

Ατομική Διπλωματική Εργασία

**ΣΥΣΤΗΜΑ ΥΙΟΘΕΣΙΑΣ ΚΑΤΟΙΚΙΔΙΩΝ ΜΕΣΩ ΕΞΥΠΝΩΝ
ΚΙΝΗΤΩΝ ΣΥΣΚΕΥΩΝ (ANDROID ΚΑΙ IOS ΕΦΑΡΜΟΓΗ)**

Δανήλ Ιωάννου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάρτιος 2017

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Σύστημα υιοθεσίας κατοικίδιων μέσω έξυπνων
κινητών συσκευών(Android και IOS εφαρμογή)**

Δανιήλ Ιωάννου

Επιβλέπων Καθηγητής
Ελπίδα Κεραυνού - Παπαηλίου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των
απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του
Πανεπιστημίου Κύπρου

Μάρτιος 2017

Ευχαριστίες

Θα ήθελα να εκφράσω τις ευχαριστίες μου στην επιβλέπων καθηγήτρια μου κ. Ελπίδα Κεραυνού – Παπαηλίου, για την βοήθεια της, την υποστήριξη της και την πολύ καλή συνεργασία που είχαμε καθ' όλο το χρονικό διάστημα εκπόνησης της διπλωματικής αυτής εργασίας. Οι συμβουλές της, οδηγίες αλλά και κατευθύνσεις της, συνέβαλαν στο μέγιστο, για την επιτυχία κατά την άποψη μου ολοκλήρωση της εργασίας αυτής.

Επίσης θα ήθελα να ευχαριστήσω την 11pets που μου πρόσφερε αυτήν την ευκαιρία και με εμπιστεύτηκε ώστε να υλοποιήσω την ιδέα που είχαν για το σύστημα.

Ακόμη θα ήθελα να ευχαριστήσω θερμά τον Κυριάκο Σταύρου, ένα από τους ιδρυτές της 11pets με τον οποίο είχα στενή επικοινωνία καθ' όλη την διάρκεια της διπλωματικής εργασίας. Επίσης θα ήθελα να εκφράσω τις ευχαριστίες μου, για τη συνεργασία που είχαμε καθ' όλο το χρονικό διάστημα εκπόνησης της διπλωματικής αυτής εργασίας και για τη βοήθεια που μου παρείχε σε όλα τα στάδια ανάπτυξης της εργασίας αυτής.

Τέλος, ευχαριστώ όλα τα καταφύγια που προσφέρθηκαν να δοκιμάσουν τις εφαρμογές δίνοντας feedback για την βελτίωση των εφαρμογών και συνέβαλαν στον καθορισμό των απαιτήσεων.

Περίληψη

Η διπλωματική αυτή εργασία ασχολείται με την ανάπτυξη και επέκταση του οικοσυστήματος της 11Pets στο μέρος που αφορά τις υιοθεσίες κατοικίδιων από τα καταφύγια ζώων, μέσω έξυπνων κινητών συσκευών.

Το θέμα της διπλωματικής εργασίας, είχε σκοπό να αναπτυχθεί ένα σύστημα για την κάλυψη της ανάγκης για μηχανογράφηση των κατοικίδιων που φιλοξενούν τα καταφύγια και οι φιλοζωικές οργανώσεις. Αλλά κυρίως την γρήγορη και αποτελεσματική προώθηση τους προς υιοθεσία για να βρουν ένα καλό σπίτι όσο το δυνατόν πιο νωρίς, μαζεύοντας τα κατοικίδια από όλα τα καταφύγια σε ένα σύστημα ούτως ώστε ο ανάδοχος να μπορεί να τα δει ηλεκτρονικά εύκολα.

Την ευκαιρία να χρησιμοποιήσουν το σύστημα αυτό θα έχουν όλα τα εγκεκριμένα καταφύγια και φιλοζωικές οργανώσεις σε όλο τον κόσμο που θέλουν να συνεργαστούν με την 11pets. Το σύστημα θα παρέχεται αφιλοκερδώς και χωρίς κανένα επιπλέον κόστος. Οι υποψήφιοι ανάδοχοι θα έχουν τη δυνατότητα να χρησιμοποιήσουν το σύστημα χωρίς κανένα κόστος. Οι εφαρμογές θα είναι διαθέσιμες στο play store για Android και του apple store για iOS.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	1
1.1	Σκοπός της Διπλωματικής Εργασίας	1
1.2	Λειτουργικό σύστημα Android	2
1.3	Λειτουργικό σύστημα iOS	3
1.4	Εργαλεία Ανάπτυξης	4
1.5	Μεθοδολογία Ανάπτυξης	4
1.6	Προκαταρκτική έρευνα	5
Κεφάλαιο 2	Απαιτήσεις Συστήματος.....	8
2.1	Εισαγωγή	8
2.2	Γενική Περιγραφή Συστήματος	10
2.3	Συγκεκριμενοποίηση Απαιτήσεων	13
2.4	Ιδιότητες Συστήματος Λογισμικού	15
Κεφάλαιο 3	Προδιαγραφές Συστήματος	19
3.1	Εισαγωγή	20
3.2	Διαγράμματα Ροής Δεδομένων (DFD)	21
3.3	Αντικειμενοστραφής Ανάλυση	28
Κεφάλαιο 4	Σχεδιασμός Συστήματος	37
4.1	Εισαγωγή	37
4.2	Αντικειμενοστραφής Σχεδίαση	38
4.2.1	Interaction Diagrams	38
4.2.2	Αναλυτικό Class Diagram	47
Κεφάλαιο 5	Υλοποίηση Συστήματος.....	48
5.1	Εισαγωγή	48
5.2	Γραφική διεπαφή χρήστη	49
5.3	Υλοποίηση εφαρμογής Android	61
5.4	Υλοποίηση εφαρμογής iOS	70

ΚΕΦΑΛΑΙΟ 6 Συμπεράσματα	79
6.1 Εισαγωγή	79
6.2 Συμπεράσματα	79
6.3 Μελλοντική Εργασία	80
 Βι β λ ι ο γ ρ α φ ί α	 82

Κεφάλαιο 1

Εισαγωγή

1.1 Σκοπός της Διπλωματικής Εργασίας	1
1.2 Λειτουργικό σύστημα Android	2
1.3 Λειτουργικό σύστημα iOS	3
1.4 Εργαλεία Ανάπτυξης	4
1.5 Μεθοδολογία Ανάπτυξης	4
1.6 Προκαταρκτική έρευνα	5

1.1 Σκοπός της Διπλωματικής Εργασίας

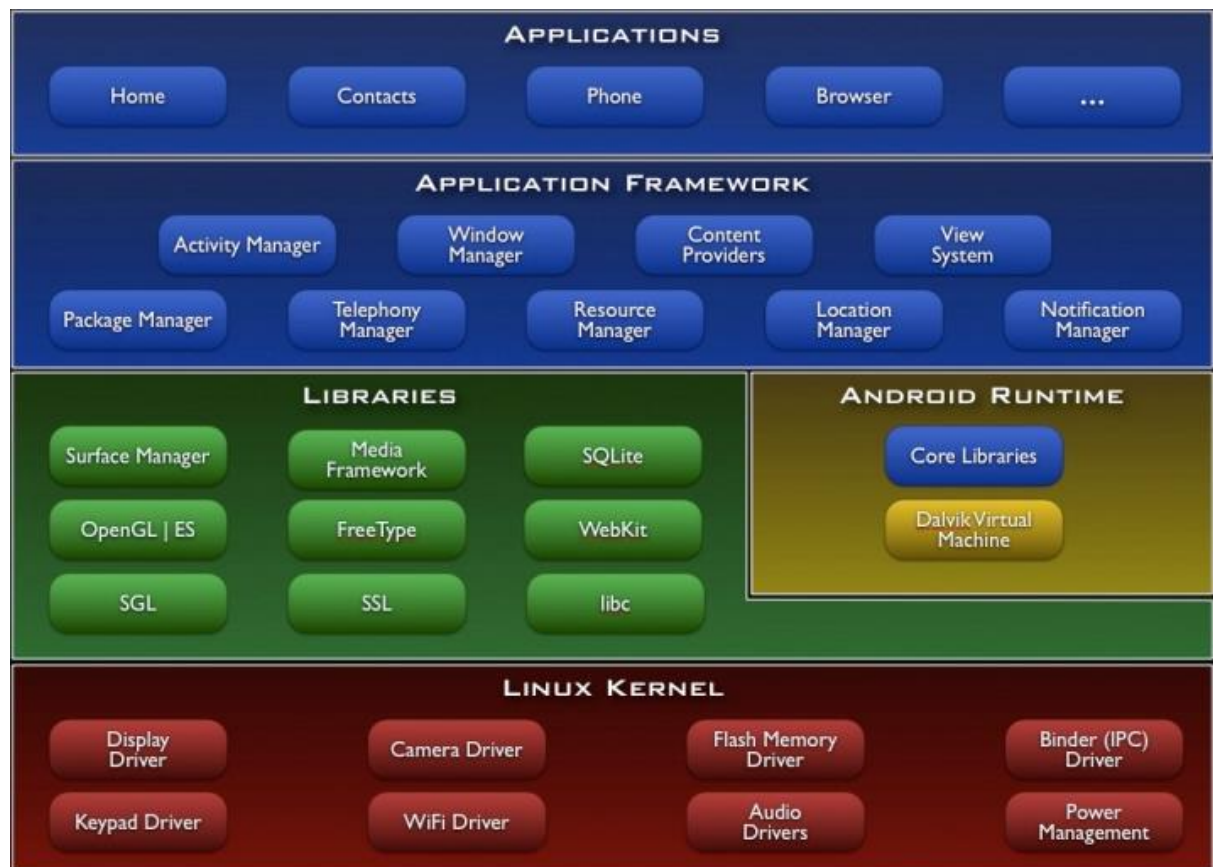
Η διπλωματική εργασία εκπονήθηκε σε συνεργασία με την 11Pets^[1]. Η 11Pets είναι μια βραβευμένη startup με διεθνή δραστηριότητα που αναπτύσσει ένα πλήρες οικοσύστημα με εργαλεία λογισμικού για τον χώρο των κατοικίδιων, που απευθύνονται τόσο για τους ιδιοκτήτες όσο και για τους επαγγελματίες του χώρου και τις φιλοζωικές οργανώσεις. Υπάρχουν διαφορετικά εργαλεία για την κάθε ανάγκη τα οποία όμως είναι σχεδιασμένα για να είναι ομοιογενή και να συνεργάζονται άψογα μεταξύ τους. Η 11Pets έχει ήδη δύο πακέτα στην αγορά και περισσότερους από δεκαπέντε χιλιάδες χρήστες.

Σκοπός της διπλωματικής εργασίας είναι η ανάπτυξη και επέκταση του οικοσυστήματος της 11Pets στο κομμάτι που αφορά τις υιοθεσίες κατοικίδιων από τα καταφύγια ζώων, μέσω έξυπνων κινητών συσκευών. Το σύστημα που θα υλοποιηθεί στο πλαίσιο της διπλωματικής εργασίας θα προσφέρει διάφορες λειτουργίες όπως εισαγωγή νέων κατοικίδιων προς υιοθεσία (από τους εθελοντές των καταφυγίων και φιλοζωικών οργανώσεων που συμμετέχουν στο σύστημα), διαχείριση των ζώων κατά τη διάρκεια που είναι υπό την φροντίδα ενός καταφυγίου ή φιλοζωικής. Επίσης θα προσφέρεται η

δυνατότητα προβολής των προς υιοθεσία κατοικίδιων στους ανάδοχους και η αναζήτηση βάσει προκαθορισμένων κριτηρίων αναζήτησης.

1.2 Λειτουργικό σύστημα Android

Το Android είναι ένα λειτουργικό σύστημα που τρέχει κυρίως σε κινητές συσκευές με οθόνη αφής όπως κινητά τηλέφωνα και tablets. Το Android είναι λειτουργικό σύστημα ανοιχτού κώδικα που ανήκει στην Google^[11]. Λόγω του ότι είναι λογισμικό ανοικτού κώδικα μπορεί να τροποποιηθεί από οποιοδήποτε όπως για παράδειγμα κατασκευαστές συσκευών, εταιρίες τηλεπικοινωνιών και διάφορους developers. Το λειτουργικό είναι σχεδιασμένο χρησιμοποιώντας τις γλώσσες C/C++ και Java, βασίζεται στο Linux και χρησιμοποιεί τον Monolithic kernel (μονολιθικό πυρήνα) που είναι μια τροποποιημένη έκδοση του πυρήνα του λειτουργικού συστήματος Linux (Linux kernel). Ένα επιπλέον πλεονέκτημα του Android είναι πως τρέχει και στις δύο κύριες αρχιτεκτονικές ARM και x86 καθώς επίσης και σε MIPS αρχιτεκτονικές. Στο σχήμα 1 περιγράφεται η αρχιτεκτονική του Android.

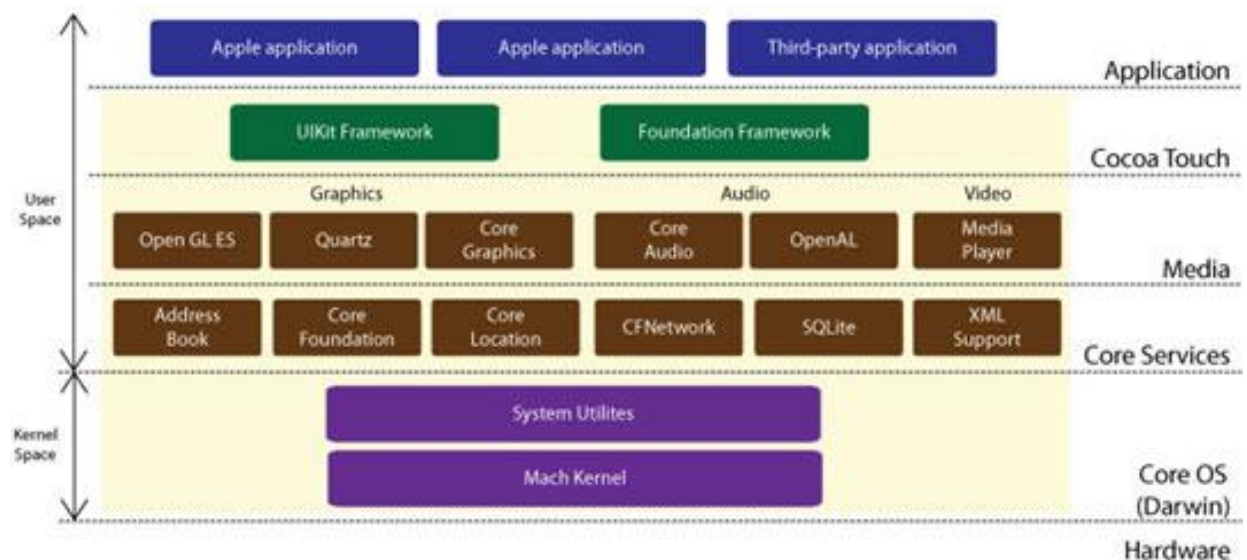


Σχήμα 1: Αρχιτεκτονική Android

Οι εφαρμογές που σχεδιάζονται για το λειτουργικό Android είναι γραμμένες στη γλώσσα προγραμματισμού Java και η Google προτείνει τη σχεδίαση με τη χρήση του εργαλείου Android Studio^[2].

1.3 Λειτουργικό σύστημα iOS

Το iOS^[13,12] (προερχόμενο iPhone OS) είναι ένα κινητό λειτουργικό σύστημα που δημιουργήθηκε και αναπτύχθηκε από την Apple, αποκλειστικά για τα προϊόντα της. Είναι το λειτουργικό σύστημα που τρέχουν πολλές από τις κινητές συσκευές της εταιρείας, συμπεριλαμβανομένων των iPhone, iPad και iPod Touch. Είναι το δεύτερο πιο δημοφιλές κινητό λειτουργικό σύστημα παγκοσμίως μετά το Android. Το iOS τρέχει σε αρχιτεκτονική ARM 64 bit (παλαιότερα 32 bit). Στο σχήμα 2 περιγράφεται η αρχιτεκτονική του iOS.



Σχήμα 2: Αρχιτεκτονική iOS

Οι εφαρμογές που σχεδιάζονται για το λειτουργικό είναι γραμμένες σε Swift^[52] ή σε Objective-C, και για την σχεδίαση τους απαιτείται η χρήση του εργαλείου Xcode^[3] το οποίο τρέχει αποκλειστικά σε λογισμικό macOS.

Το σύστημα επιλέχθηκε να αναπτυχθεί σε Android και iOS λειτουργικό σύστημα κυρίως λόγω του ποσοστού που κατέχουν στην αγορά τα δυο αυτά λειτουργικά συστήματα. Σύμφωνα με τις τελευταίες μελέτες το μεγαλύτερο ποσοστό κινητών συσκευών αυτή τη στιγμή χρησιμοποιεί το λειτουργικό σύστημα Android ή iOS^[62].

1.4 Εργαλεία Ανάπτυξης

Τα εργαλεία που θα χρησιμοποιηθούν για την ανάπτυξη του συστήματος της διπλωματικής εργασίας είναι τα ακόλουθα:

- Για την Android εφαρμογή θα χρησιμοποιεί το Android Studio^[2] με χρήση της γλώσσας Java^[15] και XML για τη διεπαφή της εφαρμογής.
- Για την iOS εφαρμογή θα χρησιμοποιεί το XCODE 8^[3] με χρήση της γλώσσας SWIFT 3^[14].

1.5 Μεθοδολογία Ανάπτυξης

Για την ανάπτυξη του συστήματος θα ακολουθηθεί η μεθοδολογία της Τεχνολογίας Λογισμικού^[16,17]. Συγκεκριμένα το σύστημα θα περάσει από τις τέσσερις φάσεις κύκλου ζωής ενός συστήματος. Οι τέσσερις φάσεις είναι οι εξής :

1η Φάση : Απαιτήσεις Συστήματος

Η πρώτη φάση κύκλου ζωής ενός συστήματος στοχεύει στην εξαγωγή των απαιτήσεων του συστήματος, την ανεύρεση και την καταγραφή όλων των λειτουργικοτήτων που πρέπει να ικανοποιεί. Όπως επίσης και το αναγκαίο λογισμικό και υλικό για να μπορεί να τεθεί η εφαρμογή σε λειτουργία. Σε αυτή τη φάση γίνεται μια ανάλυση της κατάστασης και του τρόπου λειτουργίας των καταφύγιων με σκοπό να βρεθούν και να διαπιστωθούν οι πραγματικές ανάγκες, έτσι ώστε να αναλυθούν λεπτομερώς οι απαιτήσεις του συστήματος.

2η Φάση : Προδιαγραφές συστήματος

Η δεύτερη φάση αφορά τη λεπτομερή ανάλυση των λειτουργιών που καταγράφηκαν στην φάση των απαιτήσεων με διάφορες τεχνικές ανάλυσης.

3η Φάση : Σχεδιασμός συστήματος

Η τρίτη φάση του σχεδιασμού είναι η λεπτομερής ανάλυση του πως θα λειτουργεί το σύστημα, καλύπτοντας όλες τις ανάγκες, απαιτήσεις και προδιαγραφές που εντοπίστηκαν και αναλύθηκαν. Σε αυτή τη φάση αναμένεται να υπάρχει συνεργασία με τον designer της εταιρίας για να καθοριστεί η διεπαφή.

4η Φάση : Υλοποίηση συστήματος

Στη φάση αυτή, αφού έχει σχεδιαστεί το σύστημα που θα υλοποιηθεί, μεταφράζεται η λεπτομερής σχεδίαση σε κώδικα και αναφέρονται τα εργαλεία που χρησιμοποιήθηκαν.

1.6 Προκαταρκτική έρευνα

Χιλιάδες ζώα εγκαταλείπονται από τους ιδιοκτήτες τους. Συνήθως ο λόγος είναι ότι τα βαρέθηκαν οι ιδιοκτήτες τους, ή ανακάλυψαν τι σημαίνει πραγματικά να φροντίζεις ένα ζώο, ή ακόμη γιατί το ζώο δεν ανταποκρίθηκε στις προσδοκίες τους. Αυτό το φαινόμενο δεν παρατηρείται μόνο στην Κύπρο αλλά και σε πολλές χώρες σε ολόκληρο τον κόσμο. Οι φιλοζωικές οργανώσεις και τα καταφύγια ζώων με τα ελάχιστα μέσα που διαθέτουν και με εξαιρετικά δύσκολο οικονομικό κλίμα, προσφέρουν ένα προσωρινό χώρο διαμονής αλλά και μια εξειδικευμένη υπηρεσία στα ζώα αλλά και στην κοινωνία, καταστρώνοντας τα καταφύγια, ασφαλή άσυλα για τα ζώα. Τα καταφύγια στελεχώνονται κυρίως από εθελοντές με μόνο τους κίνητρο την αγάπη για τα ζώα.

Τα καταφύγια ζώων στηρίζουν τις δραστηριότητες τους στην αγάπη και στην εθελοντική δράση των μελών τους, αλλά χωρίς ουσιαστική οικονομική βοήθεια από τους αρμόδιους φορείς και όντας μη κερδοσκοπικοί οργανισμοί το έργο τους γίνεται αυτόματα εξαιρετικά δύσκολο.

Προώθηση κατοικίδιων προς υιοθεσία

Από πλευράς προώθησης κατοικίδιων προς υιοθεσία το κυριότερο πρόβλημα είναι η πολυπλοκότητα στην αναζήτηση. Η κάθε φιλοζωική οργάνωση «προβάλλει» τα ζώα που φιλοξενεί για υιοθεσία στις προσωπικές τους διαδικτυακές σελίδες. Πολλά καταφύγια

δεν έχουν μια προσωπική ιστοσελίδα ή έστω κάποιο χώρο σε μια ιστοσελίδα έτσι ώστε να προωθήσουν τα ζώα που φιλοξενούν με αποτέλεσμα να χρησιμοποιούν, για την προώθηση των ζώων τα μέσα κοινωνικής δικτύωσης τα οποία δεν μπορούν να προσφέρουν εξειδικευμένες υπηρεσίες όπως για παράδειγμα εργαλεία αναζήτησης. Συνεπώς τα ζώα με περισσότερο χρόνο αναμονής δεν έχουν την ίδια προβολή με τα υπόλοιπα ή τα καταφύγια αναγκάζονται να τα αναδημοσιεύουν ανά τακτά χρονικά διαστήματα.

Ο μεγάλος αριθμός των μη κερδοσκοπικών καταφυγίων ζώων και των φιλοζωικών οργανώσεων δυσκολεύει αυτή την αναζήτηση και κάνει την διαδικασία πολύ πιο χρονοβόρα από πλευράς του υποψήφιου αναδόχου. Οι υποψήφιοι ανάδοχοι δεν μπορούν να γνωρίζουν όλες τις φιλοζωικές ομάδες και πολλά από τα διαθέσιμα προς υιοθεσία ζώα μένουν συνεχώς στα αζήτητα.

Πολλά από τα ζώα υιοθετούνται από το εξωτερικό. Όμως η διαδικασία αναζήτησης για ένα ανάδοχο από ξένη χώρα γίνεται ακόμη πιο δύσκολη. Ο ανάδοχος σε αυτήν την περίπτωση έχει να ψάξει σε πολλές χώρες, ακόμη δεν έχει τη δυνατότητα να γνωρίζει όλα τα καταφύγια και φιλοζωικές οργανώσεις. Την αναζήτηση δυσκολεύει ακόμη περισσότερο η γλώσσα, αφού κάθε ιστοσελίδα ή δημοσίευση στα μέσα κοινωνικής δικτύωσης γίνεται στην γλώσσα της χώρας που είναι το εκάστοτε καταφύγιο.

Φροντίδα ζώων στα καταφύγια

Τα ζώα που φιλοξενούνται στα καταφύγια έχουν όλη την προσοχή και αγάπη των εθελοντών που στελεγχώνουν το καταφύγιο. Όμως τα πλείστα καταφύγια δεν έχουν την απαραίτητη τεχνική υποστήριξη. Και χωρίς οικονομική βοήθεια είναι δύσκολο να την αποκτήσουν. Δεν υπάρχει το σύστημα έτσι ώστε όσο τα ζώα τυγχάνουν της φροντίδας των εθελοντών, να κρατείται πλήρες ιστορικό για τα βασικά τουλάχιστο.

Για ένα ζώο που φιλοξενείται στα καταφύγια, θα πρέπει να κρατείται πλήρες ιατρικό ιστορικό, για τις ιατρικές εξετάσεις, την ιατρική φροντίδα και την υγιεινή φροντίδα. Τα υφιστάμενα συστήματα, όπου υπάρχουν, δεν προσφέρουν τις κατάλληλες λειτουργίες.

Υφιστάμενα συστήματα

Υπάρχουν κάποιες εφαρμογές για κινητά τηλέφωνα με σκοπό την προβολή των προς υιοθεσία κατοικίδιων στους ανάδοχους όμως δεν υπάρχουν εφαρμογές για τη διαχείριση των ζώων κατά τη διάρκεια της παραμονής τους στις φιλοζωικές οργανώσεις και τα καταφύγια.

Οι πλείστες εφαρμογές που υπάρχουν διαθέσιμες είναι ελλιπείς και δεν ανταποκρίνονται στις ανάγκες των καταφύγιων ή δεν καλύπτουν όλες τις ανάγκες τους, όπως επίσης εστιάζονται σε κάποια συγκεκριμένη μόνο χώρα.

Οι κυριότερες εφαρμογές που είναι διαθέσιμες σε κινητά τηλέφωνα (Play Store ή apple store) είναι οι ακόλουθες:

PetFinder^[18]: Τα ζώα που είναι προς υιοθεσία προσθέτονται από τους χρήστες της εφαρμογής και όχι από τους επαγγελματίες του χώρου (φιλοζωικές οργανώσεις και τα καταφύγια). Στα αρνητικά συγκαταλέγεται και το γεγονός ότι η εφαρμογή δεν είναι φιλική προς τον χρήστη. Παρόλα αυτά υπάρχουν εκατοντάδες ζώα καταχωρημένα προς υιοθεσία όλα όμως βρίσκονται στην Μάλαισία.

Pet Adoption^[19]: Η εφαρμογή εστιάζεται μόνο στις Ηνωμένες Πολιτείες Αμερικής. Μέσα από την εφαρμογή ο χρήστης μπορεί να βρει τα πιο κοντινά ζώα στην τοποθεσία του κάνοντας αναζήτηση μέσω του ταχυδρομικού κώδικα.

Pet Breeds^[20]: Όπως και η Pet Adoption, και αυτή η εφαρμογή εστιάζεται μόνο στις Ηνωμένες Πολιτείες Αμερικής. Προσφέρεται αναζήτηση μέσω του location της συσκευής.

Adopt Buy Dogs Shop^[21]: Παρομοίως με την εφαρμογή PetFinder τα ζώα που είναι προς υιοθεσία προσθέτονται από τους χρήστες. Η εφαρμογή έχει ζώα που προσφέρονται για υιοθεσία ή για πώληση. Στα αρνητικά της εφαρμογής είναι η υποχρεωτική αναζήτηση που πρέπει να κάνει ο χρήστης για να δει τα ζώα, για παράδειγμα πρέπει να διαλέξει μέγεθος ανάμεσα σε μεγάλο μικρό και μεσαίο χωρίς να μπορεί να δει ζώα και από τις δυο κατηγορίες. Ακόμη ένα αρνητικό δεν υπάρχουν φωτογραφίες των κατοικίδιων.

Adopt a Pet^[22]: Υπάρχουν πάνω από 900 ζώα για υιοθεσία, όμως όλα είναι από την Σλοβακία, χώρα στην οποία εστιάζεται η εφαρμογή.

Κεφάλαιο 2

Απαιτήσεις Συστήματος

2.1 Εισαγωγή	8
2.2. Γενική Περιγραφή Συστήματος	10
2.3 Συγκεκριμενοποίηση Απαιτήσεων	13
2.4 Ιδιότητες Συστήματος Λογισμικού	15

2.1 Εισαγωγή

2.1.1 Σκοπός

Σκοπός του κεφαλαίου αυτού είναι η καταγραφή των απαιτήσεων για το σύστημα που πρόκειται να υλοποιηθεί. Το συγκεκριμένο σύστημα, θα υλοποιηθεί για σκοπούς ολοκλήρωσης της ατομικής διπλωματικής εργασίας που πρέπει να εκπονήσει κάθε φοιτητής της Πληροφορικής κατά το 4ο έτος της φοίτησης του, για την εκπλήρωση των ακαδημαϊκών του υποχρεώσεων. Οι απαιτήσεις αυτές εξάχθηκαν μέσα από συναντήσεις με την επιβλέπουσα καθηγήτρια της εργασίας αυτής, την κα Ελπίδα Κεραυνού Παπαηλίου και τον Κυριάκο Σταύρου Co-founder της 11pets.

2.1.2 Πεδίο

Το λογισμικό που πρόκειται να παραχθεί έχει σκοπό να βοηθήσει τις φιλοζωικές οργανώσεις και τα καταφύγια ζώων να διαχειριστούν τα ζώα που είναι υπό την φροντίδα τους μέσω μιας εφαρμογής στο κινητό τηλέφωνο. Παράλληλα θα προσφέρεται η δυνατότητα προβολής προς υιοθεσία κατοικίδιων στους ανάδοχους. Με το σύστημα αυτό, οι φιλοζωικές οργανώσεις και τα καταφύγια θα μπορούν να φέρουν σε πέρας τις πιο σημαντικές από τις καθημερινές τους λειτουργίες αποκλειστικά και μόνο μέσω της εφαρμογής. Όλες οι καθημερινές τους λειτουργίες μέχρι στιγμής γίνονται από συστήματα που δεν είναι σχεδιασμένα αποκλειστικά για αυτό το σκοπό. Τα δε καταφύγια που συνεργάζονται με την 11pets φέρνουν εις πέρας αυτές τις λειτουργίες αποκλειστικά και μόνο από το web. Την ευκαιρία να χρησιμοποιήσουν το σύστημα αυτό θα έχουν όσα

καταφύγιο το επιθυμούν χωρίς καμία δέσμευση, οικονομική ή οποιαδήποτε άλλη μορφής. Στόχος του συστήματος είναι να επεκταθεί προσελκύοντας όσο το δυνατόν περισσότερο τις φιλοζωικές οργανώσεις τέτοιου είδους ώστε να το χρησιμοποιούν για τις καθημερινές τους ανάγκες.

2.1.3 Ορισμοί, Ακρωνύμια και Συντομογραφίες

Δεν χρησιμοποιήθηκαν στο κείμενο αυτό ορισμοί, ακρωνύμια ή συντομογραφίες.

2.1.4 Επισκόπηση

Στη συνέχεια περιγράφεται αναλυτικά το σύστημα που θα δημιουργηθεί. Συγκεκριμένα, στο δεύτερο μέρος δίνεται μια γενική περιγραφή του συστήματος, χωρίς να μπαίνουμε σε λεπτομέρειες γύρω από τις απαιτήσεις. Θα συζητηθεί η προοπτική του συστήματος, με αναφορά στις διαπροσωπικές του με άλλα συστήματα, με τους χρήστες, με το υλικό και το λογισμικό και με τις επικοινωνίες. Αναφορά θα γίνει, επίσης, σε περιορισμούς σχετικά με τη διαθέσιμη μνήμη, στις λειτουργίες και σε πιθανές απαιτήσεις για νέο υλικό εξοπλισμό που θα υποστηρίξει το παραγόμενο λογισμικό. Θα παρατεθεί λίστα με σύντομη περιγραφή της κάθε λειτουργίας καθώς και τα χαρακτηριστικά των χρηστών. Ακολούθως, θα αναφερθούν στοιχεία που περιορίζουν τον τρόπο που θα σχεδιαστεί και τελικά υλοποιηθεί το σύστημα, παράγοντες που μελλοντικά μπορεί να επηρεάσουν και να αλλάξουν τις απαιτήσεις και τέλος θα αναγνωρισθούν οι λειτουργίες που θα αναβληθούν για μεταγενέστερο στάδιο.

Στο τρίτο μέρος θα γίνει πιο λεπτομερής ανάλυση και περιγραφή όσων αναφέρθηκαν στο δεύτερο μέρος. Αρχικά, θα καταγραφούν οι εξωτερικές διαπροσωπικές απαιτήσεις και θα ομαδοποιηθούν οι λειτουργίες του συστήματος. Ακολούθως, θα αναφερθούν οι χρονικοί περιορισμοί, περιορισμοί σχετικά με την απόδοση του συστήματος, και περιορισμοί όσον αφορά στη σχεδίαση του συστήματος. Επίσης, θα καταγραφούν οι συμπεριφορές του συστήματος, καθώς και οι απαιτήσεις για τη βάση του συστήματος. Τέλος, θα δηλωθούν οποιεσδήποτε άλλες απαιτήσεις.

Στο τέταρτο μέρος, ως παράρτημα, θα παρατεθεί το rapid prototype, το οποίο θα χρησιμεύσει στο να έχει ο πελάτης – σε αυτή την περίπτωση ο επιβλέπων καθηγητής - μια πρώτη επαφή με το σύστημα, ώστε να δώσει την τελική του έγκριση.

2.2 Γενική Περιγραφή Συστήματος

2.2.1 Προοπτικές Προϊόντος

2.2.1.1 Διεπαφές Συστήματος

Το σύστημα θα είναι προσβάσιμο μέσω της εφαρμογής της 11Pets η οποία είναι διαθέσιμη σε Android και iOS. Παρόλα αυτά το σύστημα είναι ανεξάρτητο και δεν συνεργάζεται με οποιοδήποτε άλλη λειτουργικότητα της υπόλοιπης εφαρμογής. Ως λογισμικό δεν θα είναι πλήρως ανεξάρτητο, αφού όλες οι πληροφορίες, τα στοιχεία και τα δεδομένα πρέπει να υπάρχουν σε κάποια βάση δεδομένων έτσι ώστε οποιαδήποτε στιγμή χρειαστεί να μπορούν να ανακληθούν ή να ενημερωθούν. Αυτή η βάση δεδομένων βρίσκεται υλοποιημένη στον Server της εταιρίας.

2.2.1.2 Διεπαφές Χρηστών

Οι χρήστες θα αλληλοεπιδρούν με το σύστημα μέσω διάφορων οθονών. Κάθε οθόνη θα παρουσιάζει στο χρήστη κάποια λειτουργία. Κατ' αρχάς, υπάρχουν δυο ομάδες χρηστών οι ανάδοχοι που θα ψάχνουν ένα κατοικίδιο για να υιοθετήσουν, και τα καταφύγια ζώων που έχουν υπό την προστασία τους τα ζώα.

Ο ανάδοχος αρχικά θα βλέπει μια λίστα με όλα τα κατοικίδια που είναι προς υιοθεσία. Επιλέγοντας ένα από τα κατοικίδια από την λίστα θα εμφανίζονται στο χρήστη όλα τα στοιχεία του επιλεγμένου ζώου.

Κάθε καταφύγιο ανά πάσα στιγμή αφού κάνει login (θα παρουσιάζεται ανάλογο μήνυμα λάθους όταν τα στοιχεία είναι λανθασμένα) θα μπορεί να διαχειριστεί όλα τα ζώα που καταχώρισε στο σύστημα. Όπως επίσης θα έχει τη δυνατότητα να προσθέσει ή να διαγράψει κάποιο. Τα ζώα που καταχωρούνται στο σύστημα δεν προωθούνται αυτόματα για υιοθεσία από το σύστημα αλλά αποθηκεύονται στη βάση δεδομένων για το κάθε καταφύγιο. Το καταφύγιο έχει τη δυνατότητα 'δημοσίευσης'(να γίνει public) τους οποιαδήποτε στιγμή το επιθυμεί.

2.2.1.3 Διεπαφές Υλικού (Hardware)

Το σύστημα θα τρέχει σε λογισμικό android^[11] και iOS^[12,13], και θα είναι διαθέσιμο σε όλες τις συσκευές που υποστηρίζουν android και iOS ανεξαρτήτου έκδοσης λογισμικού. Επίσης πρέπει να ληφθεί υπόψη ότι το σύστημα θα επικοινωνεί με τον web server της εταιρείας στον οποίο θα αποθηκεύονται όλα τα δεδομένα του συστήματος.

2.2.1.4 Διεπαφές Λογισμικού (Software)

Το σύστημα έχει τη μορφή εφαρμογής για κινητά τηλέφωνα έτσι για την ανάπτυξη της Android εφαρμογής είναι απαραίτητη η χρήση του εργαλείου Android Studio^[2] και την υψηλού επιπέδου γλώσσα προγραμματισμού Java^[15] στην 8η έκδοση. Ενώ για την ανάπτυξη της iOS εφαρμογής, απαραίτητη είναι η χρήση του εργαλείου Xcode^[3] με την γλώσσα προγραμματισμού Swift^[14] στην 3η έκδοση.

2.2.1.5 Διεπαφές Επικοινωνίας

Όπως προαναφέρθηκε, το σύστημα αποτελεί εφαρμογή για κινητά τηλέφωνα και θα είναι διαθέσιμο στο play store για Android και στο apple store για iOS. Για την προβολή των προς υιοθεσία κατοικίδιων στους ανάδοχους θα απαιτείται σύνδεση στο διαδίκτυο για να μπορούν να ανακληθούν όλα τα δεδομένα των προς υιοθεσία κατοικίδιων από τον Server της εταιρίας.

2.2.1.6 Περιορισμοί μνήμης

Οι περιορισμοί της μνήμης θα εξαρτώνται από τη συσκευή του χρήστη. Το μεγαλύτερο πρόβλημα θα είναι η διαχείριση των φωτογραφιών που θα προσθέτονται από τους χρήστες. Επίσης κατά την προβολή των κατοικίδιων στους ανάδοχους οι πληροφορίες (περιέχονται και φωτογραφίες) δεν αποθηκεύονται στην κύρια μνήμη της συσκευής αλλά στην προσωρινή μνήμη (cache). Αυτές οι πληροφορίες θα πρέπει να διαχειριστούν με προσοχή λαβαίνοντας υπόψη ότι η προσωρινή μνήμη είναι περιορισμένη στα κινητά τηλέφωνα και διαφέρει από συσκευή σε συσκευή.

2.2.1.7 Διαδικασίες

Το καταφύγιο θα μπορεί να τροποποιήσει ανά πάσα στιγμή οτιδήποτε αφορά στα ζώα που είναι υπό την προστασία του, να προσθέσει καινούρια ή να διαγράψει κάποιο εκτός φυσικά από τα στοιχεία άλλων καταφυγίων.

2.2.1.8 Απαιτήσεις Προσαρμογής Περιοχών

Όπως προαναφέρθηκε, το σύστημα θα είναι διαθέσιμο σε όλες τις συσκευές που υποστηρίζουν android και iOS ανεξαρτήτως έκδοσης λογισμικού. Οι χρήστες θα μπορούν να κατεβάσουν την εφαρμογή μέσω του play store για Android και του apple store για iOS.

2.2.2 Λειτουργίες Συστήματος

Με βάση τις απαιτήσεις, καθορίζονται οι πιο κάτω λειτουργίες προς υλοποίηση:

Καταφύγια:

- Σύνδεση στο λογαριασμό για ελεγχόμενη πρόσβαση των χρηστών.
- Ελεγχόμενη πρόσβαση.
- Δυνατότητα εισαγωγής/τροποποίησης/διαγραφής κατοικίδιων.
- Δυνατότητα εισαγωγής/τροποποίησης/διαγραφής Πακέτων υιοθεσίας.
- Δυνατότητα εισαγωγής/διαγραφής φωτογραφιών για ένα κατοικίδιο.
- Δυνατότητα προβολής κατοικίδιων για υιοθεσία.

Υποψήφιοι ανάδοχοι:

- Εμφάνιση όλων των ζώων προς ενδεχόμενη υιοθεσία
- Προβολή της σελίδας ενός ζώου με όλες τις απαραίτητες πληροφορίες.
- Δυνατότητα αναζήτησης κατοικίδιου με συγκεκριμένα φίλτρα αναζήτησης όπως το είδος (πχ: σκύλος, γάτα κτλ.).
- Δυνατότητα επικοινωνίας με το καταφύγιο που έχει υπό τη φροντίδα του ένα συγκεκριμένο ζώο.
- Δυνατότητα κοινοποίησης ενός προς υιοθεσία ζώου
- Δυνατότητα υποστήριξης-χορηγίας (sponsor) ενός προς υιοθεσία ζώου

2.2.3 Χαρακτηριστικά Χρηστών

Όλοι οι χρήστες που θα χρησιμοποιούν το συγκεκριμένο λογισμικό θα πρέπει να έχουν στη διάθεση τους κινητό τηλέφωνο με λογισμικό Android ή iOS. Οι χρήστες χωρίζονται σε δύο κατηγορίες με κριτήριο το σκοπό της πρόσβασής τους στο σύστημα και τα δικαιώματα τροποποίησης της βάσης. Κατ' αρχάς, έχουμε τους υποψήφιους ανάδοχους. Είναι χρήστες που θα χρησιμοποιήσουν το σύστημα για να βρουν ένα κατοικίδιο για να υιοθετήσουν. Οι υποψήφιοι ανάδοχοι δεν έχουν οποιοδήποτε δικαίωμα τροποποίησης των δεδομένων. Η άλλη κατηγορία χρηστών, είναι τα καταφύγια που συνεργάζονται με την 11pets.

2.2.4 Περιορισμοί

Η εφαρμογή τρέχει σε λογισμικό Android ή iOS σχεδιασμένη με Android και iOS Development Tools^[2,3] με χρήση γλώσσας προγραμματισμού swift^[52] και java.

2.2.5 Υποθέσεις και Εξαρτήσεις

Οι απαιτήσεις του συστήματος μπορούν να επηρεαστούν μερικώς και να χρειαστεί να διαφοροποιηθούν ελαφρώς ανάλογα με το πως θα εξελιχθεί η υλοποίηση του συστήματος στην πορεία.

2.2.6 Διανομή Απαιτήσεων

Δεν αναβάλλεται καμιά από τις απαιτήσεις στο παρόν στάδιο. Σκοπός είναι να υλοποιηθεί ότι ζητήθηκε. Δύναται όμως, αστάθμητοι παράγοντες να εμποδίσουν ή να καθυστερήσουν, στην πορεία, την υλοποίηση κάποιων απαιτήσεων.

2.3 Συγκεκριμενοποίηση Απαιτήσεων

2.3.1 Εξωτερικές Απαιτήσεις Διεπαφών

2.3.1.1 Διεπαφές Χρηστών

Η μορφή της διεπαφής με τους χρήστες παρέχεται στην υποενότητα [2.2.1.2] του κεφαλαίου αυτού.

2.3.1.2 Διεπαφές Υλικού (Hardware)

Δεν υπάρχουν οποιεσδήποτε απαιτήσεις για τη συσκευή στην οποία θα τρέχει η εφαρμογή πάρα μόνο να υποστηρίζει λογισμικό android ή iOS.

2.3.1.3 Διεπαφές Λογισμικού (Software)

Όπως προαναφέρθηκε οι συσκευές στις οποίες θα χρησιμοποιείται το σύστημα θα υποστηρίζουν android ή iOS ανεξαρτήτως έκδοσης λογισμικού.

2.3.1.4 Διεπαφές Επικοινωνίας

Η εφαρμογή όσο αφορά τη λειτουργία της προβολής των προς υιοθεσία κατοικίδιων είναι client/server application. Συνεπώς κάθε συσκευή θα πρέπει να έχει πρόσβαση στο διαδίκτυο για να επικοινωνεί με το web server της 11pets για να μπορεί να ανακτεί όλες τις πληροφορίες για τα κατοικίδια που είναι προς υιοθεσία. Όσο αφορά τις λειτουργίες που σχετίζονται με τα κατοικίδια, χρειάζεται πρόσβαση στο διαδίκτυο για να μπορεί να ενημερώνει τα δεδομένα ενός προς υιοθεσία ζώου.

2.3.1.5 Διεπαφές συστήματος

Μετά από το login ενός χρήστη στο σύστημα, θα μπορεί να χρησιμοποιεί οποιαδήποτε λειτουργία από την εφαρμογή. Για την οποιαδήποτε αλλαγή θα επικοινωνεί και θα ενημερώνει τα δεδομένα στη βάση δεδομένων. Οι υποψήφιοι ανάδοχοι δεν χρειάζεται απαραίτητα να κάνουν login.

2.3.2 Απαιτήσεις Απόδοσης

Το σύστημα, αφού πρόκειται για εφαρμογή για κινητά τηλέφωνα, θα πρέπει να είναι ικανό να παρέχει τη δυνατότητα σε περισσότερους από ένα χρήστες να το χρησιμοποιούν ταυτόχρονα. Θα πρέπει να λάβουμε υπόψη μας το διαθέσιμο χώρο μνήμης, το είδος του software (android ή iOS) και του hardware που έχουμε στη διάθεσή μας για να το χειριστούμε με τον καλύτερο τρόπο και να έχουμε τη βέλτιστη απόδοση που μπορούμε να έχουμε από την αλληλεπίδραση των πιο πάνω.

2.3.3 Περιορισμοί Σχεδιασμού

- Το σύστημα θα αναπτυχθεί χρησιμοποιώντας το εργαλείο Android Studio^[2] για την Android εφαρμογή.
- Για την Android εφαρμογή είναι απαραίτητη η χρήση της γλώσσας Java^[15] και XML για τη διεπαφή της εφαρμογής.
- Η iOS εφαρμογή θα αναπτυχθεί χρησιμοποιώντας το εργαλείο XCODE 8^[3]. Αυτό σημαίνει ότι για τη διαδικασία δημιουργίας του συστήματος θα χρειαστεί apple υπολογιστής καθώς το εργαλείο XCODE τρέχει μόνο σε Mac OS.
- Η iOS εφαρμογή θα μπορεί να αναπτυχθεί με τη γλώσσα προγραμματισμού objective c ή swift^[14]. Προτιμήθηκε η swift καθώς είναι η γλώσσα που προτείνει η ίδια η apple για την ανάπτυξη και παρέχει περισσότερα εργαλεία στο θέμα ασφάλειας.
- Πρέπει να παρέχεται δυο ειδών πρόσβαση στο σύστημα, ανάλογα με την ιδιότητα του χρήστη. Δηλαδή ο υποψήφιος ανάδοχος θα έχει μόνο τη δυνατότητα να δει όλα τα προς υιοθεσία ζώα αλλά όχι να τα τροποποιήσει. Με άλλα λόγια θα έχει μόνο τη δυνατότητα να κάνει ανάκτηση πληροφοριών από το Server. Ακόμη το κάθε καταφύγιο θα πρέπει να μπορεί να τροποποιήσει μόνο της δικές του εγγραφές.
- Το σύστημα θα πρέπει να επικοινωνεί με το Server της εταιρίας αφού εκεί θα είναι αποθηκευμένη η βάση στην οποία φυλάγονται και ανακτώνται τα διάφορα δεδομένα για το σύστημα.

2.4 Ιδιότητες Συστήματος Λογισμικού

2.4.1 Αξιοπιστία

Για μεγαλύτερη αξιοπιστία το σύστημα που θα παραχθεί πρέπει να ελεγχτεί με κατάλληλα test cases^[23,24] ως προς τον κώδικα για αποφυγή σφαλμάτων τα οποία ο χρήστης θα είναι ανίκανος να αντιμετωπίσει και να χειριστεί. Το σύστημα μετά την παράδοσή του πρέπει να παράγει ορθά αποτελέσματα. Επίσης, είναι χρήσιμο να αποκλείσουμε την πιθανότητα ανεξέλεγκτης διακοπής της λειτουργίας του συστήματος σε μη ελεγχόμενη από το χρήστη περίπτωση, δηλ. να αποκλειστεί η περίπτωση κατάρρευσης του συστήματος.

Θα υπάρχουν περισσότερα από ένα είδη προσβασιμότητας ανάλογα με την ιδιότητα του χρήστη, ώστε να μην μπορεί ένας απλός χρήστης (υποψήφιος ανάδοχος) να τροποποιήσει τα αποτελέσματα που βρίσκονται καταχωρημένα στη βάση δεδομένων του συστήματος. Τα καταφύγια ζώων θα έχουν πρόσβαση στην εισαγωγή, τροποποίηση και διαγραφή για διάφορες πληροφορίες, οι οποίες καταχωρήθηκαν από το συγκεκριμένο καταφύγιο που πραγματοποιεί την τροποποίηση. Δεν θα μπορεί να αλλάξει τα δεδομένα (πληροφορίες) που παρήχθησαν από το σύστημα. Τα μηνύματα εξόδου που απευθύνονται στους χρήστες θα πρέπει να είναι καθοδηγητικά για να διασφαλίσουμε πως ο χρήστης θα ξέρει κάθε στιγμή πώς θα χειριστεί το σύστημα για να μην οδηγηθεί σε λάθη.

2.4.2 Διαθεσιμότητα

Θα αναγνωρίζονται οι χρήστες ως προς την ιδιότητα τους, δηλαδή αν είναι υποψήφιοι ανάδοχοι ή φιλοξενική οργάνωση. Το σύστημα θα είναι διαθέσιμο και για τις δυο ομάδες, για οποιαδήποτε στιγμή θελήσουν να επισκεφτούν (ανάδοχοι) ή να επεξεργαστούν και να δημιουργήσουν τα δεδομένα (καταφύγια ζώων). Όπως προαναφέραμε το σύστημα θα είναι διαθέσιμο στο play store για Android και στο apple store για iOS.

2.4.3 Ασφάλεια

Το σύστημα θα προστατεύεται από τους μη εξουσιοδοτημένους χρήστες. Οι ανάδοχοι έχουν τη δυνατότητα να επισκεφτούν μόνο τις δημόσιες πληροφορίες (public) και δεν έχουν τη δυνατότητα να κάνουν οποιαδήποτε τροποποίηση. Κάθε φιλοζωική οργάνωση ή καταφύγιο ζώων που συνεργάζεται με την 11pets θα έχει προσωπικούς κωδικούς πρόσβασης (log in). Επίσης, με το διαχωρισμό των χρηστών σε κατηγορίες προστατεύουμε το σύστημά μας από τους μη επιστημονικά καταρτισμένους χρήστες που αν είχαν στη διάθεσή τους τις βάσεις δεδομένων του συστήματος θα μπορούσαν να δημιουργήσουν κάποιο πρόβλημα. Οι πληροφορίες που καταχωρούνται από τα καταφύγια δεν είναι όλες public αλλά μόνο όσες επέλεξε το καταφύγιο που τις καταχώρισε. Έτσι διαφυλάττονται οι απόρρητες πληροφορίες που περιέχονται στις βάσεις δεδομένων του συστήματος.

2.4.4 Συντηρησιμότητα

Σε ένα σύστημα, σημαντικός παράγοντας είναι η συντηρησιμότητά του, για να μπορεί να επιβιώσει για αρκετό καιρό στις συνεχείς τεχνολογικές εξελίξεις. Θα πρέπει δηλαδή να υπάρχει δυνατότητα αναβάθμισης και προσαρμογής του σε νέα τεχνολογικά δεδομένα ως προς το Hardware και Software. Η ανάγκη αυτή γίνεται πολύ πιο σημαντική όταν πρόκειται για εφαρμογή σε κινητά τηλέφωνα όπου οι αναβαθμίσεις σε Hardware και Software γίνονται με πολύ πιο γρήγορους ρυθμούς.

Το παραχθέν σύστημα θα πρέπει να δημιουργηθεί με βάση τις τεχνικές που διδαχθήκαμε στο μάθημα της Τεχνολογίας Λογισμικού για να δημιουργήσουμε την καλύτερη για μας σύζευξη και συνεκτικότητα, δηλαδή θα πρέπει να επιτύχουμε χαμηλή σύζευξη και υψηλή συνεκτικότητα ανάμεσα στις λογισμικές ενότητες που θα αποτελούν το σύστημα.

Ο κώδικας θα είναι όσο γίνεται πιο κατανοητός για άλλους προγραμματιστές με σχόλια για να υπάρχει δυνατότητα συντήρησης και αναβάθμισης του συστήματος από κάποιους άλλους αργότερα. Επίσης θα επιδιωχθεί η ανάπτυξη επαναχρησιμοποιήσιμων λογισμικών ενοτήτων, τα οποία να μπορούν να αξιοποιηθούν και στο μέλλον.

2.4.5 Φορητότητα

Το σύστημα για να μπορεί να λειτουργήσει απαιτεί κάποιο λειτουργικό σύστημα για Android ή iOS. Η μόνη άλλη απαίτηση που υπάρχει είναι η συσκευή να έχει πρόσβαση στο διαδίκτυο.

Κεφάλαιο 3

Προδιαγραφές Συστήματος

3.1 Εισαγωγή	19
3.2 Διαγράμματα Ροής Δεδομένων (DFD)	20
3.3 Αντικειμενοστραφής Ανάλυση	28
3.3.1 Use Case Diagrams	28
3.3.2 Analysis Class Diagram	31
3.3.3 State Chart diagrams	32

3.1 Εισαγωγή

3.1.1 Σκοπός

Σκοπός του κεφαλαίου αυτού είναι η ανάλυση των απαιτήσεων καθώς και η εξακρίβωση των πραγματικών αναγκών του πελάτη. Μέσα από τη δημιουργία διαφόρων διαγραμμάτων οδηγεί στη συγκεκριμενοποίηση των αναγκών του πελάτη τις οποίες καλούμαστε να καλύψουμε. Όλα τα πιο πάνω συνθέτουν το έγγραφο των τεχνικών προδιαγραφών του συστήματος που πρόκειται να αναπτυχθεί.

Η ανάλυση των απαιτήσεων που έχουν καταγραφεί στο προηγούμενο κεφαλαίο διευκολύνει τη δημιουργία σημαντικών διαγραμμάτων τα οποία θα ακολουθήσουν στο κεφαλαίο αυτό, και στη συνέχεια θα είναι πολύ χρήσιμα για τη σχεδίαση του συστήματος.

Μέσα από τα προαναφερθέντα διαγράμματα (Data Flow Diagram-DFD, Entity Relations Diagram-ERD, Use Case Diagram, Analysis Class Diagram, State Chart Diagram κλπ) συγκεκριμενοποιούνται οι ανάγκες του χρήστη και καταγράφονται με τρόπο που να είναι πιο σαφής, γιατί με τη χρήση της φυσικής γλώσσας που χρησιμοποιήσαμε στο

προηγούμενο κεφαλαίο δεν περιγράφονται τα χαρακτηριστικά του συστήματος με μεγάλη σαφήνεια και ακρίβεια.

Επίσης, μέσα από το Software Project Management Plan (Gantt Chart)^[5] οργανώνεται η μετέπειτα πορεία και τίθεται ένα πρόγραμμα που πρέπει να ακολουθηθεί για την διεκπεραίωση του έργου και την παράδοση του πλήρους λογισμικού εντός συγκεκριμένων χρονικών πλαισίων και περιορισμών. Οι Προδιαγραφές πρέπει να είναι ορθές, πλήρεις και όχι ημιτελείς, και καταγεγραμμένες με βάση το έγγραφο των Απαιτήσεων. Αυτό είναι σημαντικό για την παραγωγή του ορθού λογισμικού και για τη διευκόλυνση των φάσεων που πρόκειται να ακολουθήσουν.

3.1.2 Αποδέκτες

Το σύστημα που θα υλοποιηθεί στα πλαίσια της ατομικής διπλωματικής εργασίας θα υλοποιηθεί για λογαριασμό της 11pets όπως επίσης το κεφαλαίο αυτό, των προδιαγραφών του συστήματος, θα παραδοθεί στα πλαίσια της διπλωματικής εργασίας, και για το λόγο αυτό θα πρέπει να είναι σαφές, χωρίς διφορούμενες έννοιες και πλήρως κατανοητό από τους χρήστες του συστήματος.

3.2 Διάγραμμα ροής Δεδομένων (DFD)

3.2.1 Εισαγωγή στα Διάγραμμα ροής Δεδομένων

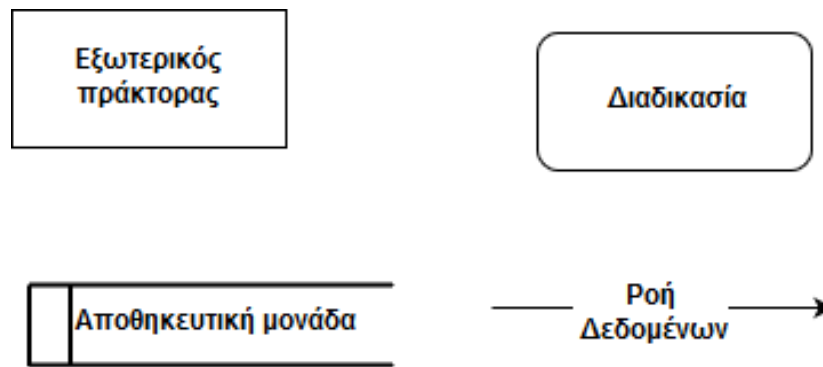
Εξωτερικός πράκτορας: Οι εξωτερικοί πράκτορες αντιπροσωπεύουν τα σημεία αλληλεπίδρασης του συστήματος με το εξωτερικό περιβάλλον και καθορίζουν την εμβέλεια του συστήματος.

Διαδικασία: Οι διαδικασίες αντιπροσωπεύουν τις διάφορες διαδικασίες που εκτελούνται στο σύστημα ή που προκαλούνται από την είσοδο δεδομένων στο σύστημα.

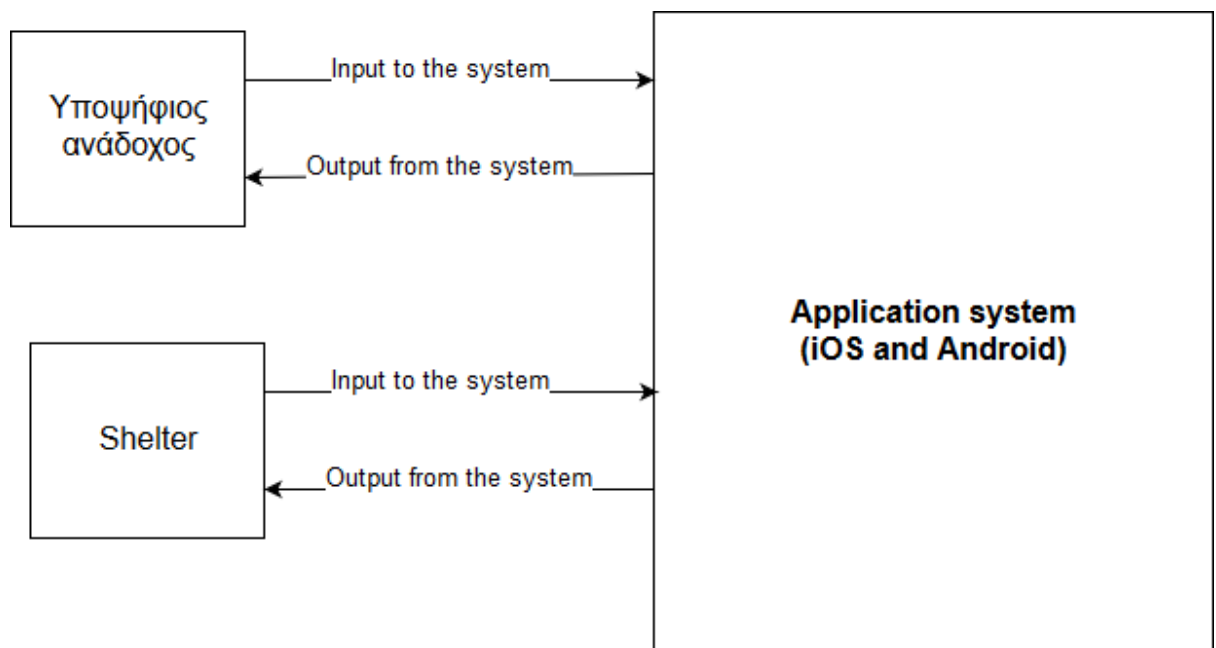
Αποθηκευτική μονάδα: Η αποθηκευτική μονάδα είναι ο χώρος στον οποίο αποθηκεύονται δεδομένα τα οποία χρησιμοποιεί το σύστημα.

Ροή Δεδομένων (βέλη) : Ροή είναι η μεταβίβαση από ένα μέρος του συστήματος σε ένα άλλο μεταφέροντας πληροφορία.

Παρακάτω παρατίθεται ο συμβολισμός για τα σχήματα αυτά:



3.2.2 Level 0

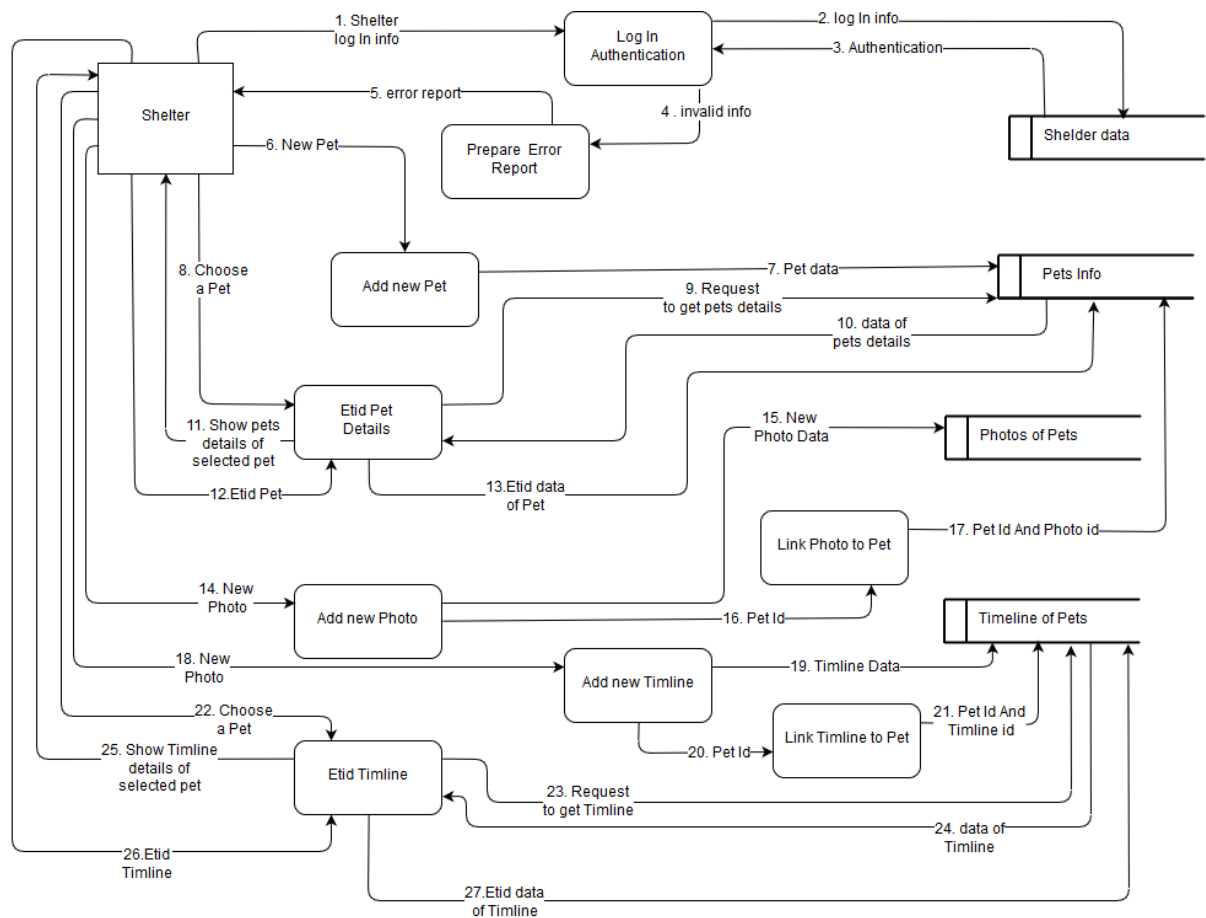


Το σύστημα απαιτεί επικοινωνία με τους χρήστες (υποψήφιους ανάδοχους) που ψάχνουν ένα κατοικίδιο για υιοθεσία και τα καταφύγια (shelter) που φιλοξενούν προσωρινά τα κατοικίδια.

Θα δίνονται δεδομένα εισόδου από τους προαναφερθέντες εξωτερικούς πράκτορες και θα επιστρέφεται έξοδος (χρήσιμες πληροφορίες) από το σύστημα αφού επεξεργαστεί τα δεδομένα εισόδου.

3.2.3 Level 1

3.2.3.1 Καταφύγια ζώων και φιλοζωικές οργανώσεις



Σχήμα 3: Level 1 διάγραμμα ροής δεδομένων για τα
Καταφύγια ζώων και φιλοζωικές οργανώσεις

Το διάγραμμα ροής δεδομένων^[4], που δίνεται στο σχήμα 3, αποτελεί το πρώτο επίπεδο αποσύνθεσης του συστήματος μας για τις λειτουργίες που αφορούν τα καταφύγια ζώων και τις φιλοζωικές οργανώσεις.

Ένα καταφύγιο που συνεργάζεται με την 11pets, δηλαδή είναι καταχωρημένο στο σύστημα, μπορεί να κάνει login στο σύστημα δίνοντας το login information του (ροή 1) στη διεργασία login authentication. Η διεργασία αυτή στέλνει το login information στη βάση δεδομένων (ροή 2) και παίρνει πίσω πληροφορία σχετικά με το authentication του login information αυτού (ροή 3). Αν το login information ήταν λανθασμένο τότε η διεργασία αυτή στέλνει τις πληροφορίες αυτές (ροή 4) στη διεργασία prepare error report. Η διεργασία αυτή ετοιμάζει ένα error report και το στέλνει στο χρήστη (ροή 5). Αν το login γίνει με επιτυχία τότε το καταφύγιο μπορεί να πραγματοποιεί όλες τις λειτουργίες που θα περιγραφούν πιο κάτω.

Ένα καταφύγιο μπορεί να προσθέσει νέα κατοικίδια προς υιοθεσία δίνοντας τις απαραίτητες πληροφορίες για ένα νέο κατοικίδιο (ροή 6). Η διεργασία Add new Pet θα στείλει τις πληροφορίες (ροή 7) στον εξυπηρετητή της εταιρίας για να αποθηκευτούν στην βάση δεδομένων.

Σε περίπτωση που το καταφύγιο θέλει να επεξεργαστεί ένα από τα κατοικίδια που έχει καταχωρημένα στο σύστημα τότε αφού πρώτα επιλέξει το κατοικίδιο (ροή 8) μέσω της διεργασίας Etid Pet Details ανακτώνται όλα τα στοιχεία του συγκεκριμένου κατοικίδιου από την βάση δεδομένων του εξυπηρετητή (από την ροή 9 μέχρι την 10) και προβάλλονται στον χρήστη, δηλαδή καταφύγιο (ροή 11). Ακολούθως δίνοντας τα δεδομένα που επιθυμεί να τροποποιηθούν (ροή 12) αποθηκεύονται στην βάση δεδομένων του εξυπηρετητή (ροή 13).

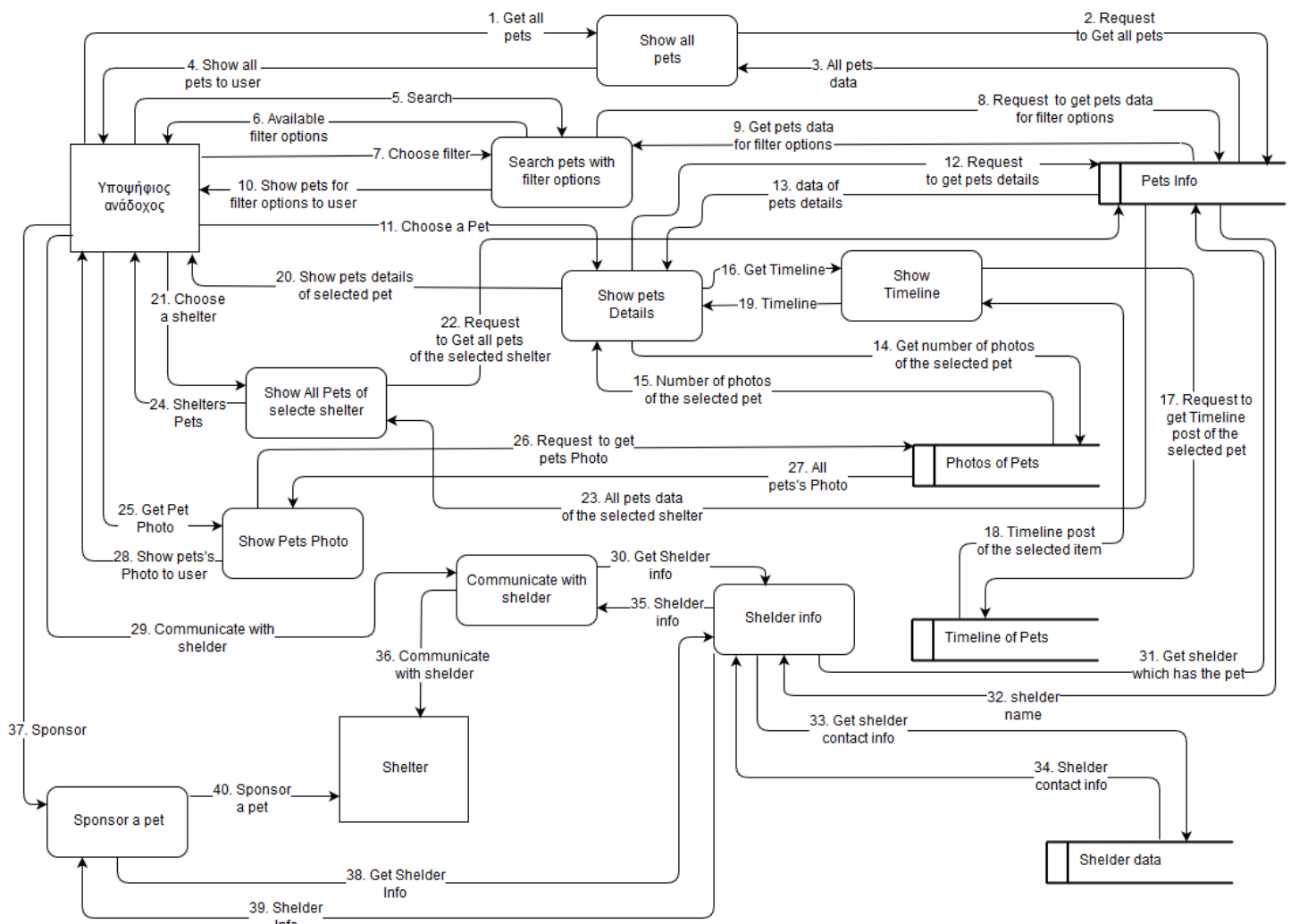
Η διαδικασία προσθήκης μιας νέας φωτογραφίας για ένα κατοικίδιο πραγματοποιείται προσθέτοντας ο χρήστης μια φωτογραφία (ροή 14) και μέσω της διεργασίας Add new Photo θα στείλει την φωτογραφία (ροή 15) στον εξυπηρετητή της εταιρίας για να αποθηκευτούν στην βάση δεδομένων και γίνεται σύνδεση της φωτογραφίας με το συγκεκριμένο κατοικίδιο (διεργασία Link Photo to Pet).

Με πανομοιότυπο τρόπο πραγματοποιείται η προσθήκη μια νέας δημοσίευσης στο χρονολόγιο (Timeline). Πιο συγκεκριμένα προσθέτοντας ο χρήστης μια δημοσίευση (ροή 18) και μέσω της διεργασίας Add new Timeline θα στείλει την δημοσίευση (ροή 19) στον εξυπηρετητή της εταιρίας για να αποθηκευτούν στη βάση δεδομένων και γίνεται σύνδεση

της δημοσίευσης με το χρονολόγιο του συγκεκριμένου κατοικίδιου (διεργασία Link Timeline to Pet).

Σε περίπτωση επεξεργασίας του χρονολογίου ενός κατοικίδιου που είναι καταχωρημένο στο σύστημα τότε αφού πρώτα επιλέξει ο χρήστης το κατοικίδιο (ροή 22) μέσω της διεργασίας Etid Timeline ανακτώνται όλες οι δημοσιεύσεις του συγκεκριμένου κατοικίδιου από την βάση δεδομένων του εξυπηρετητή (από την ροή 23 μέχρι την 24) και προβάλλονται στον χρήστη, (ροή 25). Ακολούθως δίνοντας τα δεδομένα που επιθυμεί να τροποποιηθούν (ροή 26) αποθηκεύονται στην βάση δεδομένων του εξυπηρετητή (ροή 27).

3.2.3.2 Υποψήφιος ανάδοχος



Σχήμα 4: Level 1 διάγραμμα ροής δεδομένων για τον υποψήφιο ανάδοχο

Αυτό το διάγραμμα ροής δεδομένων, που δίνεται στο σχήμα 4, αποτελεί το πρώτο επίπεδο αποσύνθεσης του συστήματος μας για τις λειτουργίες του Υποψήφιου αναδόχου. Ο υποψήφιος ανάδοχος όταν εισέλθει στο σύστημα βλέπει όλα τα κατοικίδια που έχουν καταχωρηθεί από τα καταφύγια. Συγκεκριμένα μέσω ενός REST API που επιτρέπει την πρόσβαση στα συγκεκριμένα δεδομένα Πραγματοποιείται ένα αίτημα (request) προς τον εξυπηρετητή της εταιρίας (ροή 2) όπου βρίσκεται η βάση δεδομένων. Τα δεδομένα που παίρνονται από τον εξυπηρετητή (ροή 3) χρειάζονται επεξεργασία πριν προβληθούν στον χρήστη (ροή 4).

Αν το επιθυμεί ο χρήστης έχει την δυνατότητα να κάνει αναζήτηση (ροή 5) μέσω των συγκεκριμένων φίλτρων αναζήτησης (ροή 6). Αφού επιλέξει τα φίλτρα που επιθυμεί (ροή 7) τότε εμφανίζονται στον χρήστη τα κατοικίδια που ανταποκρίνονται στα φίλτρα αναζήτησης (από την ροή 8 μέχρι την 10), παίρνοντας τα δεδομένα μέσω του rest API που προαναφέρθηκε.

Ο χρήστης μπορεί να επιλέξει οποιοδήποτε από τα κατοικίδια (ροή 11) που του έχουν προβληθεί για να δει περισσότερες πληροφορίες για το κατοικίδιο που επιθυμεί. Το σύστημα μέσω της διεργασίας Show pets Details ανακτά από την βάση δεδομένων όλα τα απαραίτητα δεδομένα (από την ροή 12 μέχρι την 13) για το κατοικίδιο που επέλεξε ο χρήστης. Επιπλέον είναι αναγκαίο ακόμη ένα αίτημα προς τον εξυπηρετητή (από την ροή 14 μέχρι την 15) για τον αριθμό των φωτογραφιών που έχει το κατοικίδιο που επέλεξε ο χρήστης. Επίσης για το χρονολόγιο (Timeline) πραγματοποιείται ακόμη, αίτημα προς τον εξυπηρετητή. Στο χρονολόγιο προβάλλονται δημοσιεύσεις για το συγκεκριμένο κατοικίδιο με χρονολογική σειρά από το πιο πρόσφατο στο πιο «παλιό» (από την ροή 16 μέχρι την 19).

Ακόμη μια δυνατότητα που δίνεται στον χρήστη(ανάδοχο) είναι να δει όλα τα κατοικίδια που φιλοξενεί ένα συγκεκριμένο καταφύγιο. Συγκεκριμένα ο χρήστης όταν επιλέξει ένα κατοικίδιο (ροή 21) ανακτώνται όλα τα κατοικίδια του συγκεκριμένου καταφύγιου από τη βάση δεδομένων του εξυπηρετητή μέσω του rest API που προαναφέρθηκε (από την ροή 22 μέχρι την 24). Υπάρχει η δυνατότητα αν ο χρήστης το επιθυμεί να κάνει αναζήτηση (από την ροή 5 μέχρι την 10) μέσω συγκεκριμένων φίλτρων αναζήτησης στα κατοικίδια που φιλοξενεί ένα συγκεκριμένο καταφύγιο.

Αν ένα κατοικίδιο έχει φωτογραφίες τότε ανακτώνται όλες οι φωτογραφίες του συγκεκριμένου κατοικίδιου από τη βάση δεδομένων του εξυπηρετητή (από την ροή 26 μέχρι την 27) για να προβληθούν στον χρήστη (ροή 28) αν αυτός το επιθυμεί (ροή 25).

Αν ο χρήστης ενδιαφέρεται για ένα κατοικίδιο για να το υιοθετήσει ή ακόμη να πάρει επιπρόσθετες πληροφορίες μέσω του συστήματος, ο χρήστης μπορεί να επικοινωνήσει με το καταφύγιο που φιλοξενεί το κατοικίδιο για το οποίο ενδιαφέρεται είτε μέσω ηλεκτρονικού ταχυδρομείου ή τηλεφωνικώς (από την ροή 29 μέχρι την 36).

Επιπρόσθετα δίνεται η δυνατότητα αν ο χρήστης επιθυμεί να βοηθήσει στη φροντίδα κάποιου κατοικίδιου να αφήσει τα στοιχεία του στο σύστημα (ροή 37) για να αποσταλούν στο καταφύγιο (ροή 40) που φροντίζει το συγκεκριμένο κατοικίδιο ούτως ώστε το καταφύγιο να επικοινωνήσει μαζί του.

3.2.4 Ποιά τμήματα πρέπει να μηχανογραφηθούν και πώς;

Πρόκειται για ένα πλήρως μηχανογραφημένο σύστημα. Το σύστημα θα αναπτυχθεί χρησιμοποιώντας το εργαλείο Android Studio^[2] για την ανάπτυξη της Android εφαρμογής και το εργαλείο Xcode^[3] για την ανάπτυξη της iOS εφαρμογής. Με αυτά τα εργαλεία πρόκειται να συνεχιστεί η διεκπεραίωση του έργου. Το σύστημα πρέπει να έχει τη μορφή εφαρμογής για κινητά τηλέφωνα που έχουν λειτουργικό σύστημα Android ή iOS. Χωρίς ένα από τα δυο λειτουργικά είναι αδύνατη η χρήση του συστήματος. Η δημιουργία κουμπιών μπρος, πίσω, έξοδος κτλ. είναι απαραίτητη στην υλοποίηση του συστήματος. Το σύστημα θα πρέπει να επικοινωνεί με το Server της 11pets μέσω ενός REST API, και θα πρέπει να υπάρχει συνεχής επαφή με τον υπεύθυνο συντήρησης του Server που εργάζεται στην εταιρία, ώστε να αποφευχθούν τυχόν προβλήματα συμβατότητας του συστήματος.

Συγκεκριμένα προδιαγράφονται με σκοπό να υλοποιηθούν τα ακόλουθα μέρη/τμήματα του συστήματος για τα οποία εκκρεμεί η μηχανογράφηση:

Καταφύγια

- Σύνδεση στο λογαριασμό για ελεγχόμενη πρόσβαση των χρηστών.
- Ελεγχόμενη πρόσβαση.
- Δυνατότητα εισαγωγής/τροποποίησης/διαγραφής κατοικίδιων.
- Δυνατότητα εισαγωγής/τροποποίησης/διαγραφής Πακέτων υιοθεσίας.
- Δυνατότητα εισαγωγής/διαγραφής/αντικατάστασης φωτογραφιών για ένα κατοικίδιο.
- Δυνατότητα προβολής κατοικίδιων για υιοθεσία.
- Δυνατότητα εισαγωγής/τροποποίησης/διαγραφής δημοσιεύσεων στο χρονολόγιο (Timeline) ενός κατοικίδιου.

Υποψήφιοι ανάδοχοι

- Εμφάνιση όλων των ζώων προς υιοθεσία
- Προβολή της σελίδας ενός ζώου με όλες τις απαραίτητες πληροφορίες.
- Δυνατότητα αναζήτησης κατοικίδιου με συγκεκριμένα φίλτρα αναζήτησης όπως το είδος (πχ: σκύλος, γάτα κτλ.).
- Δυνατότητα επικοινωνίας τηλεφωνικώς ή μέσω ηλεκτρονικού ταχυδρομείου με το καταφύγιο που έχει υπό την φροντίδα του ένα συγκεκριμένο ζώο.
- Δυνατότητα κοινοποίησης προς υιοθεσία ενός ζώου
- Δυνατότητα υποστήριξης-χορηγίας (sponsor) προς υιοθεσία ενός ζώου
- Δυνατότητα προβολής όλων των κατοικίδιων που φιλοξενεί ένα συγκεκριμένο καταφύγιο.
- Δυνατότητα προβολής όλων των φωτογραφιών ενός συγκεκριμένου κατοικίδιου.

3.2.5 Απαιτήσεις σε Υλικό (Hardware)

Όσον αφορά τις απαιτήσεις σε υλικό, όπως προαναφέρθηκε, το σύστημα θα τρέχει σε οποιοδήποτε ηλεκτρονική συσκευή που υποστηρίζει Android ή iOS λογισμικό

ανεξαρτήτως έκδοσης του λογισμικού συνεπώς το hardware οποιασδήποτε συσκευής που υποστηρίζει ένα από τα δυο λογισμικά είναι επαρκές. Δεν είναι απαραίτητο οι συσκευές, που θα τρέχουν το σύστημα να έχουν κάποια συγκεκριμένη δυνατότητα, οπότε είναι ικανοποιητική η χρήση των ‘μικρών’ σε δυνατότητες συσκευών. Επομένως, μια απλή και συνηθισμένη συσκευή χωρίς ιδιαίτερα χαρακτηριστικά αν είναι συνδεδεμένη στο διαδίκτυο μπορεί να τρέξει το σύστημα.

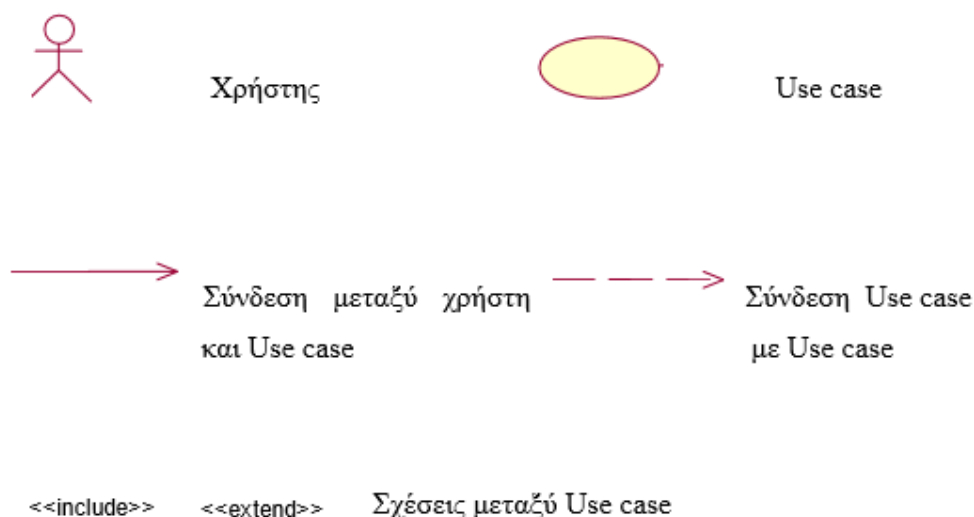
3.3 Αντικειμενοστραφής Ανάλυση

Στη συνέχεια παρουσιάζονται τα βήματα της αντικειμενοστρεφούς ανάλυσης^{[7] [8]}, Use case Diagram, Class Diagram και State chart diagram.

3.3.1 Use Case Diagrams

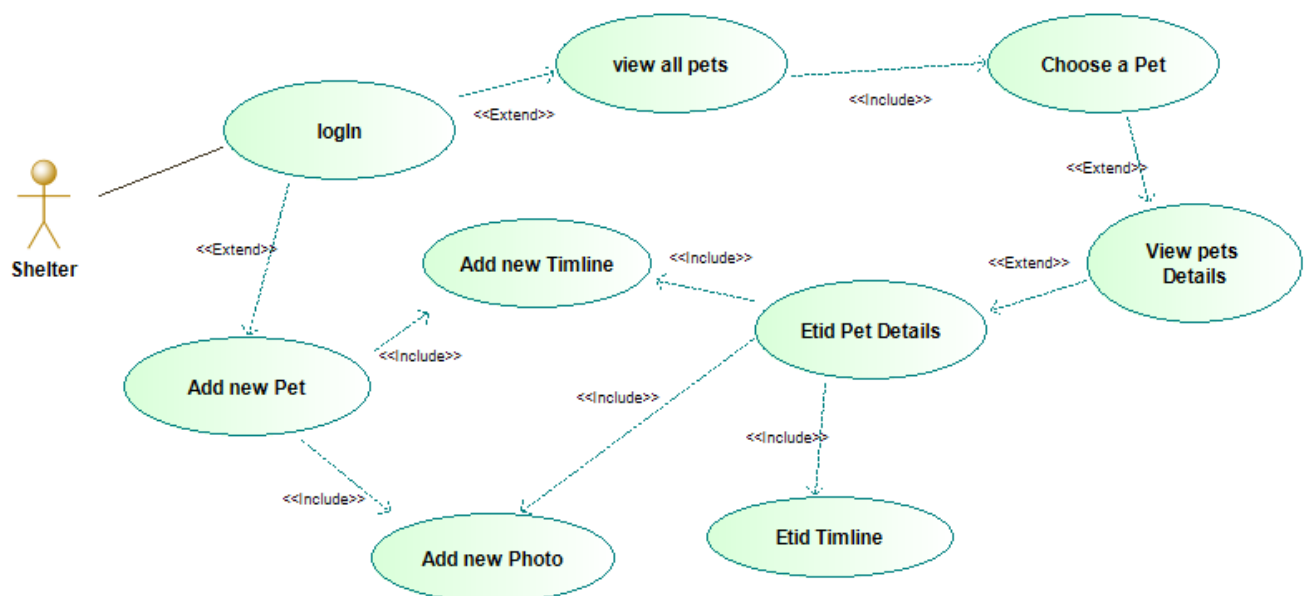
Ένα διάγραμμα σεναρίου χρήσης (use case) χρησιμοποιείται για να αναγνωρίσει τα πρωτεύοντα στοιχεία και διαδικασίες που σχετίζονται με το σύστημα. Τα πρωτεύοντα στοιχεία αναφέρονται ως actors και οι διαδικασίες αναφέρονται ως use cases.

Παρακάτω παρατίθεται ο συμβολισμός για τα σχήματα αυτά:



Οι σχέσεις <<include>> και <<extend>> εμφανίζονται πάνω από τα διακεκομμένα βέλη που ενώνουν δύο use case μεταξύ τους. Η πρώτη σχέση χρησιμοποιείται όταν υπάρχει κοινή συμπεριφορά (λειτουργία) μεταξύ τους και είναι χρήσιμη για περιπτώσεις επαναχρησιμοποίησης του use case. Η δεύτερη χρησιμοποιείται όταν ένα use case μπορεί να ενσωματώνει δύο ή περισσότερα σενάρια ανάλογα με κάποιες συνθήκες και αποτελείται από την κεντρική use case και τις εξαιρέσεις της.

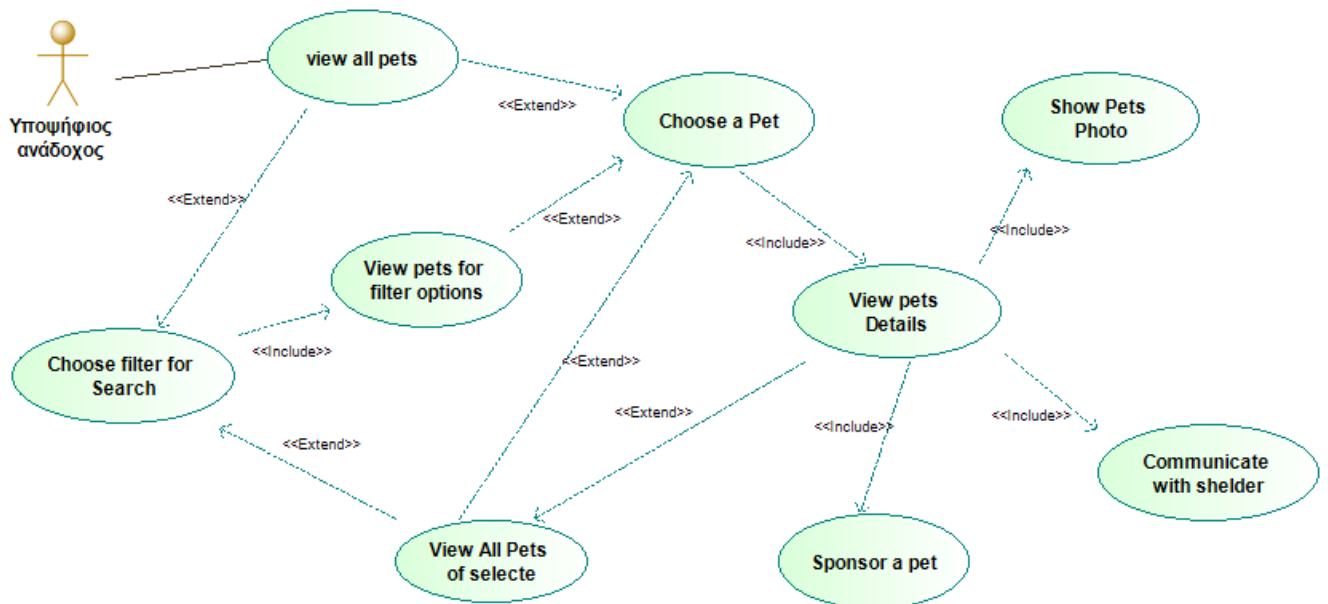
3.3.1.1 Καταφύγια ζώων και φιλοζωικές οργανώσεις



Σχήμα 5: Use Case Diagram για τα καταφύγια ζώων και τις φιλοζωικές οργανώσεις

Όπως διαφαίνεται στο σχήμα 5, ένα καταφύγιο μπορεί να προσθέσει νέα ζώα προς υιοθεσία, αφού κάνει login στο σύστημα. Μπορεί ακολούθως να εισάγει και μετά αν το επιθυμεί να τροποποιήσει μια δημοσίευση στο χρονολόγιο (Timeline) του ζώου που πρόσθεσε. Ακόμη μπορεί να προσθέσει φωτογραφίες σε ένα κατοικίδιο. Οποιαδήποτε στιγμή ο χρήστης έχει την δυνατότητα να επισκεφτεί όλα τα κατοικίδια που έχει προσθέσει και αφού επιλέξει ένα από αυτά μπορεί να δει όλες τις σχετικές πληροφορίες για το συγκεκριμένο ζώο και να τις τροποποιήσει.

3.3.1.2 Υποψήφιος ανάδοχος



Σχήμα 6: Use Case Diagram για
τους υποψήφιους ανάδοχους

Ένα υποψήφιος ανάδοχος έχει την δυνατότητα να επισκεφτεί όλα τα ζώα που είναι προς υιοθεσία (βλέπε σχήμα 6). Ακολούθως αν το επιθυμεί μπορεί να πραγματοποιήσει αναζήτηση μέσα από προκαθορισμένα φίλτρα αναζήτησης. Επιλέγοντας ένα κατοικίδιο από αυτά που του εμφανίζονται έχει τη δυνατότητα να δει όλες τις πληροφορίες για ένα κατοικίδιο σε δομημένη μορφή. Επιπρόσθετα μπορεί να επικοινωνήσει με το καταφύγιο που φιλοξενεί το συγκεκριμένο κατοικίδιο μέσω του συστήματος τηλεφωνικά ή με ηλεκτρονικό ταχυδρομείο για να υιοθετήσει το ζώο ή ακόμη για να πάρει επιπρόσθετες πληροφορίες. Επιπρόσθετα, δίνεται η δυνατότητα να αφήσει ο χρήστης τα στοιχεία του στο σύστημα για να αποσταλούν στο καταφύγιο που φροντίζει το συγκεκριμένο κατοικίδιο ούτως ώστε το καταφύγιο να επικοινωνήσει μαζί του. Τέλος δίνεται η επιλογή στο χρήστη να επισκεφτεί όλα τα ζώα που φιλοξενεί ένα συγκεκριμένο καταφύγιο.

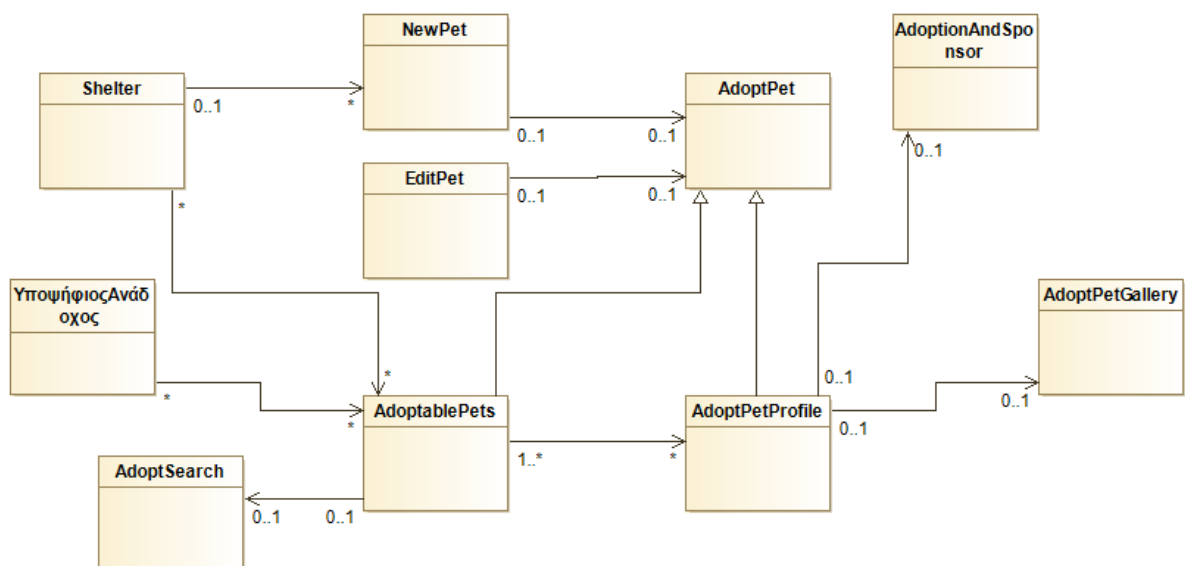
3.3.2 Analysis Class Diagram

Στο βήμα αυτό μοντελοποιείται ο τρόπος λειτουργίας του συστήματος με τη χρήση κλάσεων και σχέσεων μεταξύ τους. Η παρουσίαση γίνεται με το διάγραμμα κλάσεων. Στη φάση αυτή των προδιαγραφών παρουσιάζεται μια γενική μορφή του διαγράμματος κλάσεων. Πιο λεπτομερής και αναλυτική παρουσίαση θα γίνει στο επόμενο κεφάλαιο, της σχεδίασης.

Τα αντικείμενα - σύμβολα που εμφανίζονται στο διάγραμμα είναι τα εξής:



Ακολουθεί το Class Diagrams του συστήματος



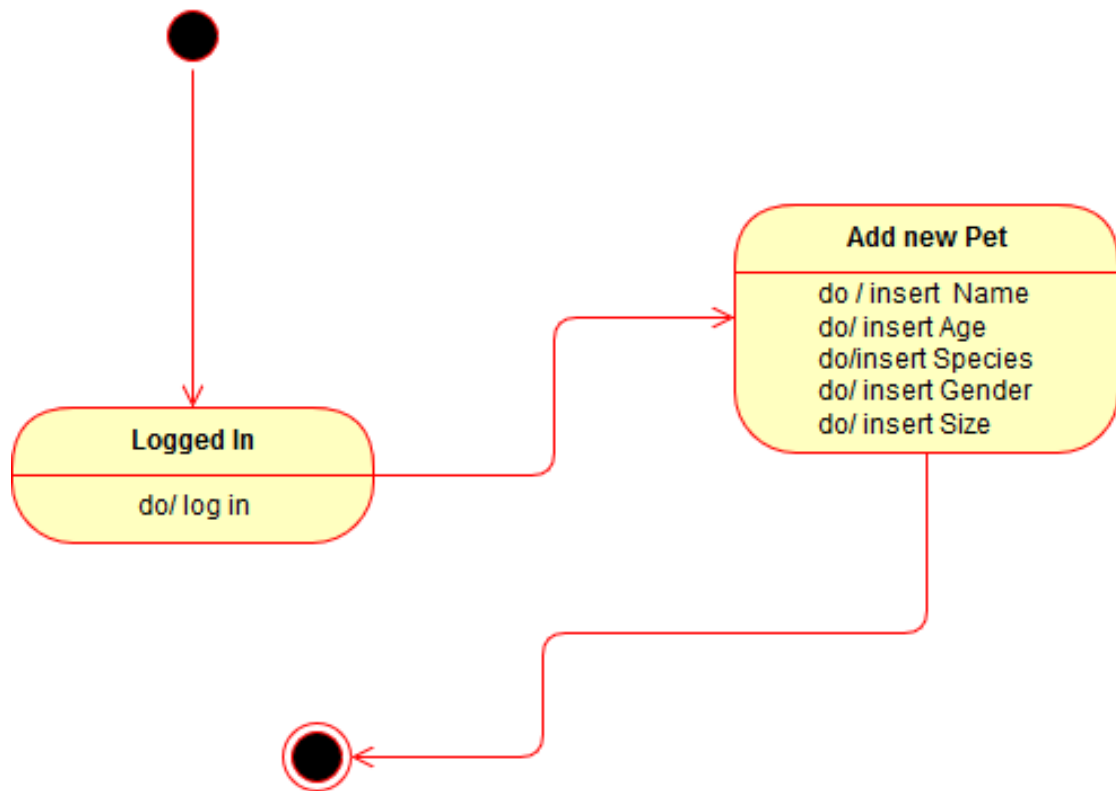
3.3.3 State Chart Diagrams

Το βήμα αυτό μοντελοποιεί τη λειτουργικότητα του συστήματος με αποφάσεις για το ποιες είναι οι ενέργειες που γίνονται από / προς κάθε κλάση. Η παρουσίαση γίνεται με χρήση διαγραμμάτων καταστάσεων.

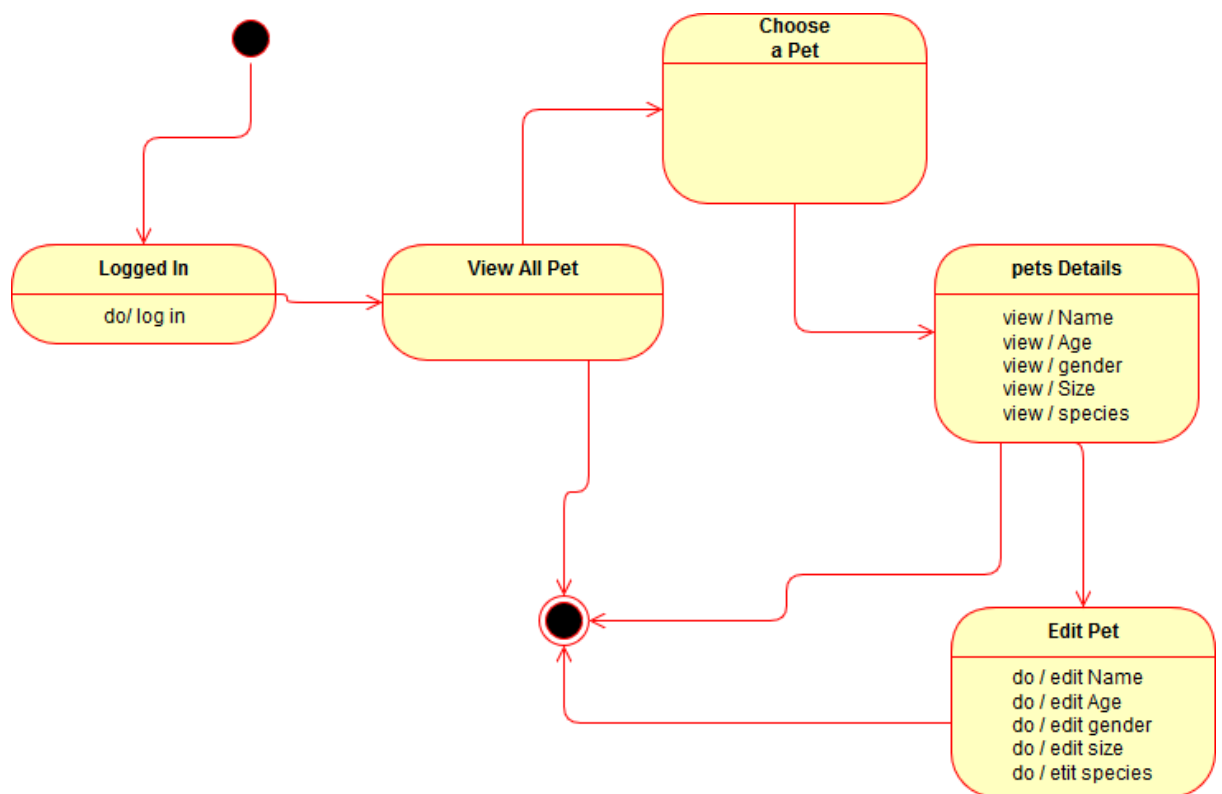
Τα αντικείμενα - σύμβολα που εμφανίζονται στο διάγραμμα είναι τα εξής:



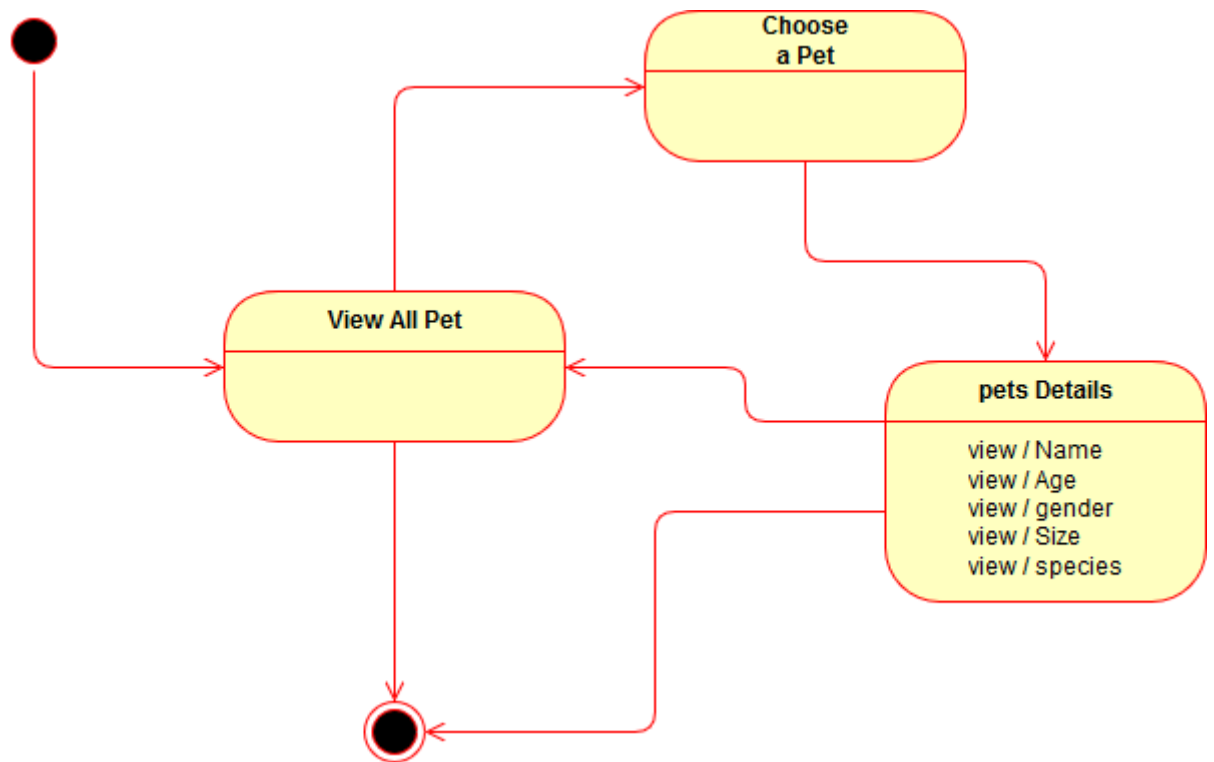
Ακολουθούν τα state diagrams του συστήματος



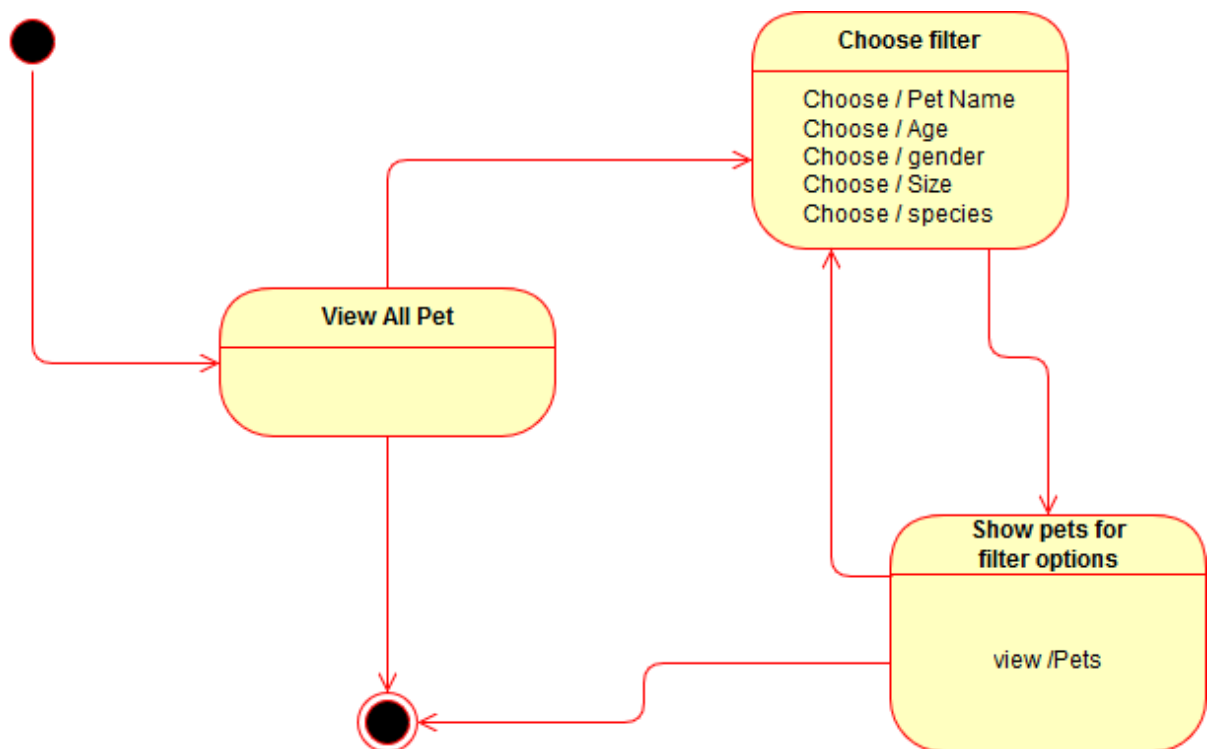
Το πιο πάνω διάγραμμα δείχνει τη δυναμική συμπεριφορά της κλάσης Shelter (καταφύγιο). Η αρχική κατάστασή του είναι η Logged In, κατά την οποία μπαίνει στο σύστημα. Αν επιλέξει να ξεκινήσει τη δημιουργία/ εισαγωγή ενός νέου κατοικίδιου, τότε θα μεταφερθεί στην κατάσταση Add new Pet, κατά την οποία εισάγει όλες τις απαραίτητες πληροφορίες. Όταν η καταχώριση έχει ολοκληρωθεί, τότε μεταφέρεται στην κατάσταση Finish και τερματίζει η καταχώριση.



Το πιο πάνω διάγραμμα δείχνει τη δυναμική συμπεριφορά της κλάσης Shelter (καταφύγιο). Η αρχική κατάσταση του είναι η Logged In, κατά την οποία μπαίνει στο σύστημα. Αν επιλέξει να επισκεφτεί όλα τα ζώα που έχει προσθέσει προηγουμένως στο σύστημα, τότε θα μεταφερθεί στην κατάσταση View All Pet, κατά την οποία μπορεί να τερματίσει το application και να μεταβεί στην κατάσταση Finish ή να επιλέξει ένα ζώο μεταβαίνοντας στην κατάσταση Choose a Pet. Από εκεί μπορεί να δει όλα τα στοιχεία του επιλεγμένου κατοικίδιου (κατάσταση pets Details) και ακολούθως αν το επιθυμεί να τερματίσει το application και να μεταβεί στην κατάσταση Finish ή να τροποποιήσει το κατοικίδιο, μεταβαίνοντας στην κατάσταση Edit Pet, εισάγοντας όλες τις απαραίτητες πληροφορίες. Όταν η τροποποίηση έχει ολοκληρωθεί, τότε μεταφέρεται στην κατάσταση Finish και τερματίζει την τροποποίηση.



Το πιο πάνω διάγραμμα δείχνει τη δυναμική συμπεριφορά της κλάσης Υποψήφιος Ανάδοχος. Η αρχική κατάσταση του είναι η View All Pet, κατά την οποία μπαίνει στο σύστημα. Αν επιλέξει ένα ζώο από αυτά που εμφανίζονται μπροστά του τότε μεταβαίνει στην κατάσταση Choose a Pet. Από εκεί μπορεί να δει όλα τα στοιχεία του επιλεγμένου κατοικίδιου (κατάσταση pets Details) και ακολούθως αν το επιθυμεί να τερματίσει το application και να μεταβεί στην κατάσταση Finish.



Το πιο πάνω διάγραμμα δείχνει τη δυναμική συμπεριφορά της κλάσης Υποψήφιος Ανάδοχος. Η αρχική κατάσταση του είναι η View All Pet, κατά την οποία μπαίνει στο σύστημα. Αν επιλέξει να πραγματοποιήσει αναζήτηση μεταφέρεται στην κατάσταση Choose filter. Επιλέγοντας τα φίλτρα που επιθυμεί για την αναζήτηση τότε, μεταβαίνοντας στην κατάσταση Show pets for filter options, του εμφανίζονται όλα τα κατοικίδια που η περιγραφή τους ταιριάζει σε αυτά τα φίλτρα. Αν το επιθυμεί, τερματίζει το application μεταβαίνοντας στην κατάσταση Finish.

Κεφάλαιο 4

Σχεδιασμός Συστήματος

4.1 Εισαγωγή	37
4.2 Αντικειμενοστραφής Σχεδίαση	38
4.2.1 Interaction Diagrams	38
4.2.2 Αναλυτικό Class Diagram	47

4.1 Εισαγωγή

4.1.1 Σκοπός

Το κεφαλαίο αυτό έχει σαν σκοπό την πιο λεπτομερή περιγραφή των όσων καταγράφηκαν στην φάση των απαιτήσεων και τα περιγράψαμε στη φάση των προδιαγραφών. Μέσα από αυτό το έγγραφο θα σχεδιαστεί το σύστημα το οποίο πρόκειται να υλοποιηθεί έχοντας πάντα κατά νου πως ο προγραμματιστικός στόχος είναι να επιτευχθεί όσο το δυνατό υψηλή συνεκτικότητα καθώς και χαμηλή σύζευξη ανάμεσα στα modules, στα οποία πρόκειται να υλοποιηθούν οι λειτουργίες του συστήματος. Έχοντας ένα πλήρες και τεκμηριωμένο έγγραφο για τη φάση της σχεδίασης διευκολύνεται η φάση της υλοποίησης, όπου όλα αυτά που έχουν καταγραφεί, μελετηθεί και περιγραφεί μέχρι στιγμής θα μετατραπούν στο παραχθέν σύστημα.

4.1.2 Αποδέκτες

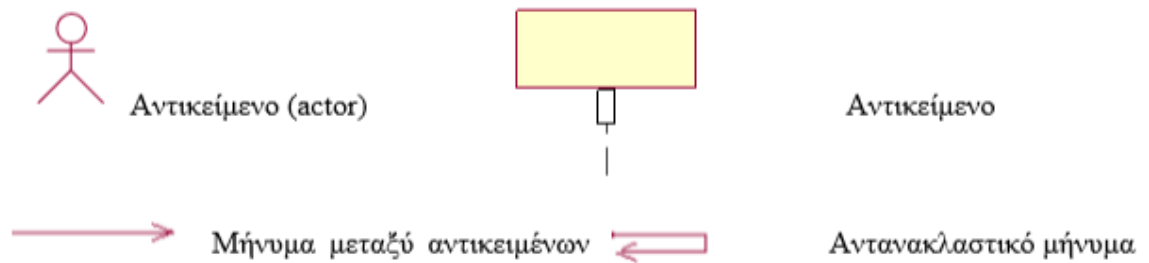
Το έγγραφο αυτό, απευθύνεται σε όποιον ενδιαφέρεται να καταλάβει πως λειτουργεί το σύστημα και στους μελλοντικούς συντηρητές του συστήματος, οι οποίοι πρόκειται να κληθούν να καταλάβουν πλήρως τον τρόπο με τον οποίο έχει σχεδιαστεί και υλοποιηθεί το σύστημα. Το έγγραφο αυτό πρέπει να παρέχει τις απαραίτητες πληροφορίες που χρειάζονται τα άτομα που θα κληθούν να συντηρούν και να βελτιώνουν το σύστημα. Θα πρέπει να είναι σαφές, χωρίς διφορούμενες έννοιες και πλήρως κατανοητό από τους μελλοντικούς χρήστες του συστήματος, καθώς και πλήρως τεκμηριωμένο με σκοπό να διευκολυνθούν τα άτομα που θα αναλάβουν την αναβάθμιση του συστήματος στο μέλλον.

4.2 Αντικειμενοστραφής Σχεδίαση

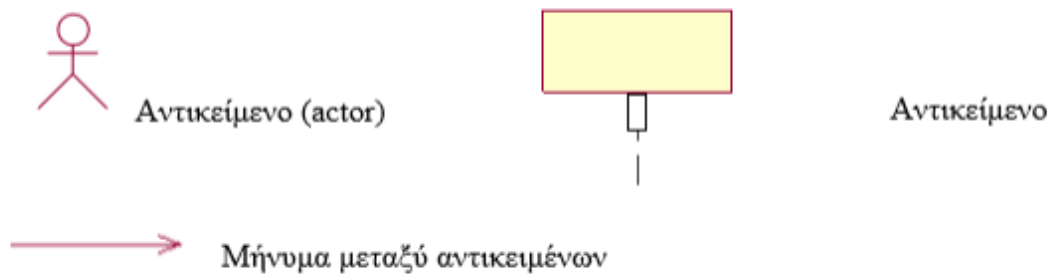
4.2.1 Interaction Diagrams

Τα διαγράμματα αλληλεπίδρασης παρουσιάζουν την αλληλεπίδραση μεταξύ αντικειμένων με ανταλλαγή μηνυμάτων. Υπάρχουν δύο είδη διαγραμμάτων αλληλεπίδρασης τα οποία παρουσιάζουν το ίδιο πράγμα αλλά με διαφορετικό τρόπο: τα ακολουθιακά διαγράμματα και τα διαγράμματα συνεργασίας. Τα ακολουθιακά διαγράμματα δίνουν έμφαση στην ακριβή χρονολογική ακολουθία των μηνυμάτων που ανταλλάσσονται μεταξύ των αντικειμένων και είναι χρήσιμα για την παρουσίαση της σειράς εμφάνισης των γεγονότων. Εξελίσσονται από πάνω προς τα κάτω. Τα συνεργασιακά διαγράμματα είναι διαγράμματα αλληλεπίδρασης που παρουσιάζουν τη ροή διαμέσου ενός συγκεκριμένου σεναρίου. Αυτά εστιάζονται περισσότερο στις σχέσεις μεταξύ των αντικειμένων.

Ο συμβολισμός των ακολουθιακών διαγραμμάτων φαίνεται πιο κάτω:

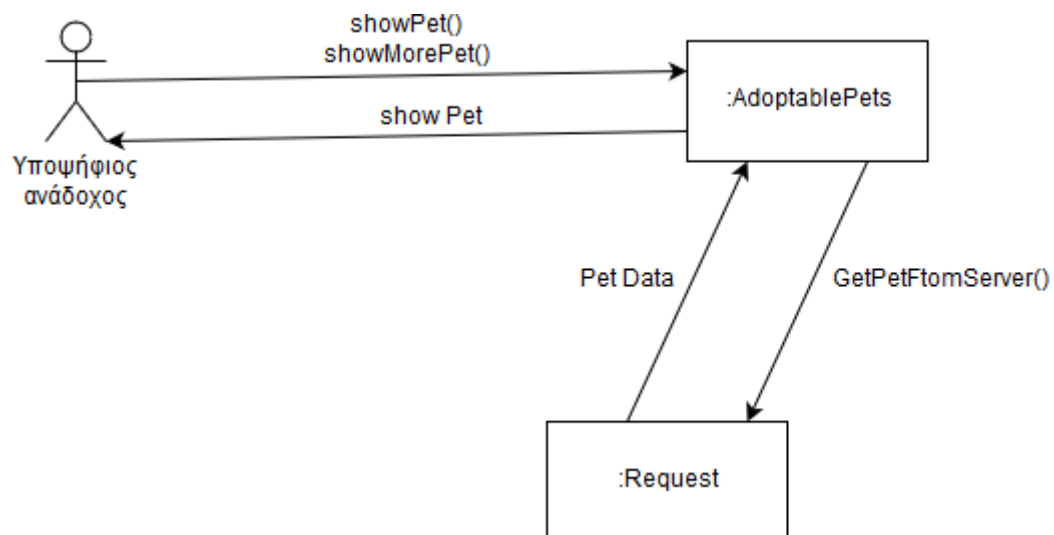
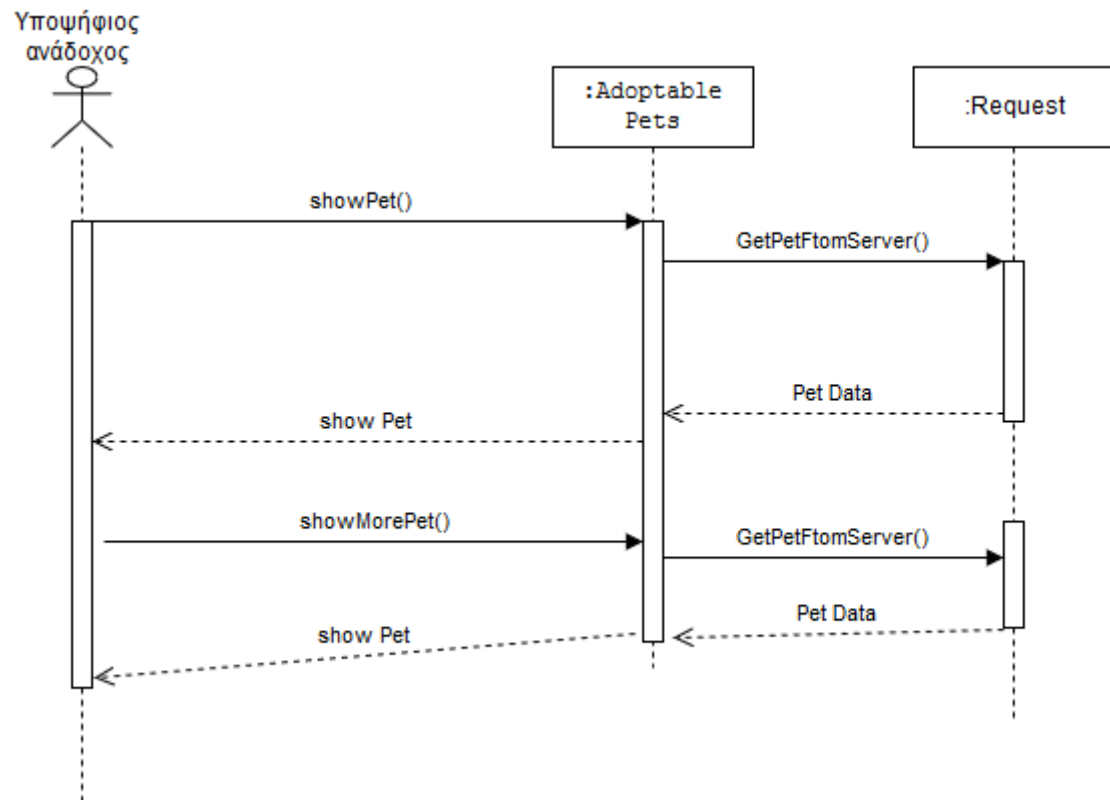


Ο συμβολισμός των συνεργασιακών διαγραμμάτων φαίνεται πιο κάτω:

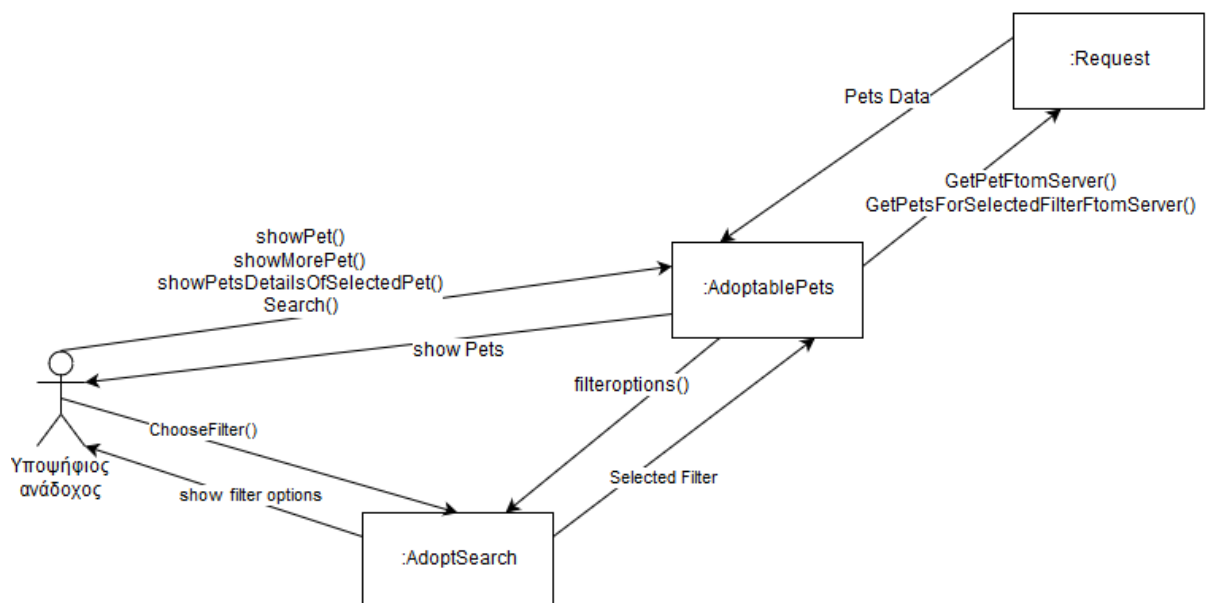
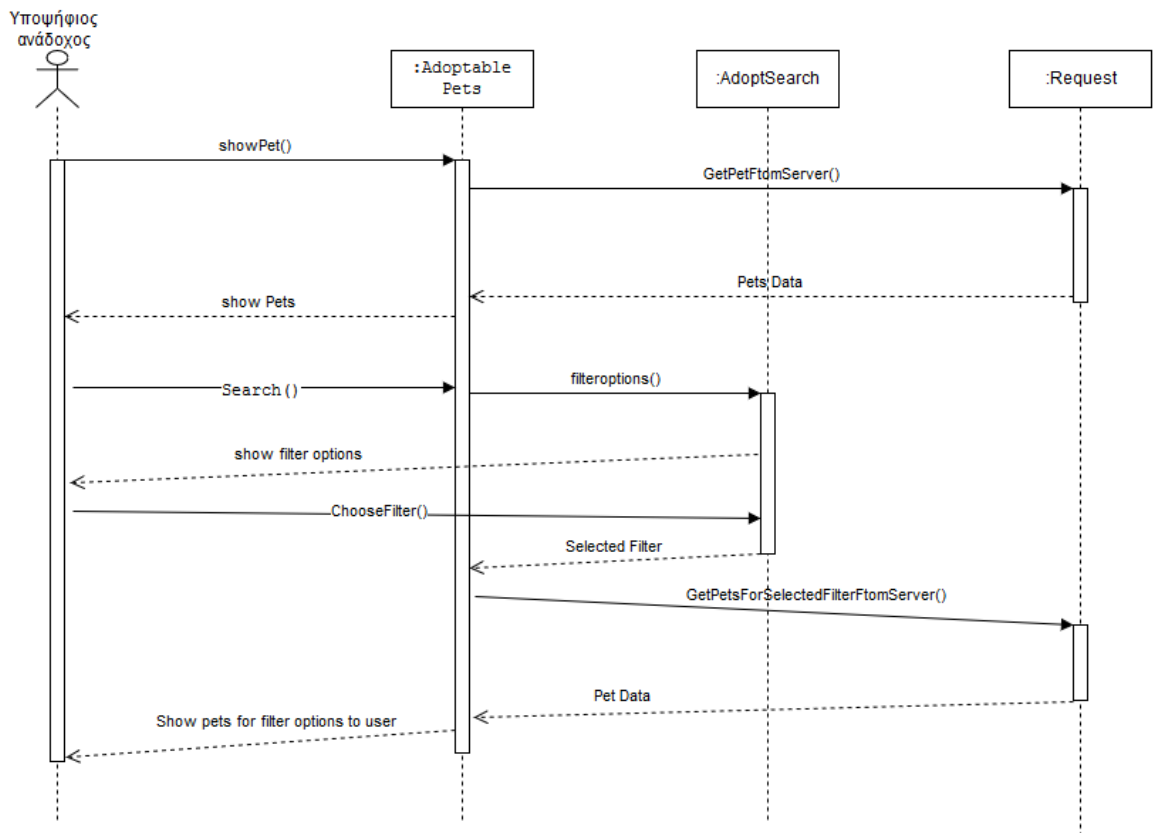


4.2.1.1 Υποψήφιος ανάδοχος

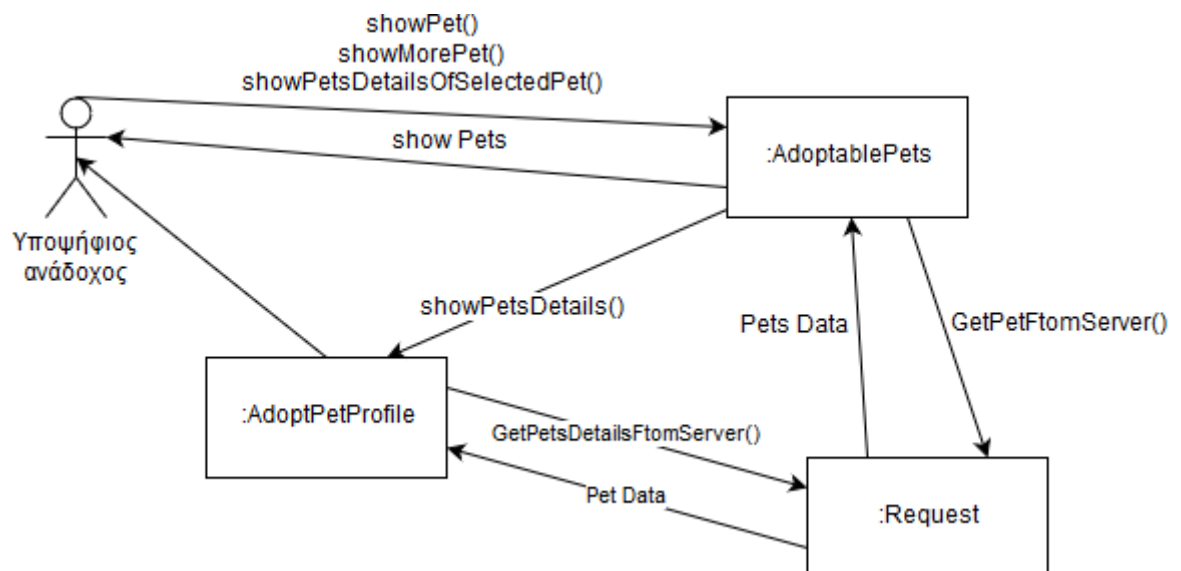
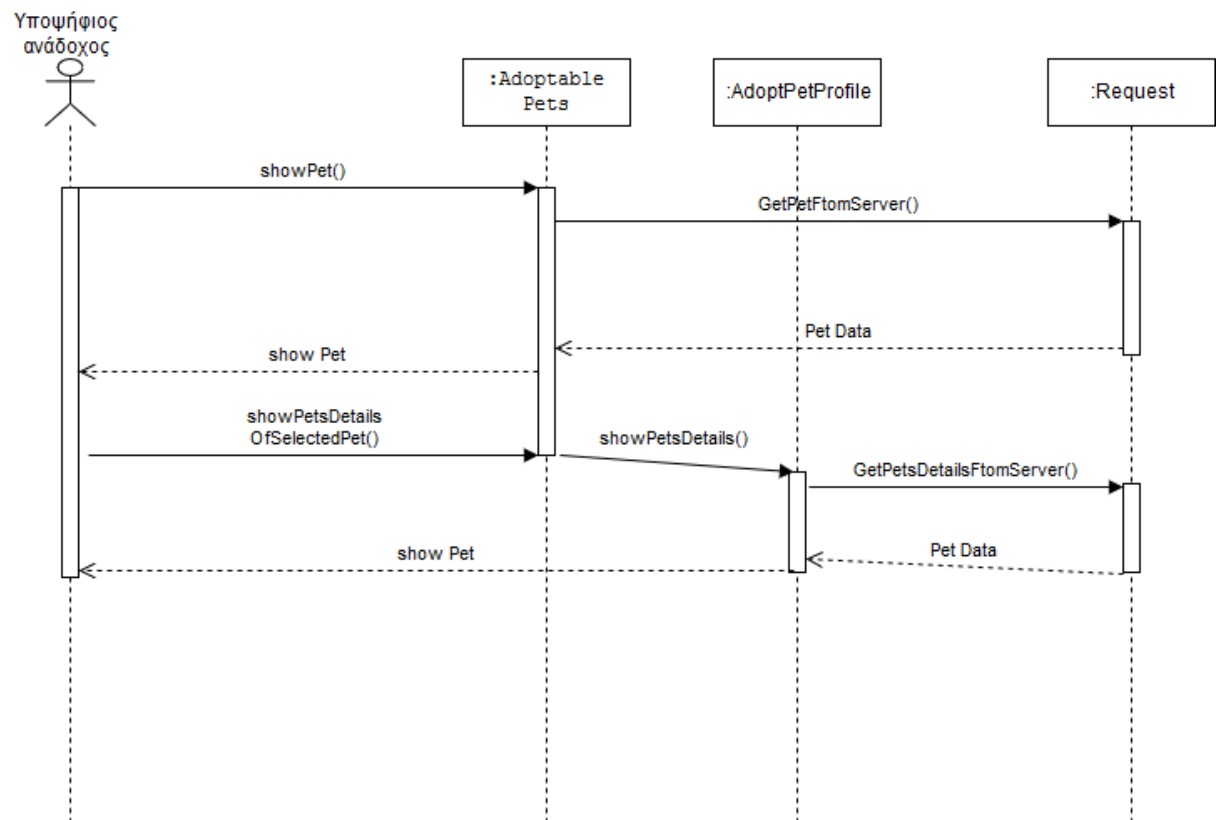
4.2.1.1.1 Προβολή κατοικίδιων



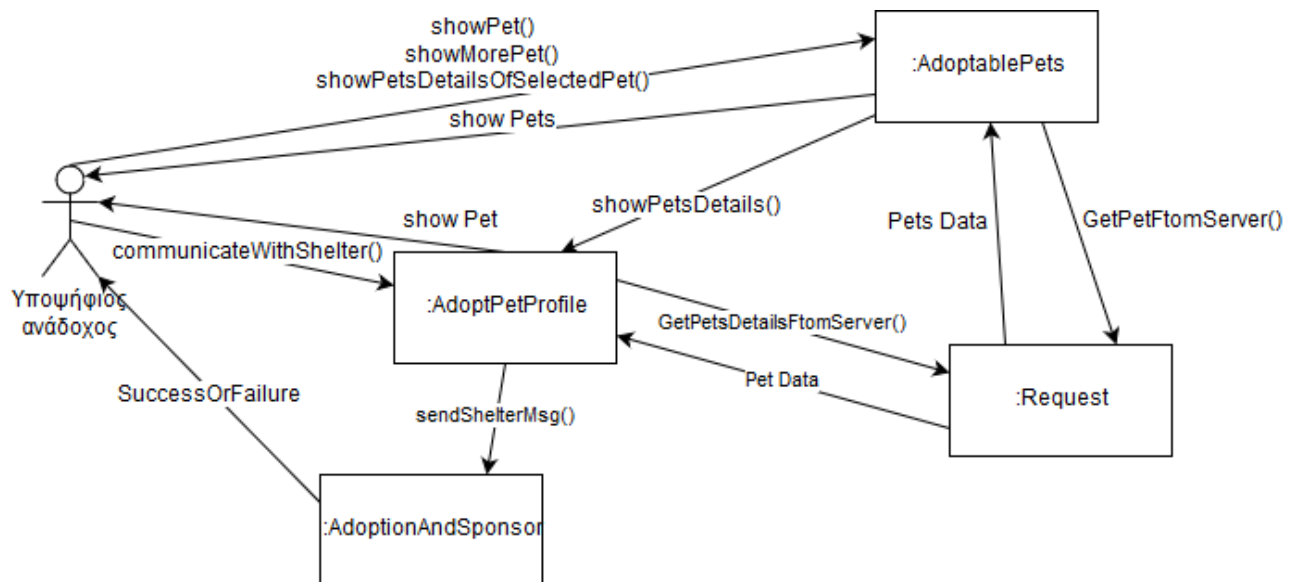
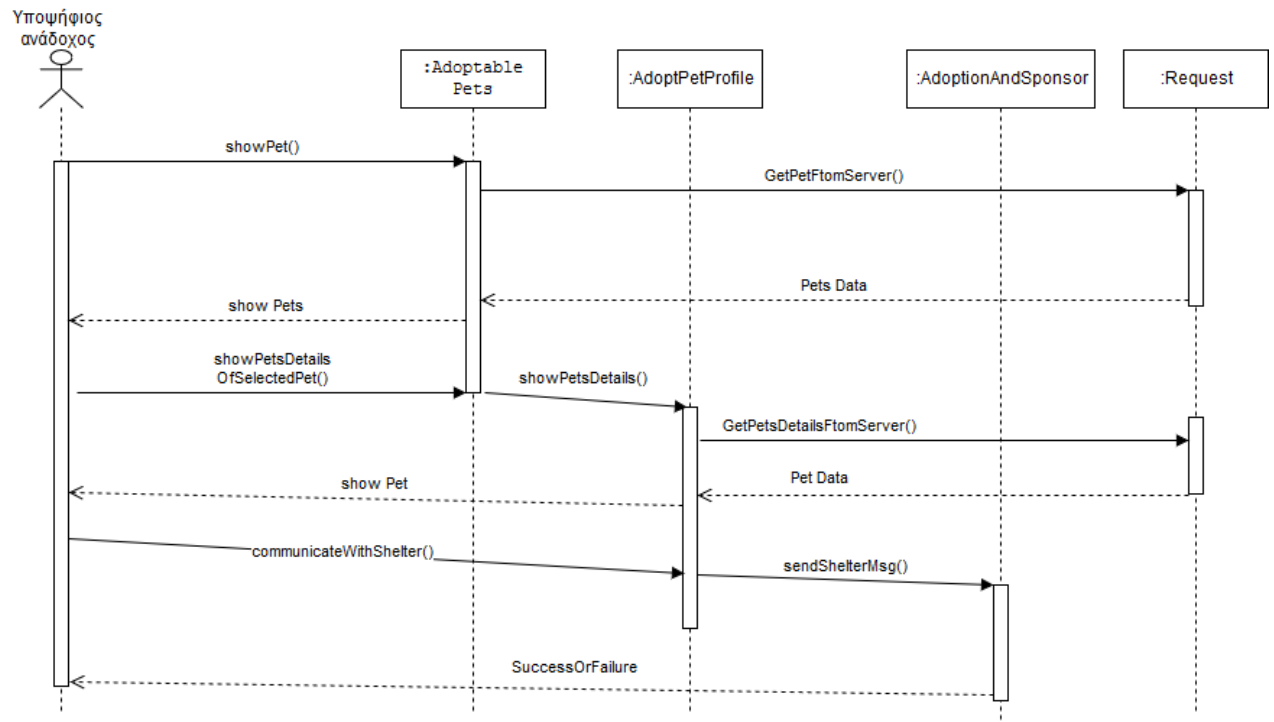
4.2.1.1.2 Αναζήτηση κατοικίδιων βάση συγκεκριμένων φίλτρων αναζήτησης



4.2.1.1.3 Προβολή περισσότερων πληροφοριών για ένα κατοικίδιο



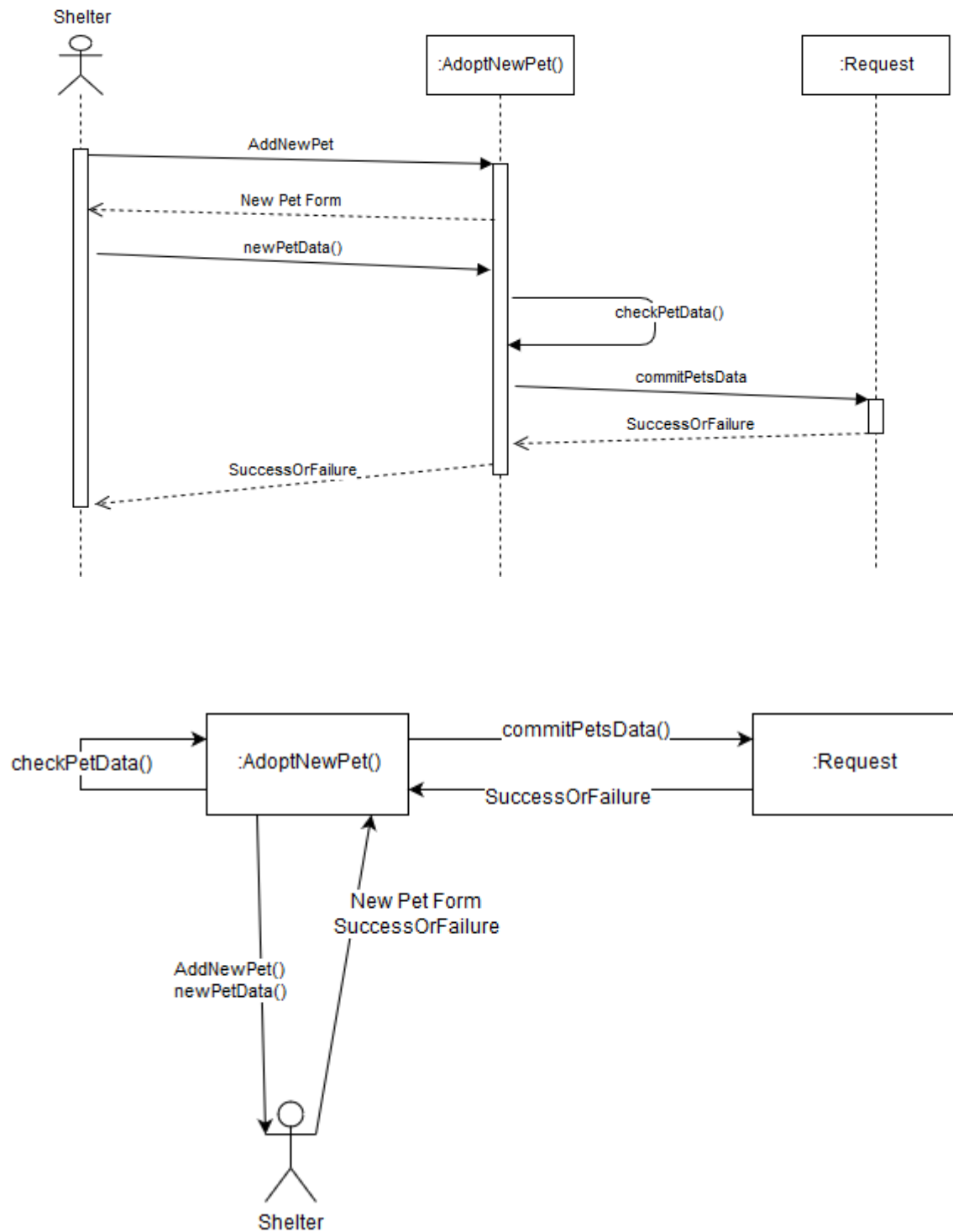
4.2.1.1.4 Επικοινωνία με το καταφύγιο που φροντίζει ένα ζώον



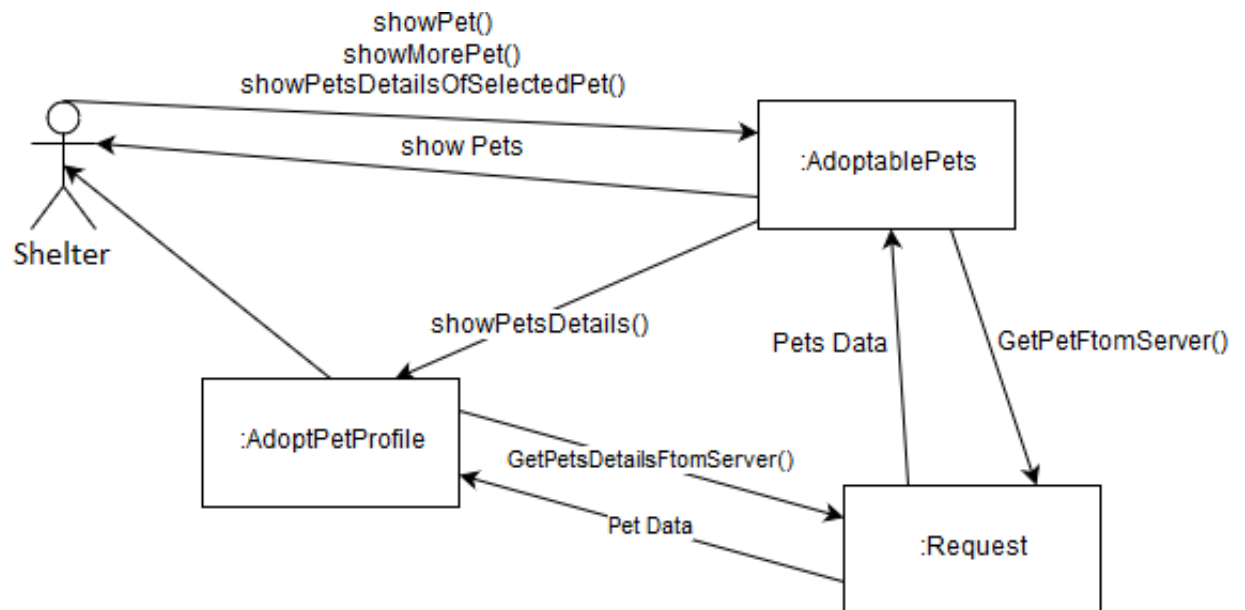
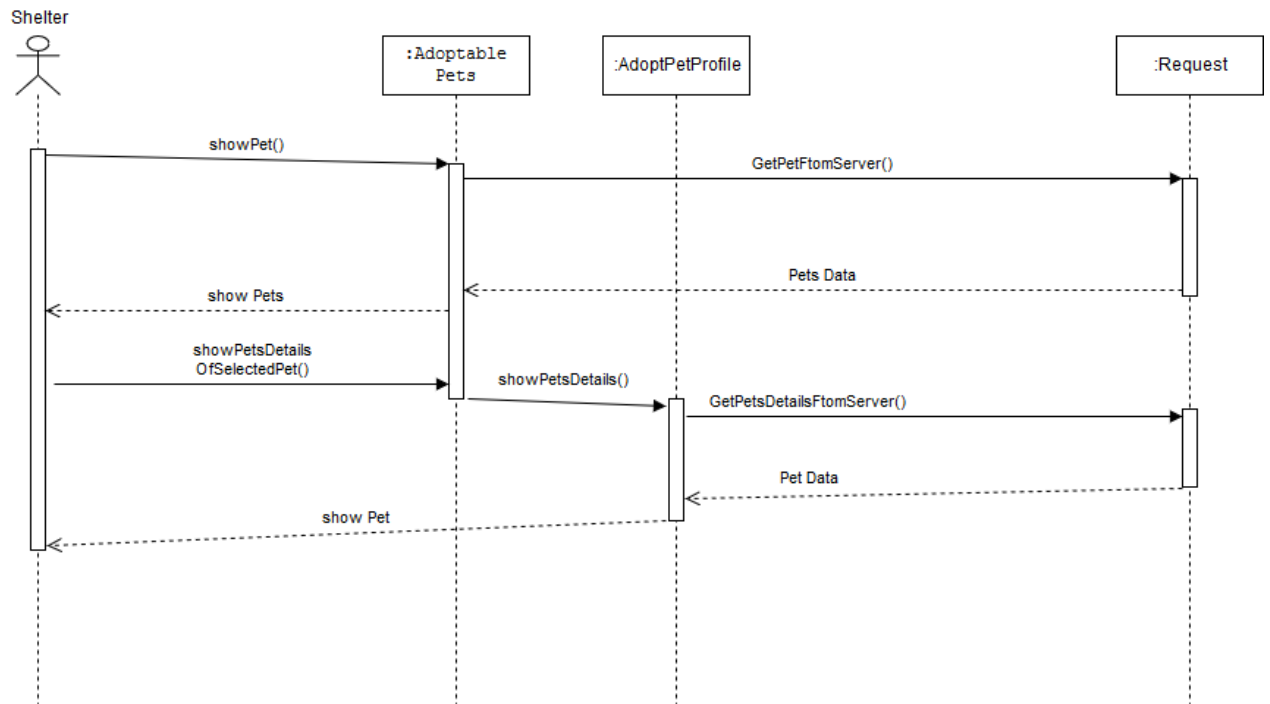
** Η διαδικασία επικοινωνίας με ένα καταφύγιο γίνεται σε όλες τις περιπτώσεις με τον ίδιο τρόπο. Δηλαδή αν ο χρήστης θέλει να επικοινωνήσει με το καταφύγιο για να υιοθετήσει κάποιο ζώο, ή να πάρει επιπρόσθετες πληροφορίες ή ακόμη και να βοηθήσει στην φροντίδα του (Sponsor) θα ακολουθήσει τη ίδια διαδικασία.

4.2.1.2 Καταφύγιο ή φιλοζωική οργάνωση

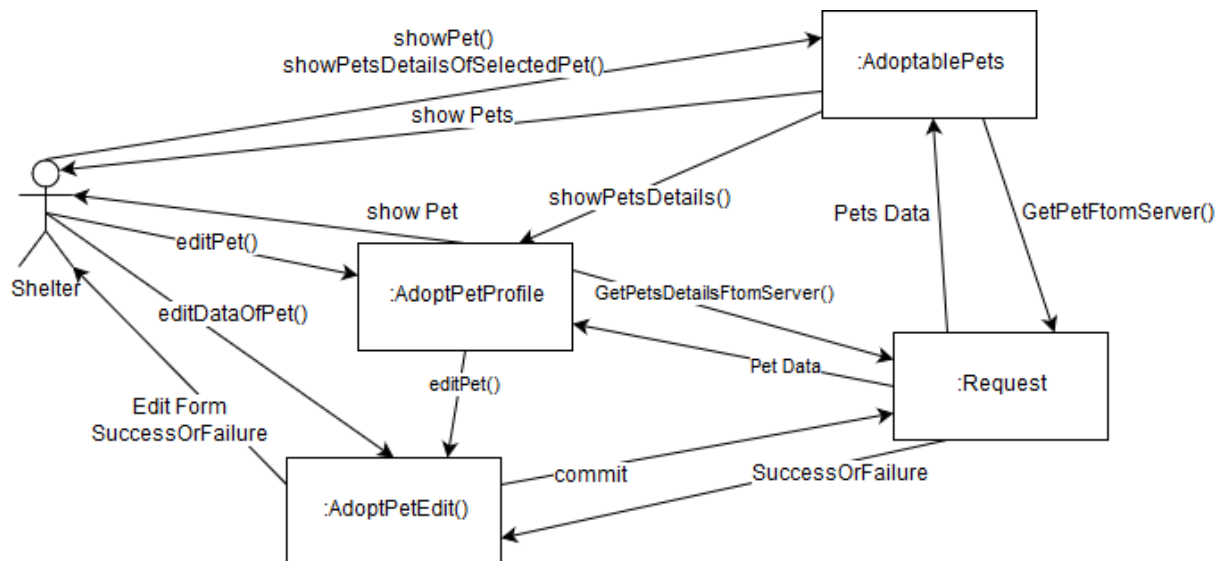
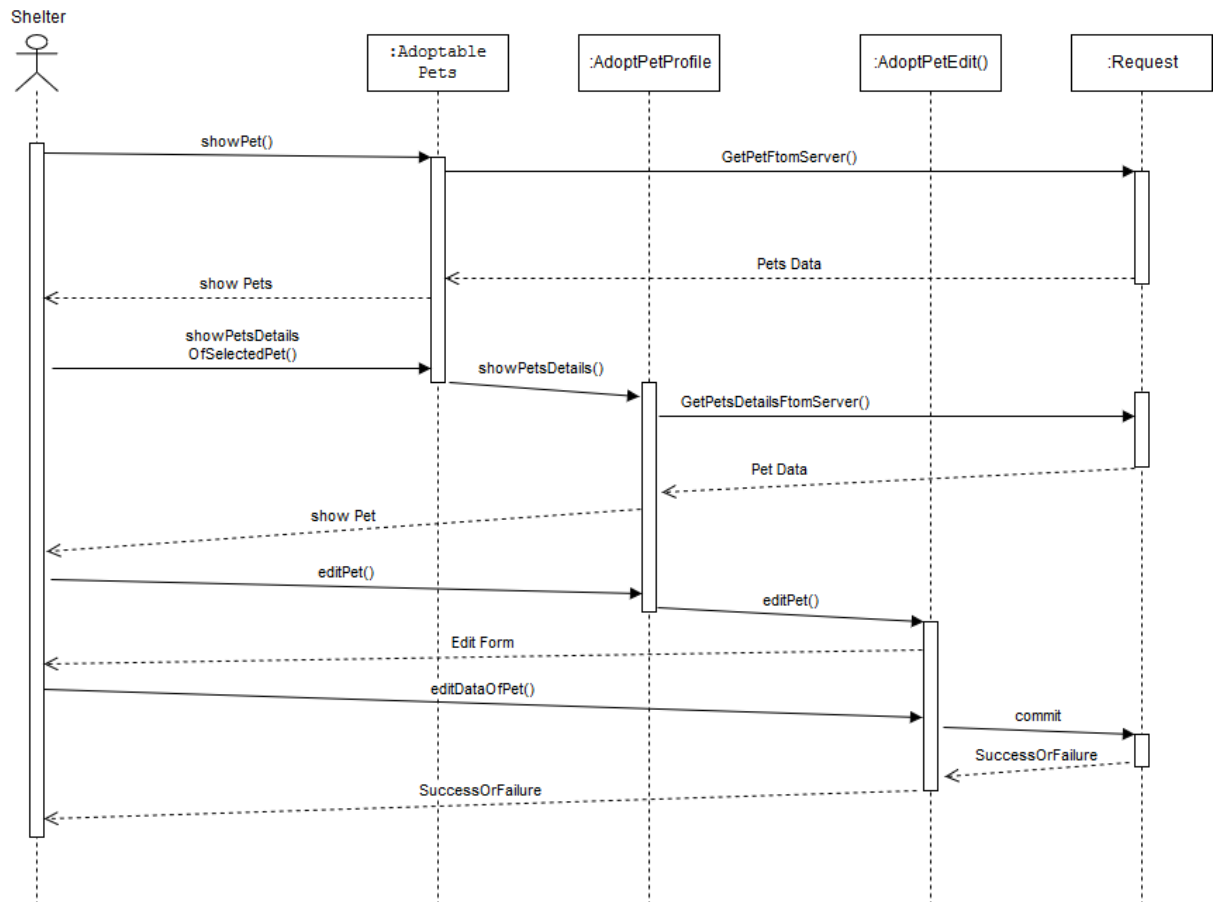
4.2.1.2.1 Εισαγωγή νέου ζώου



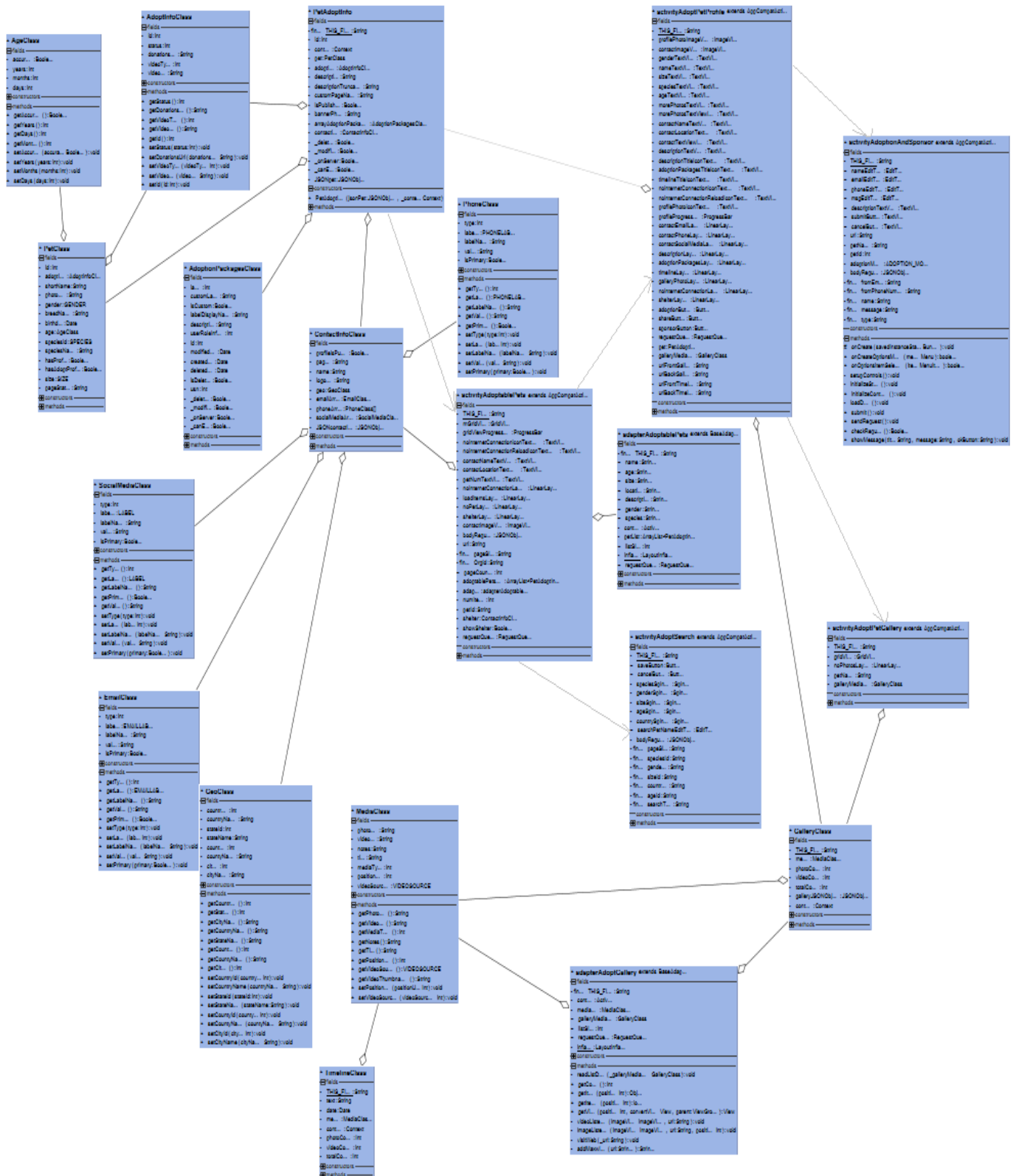
4.2.1.2.2 Προβολή καταχωρημένων κατοικίδιων ενός καταφύγιου



4.2.1.2.3 Επεξεργασία ενός καταχωρημένου ζώου από ένα καταφύγιο



4.2.2 Αναλυτικό Class Diagram



Κεφάλαιο 5

Υλοποίηση Συστήματος

5.1 Εισαγωγή	48
5.2 Γραφική διεπαφή χρήστη	49
5.3 Υλοποίηση εφαρμογής Android	61
5.4 Υλοποίηση εφαρμογής iOS	70

5.1 Εισαγωγή

5.1.1 Σκοπός

Σκοπός του κεφαλαίου αυτού είναι η παρουσίαση της φάσης της υλοποίησης της ανάπτυξης του παρόντος έργου ανάπτυξης λογισμικού, με απώτερο στόχο τη διευκόλυνση και την εξομάλυνση της φάσης συντήρησης του συστήματος.

5.1.2 Αποδέκτες

Το έγγραφο αυτό έχει προορισμό τα άτομα τα οποία πρόκειται να αναλάβουν τη συντήρηση του έργου (προγραμματιστές της 11pets). Αυτά τα άτομα θα βοηθηθούν από τις διάφορες σημειώσεις που έχουν παρατεθεί σε αυτό το έγγραφο.

5.2 Γραφική διεπαφή χρήστη

5.2.1 Υποψήφιος ανάδοχος

Το πρώτο παράθυρο που θα εμφανιστεί στον υποψήφιο ανάδοχο μόλις ξεκινήσει την εφαρμογή θα είναι η προβολή όλων των κατοικίδιων(εικόνα 5.2.1.1).

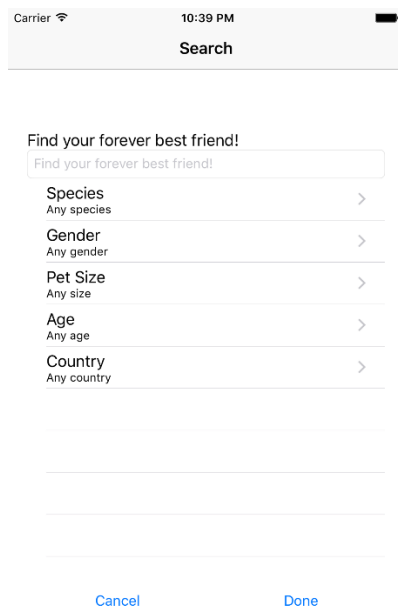


Εικόνα 5.2.1.1(α) iOS εφαρμογή
προβολή των κατοικίδιων

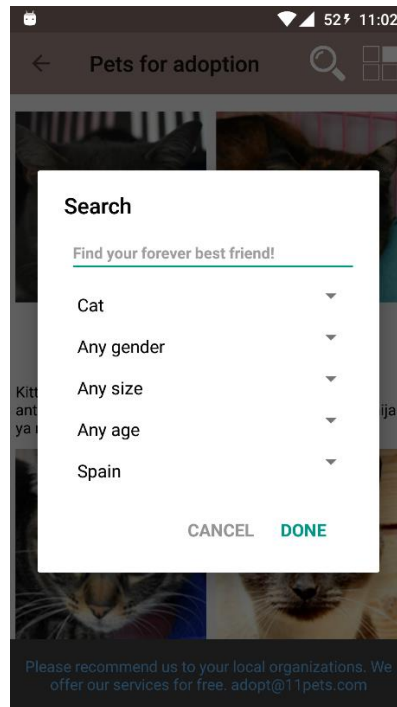


Εικόνα 5.2.1.1(β) Android εφαρμογή
προβολή των κατοικίδιων

Πατώντας πάνω στο εικονίδιο της αναζήτησης (πάνω δεξιά στην οθόνη) εμφανίζονται όλα τα φίλτρα αναζήτησης (εικόνα 5.2.1.2).

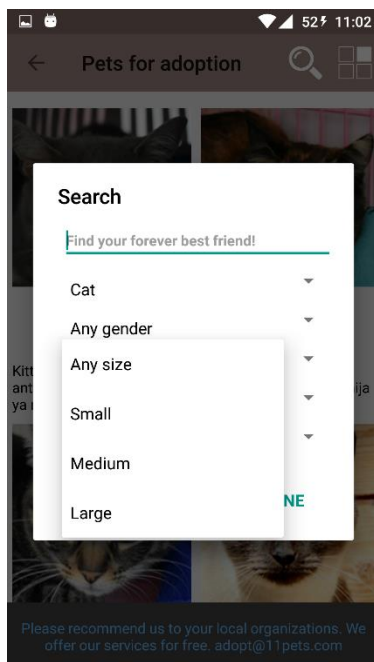


Εικόνα 5.2.1.2(α) αναζήτηση
iOS εφαρμογή

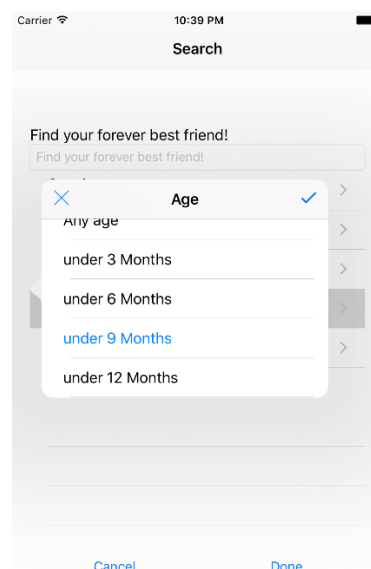


Εικόνα 5.2.1.2(β) αναζήτηση
android εφαρμογή

Επιλέγοντας ένα φίλτρο εμφανίζονται όλες οι επιλογές για το συγκεκριμένο φίλτρο (εικόνα 5.2.1.3). Ακολούθως πατώντας το Done(εικόνα 5.2.1.2) πραγματοποιείται η αναζήτηση και εμφανίζονται στο χρήστη όλα τα κατοικίδια που ταιριάζουν στην αναζήτηση που έκανε.

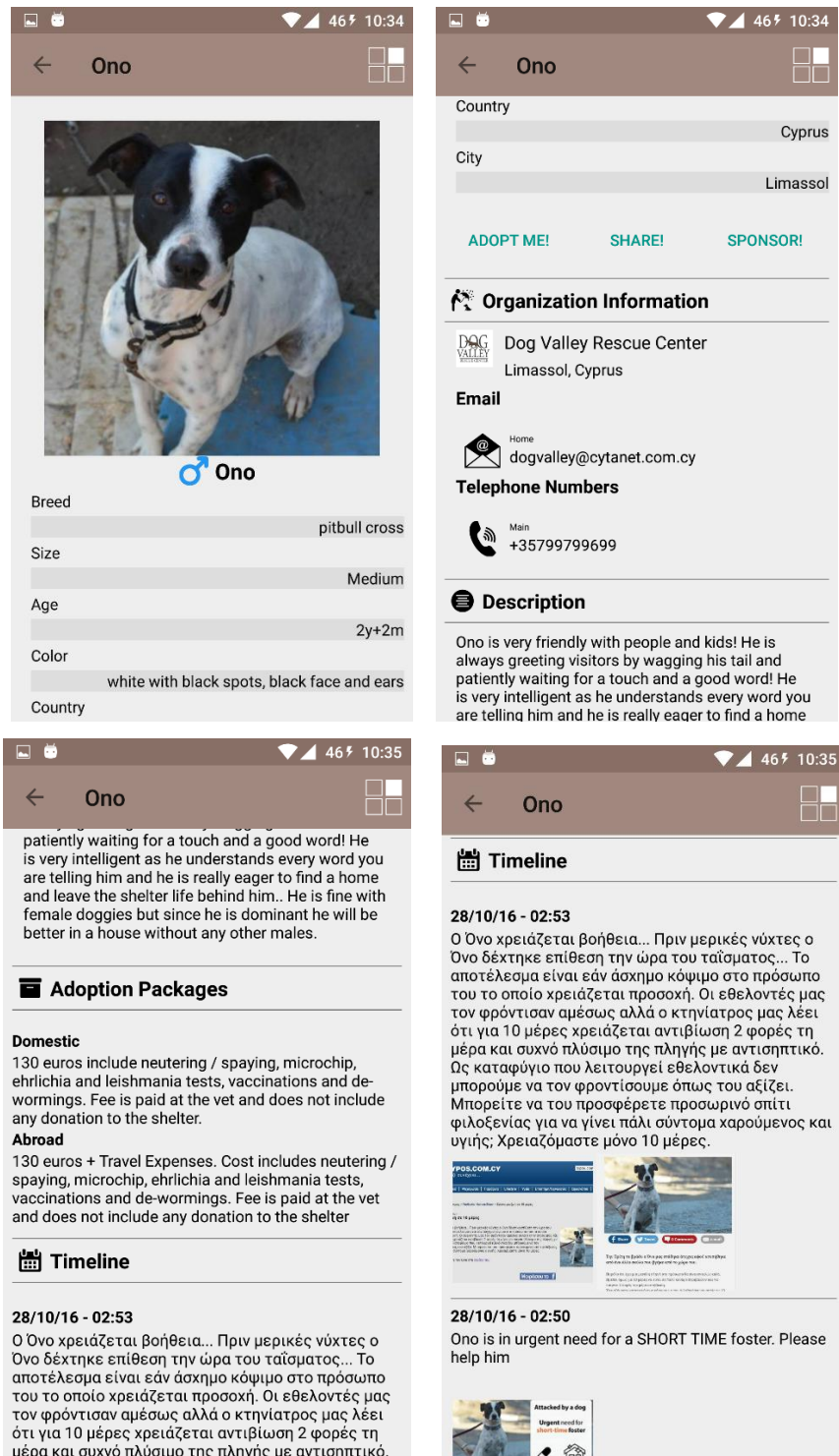


Εικόνα 5.2.1.3 (α)
android εφαρμογή

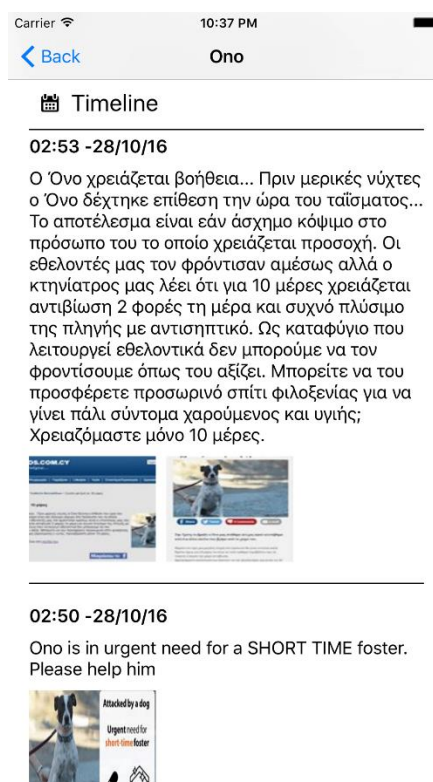
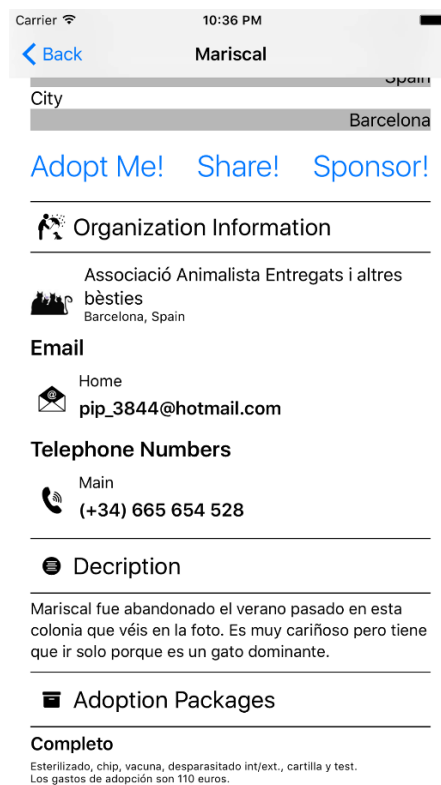


Εικόνα 5.2.1.3 (β)
iOS εφαρμογή

Επιλέγοντας ένα από τα κατοικίδια(εικόνα 5.2.1.1), εμφανίζονται σε δομημένη μορφή περισσότερες πληροφορίες για το συγκεκριμένο ζώον(εικόνα 5.2.1.4).



Εικόνα 5.2.1.4(α)
Android εφαρμογή

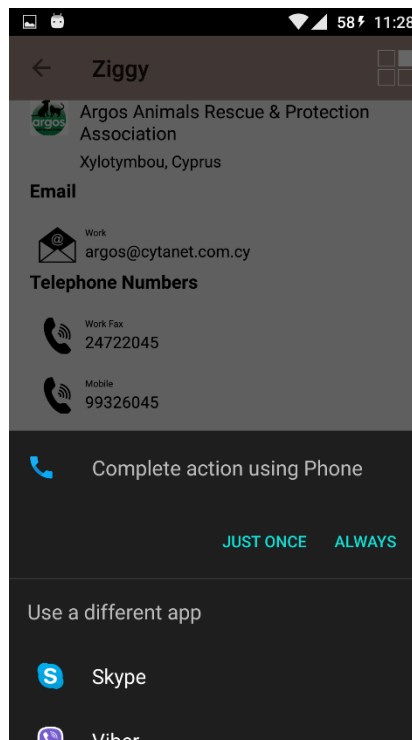


Εικόνα 5.2.1.4(β)
iOS εφαρμογή

Στο πάνω μέρος εμφανίζονται οι φωτογραφίες και σύροντας (Swipe) το δάκτυλο του ο χρήστης πάνω στην φωτογραφία προς τα δεξιά εμφανίζεται η επόμενη φωτογραφία ενώ

προς τα αριστερά η προηγούμενη φωτογραφία. Πατώντας πάνω στην φωτογραφία εμφανίζεται η φωτογραφία σε πλήρη οθόνη.

Από την σελίδα ενός κατοικίδιου (εικόνα 5.2.1.5) αν ο χρήστης επιλέξει το ηλεκτρονικό ταχυδρομείο που εμφανίζεται ή το τηλέφωνο τότε μεταβαίνει στις αντίστοιχες εφαρμογές που είναι εγκατεστημένες στην συσκευή για να σταλεί email ή να πάρει τηλέφωνο αντίστοιχα (εικόνα 5.3.1.1).



Εικόνα 5.2.1.5
android εφαρμογή

Ακόμη ένας τρόπος να επικοινωνήσει μαζί με το καταφύγιο που φροντίζει το συγκεκριμένο ζώο είναι μέσω του κουμπιού Adopt Me!(για υιοθεσία) ή του κουμπιού Sponsor!(για βοήθεια στην φροντίδα του ζώου). Επιλέγοντας ένα από τα δυο τότε εμφανίζεται η φόρμα (εικόνα 5.2.1.6) που πρέπει να συμπληρώσει αφήνοντας τα προσωπικά του στοιχεία για να επικοινωνήσει το καταφύγιο μαζί του.

Carrier 10:38 PM

< Ono Adoption form

Thank you for showing interest in adopting Ono. Please fill in the contact form below and we a member of our team will get in touch with you are soon as possible.

Your Name
Your Name

Your Email
Your Email

Your phone number
Your phone number

Your message
Your message

Cancel Submit

Εικόνα 5.2.1.6(α)
iOS εφαρμογή

Carrier 59% 11:31

Thank you for showing interest in adopting Ziggy. Please fill in the contact form below and we a member of our team will get in touch with you are soon as possible.

Your Name

Your Email

Your phone number

Your message

CANCEL SUBMIT

Εικόνα 5.2.1.6(β)
Android εφαρμογή

Αν παραλείψει ένα από τα απαραίτητα στοιχεία τότε εμφανίζεται το κατάλληλο μήνυμα στον χρήστη (εικόνα 5.3.1.1).

Carrier 10:39 PM

< Ono Adoption form

Thank you for showing interest in adopting Ono. Please fill in the contact form below and we a member of our team will get in touch with you are soon as possible.

Your Name
Daniel

Your Email
email@gmail.com

Your phone number
Your phone number

Cannot be empty

Your message

Cancel Submit

Εικόνα 5.2.1.6(α)
iOS εφαρμογή

Carrier 60% 11:38

Thank you for showing interest in adopting Patas. Please fill in the contact form below and we a member of our team will get in touch with you are soon as possible.

Your Name

Invalid Data

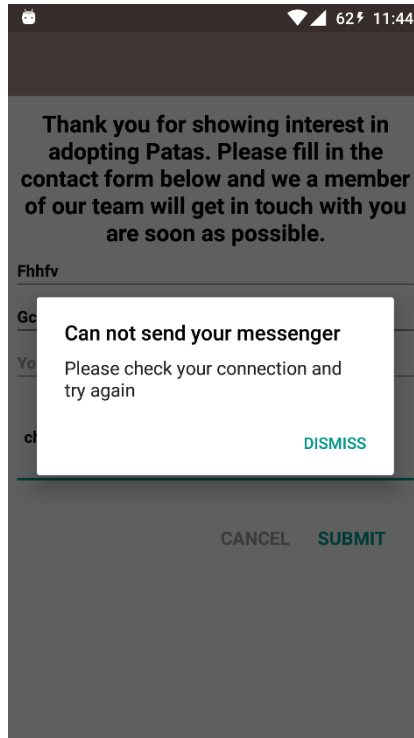
Please give your name

DISMISS

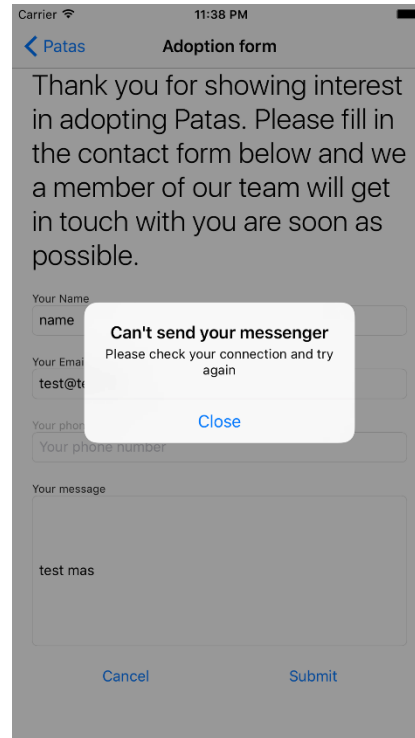
CANCEL SUBMIT

Εικόνα 5.2.1.6(β)
android εφαρμογή

Τέλος αν για κάποιο λόγο δεν μπορούν να αποσταλούν τα στοιχεία του τότε πάλι εμφανίζεται το κατάλληλο μήνυμα στο χρήστη (εικόνα 5.2.1.7).



Εικόνα 5.2.1.7(α)
android εφαρμογή



Εικόνα 5.2.1.7(β)
iOS εφαρμογή

Όταν ο χρήστης πατήσει πάνω στο όνομα του καταφύγιου (εικόνα 5.2.1.4). του εμφανίζονται όλα τα ζώα που φιλοξενεί το συγκεκριμένο καταφύγιο (εικόνα 5.2.1.8). Δίνεται η δυνατότητα στον χρήστη να πραγματοποιεί αναζήτηση στα ζώα ενός καταφύγιου (σύμβολο της αναζήτησης πάνω δεξιά).

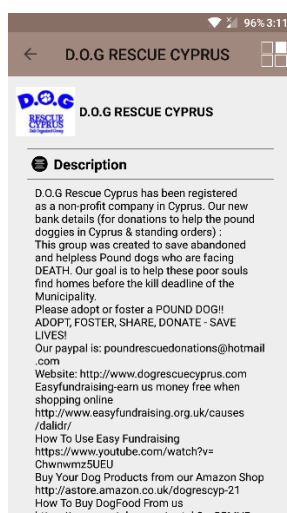


Εικόνα 5.2.1.8(α)
iOS εφαρμογή

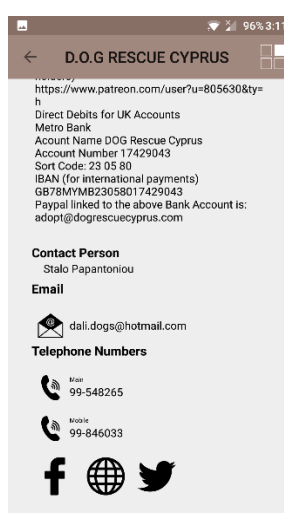


Εικόνα 5.2.1.8(β)
android εφαρμογή

Ο χρήστης έχει την δυνατότητα να δει περισσότερες πληροφορίες για ένα καταφύγιο (εικόνα 5.2.1.9) πατώντας στο εικονίδιο των πληροφοριών δεξιά από το όνομα του καταφύγιου (εικόνα 5.2.1.8). Δίνεται η δυνατότητα στον χρήστη να πάρει τηλέφωνο ή να στείλει email πατώντας πάνω στο τηλέφωνο του καταφύγιου ή στο ηλεκτρονικό ταχυδρομείο αντίστοιχα. Ακόμη πατώντας πάνω στα κοινωνικά δίκτυα του καταφύγιου μεταφέρεται στην σελίδα του καταφύγιου στο αντίστοιχο κοινωνικό δίκτυο.



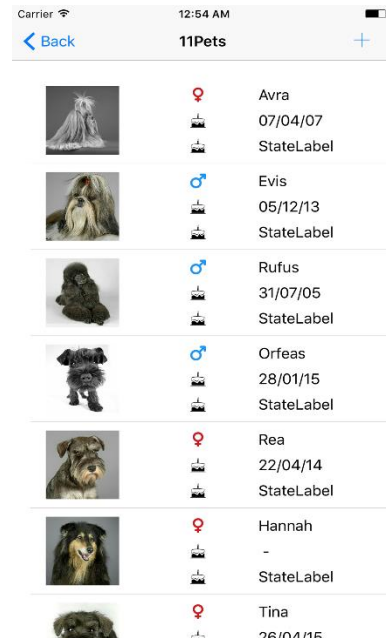
Εικόνα 5.2.1.9(α)
android εφαρμογή



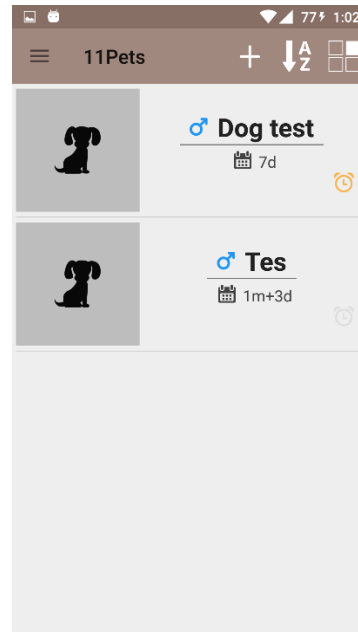
Εικόνα 5.2.1.9(β)
iOS εφαρμογή

5.2.2 Καταφύγιων ζώων ή φιλοζωική οργάνωση

Το πρώτο παράθυρο που θα εμφανιστεί σε ένα καταφύγιο μόλις ξεκινήσει την εφαρμογή θα είναι η προβολή όλων των κατοικίδιων που έχει προσθέσει στο σύστημα (εικόνα 5.2.2.1).

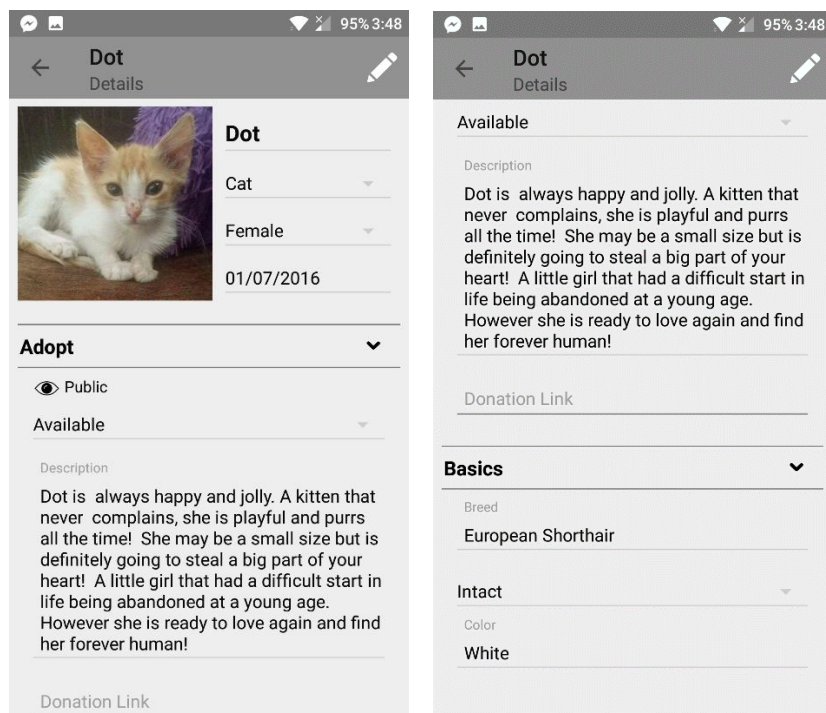


Εικόνα 5.2.2.1(α)
iOS εφαρμογή

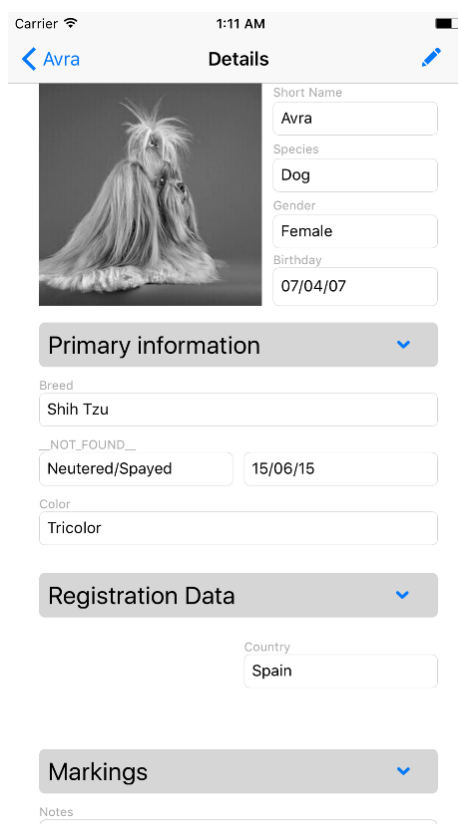


Εικόνα 5.2.2.1(β)
android εφαρμογή

Επιλέγοντας ένα από τα κατοικίδια εμφανίζονται περισσότερες πληροφορίες για το συγκεκριμένο ζώο (εικόνα 5.2.2.2).



Εικόνα 5.2.2.2(α)
android εφαρμογή



Εικόνα 5.2.2.2(β)
iOS εφαρμογή

Πατώντας το κουμπί της επεξεργασίας (μολύβι πάνω δεξιά στην οθόνη) ο χρήστης έχει την δυνατότητα να τροποποιήσει τα δεδομένα (εικόνα 5.2.2.3).

Carrier 12:55 AM

< Avra Details

Dog

Gender Female

Birthday 07/04/07

Edit

Primary information

Breed Shih Tzu

Neutered/Spayed 15/06/15

Color Tricolor

Description

Registration Data

Markings

Notes

Has a surgery mark at the belly

Εικόνα 5.2.2.3(α)
iOS εφαρμογή

Dot Details

Cat

Female

01/07/2016

Adopt

Public

Available

Description

Dot is always happy and jolly. A kitten that never complains, she is playful and purrs all the time! She may be a small size but is definitely going to steal a big part of your heart! A little girl that had a difficult start in life being abandoned at a young age. However she is ready to love again and find her forever human!

Donation Link

Dot Details

me being abandoned at a young age. However she is ready to love again and find her forever human!

Donation Link

Basics

Breed European Shorthair

Intact

Color White

Notes

Advanced mode

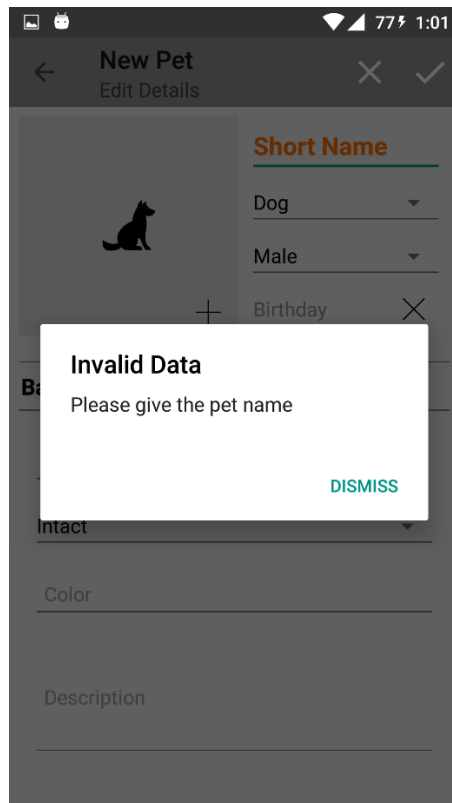
Εικόνα 5.2.2.3(β)
android εφαρμογή

Η διαδικασία πρόσθεσης ενός νέου κατοικίδιου ξεκινά από την οθόνη προβολής των ζώων και επιλέγοντας το σύμβολο του συν (+ πάνω δεξιά στην οθόνη) εμφανίζεται η φόρμα εισαγωγής νέου κατοικίδιου (εικόνα 5.2.2.4).

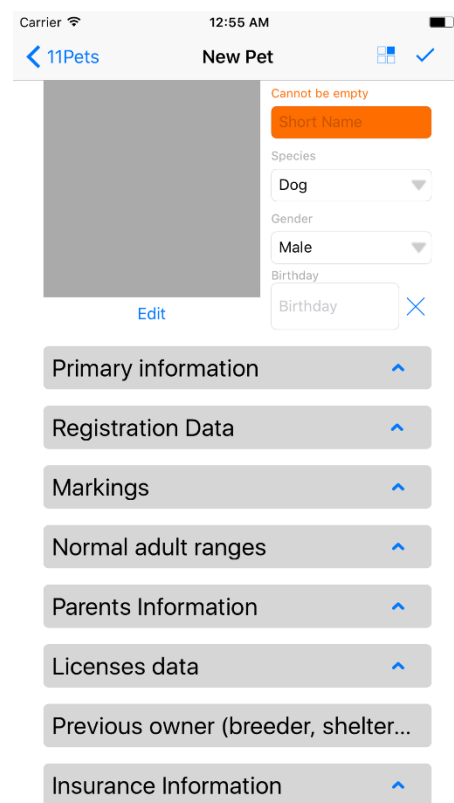
Εικόνα 5.2.2.4(α)
iOS εφαρμογή

Εικόνα 5.2.2.4(β)
android εφαρμογή

Αν ο χρήστης προσπαθήσει να αποθηκεύσει το νέο ζώο χωρίς να συμπληρώσει όλα τα απαιτούμενα πεδία τότε εμφανίζεται το κατάλληλο μήνυμα στον χρήστη (εικόνα 5.2.2.5).



Εικόνα 5.2.2.5(α)
android εφαρμογή



Εικόνα 5.2.2.5(β)
iOS εφαρμογή

5.3 Υλοποίηση εφαρμογής Android

Σε αυτή την ενότητα θα αναλυθεί λεπτομερώς ο τρόπος και οι τεχνικές που χρησιμοποιήθηκαν για την υλοποίηση των κυριότερων λειτουργιών της android εφαρμογής.

Πρώτα να αναφέρουμε κάποια βασικά χαρακτηριστικά των android εφαρμογών. Μια εφαρμογή για το λειτουργικό σύστημα android αποτελείται κυρίως από πολλά activities^[25] ενώ μπορεί να διαθέτει επίσης services^[27], Listener^[26] και Adapter.

Ένα Activity είναι μια σελίδα στην εφαρμογή. Ουσιαστικά activity είναι η κάθε κλάση η οποία διαθέτει δική της γραφική διεπαφή χρήστη. Ένα activity μπορεί να τρέχει μόνο στο προσκήνιο και κλείνει όταν ο χρήστης μετακινηθεί σε ένα άλλο activity.

Ενας Listener είναι το συστατικό της εφαρμογής το οποίο «ακούει» και περιμένει για μια συγκεκριμένη ενέργεια από το χρήστη που προκαλεί κάποιο event. Για παράδειγμα ο Listener μπορεί να καλείται στο ξεκίνημα ενός activity και μόλις συμβεί το συγκεκριμένο event ο Listener θα εκτελέσει τον κώδικά του.

Ενας Adapter είναι ένα αντικείμενο το οποίο λειτουργεί ως μια γέφυρα μεταξύ ενός AdapterView και των υποκείμενων δεδομένων για αυτήν την προβολή. Ο Adapter παρέχει πρόσβαση στα δεδομένα που θα προβληθούν στον χρήστη. Ο Adapter είναι επίσης υπεύθυνος για τη δημιουργία μιας προβολής (View) για κάθε στοιχείο του συνόλου δεδομένων. Για την δημιουργία μιας κλάσης-Adapter απαιτείται πρωτίστως η κλάση αυτή να είναι κληρονόμος της κλάσης android.widget.BaseAdapter. Η κλάση αυτή περιέχει τις μεθόδους getCount(), getItem(), getItemId(), getView(), όπου η νέα κλάση πρέπει υποχρεωτικά να τις υλοποιήσει κάνοντας Override τις αρχικές.

Τέλος ένα Service είναι το συστατικό της εφαρμογής το οποίο τρέχει συνεχώς συνήθως για την εκτέλεση κάποιας χρονοβόρας διαδικασίας (π.χ. κατέβασμα αρχείου) ή για την εκτέλεση κάποιας λειτουργίας που δεν απαιτείται αλληλεπίδραση με τον χρήστη (π.χ. εξακρίβωση θέσης). Τα service δεν διαθέτουν γραφική διεπαφή χρήστη.

Επίσης μια εφαρμογή σε android διαθέτει και άλλα σημαντικά στοιχεία όπως το Manifest αρχείο το οποίο περιέχει σημαντικές λεπτομέρειες για την εφαρμογή π.χ. ποιες εκδόσεις του android υποστηρίζει η εφαρμογή κτλ, το strings αρχείο το οποίο περιέχει το κείμενο που χρησιμοποιείται στην εφαρμογή και το styles αρχείο που περιέχει τα θέματα και άλλα δεδομένα που έχουν να κάνουν με την εμφάνιση της εφαρμογής.

5.3.1 Υλοποίηση γραφικής διεπαφής χρήστη

Μια διεπαφή χρήστη για android εφαρμογή είναι γραμμένη σε XML^[35,36] και διαθέτει διάφορα Views και Viewgroups. Views είναι τα στοιχεία τα οποία αποτελούν το γραφικό περιβάλλον και προσφέρουν δυνατότητα αλληλεπίδρασης με την εφαρμογή στο χρήστη. Παραδείγματα view είναι το button, το textview, το checkbox, ImageView κτλπ. Viewgroups είναι σύνολα από views τα οποία επίσης υποδεικνύουν το πώς είναι δομημένα τα Views μέσα στο συγκεκριμένο group. Παραδείγματα viewgroup είναι το RelativeLayout που επιτρέπει την τοποθέτηση των views οπουδήποτε στο viewgroup (στο πάνω/κάτω μέρος του group, δεξιά/αριστερά από ένα άλλο view, πάνω/κάτω από ένα άλλο view κτλπ.) και το LinearLayout που επιτρέπει την τοποθέτηση views στην σειρά (το ένα δίπλα στο άλλο ή το ένα πάνω από το άλλο) κτλ. Επίσης αξίζει να αναφερθεί πως μέσα σε ένα Viewgroup μπορούν να υπάρχουν πέρα από Views και Viewgroups. Για παράδειγμα, μέσα σε ένα LinearLayout μπορεί να υπάρχει ένα button και ένα RelativeLayout. Ακολουθεί ένα παράδειγμα αρχείου (σχήμα 7) που περιέχει μια γραφική διεπαφή. Στο παράδειγμα αυτό υπάρχει ένα LinearLayout οπου έχει οριζόντια προσέγγιση (horizontally) μέσα στο οποίο υπάρχουν ένα RelativeLayouts δυο LinearLayout και 4 TextView. Μέσα σε στο RelativeLayout υπάρχει ένα Image View και ένα ProgressBar το οποίο εμφανίζεται στη μέση του ImageView.

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    style="@style/activitySectionLayout">

    <LinearLayout style="@style/simpleLinearLayout.MW.H">

        <RelativeLayout
            android:id="@+id/adoptListItem_imageLayout"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content">

            <ImageView
                android:id="@+id/adoptPetProfile_profilePhotoImageView"
                android:layout_width="150dp"
                android:layout_height="150dp"
                android:src="@drawable/no_photo" />

            <ProgressBar
                android:id="@+id/adoptPetProfile_profilePhotoProgressBar"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:gravity="center_vertical|center_horizontal"/>

        </RelativeLayout>
    </LinearLayout>
```

```

style="@style/simpleLinearLayout.MM.V"
android:layout_marginLeft="10dp">

<LinearLayout
    style="@style/simpleLinearLayout.MW.H"
    android:gravity="center_vertical|center_horizontal">

    <TextView
        android:id="@+id/adoptPetProfile_genderTextView"
        android:gravity="left"
        android:textColor="@color/black"
        android:textSize="@dimen/textSizeXXL" />

    <TextView
        android:id="@+id/adoptPetProfile_nameTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textStyle="bold" />
</LinearLayout>

<TextView
    android:id="@+id/adoptPetProfile_sizeTextView"
    style="@style/simpleTextView.S"
    android:layout_marginTop="4dp"
    android:text="Pet Size" />

<TextView
    android:id="@+id/adoptPetProfile_ageTextView"
    style="@style/simpleTextView.S"
    android:layout_marginTop="4dp"
    android:text="Pet age" />

<LinearLayout
    style="@style/simpleLinearLayout.MW.H"
    android:layout_marginBottom="4dp"
    android:layout_marginTop="4dp"
    android:layout_weight="1">

    <LinearLayout
        android:id="@+id/adoptPetProfile_MorePhotosLinearLayout"
        style="@style/simpleLinearLayout.MW.H"
        android:layout_gravity="bottom"
        android:visibility="gone">

        <Button
            android:id="@+id/adoptPetProfile_MorePhotosButton"
            style="@style/buttonWithCustomFont.size_XXL"
            android:minWidth="0dp"
            android:text="#" />

        <TextView
            android:id="@+id/adoptPetProfile_MorePhotosTextView"
            style="@style/simpleTextView.bold.S"
            android:layout_marginLeft="8dp"
            android:text="@string/adoptGallery" />

    </LinearLayout>
</LinearLayout>

</LinearLayout>

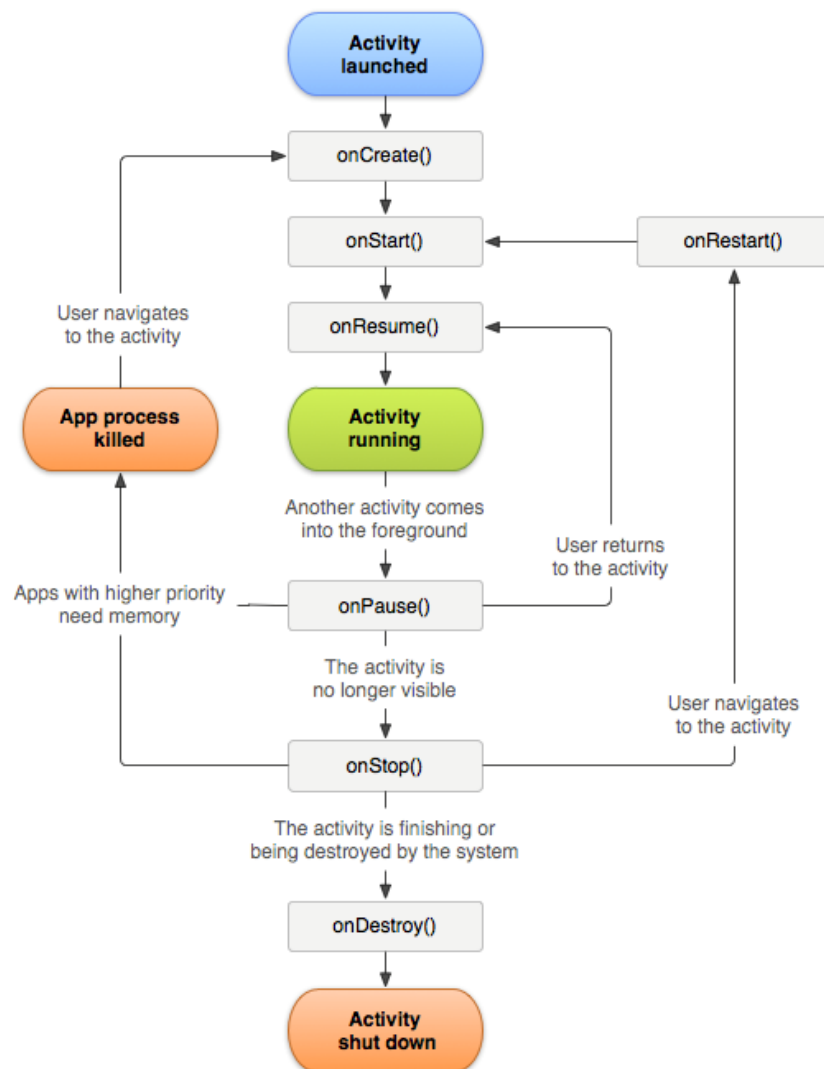
</LinearLayout>

```

Σχήμα 7: activity_adopt_pet_profile_pet_details.xml

5.3.2 Υλοποίηση λειτουργιών

Η δημιουργία Activity για μια android εφαρμογή απαιτεί μια συγκεκριμένη διαδικασία. Ξεκινώντας από το Activity, για τη δημιουργία μιας κλάσης-Activity απαιτείται πρωτίστως η κλάση αυτή να είναι κληρονόμος της κλάσης android.app.activity. Η κλάση αυτή περιέχει τις μεθόδους onStart(), onResume(), onRestart(), onPause(), onStop(), onDestroy() και τέλος την abstract μέθοδο onCreate() η οποία πρέπει υποχρεωτικά να υλοποιηθεί στην καινούργια κλάση-Activity. Οι μέθοδοι αυτοί καλούνται από το σύστημα κάθε φορά που τρέχει το συγκεκριμένο Activity με μια συγκεκριμένη σειρά που καθορίζεται από τον κύκλο ζωής ενός Activity (σχήμα 8).



Σχήμα 8: Android Activity Lifecycle

Πέρα από τις βασικές λειτουργίες, η υλοποίηση σχεδόν όλων των λειτουργιών της εφαρμογής απαιτεί την επικοινωνία με την βάση δεδομένων που βρίσκεται εγκατεστημένη στον εξυπηρετητή του της 11pets. Για την επικοινωνία με τον εξυπηρετητή έχει δημιουργηθεί μια κλάση με το όνομα `JSONRequest` η οποία κληρονομεί από την κλάση `android.os.AsyncTask`^[28]. Η κλάση αυτή επιτρέπει την εκτέλεση χρονοβόρων διεργασιών ασύγχρονα. Κληρονομώντας από την κλάση αυτή απαιτείται η υλοποίηση της μεθόδου `doInBackground()`. Συγκεκριμένα μέσω ενός REST API που επιτρέπει την πρόσβαση στα δεδομένα της βάσης δεδομένων χρησιμοποιήθηκε η βιβλιοθήκη `volley`^[39] του android για να δημιουργηθούν αιτήματα (Request) και να σταλούν στον εξυπηρετητή έτσι ώστε να λάβουμε ένα αρχείο τύπου JSON^[29] με τα δεδομένα. Το JSON επεξεργάζεται (Parsing)^[30] έτσι ώστε να παρθούν τα χρήσιμα δεδομένα για την εφαρμογή καταλήγοντας έτσι σε δεκατέσσερις κλάσεις που περιέχουν όλα τα απαραίτητα δεδομένα.

Πέρα από τα δεδομένα των κατοικίδιων υπήρχε η ανάγκη για την επικοινωνία με τον εξυπηρετητή για ακόμη ένα λόγο, τη λήψη όλων των απαραίτητων φωτογραφιών. Για αυτούς τους λόγους δημιουργήθηκε η κλάση `BitmapDownloader`. Με κάθε κλήση της μεθόδου `downloadImage`, με παράμετρο το id μιας φωτογραφίας, γίνεται το απαραίτητο αίτημα προς τον εξυπηρετητή για να ληφθεί η φωτογραφία^[40]. Λαμβάνοντας υπόψη τον περιορισμένο χώρο μνήμης που υπάρχει λόγω του ότι η εφαρμογή τρέχει σε κινητές συσκευές, όταν ολοκληρωθεί η λήψη της φωτογραφίας γίνεται η απαραίτητη επεξεργασία^[41] ούτως ώστε να μην υπάρχουν προβλήματα μνήμης. Κατά την διάρκεια της επεξεργασίας λαμβάνεται υπόψη η κύρια μνήμη (RAM) που διαθέτει η συσκευή στην οποία τρέχει η εφαρμογή. Τέλος όταν ολοκληρωθεί η επεξεργασία, η φωτογραφία αποθηκεύεται στην προσωρινή μνήμη (Cache)^[42] μέχρι να τερματιστεί η εφαρμογή έτσι ώστε να μην χρειάζεται να επαναλαμβάνεται η διαδικασία συνεχώς. Δηλαδή με την κλήση της μεθόδου `downloadImage` για φωτογραφία που έχει ήδη ληφθεί τότε ανακτάται από την προσωρινή μνήμη χωρίς να χρειάζεται να επαναληφθεί η διαδικασία.

5.3.2.1 Προβολή κατοικίδιων

Για την προβολή κατοικίδιων δημιουργήθηκε το activity `activityAdoptablePets` και ο adapter `adapterAdoptablePets`. Αρχικά για την διάταξη της προβολής των κατοικίδιων

χρησιμοποιήθηκε ένα GridView^[43,44]. Ένα GridView αποτελείται από πολλά View, και υπεύθυνος για την ανανέωση του περιεχομένου του κάθε View είναι ο adapter που υλοποιήθηκε. Ο adapter είναι υπεύθυνος επίσης για την προσαρμογή των στηλών ανάλογα με το μέγεθος της οθόνης του κινητού (μεγαλύτερη οθόνη περισσότερες στήλες). Όπως επίσης και για την λήψη των φωτογραφιών που προβάλλονται καλώντας την `downloadImage` από την κλάση `BitmapDownloader` που αναλύθηκε πιο πάνω. Η `activityAdoptablePets` είναι υπεύθυνη να πάρει όλα τα απαραίτητα δεδομένα από τον εξυπηρετητή με τη χρήση της κλάσης `JSONRequest` να τα επεξεργαστεί και να τα περάσει σαν παράμετρο στον adapter. Παρόλα αυτά θα ήταν πολύ χρονοβόρα η διαδικασία αν γινόταν η λήψη όλων των δεδομένων στο ξεκίνημα της εφαρμογής και έτσι δημιουργήθηκε η `EndlessScrollListener` που υλοποιεί την abstract κλάση `AbsListView.OnScrollListener`^[45]. Στην ουσία `EndlessScrollListener` είναι ένα `Listener` που ο κώδικας του τρέχει συνεχώς στο `background thread`^[48] καθώς ο χρήστης κάνει `Scroll` και ‘γνωρίζει’ ανά πάσα στιγμή που βρίσκεται ο χρήστης. Όταν ο χρήστης έχει επισκεφτεί τα 2/3 από τα ζώα που έχουν κατεβεί στη συσκευή του τότε ενημερώνει την `activityAdoptablePets` να ξεκινήσει τη διαδικασία για να πάρει τα δεδομένα για τα επόμενα κατοικίδια από τον εξυπηρετητή. Για την προβολή όλων των κατοικίδιων ενός καταφυγίου γίνεται επαναχρησιμοποίηση του `activity` ανακτώντας από τη βάση δεδομένων, μόνο τα κατοικίδια από ένα συγκεκριμένο καταφύγιο αλλάζοντας το `Request` προς τον εξυπηρετητή.

5.3.2.2 Αναζήτηση κατοικίδιων βάσει συγκεκριμένων φίλτρων αναζήτησης

Για την Αναζήτηση κατοικίδιων δημιουργήθηκε το `activity activityAdoptSearch`. Πατώντας ο χρήστης πάνω στο εικονίδιο της αναζήτησης καλείται η `activityAdoptSearch` χωρίς όμως να τερματιστεί το τρέχων `activity`, μεταβαίνει στο βήμα `onPause` στον κύκλο ζωής ενός `activity` (σχήμα 8)^[46]. Όταν ο χρήστης επιλέξει τα φίλτρα αναζήτησης που επιθυμεί η `activityAdoptSearch` επιστρέφει τα αποτελέσματα στο `activity` που την κάλεσε έτσι ώστε να παρθούν από τη βάση δεδομένων τα δεδομένα για τα κατοικίδια που ταιριάζουν στα φίλτρα που επέλεξε ο χρήστης. Για την προβολή των επιλογών στα φίλτρα αναζήτησης χρησιμοποιήθηκαν `Spinner`^[47].

5.3.2.3 Προβολή περισσότερων πληροφοριών για ένα κατοικίδιο

Για την προβολή περισσότερων πληροφοριών για ένα κατοικίδιο δημιουργήθηκε το activity `activityAdoptPetProfile`. Όταν ξεκινήσει το συγκεκριμένο activity ανακτώνται από την βάση δεδομένων όλες οι απαραίτητες πληροφορίες μέσω της `JSONRequest`, όπως και όλες οι φωτογραφίες ενός κατοικίδιου μέσω της `BitmapDownloader`, που περιγράφηκαν πιο πάνω (υπό ενότητα 5.3.2). Η `activityAdoptPetProfile` εκτός από την ανάκτηση όλων των απαραίτητων πληροφοριών είναι υπεύθυνη και για την προβολή τους στον χρήστη σε δομημένη μορφή. Για την εναλλαγή (Swipe) των φωτογραφιών δεν υπήρχε κάποιος Listener του android που να μπορεί να διεκπεραιώσει αυτήν την εργασία έτσι δημιουργήθηκε η κλάση `OnSwipeTouchListener` που υλοποιεί την abstract κλάση `android.view.View.OnTouchListener`^[51]. Στην ουσία η `OnSwipeTouchListener` είναι ένα Listener αναγνωρίζει αν ο χρήστης έκανε `Swipe Right` ή `Swipe Left` ή `Click` στο `ImageView` (φωτογραφία) εκτελώντας τον ανάλογο κώδικα κάθε φορά.

5.3.2.4 Επικοινωνία με το καταφύγιο που φροντίζει ένα ζώο

Για την προβολή περισσότερων πληροφοριών για ένα κατοικίδιο δημιουργήθηκε το activity `activityAdoptionAndSponsor`. Η φόρμα που συμπληρώνει ο χρήστης αποτελείται από `EditText`, `TextView` και `Button`. Όταν ο χρήστης πατήσει το `submit Button` τότε ελέγχεται αν έχει συμπληρώσει όλα τα απαραίτητα πεδία, αν δεν τα έχει συμπληρώσει του εμφανίζεται ένα μήνυμα το οποίο είναι ένα `alertdialog builder`^[53]. Αν έχει συμπληρώσει όλα τα απαραίτητα στοιχεία τότε στέλνεται αυτόματα από το σύστημα ένα email στο καταφύγιο με τα στοιχεία που έδωσε ο χρήστης για να επικοινωνήσει το καταφύγιο με το χρήστη. Αν για οποιοδήποτε λόγο δεν μπορεί να σταλεί το email (π.χ. δεν υπάρχει πρόσβαση στο διαδίκτυο) τότε εμφανίζεται ένα μήνυμα (`alertdialog builder`) για να ενημερώσει το χρήστη.

5.3.2.5 Προβολή πληροφοριών για ένα καταφύγιο

Για την προβολή περισσότερων πληροφοριών για ένα καταφύγιο δημιουργήθηκε το activity `activityShelterDetails`. Όταν ξεκινήσει το συγκεκριμένο activity ανακτώνται από την βάση δεδομένων όλες οι απαραίτητες πληροφορίες μέσω της `JSONRequest`, όπως

και η φωτογραφία προφίλ του καταφύγιου μέσω της BitmapDownloader, που περιεγράφηκαν πιο πάνω (υπό ενότητα 5.3.2). Η activityShelterDetails εκτός από την ανάκτηση όλων των απαραίτητων πληροφοριών είναι υπεύθυνη και για την προβολή τους στον χρήστη σε δομημένη μορφή. Για την δυνατότητα κλήσης στο τηλέφωνο, αποστολής στο ηλεκτρονικό ταχυδρομείο και επίσκεψης στα κοινωνικά δίκτυα του καταφύγιου χρησιμοποιήθηκαν Listener.

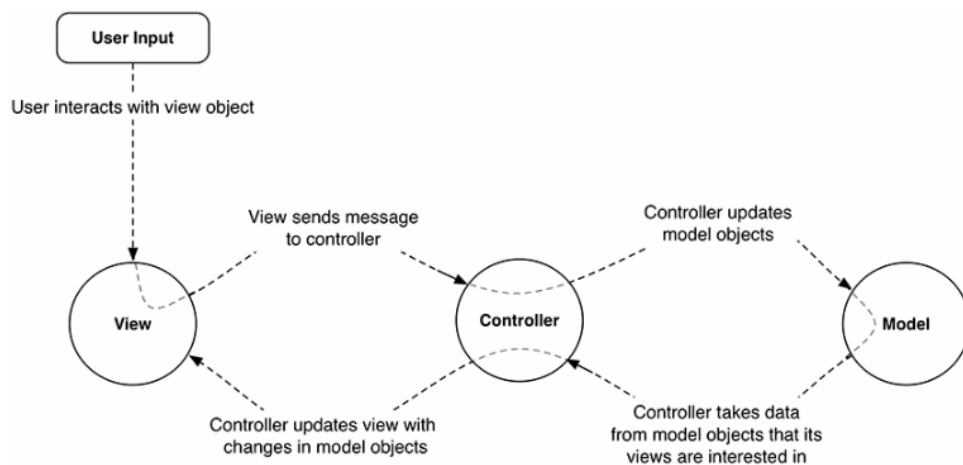
5.3.2.6 Εισαγωγή/τροποποίηση/διαγραφή ζώων από τα καταφύγια

Τα κατοικίδια που προσθέτονται από ένα καταφύγιο κρατούνται σε δυο βάσεις δεδομένων. Μια τοπική βάση δεδομένων η οποία βρίσκεται υλοποιημένη στη συσκευή που τρέχει η εφαρμογή, και μια στον εξυπηρετητή από την οποία αντλούνται οι πληροφορίες που βλέπει ο ανάδοχος. Με αλλά λογία σε αντίθεση με όλες τις λειτουργίες που αφορούν τον υποψήφιο ανάδοχο, τα δεδομένα ενός καταφύγιου κρατούνται και τοπικά δίνοντας έτσι τη δυνατότητα πρόσβασης και τροποποίησης τους χωρίς να υπάρχει πρόσβαση στο διαδίκτυο. Όταν η συσκευή έχει πρόσβαση στο διαδίκτυο τότε στο background thread^[48] και χωρίς να επηρεαστούν καθόλου οι ενέργειες του χρήστη ενημερώνεται η βάση δεδομένων στον εξυπηρετητή για να μπορεί ο υποψήφιος ανάδοχος να αντλεί τις νέες πληροφορίες. Προς υλοποίηση των πιο πάνω έγινε επέκταση στο υπάρχον σύστημα της εταιρίας και επέκταση στην υπάρχουσα τοπική βάση δεδομένων. Για την προβολή των κατοικίδιων που καταχωρήθηκαν από ένα καταφύγιο γίνεται επαναχρησιμοποίηση του activity activityAdoptablePets με τη διαφορά ότι σε αυτήν την περίπτωση οι πληροφορίες αντλούνται από την τοπική βάση δεδομένων.

5.4 Υλοποίηση εφαρμογής iOS

Σε αυτή την ενότητα θα αναλυθεί λεπτομερώς ο τρόπος και οι τεχνικές που χρησιμοποιήθηκαν για την υλοποίηση των κυριότερων λειτουργιών της iOS εφαρμογής.

Πρώτα να αναφέρουμε κάποια βασικά χαρακτηριστικά των iOS εφαρμογών. Μια εφαρμογή για το λειτουργικό σύστημα iOS ακολουθεί το σχεδιαστικό μοντέλο Model-View-Controller^[32] (σχήμα 9) και αποτελείται κυρίως από πολλούς View Controllers^[31], πολλά View^[33] και το storyboard^[34].



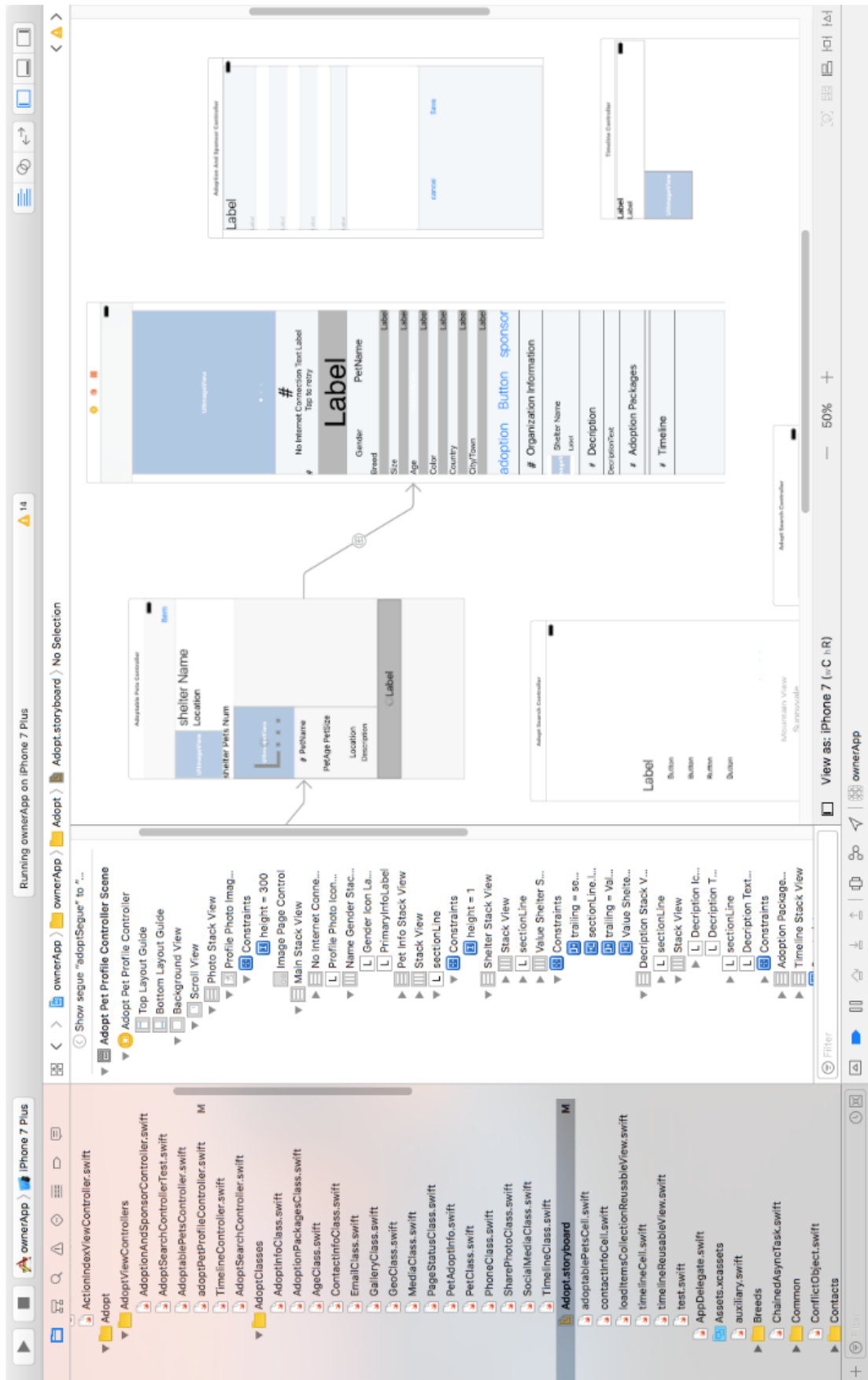
Σχήμα 9: Model-View-Controller

Ένα View Controller είναι μια σελίδα στην εφαρμογή. Ουσιαστικά Controller είναι η κάθε κλάση η οποία διαθέτει δική της γραφική διεπαφή χρήστη. Ένα Controller παρομοίως με τα activity του android μπορεί να τρέχει μόνο στο προσκήνιο και κλείνει όταν ο χρήστης μετακινηθεί σε ένα άλλο controller. Τέλος ένα Controller διαχειρίζεται και ενημερώνει όλα τα Controller που ανήκουν στη συγκεκριμένη γραφική διεπαφή.

5.4.1 Υλοποίηση γραφικής διεπαφής χρήστη

Μια διεπαφή χρήστη για iOS εφαρμογή σχεδιάζεται και υλοποιεί σε ένα αρχείο τύπου storyboard (σχήμα 10). Στο storyboard^[34] υλοποιούνται οι Controller οι οποίοι αποτελούνται κυρίως από διάφορα Views ενώ μπορεί να διαθέτει επίσης Gesture Recognizer. Views είναι τα στοιχεία τα οποία αποτελούν το γραφικό περιβάλλον και

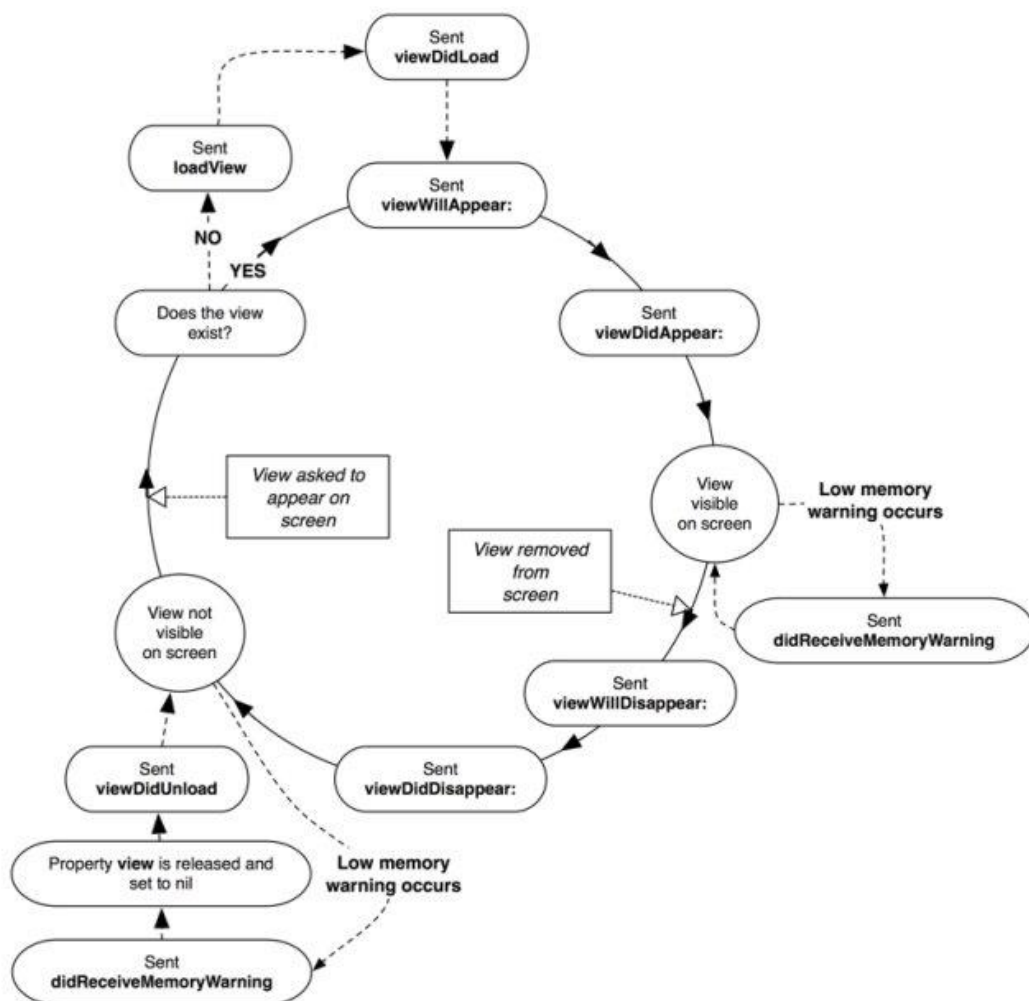
προσφέρουν δυνατότητα αλληλεπίδρασης με την εφαρμογή στο χρήστη. Παραδείγματα views είναι το button, το label, το switch, το ImageView, scroll View κλπ. Ένα από τα πιο συνηθισμένα View είναι το stack View το οποίο τοποθετεί τα views που εσωκλείει στην σειρά, το ένα δίπλα στο άλλο (horizontal stack View) ή το ένα κάτω από το άλλο (Vertical stack View). Εκτός από τα stack View η θέση των View καθώς και της απόστασης μεταξύ τους καθορίζονται από τα constraints. Τα constraints καθορίζουν επίσης το μέγεθος των View. Τέλος Gesture Recognizer «ακούει» και περιμένει για μια συγκεκριμένη ενέργεια από το χρήστη όπως ένα κλικ του χρήστη στην οθόνη αναγνωρίζει ο Tap Gesture Recognizer.



Σχήμα 10: Adopt.storyboard

5.4.2 Υλοποίηση λειτουργιών

Η δημιουργία μια σελίδας στην εφαρμογή για iOS απαιτεί μια συγκεκριμένη δομή. Ξεκινώντας από τον Controller, για την δημιουργία μιας κλάσης-Controller απαιτείται πρωτίστως η κλάση αυτή να είναι κληρονόμος της κλάσης UIViewController. Η UIViewController έχει υλοποιημένες τις μεθόδους `viewDidAppear()`, `viewWillDisappear()`, `viewDidDisappear()`. Ενώ η νέα κλάση πρέπει υποχρεωτικά να υλοποιεί τις μεθόδους `viewWillAppear()` και `viewDidLoad()`, οι μέθοδοι αυτοί καλούνται από το σύστημα κάθε φορά που τρέχει το συγκεκριμένο Controller με μια συγκεκριμένη σειρά που καθορίζεται από τον κύκλο ζωής ενός Controller (σχήμα 11).



Σχήμα 11: iOS Controller Lifecycle

Όπως και με την android εφαρμογή πέρα από τις βασικές λειτουργίες, η υλοποίηση σχεδόν όλων των λειτουργιών της εφαρμογής απαιτεί την επικοινωνία με τη βάση δεδομένων που βρίσκεται εγκατεστημένη στον εξυπηρετητή του της 11pets.

Για την επικοινωνία με τον εξυπηρετητή έχει δημιουργηθεί μια κλάση με το όνομα NetworkRequest η οποία κληρονομεί από την κλάση URLSession^[37]. Η κλάση αυτή επιτρέπει τη λήψη δεδομένων από το διαδίκτυο ασύγχρονα. Συγκεκριμένα μέσω ενός REST API που επιτρέπει την πρόσβαση στα δεδομένα της βάσης δεδομένων χρησιμοποιείται η URLSession για να δημιουργηθούν αιτήματα (Request) και να σταλούν στον εξυπηρετητή έτσι ώστε να λάβουμε ένα αρχείο τύπου JSON^[29] με τα δεδομένα. Το JSON επεξεργάζεται (Parsing) ^[38] έτσι ώστε να παρθούν τα χρήσιμα δεδομένα για την εφαρμογή καταλήγοντας έτσι σε δεκατέσσερις κλάσεις (που είναι οι ίδιες με το android) που περιέχουν όλα τα απαραίτητα δεδομένα.

Για τη λήψη των φωτογραφιών από τον εξυπηρετητή δημιουργήθηκε η κλάση ImageLoader. Με την δημιουργία της ImageLoader με παράμετρο στον κατασκευαστή (Constructor) το id μιας φωτογραφίας, γίνεται το απαραίτητο αίτημα προς τον εξυπηρετητή για να ληφθεί η φωτογραφία^[49]. Όταν ληφθεί η φωτογραφία από τον εξυπηρετητή ακολουθείται η ίδια λογική με την εφαρμογή στο android. Δηλαδή λαβαίνοντας υπόψη τον περιορισμένο χώρο μνήμης που υπάρχει λόγω του ότι η εφαρμογή τρέχει σε κινητές συσκευές, γίνεται η απαραίτητη επεξεργασία στην φωτογραφία ούτως ώστε να μην υπάρχουν προβλήματα μνήμης. Κατά την διάρκεια της επεξεργασίας λαμβάνεται υπόψη η κύρια μνήμη (RAM) που διαθέτει η συσκευή στην οποία τρέχει η εφαρμογή. Τέλος όταν ολοκληρωθεί η επεξεργασία η φωτογραφία αποθηκεύεται στην προσωρινή μνήμη (Cache) χρησιμοποιώντας στην NSCache^[50] μέχρι να τερματίσει η εφαρμογή έτσι ώστε να μην χρειάζεται να επαναλαμβάνεται η διαδικασία συνεχώς. Δηλαδή με τη δημιουργία της κλάσης ImageLoader για φωτογραφία που έχει ήδη ληφθεί από τον εξυπηρετητή τότε ανακτάται από την προσωρινή μνήμη χωρίς να χρειάζεται να επαναληφθεί η διαδικασία.

5.4.2.1 Προβολή κατοικίδιων

Για την προβολή κατοικίδιων δημιουργήθηκε το Controller AdoptablePetsController. Ο AdoptablePetsController είναι υπεύθυνος να πάρει όλα τα απαραίτητα δεδομένα από τον εξυπηρετητή με τη χρήση της κλάσης NetworkRequest. Για την διάταξη της προβολής των κατοικίδιων χρησιμοποιήθηκε ένα collection view^[54]. Ένα collection view αποτελείται από πολλά cell view. Κανένα από τα υπάρχοντα cell που προσφέρει το iOS δεν μπορούσε να αναπαραστήσει τις πληροφορίες με την μορφή που χρειαζόταν για αυτό τον λόγο δημιουργήθηκε η κλάση adoptablePetsCell που στην ουσία είναι ένα custom cell που κληρονομεί από την UICollectionViewCell^[55]. Για την ανανέωση του περιεχομένου του κάθε cell χρειάστηκε να υλοποιηθεί ένας collection view Controller^[56]. Αυτός ο Controller υλοποιήθηκε στην κλάση AdoptablePetsController. Ο Controller είναι υπεύθυνος επίσης για την προσαρμογή των στηλών ανάλογα με το μέγεθος της οθόνης του κινητού (μεγαλύτερη οθόνη περισσότερες στήλες), καθώς επίσης και για την λήψη των φωτογραφιών που προβάλλονται καλώντας την ImageLoader. Σε αντίθεση με την android εφαρμογή ο Controller είναι σε θέση να ‘γνωρίζει’ ανά πάσα στιγμή που βρίσκεται ο χρήστης όταν ο χρήστης κάνει Scroll. Όταν ο χρήστης έχει επισκεφτεί τα 2/3 από τα ζώα που έχουν κατεβεί στην συσκευή του, τότε ο Controller ξεκινάει τη διαδικασία για να πάρει τα δεδομένα για τα επόμενα κατοικίδια από τον εξυπηρετητή. Λόγω του ότι υπεύθυνος για όλη την προβολή των κατοικίδιων είναι μόνο ο AdoptablePetsController (στην android εφαρμογή υπάρχει το Activity, ο ScrollListener και ο adapter) χρειάστηκαν να γίνουν πολλές βελτιστοποιήσεις για να γίνεται το Scroll όσο καλύτερα γίνεται (Smooth Scrolling). Για την προβολή όλων των κατοικίδιων ενός καταφύγιου γίνεται επαναχρησιμοποίηση του controller ανακτώντας από τη βάση δεδομένων μόνο τα κατοικίδια από ένα συγκεκριμένο καταφύγιο αλλάζοντας το Request προς τον εξυπηρετητή.

5.4.2.2 Αναζήτηση κατοικίδιων βάσει συγκεκριμένων φίλτρων αναζήτησης

Για την Αναζήτηση κατοικίδιων δημιουργήθηκε ο Controller AdoptSearchController. Πατώντας ο χρήστης πάνω στο εικονίδιο της αναζήτησης καλείται ο AdoptSearchController παίρνοντας σαν παράμετρο από τον Controller που την κάλεσε

την συνάρτηση που θα εκτελεστεί όταν τελειώσει η αναζήτηση. Όταν τρέξει η `AdoptSearchController` δεν τερματίζεται ο `Controller` που την κάλεσε, μεταβαίνει στο βήμα `viewDisappear` στον κύκλο ζωής ενός `Controller` (σχήμα 11). Για την προβολή των φίλτρων αναζήτησης χρησιμοποιήθηκε ένα `tableview`^[57]. Ένα `tableview`, όπως και ένα `collection view`, αποτελείται από πολλά `cell view`. Για την ανανέωση του περιεχομένου του κάθε `cell` χρειάστηκε να υλοποιηθεί ένας `table view Controller`^[58]. Αυτός ο `Controller` υλοποιήθηκε στην κλάση `AdoptSearchController`. Όπως επίσης και για την προβολή των επιλογών στα φίλτρα αναζήτησης χρησιμοποιήθηκαν `tableviews`. Οι `Controllers` τους υλοποιήθηκαν στην κλάση `OptionPickerViewController`. Όταν ο χρήστης επιλέξει τα φίλτρα αναζήτησης που επιθυμεί τότε ο `AdoptSearchController` τερματίζεται και καλείται η συνάρτηση που περάστηκε σαν παράμετρος στον `AdoptSearchController` έτσι ώστε να παρθούν από τη βάση δεδομένων τα δεδομένα για τα κατοικίδια που ταιριάζουν στα φίλτρα που επέλεξε ο χρήστης.

5.4.2.3 Προβολή περισσότερων πληροφοριών για ένα κατοικίδιο

Για την προβολή περισσότερων πληροφοριών για ένα κατοικίδιο δημιουργήθηκε ο `controller adoptPetProfileController`. Όταν ξεκινήσει ο συγκεκριμένος `controller` ανακτώνται από τη βάση δεδομένων όλες η απαραίτητες πληροφορίες μέσω της `NetworkRequest`, όπως και όλες οι φωτογραφίες ενός κατοικίδιου μέσω της `ImageLoader`, που περιεγράφηκαν πιο πάνω (υπόενότητα 5.4.2). Ο `adoptPetProfileController` εκτός από την ανάκτηση όλων των απαραίτητων πληροφοριών είναι υπεύθυνος και για την προβολή τους στο χρήστη σε δομημένη μορφή. Για την εναλλαγή (Swipe) των φωτογραφιών χρησιμοποιήθηκε ένα `Swipe Gesture Recognizer`^[59]. Ο `Swipe Gesture Recognizer` αναγνωρίζει αν ο χρήστης έκανε `Swipe Right` ή `Swipe Left` στο `ImageView` (φωτογραφία) εκτελώντας τον ανάλογο κώδικα κάθε φορά. Και τέλος για να αναγνωρίζεται αν ο χρήστης πάτησε (Click) πάνω στην φωτογραφία χρησιμοποιήθηκε ένα `Tap Gesture Recognizer`^[60].

5.4.2.4 Επικοινωνία με το καταφύγιο που φροντίζει ένα ζώο

Για την προβολή περισσότερων πληροφοριών για ένα κατοικίδιο δημιουργήθηκε ο `controller AdoptionAndSponsorController`. Η φόρμα που συμπληρώνει ο χρήστης

αποτελείται από TextField, Label και Button. Όταν ο χρήστης πατήσει το submit Button τότε ελέγχεται αν έχει συμπληρώσει όλα τα απαραίτητα πεδία, αν δεν τα έχει συμπληρώσει του εμφανίζεται ένα μήνυμα. Αν έχει συμπληρώσει όλα τα απαραίτητα στοιχεία τότε στέλνεται αυτόματα από το σύστημα ένα email στο καταφύγιο με τα στοιχεία που έδωσε ο χρήστης για να επικοινωνήσει το καταφύγιο με τον χρήστη. Αν για οποιοδήποτε λόγο δεν μπορεί να σταλεί το email (π.χ. δεν υπάρχει πρόσβαση στο διαδίκτυο) τότε εμφανίζεται ένα μήνυμα για να ενημερώσει το χρήστη. Για την εμφάνιση του μηνύματος στο χρήστη χρησιμοποιήθηκε ο UIAlertController^[61] του iOS

5.4.2.5 Προβολή πληροφοριών για ένα καταφύγιο

Για την προβολή περισσότερων πληροφοριών για ένα καταφύγιο δημιουργήθηκε το controller ShelterDetailsController. Όταν ξεκινήσει το συγκεκριμένο activity ανακτώνται από την βάση δεδομένων όλες οι απαραίτητες πληροφορίες μέσω της NetworkRequest, όπως και η φωτογραφία προφίλ του καταφυγίου μέσω της ImageLoader, που περιγράφηκαν πιο πάνω (υπό ενότητα 5.4.2). Η ShelterDetailsController εκτός από την ανάκτηση όλων των απαραίτητων πληροφοριών είναι υπεύθυνη και για την προβολή τους στον χρήστη σε δομημένη μορφή. Για την δυνατότητα κλήσης στο τηλέφωνο, αποστολής στο ηλεκτρονικό ταχυδρομείο και επίσκεψης στα κοινωνικά δίκτυα του καταφυγίου χρησιμοποιήθηκαν Target.

5.4.2.6 Εισαγωγής/τροποποίησης/διαγραφής ζώων από τα καταφύγια

Στην iOS εφαρμογή ακολουθείται η ίδια λογική με την android εφαρμογή. Δηλαδή υπάρχουν δυο βάσεις δεδομένων και τα κατοικίδια που προσθέτονται από ένα καταφύγιο κρατούνται και στις δυο βάσεις δεδομένων. Η μια είναι τοπική και βρίσκεται υλοποιημένη στην συσκευή που τρέχει η εφαρμογή, και μια στον εξυπηρετητή από την οποία αντλούνται οι πληροφορίες που βλέπει ο ανάδοχος. Με άλλα λόγια σε αντίθεση με όλες τις λειτουργίες που αφορούν τον υποψήφιο ανάδοχο, τα δεδομένα ενός καταφυγίου κρατούνται και τοπικά δίνοντας έτσι τη δυνατότητα πρόσβασης και τροποποίησης τους χωρίς να υπάρχει πρόσβαση στο διαδίκτυο. Όταν η συσκευή έχει πρόσβαση στο διαδίκτυο τότε ενημερώνεται η βάση δεδομένων στον εξυπηρετητή για να μπορεί ο υποψήφιος ανάδοχος να αντλεί τις νέες πληροφορίες. Σε αντίθεση με την android

εφαρμογή όταν εκπονήθηκε η διπλωματική εργασία το iOS σύστημα της εταιρίας ήταν υπό κατασκευή έτσι έγινε πλήρης ανάπτυξη της βάσης δεδομένων και όχι επέκταση της. Για την προβολή των κατοικίδιων που καταχωρήθηκαν από ένα καταφύγιο γίνεται επαναχρησιμοποίηση του `Controller AdoptablePetsController` με την διαφορά σε αυτήν την περίπτωση αντλεί τις πληροφορίες από την τοπική βάση δεδομένων.

Κεφάλαιο 6

Συμπεράσματα

6.1 Εισαγωγή	79
6.2 Συμπεράσματα	79
6.3 Μελλοντική Εργασία	80

6.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζονται τα συμπεράσματα που προέκυψαν από την ανάπτυξη της διπλωματικής αυτής εργασίας όπως επίσης και κάποιες μελλοντικές επεκτάσεις που μπορεί να βελτιώσουν το σύστημα που δημιουργήθηκε και τη λειτουργικότητά του.

6.2 Συμπεράσματα

Ο σκοπός της διπλωματικής εργασίας ήταν η ανάπτυξη ενός συστήματος για να καλύπτει την ανάγκη για μηχανογράφηση των κατοικίδιων που φιλοξενούν τα καταφύγια και οι φιλοζωικές οργανώσεις. Αλλά κυρίως τη γρήγορη και αποτελεσματική προώθησή τους προς υιοθεσία για να βρουν ένα καλό σπίτι όσο το δυνατόν πιο νωρίς, μαζεύοντας τα κατοικίδια από όλα τα καταφύγια σε ένα σύστημα έτσι ώστε ο ανάδοχος να μπορεί να τα δει ηλεκτρονικά εύκολα.

Η ευχρηστία του συστήματος είναι ικανοποιητική αφού το σύστημα δημιουργήθηκε με τρόπο ώστε να είναι αρκετά προσιτό προς το χρήστη και να παρέχει την απαραίτητη βοήθεια για τη λειτουργικότητά του.

Η λειτουργικότητα του συστήματος επίσης είναι ικανοποιητική εφόσον μέσα από τις λειτουργίες που παρέχει, ικανοποιεί το σκοπό για τον οποίο δημιουργήθηκε, ελέγχει για

σφάλματα και εξάγει κάποια στατιστικά στοιχεία και αποτελέσματα για περαιτέρω πληροφόρηση.

Ένα μέρος της android εφαρμογής είναι ήδη διαθέσιμο στο play store και πολύ σύντομα θα είναι ολόκληρη διαθέσιμη όπως και η iOS εφαρμογή θα είναι σύντομα διαθέσιμη στο apple store. Στο σύστημα έχουν καταχωρηθεί πάνω από 500 κατοικίδια κυρίως από την Κύπρο, ενώ με την 11pets αρχίσαν να συνεργάζονται και καταφύγια από την Ισπανία και πιο πρόσφατά από Ελλάδα που καταχωρούν συνεχώς νέα ζώα.

6.3 Μελλοντική Εργασία

Η διπλωματική εργασία εκπονήθηκε σε συνεργασία με την 11Pets και μέρος του συστήματος τέθηκε σε λειτουργία. Συνεπώς όπως και με τις πλείστες εφαρμογές που βρίσκονται στο play store και apple store θα συνεχίσει να αναπτύσσεται προσθέτοντας συνεχώς νέες λειτουργίες.

Μια νέα λειτουργία που μπορεί να προσθέσει είναι οι ετικέτες (Tag) για τα ζώα που προθέτονται, όπως για παράδειγμα φιλικό με τα παιδιά. Με την προσθήκη αυτών των ετικετών δίνεται η δυνατότητα προσθήκης νέας πληροφορίας για τα κατοικίδια ή στις πλείστες περιπτώσεις οργάνωσης αυτής της πληροφορίας καθώς τα πλείστα καταφύγια βάζουν αυτές τις πληροφορίες σε κείμενο στην περιγραφή του ζώου.

Ακόμη χρησιμοποιώντας τα Tag μπορεί να προστεθεί ακόμη μια νέα λειτουργικότητα, την δυνατότητα το σύστημα να προτείνει κατοικίδια για υιοθεσία που ταιριάζουν στην περιγραφή του υποψήφιου αναδόχου.

Με το παρόν σύστημα αν ένας ανάδοχος βρει ένα κατοικίδιο που το ενδιαφέρει μπορεί να επιστρέψει στο σύστημα στο μέλλον και να κάνει αναζήτηση για να το βρει. Σε μελλοντική αναβάθμιση του συστήματος αναμένετε να προσθέθει η δυνατότητα δημιουργίας λίστα με αγαπημένα (favorited). Διευκολύνοντας τον χρήστη να επιστρέψει στο σύστημα και να δει τα ζώα που τον ενδιαφέρουν για υιοθεσία.

Στο παρόν στάδιο οι πιο σπάνιες εργασίες εκτελούνται αποκλειστικά από την ιστοσελίδα. Δηλαδή η επεξεργασία των στοιχείων ενός καταφύγιου όπως είναι για παράδειγμα τα στοιχεία επικοινωνίας του ή η δημιουργία λογαριασμού ενός νέου εθελοντή που εργάζεται στο καταφύγιο γίνονται αποκλειστικά από την ιστοσελίδα. Στο μέλλον θα υπάρχει η δυνατότητα αυτές οι λειτουργίες να γίνονται και από τις εφαρμογές διευκολύνοντας έτσι τους χρήστες καθώς όλοι έχουν το κινητό τηλέφωνο τους συνεχώς μαζί τους.

Παρότι το Android και iOS είναι με διαφορά τα δημοφιλέστερα λειτουργικά για κινητές συσκευές, θα ήταν χρήσιμη η υλοποίηση της εφαρμογής και για άλλα λειτουργικά συστήματα όπως Windows Phone και BlackberryOS.

Τέλος μελλοντικά θα προσθέσουν και άλλα φίλτρα αναζήτησης καθώς και νέες λειτουργίες που ζητώνται από τους χρήστες του συστήματος.

Βιβλιογραφία

- [1] Web page 11 Pets: Pets care:
<https://11pets.com/> (Valid on: October/2016)
- [2] Android Development Tools:
<https://developer.android.com/studio/index.html> (Valid on: November/2016)
- [3] Xcode: Complete developer toolset for creating iOS apps:
<https://developer.apple.com/xcode/> (Valid on: February/2017)
- [4] Data flow diagram
https://en.wikipedia.org/wiki/Data_flow_diagram (Valid on: April /2017)
- [5] Gantt chart:
https://en.wikipedia.org/wiki/Gantt_chart (Valid on: January/2017)
- [6] Chapters 6-7, 10 (partially), Software Engineering: A Practitioner's Approach. R. Pressman (7th ed., Mc Graw Hill, 2010).
- [7] Chapters 5-11, P. Stevens, R. Pooley, Using UML Software Engineering with Objects and Components (2nd edition, Addison-Wesley, 2006)
- [8] Chapter 10, Software Engineering: Principles and Practice, H. van Vliet, (3rd rd., John Wiley & Sons, 2008)
- [9] Chapter 5, 7 (partially), Software Engineering, I. Sommerville, (9th ed., Addison-Wesley, 2011).
- [10] Chapter 12-13, 17, Classical and Object-Oriented Software Engineering, S. R. Schach (8th ed., McGraw-Hill, 2011)

- [11] Information about Android
<http://www.android.com/about/> (Valid on: September /2016)

- [12] Information about iOS
<https://en.wikipedia.org/wiki/IOS> (Valid on: September /2016)

- [13] Information about iOS
<https://www.apple.com/lae/ios/> (Valid on: September /2016)

- [14] Programming language Swift
<https://developer.apple.com/swift/> (Valid on: September /2016)

- [15] Java (programming language)
<https://www.java.com/en/> (Valid on: September /2016)

- [16] Chapters 2, 3, Software Engineering: A Practitioner's Approach. R. Pressman (7th ed., Mc Graw Hill,, 2010).

- [17] Chapter 2, 3, Software Engineering: Principles and Practice, H. van Vliet, (3rd ed., John Wiley & Sons, 2008).

- [18] Application: PetFinder.my
<https://play.google.com/store/apps/details?id=com.petfinder.andykoh> (Valid on: October /2016)

- [19] Application: Adopt A Rescue Pet Adoption
<https://play.google.com/store/apps/details?id=mjh.com.petadoption> (Valid on: October /2016)

- [20] Application: Pet Breeds - Find and Adopt
<https://play.google.com/store/apps/details?id=com.findthebest.android.pets>
(Valid on: October /2016)

- [21] Application: Adopt Buy Dogs Shop
<https://play.google.com/store/apps/details?id=com.pinnaculum.nex>
(Valid on: October /2016)

- [22] Application: Adopt a pet
<https://play.google.com/store/apps/details?id=com.adoptujzvieratko>
(Valid on: October /2016)

- [23] Test cases for android
<https://developer.android.com/training/testing/start/index.html>
(Valid on: May/2017)

- [24] Test cases for iOS
https://developer.apple.com/library/content/documentation/DeveloperTools/Conceptual/testing_with_xcode/chapters/04-writing_tests.html (Valid on: May/2017)

- [25] Android Activity
<https://developer.android.com/reference/android/app/Activity.html>
(Valid on: April /2017)

- [26] Android Event Listeners
<https://developer.android.com/guide/topics/ui/ui-events.html>
(Valid on: April /2017)

- [27] Android Services
<https://developer.android.com/guide/components/services.html>
(Valid on: April /2017)

- [28] Android AsyncTask
<https://developer.android.com/reference/android/os/AsyncTask.html>
(Valid on: April /2017)

- [29] Introducing JSON
 <http://www.json.org/>
 (Valid on: February /2017)
- [30] Android JSON
 <https://developer.android.com/reference/org/json/JSONObject.html>
 (Valid on: February /2017)
- [31] iOS View Controllers
 <https://developer.apple.com/library/content/featuredarticles/ViewControllerPGforiPhoneOS/> (Valid on: March /2017)
- [32] Model-View-Controller
 <https://developer.apple.com/library/content/documentation/General/Conceptual/DevPedia-CocoaCore/MVC.html> (Valid on: March /2017)
- [33] iOS Views
 https://developer.apple.com/library/content/documentation/WindowsViews/Conceptual/ViewPG_iPhoneOS/CreatingViews/CreatingViews.html
 (Valid on: March /2017)
- [34] iOS Storyboard
 <https://developer.apple.com/library/content/documentation/General/Conceptual/DevPedia-CocoaApp/Storyboard.html> (Valid on: March /2017)
- [35] Android Xml
 <https://developer.android.com/reference/android/util/Xml.html>
 (Valid on: February /2017)
- [36] Android Layout
 <https://developer.android.com/guide/topics/ui/declaring-layout.html>
 (Valid on: February /2017)

- [37] iOS downloading data with URLSession
<https://developer.apple.com/reference/foundation/urlsession>
(Valid on: May /2017)

- [38] iOS JSON in Swift
<https://developer.apple.com/swift/blog/?id=37>
(Valid on: May /2017)

- [39] Android Volley Request
<https://developer.android.com/training/volley/request.html>
(Valid on: April /2017)

- [40] Android Loading Large Bitmaps Efficiently
<https://developer.android.com/topic/performance/graphics/load-bitmap.html#read-bitmap> (Valid on: April /2017)

- [41] Android Handling Bitmaps
<https://developer.android.com/topic/performance/graphics/index.html>
(Valid on: April /2017)

- [42] Android Caching Bitmaps
<https://developer.android.com/topic/performance/graphics/cache-bitmap.html#disk-cache> (Valid on: April /2017)

- [43] Android GridView
<https://developer.android.com/reference/android/widget/GridView.html>
(Valid on: February /2017)

- [44] Android GridView
<https://developer.android.com/guide/topics/ui/layout/gridview.html>
(Valid on: February /2017)

- [45] Android Scroll Listener
<https://developer.android.com/reference/android/widget/AbsListView.OnScrollListener.html> (Valid on: February /2017)

- [46] Android Getting a Result from an Activity
<https://developer.android.com/training/basics/intents/result.html>
(Valid on: February /2017)

- [47] Android Spinners
<https://developer.android.com/guide/topics/ui/controls/spinner.html>
(Valid on: February /2017)

- [48] Android Processes and Threads
<https://developer.android.com/guide/components/processes-and-threads.html>
(Valid on: February /2017)

- [49] Chapter 19, iOS Programming: The Big Nerd Ranch Guide (5th rd., Christian Keur & Aaron Hillegass, 2015)

- [50] iOS NSCache
<https://developer.apple.com/reference/foundation/nscache>
(Valid on: May /2017)

- [51] Android OnTouchListener
<https://developer.android.com/reference/android/view/View.OnTouchListener.html> (Valid on: February /2017)

- [52] Chapter 2, iOS Programming: The Big Nerd Ranch Guide (5th rd., Christian Keur & Aaron Hillegass, 2015)

- [53] Android AlertDialog builder
<https://developer.android.com/reference/android/app/AlertDialog.Builder.html>
(Valid on: February /2017)

- [54] iOS UICollectionView
<https://developer.apple.com/reference/uikit/UICollectionView>
(Valid on: April /2017)
- [55] iOS UICollectionViewCell
<https://developer.apple.com/reference/uikit/UICollectionViewCell>
(Valid on: April /2017)
- [56] Chapter 20, iOS Programming: The Big Nerd Ranch Guide (5th rd., Christian Keur & Aaron Hillegass, 2015)
- [57] iOS UITableView
<https://developer.apple.com/reference/uikit/UITableView>
(Valid on: April /2017)
- [58] Chapter 9, 10, iOS Programming: The Big Nerd Ranch Guide (5th rd., Christian Keur & Aaron Hillegass, 2015)
- [59] iOS UISwipeGestureRecognizer
<https://developer.apple.com/reference/uikit/UISwipeGestureRecognizer>
(Valid on: May /2017)
- [60] iOS UITapGestureRecognizer
<https://developer.apple.com/reference/uikit/UITapGestureRecognizer>
(Valid on: May /2017)
- [61] iOS UIAlertController
<https://developer.apple.com/reference/uikit/UIAlertController>
(Valid on: May /2017)
- [62] Operating system market share survey
<https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=10&qpcustomd=1> (Valid on: May /2017)