9], βάση αυτής της αρχής κάθε μέρος της εφαρμογής είναι χωρισμένο σε δικό του αρχείο και αναλαμβάνει μια ευθύνη. Συγκεκριμένα, το Front End αποτελείται από Angular templates, components και services τα οποία αλληλοεπιδρούν μεταξύ τους για να

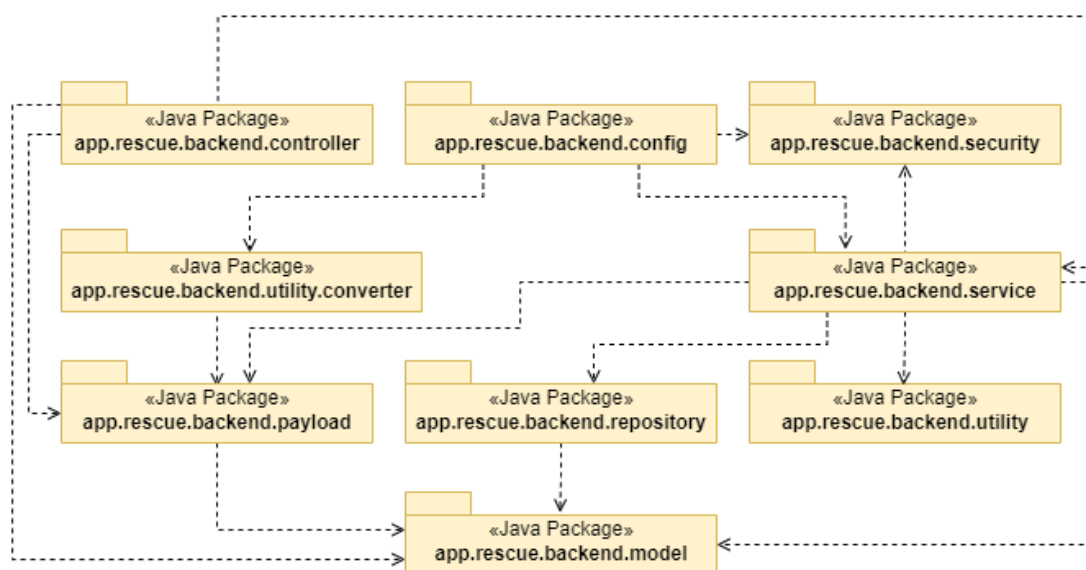
δημιουργήσουν μια ολοκληρωμένη εμπειρία περιήγησης και χρήσης της εφαρμογής από τον κάθε χρήστη. Παρακάτω φαίνεται η συνολική αρχιτεκτονική του συστήματος.



Εικόνα 4.9 Πλήρης Αρχιτεκτονική Συστήματος

#### 4.2.1 Πακέτα Συστήματος Back End

Το Back End αποτελείται από εννιά πακέτα και θα αναλυθούν στη συνέχεια, παρακάτω φαίνονται αυτά τα πακέτα καθώς και οι σχέσεις μεταξύ τους.



Εικόνα 4.10 Διάγραμμα Πακέτων Συστήματος

- **config:** Ένα πακέτο που περιέχει κλάσεις που χειρίζονται τις γενικές ρυθμίσεις της εφαρμογής.
- **controller:** Σε αυτό το πακέτο, περιλαμβάνονται οι κλάσεις που ορίζουν το REST API[10] της εφαρμογής και επιτρέπουν την επικοινωνία μεταξύ του Back End και του Front End.
- **model:** Αυτό το πακέτο αποτελεί τον πυρήνα της εφαρμογής. Οι κλάσεις του ορίζουν την αναπαράσταση των δεδομένων που χειρίζεται η εφαρμογή.
- **payload:** Το πακέτο αυτό μαζί με τις κλάσεις του, χρησιμοποιείται σε συνεργασία με τις κλάσεις του πακέτου controller και επιτρέπει την μεταφορά δεδομένων μεταξύ Back End και Front End.
- **repository:** Αποτελείται από κλάσεις που εκτελούν βασικές λειτουργίες βάσης δεδομένων που αντιστοιχίζουν τα δεδομένα της εφαρμογής.
- **security:** Ένα πακέτο με κλάσεις που επιτρέπουν στην εφαρμογή να χρησιμοποιηθεί με λογαριασμό χρηστών και είναι υπεύθυνο για την αυθεντικοποίηση και εξουσιοδότηση των χρηστών στην εφαρμογή.
- **service:** Οι κλάσεις αυτού του πακέτου αποτελούν τη γέφυρα μεταξύ ενεργειών των κλάσεων του πακέτου controller και αυτών του repository. Στην ουσία είναι το business logic της εφαρμογής και ευθύνονται για την πραγματοποίηση όλων των ενεργειών, περίπλοκων και μη.
- **utility:** Οι κλάσεις αυτού του πακέτου αποτελούν βοηθητικά κομμάτια της εφαρμογής και συνεισφέρουν στην λειτουργία άλλων κλάσεων.
- **utility/converter:** Τέλος, εδώ βρίσκονται κλάσεις που επιτρέπουν την μετατροπή αντικειμένων τύπου Form Data από το Front End σε οντότητες τύπου payload, που θα χρησιμοποιήσουν οι κλάσεις των πακέτων controller και service.

#### 4.2.2 Model

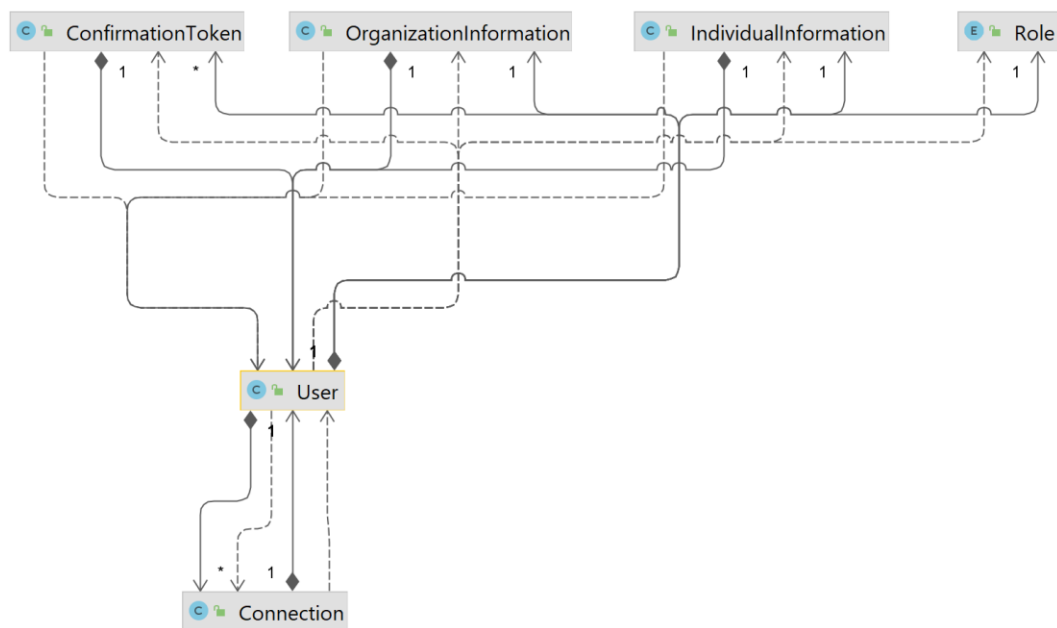
Στο συγκεκριμένο πακέτο περιέχονται οι κλάσεις που ευθύνονται για την αναπαράσταση των οντοτήτων της εφαρμογής και αποτελούν το επίπεδο του πυρήνα της εφαρμογής. Κυριότερη των οποίων αποτελεί η κλάση User, όπως και είναι αναμενόμενο για μια εφαρμογή κοινωνικής δικτύωσης. Μέσω αυτής, προκύπτουν και όλες οι υπόλοιπες οντότητες καθώς για την ύπαρξη τους πρέπει να ανήκουν σε κάποιον. Για παράδειγμα, οι δημοσιεύσεις έχουν τον χρήστη ως αυτόν που τις δημιούργησε και του ανήκουν. Μια άλλη σημαντική οντότητα με αρκετές συσχετίσεις είναι η κλάση Post. Η δημοσίευση αποτελεί την οντότητα με την οποία θα αλληλοεπιδρά ο χρήστης μέσα από την εφαρμογή, για αυτό το λόγο πρέπει να «κρατάει» όλες τις σημαντικές πληροφορίες και να συσχετίζεται με τα υπόλοιπα χαρακτηριστικά των δημοσιεύσεων όπως οι εικόνες και τα σχόλια.

Αξίζει να γίνει λόγος και για τις κλάσεις, Individual Information, Organization Information, Animal Characteristics και Event Properties, αυτές οι οντότητες συμπληρώνουν τις κλάσεις User και Post. Οι πρώτες δυο αφορούν τα στοιχεία του χρήστη ανάλογα με το είδος λογαριασμού που έχει επιλέξει να φτιάξει. Οι επόμενες δυο

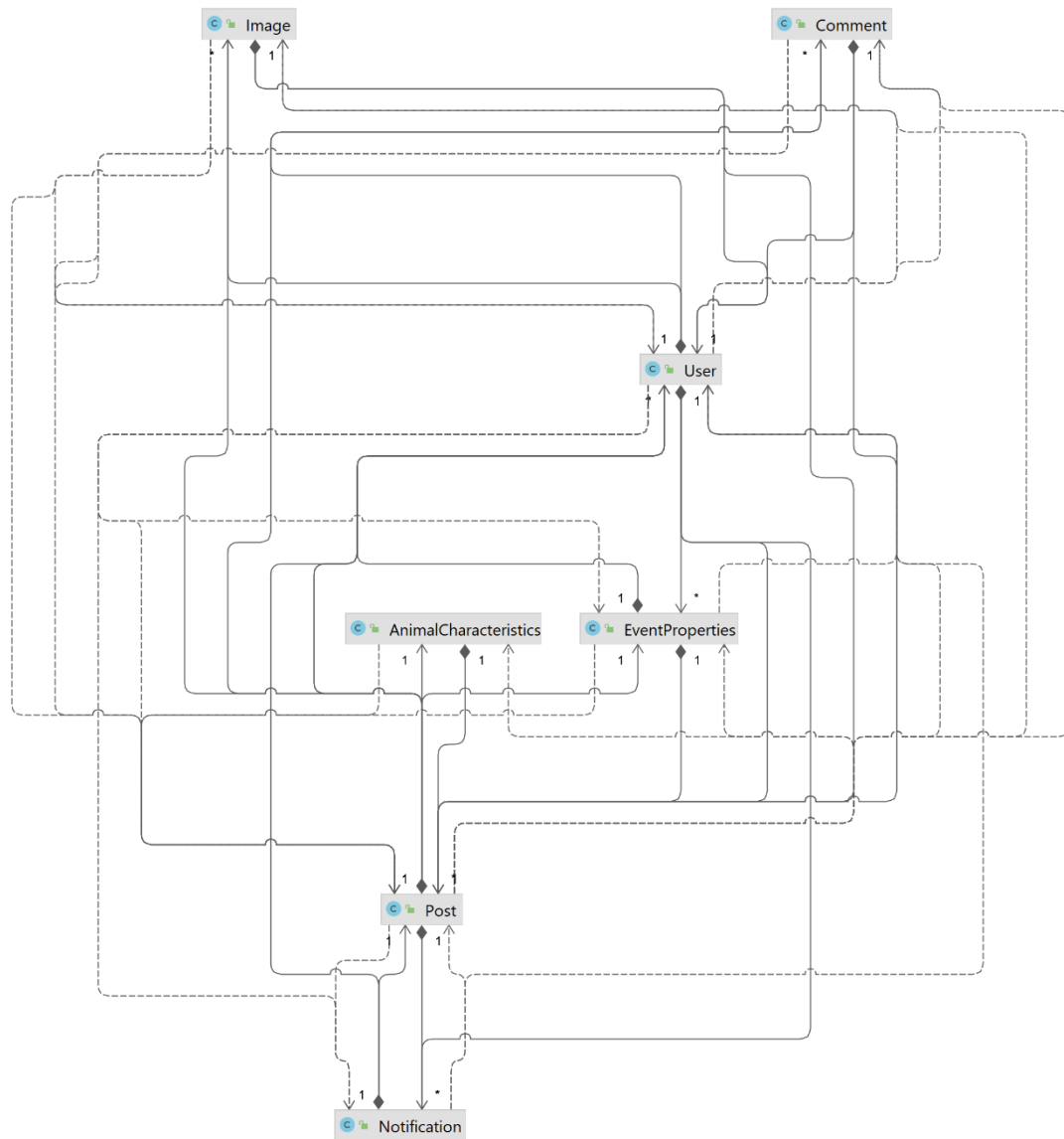
σχετίζονται με τις δημοσιεύσεις και επιτρέπουν τον διαχωρισμό ανάμεσα σε δημοσιεύσεις με περιεχόμενο που αφορά ζώα και στις δημοσιεύσεις εκδηλώσεων.

Τα δυο παρακάτω διαγράμματα αφορούν τις κλάσεις του πακέτου model και απεικονίζουν τις σχέσεις μεταξύ τους. Όπως αναφέραμε παραπάνω οι δυο κυριότερες κλάσεις του συστήματος είναι η User και η Post, για αυτόν το λόγο έχει γίνει ο εξής διαχωρισμός:

1. Κλάσεις που σχετίζονται αποκλειστικά με την κλάση User.
2. Κλάσεις που σχετίζονται με τη κλάση User αλλά και με την κλάση Post.



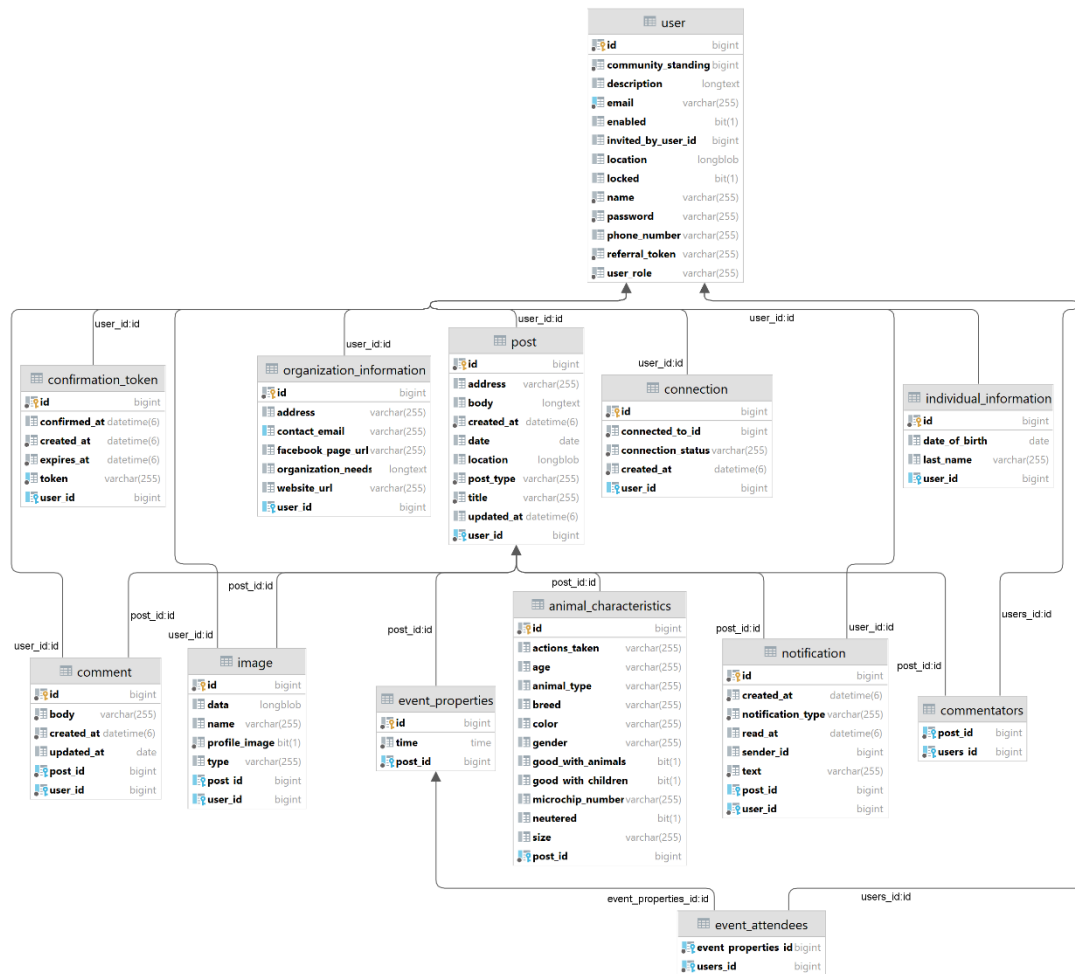
Εικόνα 4.11 Διάγραμμα Κλάσεων Πακέτου model (User)



Εικόνα 4.12 Διάγραμμα Κλάσεων Πακέτου model (User – Post)

Τα δεδομένα είναι οργανωμένα σε μια σχεσιακή βάση δεδομένων (MySQL)[11], με τη βοήθεια της Spring και του Java Persistence Api (JPA)[12]. Έτσι μπορεί να γίνει ευκολά η αντιστοίχιση των κλάσεων που είδαμε παραπάνω σε πίνακες της βάσης δεδομένων.

Παρακάτω φαίνεται αναλυτικά το τελικό σχήμα της βάσης, που περιλαμβάνει λογικούς περιορισμούς, όπως ονόματα πινάκων, πεδία, τύπους δεδομένων και τις σχέσεις μεταξύ αυτών των οντοτήτων.



Εικόνα 4.13 Σχήμα Βάσης Δεδομένων

### 4.2.3 Repository

Αυτό το πακέτο αντιπροσωπεύει ένα ακόμα από τα επίπεδα του λογισμικού, συγκεκριμένα το επίπεδο Data Access Objects (DAO)[13]. Οι κλάσεις που ανήκουν σε αυτό το πακέτο έχουν την αρμοδιότητα να επικοινωνούν με τη βάση δεδομένων και να εκτελούν κάθε query: insert, select, update και delete. Κατά συνέπεια υπάρχουν τόσες κλάσεις repository όσες και στο πακέτο model, με μοναδική εξαίρεση την οντότητα Role που αποτελεί Enumeration των διαφορετικών ρολών της εφαρμογής και αποθηκεύετε στον πίνακα των χρηστών. Όλες οι κλάσεις αυτού του πακέτου αποτελούν διεπαφές (interface) που υλοποιούν το παρεχόμενο από τη Spring, JpaRepository. Μέσω αυτής της υλοποίησης πραγματοποιούνται εύκολα τα βασικά query και ταυτόχρονα μπορούν να συνταχθούν και εκτελεστούν προσαρμοσμένα queries για κάθε περίπτωση με τη σήμανση @Query.

#### 4.2.4 Service

Οι κλάσεις αυτού του επιπέδου αποτελούν την καρδιά του συστήματος, καθώς αναλαμβάνουν να συντονίσουν όλα τα υπόλοιπα επίπεδα και να επικοινωνήσουν με τα ξεχωριστά μέρη από τα οποία απαρτίζεται το σύστημα. Συγκεκριμένα τα services αποτελούν τις προεκτάσεις των μεθόδων των controller και με άμεση συνεργασία με τα αντίστοιχα repository, αναλαμβάνουν να πραγματοποιήσουν την περίπλοκη λογική πριν γίνει κάποια ενέργεια που αφορά τη βάση δεδομένων. Σε αυτό το επίπεδο επίσης, πραγματοποιείται η μετατροπή των δεδομένων που, είτε προέρχονται από το Front End και πρέπει να αντιστοιχηθούν σε οντότητες τύπου model για την αποθήκευση τους στη βάση δεδομένων, είτε στέλνονται από το Back End και πρέπει να αντιστοιχηθούν σε οντότητες τύπου payload για να αναπαρασταθούν στο Front End και να προβληθούν στους χρήστες. Παρακάτω αναγράφονται οι κλάσεις του πακέτου service μαζί με τις αρμοδιότητες τους:

- **AuthService**, προσφέρει τη λογική για την εγγραφή των καινούργιων χρηστών, την ενεργοποίηση των λογαριασμών τους και τη σύνδεση τους στην εφαρμογή.
- **CommentService**, προσφέρει τη λογική για τη δημιουργία και αποθήκευση των καινούργιων σχολίων στις δημοσιεύσεις. Αναλαμβάνει επίσης την μετατροπή Comment σε CommentDto για την προβολή των σχολίων στο Front End.
- **ConfirmationTokenService**, προσφέρει τη λογική για τη δημιουργία και αποθήκευση των ConfirmationToken που χρειάζονται για την ενεργοποίηση των λογαριασμών στην εφαρμογή.
- **ConnectionService**, προσφέρει τη λογική για τη δημιουργία, αποθήκευση, τροποποίηση αλλά και διαγραφή των συνδέσεων μεταξύ των χρηστών.
- **EmailService**, προσφέρει τη λογική για τη δημιουργία και αποστολή των email που στέλνονται από το σύστημα.
- **ImageService**, προσφέρει τη λογική για τη δημιουργία και αποθήκευση των εικόνων που αναρτιούνται στην εφαρμογή.
- **LocationService**, προσφέρει τη λογική για την τροποποίηση συντεταγμένων που προέρχονται από το frontend σε οντότητες Geometry, που μπορεί να επεξεργαστεί το σύστημα και επίσης υπολογίζει την απόσταση χρήστη και δημοσίευσης.
- **NotificationService**, προσφέρει τη λογική για τη δημιουργία και αποθήκευση των ειδοποιήσεων που πρέπει να λάβει ο χρήστης. Αναλαμβάνει επίσης την μετατροπή Notification σε NotificationDto για την προβολή των ειδοποιήσεων στο Front End.
- **PostService**, προσφέρει τη λογική για τη δημιουργία και αποθήκευση των δημοσιεύσεων που αναρτώνται στην εφαρμογή. Αναλαμβάνει επίσης την μετατροπή Post σε PostDto για την προβολή των δημοσιεύσεων στο Front End.
- **UserDetailsServiceImpl**, πραγματοποιεί έλεγχο πιστοποίησης κατά τη διάρκεια της σύνδεσης και εξουσιοδοτεί τον χρήστη με τα ανάλογα προνόμια.
- **UserService**, προσφέρει τη λογική για τη δημιουργία, αποθήκευση και τροποποίηση των χρηστών της εφαρμογής. Αναλαμβάνει επίσης την μετατροπή User σε UserDto για την προβολή των χρηστών στο Front End.

## 4.2.5 Controller

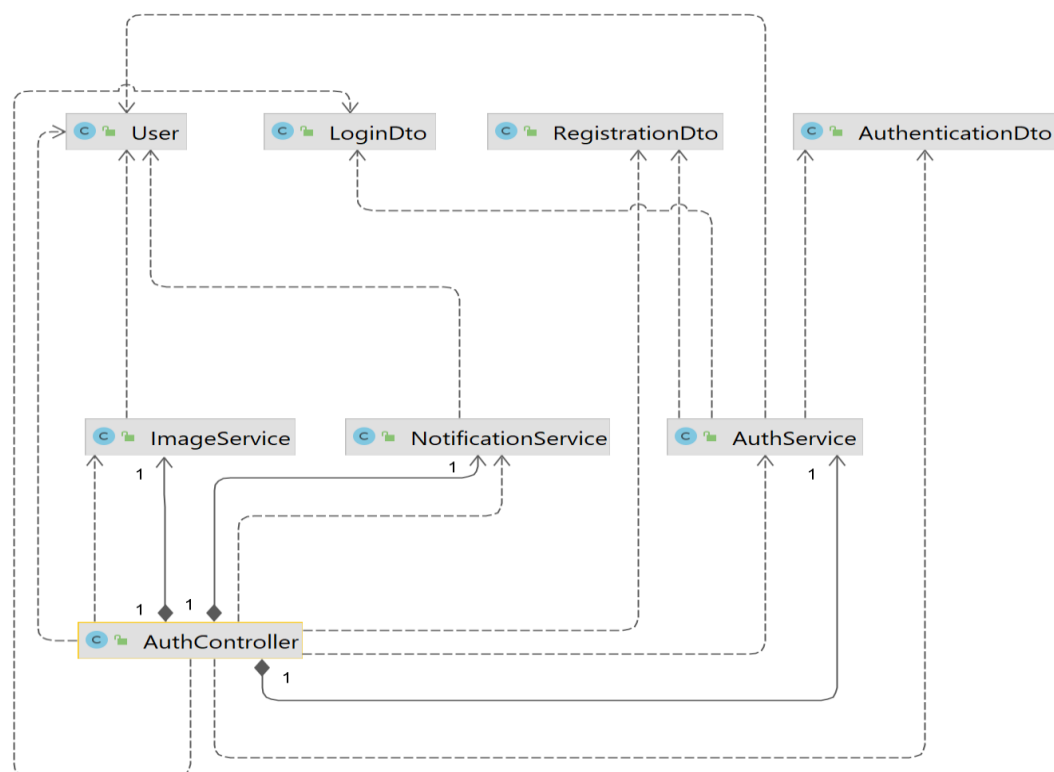
Το τελευταίο επίπεδο της αρχιτεκτονικής του Back End, ορίζει τα τελικά σημεία του API του συστήματος. Οι κλάσεις αυτού του πακέτου αποτελούν την αρχή κάθε λειτουργίας της εφαρμογής και θα αναλυθούν παρακάτω, μαζί με τις κλάσεις των υπόλοιπων επιπέδων που συνεισφέρουν στο να ολοκληρωθεί μια συγκεκριμένη ενέργεια.

### 4.2.5.1 Auth Controller

Ο συγκεκριμένος controller ευθύνεται για τη διαδικασία εγγραφής και σύνδεσης στο σύστημα. Αναφέρονται παρακάτω οι λειτουργίες που υποστηρίζονται:

1. Δημιουργία λογαριασμού, μέσω μιας μεθόδου τύπου post που λαμβάνει τα δεδομένα που περιέχουν τις πληροφορίες που όρισε ο χρήστης για τον λογαριασμό του.
2. Σύνδεση στην εφαρμογή, μέσω μιας μεθόδου τύπου post που λαμβάνει τα δεδομένα που περιέχουν τα στοιχεία σύνδεσης του χρήστη και επιστρέφει το αναγνωριστικό που χρειάζεται για την αυθεντικοποίηση του χρήστη.
3. Ενεργοποίηση λογαριασμού, μέσω μιας μεθόδου τύπου get που λαμβάνει το μοναδικό αναγνωριστικό που στάλθηκε στο email που όρισε ο χρήστης κατά τη διάρκεια εγγραφής του και επιστρέφει την ανάλογη απάντηση αν ενεργοποιηθεί ο λογαριασμός.

Στο παρακάτω διάγραμμα φαίνονται τα services και οι οντότητες που συνεργάζονται με τον συγκεκριμένο controller για την εκπλήρωση του στόχου του.



Εικόνα 4.14 Κλάση AuthController και Σχέσεις με τα Διαφορετικά Επίπεδα

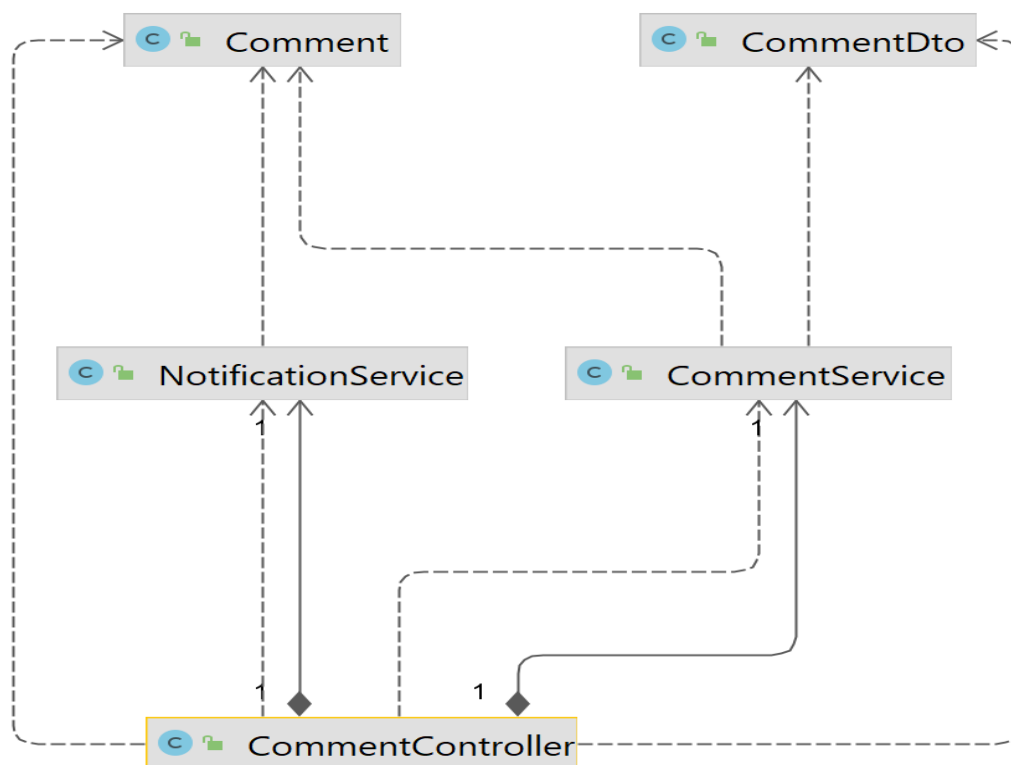


#### 4.2.5.2 Comment Controller

Ο συγκεκριμένος controller ευθύνεται για τη διαδικασία δημιουργίας και προβολής των σχολίων σε δημοσιεύσεις της εφαρμογής. Αναφέρονται παρακάτω οι λειτουργίες που υποστηρίζονται:

1. Δημιουργία και ανάρτηση νέου σχολίου σχετικό με κάποια δημοσίευση στην εφαρμογή, μέσω μιας μεθόδου τύπου post που λαμβάνει το κείμενο που επιθυμεί να δημοσιεύσει ο χρήστης.
2. Προβολή των σχολίων μιας δημοσίευσης, μέσω μιας μεθόδου τύπου get που δέχεται ως παράμετρο το αναγνωριστικό της δημοσίευσης και επιστρέφει τα σχετικά σχόλια με τις απαραίτητες πληροφορίες.

Στο παρακάτω διάγραμμα φαίνονται τα services και οι οντότητες που συνεργάζονται με τον συγκεκριμένο controller για την εκπλήρωση του στόχου του.



Εικόνα 4.15 Κλάση *CommentController* και Σχέσεις με τα Διαφορετικά Επίπεδα

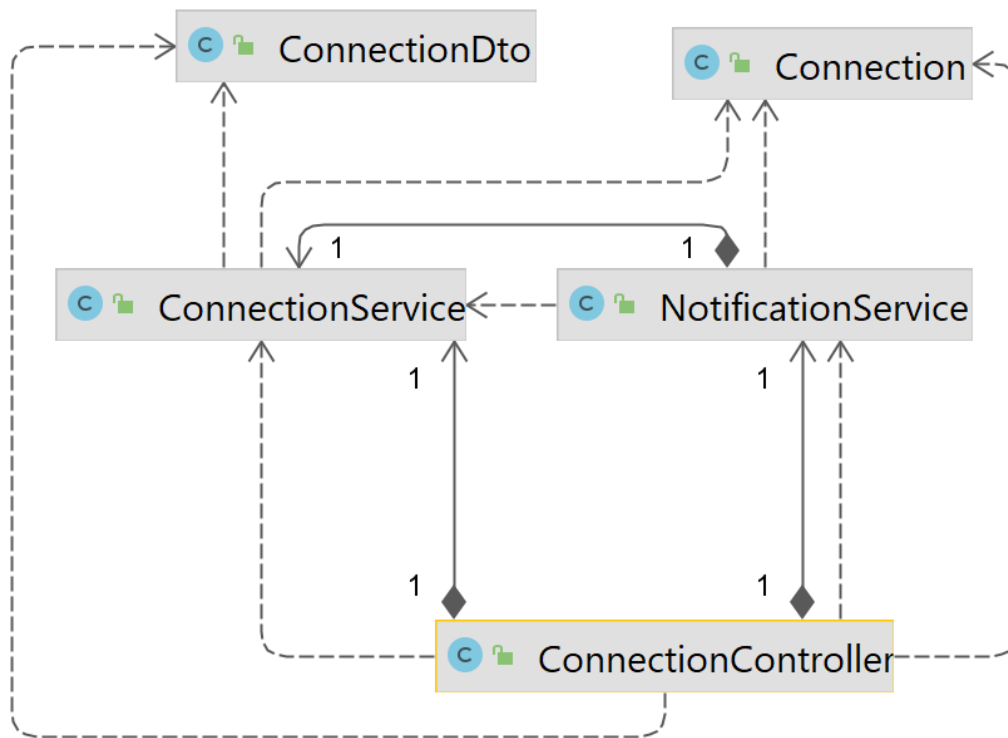
#### 4.2.5.3 Connection Controller

Ο συγκεκριμένος controller ευθύνεται για τη διαδικασία σύνδεσης μεταξύ των χρηστών και για την προβολή των συνολικών συνδέσεων κάθε χρήστη. Αναφέρονται παρακάτω οι λειτουργίες που υποστηρίζονται:

1. Δημιουργία αιτήματος σύνδεσης, μέσω μιας μεθόδου τύπου post που δέχεται ως παράμετρο το αναγνωριστικό του χρήστη για τον οποίο γίνεται το αίτημα.

2. Προβολή των συνδέσεων του χρήστη, μέσω μιας μεθόδου τύπου get που δέχεται ως παράμετρο το είδος της σύνδεσης και επιστρέφει τις συνδέσεις που ταιριάζουν στο είδος που ζητήθηκε.
3. Προβολή της κατάστασης μιας σύνδεσης του χρήστη με κάποιον άλλον, μέσω μιας μεθόδου τύπου get που δέχεται ως παράμετρο το αναγνωριστικό του άλλου χρήστη και επιστρέφει την κατάσταση της σύνδεσης αν υπάρχει.
4. Την απόρριψη ενός αιτήματος σύνδεσης, μέσω μιας μεθόδου τύπου delete που λαμβάνει το αναγνωριστικό του χρήστη που έκανε την αίτηση.
5. Τη διαγραφή μιας υπάρχουσα σύνδεσης μεταξύ του χρήστη και κάποιου άλλου, μέσω μιας μεθόδου τύπου delete που λαμβάνει το αναγνωριστικό του άλλου χρήστη.

Στο παρακάτω διάγραμμα φαίνονται τα services και οι οντότητες που συνεργάζονται με τον συγκεκριμένο controller για την εκπλήρωση του στόχου του.



Εικόνα 4.16 Κλάση ConnectionController και Σχέσεις με τα Διαφορετικά Επίπεδα

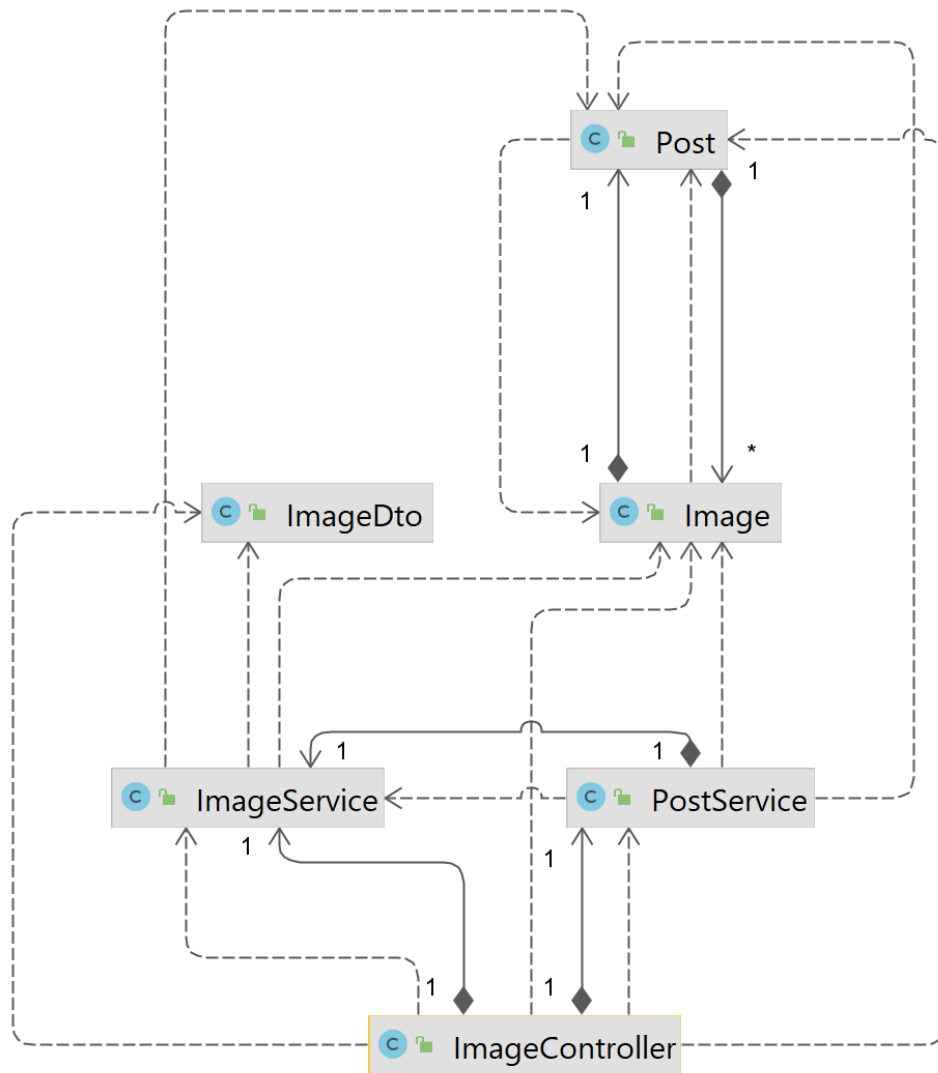
#### 4.2.5.4 Image Controller

Ο συγκεκριμένος controller ευθύνεται για τη διαδικασία συλλογής και προβολής των εικόνων που έχουν αποθηκευτεί στη βάση δεδομένων. Αναφέρονται παρακάτω οι λειτουργίες που υποστηρίζονται:

1. Συλλογή και μετατροπή των εικόνων κάθε δημοσίευσης σε δεδομένα για το Front End, μέσω μιας μεθόδου τύπου get που δέχεται ως παράμετρο το αναγνωριστικό της εκάστοτε δημοσίευσης και επιστρέφει μια λίστα με τα δεδομένα των σχετικών εικόνων.

2. Οπτικοποίηση των δεδομένων μιας εικόνας, μέσω μιας μεθόδου τύπου `get` που δέχεται ως παράμετρο το αναγνωριστικό κάθε εικόνας και δημιουργεί και μετά επιστρέφει συνδέσμου που «δείχνει» σε αυτή.

Στο παρακάτω διάγραμμα φαίνονται τα services και οι οντότητες που συνεργάζονται με τον συγκεκριμένο controller για την εκπλήρωση του στόχου του.



Εικόνα 4.17 Κλάση ImageController και Σχέσεις με τα Διαφορετικά Επίπεδα

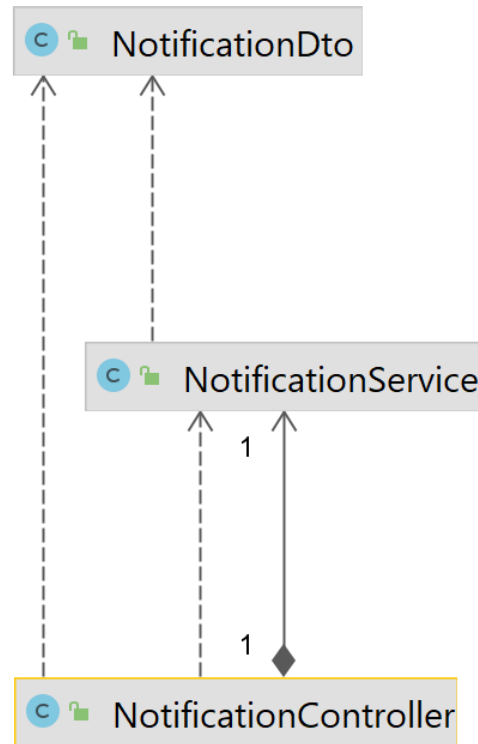
#### 4.2.5.5 Notification Controller

Ο συγκεκριμένος controller ευθύνεται για τη διαδικασία συλλογής, προβολής και επεξεργασίας όλων των ειδοποιήσεων κάθε χρήστη. Αναφέρονται παρακάτω οι λειτουργίες που υποστηρίζονται:

1. Προβολή των ειδοποιήσεων που ανήκουν στον κάθε χρήστη, μέσω μιας μεθόδου τύπου `get` που επιστρέφει τις αντίστοιχες ειδοποιήσεις ταξινομημένες ως προς τη στιγμή δημιουργίας τους.

2. Προβολή των συνολικών αδιάβαστων ειδοποιήσεων του κάθε χρήστη, μέσω μιας μεθόδου τύπου `get` που επιστρέφει τον αριθμό αδιάβαστων ειδοποιήσεων.
3. Ανάγνωση μιας συγκεκριμένης ειδοποίησης, μέσω μιας μεθόδου τύπου `put` που δέχεται ως όρισμα το αναγνωριστικό της ειδοποίησης.
4. Διαγραφή μιας συγκεκριμένης ειδοποίησης, μέσω μιας μεθόδου τύπου `delete` που δέχεται ως όρισμα το αναγνωριστικό της ειδοποίησης.

Στο παρακάτω διάγραμμα φαίνονται τα `services` και οι οντότητες που συνεργάζονται με τον συγκεκριμένο `controller` για την εκπλήρωση του στόχου του.



Εικόνα 4.18 Κλάση `NotificationController` και Σχέσεις με τα Διαφορετικά Επίπεδα

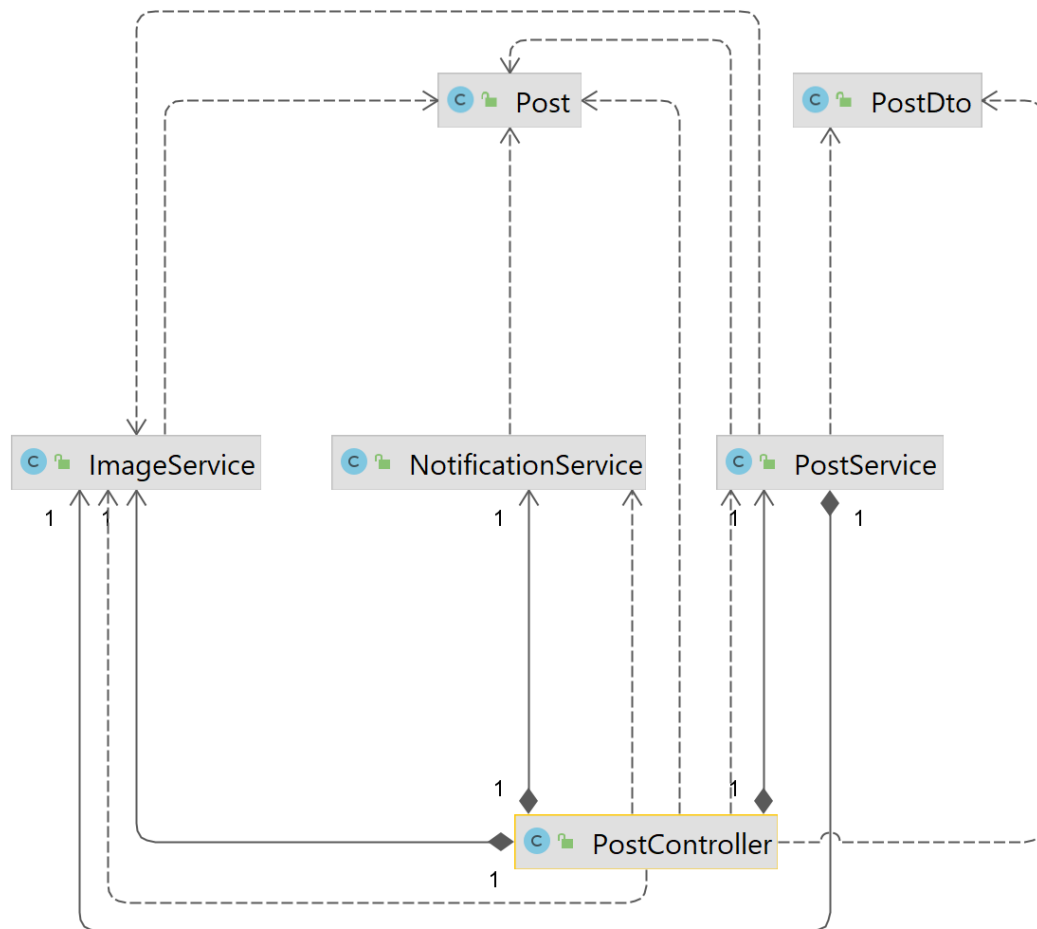
#### 4.2.5.6 Post Controller

Ο συγκεκριμένος `controller` ευθύνεται για τη διαδικασία δημιουργίας, προβολής, και τροποποίησης των δημοσιεύσεων. Αναφέρονται παρακάτω οι λειτουργίες που υποστηρίζονται:

1. Δημιουργία και ανάρτηση νέας δημοσίευσης, μέσω μιας μεθόδου τύπου `post` που λαμβάνει τα δεδομένα που περιέχουν τις πληροφορίες που όρισε ο χρήστης για τη δημοσίευση του.
2. Προβολή όλων των δημοσιεύσεων που έχουν αναρτηθεί στην εφαρμογή, μέσω μιας μεθόδου τύπου `get` που δέχεται ως προαιρετική παράμετρο τα επιθυμητά στοιχεία των δημοσιεύσεων για προβολή και επιστρέφει τα αποτελέσματα αφού έχει γίνει το προαιρετικό φιλτράρισμα.
3. Προβολή πληροφοριών κάποιας συγκεκριμένης δημοσίευσης, μέσω μιας μεθόδου τύπου `get` που δέχεται ως παράμετρο το αναγνωριστικό της σχετικής δημοσίευσης.

4. Ενημέρωση απάντησης δήλωσης συμμετοχής σε αναρτημένες εκδηλώσεις, μέσω μιας μεθόδου τύπου put που δέχεται ως παράμετρο το αναγνωριστικό της σχετικής εκδήλωσης και προσθέτει τον χρήστη στους ενδιαφερόμενους για την εκδήλωση ή τον αφαιρεί αν έχει ήδη δηλώσει ενδιαφέρον.
5. Διαγραφή μιας αναρτημένης δημοσίευσης, μέσω μιας μεθόδου τύπου delete που δέχεται ως παράμετρο το αναγνωριστικό της σχετικής δημοσίευσης.

Στο παρακάτω διάγραμμα φαίνονται τα services και οι οντότητες που συνεργάζονται με τον συγκεκριμένο controller για την εκπλήρωση του στόχου του.



Εικόνα 4.19 Κλάση PostController και Σχέσεις με τα Διαφορετικά Επίπεδα

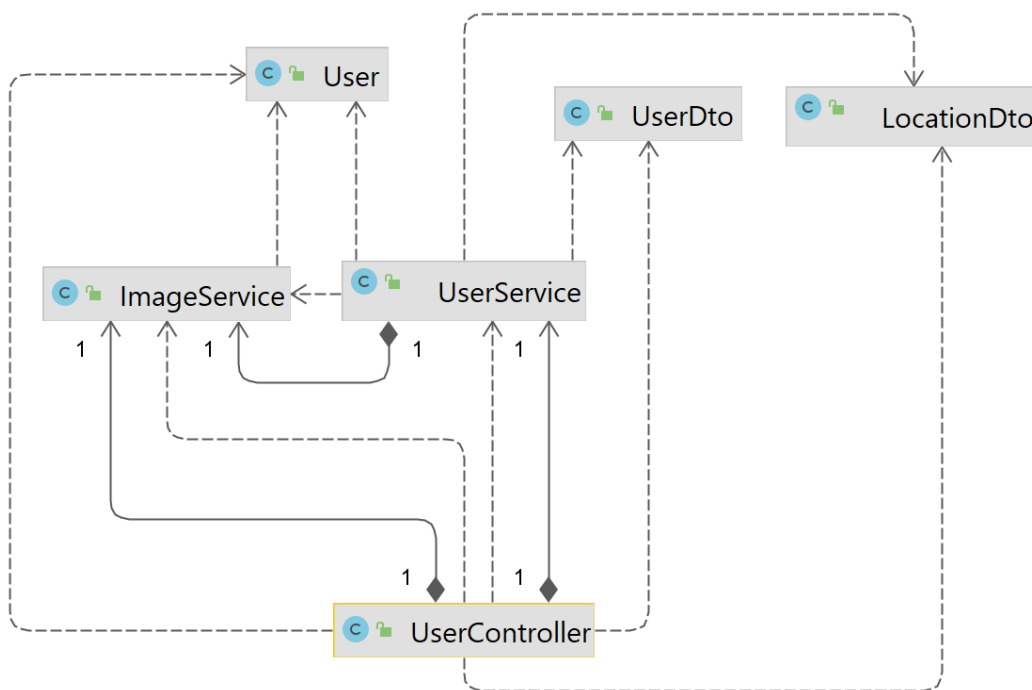
#### 4.2.5.7 User Controller

Ο συγκεκριμένος controller ευθύνεται για τη διαδικασία προβολής των χρηστών και τροποποίησης των στοιχείων κάθε χρήστη. Αναφέρονται παρακάτω οι λειτουργίες που υποστηρίζονται:

1. Πρόσκληση νέου χρήστη, μέσω μια μεθόδου τύπου post που λαμβάνει ως παράμετρο το email αυτού του χρήστη και αποστέλλει το ανάλογο μήνυμα πρόσκλησης.

2. Προβολή όλων των χρηστών που έχουν εγγραφεί στην εφαρμογή, μέσω μιας μεθόδου τύπου `get` που δέχεται ως προαιρετική παράμετρο το όνομα του χρήστη προς αναζήτηση και επιστρέφει τα αποτελέσματα.
3. Προβολή πληροφοριών κάποιου συγκεκριμένου χρήστη, μέσω μιας μεθόδου τύπου `get` που δέχεται ως παράμετρο το αναγνωριστικό του εκάστοτε χρήστη.
4. Ενημέρωση της τοποθεσίας του χρήστη, μέσω μιας μεθόδου τύπου `put` που λαμβάνει τα νέα γεωγραφικά δεδομένα και αλλάζει τα αντίστοιχα στοιχεία του χρήστη στη βάση δεδομένων.
5. Ενημέρωση των προσωπικών στοιχείων του χρήστη, μέσω μιας μεθόδου τύπου `put` που λαμβάνει τα νέα δεδομένα και αλλάζει τα αντίστοιχα στοιχεία του χρήστη στη βάση δεδομένων.

Στο παρακάτω διάγραμμα φαίνονται τα `services` και οι οντότητες που συνεργάζονται με τον συγκεκριμένο `controller` για την εκπλήρωση του στόχου του.



Εικόνα 4.20 Κλάση `UserController` και Σχέσεις με τα Διαφορετικά Επίπεδα

## 4.2.6 Συμπληρωματικές Κλάσεις

Για την ολοκλήρωση όλων των απαραίτητων ενεργειών που θέλαμε να πραγματοποιεί το σύστημα μας, χρειάστηκε να δημιουργηθούν συμπληρωματικές κλάσεις που ανήκουν σε ξεχωριστά πακέτα του συστήματος και είναι υπεύθυνες για τη διεκπεραίωση συγκεκριμένων δουλειών. Παρακρατώ αναγράφονται τα σχετικά πακέτα, καθώς και μερικά λογία που περιγράφουν τις κλάσεις του καθένα.

#### **4.2.6.1 config**

Αυτό το πακέτο περιέχει δυο κλάσεις, `SecurityConfig` και `WebConfig`, που είναι υπεύθυνες για την διαχείριση των ρυθμίσεων της εφαρμογής. Η πρώτη είναι υπεύθυνη για τη διαμόρφωση της ασφάλειας της εφαρμογής, επιτρέπει τη δημιουργία κρυπτογραφημένων κωδίκων και τους λογαριασμούς των χρηστών και ασφαλίζει τα τερματικά σημεία του API. Η δεύτερη επιτρέπει την επικοινωνία μεταξύ Back End και Front End μέσω του μηχανισμού CORS και ευθύνεται για την λειτουργία των κλάσεων του πακέτου `converter`.

#### **4.2.6.2 payload**

Σε αυτό το πακέτο βρίσκονται οι κλάσεις που είναι υπεύθυνες για την αναπαράσταση των οντοτήτων της βάσης δεδομένων σε δεδομένα που μπορούν να προβληθούν στο Front End. Ως συνέπεια γίνεται αντιστοίχιση των κλάσεων του πακέτου `model` σε κλάσεις αυτού του πακέτου, π.χ. Η κλάση `Post` του `model` αντιστοιχίζεται στη κλάση `PostDto` του `payload`. Επίσης, μέσω αυτών των κλάσεων γίνεται η ανταλλαγή δεδομένων μεταξύ Back End και Front End

#### **4.2.6.3 security**

Οι κλάσεις του πακέτου αυτού, είναι υπεύθυνες για την πραγματοποίηση της σύνδεσης των χρηστών στο σύστημα και την παροχή των απαραίτητων προνομίων στον κάθε χρήστη αναλόγως. Επίσης σε αυτό το πακέτο γίνεται η δημιουργία των JSON Web Tokens (jwt)[14] που χρειάζονται για την επικύρωση των αιτήσεων από το Front End.

#### **4.2.6.4 utility και converter**

Σε αυτά τα δυο πακέτα βρίσκονται βοηθητικές κλάσεις. Πιο συγκεκριμένα, το πακέτο `utility` περιέχει την διεπαφή `EmailSender` που χρησιμοποιεί το σύστημα για την αποστολή ηλεκτρονικών μηνυμάτων και τη κλάση `EmailValidator` που είναι υπεύθυνη για τη πιστοποίηση των διευθύνσεων email τις οποίες χρησιμοποιούν οι χρήστες για την εγγραφή τους στην εφαρμογή.

Στο πακέτο `converter` βρίσκονται οι βοηθητικές κλάσεις που επιτρέπουν την μετατροπή των σύνθετων δεδομένων που προέρχονται από το Front End σε οντότητες του Back End.

### **4.2.7 Front End**

Στην πολυεπίπεδη αρχιτεκτονική που επιλέξαμε για το σύστημα μας, αυτό είναι το τελικό επίπεδο. Διαμορφώνει την γραφική διεπαφή με την οποία αλληλοεπιδρά ο

χρήστης με την εφαρμογή και πραγματοποιεί την ανταλλαγή και αναπαράσταση δεδομένων από το Back End.

Παρακάτω αναφέρεται η δομή του συστήματος:

- **auth:** Ριζικός φάκελος που περιέχει τα components σχετικά με τη σύνδεση και εγγραφή του χρήστη.
  - **login:** Component που μέσω μιας φόρμας επιτρέπει την σύνδεση στην εφαρμογή.
  - **registration:** Υποφάκελος που περιέχει τα components που αφορούν την εγγραφή.
    - **register:** Component που μέσω μιας φόρμας επιτρέπει την εγγραφή στην εφαρμογή. Προσφέρει διαφορετική φόρμα ανάλογα το είδος του χρήστη, ιδιώτης ή φιλοζωικός οργανισμός.
    - **register-account-type-selection:** Component που επιτρέπει στο χρήστη να επιλέξει το είδος του λογαριασμού που τον ενδιαφέρει να δημιουργήσει.
    - **register-header:** Βοηθητικό Component που εμφανίζεται μαζί με τα προηγούμενα δυο και προσφέρει πληροφορίες για την εγγραφή στον χρήστη.
    - **register-success:** Component που ενημερώνει τον χρήστη για την επιτυχή εγγραφή του στην εφαρμογή και τον παρακινεί να ελέγξει τα email του για την ενεργοποίηση του λογαριασμού του.
- **errors:** Ριζικός φάκελος που περιέχει τα components σχετικά με τα σφάλματα που μπορούν να συμβούν κατά τη χρήση της εφαρμογής.
  - **location-error:** Component που εμφανίζεται μετά από σφάλμα ορισμού τοποθεσίας χρήστη και περιλαμβάνει μήνυμα λάθους.
  - **login-error:** Component που εμφανίζεται μετά από σφάλμα σύνδεσης χρήστη και περιλαμβάνει μήνυμα λάθους.
  - **post-error:** Component που εμφανίζεται μετά από σφάλμα ανάρτησης δημοσίευσης και περιλαμβάνει μήνυμα λάθους.
  - **register-error:** Component που εμφανίζεται μετά από σφάλμα εγγραφής χρήστη και περιλαμβάνει μήνυμα λάθους.
- **home:** Ριζικός φάκελος που περιέχει τα components από τα οποία απαρτίζεται η κεντρική σελίδα της εφαρμογής.
  - **filter:** Component που μέσω μιας φόρμας επιτρέπει στον χρήστη να επιλέξει τα επιθυμητά κριτήρια των δημοσιεύσεων που τον ενδιαφέρει να δει στην αρχική του σελίδα.
  - **header:** Component που αποτελεί την κεφαλίδα της εφαρμογής και περιέχει χρήσιμες συντομεύσεις για την περιήγηση στην εφαρμογή.
  - **homepage:** Component που αποτελεί την κεντρική σελίδα της εφαρμογής. Σε αυτό το component γίνεται η προεπισκόπηση όλων των δημοσιεύσεων της εφαρμογής και ο διαχωρισμός των κανονικών δημοσιεύσεων και των εκδηλώσεων.
  - **near-me:** Component που επιτρέπει στον χρήστη να ορίσει την τοποθεσία του και να επιλέξει την ακτίνα μέσα στην οποία επιθυμεί να βρίσκονται οι δημοσιεύσεις που θέλει να βλέπει.



- **map:** Αποτελεί βοηθητικό Component που περιέχει έναν χάρτη μαζί με πεδίο αναζήτησης. Έχει χρησιμοποιηθεί σε συνδυασμό με άλλα Component που χρειάζονται λειτουργικότητα χάρτη, όπως το near-me και add-post.
- **payload:** Ριζικός φάκελος που περιέχει διεπαφές που θα χρησιμοποιηθούν για την ανταλλαγή δεδομένων μεταξύ frontend και backend.
- **posts:** Ριζικός φάκελος που περιέχει τα components που σχετίζονται με τη δημιουργία και προβολή των δημοσιεύσεων.
  - **add-post:** Component που μέσω μιας φόρμας επιτρέπει στον χρήστη να δημιουργήσει δημοσίευση για το θέμα που τον αφορά.
  - **comments:** Υποφάκελος που περιέχει τα components που αφορούν τα σχόλια των δημοσιεύσεων.
    - **add-comment:** Component που μέσω ενός πεδίου υποδοχής κειμένου επιτρέπει στον χρήστη να σχολιάσει τη δημοσίευση που τον ενδιαφέρει.
    - **discussion:** Component που εμφανίζει όλα τα σχόλια που έχουν γίνει στην εκάστοτε δημοσίευση.
  - **post:** Component που αποτελεί τη σελίδα μιας αναρτημένης δημοσίευσης και περιέχει όλες τις σχετικές πληροφορίες.
  - **post-type-selection:** Component που επιτρέπει στο χρήστη να επιλέξει το είδος της δημοσίευσης που τον ενδιαφέρει να δημιουργήσει.
- **security:** Ριζικός φάκελος που περιέχει τα components που σχετίζονται με την ασφάλεια της εφαρμογής.
- **service:** Ριζικός φάκελος που περιέχει τις υπηρεσίες που πραγματοποιούν τις κλήσεις στο backend και χρησιμοποιούνται από τα υπόλοιπα components για να ανταλλάξουν δεδομένα.
- **users:** Ριζικός φάκελος που περιέχει τα components που σχετίζονται με τους χρήστες και τη σελίδα προφίλ του κάθε χρήστη της εφαρμογής.
  - **community:** Component που αποτελεί τη σελίδα όπου προβάλλονται όλοι οι χρήστες της εφαρμογής. Εδώ μπορεί να πραγματοποιηθεί αναζήτηση για κάποιον άλλο χρήστη.
  - **connections:** Component που περιέχει όλες τις συνδέσεις του χρήστη και επιτρέπει την απάντηση σε αιτήματα φιλίας.
  - **notifications:** Component που περιέχει όλες τις ειδοποιήσεις του χρήστη και επιτρέπει την ανάγνωση και διαγραφή τους.
  - **profile:** Component που προβάλλει τα στοιχεία του ιδίου του χρήστη και του επιτρέπει την τροποποίησή τους.
  - **user:** Component που αποτελεί τη σελίδα προφίλ των χρηστών.

## Κεφάλαιο 5. Έλεγχος

Στο κεφάλαιο αυτό θα αναλυθούν οι έλεγχοι που υλοποιήθηκαν για τον κώδικα της εφαρμογής. Πραγματοποιήθηκαν unit καθώς και integration/system tests. Οι μέθοδοι των κλάσεων των πακέτων repository και service ελέγχθηκαν με unit tests, σκοπός των οποίων ήταν η επιβεβαίωση ορθής λειτουργίας των επιμέρους κομματιών της εφαρμογής σε απομόνωση. Στις μεθόδους των κλάσεων του πακέτου controller, πραγματοποιήθηκαν integration/system tests με σκοπό την επιβεβαίωση ορθής λειτουργίας όλων των επιπέδων της εφαρμογής. Για την υλοποίησή τους χρησιμοποιήθηκε η βιβλιοθήκη Junit Jupiter[15] καθώς και η βιβλιοθήκη Mockito[16].

### 5.1 Έλεγχος Μεθόδων Πακέτου repository

[GitHub link](#)

Οι έλεγχοι των μεθόδων του πακέτου repository αξιοποιούν την H2 DB, μια βάση δεδομένων στη μνήμη η οποία εξαλείφει την ανάγκη διαμόρφωσης και εκκίνησης μιας πραγματικής βάσης δεδομένων. Αυτή η διαρρύθμιση αρκεί για τις απαιτήσεις των αναγκαίων ελέγχων σε αυτό το σημείο.

Κάθε κλάση έλεγχος αντιπροσωπεύει μια κλάση repository. Οι έλεγχοι αρχικά δημιουργούν και αποθηκεύουν τις αντίστοιχες οντότητες κάθε repository π.χ. το UserRepositoryTest δημιουργεί αντικείμενα τύπου User, το PostRepositoryTest αντικείμενα τύπου Post και ούτω καθεξής. Ύστερα, γίνεται χρήση των μεθόδων της κλάσης με τελικό στόχο τη σύγκριση των αναμενόμενων αποτελεσμάτων με των πραγματικών αποτελεσμάτων που επιστρέφουν οι μέθοδοι.

Σε ορισμένες περιπτώσεις διαπιστώθηκε η ορθή λειτουργία των μεθόδων όχι μόνο από τη ταύτιση αναμενόμενων και πραγματικών αποτελεσμάτων, αλλά και από την μη επιστροφή αποτελεσμάτων κατά τη διάρκεια ελέγχου της μεθόδου, σε στιγμές όπου δεν

αναμενόταν αποτέλεσμα. Τέτοιοι έλεγχοι συμβολίζονται από το πρόθεμα **'not'** στην αρχή του ονόματος τους.

Παρακάτω φαίνεται η επιτυχής έκβαση των 32 ελέγχων που αφορούν το πακέτο repository.

✓ repository (app.rescue.backend)	826 ms
✓ ConfirmationTokenRepositoryTest	243 ms
✓ notFindByToken()	199 ms
✓ updateConfirmedAt()	33 ms
✓ findByToken()	11 ms
✓ UserRepositoryTest	146 ms
✓ existsByEmailAndEnabled()	49 ms
✓ notExistsByEmailAndEnabled()	11 ms
✓ notFindByEmail()	19 ms
✓ notFindByReferralToken()	11 ms
✓ findByEmail()	10 ms
✓ enableUser()	30 ms
✓ findByReferralToken()	16 ms
✓ IndividualInformationRepositoryTest	46 ms
✓ setIndividualInformationUser()	46 ms
✓ ConnectionRepositoryTest	224 ms
✓ findPendingConnection()	21 ms
✓ notFindConnectionByUserAndConnectedTold()	20 ms
✓ findRefPendingConnection()	19 ms
✓ findConnectionByUserAndConnectedTold()	9 ms
✓ findConnectionsByUser()	21 ms
✓ findRefFollowerConnection()	13 ms
✓ getAllByUserAndConnectionStatus()	10 ms
✓ completeRefOrgConnection()	40 ms
✓ completeConnection()	30 ms
✓ notFindPendingConnection()	11 ms
✓ notFindRefPendingConnection()	12 ms
✓ getAllByConnectedToldAndConnectionStatus()	18 ms
✓ CommentRepositoryTest	29 ms
✓ findAllCommentsByPost()	29 ms
✓ EventPropertiesRepositoryTest	41 ms
✓ DoesNotExistByPostAndEventAttendeesContaining()	23 ms
✓ existsByPostAndEventAttendeesContaining()	18 ms
✓ ImageRepositoryTest	29 ms
✓ findByPost()	20 ms
✓ findByUserAndProfileImage()	9 ms
✓ NotificationRepositoryTest	49 ms
✓ notificationRead()	19 ms
✓ findAllByUserOrderByIdDesc()	21 ms
✓ getUnreadNotification()	9 ms
✓ OrganizationInformationRepositoryTest	19 ms
✓ setOrganizationInformationUser()	19 ms

Εικόνα 5.1 Έλεγχοι Πακέτου repository

## 5.2 Έλεγχος Μεθόδων Πακέτου service

[GitHub link](#)

Οι έλεγχοι των μεθόδων του πακέτου service αξιοποιούν τη βιβλιοθήκη Mockito, με την οποία μπορούμε να προσομοιώσουμε τα κρίσιμα κομμάτια κάθε κλάσης και μεθόδου που είναι απαραίτητες για την ροή του προγράμματος, αλλά δεν είναι τα μέλη που ελέγχονται τη δεδομένη στιγμή.

Κάθε κλάση έλεγχος αντιπροσωπεύει μια κλάση service. Οι έλεγχοι αξιοποιούν mocks για να επιτευχθεί η φυσική ροή κάθε μεθόδου που περιλαμβάνει τη κλήση μεθόδων άλλων services και repositories. Σκοπός είναι να διαπιστωθεί ότι πράγματι έγινε κλήση στις αναμενόμενες μεθόδους με τις σωστές παραμέτρους. Έτσι επιβεβαιώνεται η αναμενόμενη λειτουργία του συστήματος, γιατί οι μέθοδοι που καλούνται έχουν ελεγχθεί για την ορθή τους λειτουργία σε άλλα σημεία των ελέγχων μας.

Ορισμένες μέθοδοι που ελέγχονται διαθέτουν πολλαπλές ροές λειτουργίας, για τον έλεγχο όλων των πιθανών ροών χρησιμοποιήθηκε η μέθοδος `given().willReturn()` του Mockito, έτσι ώστε να φέρουμε το σύστημα στην επιθυμητή κατάσταση κάθε φορά.

Τέλος υπάρχουν ροές που δεν επιτρέπονται από το σύστημα και επιστρέφουν σφάλμα. Για τον έλεγχο αυτών φέραμε το σύστημα σε μη επιτρεπτή κατάσταση και υστέρα ελέγξαμε ότι παρουσιάστηκε το αναμενόμενο σφάλμα και παράλληλα δεν πραγματοποιήθηκε καμία κλήση σε μέθοδο σύμφωνα με την κανονική ροή της μεθόδου. Τέτοιοι έλεγχοι συμβολίζονται από σύμβαση ονομασίας τύπου: Όνομα Μεθόδου-WillThrowWhen-Συνθήκη Σφάλματος.

Παρακάτω φαίνεται η επιτυχής έκβαση των 108 ελέγχων που αφορούν το πακέτο service.

✓ service (app.rescue.backend)	1 sec 879 ms
✓ LocationServiceTest	126 ms
✓ canTurnUserLocationToCircle()	30 ms
✓ proximityCheckWillReturnTrue()	81 ms
✓ getDistanceFromPostInMetersWillReturnMinus1WhenPostLocationIsNullOrEmpty()	5 ms
✓ getDistanceFromPostInMetersWillReturnMinus1WhenUserLocationIsNullOrEmpty()	1 ms
✓ proximityCheckWillReturnFalse()	
✓ canGetDistanceFromPostInMeters()	
✓ canTurnPostLocationToPoint()	9 ms
✓ EmailServiceTest	551 ms
✓ canSendInvitationEmail()	551 ms
✓ canSendConfirmAccountEmail()	
✓ ImageServiceTest	520 ms
✓ canStoreDefaultOrganizationProfileImage()	393 ms
✓ canGetPostImages()	47 ms
✓ getImageWillThrowWhenImageDoesNotExist()	19 ms
✓ canGetProfileImage()	
✓ canCreateFileDownloadUri()	3 ms
✓ canStoreCustomProfileImage()	10 ms
✓ canStoreDefaultIndividualProfileImage()	30 ms
✓ getPostImagesReturnsNullIfPostHasNoImages()	8 ms
✓ canGetImage()	
✓ canStorePostImages()	4 ms
✓ storeProfileImageWillThrowWhenNotAnUserRole()	
✓ canUpdateProfileImage()	6 ms
✓ PostServiceTest	217 ms
✓ CreateNewPostWillThrowWhenPostTypesNotAllowed()	98 ms
✓ canGetSinglePost()	54 ms
✓ canAddCommentator()	5 ms
✓ canFindById()	2 ms
✓ findByIdWillThrowWhenPostDoesNotExist()	
✓ deletePostWillThrowWhenNotPostOwner()	
✓ canCreateNewEventPost()	8 ms
✓ canDeletePost()	1 ms
✓ canNotAttendEvent()	10 ms
✓ canAttendEvent()	
✓ canCreateNewAnimalPost()	9 ms
✓ canGetAllPosts()	30 ms
✓ attendEventWillThrowIfPostTypesNotEvent()	
✓ UserServiceTest	201 ms
✓ canGetSingleUser()	111 ms
✓ canUpdateUserInfoIndividual()	30 ms
✓ findAll()	
✓ getUserByIdWillThrowWhenUserDoesNotExist()	2 ms
✓ canInviteFriend()	
✓ canGetUserById()	8 ms
✓ canSignUpOrganization()	
✓ canGetUserByEmail()	
✓ findByReferralTokenWillThrowWhenNotAReferralToken()	
✓ inviteFriendWillThrowWhenEmailIsNotValid()	10 ms
✓ canEnableUser()	
✓ doesNotExistById()	
✓ doesExistById()	
✓ canSignUpIndividual()	8 ms
✓ canUpdateUserInfoOrganization()	2 ms
✓ signUpUserWillThrowWhenEmailIsTaken()	
✓ isEnabled()	
✓ signUpUserWillDeleteUserWhenNotEnabled()	9 ms
✓ canGetAllUsers()	
✓ getUserByEmailWillThrowWhenEmailNotFound()	
✓ findByReferralToken()	

Εικόνα 5.2 Έλεγχοι Πακέτου service

✓ canUpdateUserLocation()	21 ms
✓ AuthServiceTest	68 ms
✓ invitationRegistration()	30 ms
✓ loginWillThrowWhenUserNotEnabled()	9 ms
✓ canRegister()	
✓ registerWillThrowWhenEmailNotValid()	
✓ canLogin()	9 ms
✓ confirmTokenWillThrowWhenTokenIsExpired()	11 ms
✓ confirmTokenWillThrowWhenEmailAlreadyConfirmed()	9 ms
✓ canConfirmToken()	
✓ CommentServiceTest	50 ms
✓ canFindById()	39 ms
✓ canAddNewCommentAndAddCommentator()	
✓ canAddNewCommentAndWillNotAddCommentator()	
✓ canGetAllPostComments()	9 ms
✓ findByIdWillThrowWhenCommentDoesNotExist()	2 ms
✓ NotificationServiceTest	73 ms
✓ canSendNewPostNotificationToConnections()	30 ms
✓ deleteNotificationWillThrowWhenNotNotificationOwner()	19 ms
✓ canGetNumberOfUnreadNotifications()	
✓ canSendNewCommentNotification()	
✓ canSendPendingConnectionRequestNotification()	9 ms
✓ canSendNewPostProximityNotification()	4 ms
✓ canSendInvitationCompletedNotification()	2 ms
✓ canDeleteNotification()	1 ms
✓ canReadNotification()	2 ms
✓ canGetAllNotifications()	2 ms
✓ readNotificationWillThrowWhenNotNotificationOwner()	
✓ canSendConnectionAcceptedNotification()	
✓ canSendFollowerConnectionRequestNotification()	4 ms
✓ ConfirmationTokenServiceTest	17 ms
✓ setConfirmedAt()	13 ms
✓ canSaveConfirmationToken()	
✓ getTokenWillThrowWhenTokenNotFound()	4 ms
✓ canGetToken()	
✓ ConnectionServiceTest	56 ms
✓ canGetAllOrganizationsConnections()	17 ms
✓ canConnectWithOrganization()	
✓ canGetAllFriendsConnections()	
✓ canDeclineConnection()	1 ms
✓ connectWithWillThrowWhenUsersAlreadyConnected()	
✓ canConnectWithIndividual()	
✓ canGetAllFriendRequestsConnections()	
✓ acceptConnectionWillThrowWhenUserAlreadyConnected()	10 ms
✓ connectWithWillThrowWhenUsersAreSame()	1 ms
✓ getAllConnectionsWillThrowWhenUnknownConnectionType()	
✓ canDeleteConnection()	
✓ canCompleteRefOrganizationConnection()	
✓ canCompleteRefIndividualConnection()	9 ms
✓ acceptConnectionWillThrowWhenNoPendingConnection()	
✓ canGetAllFollowersConnections()	
✓ invitationFromIndividualRegistration()	
✓ connectWithWillThrowWhenUserIsOrganization()	9 ms
✓ canAcceptConnection()	
✓ invitationFromOrganizationRegistration()	
✓ acceptConnectionWillThrowWhenUserIsOrganization()	
✓ canFindConnectionsByUser()	
✓ canGetConnectionStatus()	9 ms

Εικόνα 5.3 Έλεγχοι Πακέτου service (συνέχεια)

## 5.3 Έλεγχος Μεθόδων Πακέτου controller

[GitHub link](#)

Για τους ελέγχους των μεθόδων του πακέτου controller δημιουργήθηκε μια βάση δεδομένων αποκλειστικά για αυτούς τους ελέγχους, καθώς τώρα θα έπρεπε να διαπιστωθεί η σωστή λειτουργία διαφορετικών σεναρίων χρήσης της εφαρμογής από την αρχή μέχρι το τέλος.

Κάθε κλάση έλεγχος αντιπροσωπεύει μια κλάση controller και η λογική των ελέγχων περιλαμβάνει την εκτέλεση αιτήσεων post, get, put και delete στα σχετικά endpoints με σκοπό την εξακρίβωση της επιτυχής εκτέλεσης των αιτήσεων.

- Για τους ελέγχους των αιτήσεων τύπου post, πρώτα γίνεται αυτή και υστέρτα εκτελείται μια αίτηση τύπου get της αντίστοιχης κλάσης με την οποία παίρνουμε τα πραγματικά αποτελέσματα, τα οποία και συγκρίνουμε με τα αναμενόμενα.
- Για τους ελέγχους των αιτήσεων τύπου get, πρώτα γίνονται οι απαραίτητες αιτήσεις post για να δημιουργηθούν τα απαιτούμενα δεδομένα και μετά εκτελούνται οι αιτήσεις get που μας επιστρέφουν τα πραγματικά δεδομένα, τα οποία και ξανά-συγκρίνουμε με τα αναμενόμενα.
- Για τους ελέγχους των αιτήσεων τύπου put, πρέπει ξανά πρώτα να δημιουργηθούν και να αποθηκευτούν στην βάση τα δεδομένα με αιτήσεις τύπου post. Ύστερα από αυτό γίνεται η εκτέλεση της αίτησης put που μας ενδιαφέρει να ελέγξουμε και τελικά με αίτηση get λαμβάνουμε τα δεδομένα, τα οποία και συγκρίνουμε πάλι με τα αναμενόμενα.
- Τέλος για τους ελέγχους των αιτήσεων τύπου delete, πρέπει για άλλη μια φορά να γίνουν οι απαραίτητες αιτήσεις post που θα δημιουργήσουν τα δεδομένα πρώτα και τα οποία με εκτέλεση αίτησης delete σκοπεύουμε να διαγράψουμε. Τελικά με αίτηση get επιβεβαιώνουμε πως πράγματι έχουν διαγραφεί τα αναμενόμενα δεδομένα.

Παρακάτω φαίνεται η επιτυχής έκβαση των 34 ελέγχων που αφορούν το πακέτο controller.

✓ controller (app.rescue.backend)	4 sec 379 ms
✓ ConnectionControllerTest	446 ms
✓ canGetAllOrganizationsConnections()	106 ms
✓ canGetAllFriendsConnections()	30 ms
✓ canDeclineConnection()	69 ms
✓ canConnectWith()	61 ms
✓ canGetAllFriendRequestConnections()	30 ms
✓ canDeleteConnection()	39 ms
✓ canGetAllFollowersConnections()	31 ms
✓ canAcceptConnection()	63 ms
✓ canGetConnectionStatus()	17 ms
✓ AuthControllerTest	1 sec 249 ms
✓ canRegister()	250 ms
✓ canConfirm()	249 ms
✓ canLogin()	431 ms
✓ canRefRegister()	319 ms
✓ PostControllerTest	671 ms
✓ canGetAllPostsWithUser()	72 ms
✓ canGetSinglePostWithoutUser()	40 ms
✓ canCreateNewEventPost()	50 ms
✓ canDeletePost()	48 ms
✓ canAttendEvent()	311 ms
✓ canCreateNewAnimalPost()	69 ms
✓ canGetAllPostsWithoutUser()	49 ms
✓ canGetSinglePostWithUser()	32 ms
✓ CommentControllerTest	131 ms
✓ canAddNewComment()	82 ms
✓ canGetAllPostComments()	49 ms
✓ ImageControllerTest	101 ms
✓ canGetFile()	50 ms
✓ canGetPostImages()	51 ms
✓ UserControllerTest	1 sec 601 ms
✓ canGetSingleUser()	248 ms
✓ canInviteFriend()	371 ms
✓ canUpdateUserInfo()	241 ms
✓ canGetAllUsers()	460 ms
✓ canUpdateUserLocation()	281 ms
✓ NotificationControllerTest	180 ms
✓ getNumberOfUnreadNotifications()	60 ms
✓ canDeleteNotification()	39 ms
✓ canReadNotification()	44 ms
✓ canGetAllNotifications()	37 ms

Εικόνα 5.4 Έλεγχοι Πακέτου controller

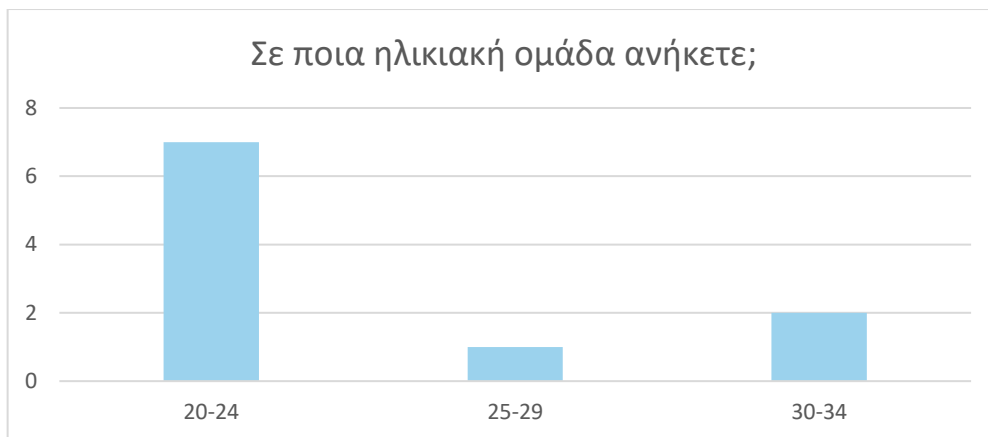


## **Κεφάλαιο 6. Αξιολόγηση της Εφαρμογής από τους Χρήστες**

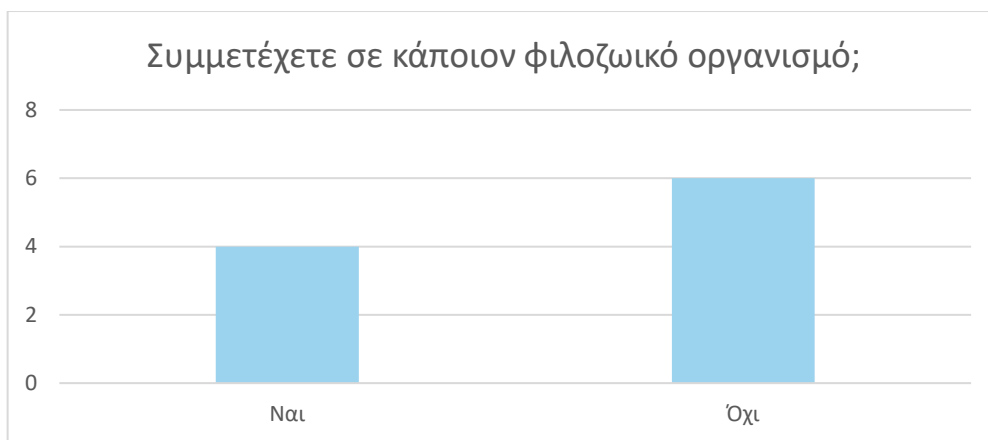
Σε αυτό το κεφάλαιο παρουσιάζονται τα αποτελέσματα της αξιολόγησης της εφαρμογής πάνω σε ποιοτικά χαρακτηριστικά της. Η αξιολόγηση μίας εφαρμογής είναι πολύ σημαντική για την εξέλιξή της και μας βοηθά να καταλάβουμε πως οι άνθρωποι αλληλοεπιδρούν με αυτή. Μας προσφέρει πληροφορίες για το ποια χαρακτηριστικά της εφαρμογής είναι επιθυμητά, μαθαίνουμε αν οι χρήστες θα την χρησιμοποιούσαν ξανά ή θα την πρότειναν σε άλλους και ταυτόχρονα ακούμε νέες προτάσεις για αλλαγές και βελτιώσεις. Για αυτούς του λόγους αποφασίσαμε να αξιολογήσουμε την εφαρμογή μας μέσω ενός User Study. Δώσαμε σε 10 χρήστες, να χρησιμοποιήσουν την εφαρμογή μας, οι οποίοι έπειτα κλήθηκαν να απαντήσουν σε με μία σειρά από ερωτήσεις. Οι ερωτήσεις είναι 15 και αφορούν το προφίλ των χρηστών και την ποιότητα της εφαρμογής, ενώ οι τέσσερις τελευταίες είναι ερωτήσεις ανάπτυξης. Οι ερωτήσεις 1-4 αφορούν τους ίδιους του χρήστες, οι 5-11 την αλληλεπίδρασή τους με την εφαρμογή, ενώ οι 12-15 τις παρατηρήσεις των χρηστών.

### **6.1 Προφίλ των Χρηστών**

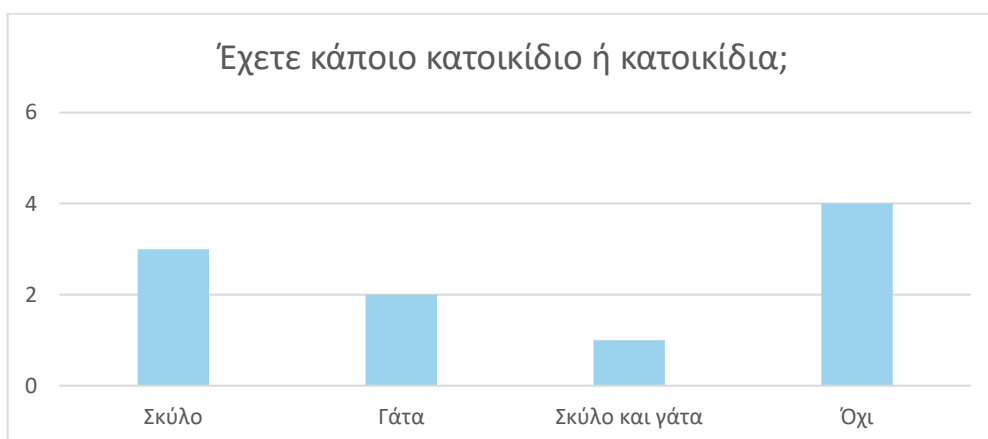
Σε αυτή την ενότητα περιέχονται τα διαγράμματα των απαντήσεων στις ερωτήσεις που κάνουν αντιληπτό το γενικό υπόβαθρο των χρηστών.



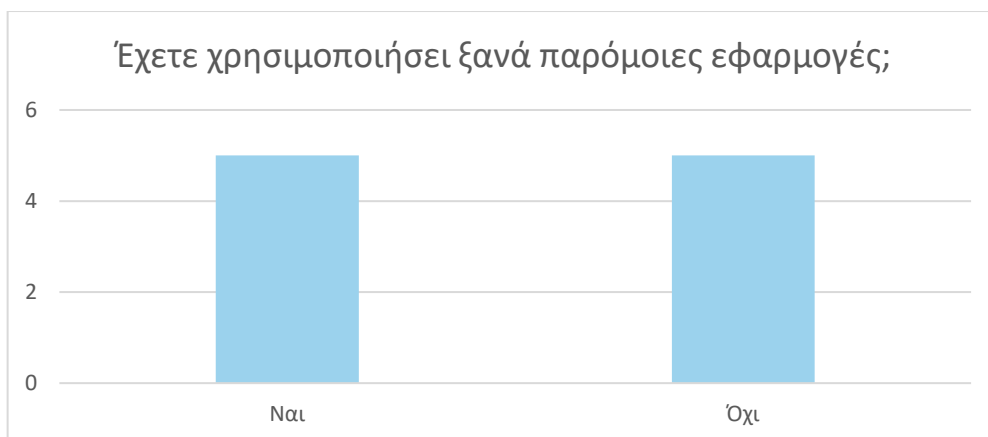
Σχήμα 6.1 Η Ηλικία των Χρηστών



Σχήμα 6.2 Φιλοζωικό Υπόβαθρο



Σχήμα 6.3 Επαφή με Κατοικίδια



Σχήμα 6.4 Χρήση Παρομοίων Εφαρμογών

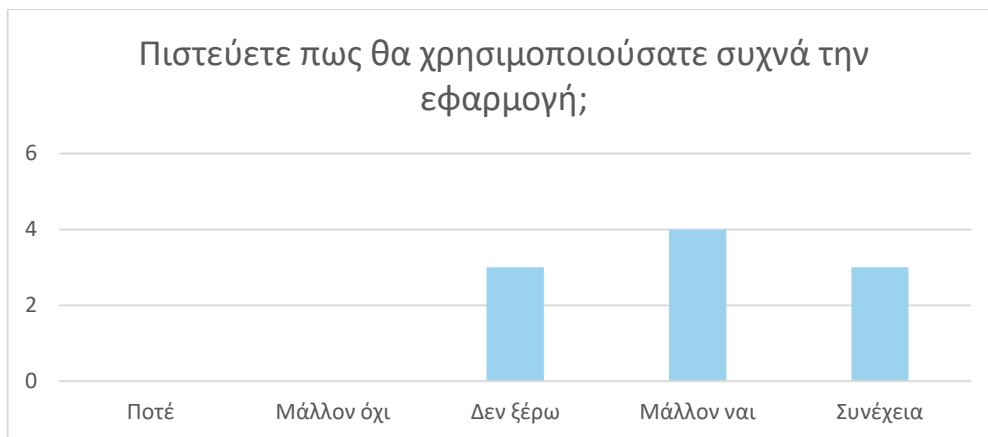
Από τις παραπάνω απαντήσεις μπορούμε να κατανοήσουμε ποιοι ήταν οι χρήστες της εφαρμογής κατά την αξιολόγηση. Κοιτώντας το Σχήμα 6.3, βλέπουμε πως παραπάνω από τους μισούς χρήστες έχουν κάποιο κατοικίδιο, γεγονός που σημαίνει πως είναι πιο ευαισθητοποιημένοι σε φιλοζωικά θέματα. Ακόμα, μπορούμε να συμπεράνουμε πως οι χρήστες που έχουν χρησιμοποιήσει παρόμοιες εφαρμογές συνήθως ανήκουν ήδη σε φιλοζωικούς οργανισμούς.

## 6.2 Ερωτήσεις Αλληλεπίδρασης με την Εφαρμογή

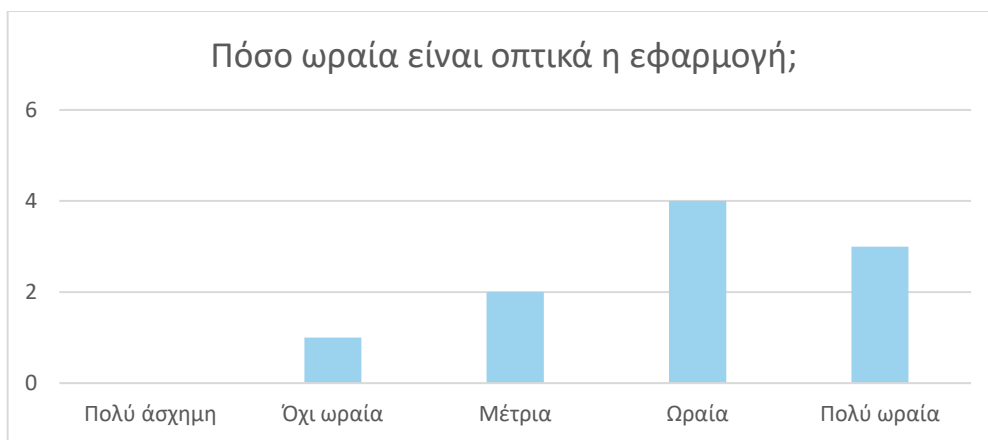
Εδώ παρουσιάζονται οι ερωτήσεις που έγιναν στους χρήστες σχετικά με την εμπειρία τους με την εφαρμογή μας.



Σχήμα 6.5 Ευκολία Χρήσης Εφαρμογής

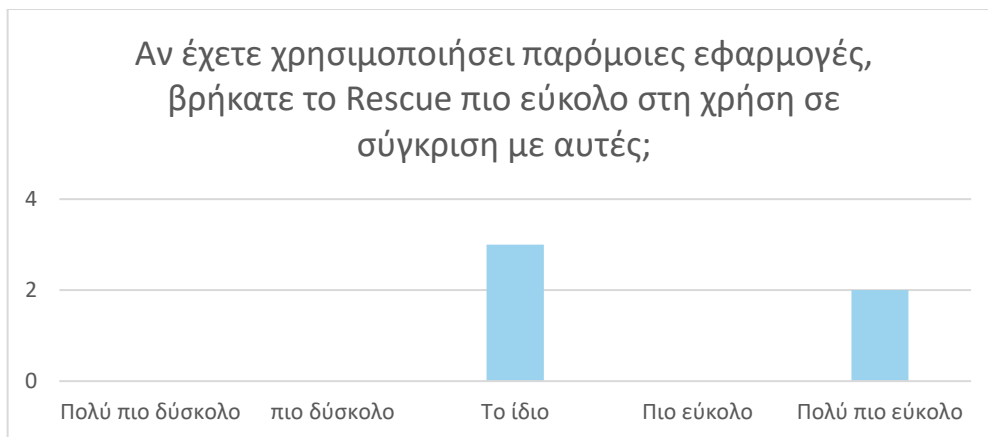


Σχήμα 6.6 Συχνότητα Χρήσης Εφαρμογής

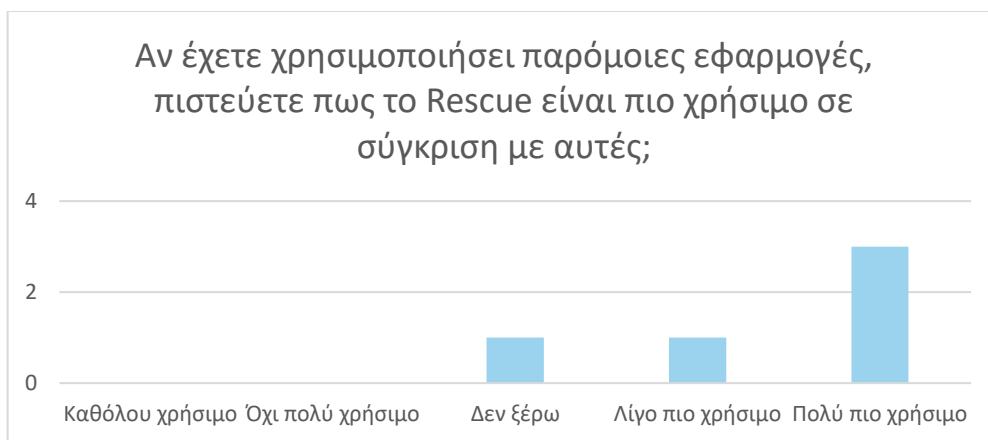


Σχήμα 6.7 Ποιότητα Γραφικής Διεπαφής Χρήστη

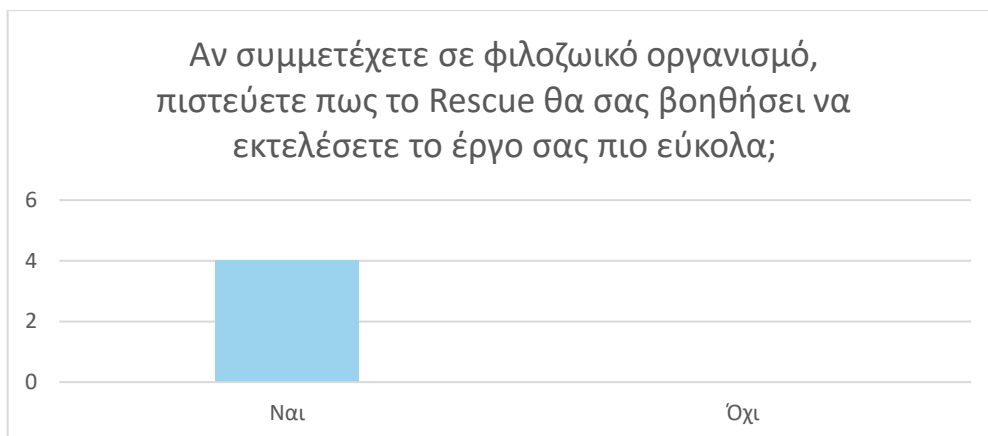
Από το παραπάνω διαγράμματα, μπορούμε ευκολά να συμπεράνουμε πως έχουμε καταφέρει να δημιουργήσουμε μια εφαρμογή εύκολη στη χρήση, όπως μας δείχνει το Σχήμα 6.5. Ακόμα, οι χρήστες δήλωσαν μια καλή συχνότητα χρήσης της εφαρμογής, που δηλώνει πως τους φάνηκε χρήσιμη. Επίσης, μπορούμε να συμπεράνουμε πως η συχνότητα χρήσης της εφαρμογής, επηρεάζεται από το πόσο όμορφη είναι οπτικά, καθώς τα διαγράμματα στα Σχήματα 6.6 και 6.7 φαίνεται να ταυτίζονται σχεδόν. Αυτό το γεγονός μας δείχνει πως με μερικές βελτιώσεις στη γραφική διεπαφή χρήστη, που θα δούμε αναλυτικά παρακάτω, μπορούμε να αυξήσουμε την συχνότητα χρήσης της εφαρμογής.



Σχήμα 6.8 Ευκολία Χρήσης Εφαρμογής Συγκριτικά με Άλλες



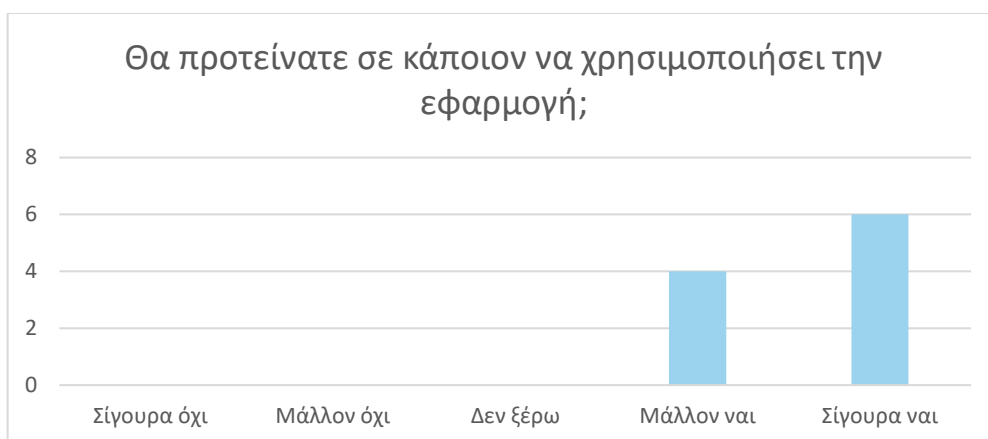
Σχήμα 6.9 Χρησιμότητα Εφαρμογής Συγκριτικά με Άλλες



Σχήμα 6.10 Ικανότητα Εκπλήρωσης Φιλοζωικού Έργου

Οι παραπάνω ερωτήσεις αφορούσαν τους χρήστες που είτε, έχουν χρησιμοποιήσει παρόμοιες εφαρμογές, όπως είδαμε 5 από τους συνολικούς 10 χρήστες, είτε συμμετέχουν και κάποιον φιλοζωικό οργανισμό, 4 από τους 10. Στο Σχήμα 6.8, φαίνεται πως η εφαρμογή μας μπορεί να συγκριθεί ως προς την ευκολία χρήσης της με τις υπόλοιπες παρόμοιες και μάλιστα είναι πιθανό να είναι πιο εύκολη σε ορισμένα θέματα. Επίσης, μπορούμε να συμπεράνουμε βλέποντας το Σχήμα 6.9 πως η εφαρμογή είναι

σίγουρα πιο χρήσιμη σε σύγκριση με τις παρόμοιες που έχουν χρησιμοποιήσει οι χρήστες μας. Επιπρόσθετα, τονίζεται η χρησιμότητα της εφαρμογής και σε εθελοντές φιλοζωικών οργανισμών, αφού όλοι δήλωσαν ότι θα του βοήθαγε στην καλύτερη εκπλήρωση του έργου τους.



Σχήμα 6.11 Πιθανότητα Πρότασης Χρήσης Εφαρμογής

Τέλος, στο Σχήμα 6.11, μπορούμε να διαπιστώσουμε πως είναι πολύ πιθανό οι χρήστες μας να προτείνουν την εφαρμογή σε γνωστούς και φίλους τους, που αποτελεί κυριότατο συστατικό για την αύξηση της φιλοζωικής κοινότητας της εφαρμογής και έτσι να πραγματοποιηθεί η αποτελεσματικότερη εκπλήρωση φιλοζωικών ενεργειών.

## 6.3 Προαιρετικές Ερωτήσεις Ανάπτυξης

Στο τέλος του ερωτηματολογίου υπήρχαν τέσσερις ερωτήσεις ανάπτυξης. Οι απαντήσεις των ερωτήσεων αυτών είναι εξίσου σημαντικές με αυτές των υπόλοιπων ερωτήσεων, καθώς αφορούν τις βελτιώσεις που θα μπορούσαν να γίνουν. Ακόμα συζητήσαμε με τους χρήστες έπειτα από την εμπειρία τους με την εφαρμογή μας, με αποτέλεσμα να μάθουμε πράγματα που ίσως να μην αναφέρονταν διαφορετικά. Τις απαντήσεις στις ερωτήσεις αλλά και τα συμπεράσματα των συζητήσεων τα λάβαμε όλα υπόψιν και θα τα αναλύσουμε στο επόμενο κεφάλαιο.

Οι ερωτήσεις ανάπτυξης ήταν οι εξής:

1. Τι σας άρεσε περισσότερο και βρήκατε πιο χρήσιμο κατά τη χρήση της εφαρμογής;
2. Τι δεν σας άρεσε ή βρήκατε λιγότερο χρήσιμο κατά τη χρήση της εφαρμογής;
3. Υπάρχει κάτι το οποίο θα αλλάζατε ή θα προσθέτατε στην εφαρμογή;
4. Πως θα περιγράφατε τη συνολική σας εμπειρία με την εφαρμογή;

Στην πρώτη ερώτηση οι απαντήσεις που πήραμε έδειξαν πως πέτυχαμε τον αρχικό μας σκοπό, τον οποίο είχαμε θέσει κατά τη διάρκεια ανάλυσης των παρομοίων εφαρμογών.

Φάνηκε πως η εφαρμογή που φτιάξαμε καταφέρνει να ενώσει όλες τις δυνατότητες που επιθυμούν οι χρήστες, τις οποίες μέχρι τώρα τις έβρισκαν σκορπισμένες σε διαφορετικές εφαρμογές. Ακόμα, αξίζει να αναφερθούν οι συγκεκριμένες δυνατότητες που οι χρήστες βρήκαν περισσότερο χρήσιμες και βοήθησαν στην καλύτερη εμπειρία τους με την εφαρμογή. Αρχικά, οι χρήστες συμφωνήσαν πως βρήκαν εύκολο τον τρόπο δημιουργίας νέων δημοσιεύσεων. Έπειτα, δήλωσαν πως βρήκαν αρκετά χρήσιμο το τρόπο που μπορούσαν να βρουν τις δημοσιεύσεις που τους ενδιέφεραν μέσω τις αναζήτησης με φίλτρα και παράλληλα βρήκαν χρήσιμο το πλήθος από πληροφορίες που διέθετε κάθε δημοσίευση. Τέλος, μερικοί σημείωσαν πως βρήκαν πολύ χρήσιμη τη ταξινόμηση των δημοσιεύσεων βάση της απόστασης και γενικότερα τη λειτουργία προβολής μόνο των πλησιέστερων δημοσιεύσεων.

Όσον αφορά τη δεύτερη ερώτηση, που αφορά τα αρνητικά της εφαρμογής, λάβαμε κυρίως κριτικές που αφορούσαν τη γραφική διεπαφή χρήστη και όπως είπαμε και παραπάνω, γνωρίζουμε πως αυτό είναι ένας τομέας της εφαρμογής που με βελτίωση θα βοηθήσει στη μελλοντική ζωή της. Ακόμα, λάβαμε ορισμένα σχόλια που αφορούσαν τη λειτουργία των ειδοποιήσεων καθώς δυο χρήστες θα προτιμούσαν να λάμβαναν και email για κάθε νέα ειδοποίηση. Τελευταία ανησυχία που προέκυψε, ήταν η έλλειψη ιδιωτικών προφίλ για τους χρήστες, κυρίως για λογούς ανωνυμίας και προστασίας προσωπικών δεδομένων.

Σειρά έχει τώρα η τρίτη ερώτηση, που περιλαμβάνει τις προτάσεις των χρηστών για επεκτάσεις και βελτιώσεις. Κυριότερη ανάγκη για την εφαρμογή φάνηκε να είναι η άμεση ιδιωτική συνομιλία μεταξύ χρηστών (chat) και μάλιστα ένα χρήστης σχολίασε πως χωρίς το chat δεν υπάρχει νόημα στις φιλίες μεταξύ χρηστών. Δεύτερη σε σειρά αναγκαιότητα αποδείχθηκε η ανάγκη για τροποποίηση των δημοσιεύσεων για την προσθήκη καινούργιων πληροφοριών. Τελικά, αξίζει να σημειωθεί ένα ενδιαφέρον σχόλιο ενός χρήστη σχετικά με την προβολή εικόνων. Ο χρήστης αυτός εξέφρασε την ανησυχία του σχετικά με εικόνες δημοσιεύσεων που περιέχουν σκληρό περιεχόμενο, θα επιθυμούσε να υπάρχει κάποια δυνατότητα για να έχει την επιλογή για τη προβολή τέτοιων εικόνων. Αυτή η δυνατότητα έχει αποδειχθεί αρκετά διάσημη σε κάθε μέσω κοινωνικής δικτύωσης όπου υπάρχει περιεχόμενο χρηστών και αποτελεί σίγουρα μια από τις μελλοντικές επεκτάσεις.

Με τη τελευταία ερώτηση επιβεβαιώνουμε πως παρά τις ελλείψεις και τις αδυναμίες της εφαρμογής, όλοι οι χρήστες συμφώνησαν πως πρόκειται για μια χρήσιμη εφαρμογή που με μελλοντική εξέλιξη μπορεί να βοηθήσει αρκετό κόσμο και να ενισχύσει την δράση φιλοζωικών οργανισμών.

## **Κεφάλαιο 7. Επίλογος**

Στο παρόν κεφάλαιο θα δοθεί μια συνολική σύνοψη και ταυτόχρονα θα αναλυθούν τα τελικά συμπεράσματα, καθώς και οι μελλοντικές επεκτάσεις και βελτιώσεις.

### **7.1 Σύνοψη και Συμπεράσματα**

Κατά τη διάρκεια αυτής της διπλωματικής, δημιουργήθηκε μια καινούργια εφαρμογή με βασικό σκοπό τη διευκόλυνση επιτέλεσης του φιλοζωικού έργου, τόσο από εθελοντές όσο και από φιλοζωικούς οργανισμούς. Η εφαρμογή βασίστηκε στις ήδη υπάρχουσες εφαρμογές, που αποτελούσαν μια απόπειρα για τη λύση αυτού του προβλήματος, εξαλείφοντας και συνεισφέροντας σε βασικές τους ελλείψεις. Κυριότερη έλλειψη αποδείχθηκε ο διαχωρισμός των διαφορετικών λειτουργιών σε διαφορετικές εφαρμογές, με αποτέλεσμα να μην μπορεί να εκτελεστεί το φιλοζωικό έργο από ένα σημείο, έτσι καταφέραμε με την εφαρμογή που δημιουργήσαμε να ενώσουμε όλες αυτές τις ενέργειες και να προσφέρουμε μια ολιστική και προσβάσιμη λύση για όλους όσους επιθυμούν να συμβάλουν στην ενημέρωση για τις φιλοζωικές ανάγκες και στην ενίσχυση του φιλοζωικού έργου. Από την ανταπόκριση των χρηστών κατά την αλληλεπίδραση τους με την εφαρμογή, μπορούμε να διαπιστώσουμε πως πράγματι φαίνεται να λύνει τα προβλήματα που ήδη αντιμετώπιζαν και οι περισσότεροι πιστεύουν πως η εφαρμογή, με μικρές επεκτάσεις και βελτιώσεις, μπορεί να πέτυχει στον αρχικό της στόχο και να αποτελέσει ένα σημαντικό εργαλείο στη προώθηση φιλοζωικών δράσεων.

### **7.2 Μελλοντικές Επεκτάσεις και Βελτιώσεις**

Στην ενότητα αυτή θα αναλυθούν μερικές προτάσεις για μελλοντικές αλλαγές αλλά και προσθήκες στο Rescue. Ορισμένες από αυτές έχουν προταθεί από τους ίδιους τους χρήστες της εφαρμογής οι οποίο κλήθηκαν να το χρησιμοποιήσουν και να το αξιολογήσουν.



Οι σημαντικότερες βελτιώσεις αφορούν το χρήστη και την καλύτερη εμπειρία του με την εφαρμογή. Αρχικά, αξίζει να ειπωθεί πως μπορεί να βελτιωθεί η διαδικασία εγγραφής στην εφαρμογή, με μια λύση social login[17]. Αυτή η λύση επιτρέπει στους χρήστες να χρησιμοποιήσουν τους ήδη υπάρχοντες λογαριασμούς που διαθέτουν σε άλλα μέσα κοινωνικής δικτύωσης ή πλατφόρμες, με αυτό το τρόπο η εγγραφή στην εφαρμογή μπορεί να γίνει ακόμα πιο εύκολη και άμεση. Ακόμα, σημαντική επέκταση θα αποτελούσε η υποστήριξη διαφορετικών γλωσσών στην εφαρμογή, πραγματικά έτσι θα μπορούσε να επεκταθεί η χρήση της εφαρμογής σε κάθε χώρα και θα ενισχυόταν ακόμα περισσότερο το φιλοζωικό έργο.

Πιστεύουμε πως επίσης, η εφαρμογή θα μπορούσε να βελτιωθεί με μια καλύτερη ενσωμάτωση των λειτουργιών χάρτη. Δηλαδή, να επεκταθούν οι δυνατότητες και ενέργειες που μπορούν να πραγματοποιηθούν μέσα από την εφαρμογή και αφορούν τον χάρτη. Ο σκοπός είναι η εύρεση σημείων φιλοζωικού ενδιαφέροντος όπως για παράδειγμα: κλινικές και ιατρεία ζώων, καταστήματα τροφών και ειδών περιποίησης κατοικίδιων αλλά και σημεία κινδύνου όπου χρήστες θα μπορούσαν να ενημερώνουν για περιοχές με φόλες ή γενικότερα σημεία προς προσοχή.

Βέβαια, πρέπει να αναφερθούν και οι επεκτάσεις που έχουν ζητηθεί από τους χρήστες, κυριότερη των οποίων αποτελεί η ιδιωτική συνομιλία μεταξύ χρηστών. Μια τέτοια δυνατότητα θα διευκολύνει σημαντικά την επικοινωνία των μελών της εφαρμογής και θα επέτρεπε σε αποδοτικότερη οργάνωση δράσεων και ενεργειών εθελοντισμού. Επίσης, σε συνέχεια του σχολίου περί ευαίσθητου περιεχομένου σε εικόνες δημοσιεύσεων στην εφαρμογή, προκύπτει η ανάγκη για κάποιου είδους ελέγχου και έγκρισης περιεχομένου των δημοσιεύσεων. Τελευταία αλλαγή που αξίζει να σημειωθεί είναι η αναβάθμιση του γραφικού περιβάλλοντος, με κυριότερο σκοπό την βελτίωσης της αλληλεπίδρασης του χρήστη με την εφαρμογή. Σε αρκετά σημεία στην εφαρμογή οι χρήστες σχολίασαν τις ελλείψεις που αφορούσαν τη διεπαφή χρήστη, συμπεραίνουμε λοιπόν πως πράγματι χρειάζεται επανασχεδιασμός του γραφικού περιβάλλοντος της εφαρμογής.

Με τις παραπάνω αλλαγές αλλά και ακόμα άλλες που θα προκύψουν σε μελλοντικές αξιολογήσεις θεωρούμε πως το Rescue έχει τη δυνατότητα να βελτιωθεί, να φτάσει όλο και περισσότερους χρήστες και εν τελεί να κάνει τις ζωές των κατοικίδιων μας καλύτερες.

# Αναφορές

- [1] Dogs' Voice Website  
<https://www.dogsvoice.gr/>
- [2] Mytracknet App  
<https://mytracknet.gr/>
- [3] Spot a Stray App  
<https://www.facebook.com/SpotaStray/>
- [4] Your Pet App  
<https://yourpet.app/>
- [5] Figma Design Platform  
<https://www.figma.com/>
- [6] Spring Boot Platform  
<https://spring.io/>
- [7] Spring Documentation for Web MVC  
<https://docs.spring.io/spring-framework/docs/current/reference/html/web.html#spring-web>
- [8] Angular Web Application Framework  
<https://angular.io/>
- [9] Single-Responsibility Principle  
[https://en.wikipedia.org/wiki/Single-responsibility\\_principle](https://en.wikipedia.org/wiki/Single-responsibility_principle)
- [10] REST API  
<https://restfulapi.net/>
- [11] MySQL  
<https://www.mysql.com/>

- [12] Spring Data JPA  
<https://spring.io/projects/spring-data-jpa>
- [13] Spring Documentation for Data Access Object  
<https://docs.spring.io/spring-framework/docs/current/reference/html/data-access.html#dao>
- [14] JSON Web Tokens  
<https://jwt.io/>
- [15] JUnit Testing Framework  
<https://junit.org/junit5/>
- [16] Mockito Framework  
<https://site.mockito.org/>
- [17] Social Login  
[https://en.wikipedia.org/wiki/Social\\_login](https://en.wikipedia.org/wiki/Social_login)