



ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Π.Μ.Σ. «ΨΗΦΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΚΑΙ ΥΠΗΡΕΣΙΕΣ»

ΚΑΤΕΥΘΥΝΣΗ: ΠΡΟΗΓΜΕΝΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ιούνιος 2020

Ανάπτυξη πληροφοριακού συστήματος και λογισμικού αναγνώρισης προσώπου

Επιβλέποντες Καθηγητές: Κυριαζής
Δημοσθένης, Μενύχτας Ανδρέας

Μαθιούδης Νικόλαος (ΜΕ 1746)
nmathioudis@gmail.com

Περιεχόμενα

1. ΕΙΣΑΓΩΓΗ	5
2. ΜΕΛΕΤΗ ΚΑΤΑΣΤΑΣΗΣ	7
2.1. ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΣΩΠΟΥ – ΥΠΑΡΧΟΥΣΑ ΚΑΤΑΣΤΑΣΗ	7
2.2. ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΣΩΠΟΥ – ΥΓΕΙΑ	8
2.3. ΠΩΣ ΛΕΙΤΟΥΡΓΕΙ	9
2.4. ΤΕΧΝΟΛΟΓΙΚΟ ΥΠΟΒΑΘΡΟ	10
2.4.1.1. <i>OpenCv</i>	10
2.4.2. <i>TensorFlow</i>	12
2.4.3. <i>Επιλογή Βιβλιοθήκης</i>	12
3. ΠΕΡΙΓΡΑΦΗ ΣΥΣΤΗΜΑΤΟΣ.....	13
3.1. ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΣΥΣΤΗΜΑΤΟΣ.....	13
3.2. ΡΟΛΟΙ	13
3.3. ΛΕΙΤΟΥΡΓΙΚΕΣ ΑΠΑΙΤΗΣΕΙΣ	15
3.3.1. <i>Εγγραφή στο Σύστημα</i>	15
3.3.2. <i>Είσοδος στο σύστημα με τον ρόλο της γραμματείας</i>	16
3.3.3. <i>Είσοδος στο σύστημα με τον ρόλο του γιατρού</i>	17
3.3.4. <i>Είσοδος στο σύστημα με τον ρόλο του φροντιστή - νοσηλεύτη</i>	18
3.3.5. <i>Είσοδος στο σύστημα με τον ρόλο του προσωπικού ασφαλείας</i>	19
3.4. ΑΠΑΙΤΗΣΕΙΣ ΑΣΦΑΛΕΙΑΣ – ΑΞΙΟΠΙΣΤΙΑΣ	20
3.5. ΣΥΝΤΗΡΗΣΗ ΣΥΣΤΗΜΑΤΟΣ	21
3.6. ΕΥΧΡΗΣΤΙΑ	22
3.7. ΤΕΧΝΟΛΟΓΙΕΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ.....	22
3.8. ΔΙΑΓΡΑΜΜΑΤΑ ΡΟΗΣ ΔΙΕΡΓΑΣΙΩΝ.....	24
3.9. ΑΣΦΑΛΕΙΑ ΔΕΔΟΜΕΝΩΝ.....	25
3.10. ΜΕΛΕΤΗ ΕΦΙΚΤΟΤΗΤΑΣ	26
3.10.1. <i>Τεχνική Εφικτότητα</i>	26
3.10.2. <i>Οικονομική Εφικτότητα</i>	27
3.10.3. <i>Λειτουργική Εφικτότητα</i>	27
4. ΦΥΣΙΚΟΣ ΣΧΕΔΙΑΣΜΟΣ – ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ.....	29
4.1. ΣΥΣΤΗΜΑ.....	29
4.2. ΕΠΙΚΟΙΝΩΝΙΑ ΣΥΣΤΗΜΑΤΩΝ.....	31
5. ΒΑΣΗ MONGODB.....	32
5.1. ΣΥΛΛΟΓΕΣ ΒΑΣΗΣ.....	32
5.2. ΣΥΝΔΕΣΗ ΜΕ ΣΥΣΤΗΜΑΤΑ.....	33
6. WEB APPLICATION.....	34
6.1. ΥΠΟΣΥΣΤΗΜΑ SERVER.....	34
6.2. ΥΠΟΣΥΣΤΗΜΑ CLIENT	36
6.3. ΠΑΚΕΤΑ (MIDDLEWARE)	38
6.4. WEB SERVICES	38
7. FACE RECOGNITION	40
7.1. ΕΓΚΑΤΑΣΤΑΣΗ ΣΥΣΤΗΜΑΤΟΣ	40
7.2. ΑΝΑΛΥΣΗ ΚΩΔΙΚΑ ΣΥΣΤΗΜΑΤΟΣ.....	42
7.3. ΛΕΙΤΟΥΡΓΙΑ FACE RECOGNITION.....	43
8. ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ	46
8.1. ΑΝΑΓΚΗ ΑΝΑΛΥΣΗΣ ΔΕΔΟΜΕΝΩΝ	46

8.2.	ΣΕΝΑΡΙΑ ΑΝΑΛΥΣΗΣ.....	46
8.3.	ΣΥΝΑΡΤΗΣΕΙΣ ΑΝΑΛΥΣΗΣ	47
8.4.	ΕΜΦΑΝΙΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ	48
9.	ΣΤΑΤΙΣΤΙΚΑ ΣΥΣΤΗΜΑΤΟΣ.....	52
9.1.	ΧΡΟΝΟΣ ΕΝΑΡΞΗΣ ΠΡΟΓΡΑΜΜΑΤΟΣ.....	52
9.2.	ΧΡΟΝΟΣ ΑΝΑΓΝΩΡΙΣΗΣ ΠΡΩΤΟΥ ΠΡΟΣΩΠΟΥ ΑΠΟ ΤΗΝ ΕΝΑΡΞΗ ΤΟΥ ΒΙΝΤΕΟ	54
9.3.	ΕΠΙΤΥΧΙΑ ΑΝΑΓΝΩΡΙΣΗΣ ΕΝΤΟΣ ΜΙΚΡΟΥ ΧΡΟΝΙΚΟΥ ΔΙΑΣΤΗΜΑΤΟΣ.....	55
9.4.	ΑΠΟΔΟΣΗ ΒΙΝΤΕΟ.....	55
10.	ΕΠΙΛΟΓΟΣ	57
10.1.	ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ	57
10.2.	ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ	57
10.3.	ΑΣΦΑΛΕΙΑ ΕΠΙΧΕΙΡΗΣΗΣ	57
10.4.	ΣΥΜΠΕΡΑΣΜΑΤΑ	58
11.	ΠΑΡΑΡΤΗΜΑ – USER MANUAL.....	59
11.1.	ΕΙΣΟΔΟΣ ΣΤΟ ΣΥΣΤΗΜΑ	59
11.2.	ΕΓΓΡΑΦΗ ΝΕΟΥ ΧΡΗΣΤΗ.....	61
11.3.	ΚΑΡΤΕΛΕΣ ΔΙΑΜΕΝΟΝΤΩΝ/ΑΣΘΕΝΩΝ	62
11.4.	ΠΛΗΡΟΦΟΡΙΕΣ ΔΙΑΜΕΝΟΝΤΑ	64
11.5.	ΠΡΟΣΘΗΚΗ ΔΡΑΣΤΗΡΙΟΤΗΤΩΝ.....	65
11.6.	ΠΡΟΣΘΗΚΗ ΔΡΑΣΤΗΡΙΟΤΗΤΩΝ ΑΝΑ ΔΙΑΜΕΝΟΝΤΑ	66
11.7.	ΣΤΑΤΙΣΤΙΚΑ.....	67
11.8.	ΚΑΝΟΝΕΣ	68
12.	ΒΙΒΛΙΟΓΡΑΦΙΑ – ΠΗΓΕΣ.....	69

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εικόνα 1:	Είσοδος - Έξοδος στο πρώτο στάδιο	9
Εικόνα 2:	Δημιουργία σημείων στο πρόσωπο.....	10
Εικόνα 3:	Νευρωνικό δίκτυο	11
Εικόνα 4:	Κατηγοριοποίηση αντικειμένων	11
Εικόνα 5:	Εγγραφή στο σύστημα.....	15
Εικόνα 6:	Είσοδος και διεργασίες Γραμματείας	16
Εικόνα 7:	Είσοδος και διεργασίες Ιατρών	17
Εικόνα 8:	Είσοδος και διεργασίες Φροντιστών	19
Εικόνα 9:	Είσοδος και διεργασίες προσωπικού ασφαλείας.....	20
Εικόνα 10:	Διάγραμμα Ροής Διεργασιών 1.....	24
Εικόνα 11:	Διάγραμμα Ροής Διεργασιών 2.....	25
Εικόνα 12:	Δομή Συστήματος	29
Εικόνα 13:	Δομή Web Application.....	30
Εικόνα 14:	Πίνακες βάσης Mongo.....	32
Εικόνα 15:	Δομή υποσυστήματος Server	34
Εικόνα 16:	Δομή υποσυστήματος Client	36
Εικόνα 17:	Διάταξη 1 Face Recognition	40
Εικόνα 18:	Διάταξη 2 Face_recognition.....	41
Εικόνα 19:	Δοκιμαστική Διάταξη	42
Εικόνα 20:	Παράδειγμα Live εικόνας Face Recognition	43
Εικόνα 21:	Αποθήκευση δεδομένων στην βάση	44

Εικόνα 22: 4165 εγγραφές σε 10 ώρες.....	46
Εικόνα 23: data_normalization()	47
Εικόνα 24: Επιλογή πελάτη για στατιστικά	48
Εικόνα 25: Πίτα δεδομένων προτίμησης κάμερας	49
Εικόνα 26: Επιλογή κάμερας για απενεργοποίηση στην ανάλυση.....	50
Εικόνα 27: Διάγραμμα Χρόνου - Κάμερας.....	51
Εικόνα 28: Πίνακας και διαγράμματα στατιστικών 1	52
Εικόνα 29: Πίνακας και διαγράμματα στατιστικών 2	53
Εικόνα 30: Πίνακας και διαγράμματα στατιστικών 3	54
Εικόνα 31: Διάγραμμα στατιστικών 4	55

1. Εισαγωγή

Η δημιουργία πληροφοριακών συστημάτων που καλύπτουν μεγάλο εύρος αναγκών επιχειρήσεων είναι σύνθηρες φαινόμενο τις τελευταίες δεκαετίες. Η συνεχής ανάπτυξη της τεχνολογίας όμως έχει δημιουργήσει νέες δυνατότητες σε κάθε τομέα εργασίας, με αποτέλεσμα τα πληροφοριακά συστήματα που δημιουργούνται να ενσωματώνουν νέες τεχνολογίες και να προσφέρουν πολύ περισσότερο από ότι ένα παραδοσιακή εφαρμογή γραφείου.

Στον τομέα της υγείας και της περίθαλψης και συγκεκριμένα σε ένα σύγχρονο κέντρο ευγηρίας δημιουργούνται ανάγκες που ξεφεύγουν από την απλή διαμονή ενός ηλικιωμένου και παροχή φροντίδας. Υπάρχει συνεχής ιατρική υποστήριξη από γιατρούς όπως παθολόγος, νευρολόγος και ψυχίατρος, υποστήριξη από ψυχολόγο, περιλαμβάνονται τμήματα μακροχρόνιας και βραχυχρόνιας θεραπείας από εξειδικευμένο προσωπικό, παρέχονται ιατρικές εξετάσεις από εξοπλισμό που βρίσκεται εντός του κέντρου και υπάρχει συνεχής επίβλεψη των ηλικιωμένων – ασθενών.

Ο οίκος ευγηρίας έχοντας πολλά χρόνια στον χώρο έχει αναλάβει την φροντίδα αρκετών ανθρώπων προσέχοντας τις συνθήκες διαβίωσής τους και την υγεία τους. Η φιλοξενία ανθρώπων μεγάλης ηλικίας στον χώρο δημιουργεί πολλές απαιτήσεις τόσο σε ανθρώπινο δυναμικό όσο και σε φυσικούς πόρους όπως τρόφιμα, φάρμακα, δωμάτια.

Μέχρι τη στιγμή που ανατίθεται το έργο το μεγαλύτερο πρόβλημα που παρατηρήθηκε ήταν η αρχειοθέτηση των φακέλων των φιλοξενούμενων στον οίκο. Όλη η διαδικασία είναι χειρόγραφη, η αποθήκευση γίνεται σε ειδικό δωμάτιο αρχείου και η ενημέρωση των καρτελών είναι ιδιαίτερα χρονοβόρα. Παρατηρήθηκαν επίσης σημαντικά προβλήματα στην οργάνωση. Η διαδικασία ένταξης ενός ανθρώπου σε μια από τις δραστηριότητες γίνεται απλά μέσω μιας σημείωσης σε ημερολόγιο. Αξίζει να σημειωθεί ότι η απόφαση της επιχείρησης να επεκταθεί και να προσφέρει πλήρη ιατρική φροντίδα επιβάρυνε την κατάσταση και για αυτό εμφανίστηκε άμεσα η ανάγκη εκσυγχρονισμού και δημιουργίας πληροφοριακών συστημάτων.

Σημαντική παρατήρηση είναι πως μέχρι και τη στιγμή ανάθεσης δεν υπήρχε κάποιο σύστημα καταγραφής δραστηριοτήτων που είναι δημοφιλείς, προσωπικών επιλογών των ενοίκων καθώς και κάποιος τρόπος μελέτης των παραπάνω δεδομένων. Η επιλογή νέων προγραμμάτων γίνεται αποκλειστικά σύμφωνα με τις απόψεις του προσωπικού και τις παρατηρήσεις τους για τις συνήθειες των φιλοξενούμενων.

Σκοπός της εργασίας είναι η δημιουργία μιας πλατφόρμας που θα διευκολύνει όλες τις διαδικασίες της επιχείρησης και η δημιουργία ενός συστήματος αναγνώρισης προσώπου για τις κάμερες της επιχείρησης. Η δημιουργία πλατφόρμας θα βοηθήσει στην δημιουργία μιας βάσης δεδομένων με όλες τις απαραίτητες

πληροφορίες που πρέπει να έχει ο κάθε φιλοξενούμενος. Έτσι αποφεύγεται η συνεχής αύξηση του όγκου των φακέλων και μειώνεται ο χρόνος αναζήτησης πληροφοριών διευκολύνοντας την δουλειά των εργαζομένων. Η σύνδεση των καμερών με ένα σύστημα αναγνώρισης προσώπου βοηθάει την επιχείρηση να εξελιχθεί χρησιμοποιώντας τα δεδομένα τα οποία θα καταγράφονται. Μέσα από αυτά υπάρχει η δυνατότητα να εξαγονται συμπεράσματα για την προτίμηση, την συμπεριφορά και την ψυχολογία των ενοίκων. Επίσης προσφέρει νέα δεδομένα στον τομέα της ασφάλειας. Ουσιαστικά χρησιμοποιεί της δυνατότητες που προσφέρει η ανάλυση δεδομένων ώστε να βελτιώσει τις συνθήκες διαβίωσης των ενοίκων. Με αυτό τον τρόπο οι υπηρεσίες που προσφέρει ο οίκος ευγηρίας γίνονται ποιοτικότερες και ελκυστικότερες για νέες εγγραφές ατόμων. Δεν πρόκειται για ένα νέο σύστημα καθώς χρησιμοποιείται ήδη σε άλλους τομείς, όμως δεν υπάρχει ακόμα σε οίκους ευγηρίας. Δημιουργείται μια ευκαιρία να χρησιμοποιηθεί η τεχνογνωσία που ήδη υπάρχει και να ενσωματωθεί αρμονικά με σκοπό να επιλύσει προβλήματα και να ωφελήσει την επιχείρηση έναντι του ανταγωνισμού.

2. Μελέτη Κατάστασης

Στο κεφάλαιο αυτό θα μελετήσουμε την υπάρχουσα κατάσταση, κυρίως για το σύστημα της αναγνώρισης προσώπου, τόσο στον τομέα των υπηρεσιών υγείας όσο και σε διαφορετικούς τομείς. Δεν γίνεται ιδιαίτερη αναφορά στο πληροφοριακό σύστημα καθώς πρόκειται για πλατφόρμες που υπάρχουν στις περισσότερες σύγχρονες επιχειρήσεις με βασική λειτουργία την διαχείριση δεδομένων. Θα γίνει όμως αναφορά στους τρόπους σύνδεσης με το face recognition καθώς και στους τρόπους ανάλυσης δεδομένων που ήδη υπάρχουν.

2.1. Αναγνώριση προσώπου - Υπάρχουσα Κατάσταση

Το λογισμικό αναγνώρισης προσώπου δεν είναι μια καινούρια τεχνολογία. Υπάρχει αρκετά χρόνια στον χώρο και έχει χρησιμοποιηθεί σε μεγάλο εύρος εφαρμογών. Μερικά χαρακτηριστικά παραδείγματα:

- **Κινητά τηλέφωνα:** Χρησιμοποιείται ευρέως ως μέσο ασφαλείας στο ξεκλείδωμα των συσκευών. Αν και σε αρκετές περιπτώσεις χρήζει βελτίωσης είναι ένας ισχυρός τρόπος προστασίας των προσωπικών δεδομένων του χρήστη σε περίπτωση κλοπής.
- **Μέσα κοινωνικής δικτύωσης:** Γίνεται χρήση του λογισμικού για αναγνώριση προσώπων αυτόματα πάνω στις φωτογραφίες που ανεβάζουν οι χρήστες.
- **Αεροδρόμια:** Στα αεροδρόμια πέρα από την ύπαρξη αναγνώρισης προσώπου ώστε να ελέγχονται όσοι εισέρχονται στον χώρο με βάση το ποινικό τους μητρώο εφόσον είναι καταχωρημένοι στις αντίστοιχες βάσεις έχει και άλλες χρήσεις. Αεροπορικές εταιρίες το χρησιμοποιούν για να κάνουν την επιβίβαση επιβατών και παράδοση αποσκευών πολύ πιο γρήγορη.
- **Αυτοκίνητα:** Σε αρκετά καινούρια αυτοκίνητα η χρήση κάμερας για αναγνώριση του οδηγού χρησιμοποιείται ως επιπλέον μέτρο για την αντιμετώπιση κινδύνου κλοπής.
- **Σχολεία:** Ύστερα από ορισμένα συμβάντα κυρίως στις Ηνωμένες Πολιτείες Αμερικής το λογισμικό χρησιμοποιήθηκε στα σχολεία με σκοπό να βελτιώσει την ασφάλεια των μαθητών. Έτσι άτομα που θεωρούνται επικίνδυνα για τα παιδιά αναγνωρίζονται άμεσα με την είσοδό τους στον χώρο. Ενημερώνοντας το προσωπικό ασφαλείας μπορούν να ληφθούν άμεσα όλα τα μέτρα που χρειάζονται ώστε να μειωθεί ο κίνδυνος νέου συμβάντος. Επιπροσθέτως σε ορισμένα σχολεία χρησιμοποιείται ως σύστημα απουσιολογίου. Έτσι εξασφαλίζεται η παρουσία των παιδιών στα μαθήματα και αποφεύγονται οι χαμένες διδακτικές ώρες.
- **Αναζήτηση χαμένων ατόμων:** Σε χώρες που η χρήση του λογισμικού αναγνώρισης προσώπου είναι επιτρεπτή και έχει αναπτυχθεί εντοπίζονται

άτομα που θεωρούνται χαμένα μέσω του συστήματος των καμερών. Χαρακτηριστικό παράδειγμα στην Ινδία όπου τρεις χιλιάδες (3.000) χαμένα παιδιά βρέθηκαν μέσα σε μόνο τέσσερις μέρες.

- Καζίνο: Στις συγκεκριμένες επιχειρήσεις η λειτουργία γίνεται για πρόβλεψη κινδύνου από άτομα που ίσως έχουν κάποιο ιστορικό κλοπών ή εγκληματικότητας γενικότερα. Όμως ιδιαίτερη σημασία έχει η χρήση του για την συγκέντρωση δεδομένων που αναλύουν την συμπεριφορά των παικτών ώστε να δημιουργούνται και τα κατάλληλα προωθητικά πακέτα από τα τμήματα μάρκετινγκ.
- Υγεία: Χρησιμοποιείται για την αναγνώριση χαρακτηριστικών των ασθενών που μπορεί να παρατηρηθούν πιο εύκολα από τον υπολογιστή παρά ίσως από έναν άνθρωπο. Έτσι προλαμβάνονται ασθένειες διότι δεν χρειάζεται πλέον η αναμονή για το ραντεβού.

Τα παραπάνω είναι μόνο μερικά από τα παραδείγματα της χρήσης του προγράμματος αναγνώρισης προσώπου. Οι εφαρμογές είναι αμέτρητες και μπορεί σε έναν κλάδο να υπάρχει παραπάνω από ένας τρόπος χρήσης της εφαρμογής.

2.2. Αναγνώριση προσώπου – Υγεία

Στον τομέα της υγείας, όπως γενικότερα μπορεί να ενταχθεί ένας οίκος ευημερίας, ένα σύστημα σαν αυτό που μελετάμε μπορεί να προσφέρει πολλές νέες δυνατότητες.

Στον τομέα της ασφάλειας των εγκαταστάσεων μπορεί να γίνει χρήση προγραμμάτων που ήδη έχουν δημιουργηθεί από οργανισμούς ασφάλειας ώστε να παρακολουθείται συνεχώς ο χώρος και να εξασφαλίζεται η ακεραιότητα των μηχανημάτων που έχουν ιδιαίτερα υψηλό κόστος τόσο από εξωτερικούς όσο και από εσωτερικούς παράγοντες. Δεν είναι λίγες οι φορές που ακόμα και προσωπικό ενός νοσοκομείου έχει κλέψει προμήθειες που προορίζονται για τους ασθενείς. Μια συνεχής καταγραφή του χώρου και των θέσεων που βρίσκονται οι πελάτες αλλά και οι υπάλληλοι μπορεί να συμβάλλει στην καταστολή ανεπιθύμητων καταστάσεων πριν ακόμα δημιουργηθούν.

Όσον αφορά ασθενείς η ύπαρξη μιας άρτιας βάσης με φωτογραφίες και η χρήση της αναγνώρισης προστατεύει την επιχείρηση από περιπτώσεις εξαπάτησης ταυτότητας. Έτσι θα μπορεί με πολύ μεγάλη πιθανότητα το σύστημα να καταλαβαίνει αν ο ασθενής δεν είναι αυτός που ισχυρίζεται ότι είναι. Με τις κατάλληλες διαδικασίες σε μεταγενέστερο χρόνο μπορεί να γίνει η εξακρίβωση των στοιχείων ώστε να εξασφαλιστεί ότι δεν είναι μια από τις περιπτώσεις που απλά είχες λάθος το σύστημα.

Στον τομέα της αντιμετώπισης παθήσεων χρησιμοποιείται για την αναγνώριση χαρακτηριστικών των ασθενών που μπορεί να παρατηρηθούν πιο εύκολα από τον

υπολογιστή παρά ίσως από έναν άνθρωπο. Έτσι προλαμβάνονται ασθένειες και δεν χρειάζεται πλέον η αναμονή για το ραντεβού.

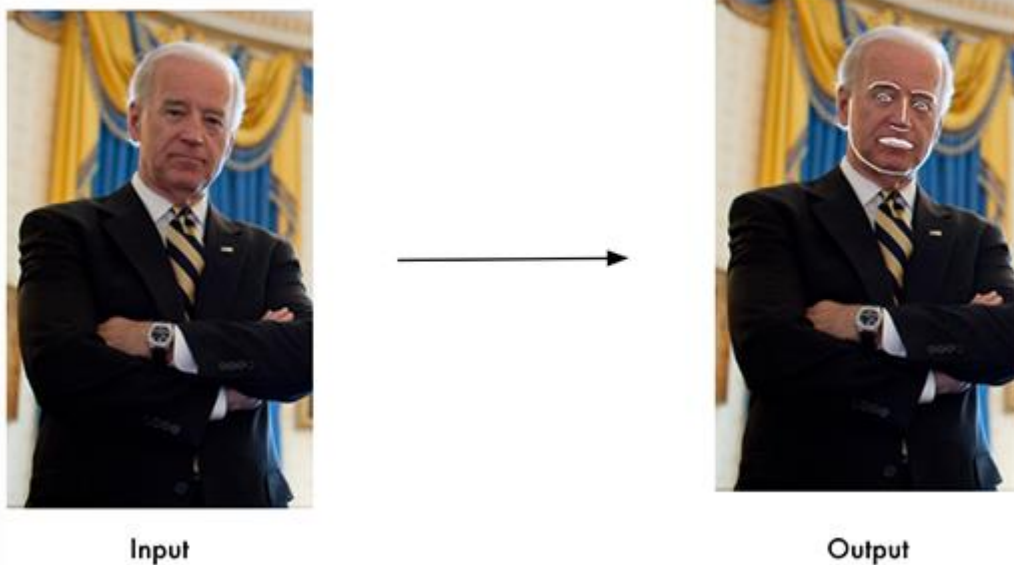
Η δημιουργία στατιστικών στοιχείων για τον τρόπο που κινούνται οι ασθενείς ή στην περίπτωση του οίκου ευγηρίας οι ένοικοι μπορεί να χρησιμοποιηθεί για την εξαγωγή συμπερασμάτων σχετικά με τις συνήθειες και τις προτιμήσεις των ανθρώπων.

2.3. Πως λειτουργεί

Η αναγνώριση προσώπου γίνεται μέσω εφαρμογών που δημιουργούνται από διάφορες επιχειρήσεις από το μηδέν ή μέσω έτοιμων βιβλιοθηκών που προσφέρουν έτοιμες τις επιλογές και απλά προσαρμόζονται στις απαιτήσεις του κάθε χρήστη. Όλες οι εφαρμογές όμως έχουν μια κοινή λογική στον τρόπο λειτουργίας, ο οποίος αποτελείται από τα εξής στάδια:

- Αναγνώριση προσώπων σε εικόνες και βίντεο.
- Αποθήκευση χαρακτηριστικών προσώπου για κάθε αναγνώριση.
- Αντιστοίχιση με βάση μια υπάρχουσα βάση δεδομένων προσώπων.

Στο πρώτο στάδιο ο κώδικας λαμβάνει ως είσοδο μια εικόνα ή ένα βίντεο. Προσπαθεί να κάνει την ίδια ανάλυση που κάνει και ο εγκέφαλος μας δηλαδή να εντοπίσει τα χαρακτηριστικά που δημιουργούν ένα πρόσωπο όπως μάτια, μύτη, χρώμα δέρματος, χείλη. Πραγματοποιώντας όλη την διαδικασία σε όλη την έκταση της εικόνας ουσιαστικά εντοπίζει όλα τα πρόσωπα που υπάρχουν.



Εικόνα 1: Είσοδος - Έξοδος στο πρώτο στάδιο

Στο δεύτερο στάδιο γίνεται ανάλυση του κάθε προσώπου σε ένα σύνολο από σημεία (64 ή 128 ή 256) τα οποία αντιπροσωπεύουν πλέον το πρόσωπο. Τα σημεία

αυτά περιέχουν χαρακτηριστικά όπως η διάταξη του προσώπου(σχήμα, αποστάσεις) το χρώμα, τα μαλλιά. Έτσι πλέον στον κώδικα κάθε ένα από τα πρόσωπα εμφανίζεται όχι ως μια φωτογραφία αλλά ως ένα σύνολο σημείων. Η διαδικασία αυτή δεν γίνεται μόνο για τα άτομα της φωτογραφίας αλλά έχει ήδη πραγματοποιηθεί και για τα άτομα που έχουν ήδη αποθηκευμένες φωτογραφίες στη βάση δεδομένων



Εικόνα 2: Δημιουργία σημείων στο πρόσωπο

Στο τρίτο στάδιο πραγματοποιείται η σύγκριση των σημείων από την εικόνα στην ζωντανή μετάδοση με τα σημεία των εικόνων της βάσης και γίνεται σύγκριση. Αν κάποιο αποτέλεσμα είναι αρκετά κοντά τότε γίνεται ταυτοποίηση και εμφανίζεται το όνομα.

2.4. Τεχνολογικό Υπόβαθρο

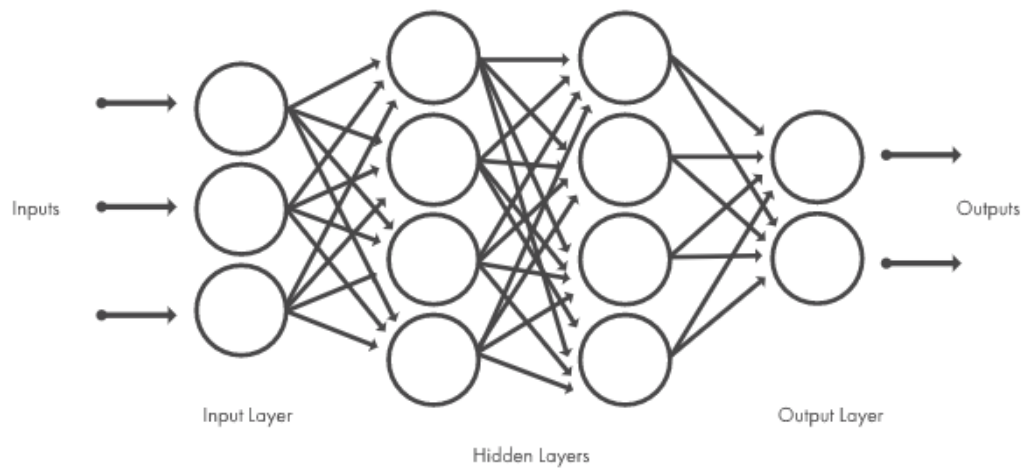
Η ύπαρξη πολλών βιβλιοθηκών για αναγνώριση προσώπου δεν σημαίνει ότι στηρίζονται όλες σε διαφορετικούς αλγορίθμους. Στις περισσότερες βιβλιοθήκες γίνεται η χρήση είτε της OpenCv είτε της TensorFlow.

2.4.1.1. OpenCv

Η OpenCv χαρακτηρίζεται ως "*Open Source Computer Vision Library*" δηλαδή μια ανοιχτή προς όλους βιβλιοθήκη για ανάλυση εικόνων και βίντεο. Γίνεται χρήση της όχι μόνο για αναγνώριση προσώπου αλλά και για αναγνώριση αντικειμένων, συναισθημάτων και άλλα.

Η OpenCv χρησιμοποιεί ουσιαστικά της τεχνικής Deep Learning. Η τεχνική αυτή χρησιμοποιεί τους αλγόριθμους για να μιμηθεί τον ανθρώπινο εγκέφαλο και να φτιάξει ένα σύνολο από νευρωνικά δίκτυα τα οποία επεξεργάζονται τις πληροφορίες

που δέχεται ως είσοδο. Ουσιαστικά αναλύει την κάθε εικόνα σε pixel και το κάθε pixel αποτελεί έναν νευρώνα. Έτσι για μια εικόνα με 1000 pixel έχουμε 1000 αρχικούς νευρώνες.



Εικόνα 3: Νευρωνικό δίκτυο

Τα δεδομένα αυτά μέσω ενός συστήματος μεταφοράς σε κρυφούς νευρώνες καταλήγουν σε μια μορφή ομαδοποίησης και έτσι δημιουργούνται κατηγορίες αντικειμένων.



Εικόνα 4: Κατηγοριοποίηση αντικειμένων

Έτσι οι βιβλιοθήκες που υπάρχουν και στηρίζονται στην OpenCv ουσιαστικά αναγνωρίζουν τα πρόσωπα και προχωράνε στην επεξεργασία των προσώπων η κάθε μια με την δικιά της προσέγγιση.

2.4.2. TensorFlow

Η TensorFlow θεωρείται μια ανοιχτού κώδικα πλατφόρμα που χρησιμοποιείται για εκπαίδευση μηχανής. Ουσιαστικά χρησιμοποιείται για να εκπαιδεύσεις τον αλγόριθμό σου να αναζητεί και να ομαδοποιεί αντικείμενα με βάση τα κριτήρια που έχεις θέσει. Χρησιμοποιεί γραφήματα ροής δεδομένων. Ο κάθε κόμβος στο γράφημα αντιπροσωπεύει μαθηματικές λειτουργίες. Ο τρόπος που είναι φτιαγμένη η αρχιτεκτονική επιτρέπει την διενέργεια υπολογισμών σε περισσότερους από έναν επεξεργαστές με αποτέλεσμα να είναι πολύ αποδοτική.

2.4.3. Επιλογή Βιβλιοθήκης

Για την ανάπτυξη του συστήματος face recognition επιλέχθηκε η βιβλιοθήκη face_recognition της python που χρησιμοποιεί την OpenCv. Επιλέχθηκε η συγκεκριμένη γιατί είναι αποκλειστικά φτιαγμένη για πρόσωπα που είναι και το κύριο αντικείμενο της εργασίας καθώς η ανάλυση γίνεται σε αυτά. Η χρήση βιβλιοθήκης που χρησιμοποιεί την TensorFlow πιθανώς να οδηγούσε σε πιο αποδοτικό σύστημα όμως χρειαζόταν μεγαλύτερη προσαρμογή στην ομαδοποίηση των αντικειμένων η οποία δεν είναι απαραίτητη για το σύστημα

3. Περιγραφή Συστήματος

3.1. Γενική Περιγραφή Συστήματος

Το σύστημα που αναπτύσσεται θα χρησιμοποιηθεί από τρεις διαφορετικές κατηγορίες χρηστών, τους γιατρούς όλων των ειδικοτήτων που εργάζονται εντός του οργανισμού, τους νοσηλευτές και την γραμματεία. Πρόβλεψη για μερική χρήση του συστήματος και συγκεκριμένα του προγράμματος αναγνώρισης προσώπων υπάρχει και για τους υπεύθυνους ασφαλείας, βήμα όμως που θα υλοποιηθεί σε μεταγενέστερη φάση.

Η αρχική εγγραφή ενός πελάτη – ηλικιωμένου θα γίνεται από την γραμματεία, η οποία συμπληρώνει τα βασικά στοιχεία του εκάστοτε ατόμου. Η γραμματεία πρέπει να έχει πρόσβαση σε δεδομένα ώστε να ελέγχει για κάθε ηλικιωμένο σημαντικές ενημερώσεις ώστε να επικοινωνεί με τους συγγενείς του καθώς και να διαχειρίζεται τα οικονομικά στοιχεία. Επίσης μέσω του προγράμματος μπορεί να βλέπει αν κάποιος έχει άδεια να εξέλθει του χώρου ή αν για λόγους υγείας δεν μπορεί.

Ο νοσηλευτής - φροντιστής μέσω της εφαρμογής πρέπει να μπορεί να συμπληρώνει στοιχεία για τους πελάτες με βάση την καθημερινή τους δραστηριότητα. Στοιχεία που είτε αφορούν τον τομέα της υγείας είτε αφορούν τον τομέα των δραστηριοτήτων που πραγματοποιεί ή θα ήθελε να πραγματοποιήσει ο κάθε ηλικιωμένος.

Ο γιατρός μπορεί να κρατάει όλα τα ιατρικά στοιχεία στη βάση δεδομένων, να παρακολουθεί τις εξετάσεις του κάθε ασθενή – διαμένοντα, να δημιουργεί ένα ιστορικό και να το ελέγχει όποτε κρίνει αναγκαίο ώστε να εξάγει τα σωστά συμπεράσματα.

Το σύστημα αναγνώρισης προσώπων βοηθάει τον κάθε ένα από τους παραπάνω να εκτελέσει πιο σύντομα την δουλειά του. Για παράδειγμα η γραμματεία θα μπορεί άμεσα να αντιλαμβάνεται αν κάποιος από τους ηλικιωμένους προσπαθεί να εξέλθει από τον χώρο ενώ του έχει απαγορευτεί η έξοδος για λόγους υγείας. Έτσι ειδοποιεί τον αρμόδιο ασφαλείας και τον φροντιστή ώστε να επιληφθούν της κατάστασης.

3.2. Ρόλοι

Πέντε διαφορετικοί ρόλοι υπάρχουν στην πλατφόρμα.

- Admin (Διαχειριστή)
- Doctor (Γιατρός)
- Nurse (Νοσηλευτής)
- Secretary (Γραμματέας)
- Security (Προσωπικό ασφαλείας)

Από τους παραπάνω ρόλους στο σύστημα δεν μπορεί να γίνει εγγραφή νέου διαχειριστή μέσω της πλατφόρμας παρά μόνο μέσω της βάσης. Ειδικότερα:

- Admin:
 - Μπορεί να κάνει εγγραφή άλλων υπαλλήλων οποιασδήποτε ειδικότητας εκτός από της δικιάς του.
 - Μπορεί να κάνει εγγραφή νέων πελατών.
 - Μπορεί να δημιουργεί νέες δραστηριότητες.
 - Μπορεί να θέτει κανόνες ασφαλείας για το σύστημα.
 - Μπορεί να βλέπει στατιστικά στοιχεία για τον κάθε πελάτη.
 - Μπορεί να σταματάει και να ξεκινάει το σύστημα αναγνώρισης προσώπου καθώς και το σύστημα ανάλυσης δεδομένων.
 - Μπορεί να αναθέτει στον κάθε πελάτη νέες δραστηριότητες.
 - Μπορεί να βλέπει ειδοποιήσεις ασφαλείας.
 - Μπορεί να βλέπει αναλυτικά τις δραστηριότητες του κάθε πελάτη.
 - Μπορεί να βλέπει αναλυτικά το σύνολο των διαθέσιμων δραστηριοτήτων.
- Doctor - Nurse
 - Μπορεί να δημιουργεί νέες δραστηριότητες.
 - Μπορεί να βλέπει στατιστικά στοιχεία για τον κάθε πελάτη.
 - Μπορεί να αναθέτει στον κάθε πελάτη νέες δραστηριότητες.
 - Μπορεί να βλέπει αναλυτικά τις δραστηριότητες του κάθε πελάτη.
 - Μπορεί να βλέπει αναλυτικά το σύνολο των διαθέσιμων δραστηριοτήτων.
- Secretary
 - Μπορεί να κάνει εγγραφή νέων πελατών.
 - Μπορεί να δημιουργεί νέες δραστηριότητες.
 - Μπορεί να αναθέτει στον κάθε πελάτη νέες δραστηριότητες.
 - Μπορεί να βλέπει αναλυτικά το σύνολο των διαθέσιμων δραστηριοτήτων.
- Security
 - Μπορεί να θέτει κανόνες ασφαλείας για το σύστημα.
 - Μπορεί να βλέπει στατιστικά στοιχεία για τον κάθε πελάτη.

- Μπορεί να σταματάει και να ξεκινάει το σύστημα αναγνώρισης προσώπου καθώς και το σύστημα ανάλυσης δεδομένων.
- Μπορεί να βλέπει ειδοποιήσεις ασφαλείας

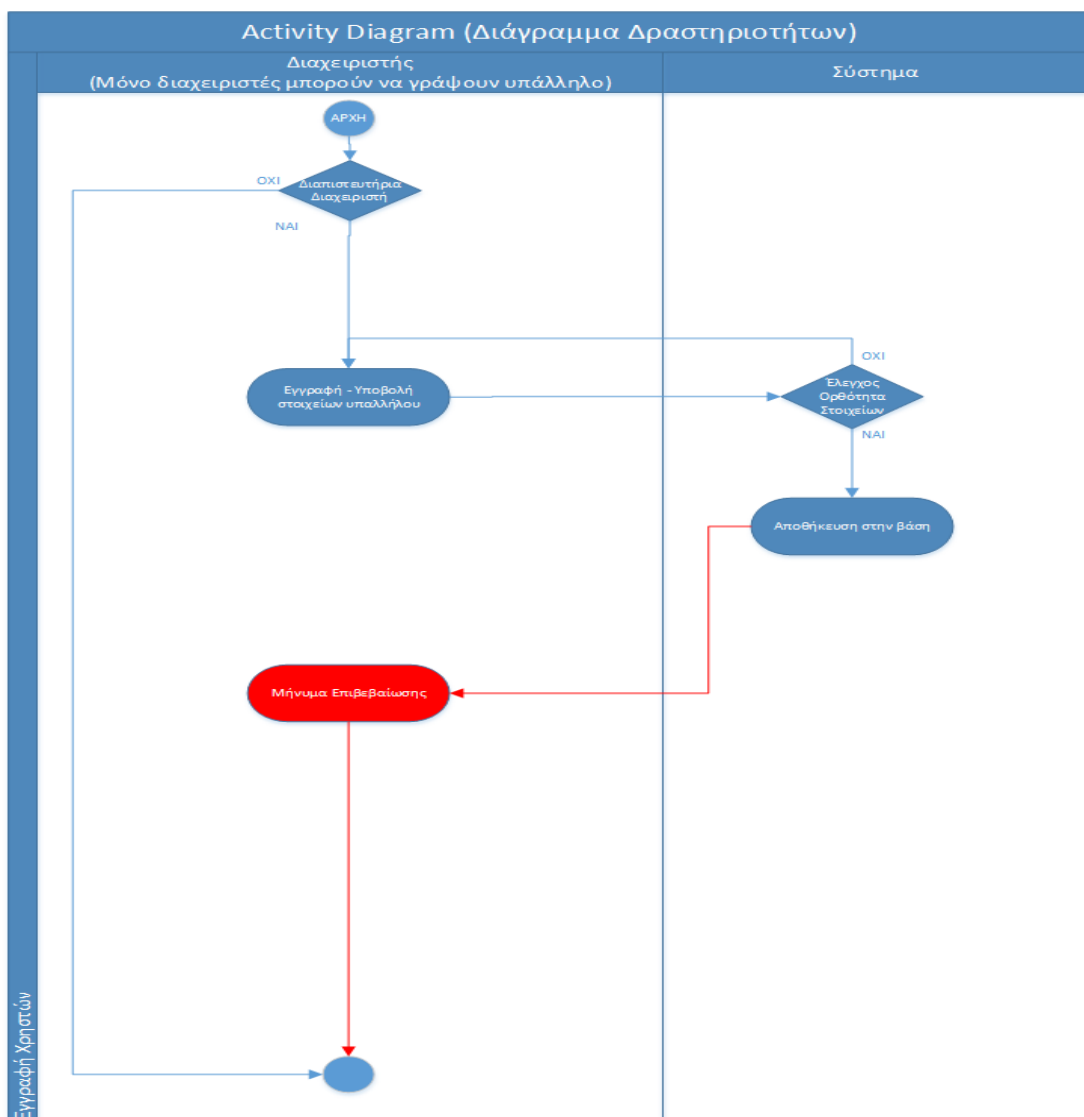
Στην ενότητα αυτή περιγράφονται οι απαιτήσεις του συστήματος και οι απαραίτητες λειτουργικές απαιτήσεις ώστε να είναι δομημένο σύμφωνα με τις προδιαγραφές της επιχείρησης.

3.3. Λειτουργικές Απαιτήσεις

Στο κεφάλαιο αυτό θα διευκρινιστούν οι λειτουργικές απαιτήσεις οι οποίες πρέπει να υλοποιεί το πληροφοριακό μας σύστημα.

3.3.1. Εγγραφή στο Σύστημα

Η εγγραφή στο σύστημα γίνεται μόνο από τους διαχειριστές του συστήματος, άτομα με εγκεκριμένους ρόλους.

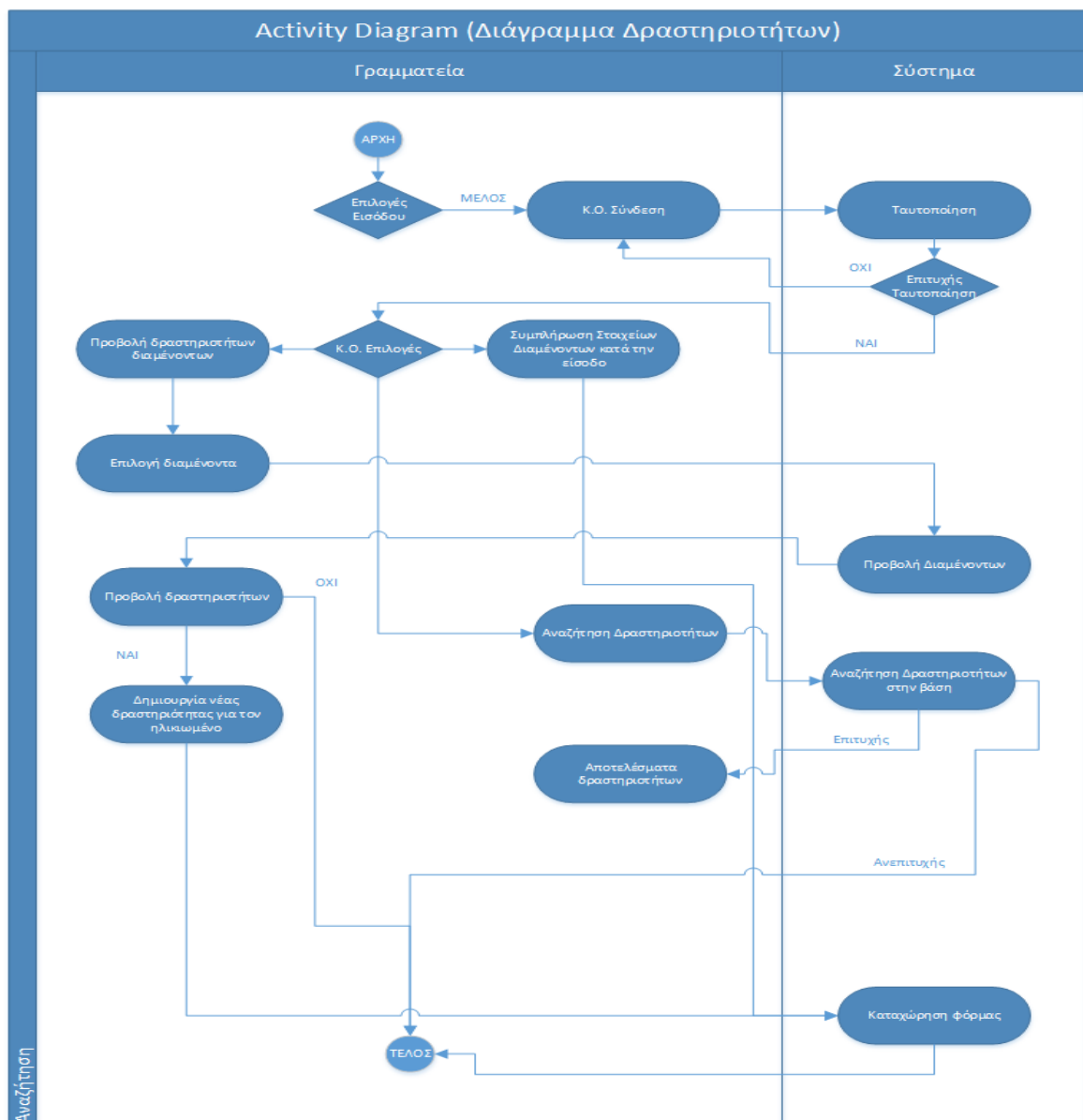


Εικόνα 5:Εγγραφή στο σύστημα

3.3.2. Είσοδος στο σύστημα με τον ρόλο της γραμματείας

Όπως έχει ήδη αναφερθεί πρόσβαση στο σύστημα έχουν τα άτομα που εργάζονται στην γραμματεία. Μερικές από τις βασικές διεργασίες που εκτελούν είναι:

- Συμπλήρωση στοιχείων ηλικιωμένου κατά την πρώτη είσοδο στον χώρο και συμπλήρωση στοιχείων επικοινωνίας με συγγενής.
- Αναζήτηση των διαμένωντων στο κέντρο ευγηρίας από την βάση δεδομένων ώστε να συμπληρωθούν επιπλέον στοιχεία ή να ανανεωθεί η καρτέλα του καθενός.
- Προβολή δραστηριοτήτων του κάθε εγγεγραμμένου ατόμου.
- Προβολή του συνόλου των δραστηριοτήτων που υπάρχουν.

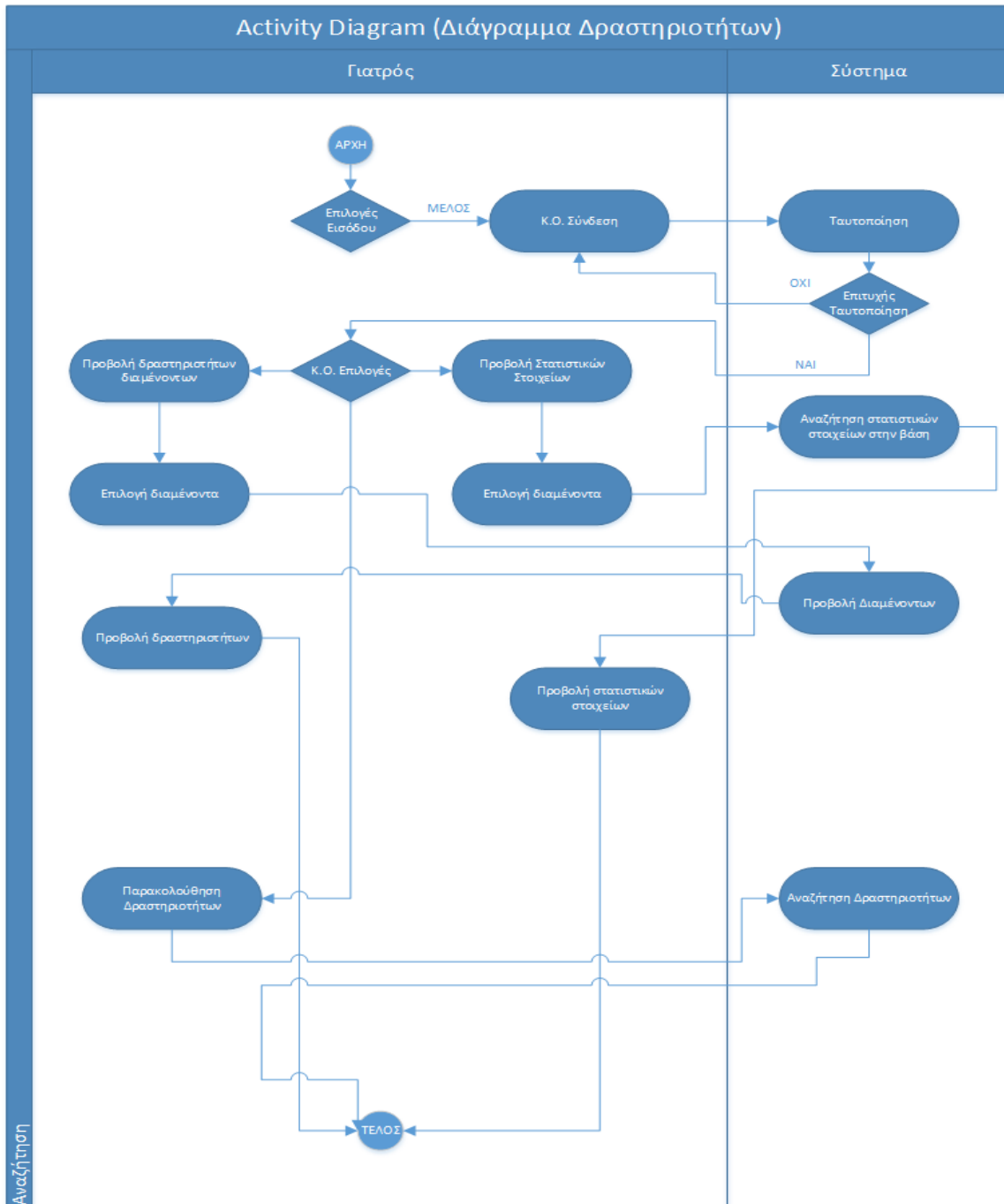


Εικόνα 6: Είσοδος και διεργασίες Γραμματείας

3.3.3. Είσοδος στο σύστημα με τον ρόλο του γιατρού

Στο κέντρο ευγηρίας όπως έχει αναφερθεί υπάρχουν συνέχεια γιατροί των βασικών ειδικοτήτων καθώς και ψυχολόγος. Εκτός από αυτούς υπάρχουν και γιατροί επισκέπτες που φροντίζουν την υγεία των ηλικιωμένων. Όλοι έχουν διαπιστευτήρια ώστε να συνδέονται στο σύστημα. Βασικές διεργασίες που πραγματοποιούν:

- Συμπλήρωση φόρμας με στοιχεία εξέτασης ασθενή και παραπεμπτικό εξετάσεων.
- Προβολή ιστορικού Ασθενή.
- Παρακολούθηση ιστορικού δραστηριοτήτων και ιατρικού ιστορικού

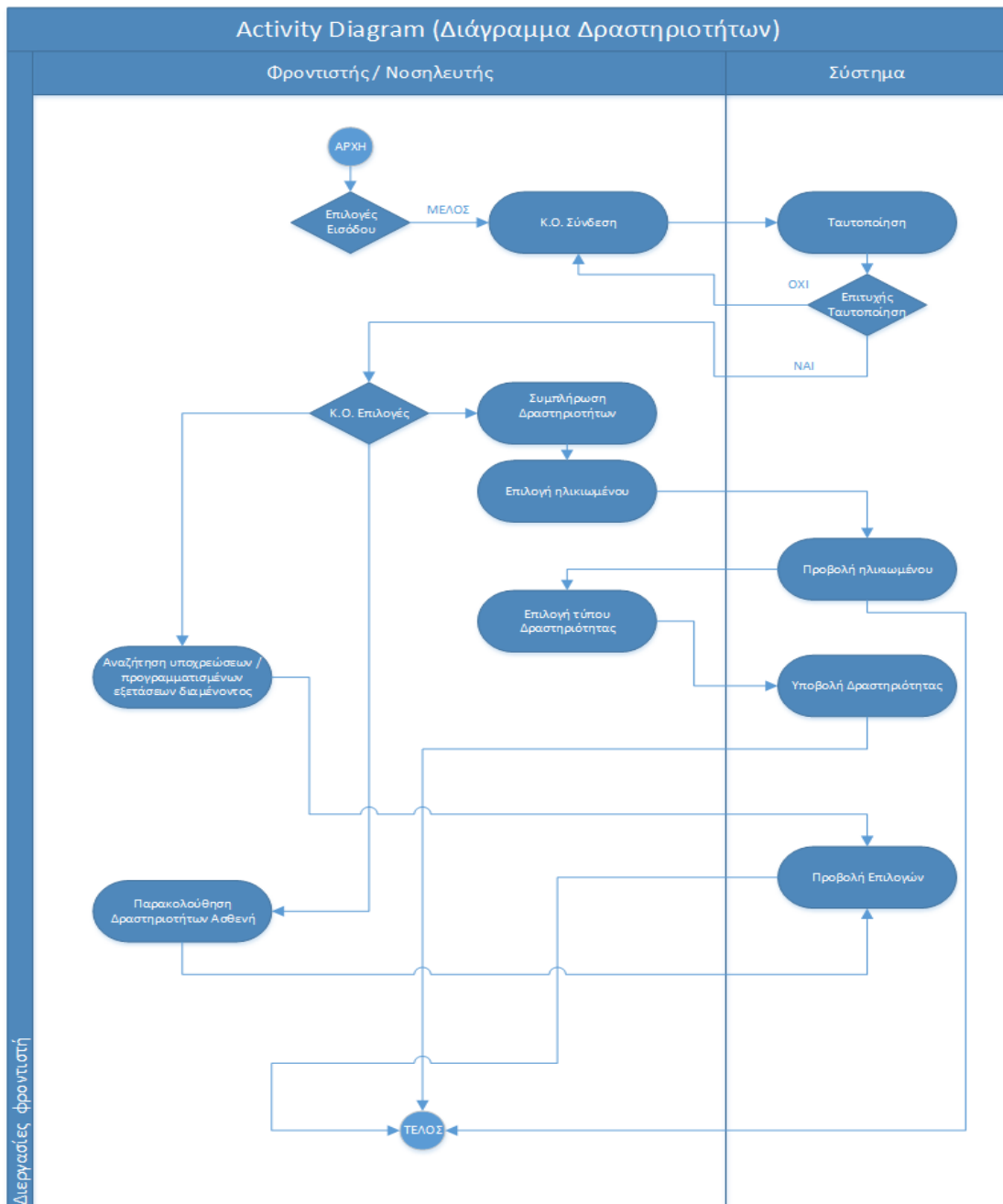


Εικόνα 7: Είσοδος και διεργασίες Ιατρών

3.3.4. Είσοδος στο σύστημα με τον ρόλο του φροντιστή - νοσηλεύτη

Οι άνθρωποι που έρχονται σε καθημερινή επαφή με τους ηλικιωμένους είναι οι φροντιστές / νοσηλεύτες. Έχουν πρόσβαση στο σύστημα και μπορούν να πραγματοποιούν τις παρακάτω διεργασίες:

- Παρακολούθηση δραστηριοτήτων και αγαπημένων ασχολιών του κάθε ηλικιωμένου.
- Αναζήτηση προγραμματισμένων εξετάσεων και δραστηριοτήτων του κάθε ηλικιωμένου όπως και των ραντεβού με γιατρό που έχει ο καθένας ώστε να φροντίζουν να γίνονται στην ώρα τους.
- Προσθήκη δραστηριοτήτων στο πρόγραμμα του ηλικιωμένου αν κρίνει ότι είναι απαραίτητο ή ζητηθούν.



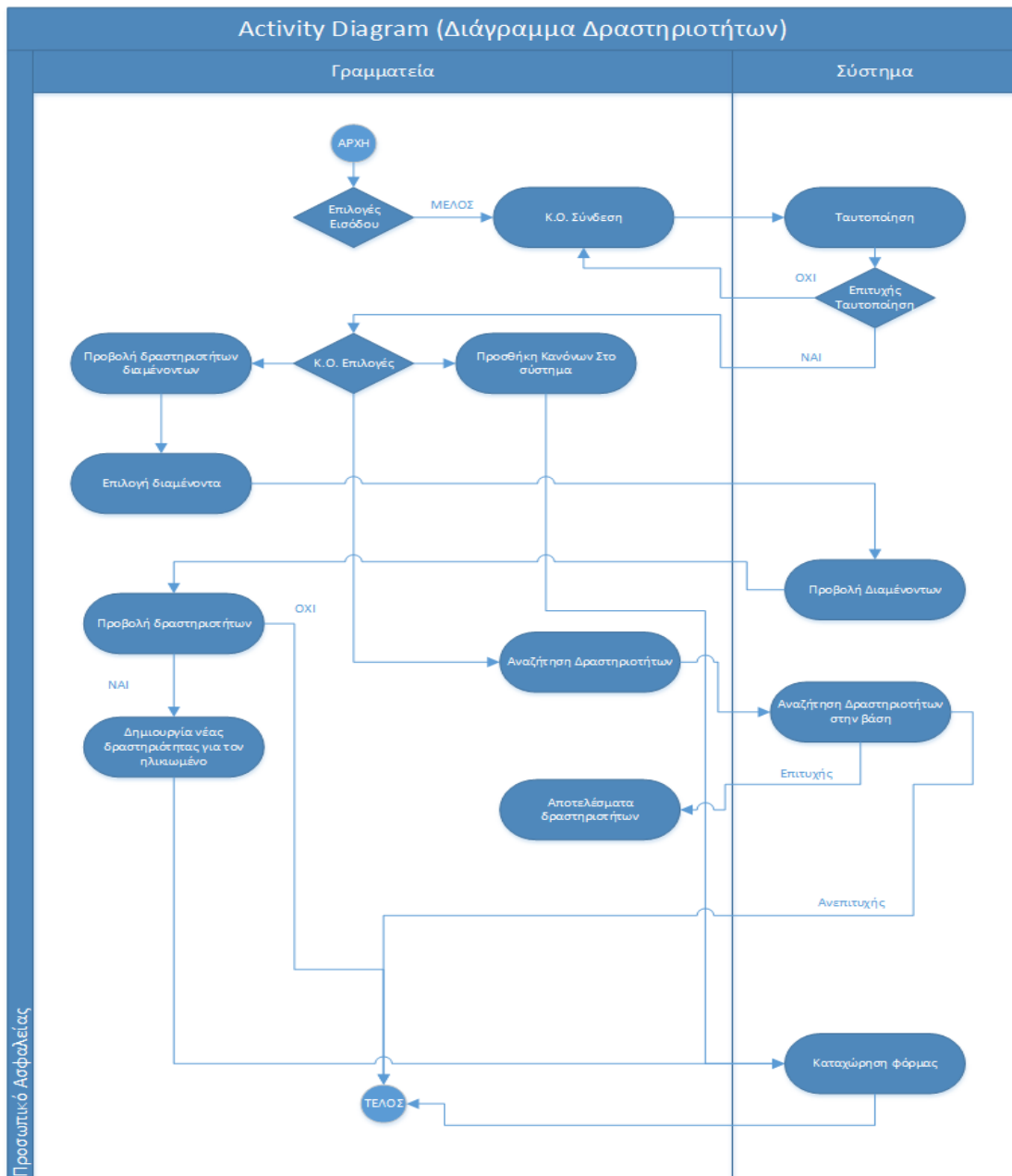
Εικόνα 8:Είσοδος και διεργασίες Φροντιστών

3.3.5. Είσοδος στο σύστημα με τον ρόλο του προσωπικού ασφαλείας

Το προσωπικό ασφαλείας έχει πρόσβαση στο σύστημα και μπορεί να πραγματοποιεί τις παρακάτω διεργασίες:

- Δημιουργία κανόνων ασφαλείας στο σύστημα
- Αναζήτηση των διαμένωντων στο κέντρο ευγηρίας από την βάση δεδομένων ώστε να συμπληρωθούν επιπλέον στοιχεία ή να ανανεωθεί η καρτέλα του καθενός.
- Προβολή δραστηριοτήτων του κάθε εγγεγραμμένου ατόμου.

- Προβολή του συνόλου των δραστηριοτήτων που υπάρχουν.



Εικόνα 9:Είσοδος και διεργασίες προσωπικού ασφαλείας

3.4. Απαιτήσεις Ασφαλείας – Αξιοπιστίας

Το σύστημα που δημιουργείται πρέπει να καλύπτει τις απαιτήσεις ασφαλείας ενός οργανισμού που χρησιμοποιεί προσωπικά δεδομένα. Ειδικά στην περίπτωση του οίκου ευγηρίας που ξεφεύγει από τα παραδοσιακά πλαίσια και περιλαμβάνει και ολοκληρωμένο σύστημα περίθαλψης η διασφάλιση των δεδομένων υγείας είναι από τα σημαντικότερα θέματα ασφαλείας.

Το πληροφοριακό σύστημα που αναπτύσσεται πραγματώνεται σε εσωτερικό δίκτυο του κέντρου ευγηρίας. Η πρόσβαση στο ίντερνετ δεν επιτρέπεται ώστε να

αποφευχθεί οποιαδήποτε περίπτωση εξωτερικής πρόσβασης ή διαρροής δεδομένων. Οι χρήστες είναι διαπιστευμένοι από την επιχείρηση έχοντας τα απαραίτητα συνθηματικά, τα οποία δίνονται από τους διαχειριστές του συστήματος. Το όνομα χρήστη καθώς και ο κωδικός ασφαλείας είναι στοιχεία που ξέρει ο κάθε χρήστης. Το συνθηματικό ασφαλείας είναι καταχωρημένο στην βάση δεδομένων σε κωδικοποιημένη μορφή.

Εκτός από την ασφάλεια από διαρροή δεδομένων, πρέπει να εξασφαλίζεται και η εγκυρότητα των δεδομένων που έχουν καταχωρηθεί στο σύστημα. Είναι θέμα αξιοπιστίας του συστήματος η βάση δεδομένων να είναι σωστά δομημένη. Έτσι αποφεύγονται λάθη όπως διπλές εγγραφές, απώλεια δεδομένων και εξαγωγή λανθασμένων συμπερασμάτων. Η δυνατότητα εγγραφής στοιχείων από πολλαπλούς χρήστες οδηγεί σε πιθανά ανθρώπινα λάθη κατά την καταχώρηση στοιχείων. Σκοπός του προγράμματος είναι να γίνει η διαδικασία όσο πιο αυτοματοποιημένη γίνεται με πολλαπλά σημεία ελέγχου ώστε να μειωθεί η παραπάνω πιθανότητα. Στην περίπτωση που γίνεται λάθος δυνατότητα διόρθωσης υπάρχει αλλά γίνεται μόνο ύστερα από συνεννόηση με τους διαχειριστές ώστε να αποφευχθεί η αλλοίωση δεδομένων από τυχαία ή από κακόβουλη προσπάθεια.

Τα δεδομένα υγείας πρέπει να είναι προσβάσιμα μόνο από τους γιατρούς εφόσον έχει δώσει άδεια ο ηλικιωμένος – ασθενής. Για αυτό το λόγο κατά την εγγραφή νέου ατόμου στο κέντρο ευγηρίας θα πρέπει να δίνεται το αντίστοιχο έντυπο στο οποίο αναφέρεται η δυνατότητα χρήσης των προσωπικών δεδομένων από συγκεκριμένους υπαλλήλους και γιατρούς καθώς και η δυνατότητα, οποιαδήποτε στιγμή, να μπορείς να κάνεις άρση του δικαιώματος αυτού.

Σημαντικό κομμάτι του συστήματος στο οποίο πρέπει να δοθεί ιδιαίτερη προσοχή είναι το λογισμικό αναγνώρισης προσώπου. Η χρήση του πρέπει να γίνεται με την σύμφωνη γνώμη των διαμενόντων και των υπαλλήλων καθώς χρησιμοποιείται για την καταγραφή και εξαγωγή συμπερασμάτων από τις κινήσεις και τις δραστηριότητες των ηλικιωμένων.

3.5. Συντήρηση Συστήματος

Η συντήρηση του προγράμματος γίνεται στα πλαίσια των ανανεώσεων της έκδοσης ανά τακτά χρονικά διαστήματα και εφόσον κρίνεται απαραίτητο.

Το σύστημα στο βαθμό που είναι δυνατό αποτελείται από αυτοτελή στοιχεία. Για παράδειγμα η γραμματεία, οι γιατροί και οι φροντιστές έχουν όλοι πρόσβαση στην καρτέλα του ασθενή όμως ο κάθε ένας έχει διαφορετική προβολή στοιχείων και διαφορετικές ενότητες με τις οποίες ασχολείται. Έτσι η τμηματική διόρθωση ή ανανέωση του προγράμματος είναι ευκολότερη και γίνεται σε σημαντικά μικρότερο χρονικό διάστημα. Πολλά στοιχεία αναγκαστικά όμως υπάρχουν σε παραπάνω από

ένας χρήστες. Τα στοιχεία αυτά θεωρούνται επαναχρησιμοποιούμενοι πόροι και σχεδιάζονται – αξιοποιούνται με τρόπο ώστε οποιαδήποτε αλλαγή αν χρειαστεί στο μέλλον να μην επηρεάζει το υπόλοιπο σύστημα.

Δοκιμές γίνονται συνέχεια ώστε να προλαμβάνονται καταστάσεις και προβλήματα που μπορεί να μην έχουν παρατηρηθεί. Πριν τοποθετηθεί μια νέα έκδοση υπάρχει αναλυτικός έλεγχος όλων των επιμέρους στοιχείων του προγράμματος ανεξάρτητα αν έχουν αλλάξει ή όχι.

3.6. Ευχρηστία

Η διεπαφή που σχεδιάζεται θα χρησιμοποιηθεί από άτομα διαφορετικών ειδικοτήτων, με εύρος χαρακτηριστικών και ικανοτήτων. Έτσι πρέπει να είναι προσιτή και η αισθητική της να επιτρέπει την άνετη και ευχάριστη αλληλεπίδραση με τον χρήστη. Το μοντέλο που ακολουθείται για τον σχεδιασμό είναι αρκετά λιτό καθώς πρόκειται για εργασιακό περιβάλλον. Λίγες καρτέλες και επιλογές ομαδοποιημένες έτσι ώστε ο χρήστης να μην κουράζεται. Με αυτό τον τρόπο αποφεύγονται λάθη που δύναται να δημιουργηθούν εξαιτίας μιας αρκετά σύνθετης διεπαφής χρήση – συστήματος. Η απλότητα βοηθάει και στην διευκόλυνση εκμάθησης, όχι μόνο από το προσωπικό που ήδη υπάρχει στην επιχείρηση αλλά και από τα άτομα που θα προσληφθούν στο μέλλον.

3.7. Τεχνολογίες που χρησιμοποιήθηκαν

- **Node.js:**

Η Node.js είναι μια πλατφόρμα ανάπτυξης λογισμικού (κυρίως διακομιστών) χτισμένη σε περιβάλλον Javascript. Στόχος του Node.js είναι να παρέχει ένα εύκολο τρόπο δημιουργίας κλιμακωτών διαδικτυακών εφαρμογών.

Βασικό χαρακτηριστικό της Node.js είναι η ασύγχρονη επικοινωνία μεταξύ υπολογιστικών πόρων. Αυτό επιτυγχάνεται με την χρήση συμβάντων (event) που προσφέρει η Javascript και ονομάζονται callbacks. Οποιοδήποτε συμβάν πυροδοτεί την λειτουργία της node. Σε περίπτωση που δεν υπάρχουν συμβάντα παραμένει αδρανής ώστε να καταναλώνει λιγότερους πόρους. Πολύ σημαντική επίσης είναι η ύπαρξη μεγάλης κοινότητας υποστήριξης με πληθώρα πακέτων και βιβλιοθηκών.

- **React.js / JSX:**

Η React είναι μια βιβλιοθήκη της Javascript που χρησιμοποιείται για την κατασκευή συστημάτων διεπαφής χρήστη. Στηρίζεται στην χρήση επιμέρους κομματιών γνωστά ως Components τα οποία μπορούν με ευκολία να επαναχρησιμοποιηθούν. Τα Components μπορεί να είναι αρκετά απλά όπως ένα κουμπί που το έχουμε μορφοποιήσει όπως θέλουμε εμείς μέχρι αρκετά σύνθετα και μεγάλα όπως ολόκληρη σελίδα. Χρησιμοποιεί μια δικιά της γλώσσα γνωστή ως JSX η οποία μοιάζει

αρκετά με HTML. Ο κώδικας HTML, δεδομένου ότι η αναπτυσσόμενη εφαρμογή εκτελείται σε κάποιο φυλλομετρητή, παράγεται αυτόματα από το προγραμματιστικό πλαίσιο. Συγκεκριμένα ο κώδικας HTML δημιουργείται αυτόματα από τα Components μέσα από μεθόδους που καλούνται render().

- **PYTHON:**

Η Python είναι διερμηνευόμενη (interpreted), γενικού σκοπού (general-purpose) και υψηλού επιπέδου, γλώσσα προγραμματισμού. Ανήκει στις γλώσσες προστακτικού προγραμματισμού (Imperative programming) και υποστηρίζει τόσο το διαδικαστικό (procedural programming) όσο και το αντικειμενοστραφές (object - oriented programming) προγραμματιστικό υπόδειγμα. Είναι δυναμική γλώσσα προγραμματισμού (dynamic typing).

Η python υποστηρίζει την δημιουργία πακέτων (packages) το οποία ενθαρρύνουν ιδιαίτερα την επαναχρησιμοποίηση του κώδικα. Αποτελείται από μια μεγάλη και συνεχώς αναπτυσσόμενη κοινότητα που παράγει συνέχεια καινούρια πακέτα και ανανεώνει τα ήδη υπάρχοντα.

- **MongoDb:**

Πρόκειται για μια NoSQL βάση δεδομένων. Στηρίζεται στην χρήση εγγράφων (Document Based) και πρόκειται για μια κατανεμημένη βάση δεδομένων. Χαρακτηρίζεται από την ευκολία στην επεκτασιμότητα κι στην «άναρχη» δημιουργία δεδομένων. Δεν είναι απαραίτητη η δημιουργία κάποιο πίνακα από την αρχή. Μπορεί να δημιουργηθεί μια εγγραφή στην βάση κατευθείαν μέσω του κώδικα. Είναι από τις μεγαλύτερες και πιο δημοφιλείς βάσεις δεδομένων NoSQL.

- **REST API:**

Το REST αποτελεί ένα σύνολο από αρχές σχεδίασης μιας δικτυακής υπηρεσίας που επικεντρώνει στους πόρους (π.χ. δεδομένα) ενός συστήματος. Η μεταβολή της κατάστασης (ενέργεια επί) των πόρων του συστήματος περιγράφεται και μεταφέρεται στο σύστημα μέσω του πρωτοκόλλου HTTP από διάφορους clients (ανεξαρτήτως της γλώσσας στην οποία έχουν υλοποιηθεί).

Η βασική αρχή σχεδίασης του REST είναι η ένα-προς-ένα αντιστοίχιση μεταξύ λειτουργιών CRUD (create, read, update, delete) και HTTP μεθόδων. Σύμφωνα με αυτή την αντιστοίχιση:

- Για τη δημιουργία ενός πόρου στον server, χρησιμοποιούμε την μέθοδο POST.
- Για την ανάσυρση ενός πόρου, χρησιμοποιούμε την GET.

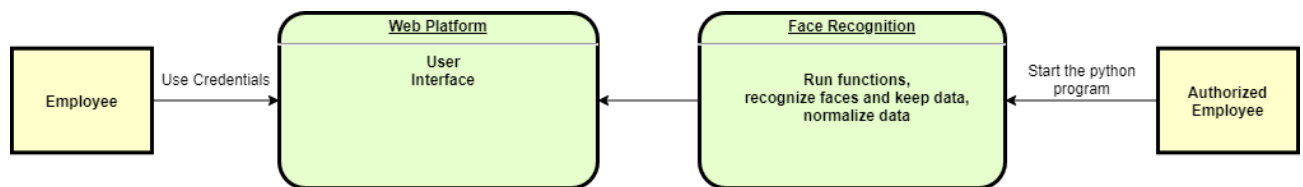
- Για την αλλαγή της κατάστασης ενός πόρου ή την ενημέρωσή του, χρησιμοποιούμε την PUT.
- Για την απομάκρυνση ή διαγραφή ενός πόρου, χρησιμοποιούμε την DELETE.

Με βάση το REST το URI δεν χρησιμοποιείται πια για την περιγραφή της ενέργειας που θέλουμε να εκτελέσουμε αλλά μόνο τον εντοπισμό του πόρου επί του οποίου θα ασκηθεί η ενέργεια, και τα δεδομένα δεν μεταφέρονται ως παράμετροι στο URI ενός GET αιτήματος αλλά ως XML ή JSON-formatted δεδομένα στο περιεχόμενο μιας POST ή PUT μεθόδου. Με άλλα λόγια σε μια υπηρεσία REST, ένα URI εκφράζει ένα αντικείμενο στο οποίο παρέχει πρόσβαση η υπηρεσία μέσω ενός HTTP αιτήματος. Το είδος του αιτήματος καθορίζει την ενέργεια που θέλουμε να εφαρμόσουμε στο αντικείμενο αυτό και το περιεχόμενο του αιτήματος περιέχει διάφορες εξειδικεύσεις τις ενέργειας.

3.8. Διαγράμματα ροής Διεργασιών

Το διάγραμμα ροής δεδομένων είναι μια γραφική αναπαράσταση της ροής δεδομένων ενός πληροφοριακού συστήματος μέσω των διεργασιών του. Αποτελεί συνήθως ένα προκαταρκτικό βήμα για την δημιουργία μιας συνολικής εικόνας του συστήματος χωρίς να υπεισερχόμαστε σε μεγάλη λεπτομέρεια.

Για το παρών σύστημα συντάχθηκε το παρακάτω γενικό διάγραμμα:



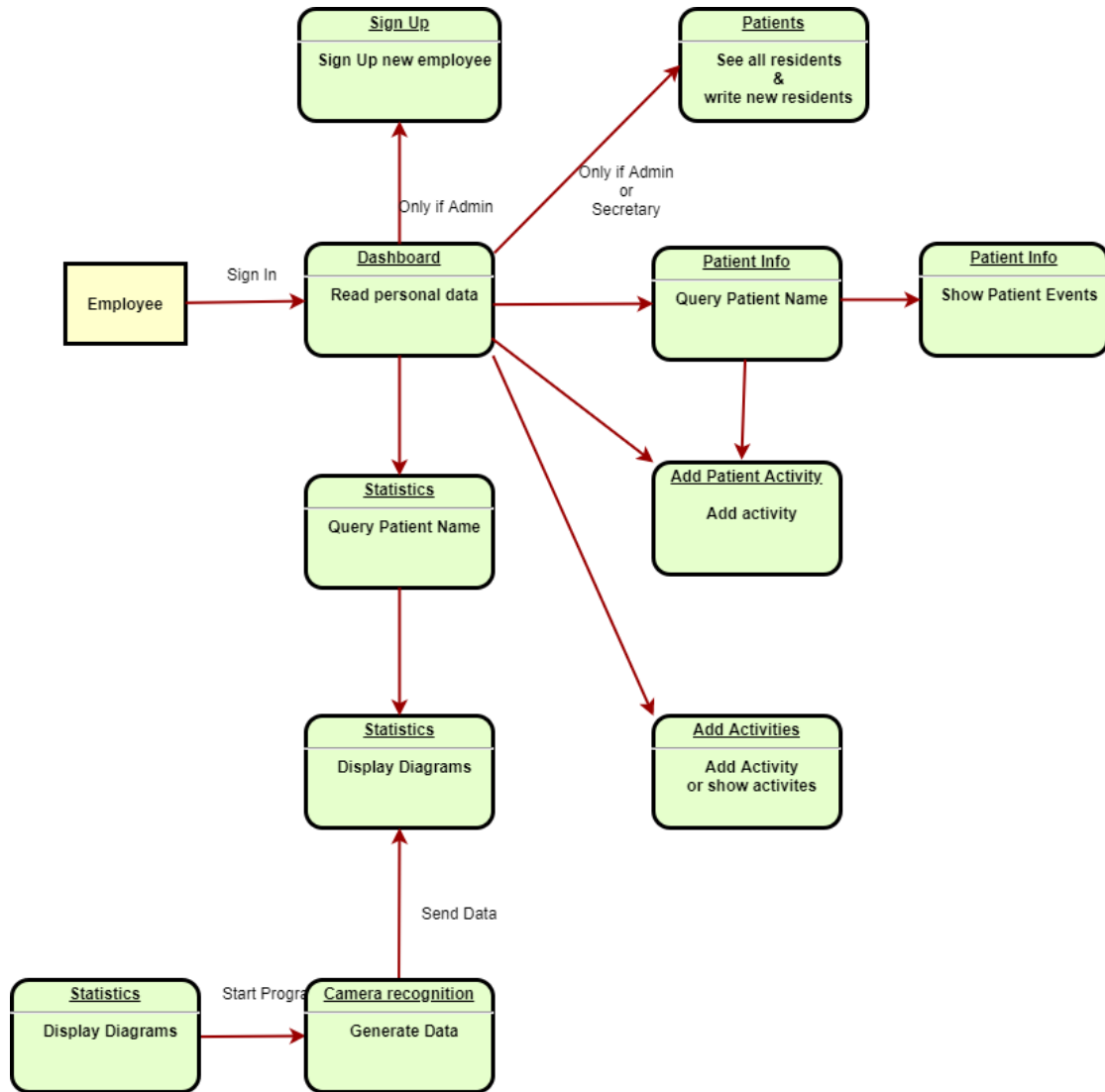
Εικόνα 10: Διάγραμμα Ροής Διεργασιών 1

Στο παραπάνω διάγραμμα φαίνεται μια γενική εικόνα του συστήματος. Ουσιαστικά το σύστημα αποτελείται από δύο επιμέρους συστήματα. Η σύνδεση γίνεται σε επίπεδο βάσης και υπάρχει ροή δεδομένων από το ένα (face recognition) στο άλλο (web platform).

Η ιδιαιτερότητα των δύο συστημάτων αντί για ένα βοηθάει καθώς το σύστημα αναγνώρισης προσώπου επιθυμούμε να είναι ανεξάρτητο από την πλατφόρμα που χρησιμοποιεί ο χρήστης. Έτσι σε περίπτωση αλλαγής της πλατφόρμας να συνεχίζει η απροβλημάτιστη λειτουργία του. Η σύνδεση γίνεται μόνο σε επίπεδο βάσεων άρα θα

πρέπει απλά οποιαδήποτε νέα πλατφόρμα δημιουργηθεί να συνδέεται μόνο με την βάση.

Επίσης συντάχθηκε ένα διάγραμμα λειτουργιών που περιγράφει τις βασικές λειτουργίες της πλατφόρμας χωρίς όμως να αναφέρεται σε έκταση στις συνδέσεις με την βάση και τους τρόπους μεταφοράς πληροφορίας:



Εικόνα 11: Διάγραμμα Ροής Διεργασιών 2

Όλες οι παραπάνω διαδικασίες θα υλοποιηθούν στο σύστημα. Ο τρόπος υλοποίησης δεν αναφέρεται απλά είναι ο οδηγός για την υλοποίηση.

3.9. Ασφάλεια Δεδομένων.

Το σύστημα για να λειτουργήσει σωστά πρέπει πρώτα από όλα να εξασφαλίζει την ακεραιότητα των δεδομένων του. Για αυτό τον λόγο έχουν ληφθεί μια σειρά από μέτρα:

- Είσοδος στο σύστημα μόνο από άτομα της επιχείρησης ο καθένας με το προσωπικό του κωδικό ο οποίος αποθηκεύεται κρυπτογραφημένα στην βάση δεδομένων.
- Επικοινωνία με την βάση μέσω server ο οποίος είναι κλεισμένος σε χώρο με πρόσβαση από εγκεκριμένα άτομα μόνο.
- Λειτουργία καταγραφής στο σημείο που είναι ο server με την βάση δεδομένων ώστε να υπάρχουν τα ανάλογα Notifications αν κάποιος προσπαθήσει να μπει.

Τα δεδομένα τόσο των ασθενών/ηλικιωμένων όσο και τα δεδομένα καταγραφής θέσεων αποτελούν δεδομένα τα οποία είναι προσωπικού χαρακτήρα και πρέπει σε οποιαδήποτε περίπτωση να επαρκούν οι κανόνες ώστε να είναι προστατευμένα. Στόχος μελλοντικών εκδόσεων είναι η προσθήκη και νέων συστημάτων ασφαλείας.

3.10. Μελέτη εφικτότητας

3.10.1. Τεχνική Εφικτότητα

Στο στάδιο αυτό γίνεται μελέτη των απαιτήσεων και της δυνατότητας να πραγματοποιηθούν σε χρονικό διάστημα που συμφέρει την επιχείρηση με βάση τις τεχνολογίες που υπάρχουν.

Η σύνδεση όλων των στοιχείων που μαζεύονται από τους γιατρούς, τους νοσηλευτές και την γραμματεία είναι ένα σύστημα που είναι εύκολο να πραγματοποιηθεί καθώς απαιτείται μια γλώσσα προγραμματισμού και μια σύνδεση σε βάση δεδομένων, χωρίς όμως να είναι απαραίτητη η δημιουργία κάποιου ιδιαίτερα δύσκολου αλγόριθμου. Το πιο δύσκολο τμήμα της διεργασίας αυτής είναι η ανάγκη να είναι απόλυτα ασφαλές το πρόγραμμα εξαιτίας της ύπαρξης σημαντικών προσωπικών δεδομένων. Τα δεδομένα υγείας είναι στοιχεία προσωποποιημένα, στα οποία πρέπει να υπάρχει δυνατότητα πρόσβασης μόνο από συγκεκριμένο αριθμό ανθρώπων, τους γιατρούς του εκάστοτε πελάτη. Έτσι οφείλει το πρόγραμμα να τα διασφαλίζει τόσο από εξωτερικούς παράγοντες όσο και από μέλη του κέντρου ευγηρίας που δεν έχουν πρόσβαση.

Το στοιχείο του προγράμματος που είναι δυσκολότερο να πραγματοποιηθεί είναι η εφαρμογή αναγνώρισης προσώπων. Τα τελευταία χρόνια όμως έχουν αναπτυχθεί αρκετοί αλγόριθμοι και έχουν δημιουργηθεί αρκετά προγράμματα τα οποία πραγματοποιούν την συγκεκριμένη διεργασία. Συνεπώς παρά την δυσκολία που μπορεί να υπάρχει στην ανάπτυξη του συγκεκριμένου τμήματος, υπάρχει ταυτόχρονα και μεγάλο πλήθος εφαρμογών που έχουν ήδη δημιουργηθεί ώστε να υπάρξει η απαραίτητη βοήθεια εφόσον καταστεί αναγκαίο.

3.10.2. Οικονομική Εφικτότητα

Η ανάπτυξη του συστήματος εξαρτάται σε μεγάλο βαθμό από το κόστος που απαιτείται για την υλοποίηση του. Παρακάτω παρατίθεται ένας ενδεικτικός πίνακας με το κόστος των φάσεων υλοποίησης, της εγκατάστασης στην επιχείρηση, της συντήρησης καθώς και το κόστος πιθανών εκπαιδεύσεων:

	Εργασία	Κόστος
Ανάπτυξη Συστήματος	Ανάλυση Απαιτήσεων	200 €
	Λογικός Σχεδιασμός	300 €
	Φυσικός Σχεδιασμός	6.000 €
	Σύνολο	6.500 €
Εγκατάσταση Συστήματος	Μετάπτωση Συστήματος	300 €
	Διενέργεια Δοκιμών	200 €
	Εκπαίδευση Χρηστών	300 €
	Σύνολο	800 €
Συντήρηση Συστήματος	Επανεκπαίδευση Προσωπικού(ανά κύκλο)	300 €
	Υποστήριξη Χρηστών(ανά επίσκεψη)	50 €
	Αντιμετώπιση Προβλημάτων(ανά περίπτωση)	150 €
	Σύνολο	500 €
Συνολικό Κόστος Ανάπτυξης και Εγκατάστασης		7.300 €

Πίνακας 1: Κόστος Συστήματος

Στον παραπάνω πίνακα αναφέρονται τα κόστη για τις φάσεις της ανάπτυξης και της εγκατάστασης του συστήματος. Δεν περιλαμβάνονται τα κόστη για αγορά τεχνικού εξοπλισμού (π.χ. υπολογιστές, κάμερες) που πιθανώς να χρειαστούν και δεν διαθέτει ήδη η επιχείρηση.

Η συντήρηση του συστήματος και συγκεκριμένα η αντιμετώπιση προβλημάτων και υποστήριξη χρηστών κοστολογείται ανάλογα με το πρόβλημα με μέγιστο ποσό το αναγραφόμενο στον παραπάνω πίνακα.

Η επανεκπαίδευση προσωπικού είτε πρόκειται για ομαδική, είτε ατομική (πρόσληψη προσωπικού) γίνεται μέσω ηλεκτρονικών μαθημάτων.

3.10.3. Λειτουργική Εφικτότητα

Η ορθή λειτουργία του συστήματος εφόσον πλέον δημιουργηθεί εξαρτάται στην δυνατότητα των υπαλλήλων να ανταποκριθούν στις απαιτήσεις του. Τα άτομα που θα το χρησιμοποιήσουν θα επωφεληθούν από αυτό. Θεωρείται λοιπόν ότι θα υπάρχει η απαραίτητη όρεξη ως προς την εκμάθησή του. Επίσης η απλότητα στην

χρήση του είναι βασικό ζητούμενο κατά τον σχεδιασμό ώστε να μην αντιμετωπίσουν κάποιο σημαντικό πρόβλημα κατά την εκμάθηση. Τα προγράμματα συνεχής εκπαίδευσης εξασφαλίζουν την επίλυση οποιουδήποτε προβλήματος ή απορίας δημιουργείται καθώς καλύπτουν και την εκπαίδευση νέου προσωπικού.

4. Φυσικός Σχεδιασμός – Αρχιτεκτονική Συστήματος

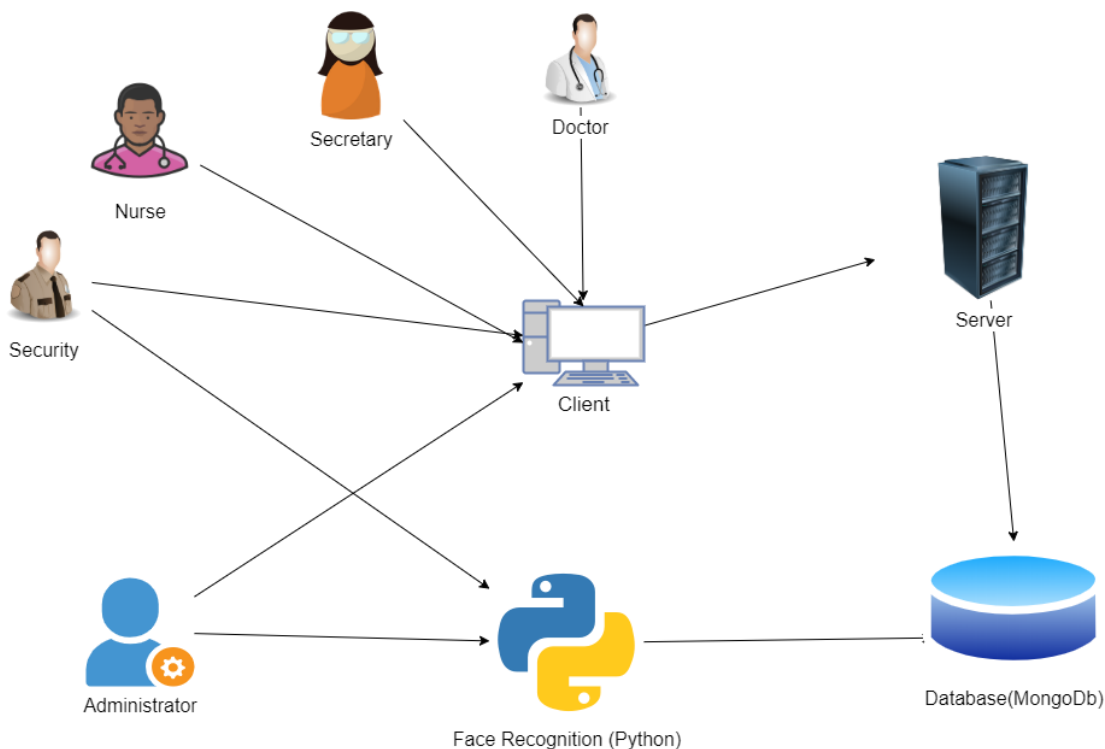
Στην ενότητα αυτή αναλύεται ο τρόπος με τον οποίο είναι δομημένο το σύστημα και ο τρόπος επικοινωνίας των υποσυστημάτων μεταξύ τους.

4.1. Σύστημα

Το σύστημα που δημιουργήθηκε αποτελείται από δύο τμήματα:

- Το πρώτο, το οποίο είναι το κομμάτι στο οποίο έρχεται σε επαφή ο χρήστης είναι μια πλατφόρμα που βρίσκεται πάντα ενεργή σε πρόγραμμα περιήγησης που χρησιμοποιεί ο εκάστοτε υπολογιστής. Μέσω του τμήματος αυτού διενεργούνται όλες οι απαραίτητες ενέργειες για την λειτουργία της επιχείρησης καθώς και θέτονται σε λειτουργία μερικοί κανόνες ασφαλείας.
- Το δεύτερο τμήμα στο οποίο δεν έχει πρόσβαση ο χρήστης παρά μόνο ο administrator είναι υπεύθυνο για την λειτουργία των καμερών και την ανάλυση, κανονικοποίηση και επεξεργασία των δεδομένων. Δεν πρόκειται για τμήμα το οποίο έχει κάποιο σύστημα διεπαφής χρήστη(User Interface) αλλά τρέχει παράλληλα με το πρώτο τμήμα και εκτελεί ανά τακτά χρονικά διαστήματα ελέγχους. Η μόνη δυνατότητα παρέμβασης είναι η έναρξη και η διακοπή της λειτουργίας του.

Η δομή του συστήματος περιγράφεται στο παρακάτω διάγραμμα:

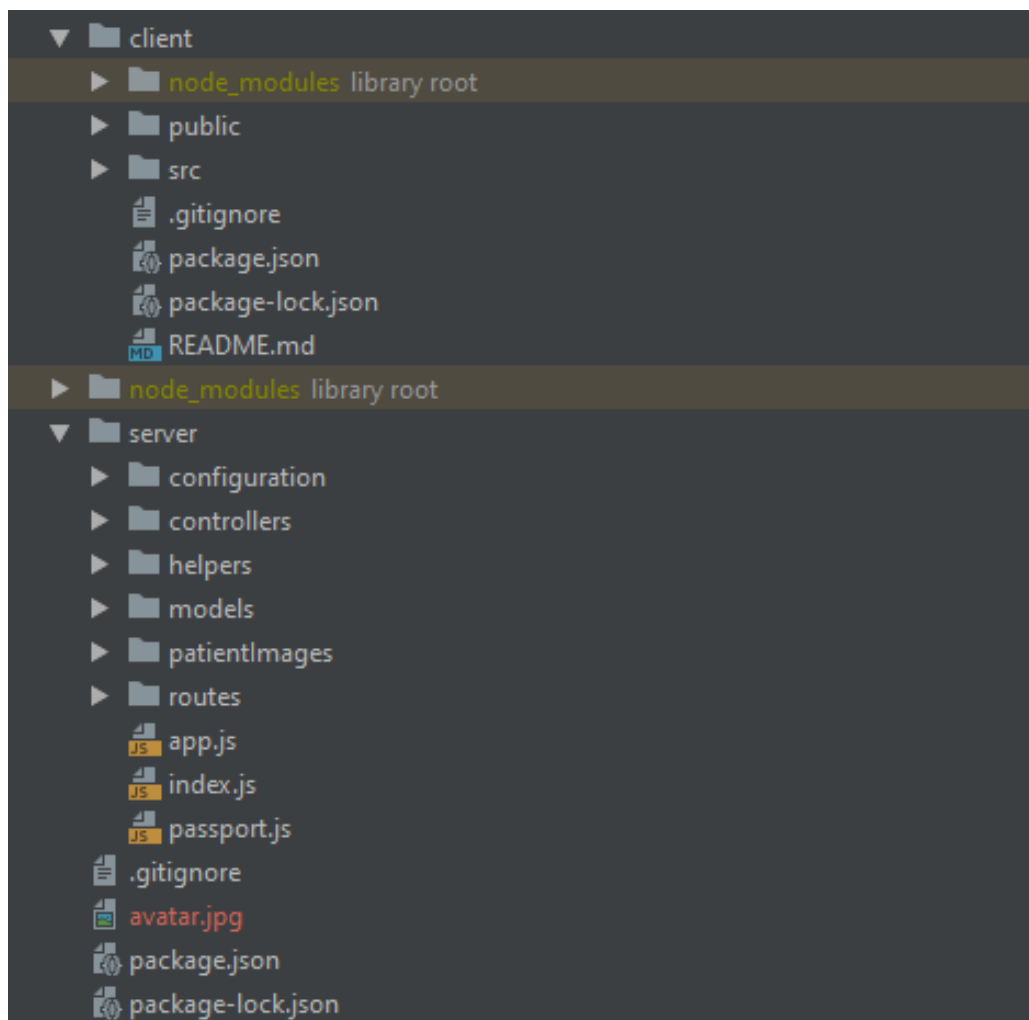


Εικόνα 12: Δομή Συστήματος

Το σύστημα δηλαδή αποτελείται από τέσσερα ξεχωριστά τμήματα. Το κάθε τμήμα λειτουργεί ανεξάρτητα από τα υπόλοιπα για να παρέχουν όμως αποτέλεσμα πρέπει να τρέχουν όλα ταυτόχρονα.

- Web App:

Η εφαρμογή χωρίζεται σε δύο υποσυστήματα που τρέχουν σε διαφορετικές διευθύνσεις. Το υποσύστημα server που τρέχει στην διεύθυνση <http://localhost:5000/> και το υποσύστημα client που τρέχει στην διεύθυνση <http://localhost:3000/>.



Εικόνα 13: Δομή Web Application

Το υποσύστημα Server λειτουργεί στην επιχείρηση πάνω σε δικό του σύστημα. Οι εργαζόμενοι δεν έχουν πρόσβαση. Η ομάδα που δημιούργησε το σύστημα είναι υπεύθυνη για την σωστή λειτουργία του τμήματος server και την συντήρησή του. Στον server δημιουργούνται όλες οι συνδέσεις με την βάση δεδομένων και λειτουργεί ως ενδιάμεσος σταθμός ανάμεσα στο υποσύστημα client και την βάση. Ενώ θα μπορούσε να δημιουργηθεί απευθείας ένωση των δύο παραπάνω προτιμήθηκε ο server καθώς τρέχει συνέχεια και σε περίπτωση που δημιουργηθεί ανάγκη δημιουργίας δεύτερης πλατφόρμας να επαναχρησιμοποιηθούν τα Rest API.

Το υποσύστημα Client δημιουργήθηκε για να δημιουργεί το γραφικό περιβάλλον που θα χρησιμοποιούν οι εργαζόμενοι. Η βασική του εργασία είναι να στέλνει τα αιτήματα στον server και να παρουσιάζει τα δεδομένα στον χρήστη. Πάνω στον client έχει στηθεί όλο το γραφικό περιβάλλον της εφαρμογής.

- Database (MongoDB)

Η βάση που αναπτύχθηκε είναι μια mongo Βάση δεδομένων. Ο βασικός λόγος επιλογής της βάσης αυτής είναι η ευκολία που παρουσιάζει στην δημιουργία νέων δεδομένων και στην ανταλλαγή δεδομένων μέσω json. Είναι στημένη παράλληλα με τον server. Ανταλλάσσει συνεχώς δεδομένα με τα προγράμματα python που έχουν φτιαχτεί. Ουσιαστικά μέσω της βάσης γίνεται σύνδεση του προγράμματος διεπαφής χρήστη με το πρόγραμμα αναγνώρισης προσώπου.

- Face Recognition (Python)

Το λογισμικό αναγνώρισης προσώπου έχει δημιουργηθεί ώστε να λειτουργεί μόνο με τη βάση. Δεν έχει ανάγκη από το υπόλοιπο σύστημα. Τα δεδομένα που ζητάει, επεξεργάζεται και επιστρέφει στην βάση έχουν κυρίως την μορφή χρονικών συμβάντων σε συνδυασμό με τοποθεσία. Ουσιαστικά καταγράφει τα άτομα που ήδη έχουν εγγραφτεί στο σύστημα και την τοποθεσία που έγινε η καταγραφή σε συνδυασμό με την ώρα καταγραφής.

4.2. Επικοινωνία Συστημάτων

Η επικοινωνία μεταξύ των υποσυστημάτων γίνεται σε επίπεδο βάσης. Ο υπάλληλος της επιχείρησης που έχει την εξουσιοδότηση(γραμματέας) κατά την εγγραφή των ενοίκων στο σύστημα φροντίζει ώστε να λαμβάνεται η αντίστοιχη φωτογραφία του κάθε ατόμου. Η φωτογραφία αποθηκεύεται στο έγγραφο του κάθε πελάτη στην αντίστοιχη συλλογή της Mongo σε μορφή binary. Το πρόγραμμα του face-recognition λαμβάνει από την βάση τις συλλογές με τις εικόνες και τις μετατρέπει εκ νέου σε png αρχεία και τα αποθηκεύει στον φάκελο «known_faces». Πάνω σε αυτά τρέχει και κωδικοποιεί τα πρόσωπα των εικόνων σε μια σειρά από σημεία.

Κατά την εκτέλεση του προγράμματος αναγνώρισης συγκρίνει τα αποτελέσματα της ζωντανής ροής του video με τις κωδικοποιήσεις των εικόνων από τον φάκελο «known_faces». Σε περίπτωση που έχει σε αρκετά μεγάλο ποσοστό ταύτιση των σημείων τότε αποθηκεύει στην βάση έγγραφα με στοιχεία όπως το όνομα του ατόμου, η ώρα που εμφανίστηκε και η τοποθεσία.

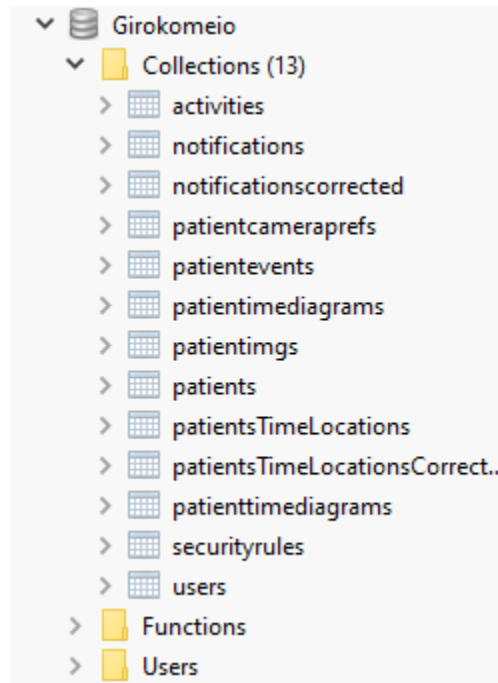
Η ανάλυση των δεδομένων γίνεται με βάση τα παραπάνω έγγραφα της βάσης. Τα αποτελέσματα της ανάλυσης αποθηκεύονται εκ νέου στην Mongo. Η εφαρμογή φυλλομετρητή λαμβάνει όλα τα απαραίτητα νούμερα για τα διαγράμματα απευθείας από τις συλλογές της βάσης, χωρίς να έχει άμεση συσχέτιση με την εφαρμογή της python.

5. Βάση MongoDB

Στην ενότητα αυτή αναφέρεται η βασική δομή της βάσης. Έτσι σε μελλοντική ανάγκη σύνδεση με κάποιο άλλο πρόγραμμα θα μπορεί να γίνει απλά κλήση στους αντίστοιχους πίνακες και αξιοποίησή τους.

5.1. Συλλογές Βάσης

Η βάση που αναπτύχθηκε αποτελείται από τις παρακάτω συλλογές που φαίνονται στην εικόνα:



Εικόνα 14: Πίνακες βάσης Mongo

- **activities:** Συλλογή που αποθηκεύει τις δραστηριότητες.
- **notifications:** Συλλογή που αποθηκεύει τις ειδοποιήσεις ασφαλείας.
- **notificationscorrected:** Συλλογή που αποθηκεύει τις κανονικοποιημένες ειδοποιήσεις ασφαλείας.
- **patientcameraprefs:** Συλλογή από τον οποίον παίρνουμε τα δεδομένα για τα διαγράμματα πίτας που χρησιμοποιούνται στην ανάλυση.
- **patientevents:** Συλλογή που αποθηκεύει όλα τα γεγονότα και τις προγραμματισμένες δραστηριότητες του κάθε ηλικιωμένου.
- **patientmediagrams:** Συλλογή από τον οποίον παίρνουμε τα δεδομένα για τα διαγράμματα χρόνου που χρησιμοποιούνται στην ανάλυση.
- **patients:** Συλλογή με όλους τους ενοίκους καταγεγραμμένους.

- **patientsTimeLocations**: Συλλογή με τα στοιχεία εντοπισμού από το πρόγραμμα του face recognition.
- **patientsTimeLocationsCorrected**: Κανονικοποίηση των στοιχείων από την συλλογή patientsTimeLocations
- **securityrules**: Συλλογή με όλους τους κανόνες ασφαλείας που θέτονται από την πλατφόρμα.
- **users**: Συλλογή με τους εργαζόμενους στην επιχείρηση

5.2. Σύνδεση με Συστήματα

Η βάση συνδέεται και με τα δύο επιμέρους συστήματα. Υπάρχουν συλλογές που συνδέονται μόνο με την πλατφόρμα φυλλομετρητή που χρησιμοποιούν οι χρήστες και συλλογές που επικοινωνούν μόνο με το λογισμικό αναγνώρισης προσώπου.

Με το λογισμικό face recognition συνδέονται:

- **securityrules**
- **patientsTimeLocations**
- **patientsTimeLocationsCorrected**
- **patientmediagrams**
- **notifications**
- **patientcameraprefs**
- **patients**
- **notificationscorrected**

Με την πλατφόρμα συνδέονται:

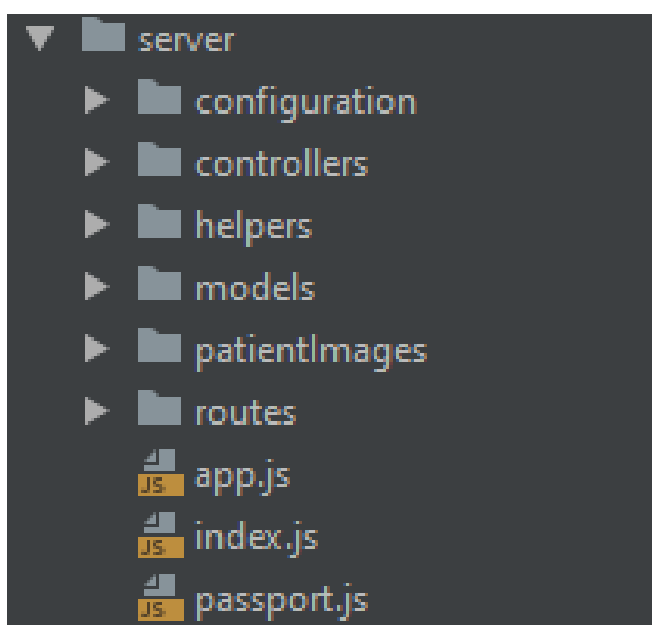
- **users**
- **patients**
- **securityrules**
- **patientmediagrams**
- **patientcameraprefs**
- **notificationscorrected**
- **activities**

6. Web Application

Στο κεφάλαιο αυτό αναλύονται τα βασικά χαρακτηριστικά ανάπτυξης της εφαρμογής φυλλομετρητή. Συγκεκριμένα θα παρουσιαστούν τα έτοιμα πακέτα που χρησιμοποιήθηκαν, τα web services που κατασκευάστηκαν και ο αναλυτικός τρόπος κατασκευής του κάθε υποσυστήματος.

6.1. Υποσύστημα Server

Το υποσύστημα του server λειτουργεί σαν ενδιάμεσος σύνδεσμος της βάσης. Ουσιαστικά περιέχονται όλες τις διαδικασίες πιστοποίησης χρηστών και όλες οι μεταφορές δεδομένων. Λειτουργεί σε ξεχωριστή διεύθυνση και οποιοδήποτε νέο πρόγραμμα δημιουργηθεί μπορεί να το χρησιμοποιήσει ώστε να λαμβάνει τα ίδια δεδομένα. Η δομή του έχει την παρακάτω μορφή:



Εικόνα 15: Δομή υποσυστήματος Server

Στον φάκελο routes υπάρχουν όλα τα αρχεία που καθορίζουν τους συνδέσμους των μεθόδων POST και Get από τα REST API ώστε να υπάρχει η σύνδεση με την βάση. Μερικά παραδείγματα:

```
router.route('/signupPatient')
  .post(validateBody(schemas.patientSchema),
  PatientController.signUp);

router.route('/fetchPatientList')
  .get(PatientController.fetchList);

router.route('/patientAddPhoto')
  .post(validateBody(schemas.patientImgSchema),
  PatientController.patientAddPhoto);

router.route('/patientAddPhotoToDb')
  .post(upload.single('upload'),
  PatientController.patientAddPhotoToDb);
```

Στον φάκελο `models` περιέχονται τα μοντέλα τα οποία καθορίζουν το σχήμα των `collections` που δημιουργούνται στην `Mongo` ώστε να έχουν όλα μια κοινή μορφή και να καθορίζονται ποια πεδία θεωρούνται απαραίτητα. Το μοντέλο ενός ασθενή έχει την εξής μορφή:

```
const patientSchema = new Schema({
  Name: {
    type: String,
    required: true,
    unique: false,
  },
  Surname: {
    type: String,
    required: true,
    unique: false,
  }
},
{
  Age: {
    type: String,
    required: true
  },
  Registered: {
    type: Date,
    required: true
  },
  Image: {
    type: Buffer
  }
});
```

Μοντέλα υπάρχουν για τους χρήστες, τους ενοίκους, τα στοιχεία των διαγραμμάτων, τις δραστηριότητες και τα γεγονότα που αφορούν τον κάθε ένοικο.

Στον φάκελο `controllers` πραγματοποιούνται όλες οι συναρτήσεις που βρίσκονται σε κάθε `route` από τον φάκελο `routes`. Εν συντομία παρουσιάζεται μια από αυτές, η `PatientController.signUp`.

```
signUp: async (req, res, next) => {
  const {Name, Surname, Age, Registered, Image} = req.value.body;

  //check if user exists
  const foundPatient = await Patient.findOne({Surname});
  if (foundPatient) {
    return res.status(403).json({error: ' patient is already
registered'})
  }
  //create new user
  const newPatient = new Patient({Name, Surname, Age, Registered,
Image});
  let dateTime = new Date();
  const newEvent = new patientEvent({
    surname: Surname,
    event: 'Registered at ' + dateTime.toISOString().slice(0,
10),
    eventDate: dateTime.toISOString().slice(0, 10)
  });
  await newPatient.save();
  await newEvent.save();
  //respond with newPatient data
```

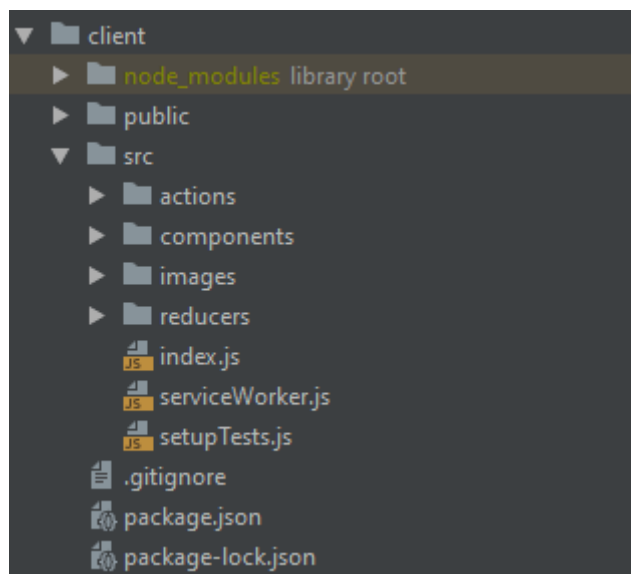
```
res.status(200).json({newPatient});  
}
```

Το υποσύστημα server ακολουθεί αυτήν την δομή ώστε το υποσύστημα client ή ο οποιοσδήποτε client δημιουργηθεί στο μέλλον να βλέπει την βάση μέσω των συνδέσμων που έχουν δημιουργηθεί στον server.

Ο φάκελος configuration έχει δεδομένα τα οποία δεν θα μπορούσαν να είναι στον κύριο κώδικα όπως τα χαρακτηριστικά κλειδιά για την δημιουργία tokens κατά την σύνδεση.

6.2. Υποσύστημα Client

Το υποσύστημα Client δημιουργήθηκε ώστε να υπάρχει το γραφικό περιβάλλον για τους υπαλλήλους να βλέπουν τα δεδομένα της βάσης και να διαχειρίζονται όλη την πληροφορία. Αναλυτική παρουσίαση λειτουργίας υπάρχει στο παράρτημα στο User Manual. Παρουσιάζεται συνοπτικά η δομή του client:



Εικόνα 16: Δομή υποσυστήματος Client

Η δομή του site και ο κώδικας html(jsx) ουσιαστικά βρίσκεται στον φάκελο components. Είναι ένα σύνολο επαναχρησιμοποιούμενων στοιχείων ώστε να είναι εύκολα επεκτάσιμο. Υπάρχουν components τα οποία χρησιμοποιούνται επανειλημμένα μέσα στο σύστημα.

Ο φάκελος actions περιέχει όλη την διαδικασία σύνδεσης με την βάση χρησιμοποιώντας τους συνδέσμους του υποσυστήματος Server. Η συνάρτηση saveJobImage παρουσιάζεται παρακάτω σύμφωνα με την οποία γίνεται αποθήκευση φωτογραφίας του ενοίκου στο σύστημα:

```
export const saveJobImage = data => {  
  return async dispatch => {  
  
    try {  
      const res = await  
      axios.post('http://localhost:5000/patient/updatePatient', data);  
      dispatch({
```

```

        type: SAVE_JOB_IMAGE,
        payload: res.data
      });
    } catch (error) {
      dispatch({
        type: AUTH_ERROR,
        payload: 'Something went terribly wrong '
      })
    }
  }
};
};
};

```

Με παρόμοιο τρόπο λειτουργούν όλες οι υπόλοιπες συναρτήσεις. Στο τέλος αποθηκεύεται μόνο το payload από τον παραπάνω κώδικα, δουλειά την οποία αναλαμβάνουν οι reducers από τον αντίστοιχο φάκελο.

```

const DEFAULT_STATE = {
  correctPatientSyntax: false,
  patients: '',
  errorMessage: '',
  events: '',
  activity: ''
};

export default (state = DEFAULT_STATE, action) => {
  switch (action.type) {
    case PATIENT_SIGN_UP:
      console.log('we are correct');
      return { ...state, correctPatientSyntax: true,
errorMessage: '' };
    case PATIENT_SIGN_UP_ERROR:
      console.log('we have error');
      return { ...state, errorMessage: action.payload };
    case PATIENT_GET_LIST :
      console.log('Lets fetch the list of patients');
      return { ...state, patients: action.payload };
    case SAVE_JOB_IMAGE:
      console.log("saving patient's image");
      return { ...state };
    case GET_PATIENT_DATA:
      return { ...state, events: action.payload };
    case ADD_PATIENT_EVENT:
      return { ...state, events: action.payload };
    case ADD_ACTIVITY:
      return { ...state, events: action.payload };
    case GET_ACTIVITY:
      return { ...state, activity: action.payload };
    default:
      return state
  }
}

```

Οι reducers τραβάνε μόνο το payload και το αποθηκεύουν ως μέρος του state (κατάσταση) της σελίδας. Με οποιαδήποτε ανανέωση του state γίνεται ανανέωση της σελίδας. Έτσι όταν επιλέγει ο χρήστης κάποιο κουμπί για να εμφανίσει δεδομένα ανανεώνεται η κατάσταση και αυτόματα και η ίδια η σελίδα.

6.3. Πακέτα (Middleware)

Πέρα από τις γλώσσες που χρησιμοποιήθηκαν ένα μεγάλο μέρος έτοιμων πακέτων βοήθησε στην υλοποίηση του συστήματος. Πρόκειται για ανοιχτά πακέτα τα οποία προσαρμόζονται στις παραπάνω γλώσσες.

- **express**: Πρόκειται για ένα ευέλικτο πλαίσιο εφαρμογών ιστού για Node.js που παρέχει ένα δυνατό σύνολο από δυνατότητες για ανάπτυξη εφαρμογών ιστού.
- **mongoose**: Βιβλιοθήκη που διαχειρίζεται την σύνδεση της MongoDB με την Node. Σημαντικό προσόν είναι η δυνατότητα δημιουργίας σχημάτων δεδομένων για κάθε έγγραφο(document) που θα γραφτεί στην βάση.
- **passport**: Βιβλιοθήκη για την σύνδεση(sign IN – sign Up) των χρηστών. Προσφέρει την δυνατότητα δημιουργίας token για την διατήρηση της σύνδεσης.
- **bcryptjs**: Βιβλιοθήκη που παρέχει την δυνατότητα να αποθηκεύεις κρυπτογραφημένους κωδικούς.
- **@hari/joi** : Βιβλιοθήκη για επικύρωση δεδομένων κατά την αποστολή REST με βάση σχημάτων(Schemas) που δημιουργούμαι.
- **bodyparser**: Πακέτο που δίνει την δυνατότητα να διαβάζει τα δεδομένα που αποστέλλονται μαζί με τα json.
- **redux**: Πακέτο που βοηθάει την react να διατηρεί την κατάσταση (state) και να την μεταφέρει από σελίδα σε σελίδα (Component σε Component)
- **face_recognition**: Πακέτο της python για ανάλυση δεδομένων προσώπου και χαρτογράφηση χαρακτηριστικών.
- **chart.js**: Πακέτα για την δημιουργία διαγραμμάτων και στατιστικών δεδομένων.

6.4. WEB Services

Συνοπτικά αναφέρονται όλα τα web services που αναπτύχθηκαν και καλούν κάποιο REST HTTP request ώστε να παρέχουν ή να τραβήξουν δεδομένα από την βάση. Τα REST που χρησιμοποιήθηκαν είναι της μορφής POST ή GET

- **signUp**: Εγγραφή χρηστών στο σύστημα. Ως δεδομένα εισάγονται τα στοιχεία από την αντίστοιχη φόρμα και ως αποτέλεσμα επιστρέφεται ένα token που διατηρεί τα στοιχεία της σύνδεσης ενεργά.
- **signIn**: Είσοδος χρηστών στο σύστημα. Ως δεδομένα εισάγονται τα στοιχεία από την αντίστοιχη φόρμα και ως αποτέλεσμα επιστρέφεται ένα token που διατηρεί τα στοιχεία της σύνδεσης ενεργά.

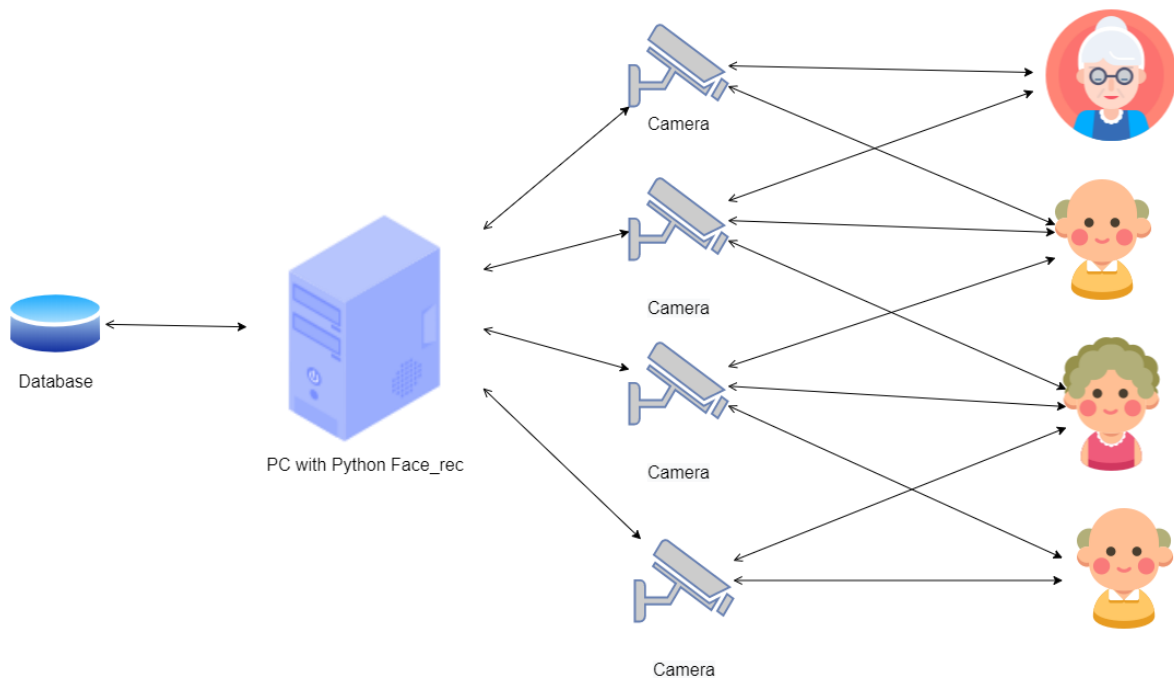
- **signOut:** Έξοδος χρηστών από το σύστημα. Το συγκεκριμένο web service διαγράφει το token από την κατάσταση που έχει αποθηκευτεί και τερματίζει ουσιαστικά την σύνδεση.
- **getSecret:** Web service το οποίο καλείται για την ανάσυρση δεδομένων χρήστη από την βάση με όρισμα το token. Τα στοιχεία αποθηκεύονται στην κατάσταση (state) της react ώστε να είναι προσβάσιμα σε όλα τα τμήματα τις πλατφόρμας.
- **patientSignUp:** Δημιουργία νέου πελάτη στην βάση. Ως δεδομένα εισάγονται τα στοιχεία από την αντίστοιχη φόρμα δημιουργίας ενοίκου. Ως αποτέλεσμα δεν επιστρέφεται κάτι παρά μόνο γίνεται προώθηση σε σελίδα για λήψη φωτογραφίας.
- **getPatientList:** Web service το οποίο καλείται και φέρνει όλα τα στοιχεία πελατών από την βάση. Δεν δέχεται κάποιο στοιχείο ως όρισμα.
- **saveJobImage:** Αποθήκευση φωτογραφίας ενοίκου στην βάση. Ως δεδομένα για την αποστολή στέλνεται το επίθετο του ηλικιωμένου.
- **fetchPatientData:** Web service το οποίο καλείται και επιστρέφει ως αποτέλεσμα τα δεδομένα ενός ηλικιωμένου και τις καθορισμένες του δραστηριότητες. Ως δεδομένο μαζί με το request στέλνουμε και το επίθετο για το οποίο ζητάμε δεδομένα.
- **addPatientEvent:** Web service το οποίο καλείται για την δημιουργία μιας νέας δραστηριότητας για έναν πελάτη. Ως δεδομένο μαζί με το request στέλνουμε και το επίθετο για το οποίο θέλουμε να εισάγουμε δεδομένα.
- **addActivity:** Web service το οποίο καλείται για την δημιουργία μιας νέας δραστηριότητας. Ως δεδομένα εισάγονται τα στοιχεία από την αντίστοιχη φόρμα δημιουργίας δραστηριότητας.
- **fetchActivity:** Web service το οποίο καλείται και επιστρέφει ως αποτέλεσμα όλες τις καθορισμένες δραστηριότητες.
- **addRules:** Web service το οποίο καλείται για την αποθήκευση νέων κανόνων ασφαλείας.
- **fetchLocationPreferences:** Web service το οποίο καλείται και επιστρέφει δεδομένα σχετικά με τις προτιμήσεις του χρήστη με βάση τις περιοχές που βρέθηκε.
- **fetchLineTimeDiagramData:** Web service το οποίο καλείται και επιστρέφει δεδομένα σχετικά με τις χρονικές στιγμές που βρέθηκε σε διάφορες θέσεις.

7. Face Recognition

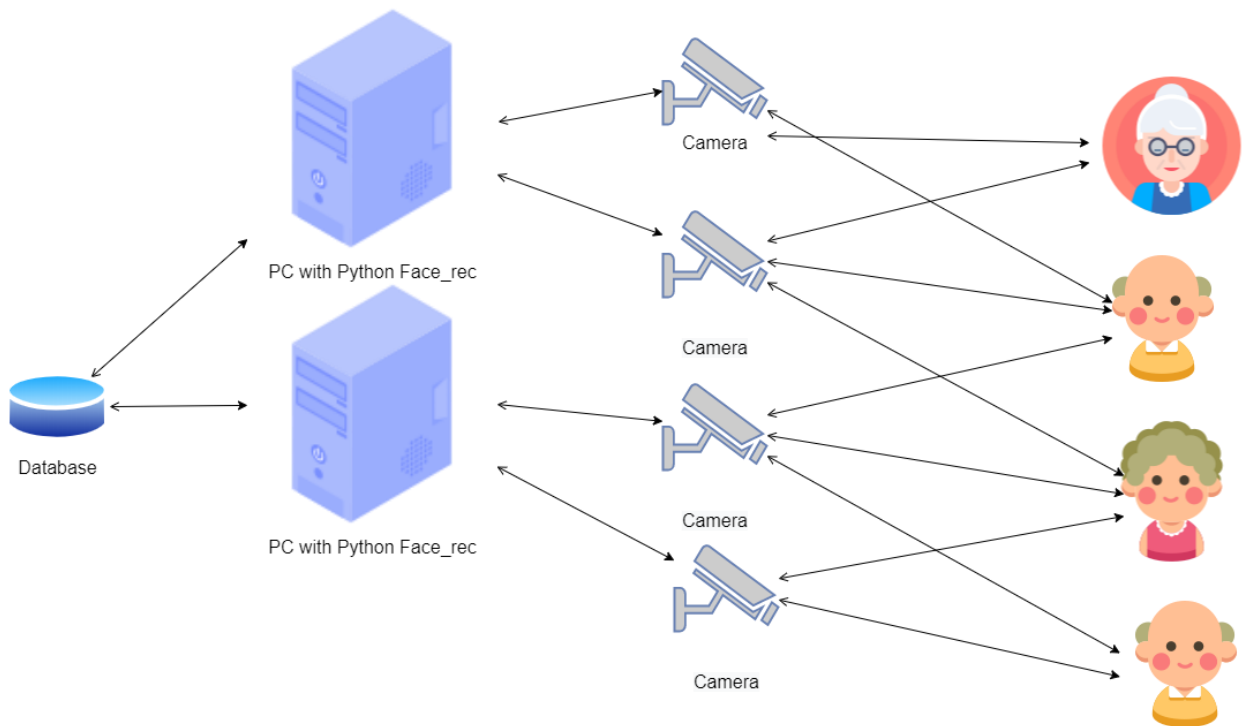
Το σύστημα του face recognition πρόκειται για ένα σύστημα που όπως έχει αναφερθεί λειτουργεί σχεδόν ανεξάρτητα από το υπόλοιπο καθώς η σύνδεση γίνεται μέσω της βάσης. Πρόκειται βασικό κομμάτι όμως του συνολικού συστήματος καθώς παρέχει τα δεδομένα για την στατιστική ανάλυση της συμπεριφοράς του κάθε ηλικιωμένου. Επίσης πρόκειται αναπόσπαστο κομμάτι στον τομέα της ασφάλειας μέσω ενός συνόλου δυνατοτήτων που παρέχει και θα αναλυθεί παρακάτω.

7.1. Εγκατάσταση Συστήματος

Το σύστημα του face recognition αποτελείται από ένα σύνολο καμερών οι οποίες συνδέονται με έναν ή περισσότερους υπολογιστές στους οποίους τρέχει ο κώδικας της ρυθμόν για την αναγνώριση προσώπου και την ανάλυση δεδομένων. Μερικές περιπτώσεις χρήσεων απεικονίζονται στις παρακάτω εικόνες:



Εικόνα 17: Διάταξη 1 Face Recognition



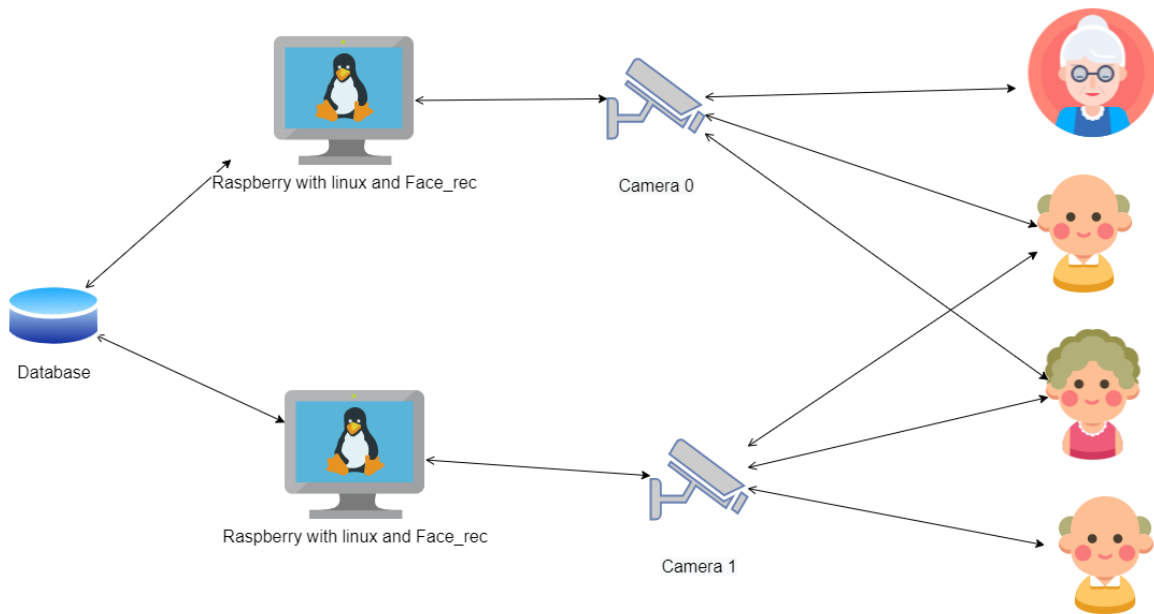
Εικόνα 18: Διάταξη 2 Face_recognition

Το σύστημα και ο τρόπος με τον οποίο θα αναπτυχθεί το face_recognition εξαρτάται άμεσα από τον αριθμό των καμερών, των αριθμό δεδομένων που έχουμε να επεξεργαστούμε και τους διαθέσιμους πόρους που έχουμε.

Για την μείωση του κόστους αντί για σταθεροί υπολογιστές έχει περαστεί ο κώδικας στους μίνι υπολογιστές raspberry pi 4 που τρέχουν σε Linux(Rasbian) με 4gb μνήμης. Το κάθε raspberry μπορεί να συνδεθεί με παραπάνω από μια κάμερες. Για καλύτερη απόδοση όμως είναι θεμιτό να συνδέεται το κάθε ένα με έως τρεις κάμερες.

Στην δοκιμαστική διάταξη που δοκιμάζουμε το σύστημα αποτελείται από δύο raspberry με μια κάμερα το κάθε ένα. Η κάθε κάμερα (camera0, camera1) αντιστοιχεί σε έναν χώρο με δραστηριότητες. Η αναβάθμιση του συστήματος σε σύστημα με περισσότερες κάμερες είναι εύκολη καθώς ο κώδικας δεν αλλάζει απλά μεταβάλλεται ο αριθμός εισόδων. Η πειραματική αυτή διάταξη χρησιμοποιείται για την εξαγωγή των πρώτων συμπερασμάτων και διόρθωση προβλημάτων που μπορεί να έχουν δημιουργηθεί χωρίς να γίνουν αντιληπτά.

Παρακάτω παρουσιάζεται ένα μοντέλο της πειραματικής διάταξης:



Εικόνα 19: Δοκιμαστική Διάταξη

7.2. Ανάλυση Κώδικα Συστήματος

Στο σύστημα αναγνώρισης προσώπου δεν υπάρχει user interface για αυτό κρίνεται σκόπιμο να αναλυθούν μερικές μεταβλητές του κώδικα καθώς και τμήματα του κώδικα που μπορούν να βελτιώσουν τα αποτελέσματα ή να χρησιμοποιηθούν για εξαγωγή διαφορετικών συμπερασμάτων.

Συγκεκριμένα:

- Αλλαγή εισόδου εικόνας

```
video = cv2.VideoCapture(0)
```

Με την αλλαγή της μεταβλητής (0) ουσιαστικά μπορεί να γίνει αλλαγή εισόδου κάμερας. Θα μπορούσε για δεύτερη κάμερα να εισαχθεί νέα μεταβλητή:

```
video_second_camera = cv2.VideoCapture(1)
```

και για όσες παραπάνω κάμερες τόσες πιο πολλές μεταβλητές

- Αλλαγή ευαισθησίας αναγνώρισης:

```
TOLERANCE = 0.6
```

Ο αλγόριθμος του face recognition έχει σαν αρχική τιμή την τιμή tolerance στο 0,6. Σε περίπτωση που κάποια άτομα μοιάζουν αρκετά μεταξύ τους ή οι αρχικές εικόνες που έχουμε τραβήξει σαν dataset μοιάζουν αρκετά μεταξύ τους μπορεί να εμφανίζονται λάθος αποτελέσματα σαν αναγνώριση προσώπου.

Όσο πιο χαμηλή η τιμή του tolerance τόσο πιο αυστηρή και με ανάλυση περισσότερων χαρακτηριστικών και σημείων του προσώπου είναι η αναγνώριση. Συστήνεται βέβαια για μεγαλύτερη ακρίβεια να υπάρχει παραπάνω από μια φωτογραφία για τον κάθε χρήστη από διαφορετικές οπτικές γωνίες.

- Αλλαγή τρόπου επεξεργασίας:

```
MODEL = 'hog'
```

Στην παραπάνω γραμμή τίθεται επεξεργασία εικόνων. μέσω του επεξεργαστή. Εναλλακτικά θα μπορούσε να χρησιμοποιηθεί η γραμμή

```
MODEL = 'cnn'
```

εάν θέλαμε να κάνουμε χρήση της κάρτας γραφικών. Ειδικά μάλιστα αν η κάρτα γραφικών έχει το μοντέλο 'CUDA' στα χαρακτηριστικά της μπορεί να επιταχυνθεί αρκετά η διαδικασία. Στο παράδειγμά μας με τα raspberry εξαιτίας της δομής του συστήματός μας χρησιμοποιείται η πρώτη επιλογή.

Τα υπόλοιπα τμήματα του κώδικα δεν είναι δυνατό να τροποποιηθούν χωρίς να επηρεάσουν αρνητικά την λειτουργία του συστήματος. Η χρήση του συστήματος προτείνεται να γίνεται για αυτό το λόγο, μόνο από άτομα με την κατάλληλη εξουσιοδότηση. Σε διαφορετική περίπτωση δεν μπορεί να εξασφαλιστεί η σωστή λειτουργία.

7.3. Λειτουργία Face Recognition

Το πρόγραμμα όπως αναφέρθηκε μπορεί να λειτουργήσει ανεξάρτητα με το υπόλοιπα σύστημα. Κατά την χρήση υπάρχει δυνατότητα να βλέπει, ο εκάστοτε χρήστης σε ζωντανή μετάδοση το video από τις κάμερες να αναγνωρίζει άμεσα ποιος είναι στην εικόνα



Εικόνα 20: Παράδειγμα Live εικόνας Face Recognition

Το πρόγραμμα face-recognition λειτουργεί και με παραπάνω από ένα πρόσωπα στην φωτογραφία. Ουσιαστικά αναλύει την φωτογραφία εντοπίζοντας το εκάστοτε πρόσωπο. Στην συνέχεια εντοπίζει τα πρόσωπα που είναι εγγεγραμμένα στην βάση. Αναλύει τις εικόνες της βάσης και χρησιμοποιεί τις αρχικές αναλύσεις και τις συγκρίνει μια προς μια με τις εικόνες της βάσης. Αν εντοπίσει το πρόσωπο σε βαθμό αποδεκτό με βάση το TOLERANCE που αναφέρθηκε προηγουμένως εμφανίζει το

όνομα στην οθόνη και στέλνει την πληροφορία στην βάση. Στα δεδομένα υπάρχει πιθανότητα σφάλματος ειδικά αν τα πρόσωπα έχουν αρκετά παρόμοια χαρακτηριστικά για αυτό συνίσταται η χρήση παραπάνω από μια εικόνες για κάθε χρήστη. Έτσι κατά την επεξεργασία του εκάστοτε καταχωρημένου στην βάση χρήστη εντοπίζονται περισσότερα χαρακτηριστικά ώστε να υπάρξει πιο ασφαλής αντιστοίχιση.

Ταυτόχρονα στην βάση αποθηκεύονται τα δεδομένα σε ένα έγγραφο(document) που αφορά τον χρήστη:

<pre> (1) ObjectId("5efa96a2ba9d1e828292dc47") _id Surname Events [] [0] Time Camera [1] Time Camera [2] Time Camera [3] Time Camera [4] [5] [6] </pre>	<pre> { 3 fields } ObjectId("5efa96a2ba9d1e828292dc47") Mathioudis { 1 field } [4165 elements] { 2 fields } 2020-06-30 04:34:27.144Z 0 { 2 fields } 2020-06-30 04:34:27.723Z 0 { 2 fields } 2020-06-30 04:34:28.297Z 0 { 2 fields } 2020-06-30 04:34:28.874Z 0 { 2 fields } { 2 fields } { 2 fields } </pre>
--	--

Εικόνα 21: Αποθήκευση δεδομένων στην βάση

Τα παραπάνω δεδομένα λαμβάνονται με αρκετά μεγάλη συχνότητα. Έτσι δημιουργείται μια βάση με χιλιάδες εγγραφές μέσα σε λίγες μόνο ώρες. Η ανάλυση και επεξεργασία των παραπάνω δεδομένων είναι απαραίτητη. Η διαδικασία ανάλυσης περιγράφεται αναλυτικά στο κεφάλαιο 9.

Το κάθε έγγραφο στην βάση έχει τις εξής ιδιότητες:

- `_id`: μοναδικό για κάθε έγγραφο.
- `Surname`: Το επίθετο του ατόμου που αναγνώρισε. Ύστερα από παραδοχή στο δοκιμαστικό σύστημα θεωρούμε ότι υπάρχει μόνο ένας χρήστης με το κάθε επώνυμο. Η αλλαγή και καταχώρηση μόνο με βάση το `_id` θα γίνει σε μεταγενέστερο στάδιο από αυτό των δοκιμών.
- `Events.[]`: Είναι η κάθε ξεχωριστή εγγραφή που αναφέρεται όταν εντοπίζεται σε ένα frame του βίντεο το κάθε άτομο και αποτελείται από δύο χαρακτηριστικά
 - `Time`: Πρόκειται για την χρονική στιγμή που εντοπίζεται το άτομο από το πρόγραμμα. Αποθηκεύεται σε μορφή ημερομηνίας (datetime) ώστε να μπορεί να επεξεργαστεί και σαν ημερομηνία

όταν το τραβήξει η πλατφόρμα η οποία τρέχει σε διαφορετική γλώσσα προγραμματισμού.

- Camera: Ο κωδικός της κάμερας στον οποίο εντοπίστηκε το άτομο. Η κάθε κάμερα αντιστοιχεί σε ένα δωμάτιο ή μια δραστηριότητα. Στο τελικό στάδιο της πραγμάτωσης θα γίνει και αντιστοίχιση των καμερών με τα ονόματα των δωματίων ή των δραστηριοτήτων.

8. Ανάλυση Δεδομένων

Στο κεφάλαιο αυτό παρουσιάζονται μερικά σενάρια ανάλυσης δεδομένων καθώς και ο τρόπος με τον οποίο επεξεργάζονται και παρουσιάζονται τα δεδομένα

8.1. Ανάγκη Ανάλυσης Δεδομένων

Η συνεχής ροή δεδομένων προς την βάση δημιουργεί την δυνατότητα να χρησιμοποιήσουμε τα δεδομένα αυτά ώστε να εξάγουμε συμπεράσματα για τις προτιμήσεις των πελατών μας. Επιπλέον το σύστημα μπορεί να λειτουργεί και ως σύστημα ασφαλείας για την επιχείρηση εφόσον καταγράφει τις κινήσεις στον χώρο σε όλη την διάρκεια του εικοσιτετράωρου.

Το πλήθος όμως των δεδομένων και η πολύ γρήγορη αύξηση των εγγραφών οδηγεί σε προβλήματα στην επεξεργασία καθώς ανά δευτερόλεπτο μπορεί να υπάρχουν παραπάνω από τρεις εγγραφές για κάθε πελάτη ανά δευτερόλεπτο

Key	Value
▼ (1) ObjectId("5efa96a2ba9d1e828292dc47")	{ 3 fields }
_id	ObjectId("5efa96a2ba9d1e828292dc47")
Surname	Mathioudis
▼ Events	{ 1 field }
> []	[4165 elements]

Εικόνα 22: 4165 εγγραφές σε 10 ώρες

Έτσι δημιουργείται η ανάγκη για κανονικοποίηση, επεξεργασία και ανάλυση των δεδομένων.

8.2. Σενάρια ανάλυσης

Ο τρόπος ανάλυσης δεδομένων δεν είναι μονόπλευρος και ανάλογα με τα συμπεράσματα που θέλουμε να εξάγουμε χρειαζόμαστε διαφορετικά δεδομένα.

Σενάριο Ανάλυσης 1: Συγκέντρωση στατιστικών στοιχείων που αναφέρονται στην συνολική προτίμηση ενός ενοίκου για τις δραστηριότητες του οίκου ευγηρίας. Στον συνολικό χρόνο παραμονής, είτε πρόκειται για μερικές μέρες, είτε για μερικούς μήνες είτε για χρόνια να μπορεί να βρεθεί η συνολική χρονική διάρκεια για την οποία ο κάθε ένας προτίμησε ή γενικότερα συμμετείχε στις δραστηριότητες.

Σενάριο Ανάλυσης 2: Κατηγοριοποίηση των προτιμήσεων του ενοίκου ανά ημερολογιακή ημέρα. Με αυτό τον τρόπο μπορεί να παρατηρηθεί πως μεταβάλλεται η συμμετοχή του σε δραστηριότητες ανάλογα με την ημέρα, τον μήνα, την εποχή και να συγκριθεί με το πρόγραμμα του οίκου ευγηρίας. Σε ένα υποθετικό σενάριο μπορεί να προτιμάει συνολικά την δραστηριότητα «Α» όμως κάθε Τρίτη ή κάθε Τρίτη στις αρχές του μήνα να επιλέγει την δραστηριότητα «Β»

Σενάριο Ανάλυσης 3: Σύγκριση των προτιμήσεων δύο ή παραπάνω ενοίκων ώστε να εξαχθούν συμπεράσματα για τις κοινές επιλογές τους. Έτσι παρατηρείται ποιοι προτιμούν να κάνουν προγράμματα μεταξύ τους γιατί ίσως είναι παρέα, ποιοι έχουν κοινές δραστηριότητες που θέλουν να απασχοληθούν. Με αυτό τον τρόπο δίνεται η

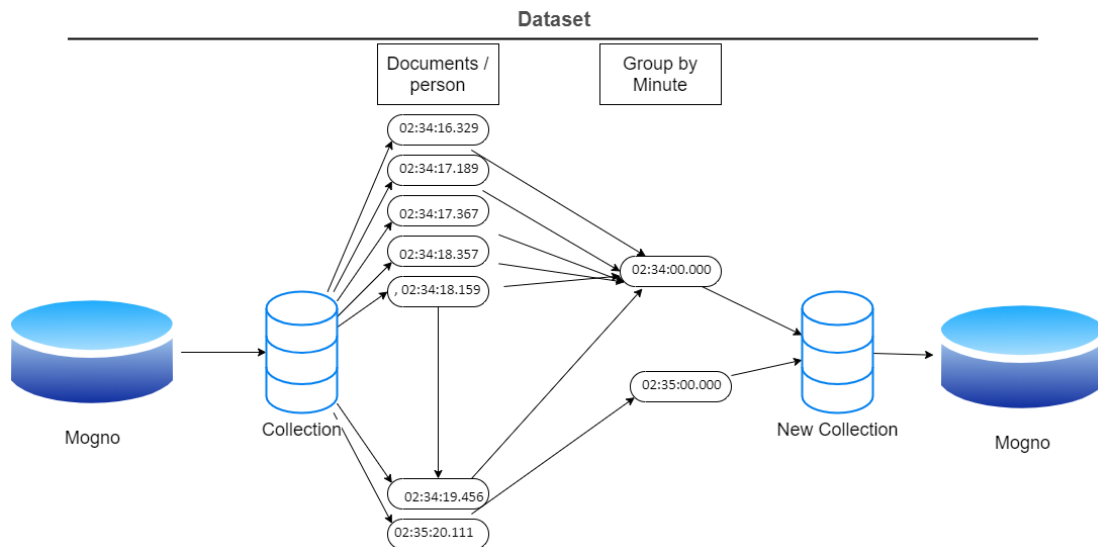
ευκαιρία ομαδοποίησης των ενοίκων σε ομάδες ομοιόμορφες και μπορούν να παρατηρηθούν και φαινόμενα κοινωνικής αποστασιοποίησης.

8.3. Συναρτήσεις Ανάλυσης

Παράλληλα με το πρόγραμμα του face recognition τρέχουν συναρτήσεις που ξεχωρίζουν τα δεδομένα και τα ομαδοποιούν. Συγκεκριμένα οι συναρτήσεις αυτές είναι προγραμματισμένες να τρέχουν ανά δέκα (10) λεπτά. Συνεπώς υπάρχει μια καθυστέρηση στην εμφάνιση των δεδομένων του τελευταίου δεκάλεπτου. Οι συναρτήσεις που κατασκευάστηκαν είναι:

- data_normalization()

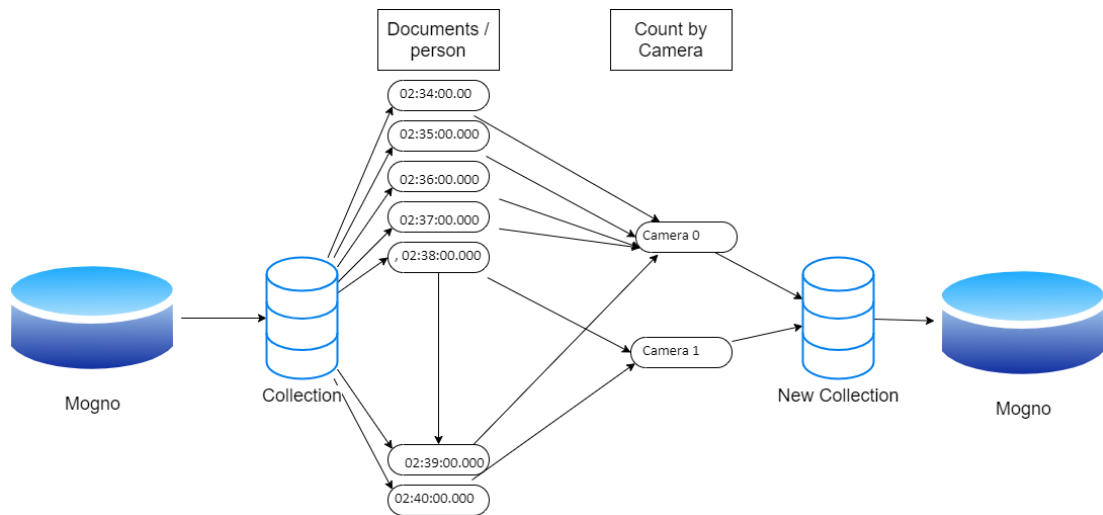
Με τη συνάρτηση αυτή εντοπίζονται όλα τα δεδομένα του κάθε χρήστη και αποθηκεύονται σε νέο πίνακα με διαφορετικό τρόπο. Συγκεκριμένα η συνάρτηση εντοπίζει αν ο χρήστης είχε παρουσία σε μια κάμερα μέσα σε ένα λεπτό και αποθηκεύει την παρουσία μια φορά για κάθε λεπτό. Αν για παράδειγμα ο πελάτης Α εμφανίστηκε στην κάμερα «0» στις 02:34:16.329, 02:34:17.189, 02:34:17.367, 02:34:18.357, 02:34:18.159, 02:34:19.456 θα κρατήσει μόνο μια τελική εγγραφή και θα αποθηκεύσει ότι ο πελάτης εμφανίστηκε στην κάμερα «0» στις 02:34:00.



Εικόνα 23: data_normalization()

- patient_camera_preference()

Στην συνάρτηση αυτή δημιουργούνται κατηγοριοποιήσεις ξεχωρίζοντας πόσα λεπτά βρέθηκε ο κάθε ένας σε κάθε κάμερα.



- `patient_time_diagrams()`

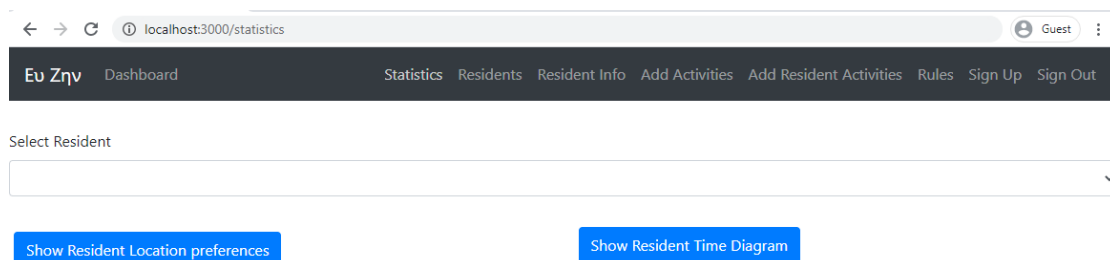
Η συνάρτηση αυτή έχοντας ως όρισμα τα κανονικοποιημένα δεδομένα από την πρώτη συνάρτηση επεξεργάζεται κατάλληλα τα δεδομένα ώστε να μπορούν να χρησιμοποιηθούν σε καρτεσιανά διαγράμματα με άξονα τον χρόνο.

- `notification_data_normalization()`

Στην συνάρτηση αυτή λαμβάνονται ως όρισμα μόνο τα δεδομένα που παραβιάζουν κάποιον κανόνα ασφαλείας και τα ομαδοποιεί ανά λεπτό όπως έκανε η συνάρτηση `data_normalization()` για το όλο το πλήθος των δεδομένων.

8.4. Εμφάνιση αποτελεσμάτων

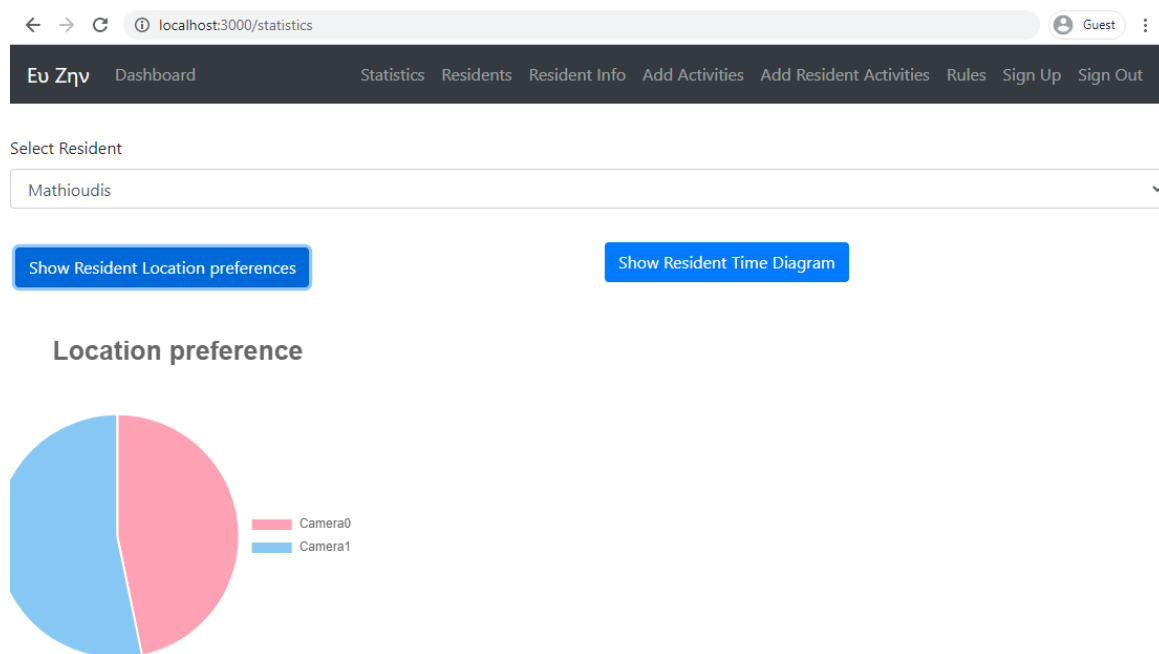
Στον χρήστη του συστήματος δεν δίνεται πρόσβαση στην βάση δεδομένων. Όλη η πληροφορία που χρειάζεται μπορεί να λαμβάνεται από διαγράμματα τα οποία εμφανίζονται στο πρόγραμμα περιηγητή όπου έχει αναπτυχθεί το σύστημα διεπαφής χρήστη. Συγκεκριμένα οι χρήστες έχουν πρόσβαση σε αυτά μέσω της καρτέλας `Statistics` (η ειδικότητα `secretary` δεν έχει πρόσβαση σε αυτά τα δεδομένα):



Εικόνα 24: Επιλογή πελάτη για στατιστικά

Επιλέγοντας τον πελάτη μπορούν να εμφανιστούν τα ανάλογα στατιστικά με το πάτημα των κουμπιών «Show Patient Location Preferences» και «Show Patient Time Diagram». Για να επιλεγθούν και να φορτώσει σωστά η σελίδα με τα αποτελέσματα είναι απαραίτητο επιλεγθεί πρώτα ο ασθενής-πελάτης.

Επιλογή μόνο του κουμπιού «Show Patient Location Preferences» δημιουργεί ένα διάγραμμα της μορφής πίτας με την προτίμηση ανάμεσα στις δύο κάμερες που έχουν θέσει ως παράδειγμα.

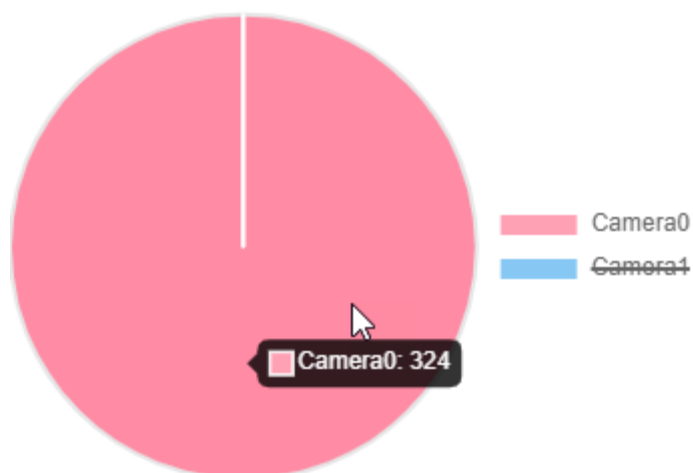


Total hours recorded: 12h

Εικόνα 25: Πίτα δεδομένων προτίμησης κάμερας

Για παραπάνω κάμερες μπορεί ο χρήστης να πατήσει πάνω στην ένδειξη με το χρώμα της αντίστοιχης κάμερας και να την απενεργοποιήσει, ώστε να συγκρίνει μόνο όσες χρειάζεται:

Location preference

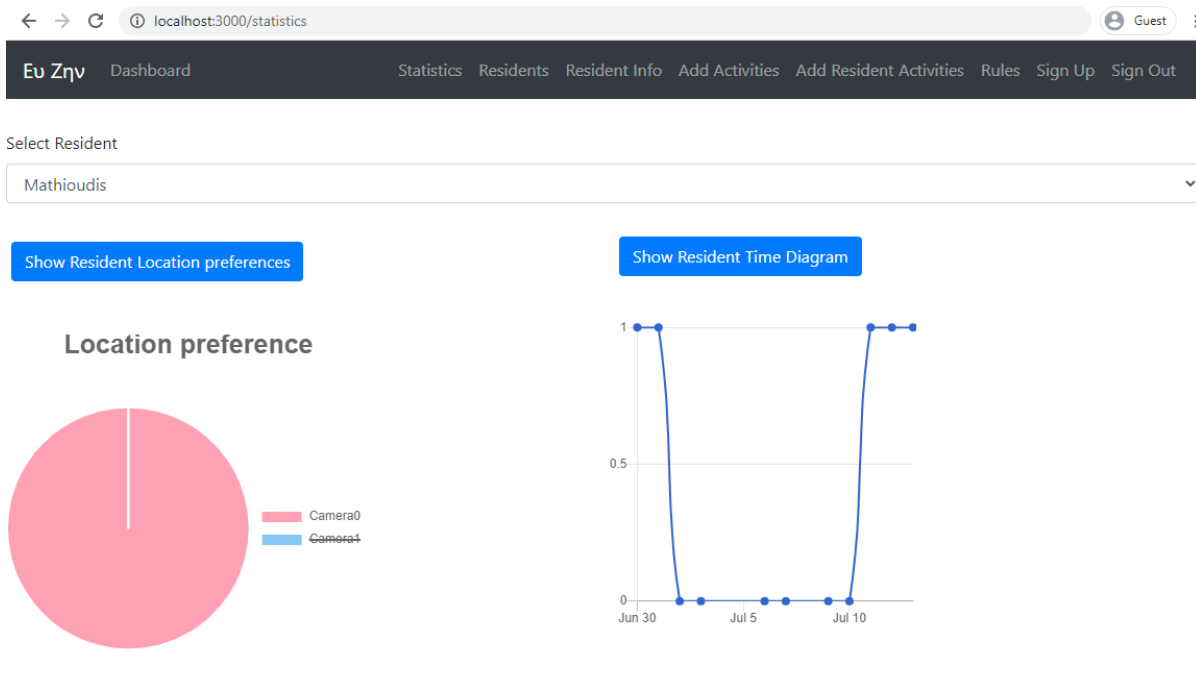


Total hours recorded: 12h

Εικόνα 26: Επιλογή κάμερας για απενεργοποίηση στην ανάλυση

Στο κάτω μέρος ανάλογα με τις ώρες που έχουν καταγραφεί δεδομένα το κείμενο αλλάζει χρώμα. Στην παραπάνω εικόνα το κείμενο με πράσινο χρώμα σημαίνει ότι υπάρχει πληθώρα δεδομένων ώστε να βγάλουμε σχετικά ασφαλή συμπεράσματα. Αν το χρώμα ήταν κίτρινο έχουμε αρκετά δεδομένα τα οποία όμως δεν είναι ακόμα αρκετά για να θεωρήσουμε το δείγμα αξιόπιστο. Τελικά αν εμφανιστεί κόκκινο σημαίνει ότι τα δεδομένα μας δεν είναι αξιόπιστα και δεν μπορούμε να εξάγουμε σωστά συμπεράσματα.

Επιλογή μόνο του κουμπιού « Show Patient Time Diagram» εμφανίζει και ένα διάγραμμα χρόνου όπου δείχνει την προτίμηση του πελάτη σε κάμερα ανά ημέρα. Αυτό δεν σημαίνει ότι ήταν μόνο σε αυτή την κάμερα αλλά ότι εκείνη την ημέρα εμφανίστηκε πιο πολύ σε αυτή την κάμερα.



Εικόνα 27: Διάγραμμα Χρόνου - Κάμερας

Το παραπάνω διάγραμμα έχει ιδιαίτερη σημασία. Μας εμφανίζει την πορεία των προτιμήσεων του ηλικιωμένου όσο περνάει ο χρόνος ή ανά ημέρα. Μπορεί για παράδειγμα ο ασθενής να έχει περισσότερες εμφανίσεις στην κάμερα «0» αλλά την Τρίτη και την Τετάρτη συνέχεια να έχει προτίμηση στην κάμερα «1». Οι λόγοι μπορεί να είναι πολύ απλοί όπως η δραστηριότητα στην συγκεκριμένη κάμερα τις ημέρες αυτές ή αρκετά πιο σύνθετοι.

9. Στατιστικά Συστήματος

Η μελέτη των στατιστικών και των χρόνων χρήσης της εφαρμογής του face recognition οδηγεί στην λήψη συμπερασμάτων για την βελτίωση του. Πραγματοποιήθηκαν ορισμένες μετρήσεις οι οποίες περιγράφονται στις παρακάτω υποενότητες.

9.1. Χρόνος έναρξης προγράμματος

Μετρήθηκε ο χρόνος που χρειάζεται για την έναρξη του προγράμματος με διαφορετικό πλήθος εικόνων στην βάση. Συγκεκριμένα μετρήθηκε ο χρόνος ανάμεσα στην έναρξη της εφαρμογής και την έναρξη της ροής βίντεο.

- Φάκελος known_faces συμπληρωμένος χωρίς νέες εγγραφές

#	Έναρξη εφαρμογής	Έναρξη video stream	Αναγνώριση πρώτου προσώπου	Διαφορά έναρξης εφαρμογής και video stream
1	17:16:09.913537	17:18:07.270153	17:16:21.604481	11,201733
2	17:18:07.270153	17:18:18.641417	17:18:22.224948	11,371264
3	17:29:43.960672	17:29:55.236240	17:29:56.504372	11,275568
4	17:48:55.068165	17:49:06.295363	17:49:06.767619	11,2274713
5	17:49:42.814828	17:49:53.998660	17:49:54.099389	11,183832
6	17:50:19.233054	17:50:30.173496	17:50:30.632757	10,940442
7	17:51:02.910437	17:51:14.659229	17:51:15.168385	11,748792
8	17:56:32.634768	17:56:44.901740	17:56:45.394435	12,266972
9	17:57:05.853703	17:57:17.155201	17:57:17.650396	11,301498
10	17:57:56.971677	17:58:08.261678	17:58:08.736419	11,290001
				11,38075733



Εικόνα 28: Πίνακας και διαγράμματα στατιστικών 1

Παρατηρήθηκε ο μέσος χρόνος έναρξης του προγράμματος ήταν στα 11,38 δευτερόλεπτα

- Φάκελος known_faces άδειος

#	Έναρξη εφαρμογής	Έναρξη video stream	Διαφορά έναρξης εφαρμογής και video stream
1	18:21:37.944199	18:21:48.907624	10,963425
2	18:22:35.492532	18:22:46.394104	10,901572
3	18:23:14.411120	18:23:25.235860	10,82474
4	18:23:50.495189	18:24:01.406242	10,911053
5	18:24:34.686593	18:24:45.618566	10,931973
6	18:25:07.864361	18:25:19.160898	11,296537
7	18:25:42.133624	18:25:53.182765	11,049141
8	18:26:10.040764	18:26:21.110433	11,069669
9	18:26:40.809693	18:26:51.896267	11,086574
10	18:27:13.091081	18:27:24.215583	11,124502
		M.O.	11,0159186



Εικόνα 29: Πίνακας και διαγράμματα στατιστικών 2

Παρατηρήθηκε ο μέσος χρόνος έναρξης του προγράμματος ήταν στα 11,01 δευτερόλεπτα

Ο χρόνος που χρειάζεται για να ξεκινήσει η ροή βίντεο είναι σχεδόν ίδιος είτε δημιουργεί τον φάκελο known_faces από την αρχή είτε τον έχει ήδη συμπληρωμένο. Στην περίπτωση που είναι ήδη συμπληρωμένος επειδή πραγματοποιείται στην αρχή έλεγχος για κάθε εγγραφή με την βάση ώστε να επιβεβαιωθεί η ύπαρξη της(για λόγους ασφαλείας) χρειάζεται λίγο χρόνο παραπάνω. Η διαφορά όμως δεν θεωρείται σημαντική. Να σημειωθεί ότι οι δοκιμές έγιναν με αριθμό εικόνων από πέντε(5) έως σαράντα(40). Η επιβάρυνση του συστήματος με μεγαλύτερο αριθμό εικόνων χρήζει περαιτέρω διερεύνησης.

9.2. Χρόνος αναγνώρισης πρώτου προσώπου από την έναρξη του βίντεο

Εφόσον συμπληρωθεί η δημιουργία του φακέλου known_faces απαιτείται ένα μικρό χρονικό διάστημα μέχρι να αναγνωριστεί το πρώτο πρόσωπο. Εάν υπάρχουν περισσότερα από ένα πρόσωπα ο χρόνος για να αναγνωριστεί το πρώτο πρόσωπο αυξάνεται αλλά μετά τα υπόλοιπα εντοπίζονται σε αρκετά μικρό χρονικό διάστημα

#	Έναρξη εφαρμογής	Έναρξη video stream	Αναγνώριση πρώτου προσώπου	Διαφορά έναρξης video stream και αναγνώρισης πρώτου προσώπου
1	17:16:09.913537	17:18:07.270153	17:16:21.604481	0,489211
2	17:18:07.270153	17:18:18.641417	17:18:22.224948	1,268132
3	17:29:43.960672	17:29:55.236240	17:29:56.504372	1,268132
4	17:48:55.068165	17:49:06.295363	17:49:06.767619	0,472256
5	17:49:42.814828	17:49:53.998660	17:49:54.099389	0,100729
6	17:50:19.233054	17:50:30.173496	17:50:30.632757	0,459261
7	17:51:02.910437	17:51:14.659229	17:51:15.168385	0,509156
8	17:56:32.634768	17:56:44.901740	17:56:45.394435	0,492695
9	17:57:05.853703	17:57:17.155201	17:57:17.650396	0,495195
10	17:57:56.971677	17:58:08.261678	17:58:08.736419	0,474741
				0,6029508

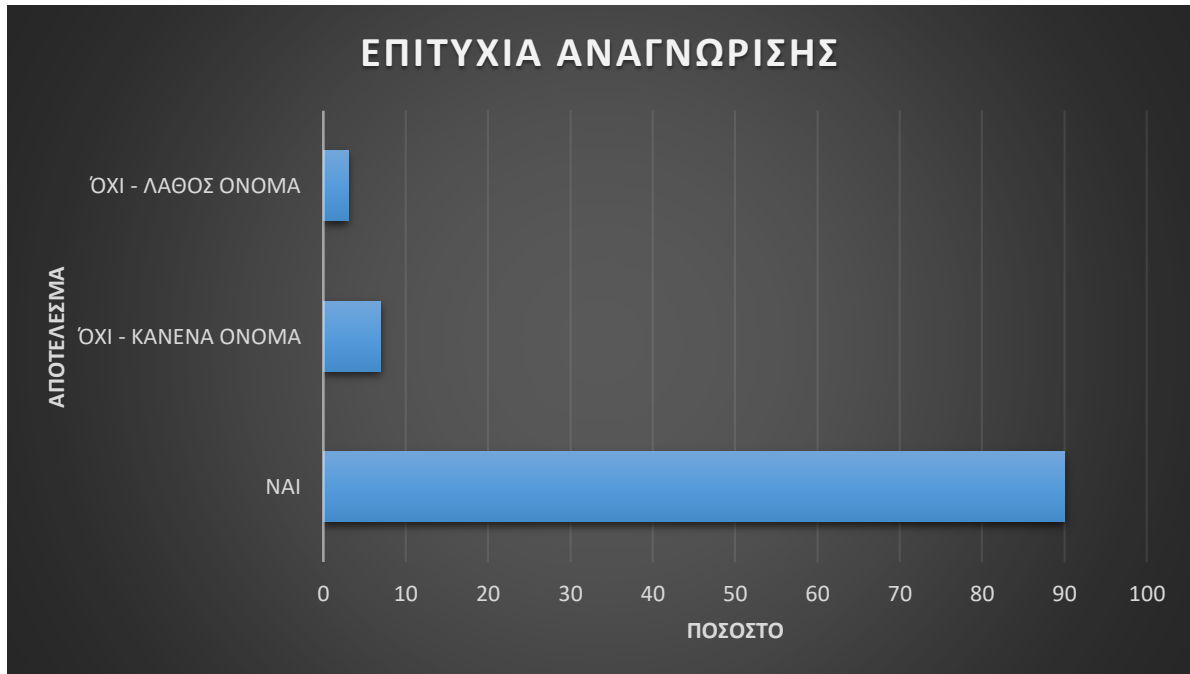


Εικόνα 30: Πίνακας και διαγράμματα στατιστικών 3

Παρατηρείται αρκετά μεγάλη διακύμανση στους χρόνους αναγνώρισης πρώτου προσώπου. Συγκεκριμένα στις προσπάθειες 2 και 3 υπήρχαν συνολικά πέντε και έξι άτομα στην εικόνα αντίστοιχα. Η διενέργεια περισσότερων δοκιμών επιβεβαίωσε τα παραπάνω στοιχεία. Δεν έχει στο δοκιμαστικό στάδιο πραγματοποιηθεί έρευνα με βάση εικόνων με περισσότερες από σαράντα εγγραφές.

9.3. Επιτυχία αναγνώρισης εντός μικρού χρονικού διαστήματος

Μελετήθηκε ο σε πόσες περιπτώσεις εμφανίστηκε όνομα με το πρόσωπο το οποίο αναγνωρίστηκε στην ροή του βίντεο και σε πόσες περιπτώσεις το όνομα δεν ήταν το σωστό. Η μέτρηση αφορά δεδομένα για τα πρώτα δευτερόλεπτα της λειτουργίας (<3s) καθώς στην συνέχεια το ποσοστό αναγνώρισης ανεβαίνει κατακόρυφα. Βασικό στοιχείο είναι ότι οι μετρήσεις έγιναν με ένα ή δύο άτομα στο βίντεο



Εικόνα 31: Διάγραμμα στατιστικών 4

Σε σύνολο εκατό δοκιμών (100) το 90% είχε απόλυτη επιτυχία αναγνωρίζοντας μέσα σε μερικά milliseconds το σωστό όνομα. Σε ελάχιστες περιπτώσεις δεν αναγνώρισε κανένα πρόσωπο (7%). Η αποτυχία αυτή οφείλεται κυρίως στις λίγες φωτογραφίες ανά άτομο. Για αποδοτικότερη λειτουργία χρειάζονται τουλάχιστον τρεις φωτογραφίες ανά άτομο και υπό διαφορετικές γωνίες. Τέλος παρατηρήθηκε λάθος σε αναγνώριση σε ακόμα μικρότερο ποσοστό (3%). Τέτοιες περιπτώσεις είναι κυρίως ανθρώπων με αρκετά κοινά χαρακτηριστικά μεταξύ τους. Μείωση του ποσοστού λάθους μπορεί να γίνει με τον ίδιο τρόπο, δηλαδή αύξηση των φωτογραφιών του εκάστοτε ανθρώπου.

9.4. Απόδοση βίντεο

Ο χρόνος έναρξης του προγράμματος είναι αρκετά μεγάλος και πρέπει σε επόμενες εκδόσεις να μειωθεί. Επίσης παρατηρήθηκε ότι η ροή του βίντεο παρουσιάζει καθυστερήσεις όταν εντοπίζονται πρόσωπα στο βίντεο. Σημαντική μείωση των καθυστερήσεων (περίπου 50%) μπόρεσε να υπάρξει όταν η αναγνώριση γινόταν ανά δύο στιγμιότυπα βίντεο (frame) αντί για κάθε frame.

10. Επίλογος

Έχοντας ολοκληρώσει το πρώτο στάδιο υλοποίησης αξίζει να αναφερθούν ορισμένα από τα επόμενα βήματα που θα πραγματοποιηθούν

10.1. Γραφικό Περιβάλλον

Βραχυπρόθεσμος στόχος είναι η βελτίωση του γραφικού περιβάλλοντος του συστήματος. Το user interface πρέπει να παραμείνει απλό καθώς πρόκειται για σύστημα που προορίζεται για χρήση αποκλειστικά από τους εργαζόμενους. Βελτιώσεις όμως αισθητικές βοηθάνε τον εργαζόμενο να δουλέψει με μεγαλύτερη ευκολία. Επιπλέον η δημιουργία νέων καρτελών με περισσότερες δυνατότητες ή η συμπλήρωση αυτών που υπάρχουν ήδη με νέο περιεχόμενο θα προσφέρει νέα εργαλεία στα χέρια τους.

10.2. Ανάλυση Δεδομένων

Η μεγαλύτερη αλλαγή που θα γίνει στο πρόγραμμα είναι στον τομέα της ανάλυσης δεδομένων. Συγκεκριμένα έχουν μελετηθεί τα επόμενα βήματα:

- Δημιουργία νέων διαγραμμάτων που αφορούν τους χρόνους ανά κάμερα (δραστηριότητα).
- Ανάλυση δεδομένων που αφορούν τις σχέσεις μεταξύ των φιλοξενούμενων. Για παράδειγμα ο «Α» είναι στην κάμερα 0 όταν είναι και ο «Β» ή σε τι ποσοστό είναι μαζί οι δύο. Έτσι δημιουργείται μια βάση δεδομένων που πλέον αναλύει και τις προσωπικές σχέσεις των ανθρώπων ώστε να οργανώνονται δραστηριότητες με βάση τις φιλικές προτιμήσεις. Μια τέτοια ανάλυση μπορεί να οδηγήσει και σε πρόβλεψη καταστάσεων κοινωνικής αποστασιοποίησης ώστε να γίνει επιπλέον προσπάθεια και από το προσωπικό.
- Δημιουργία απουσιολογίου σε ζωντανό χρόνο και ανάλυση των αποτελεσμάτων. Έτσι μπορούμε να εξάγουμε συμπεράσματα για την διάρκεια των δραστηριοτήτων. Μπορεί να ξεκινάει και να έχει δέκα(10) άτομα αλλά όσο περνάει η ώρα να μειώνεται που 'πιθανότατα να σημαίνει ότι καλό είναι να μειωθεί ο χρόνος του.

10.3. Ασφάλεια Επιχείρησης

Στον τομέα της ασφάλειας το επόμενο βήμα είναι να προστεθούν νέοι κανόνες στο σύστημα. Η δημιουργία κανόνων θα γίνει σε συνεργασία με τον υπεύθυνο ασφαλείας

του οίκου ευγηρίας. Έτσι σε κάθε χρονική στιγμή και ιδιαίτερα καθώς αναπτύσσεται το σύστημα καμερών θα μπορούν να γνωρίζουν για οποιονδήποτε κίνδυνο μπορεί να υπάρχει. Για παράδειγμα μπορεί ένας από τους ηλικιωμένους να έχει άνια. Οι κάμερες μόλις τον εντοπίσουν κοντά σε σημείο που υπάρχει έξοδος από το κτίριο θα ειδοποιούν ώστε να δοθεί η ανάλογη προσοχή.

10.4. Συμπεράσματα

Το σύστημα που αναπτύχθηκε στοχεύει στην οργάνωση των λειτουργιών του οίκου ευγηρίας και στην προσθήκη ενός νέου τομέα, της ανάλυσης δεδομένων. Και τα δύο είναι ιδιαίτερα σημαντικά για μια επιχείρηση που θέλει να αναπτυχθεί και έχει ήδη κάνει βήματα προς τα εκεί εκσυγχρονίζοντας τις δραστηριότητες και τον εξοπλισμό της. Η σωστή οργάνωση αντικατοπτρίζεται σε όλη την εικόνα της επιχείρησης και θα ανακουφίσει και το προσωπικό από την διαδικασία αρχειοθέτησης. Ο επιπλέον χρόνος που δημιουργείται για το προσωπικό με αυτό τον τρόπο μπορεί να αξιοποιηθεί στον τομέα της ανάλυσης δεδομένων από το face recognition. Θα χρειαστεί πρώτα η αντίστοιχη εκπαίδευση αλλά τα αποτελέσματα θα είναι ευεργετικά για την επιχείρηση και την καλύτερη διαβίωση των ηλικιωμένων.

Το σύστημα του face recognition, αν και δεν είναι πρωτοποριακό στον κόσμο της πληροφορικής και υπάρχει ήδη αρκετά χρόνια, είναι κάτι καινούριο για αυτό τον κλάδο επιχειρήσεων. Η χρήση του και κυρίως η ανάλυση των γραφημάτων και των δεδομένων που εξάγονται από αυτό έχει την δυνατότητα να αναβαθμίσει την θέση της επιχείρησης στον χώρο και να προσφέρει, πέρα από οικονομικά κέρδη, νέες προοπτικές και νέες ευκαιρίες ανάπτυξης.

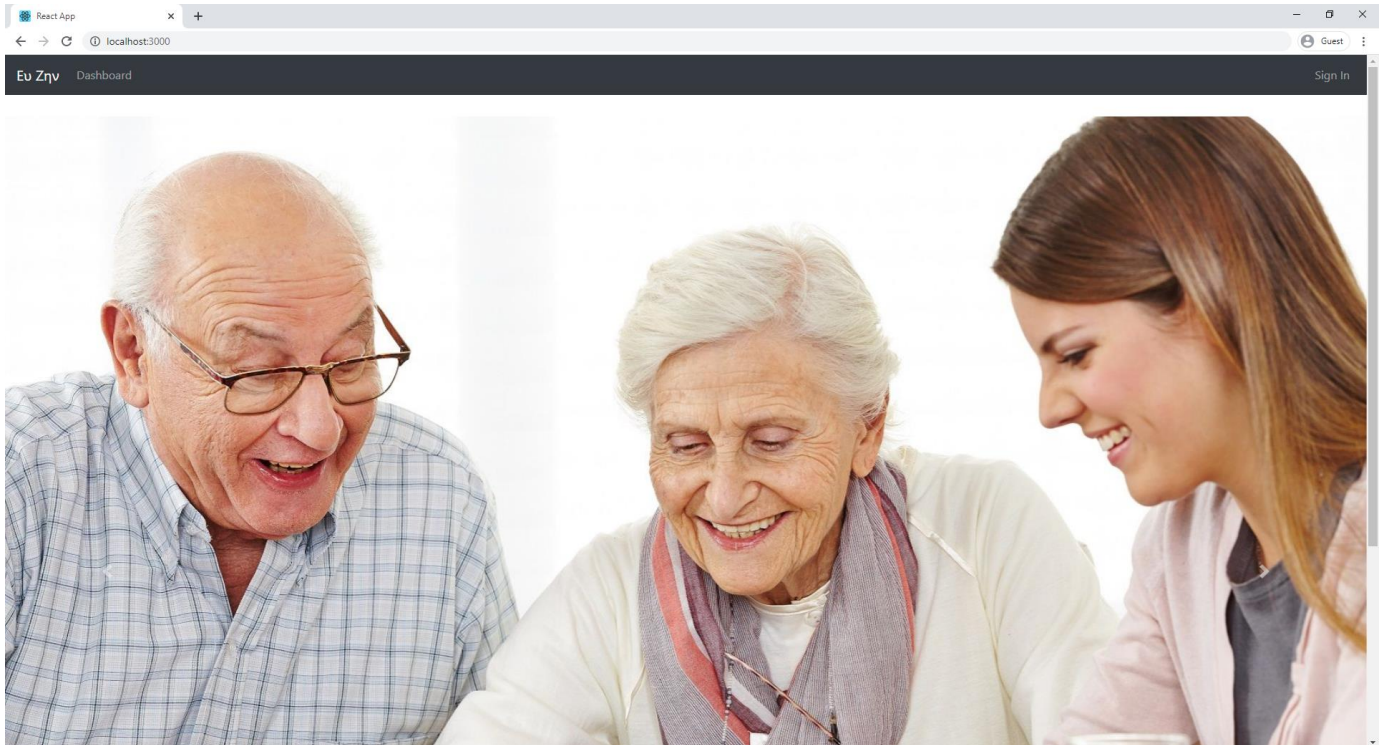
Εν κατακλείδι η πλατφόρμα σε συνδυασμό με το face recognition αναβαθμίζουν τον ρόλο της επιχείρησης και με τις μελλοντικές αναβαθμίσεις δημιουργούν ένα δυνατό εργαλείο που με σωστή χρήση μπορεί να έχει σημαντικά οικονομικά, κοινωνικά, ηθικά και ιατρικά αποτελέσματα.

11. ΠΑΡΑΡΤΗΜΑ – User Manual

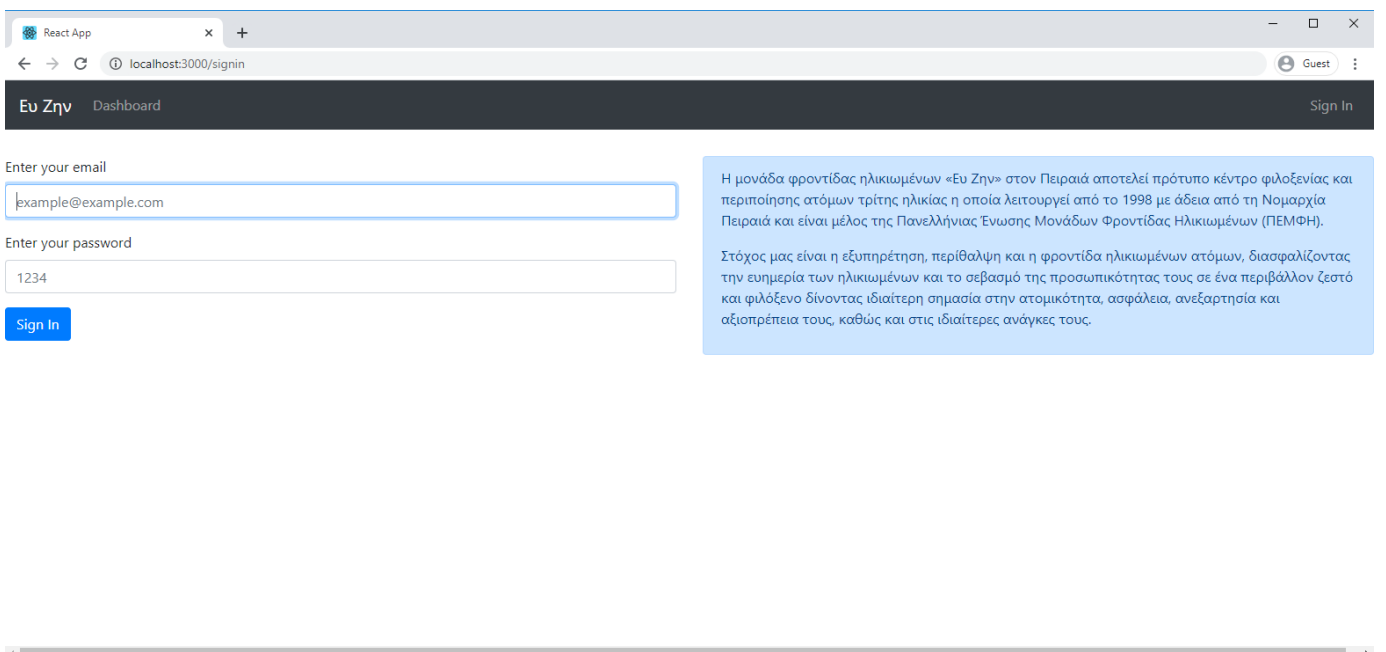
11.1. Είσοδος στο Σύστημα

Η σύνδεση στο site γίνεται στο link κατά την διάρκεια της δοκιμαστικής φάσης στο παρακάτω link το οποίο μας οδηγεί στην αρχική σελίδα:

<http://localhost:3000/>



Για την σύνδεση επιλέγουμε «Sign In» στο επάνω δεξιά μέρος της σελίδας.

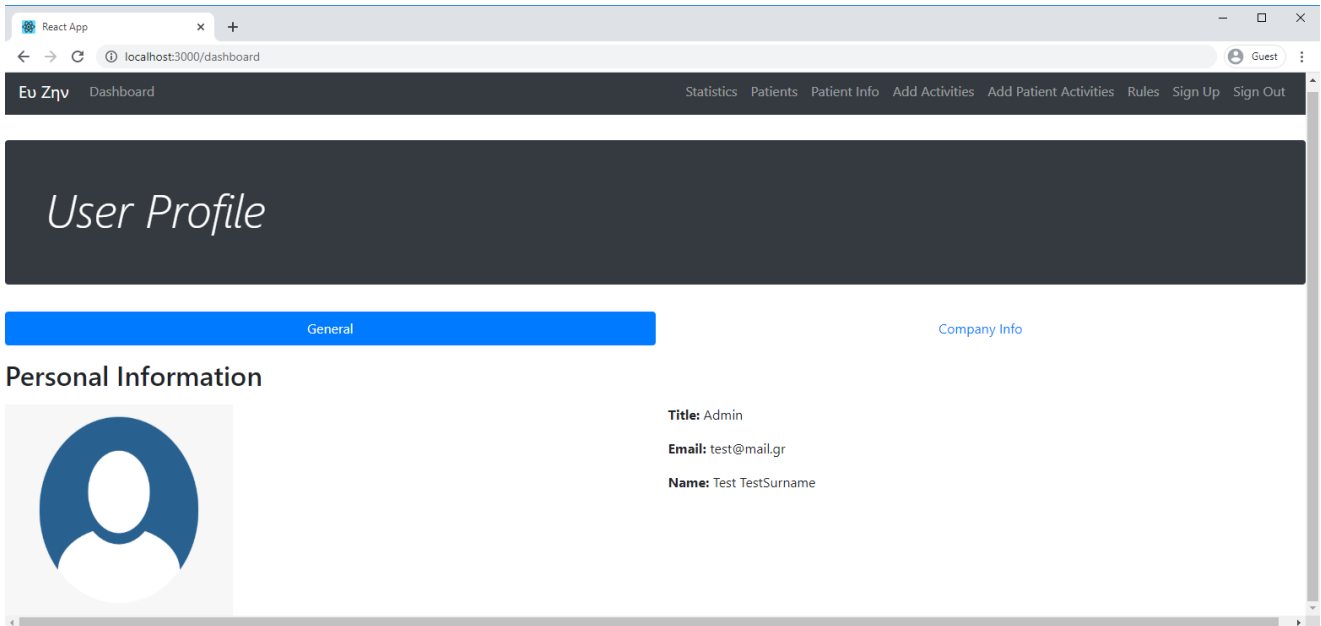


Η μονάδα φροντίδας ηλικιωμένων «Eu Zην» στον Πειραιά αποτελεί πρότυπο κέντρο φιλοξενίας και περιποίησης ατόμων τρίτης ηλικίας η οποία λειτουργεί από το 1998 με άδεια από τη Νομαρχία Πειραιά και είναι μέλος της Πανελληνίας Ένωσης Μονάδων Φροντίδας Ηλικιωμένων (ΠΕΜΦΗ).

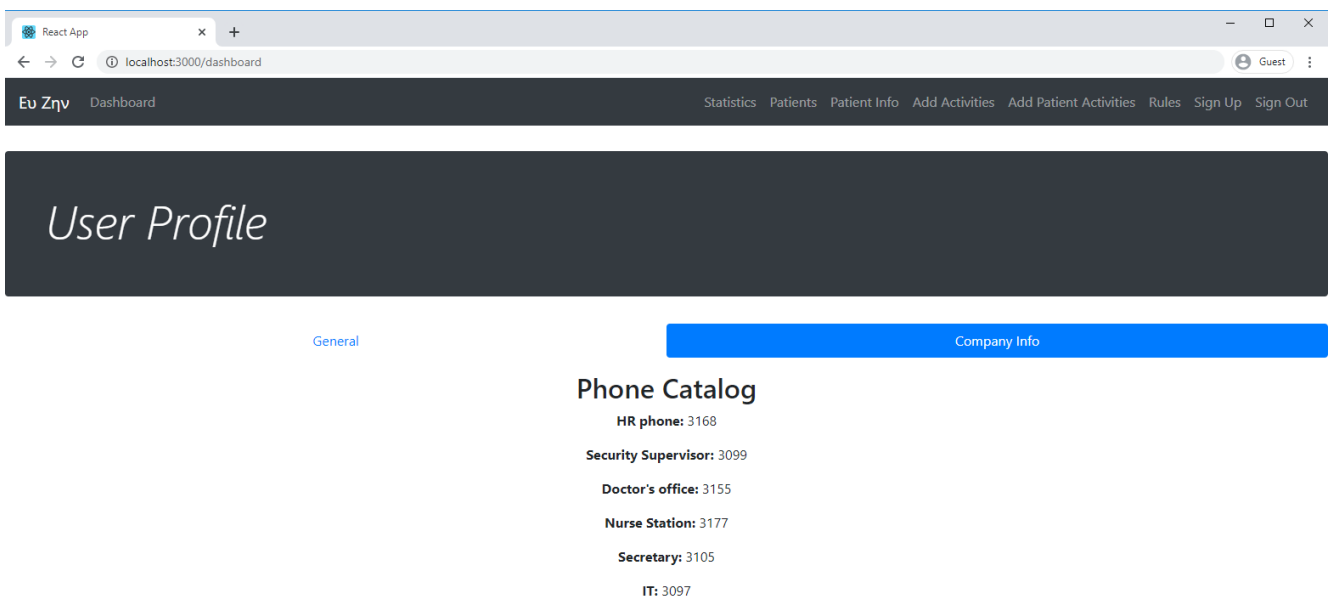
Στόχος μας είναι η εξυπηρέτηση, περιθαλψή και η φροντίδα ηλικιωμένων ατόμων, διασφαλίζοντας την ευημερία των ηλικιωμένων και το σεβασμό της προσωπικότητάς τους σε ένα περιβάλλον ζεστό και φιλόξενο δίνοντας ιδιαίτερη σημασία στην ατομικότητα, ασφάλεια, ανεξαρτησία και αξιοπρέπεια τους, καθώς και στις ιδιαίτερες ανάγκες τους.

Συμπληρώνουμε τα απαραίτητα στοιχεία ώστε να εισέλθουμε στο σύστημα.

Έχοντας τα σωστά διαπιστευτήρια μεταβαίνουμε στην σελίδα προφίλ όπου μπορούμε να δούμε βασικά στοιχεία του προφίλ μας και μερικές βασικές πληροφορίες του κέντρου ευγηρίας όπως κωδικοί τηλεφώνων διαφόρων τμημάτων.



Καρτέλα General

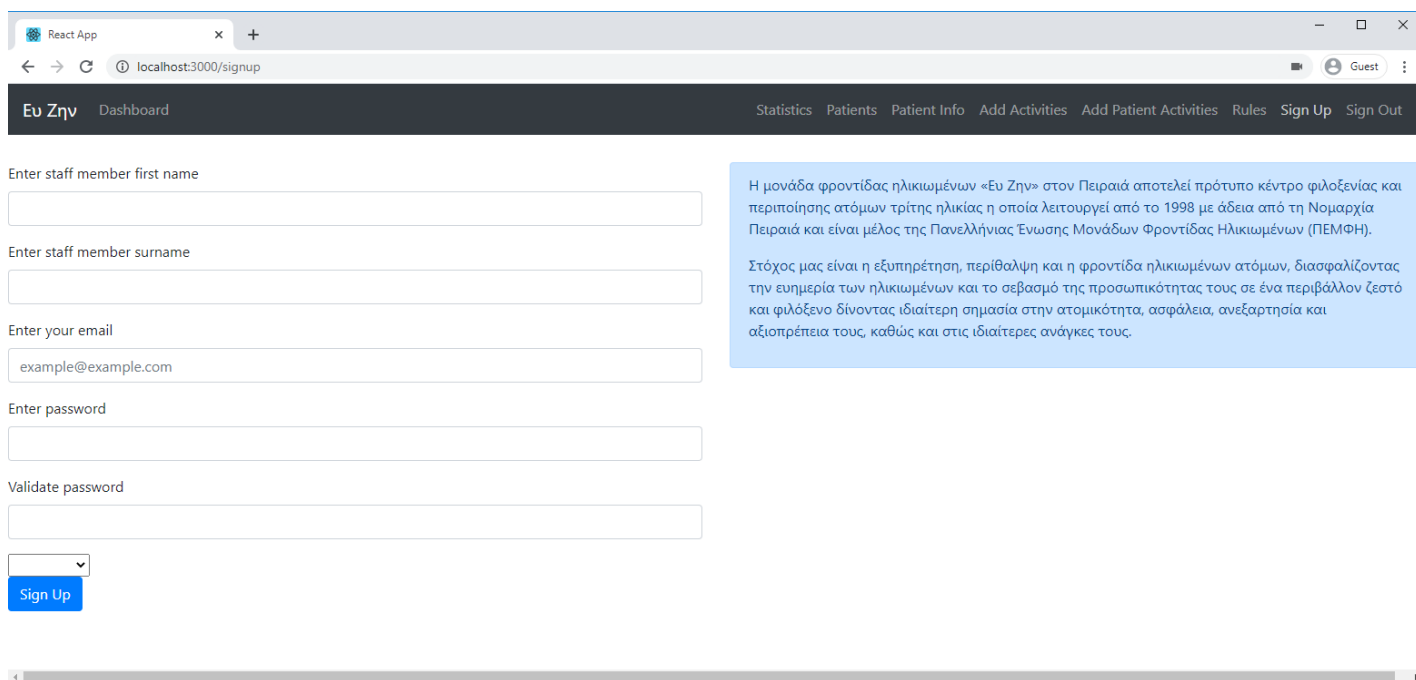


Καρτέλα Company Info.

Η μπάρα στο επάνω μέρος εμφανίζει επιλογές με βάση τα διαπιστευτήρια του χρήστη. Για παράδειγμα η γραμματεία δεν έχει πρόσβαση στα στατιστικά συνεπώς δεν εμφανίζεται η επιλογή Statistics.

11.2. Εγγραφή νέου χρήστη

Η εγγραφή νέου χρήστη στο σύστημα γίνεται μόνο μέσα από τον administrator στην καρτέλα Sign Up.



React App x +

localhost:3000/signup Guest

Eu Ζην Dashboard Statistics Patients Patient Info Add Activities Add Patient Activities Rules Sign Up Sign Out

Enter staff member first name

Enter staff member surname

Enter your email

Enter password

Validate password

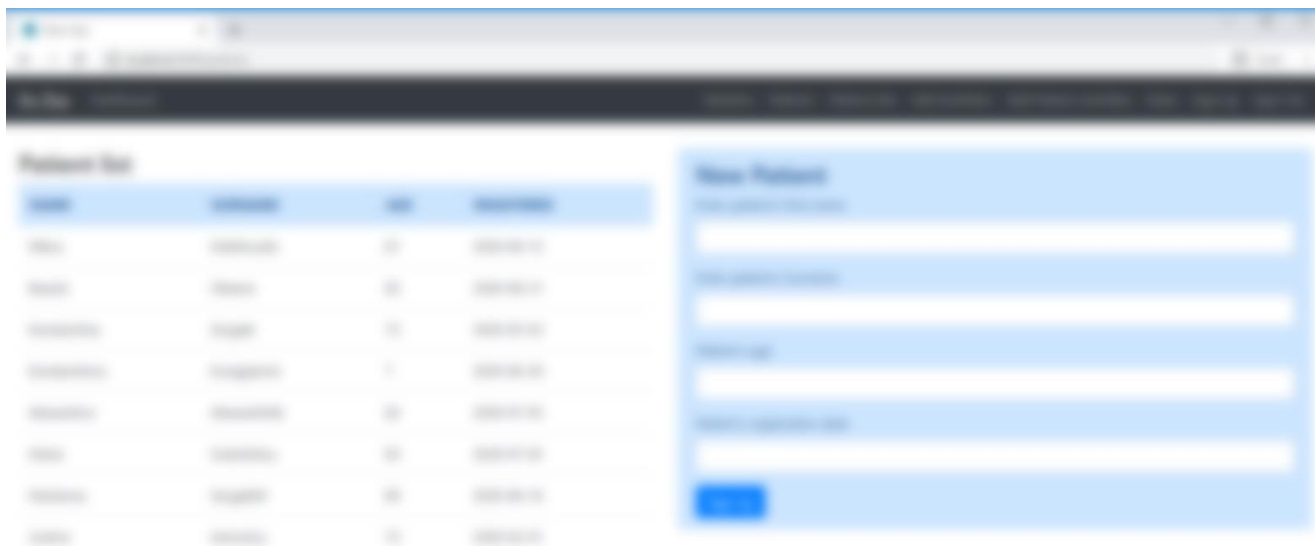
Η μονάδα φροντίδας ηλικιωμένων «Eu Ζην» στον Πειραιά αποτελεί πρότυπο κέντρο φιλοξενίας και περιποίησης ατόμων τρίτης ηλικίας η οποία λειτουργεί από το 1998 με άδεια από τη Νομαρχία Πειραιά και είναι μέλος της Πανελληνίας Ένωσης Μονάδων Φροντίδας Ηλικιωμένων (ΠΕΜΦΗ).

Στόχος μας είναι η εξυπηρέτηση, περίθαλψη και η φροντίδα ηλικιωμένων ατόμων, διασφαλίζοντας την ευημερία των ηλικιωμένων και το σεβασμό της προσωπικότητάς τους σε ένα περιβάλλον ζεστό και φιλόξενο δίνοντας ιδιαίτερη σημασία στην ατομικότητα, ασφάλεια, ανεξαρτησία και αξιοπρέπεια τους, καθώς και στις ιδιαίτερες ανάγκες τους.

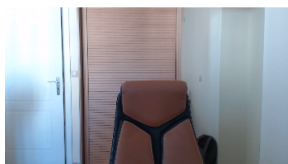
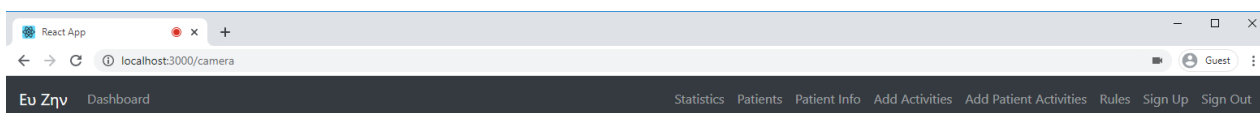
Στην συγκεκριμένη καρτέλα ο administrator συμπληρώνει όλα τα απαραίτητα στοιχεία καθώς και την ειδικότητα του χρήστη με βάση ένα dropdown menu. Επιλέγοντας «Sign Up» γίνεται εγγραφή στο σύστημα.

11.3. Καρτέλες Διαμενόντων/Ασθενών

Επιλέγοντας στην μπάρα το «Patients» μεταβαίνουμε στην σελίδα όπου εμφανίζεται μια πλήρης λίστα με τα ονόματα των φιλοξενούμενων και μερικές βασικές πληροφορίες, καθώς και μια φόρμα εγγραφής νέου ατόμου.

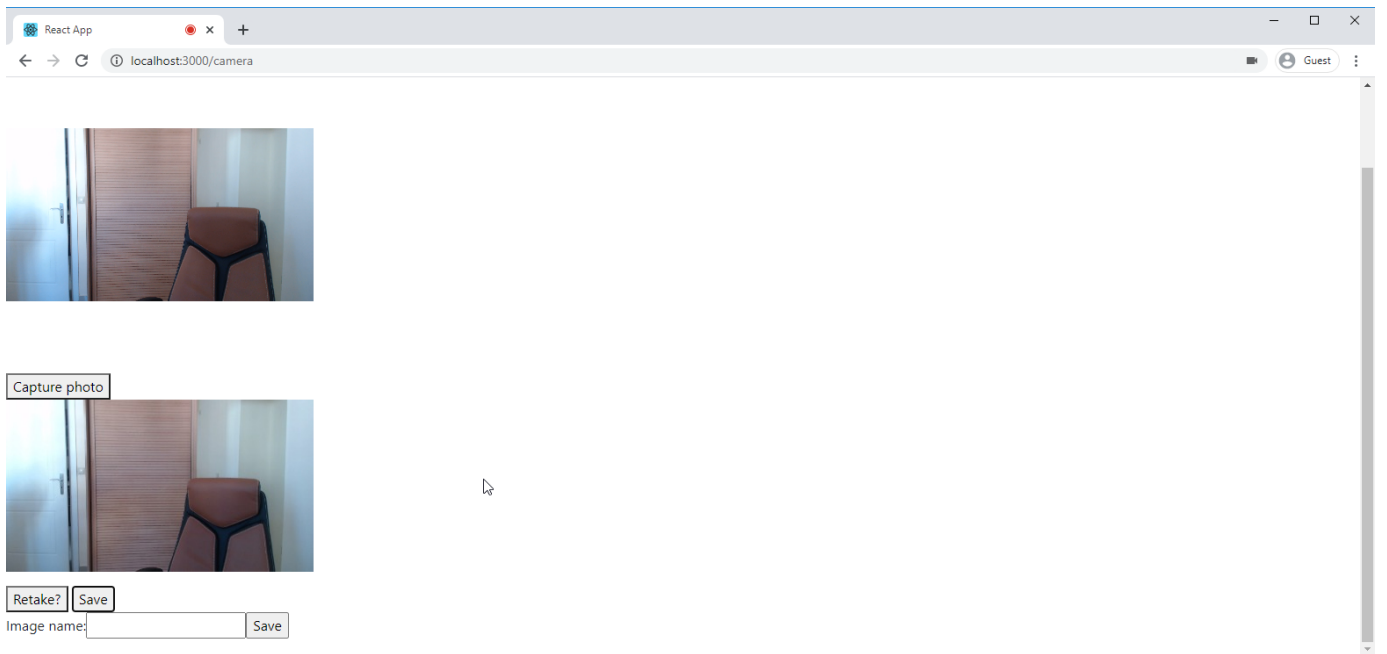


Συμπληρώνοντας την φόρμα και πατώντας το κουμπί «Sign Up» μεταβαίνουμε σε επόμενη σελίδα για λήψη φωτογραφίας.



Capture photo

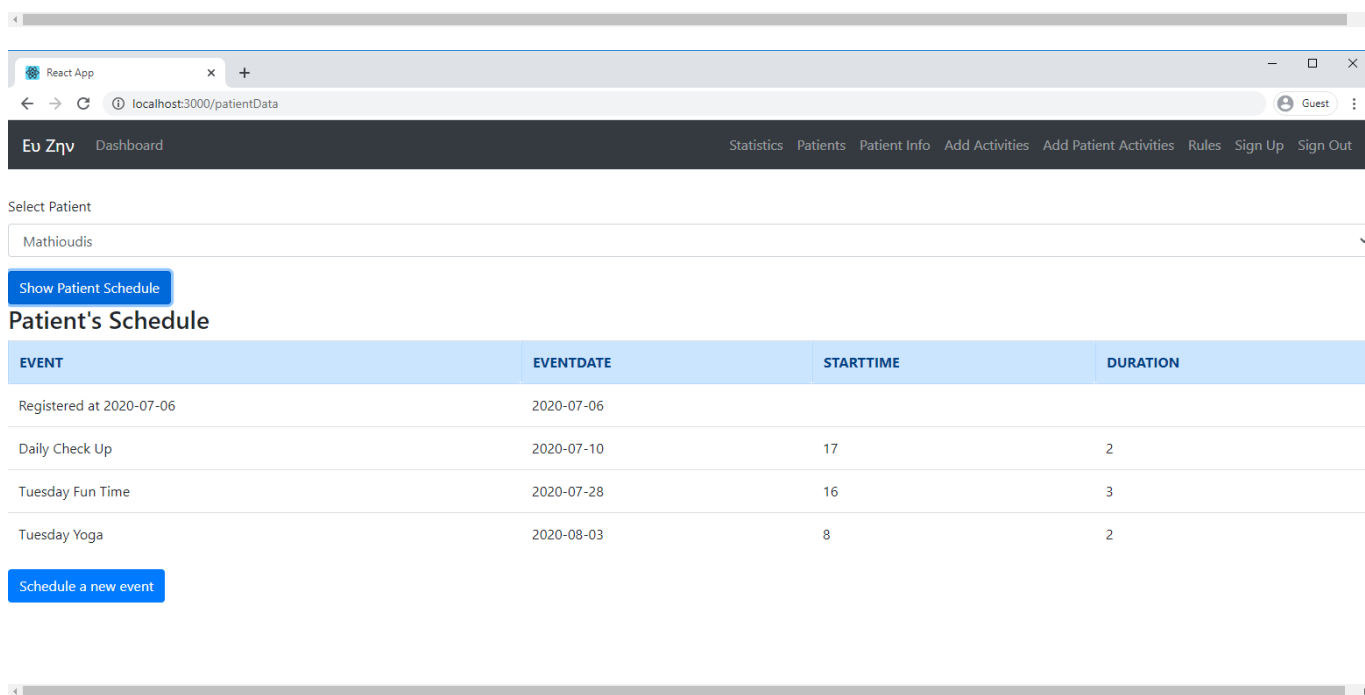
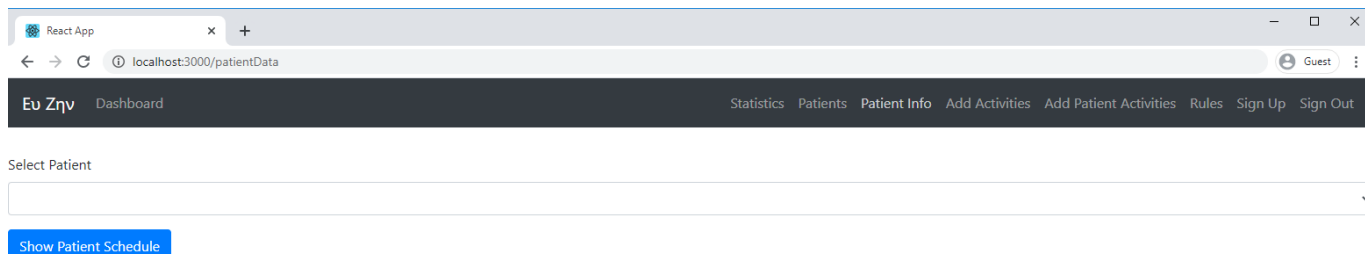
Η φωτογραφία λαμβάνεται με το πάτημα του κουμπιού Capture photo. Αν είμαστε ευχαριστημένοι πατάμε «Save». Σε διαφορετική περίπτωση «Retake?»



Ο πίνακας της καρτέλας Patients ανανεώνεται αυτόματα με τον νέο χρήστη.

11.4. Πληροφορίες Διαμένοντα

Στην καρτέλα «Patients Info» εμφανίζεται η επιλογή φιλοξενούμενου. Επιλέγοντας κάποιον υπάρχει η δυνατότητα με το κουμπί «Show Patient Schedule» να εμφανιστούν όλες οι δραστηριότητες που έχει παραγραμματισμένα να κάνει ο κάθε ένας ή που ήδη έχουν πραγματοποιηθεί.



Με το κουμπί «Schedule a new event» μπορεί να δημιουργηθεί μια καινούρια δραστηριότητα μεταβαίνοντας στην καρτέλα Add Patient Activities(Ενότητα 6).

11.5. Προσθήκη Δραστηριοτήτων

Δύο δυνατότητες υπάρχουν στην καρτέλα αυτή. Μέσω της φόρμας η δημιουργία νέων δραστηριοτήτων ενώ πιέζοντας το κουμπί «Show Activities» γίνεται προβολή όλων των διαδικασιών που υπάρχουν μέχρι σήμερα.

React App
localhost:3000/addEvent
Guest

Eu Zην Dashboard Statistics Patients Patient Info Add Activities Add Patient Activities Rules Sign Up Sign Out

Enter an activity

Enter coordinator

Add Activity Show Activities

React App
localhost:3000/addEvent
Guest

Eu Zην Dashboard Statistics Patients Patient Info Add Activities Add Patient Activities Rules Sign Up Sign Out

Enter an activity

Enter coordinator

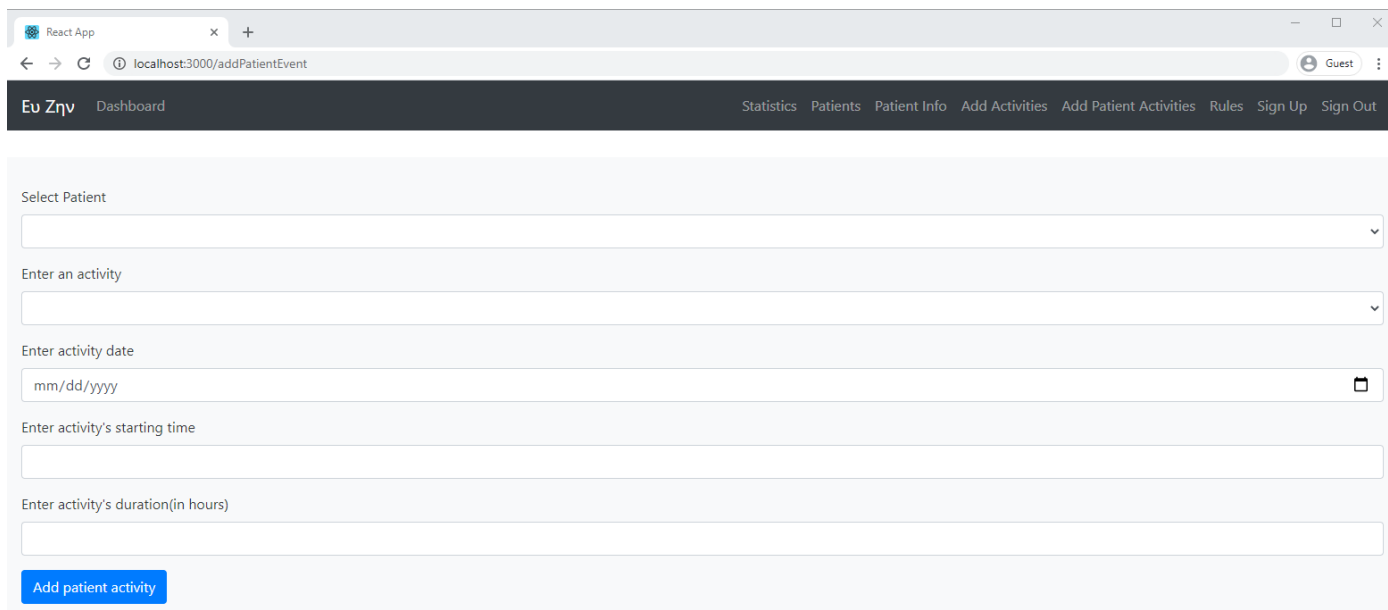
Add Activity Show Activities

Activities list

ACTIVITY	COORDINATOR
Daily Check Up	Afentoulis
Tuesday Fun Time	Asteios
Tuesday Yoga	Chrisafoudi
faf	das
Activiti 1	Demon Coordinatator

11.6. Προσθήκη Δραστηριοτήτων ανά Διαμέρισμα

Στην καρτέλα «Add Patient Activities» υπάρχει μια φόρμα συμπλήρωσης δραστηριοτήτων ανά ένοικο. Μέσω της φόρμας επιλέγεις τον ασθενή και μια από τις ήδη υπάρχουσες δραστηριότητες και πατώντας το Add Patient Activity ανανεώνεται η καρτέλα του καθενός με τα νέα δεδομένα.



The screenshot shows a web browser window with the URL `localhost:3000/addPatientEvent`. The application is running in a React App environment. The user is logged in as a Guest. The main navigation bar includes links for `Ευ Ζην`, `Dashboard`, `Statistics`, `Patients`, `Patient Info`, `Add Activities`, `Add Patient Activities`, `Rules`, `Sign Up`, and `Sign Out`.

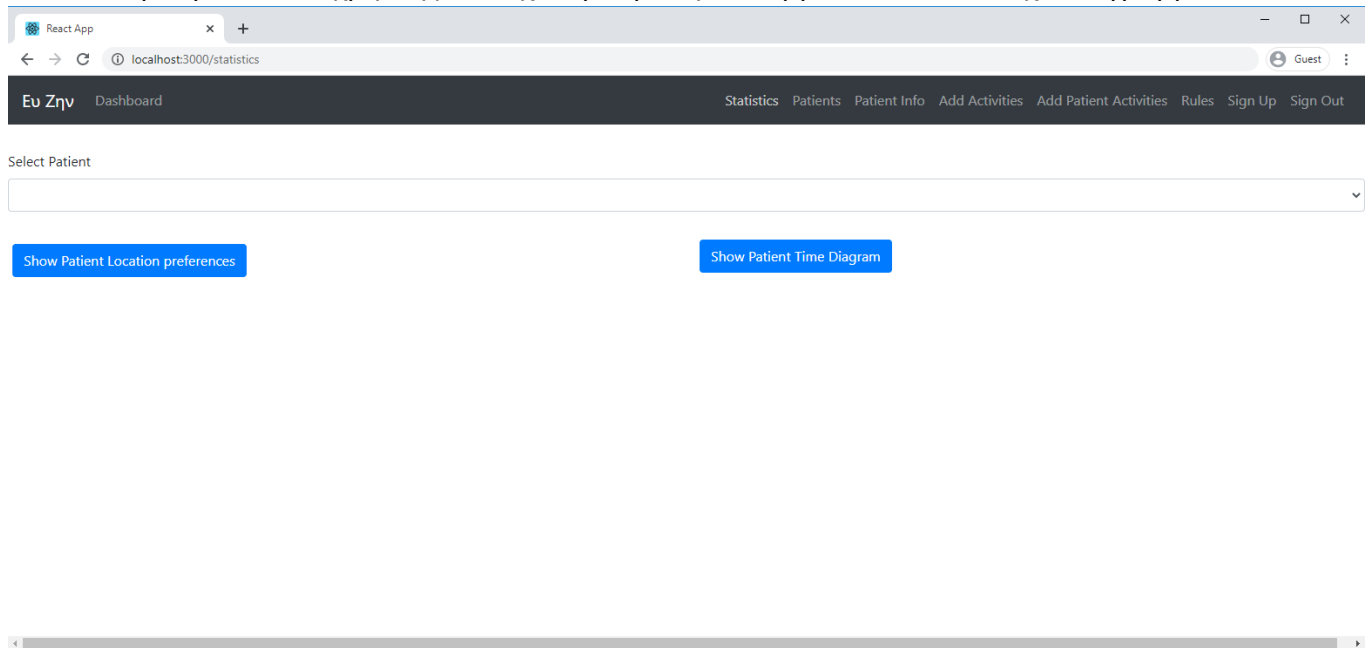
The main content area contains a form for adding a patient activity. The form fields are:

- Select Patient:** A dropdown menu.
- Enter an activity:** A dropdown menu.
- Enter activity date:** A text input field with a date picker icon and a placeholder `mm/dd/yyyy`.
- Enter activity's starting time:** A text input field.
- Enter activity's duration(in hours):** A text input field.

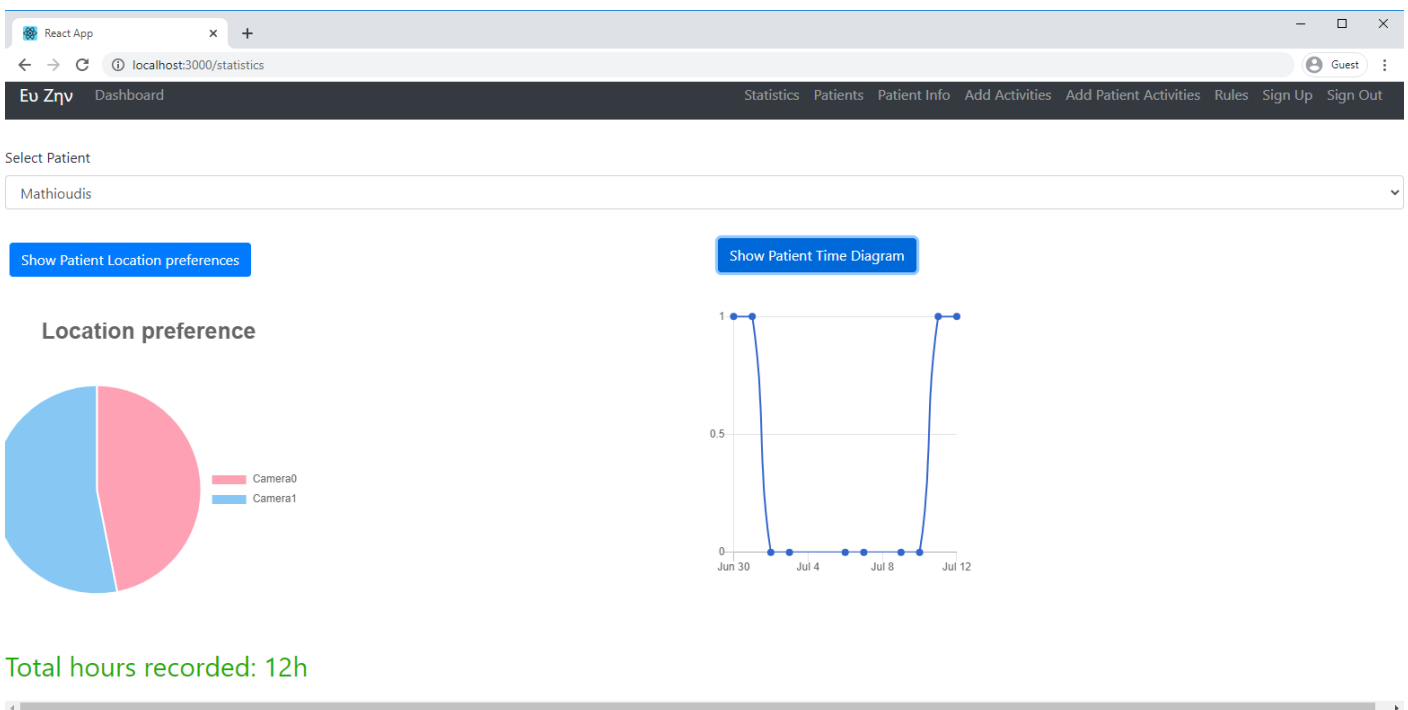
At the bottom of the form is a blue button labeled `Add patient activity`.

11.7. Στατιστικά

Η καρτέλα «Statistics» εμφανίζει όλα τα στατιστικά στοιχεία ανά ένοικο. Έτσι μπορεί ο κάθε χρήστης που έχει πρόσβαση να λαμβάνει τα αντίστοιχα διαγράμματα.



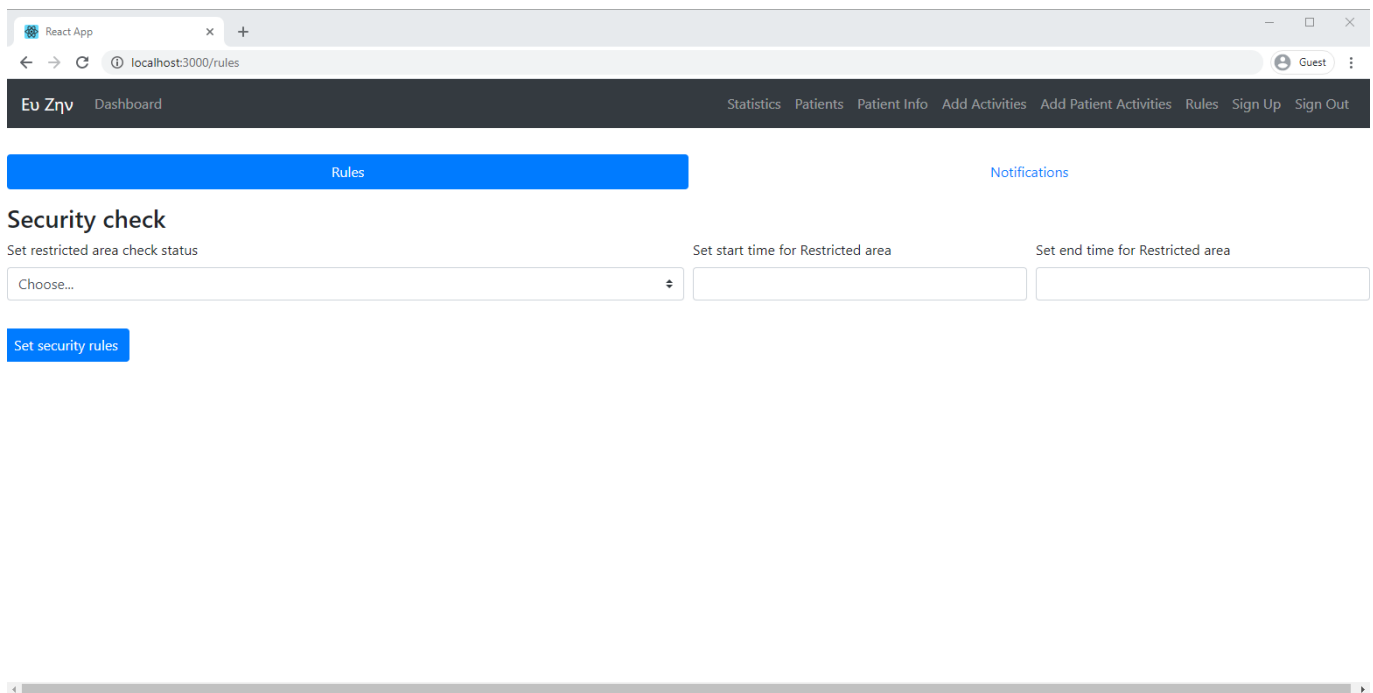
Με την επιλογή «Show Patient Location Preferences» εμφανίζεται το διάγραμμα σε μορφή πίτας για τις προτιμήσεις του κάθε ενοίκου ενώ με το κουμπί επιλογή «Show Patient Time Diagrams» εμφανίζεται η προτίμηση του κάθε χρήστη σε κάμερες ανά ημέρα.



Το κείμενο Total hours recorded αλλάζει χρώμα ανάλογα με τον αριθμό ωρών ώστε να εκφράσει την αξιοπιστία των δεδομένων.

11.8. Κανόνες

Στην καρτέλα «Rules» μπορεί ο κάθε εγκεκριμένος χρήστης να θέσει νέους κανόνες ασφαλείας.



Με την επιλογή Notifications εμφανίζονται όλες οι παραβιάσεις ασφαλείας όπως έχουν καταγραφεί.

12. ΒΙΒΛΙΟΓΡΑΦΙΑ – ΠΗΓΕΣ

1. Node.js(2020) Documentation
<https://nodejs.org/en/about/>
2. React.js(2020) Documentation
<https://reactjs.org/docs/getting-started.html>
3. face-recognition (February 2020) Documentation
<https://pypi.org/project/face-recognition/>
4. python (2020) Documentation
<https://www.python.org/doc/>
5. REST API (2020) Documentation
<https://restfulapi.net/>
6. JSON (2020) Documentation
<https://restfulapi.net/introduction-to-json/>
7. Norton face recognition, How does facial recognition work? , Joe Gervais
<https://us.norton.com/internetsecurity-iot-how-facial-recognition-software-works.html>
8. Charts.js(2020) Documentation
<https://www.chartjs.org/docs/latest/>
9. react-chart.js(2019) Documentation Github
<https://github.com/jerairrest/react-chartjs-2>
10. MongoDB (2020) The MongoDB 4.2 Manual
<https://docs.mongodb.com/manual/>
11. FACIAL RECOGNITION USE CASES(Σεπτέμβριος 2019) , Jess Davis West
<https://www.facefirst.com/blog/amazing-uses-for-face-recognition-facial-recognition-use-cases/>
12. Your Guide to Facial Recognition Technology In healthcare,(November 2019), The Medical Futurist
<https://medicalfuturist.com/your-guide-to-facial-recognition-technology-in-healthcare/>
13. Learning OpenCV: Computer Vision with the OpenCV Library, Gary Bradski and Adrian Kaehler
https://books.google.gr/books?hl=el&lr=&id=seAgiOfu2EIC&oi=fnd&pg=PR3&dq=open-cv&ots=hUM57ldBSc&sig=zjnOIULcA-IOZ58dL3UHpyQiShM&redir_esc=y#v=onepage&q=open-cv&f=false
14. Face Recognition Using Eigenfaces, Matthew A Turk and Alex P. Pentland
<https://www.cin.ufpe.br/~rps/Artigos/Face%20Recognition%20Using%20Eigenfaces.pdf>
15. Deep Learning for Computer Vision: Image Classification, Object Detection and Face Recognition in Python(2020), Jason Bownlee
https://books.google.gr/books?hl=el&lr=&id=DOamDwAAQBAJ&oi=fnd&pg=PP1&dq=python+face+recognition&ots=3qBucMCEGQ&sig=rcqReSu785dUf8dCVR5nt3353KA&redir_esc=y#v=onepage&q=python%20face%20recognition&f=false