

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

Μελέτη, Σχεδίαση και Ανάπτυξη Εφαρμογής Λογισμικού για τον Έλεγχο Εισόδου Υπαλλήλων στο Χώρο Εργασίας τους με Βάση μια Ετικέτα RFID και την Αναγνώριση του Προσώπου τους

ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΚΑΒΑΛΑ

Σεπτέμβριος 2022

Επιβλέπων Καθηγητής: **Θεόδωρος Παχίδης**, Καθηγητής ΔιΠαΕ

**Παναγιώτης Σκλίδας**

**Copyright@ 2022 Τμήμα Πληροφορικής, Διεθνές Πανεπιστήμιο της Ελλάδος,**

<https://www.cs.ihu.gr/>

Το περιεχόμενο της συγκεκριμένης Πτυχιακής Εργασίας αποτελεί πνευματική ιδιοκτησία του συγγραφέα, του επιβλέποντα καθηγητή και του τμήματος Πληροφορικής του ΔΙΠΑΕ και προστατεύεται από τους νόμους περί πνευματικής ιδιοκτησίας (Νόμος 2121/1993 και κανόνες Διεθνούς Δικαίου που ισχύουν στην Ελλάδα).

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ενυπόγραφα ότι είμαι αποκλειστικός συγγραφέας της παρούσας Πτυχιακής Εργασίας, για την ολοκλήρωση της οποίας κάθε βοήθεια είναι πλήρως αναγνωρισμένη και αναφέρεται με λεπτομέρεια στην πτυχιακή εργασία. Έχω αναφέρει πλήρως και με σαφείς αναφορές όλες τις πηγές χρήσης δεδομένων, απόψεων, θέσεων και προτάσεων, ιδεών και λεκτικών αναφορών, είτε κατά κυριολεξία, είτε βάσει επιστημονικής παράφρασης. Αναλαμβάνω την προσωπική και ατομική ευθύνη ότι σε περίπτωση αποτυχίας στην υλοποίηση των ανωτέρω δηλωθέντων στοιχείων, είμαι υπόλογος έναντι λογοκλοπής, γεγονός που σημαίνει αποτυχία στην Πτυχιακή μου Εργασία και κατά συνέπεια αποτυχία απόκτησης Τίτλου Σπουδών, πέραν των λοιπών συνεπειών του νόμου περί πνευματικών δικαιωμάτων. Δηλώνω, συνεπώς, ότι αυτή η Πτυχιακή Εργασία προετοιμάστηκε και ολοκληρώθηκε από εμένα προσωπικά και αποκλειστικά και ότι, αναλαμβάνω πλήρως όλες τις συνέπειες του νόμου στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής άλλης πνευματικής ιδιοκτησίας.

|  |  |
| --- | --- |
| **Επώνυμο** | Σκλίδας |
| **Όνομα** | Παναγιώτης |
| **Αριθμός Μητρώου** | 4360 |
| **Ημερομηνία** | 29/5/2022 |
| **Υπογραφή** |  |

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου για την καθοδήγηση και το χρόνο που διέθεσε δίνοντάς μου χρήσιμες συμβουλές και οδηγίες για την ολοκλήρωση της πτυχιακής μου εργασίας. Στο ίδιο πλαίσιο ευγνωμοσύνης, θα ήθελα να ευχαριστήσω όλους τους καθηγητές του Τμήματος Πληροφορικής για τη συμβολή τους στην επιστημονική και τεχνολογική μου κατάρτιση στα χρόνια φοίτησης μου στο τμήμα.

Ένα μεγάλο ευχαριστώ στην οικογένεια μου πρωτίστως για την ψυχολογική στήριξη αλλά και για την οικονομική τους υποστήριξη όλα αυτά τα χρόνια. Τέλος, τους φίλους για την ηθική υποστήριξη σε όλο το διάστημα των σπουδών μου. όλα αυτά

Περιεχόμενα

[Περίληψη 6](#_Toc118041849)

[Abstract 7](#_Toc118041850)

[Κεφάλαιο 1 Εισαγωγή 8](#_Toc118041851)

[1.1 Τεχνολογία RFID 8](#_Toc118041852)

[1.1.1 Τύποι RFID ετικετών 8](#_Toc118041853)

[1.1.2 Συσκευές ανάγνωσης – εγγραφής ετικετών RFID 8](#_Toc118041854)

[1.1.3 Χαρακτηρίστηκα συστημάτων RFID 9](#_Toc118041855)

[1.2 Συστήματα μηχανικής όρασης 10](#_Toc118041856)

[1.2.1 Convolutional Neural Network 10](#_Toc118041857)

[1.2.2 Χρήσεις μηχανικής όρασης 12](#_Toc118041858)

[1.2.2.1 Βιομηχανία μεταφορών 12](#_Toc118041859)

[1.2.2.2 Ιατρική 13](#_Toc118041860)

[1.2.2.3 Βιομηχανία κατασκευών 13](#_Toc118041861)

[1.3 Αναγνώριση προσώπου 14](#_Toc118041862)

[1.3.1 Αλγόριθμοι αναγνώρισης προσώπου 14](#_Toc118041863)

[1.3.1.1 Haar Cascade 15](#_Toc118041864)

[1.3.1.2 Eigenfaces 16](#_Toc118041865)

[1.3.1.3 Fisherfaces 19](#_Toc118041866)

[1.3.1.4 Local Binary Pattern (LBP) 20](#_Toc118041867)

[1.3.1.5 Αναγνώριση προσώπου με CNN 21](#_Toc118041868)

[1.3.1.6 Αναγνώριση προσώπου με ResNet 24](#_Toc118041869)

[1.4 User-friendly εφαρμογές 25](#_Toc118041870)

[1.4.1 Διεπαφή χρήστη 25](#_Toc118041871)

[1.4.2 Εμπειρία χρήστη 26](#_Toc118041872)

[1.5 Επιλογές 26](#_Toc118041873)

[Κεφάλαιο 2 Μέθοδοι, πρωτόκολλα, αλγόριθμοι, τεχνικές και βιβλιοθήκες 28](#_Toc118041874)

[2.1 Σύνδεση RFID/NFC reader με την python 28](#_Toc118041875)

[2.1.1 Reader – centric approach 29](#_Toc118041876)

[2.1.2 Answer To Reset (ATR) 30](#_Toc118041877)

[2.1.3 Card – centric approach 30](#_Toc118041878)

[2.1.4 Object – centric approach 30](#_Toc118041879)

[2.1.5 Τι χρησιμοποιήθηκε 30](#_Toc118041880)

[2.2 Αναγνώριση προσώπου (Face recognition) 31](#_Toc118041881)

[Κεφάλαιο 3 Υλικό 34](#_Toc118041882)

[3.1 Η κατασκευή 34](#_Toc118041883)

[3.2 Σύστημα ανάπτυξης 34](#_Toc118041884)

[3.3 Αναγνώστης RFID 35](#_Toc118041885)

[3.4 Εικόνες προσώπων 36](#_Toc118041886)

[Κεφάλαιο 4 Λογισμικό 38](#_Toc118041887)

[4.1 Λειτουργικές απαιτήσεις 38](#_Toc118041888)

[4.1.1 Διάγραμμα γενικού πλαισίου 38](#_Toc118041889)

[4.1.2.1 Ανάγνωση – Εγγραφή κάρτας RFID 38](#_Toc118041890)

[4.1.2.2 Αποθήκευση των στοιχείων των υπαλλήλων 38](#_Toc118041891)

[4.1.2.3 Εικόνα προσώπου των υπαλλήλων 39](#_Toc118041892)

[4.1.2.4 Εγγραφή εργαζομένου 39](#_Toc118041893)

[4.1.2.5 Ταυτοποίση εργαζομένου 39](#_Toc118041894)

[4.1.2.6 Καταγραφή στατιστικών εργαζόμενου 39](#_Toc118041895)

[4.1.2.7 Έκδοση στατιστικών στοιχείων 40](#_Toc118041896)

[4.1.2.7.1 Υπολογισμοί ανά βδομάδα 40](#_Toc118041897)

[4.1.2.7.2 Υπολογισμοί ανά μήνα 40](#_Toc118041898)

[4.1.2.7.3 Υπολογισμοί ανά χρόνο 40](#_Toc118041899)

[4.2 Προδιαγραφές 40](#_Toc118041900)

[4.2.1 Προδιαγραφή 1 40](#_Toc118041901)

[4.2.2 Προδιαγραφή 2 41](#_Toc118041902)

[4.2.3 Προδιαγραφή 3 41](#_Toc118041903)

[4.2.4 Προδιαγραφή 4 42](#_Toc118041904)

[4.2.5 Προδιαγραφή 5 42](#_Toc118041905)

[4.2.6 Προδιαγραφή 6 43](#_Toc118041906)

[4.2.7 Προδιαγραφή 7 43](#_Toc118041907)

[4.3 Διαγράμματα Ροής Δεδομένων 44](#_Toc118041908)

[4.3.1 Διάγραμμα ροής δεδομένων 1ου επιπέδου 44](#_Toc118041909)

[4.3.2 Διαγράμματα ροής δεδομένων 2ου επιπέδου 45](#_Toc118041910)

[4.3.2.1 Διάγραμμα ροής δεδομένων 2ου επιπέδου 1η διεργασία 45](#_Toc118041911)

[4.3.2.2 Διάγραμμα ροής δεδομένων 2ου επιπέδου 2η διεργασία 45](#_Toc118041912)

[4.3.2.1 Διάγραμμα ροής δεδομένων 2ου επιπέδου 3η διεργασία 46](#_Toc118041913)

[4.3.3 Διάγραμμα ροής δεδομένων 3ου επιπέδου 46](#_Toc118041914)

[4.4 Διάγραμμα οντοτήτων – συσχετίσεων (ERD) 47](#_Toc118041915)

[4.4.1 Λεξικό δεδομένων 47](#_Toc118041916)

[4.4.2 Διατύπωση επιχειρησιακών κανόνων 48](#_Toc118041917)

[4.4.3 Σχήμα σχέσεων 48](#_Toc118041918)

[4.4.4 Αντιπροσωπευτικό δείγμα 48](#_Toc118041919)

[4.5 Διαγράμματα ακολουθίας 49](#_Toc118041920)

[4.5.1 Διάγραμμα ακολουθίας για εγγραφή νέου χρήστη 49](#_Toc118041921)

[4.5.2 Διάγραμμα ακολουθίας για είσοδο χρήστη 49](#_Toc118041922)

[4.6 Μη λειτουργικές απαιτήσεις 50](#_Toc118041923)

[4.6.1 Λειτουργικό σύστημα 50](#_Toc118041924)

[4.6.2 Δυνατότητα εγκατάστασης 50](#_Toc118041925)

[4.6.3 Γλώσσα Διεπαφής 50](#_Toc118041926)

[4.6.4 Ελάχιστη συχνότητα επεξεργαστή 50](#_Toc118041927)

[4.6.5 Ελάχιστη απαιτούμενη μνήμη RAM 50](#_Toc118041928)

[4.6.6 Γλώσσα προγραμματισμού 51](#_Toc118041929)

[4.6.7 Βάση δεδομένων 51](#_Toc118041930)

[4.6.8 Προστασία προσωπικών δεδομένων 51](#_Toc118041931)

[4.6.9 Συντηρησημότητα 51](#_Toc118041932)

[4.7 Code snippets 52](#_Toc118041933)

[Κεφάλαιο 5 Πειραματικά αποτελέσματα 56](#_Toc118041934)

[5.1 Αναγνώριση έξυπνων καρτών 56](#_Toc118041935)

[5.2 Αναγνώριση προσώπου 56](#_Toc118041936)

[Κεφάλαιο 6 Συμπεράσματα 58](#_Toc118041937)

[Παράρτημα 59](#_Toc118041938)

[Παράρτημα 1 Script άντλησης εικόνων προσώπου 59](#_Toc118041939)

[Παράρτημα 2 User manual 60](#_Toc118041940)

[1 Εισαγωγή 60](#_Toc118041941)

[2 Εγκατάσταση 61](#_Toc118041942)

[3 Απεγκατάσταση 62](#_Toc118041943)

[4 Πως να χρησιμοποιήσετε την εφαρμογή 63](#_Toc118041944)

[4.1 Άνοιγμα εφαρμογής 63](#_Toc118041945)

[4.2 Εγγραφή νέου χρήστη 64](#_Toc118041946)

[4.3 Είσοδος στον χώρο εργασίας 69](#_Toc118041947)

[4.4 Έξοδος από τον χώρο εργασίας 71](#_Toc118041948)

[4.5 Admin panel 72](#_Toc118041949)

[4.5.1 Human Resources 73](#_Toc118041950)

[4.5.2 Sectors 75](#_Toc118041951)

[4.6 Συχνές ερωτήσεις 77](#_Toc118041952)

[Παράρτημα 3 Άδειες χρήσης βιβλιοθηκών 78](#_Toc118041953)

[3.1 Pyscard[46] 78](#_Toc118041954)

[3.2 Face\_Recognition[47] 79](#_Toc118041955)

[Βιβλιογραφία 85](#_Toc118041956)

# Περίληψη

Στο πλαίσιο της παρούσας πτυχιακής εργασίας έγινε βιβλιογραφική έρευνα κατά την οποία βρέθηκαν οι διάφοροι τύποι καρτών και αναγνωστών RFID. Επιπλέον, μελετήθηκαν οι πιο γνωστοί αλγόριθμοι αναγνώρισης προσώπου όπως είναι για παράδειγμα η αναγνώριση προσώπου μέσω νευρωνικών δικτυών ή μέσω τοπικού δυαδικού μοτίβου (Local Binary Pattern). Τέλος, αναπτύχθηκε στην python μια desktop εφαρμογή για Windows στην οποία μπορεί να εγγραφεί ένας χρήστης/υπάλληλος και στην συνέχεια χρησιμοποιώντας την έξυπνη κάρτα και με πιστοποίηση μέσω του προσώπου του έχει την δυνατότητα να καταγράψει την ώρα εισόδου και εξόδου του από τους χώρους εργασίας. Ορισμένοι χρήστες είναι σε θέση να δουν στατιστικά στοιχεία όπως είναι ο συνολικός χρόνος παραμονής στους χώρους εργασίας ανά βδομάδα και ανά έτος, καθώς επίσης και την δυνατότητα να επεξεργαστούν (δημιουργήσουν, τροποποιήσουν, διαγράψουν) χρήστες και χώρους εργασίας.

# Abstract

In the context of this thesis, bibliographic research was carried out in which various types of RFID cards and readers were found. In addition, the most well-known face recognition algorithms were studied, such as facial recognition through neural networks or a local binary pattern (Local Binary Pattern). Finally, a desktop application for Windows was developed in python in which a user/employee can register, and then by using a smart card, and with authentication, through his face, he can record his time of entry and exit from the workplace. Some users can see statistics such as total time spent in workplaces per week and year, as well as the ability to edit (create, modify, delete) users and spaces.

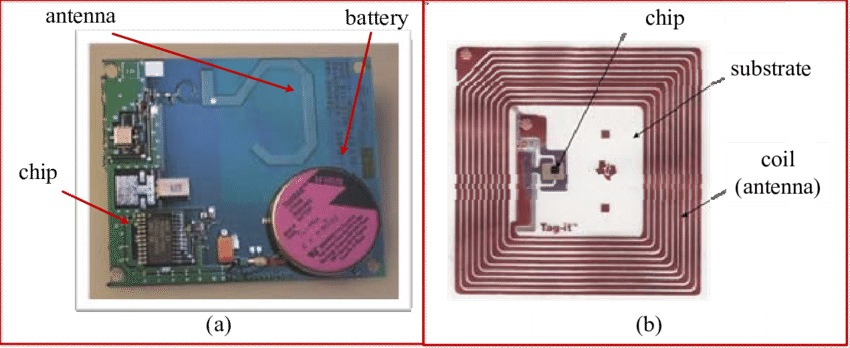
# Κεφάλαιο 1 Εισαγωγή

## 1.1 Τεχνολογία RFID

Η ταυτοποίηση μέσω ραδιοσυχνοτήτων (Radio Frequency Identification - RFID) είναι ένας τύπος ασύρματης επικοινωνίας, ο πρόγονος της οποίας παρουσιάστηκε τον Ιανουάριο του 1973 από τον Mario Cardullo[1,2], λειτουργούσε είτε με ηλεκτρομαγνητική είτε με ηλεκτροστατική σύζευξη στο τμήμα των ραδιοσυχνοτήτων ηλεκτρομαγνητικού φάσματος και χρησιμοποιείται για την αναγνώριση αντικειμένων, ζώων ή ανθρώπων. Η αναγνώριση πραγματοποιείται με την βοήθεια εξειδικευμένων μικροεπεξεργαστών, γνωστών και ως tags, οι οποίοι βρίσκονται ενσωματωμένοι είτε σε κάποια αυτοκόλλητη ετικέτα πάνω σε κάποιο αντικείμενο ή κάποια έξυπνη κάρτα την οποία έχει στην διάθεση του ένας άνθρωπος είτε είναι εμφυτευμένοι σε κάποιο ζώο.

### 1.1.1 Τύποι RFID ετικετών

Οι RFID ετικέτες μπορεί να είναι είτε ενεργητικές (active) είτε παθητικές (passive). Οι ενεργητικές κάρτες πέρα από τον μικροεπεξεργαστή διαθέτουν και μια μπαταρία η οποία παρέχει συνεχώς ρεύμα στον μικροεπεξεργαστή δίνοντας έτσι την ικανότητα στις ενεργητικές ετικέτες να εκπέμπουν το σήμα τους συνεχώς. Αντίθετα, οι παθητικές ετικέτες όπως δηλώνει και το όνομα τους, δεν διαθέτουν μπαταρία και αντλούν την απαραίτητη ενέργεια που χρειάζονται για να μεταδώσουν το σήμα από τον αναγνώστη. Έτσι οι παθητικές ετικέτες μεταδίδουν το σήμα μόνο όταν βρίσκονται αρκετά κοντά στην συσκευή ανάγνωσης. Επιπλέον, οι ενεργητικές ετικέτες έχουν την δυνατότητα να μεταδίδουν σε πολύ μεγάλες αποστάσεις (περίπου 150m για τις Ultra High Frequency), ενώ οι παθητικές μπορούν να εκπέμπουν σε σχετικά μικρές αποστάσεις (περίπου 15m για τις Ultra High Frequency)[3]. Τέλος, οι ετικέτες μπορεί να είναι ή μόνο για ανάγνωση (read only tags) ή ανάγνωσης και εγγραφής (read/write tags).



Σχήμα 1.1: (a) Ενεργητικές και (b) παθητικές RFID ετικέτες

### 1.1.2 Συσκευές ανάγνωσης – εγγραφής ετικετών RFID

Για να μπορέσουν να χρησιμοποιηθούν οι ετικέτες πρέπει να έχουμε στην διάθεση μας μια συσκευή εγγραφής – ανάγνωσης RFID (RFID reader/writer). Στο εμπόριο υπάρχουν αρκετά είδη συσκευών ανάγνωσης όπως για παράδειγμα ο RC522 RFID Reader/Writer o οποίος είναι ένα πρόσθετο (module) για το Arduino και για το Raspberry Pi. Στην ίδια κατηγόρια ανήκει και ο PN532 NFC RFID Module Reader Writer[4]. Πέρα από τα modules για μικροεπεξεργαστές τα οποία πρέπει ο χρήστης να συνδέσει με το board της επιλογής του με καλώδια και να τα προγραμματίσει ώστε να εξυπηρετούν καλύτερα τις ανάγκες του, υπάρχουν και έτοιμες λύσεις για αναγνώστες και εγγραφείς στο εμπόριο. Οι αναγνώστες RFID χειρός (Handheld RFID scanner)[5] είναι έτοιμες συσκευές που περιέχουν όλες τις απαραίτητες λειτουργίες που χρειάζεται ένας χρήστης για να μπορεί να γράφει και να κάνει ανάγνωση των πληροφοριών που είναι αποθηκευμένες στις ετικέτες. Επιπλέον, με το συγκεκριμένο είδος αναγνώστη ο χρήστης έχει την δυνατότητα να έχει πάνω του την συσκευή μπορώντας έτσι να τη χρησιμοποιεί όπου και αν βρίσκεται. Τέλος, υπάρχουν USB αναγνώστες – εγγραφείς RFID[6] οι οποίοι μπορούν να συνδεθούν με τον υπολογιστή του χρήστη και αφού αυτός κατεβάσει το αντίστοιχο πρόγραμμα μπορεί να διαβάσει και να γράψει στα RFID tags.

|  |  |
| --- | --- |
|  |  |
| (A) | (B) |
|  |  |
| (C) | (D) |

Σχήμα 1.2: (A) RC522 RFID reader/writer. (B) PN532 NFC – RFID reader/writer. (C) Handheld RFID scanner. (D) USB desktop RFID scanner.

### 1.1.3 Χαρακτηρίστηκα συστημάτων RFID

Η τεχνολογία RFID χρησιμοποιήθηκε ως συσκευή διοδίων από τον Mario Cardullo το 1973 μέχρι να καταχωρηθεί ως πατέντα το 1983[2] από τότε μέχρι και σήμερα η συγκεκριμένη τεχνολογία συνεχώς εξελίσσεται. Η βασική χρήση της είναι η αποθήκευση και η μετάδοση της αποθηκευμένης πληροφορία. Η κάθε ετικέτα έχει μοναδικό αναγνωριστικό (unique id) καθιστώντας δυνατή την αναγνώριση εξαρτημάτων, υλικών, εμπορευμάτων λιανικής, ζώων συντροφιάς ή άλλων ειδών. Τα συστήματα υπολογιστών μπορούν να χρησιμοποιήσουν το αναγνωριστικό για να ενημερώσουν αυτόματα τις εγγραφές σε συνδυασμό με άλλα δεδομένα όπως τοποθεσία, θερμοκρασία, ημερομηνία και ώρα. Η αυτόματη αναγνώριση είναι ένα ακόμα χαρακτηριστικό το οποίο βοήθησε στη διάδοση της τεχνολογίας αυτής, καθώς είναι δυνατόν οι ετικέτες να επισυνάπτονται σε αντικείμενα και να μεταδίδουν το αποθηκευμένο αναγνωριστικό στους αναγνώστες. Οι ετικέτες μπορούν να τοποθετηθούν σε σταθερά σημεία εσωτερικά ή εξωτερικά των εμπορευμάτων, ενώ οι αναγνώστες μπορούν να στηθούν σε συγκεκριμένες τοποθεσίες εντός μιας μονάδας επεξεργασίας ή μιας αποθήκης, με τον τρόπο αυτό κάθε φορά που κάποιο στοιχείο περνάει, ο αναγνώστης θα σαρώνει αυτόματα το αναγνωριστικό. Εναλλακτικά, οι αναγνώστες μπορούν να ενσωματωθούν σε κινητά τηλέφωνα, επιτρέποντας τη σάρωση των ετικετών όπου και αν βρίσκεται η οντότητα. Επιπρόσθετα, αρχεία σε ράφια ή αντικείμενα που κινούνται σε μεταφορικούς ιμάντες, μπορούν να εντοπιστούν απλά τοποθετώντας σωστά τη συσκευή ανάγνωσης. Ωστόσο, η ικανότητα ανάγνωσης μιας ετικέτας εξαρτάται από έναν συνδυασμό παραγόντων όπως ο τύπος της ετικέτας και ο αναγνώστης, το υλικό της επιφάνειας και το περιβάλλον. Ένας αναγνώστης RFID μπορεί να συλλέξει δεδομένα από μεγάλο αριθμό ετικετών ταυτόχρονα, με αποτέλεσμα την εξοικονόμηση χρόνου για παράδειγμα το περιεχόμενο μιας ολόκληρης παλέτας μπορεί να σαρωθεί καθώς εξέρχεται από την αποθήκη. Οι παθητικές ετικέτες RFID, όπως έχει προαναφερθεί ενεργοποιούνται από την πηγή ισχύος της συσκευής ανάγνωσης, ώστε να μην απαιτούν μπαταρίες για την χρήση τους. Αυτό σημαίνει ότι οι ετικέτες μπορεί να είναι πολύ μικρές ώστε να μπορούν να ενσωματωθούν σε μικροσκοπικές συσκευές. Έχουν επίσης πολύ μεγάλη διάρκεια ζωής και μερικές μπορούν να επιβιώσουν σε εξωτερικούς χώρους σε όλες τις συνθήκες για αρκετά χρόνια. Συνήθως, οι παθητικές ετικέτες χρησιμοποιούνται για στοιχεία που πρέπει να διαβαστούν σε εύρος από 1mm έως περίπου 10m. Για μεγαλύτερες αποστάσεις ή πιο εξειδικευμένες εφαρμογές, μπορούν να χρησιμοποιηθούν ενεργές ετικέτες με τη δική τους πηγή ενέργειας, οι οποίες μπορούν να παραμείνουν ενεργές για έως και πέντε χρόνια ανάλογα με τη χρήση και τις περιβαλλοντικές συνθήκες. Τέλος, ένα ακόμα καθοριστικό χαρακτηριστικό το οποίο βοήθησε την τεχνολογία RFID να εδραιωθεί είναι η ανθεκτικότητα των ετικετών στις καιρικές συνθήκες.Αντίθετα με τα barcodes (γραμμωτοί κώδικες) οι ετικέτες RFID μπορούν να χρησιμοποιηθούν ακόμη και στα πιο απαιτητικά περιβάλλοντα[7].

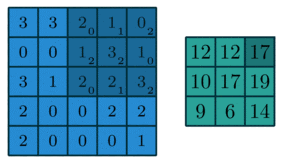
## 1.2 Συστήματα μηχανικής όρασης

Η μηχανική όραση είναι ο τομέας εκείνος της τεχνητής νοημοσύνης (AI) που καθιστά εφικτό σε υπολογιστές και συστήματα να αντλούν πληροφορίες από ψηφιακές εικόνες, βίντεο και άλλες οπτικές αναπαραστάσεις και να προβαίνουν σε ενέργειες με βάση τις πληροφορίες αυτές[8]. Όπως κάθε εφαρμογή που στηρίζεται στην AI για να λειτουργήσει αποδοτικά χρειάζεται να έχει στην διάθεση της αρκετά δεδομένα ώστε να κάνει πολλές αναλύσεις των δεδομένων αυτών μέχρι να διακρίνει τις διαφοροποιήσεις και τελικά να πάρει κάποια απόφαση, έτσι και στην μηχανική όραση χρειαζόμαστε έναν τεράστιο όγκο δεδομένων.

### 1.2.1 Convolutional Neural Network

Οι περισσότεροι αλγόριθμοι μηχανικής όρασης για να υλοποιηθούν χρησιμοποιούν συνελεκτικά νευρωνικά δίκτυα ή αλλιώς γνωστά ως CNNs. Τα CNNs, όπως και τα νευρωνικά δίκτυα ανατροφοδότησης (feedforward neural networks), μαθαίνουν από τις εισόδους, προσαρμόζοντας τις παραμέτρους για να κάνουν μια επιτυχή πρόβλεψη. Το χαρακτηριστικό που καθιστά τα CNNs ξεχωριστά είναι η ικανότητα τους να αντλούν χαρακτηριστικά από εικόνες, για να το πετύχουν αυτό χρησιμοποιούν συνελικτικά στρώματα (convolutional layers) και ομαδοποίηση (pooling).

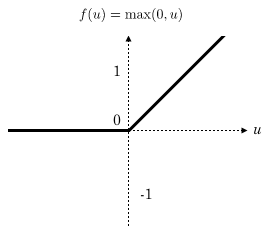
Τα convolutional layers εφαρμόζουν μια σειρά από φίλτρα εικόνας στην εικόνα που έχουν λάβει ως είσοδο, η οποία παριστάνεται με τη μορφή ενός πίνακας (matrix). Οι τελικές εικόνες που θα προκύψουν, παρουσιάζουν διαφορετικές εκδοχές της αρχικής εικόνας καθώς έχουν εξαχθεί διαφορετικά χαρακτηριστικά. Τα φίλτρα εικόνας που χρησιμοποιούνται λέγονται πυρήνες συνέλιξης (convolutional kernels).



Σχήμα 1.3: Πυρήνας συνέλιξης

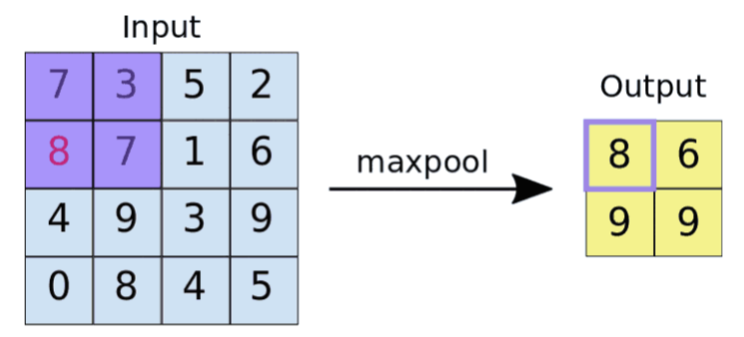
Στο σχήμα 1.3 βλέπουμε πως λειτουργεί ο πυρήνας συνέλιξης. Το μπλε τετράγωνο αναπαριστά την εικόνα εισόδου, το σκιασμένο πλαίσιο αποτελεί τον πυρήνα συνέλιξης, ο οποίος είναι ένα σύνολο από βάρη που “περνά” πάνω από την εικόνα. Κάθε τιμή του μπλε πίνακα πολλαπλασιάζεται με το αντίστοιχο βάρος και έπειτα όλα τα νούμερα στον σκιασμένο χώρο προστίθενται μεταξύ τους ώστε να προκύψει ο πράσινος πίνακας, ο οποίος ονομάζεται χάρτης χαρακτηριστικών (feature map)[8]. Στη συγκεκριμένη περίπτωση μετά τον πολλαπλασιασμό θα προκύψουν τα νούμερα 0,1,0,2,6,0,0,2,6 που όταν αθροιστούν θα δώσουν τον αριθμό 17.

Συνήθως τα συνελικτικά επίπεδα έχουν πολλαπλούς συνελικτικούς πυρήνες με διαφορετικά βάρη ώστε να είναι σε θέση να δημιουργήσουν διαφορετικά feature maps. Έπειτα τα convolutional layers ακολουθούνται από μια συνάρτηση ενεργοποίησης (activation function) συνήθως αυτή η συνάρτηση είναι η ReLU. Η συνάρτηση ενεργοποίησης εισάγεται με σκοπό να δώσει στο νευρωνικό δίκτυο την ικανότητα να παρουσιάσει τα αποτελέσματα με μη γραμμική έκφραση, ώστε να είναι σε θέση συγκρίνοντας τα αποτελέσματα που προκύπτουν να επιλέξει το καλύτερο, βελτιώνοντας έτσι την ακρίβεια του[9].



Σχήμα 1.4: Η συνάρτηση ενεργοποίησης ReLU

Αν χρησιμοποιηθούν πολλοί πυρήνες συνέλιξης τα δεδομένα θα έχουν πολλές διαστάσεις (dimensions) οδηγώντας το νευρωνικό σε overfitting, δηλαδή το νευρωνικό θα τείνει να μην μπορεί να γενικεύσει. Με άλλα λόγια το δίκτυο θα έχει πάρα πολύ καλή απόδοση στα δεδομένα εκπαίδευσης αλλά χαμηλή απόδοση στα δεδομένα ελέγχου ή σε πραγματικά δεδομένα. Για να αντιμετωπιστεί αυτό, χρησιμοποιείται ομαδοποίηση. Η πιο συχνή μέθοδος ομαδοποιήσης που χρησιμοποιείται είναι αυτή της μέγιστης ομαδοποίησης (max pooling) η οποία, όπως φαίνεται και στο σχήμα 1.5, επιλέγει την μεγαλύτερη τιμή από μια γειτονία τιμών.



Σχήμα 1.5: Maxpool

### 1.2.2 Χρήσεις μηχανικής όρασης

Η μηχανική όραση επιτρέπει στις μηχανές να προσομοιάσουν την ανθρώπινη αίσθηση της όρασης. Η τεχνολογία αυτή έχει βρει αρκετές εφαρμογές σε όλες τις βιομηχανίες όπως είναι για παράδειγμα η βιομηχανία μεταφορών, η βιομηχανία παραγωγής γεωργικών και κτηνοτροφικών προϊόντων και άλλες[10].

#### 1.2.2.1 Βιομηχανία μεταφορών

Βρισκόμενοι στο έτος 2022 αρκετά αυτοκίνητα έχουν κάποιου είδους αυτόνομη οδήγηση, κάποια χρησιμοποιώντας sonar είναι σε θέση να μεταβάλουν μόνα τους την ταχύτητα χωρίς ανθρώπινη παρέμβαση, άλλα διαθέτοντας κάμερες μπορούν να αναγνωρίσουν και να εμφανίσουν σε μια δευτερεύουσα οθόνη πληροφορίες όπως για παράδειγμα οδικά σήματα και να προειδοποιήσουν τους οδηγούς για πεζούς ή άλλα οχήματα που παρεκκλίνουν της πορείας τους κάνοντας την οδήγηση πιο ασφαλή και εύκολη.



Σχήμα 1.6: Μηχανική όραση στην αυτόνομη οδήγηση (Jeremy Cohen 2019)[11]

#### 1.2.2.2 Ιατρική

H μηχανική όραση βρήκε χρησιμότητα και στην ιατρική καθώς χωρίς την τεχνολογία αυτή οι γιατροί θα αναγκάζονταν να αφιερώνουν αρκετές ώρες στην ανάλυση δεδομένων που σχετίζονται με τους ασθενείς και στην διεκπεραίωση διοικητικών εργασιών. Πλέον αρκετά διαγνωστικά κέντρα και νοσοκομεία χρησιμοποιούν την τεχνολογία αυτή για να κάνουν μια πρώτη πρόβλεψη για την υγεία τον ασθενών σε εξετάσεις με οπτικές απεικονίσεις όπως στις αξονικές και στις ακτινογραφίες. Αφού τα μηχανήματα λάβουν την εικόνα του ασθενή την επεξεργάζονται με αλγόριθμους αναγνώρισης εικόνας (image recognition), οι οποίοι ως έξοδο θα δώσουν την αρχική εικόνα με κάποιες “σημειώσεις” πάνω τους, τις οποίες βλέπει ο θεράπων ιατρός και αναλόγως αποφασίζει για την διενέργεια συμπληρωματικών εξετάσεων ή την χορήγηση κάποιας θεραπείας στον ασθενή.

#### 1.2.2.3 Βιομηχανία κατασκευών

O κατασκευαστικός τομέας υιοθετεί με γοργό ρυθμό την τεχνολογία της μηχανικής όρασης αξιοποιώντας την για επιθεώρηση περιουσιακών στοιχείων υποδομής, για πρόληψη κινδύνου στον χώρο εργασίας ή για προγνωστική συντήρηση. Στο περιβάλλον των κατασκευών είναι σύνηθες τα υλικά να διαβρώνονται και να προκαλείται παραμόρφωση θέτοντας σε κίνδυνο την ζωή των εργατών και οδηγώντας το έργο σε μεγάλες καθυστερήσεις. Οι συσκευές υπολογιστικής όρασης παρακολουθούν τα εισερχόμενα δεδομένα από τα μηχανήματα μέσω καμερών και εντοπίζουν ελαττώματα και άλλες αλλαγές. Όταν εντοπίζουν ένα πρόβλημα, στέλνουν ένα σήμα στο σύστημα, επιτρέποντας στους ανθρώπους χειριστές να λάβουν διορθωτικά μέτρα πριν συμβεί κάποιο ατύχημα ή καταστραφεί ένα περιουσιακό στοιχείο.



Σχήμα 1.7: Μηχανική όραση στην βιομηχανία κατασκευών

## 1.3 Αναγνώριση προσώπου

Το λογισμικό το οποίο χαρτογραφεί, αναλύει και επιβεβαιώνει την ταυτότητα ενός ατόμου από το πρόσωπο του είτε από φωτογραφία είτε από video καλείται λογισμικό αναγνώρισης προσώπου[12]. Το λογισμικό αναγνώρισης προσώπου είναι μια εφαρμογή της μηχανικής όρασης που χρησιμοποιείται από τις αρχές επιβολής του νόμου και τις κυβερνήσεις[13] για να αναγνωριστούν εγκληματίες, αλλά και από ιδιωτικές εταιρίες οι οποίες θέλουν να προσφέρουν στους χρήστες τους ένα επιπλέον επίπεδο ασφαλείας στις συσκευές τους όπως είναι η Apple με το FaceID και η Microsoft με το Windows Hello ή εταιρίες και οργανισμούς που θέλουν να προστατέψουν της φυσικές τους εγκαταστάσεις ή μέρη αυτών από μη εξουσιοδοτημένη πρόσβαση.

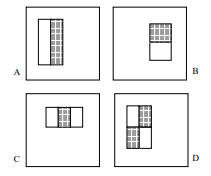
### 1.3.1 Αλγόριθμοι αναγνώρισης προσώπου

Ένας αλγόριθμος αναγνώρισης προσώπου είναι το κομμάτι ενός οποιουδήποτε συστήματος ή λογισμικού το οποίο είναι υπεύθυνο με τις λειτουργίες της ανίχνευσης ή της αναγνώρισης ενός προσώπου[14]. Οι ειδικοί χωρίζουν τους αλγορίθμους σε δύο κατηγορίες: α) τους γεωμετρικούς αλγορίθμους οι οποίοι προσεγγίζουν την αναγνώριση χρησιμοποιώντας διακριτικά χαρακτηριστικά και β) στις φωτομετρικές στατιστικές μεθόδους οι οποίες εξάγουν τιμές από την εικόνα εισόδου, τις οποίες συγκρίνονται με πρότυπα για την εξάλειψη των αποκλίσεων. Οι αλγόριθμοι εκτελούν τρεις βασικές εργασίες: α) ανιχνεύουν τα πρόσωπα σε φωτογραφία ή σε video πραγματικού χρόνου μέσω καμερών, β) υπολογίζουν το μαθηματικό μοντέλο του προσώπου και γ) συγκρίνουν το μοντέλο που προέκυψε με τα δεδομένα εκπαίδευσης ή με βάσεις δεδομένων ώστε να γίνει η αναγνώριση και η επαλήθευση ενός ατόμου.

#### 1.3.1.1 Haar Cascade

O αλγόριθμος Haar Cascade είναι ένας αλγόριθμός αναγνώρισης αντικειμένων και χρησιμοποιείται για να ταυτοποιήσει αντικείμενα και πρόσωπα σε μια εικόνα ή σε ένα video πραγματικού χρόνου. Προτάθηκε από τους Paul Viola και Michael Jones στην ερευνητική εργασία τους “Rapid Object Detection using a Boosted Cascade of Simple Features”[15] που δημοσιεύθηκε το 2001. H υλοποίηση αυτή βασίζεται στην μηχανική εκμάθηση όπου μια συνάρτηση cascade (καταρράκτη) εκπαιδεύεται από πολλές θετικές (εικόνες που έχουν το αντικείμενο) και αρνητικές εικόνες (όπου δεν υπάρχει το αντικείμενο), και έπειτα χρησιμοποιείται για την ανίχνευση σε άλλες εικόνες.

Ο συγκεκριμένος αλγόριθμος χρειάζεται αρκετές θετικές και αρνητικές εικόνες για να εκπαιδεύσει τον ταξινομητή. Για να εξαχθούν τα χαρακτηριστικά χρησιμοποιούνται τα χαρακτηριστικά Haar (Σχήμα1.8). Τα χαρακτηριστικά αυτά μοιάζουν στην λειτουργία τους με τους convolutional kernels. Κάθε χαρακτηριστικό προκύπτει από την αφαίρεση του αθροίσματος των pixels που βρίσκονται στη λευκή περιοχή από το άθροισμα αυτών που βρίσκονται στην σκιασμένη[16].



Σχήμα 1.8: Χαρακτηριστικά Haar. A-B) Ανίχνευση ακμών. C) Ανίχνευση γραμμών. D) Ένα τεσσάρων-ορθογωνίων χαρακτηριστικό

Για το κάθε πιθανό μέγεθος και πιθανή θέση του πυρήνα συνέλιξης θα υπολογιστεί ένα χαρακτηριστικό, με αποτέλεσμα στο τέλος της διαδικασίας ακόμα και για μια αρκετά μικρών διαστάσεων εικόνα να έχουν παραχθεί υπερβολικά πολλά χαρακτηριστικά. Σύμφωνα με τους ερευνητές μια εικόνα με διαστάσεις 24x24 θα μπορούσε να δώσει μέχρι και πάνω από 180.000 χαρακτηριστικά. Με σκοπό να περιορίσουν τα χαρακτηριστικά, εισήγαγαν την έννοια της ολοκληρωμένης εικόνας (integral image). Μια ολοκληρωμένη εικόνα στην τοποθεσία περιέχει το άθροισμα των pixels που υπάρχουν πάνω από την τοποθεσία: όπου είναι η ολοκληρωμένη εικόνα και είναι η αρχική εικόνα. Κάνοντας χρήση του ακόλουθου ζεύγους συναρτήσεων η integral image μπορεί να υπολογισθεί σε ένα πέρασμα της αρχικής εικόνας.

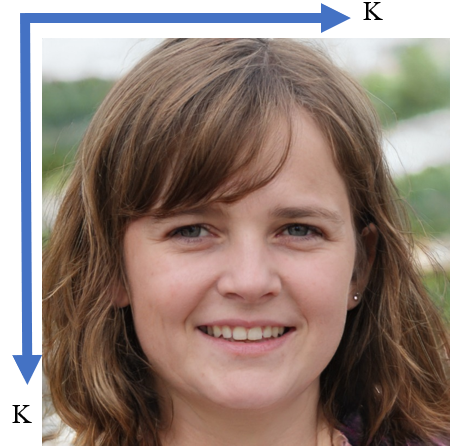
Έπειτα μια παραλλαγή του AdaBoost χρησιμοποιείται για να περιορίσει τα χαρακτηριστικά που εξάγονται και να εκπαιδεύσει τον ταξινομητή. Για κάθε χαρακτηριστικό βρίσκεται το καλύτερο κατώτερο όριο (threshold) το οποίο θα ταξινομήσει τις εικόνες σε θετικές ή αρνητικές. Στην συνέχεια, επιλέγονται τα χαρακτηριστικά εκείνα που έχουν το χαμηλότερο δείκτη λάθους (minimum error rate), καθώς όταν αυτό ισχύει σημαίνει πως το χαρακτηριστικό αυτό είναι κατάλληλο για χρήση στην ταξινόμηση. Ο τελικός ταξινομητής προκύπτει από το άθροισμα (weighted sum) των ασθενών επιμέρους ταξινομητών. Καλούνται ασθενείς διότι δεν μπορούν να ταξινομήσουν από μόνοι τους μια εικόνα. Σύμφωνα με τους συγγραφείς ακόμα και ένας μικρός όγκος χαρακτηριστικών της τάξης κάποιων εκατοντάδων είναι σε θέση να παρέχουν ακρίβεια στην ταξινόμηση περίπου 95%.

Το μεγαλύτερο ποσοστό της επιφάνειας μιας εικόνας δεν περιέχει κάποιο πρόσωπο (non-face region), οπότε οι Viola και Jones εισήγαγαν τον καταρράκτη ταξινομητών (cascade of classifiers). Τα χαρακτηριστικά ομαδοποιούνται σε διαφορετικά στάδια (stages) της διαδικασίας ταξινόμησης. Αν ένα παράθυρο αποτύχει στο πρώτο στάδιο τότε μπορεί να απορριφθεί και με αυτόν τον τρόπο δεν χρειάζεται να εξετασθούν τα υπόλοιπα χαρακτηριστικά. Αν περάσει τότε προχωράει στο δεύτερο στάδιο χαρακτηριστικών και η διαδικασία συνεχίζεται.

#### 1.3.1.2 Eigenfaces

O αλγόριθμος Eigenfaces δημιουργήθηκε από τους Matthew Turk και ο Alex Pentland στην εργασία τους “Face Recognition Using Eigenfaces”[17] το 1991 και στηρίχθηκε στην εργασία των L. Sirovich και M. Kirby με τίτλο “Low-dimensional procedure for the characterization of human face”[18] που προτάθηκε το 1987, για να αναπαραστήσουν αποτελεσματικότερα, οι πρώτοι, τις εικόνες προσώπου χρησιμοποίησαν την ανάλυση πρωτευουσών συνιστωσών (Principal Component Analysis – PCA). Ο αλγόριθμος αυτός προϋποθέτει όλες οι εικόνες που θα χρησιμοποιηθούν (για εκπαίδευση και οι προς εξέταση) πρέπει να έχουν όλες το ίδιο μέγεθος και να γίνουν ένα διάνυσμα (vector) ώστε τα μάτια και το στόμα να ευθυγραμμιστούν. Τέλος, για να είναι όσο το δυνατόν πιο επιτυχημένη η αναγνώριση πρέπει να υπάρχουν αρκετές εικόνες ανά άτομα του οποίου είναι επιθυμητή η αναγνώριση.

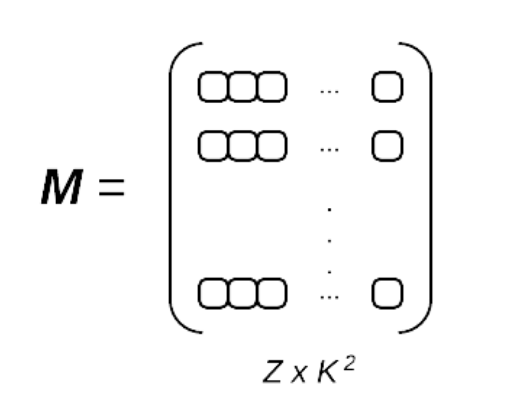
Έστω, μια τετραγωνική εικόνα διαστάσεων:



Σχήμα 1.9: Εικόνα προσώπου[19]

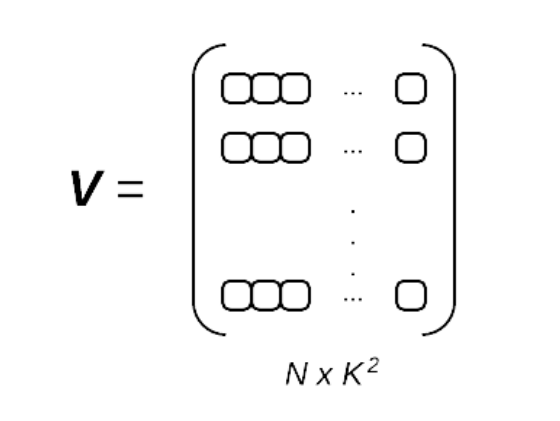
Όταν γίνεται χρήση του αλγορίθμου Eigenfaces κάθε εικόνα προσώπου πρέπει να μετατραπεί σε εικόνα με αποχρώσεις του γκρι και έναν πίνακα στοιχείων pixel (bitmap of pixels) . Όπως αναφέρθηκε προηγουμένως, η εικόνα πρέπει να γίνει διάνυσμα (vector) με διάσταση , στην συγκεριμένη περίπτωση, όπου η εικόνα δηλαδή είναι τετραγωνική, το διάνυσμα θα έχει μήκος , αυτό επιτυγχάνεται κάνοντας flattening την κάθε εικόνα.

Αφού ολοκληρωθεί το flattening όλων των εικόνων της βάσης (dataset) σχηματίζεται ένας πίνακας με “ισοπεδωμένες” εικόνες όπου το σύνολο των εικόνων που υπάρχουν στο dataset. Ο πίνακας περιέχει ολόκληρο το dataset.



Σχήμα 1.10: Πίνακας ισοπεδωμένων εικόνων

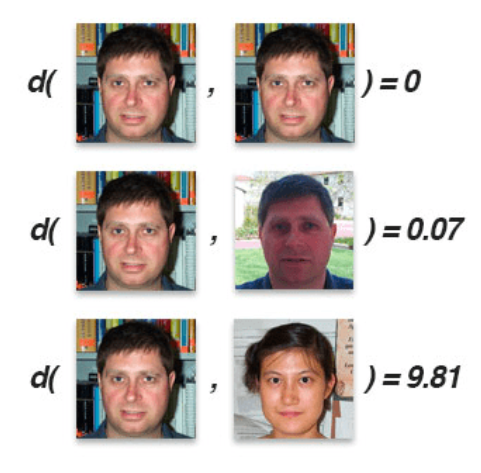
Στον πίνακα εφαρμόζεται η ανάλυση πρωτευουσών συνιστωσών όπως αναλύεται από την Lindsay Smith στο έγγραφο “A tutorial on Principal Components Analysis”[20]. To επόμενο βήμα μετά την δημιουργία του πίνακα είναι να υπολογιστεί ο μαθηματικός μέσος (mean) κάθε στήλης του πίνακα. Έπειτα, αφαιρείται ο μέσος από κάθε στήλη , η διαδικασία αυτή καλείται κεντράρισμα δεδομένων μέσου (mean-centering data). Αφού έχει κεντραριστεί ο πίνακας , υπολογίζεται ο πίνακας συνδιακύμανσης (covariance matrix) . Το επόμενο βήμα είναι η πραγματοποίηση μιας αποσύνθεσης ιδιοτιμών (eigenvalue decomposition) στον πίνακα συνδιακύμανσης με σκοπό να ληφθούν οι ιδιοτιμές και τα ιδιοδιανύσματα (eigenvectors) . Έπειτα, τα eigenvectors ταξινομούνται κατά φθίνουσα σειρά σύμφωνα με την τιμή , και επιλέγονται τα ιδιοδιανύσματα με τις αντίστοιχες μεγαλύτερες ιδιοτιμές. Τέλος, τα δεδομένα εισόδου προβάλλονται στον χώρο σύμφωνα με την εξίσωση , όπου *a* και *b* το πρώτο και δεύτερο διάνυσμα, *n* η διάσταση του διανυσματικού χώρου (vector space) και *­* είναι συνιστώσες των διανυσμάτων *a* και *b* αντίστοιχα. Ο χώρος οριοθετείται από τα κορυφαία ιδιοδιανύσματα που επιλέχθηκαν, και ονομάζονται eigenfaces.



Σχήμα 1.11: Ο πίνακας ιδιοδιανυσμάτων V

Έχοντας τα eigenfaces, τα νέα πρόσωπα μπορούν να αναπαρασταθούν λαμβάνοντας το dot product μεταξύ της ισοπεδωμένης εικόνας και των *N* eigenfaces. Έτσι κάθε πρόσωπο μπορεί να αναπαρασταθεί ως ένας γραμμικός συνδυασμός των πρωτευουσών συνιστωσών για παράδειγμα:

Οι Sirovich και Kirby για να πραγματοποιηθεί η αναγνώριση προσώπου πρότειναν τη λήψη της ευκλείδειας απόστασης μεταξύ των προβαλλόμενων αναπαραστάσεων eigenface υλοποιώντας στην ουσία έναν ταξινομητή k-NN.



Σχήμα 1.12: Ευκλείδεια απόσταση στον αλγόριθμο Eigenfaces

Όσο μικρότερη η ευκλείδεια απόσταση, τόσο πιο όμοια είναι τα δύο πρόσωπα, η συνολική ταύτιση βρίσκεται παίρνοντας την ετικέτα που σχετίζεται με το πρόσωπο με τη μικρότερη ευκλείδεια απόσταση. Στο Σχήμα 1.12 η πρώτη απόσταση είναι 0 καθώς “συγκρίνεται” η ίδια φωτογραφία. Στη δεύτερη περίπτωση παρότι είναι διαφορετική εικόνα περιέχει το ίδιο πρόσωπο και για αυτό τον λόγο η ευκλείδεια απόστασή είναι 0.07 και στην τελευταία συγκρίνονται δύο τελείως διαφορετικά πρόσωπα οδηγώντας έτσι την ευκλείδεια απόσταση σε υπερβολικά μεγάλη τιμή 9.81. Συνήθως, πέρα από έναν αλγόριθμο k-NN για να αυξηθεί η επιτυχία της ταυτοποίησης χρησιμοποιείται κάποιος πιο προχωρημένος αλγόριθμος μηχανικής όρασης όπως είναι για παράδειγμα τα SVMs (Support Vector Machines)[21].

#### 1.3.1.3 Fisherfaces

Η γραμμική ανάλυση διακρίσεων (Linear Discriminant Analysis - LDA) εκτελεί μια μείωση διαστάσεων για συγκεκριμένη κλάση (class) και εφευρέθηκε από τον στατιστικολόγο R. A. Fisher, ο οποίος την χρησιμοποίησε για την ταξινόμηση λουλουδιών στην εργασία του “The use of multiple memeters in taxonomic problems”[22] το 1936. Προκειμένου να βρεθεί ο πιο κατάλληλος συνδυασμός χαρακτηριστικών που θα είναι σε θέση να πραγματοποιεί καλύτερο διαχωρισμό των κλάσεων η LDA μεγιστοποιεί την απόσταση μεταξύ των διαφορετικών τάξεων (between-class) και ταυτοχρόνως ελαχιστοποιεί την απόσταση μέσα στην ίδια κλάση (within-class). Με άλλα λόγια, τα αντικείμενα που ανήκουν στην ίδια κλάση δεν πρέπει να έχουν μεγάλη απόσταση μεταξύ τους ενώ οι διαφορετικές κλάσεις είναι όσο πιο μακριά γίνεται[29].

Έστω ένα τυχαίο διάνυσμα με δείγματα που προέρχονται από κλάσεις , . Ο πίνακες διασποράς και υπολογίζονται ως εξής:

.

Όπου *μ* είναι ο συνολικός μέσος και είναι ο μέσος κάθε κλάσης . Ο κλασσικός αλγόριθμος Fisher ψάχνει να βρει μια προβολή,

τέτοια ώστε να μεγιστοποιείται η απόσταση μεταξύ των κλάσεων. Η λύση στην βελτιστοποίηση του προβλήματος δίνεται λύνοντας το Γενικό Πρόβλημα Ιδιοδιανύσματος (General Eigenvalue Problem)

Τέλος, η κατάταξη του είναι στο μέγιστο , όπου *N* τα δείγματα και *c* οι διαφορετικές κλάσεις. Στην αναγνώριση προτύπων το πρόβλημα είναι πως αρκετές φορές τα δείγματα είναι μικρότερων διαστάσεων από τα δεδομένα εισόδου. Αυτό λύνεται αν στα δεδομένα και στα προβαλλόμενα (projecting) δείγματα εφαρμοστεί η ανάλυση πρωτευουσών συνιστωσών (PCA), έπειτα μπορεί να πραγματοποιηθεί LDA στα δεδομένα (reduced data).

.

O πίνακας μορφοποίησης (transformation matrix) *W*, ο οποίος προβάλει ένα δείγμα μέσα στον χώρο των *(c – 1)* διαστάσεων δίνεται από τον τύπο:

#### 1.3.1.4 Local Binary Pattern (LBP)

To 2004 από τους Timo Ahonen, Abdenour Hadid και Matti Pietikäinen προτάθηκε ένας νέος αλγόριθμος για αναγνώριση προσώπου στην εργασία τους με τίτλο “Face Recognition with Local Binary Patterns” [23].



Σχήμα 1.13: Πρώτο βήμα για αναγνώριση προσώπου με LBP

Το πρώτο βήμα δοθέντος ενός προσώπου από ένα σύνολο δεδομένων προσώπων για να χρησιμοποιηθεί ο LBP είναι να χωριστεί η εικόνα σε ισομεγέθη τετράγωνα όπως στο Σχήμα 1.13. Έπειτα, για κάθε τετράγωνο από τα 49 που υπάρχουν υπολογίζεται το Local Binary Pattern ιστόγραμμα (histogram). Ένα ιστόγραμμα δεν λαμβάνει υπόψη του τις χωρικές πληροφορίες (spatial information) σχετικά με τον τρόπο με τον οποίο τα μοτίβα είναι προσανατολισμένα μεταξύ τους. Υπολογίζοντας όμως ένα ιστόγραμμα για κάθε ένα από τα κελιά είναι δυνατόν να κωδικοποιηθεί ένα επίπεδο τοπικών πληροφοριών όπως τα μάτια, η μύτη, το στόμα. Επιπλέον, η χωρική κωδικοποίηση δίνει την δυνατότητα να αποδοθούν βάρη στα ιστογράμματα που προέκυψαν για κάθε ξεχωριστό κελί παρέχοντας έτσι μεγαλύτερη διακριτική δύναμη σε πιο διακριτικά χαρακτηριστικά του προσώπου. Για παράδειγμα

|  |  |
| --- | --- |
|  |  |

Σχήμα 1.14: Σύστημα στάθμισης

To ιστόγραμμα LBP για τα λευκά κελιά (π.χ. μάτια) είναι κατά τέσσερις φορές πιο σημαντικά από τα υπόλοιπα. Η περιοχή του στόματος και των αυτιών (φωτεινό γκρι) συνεισφέρουν δύο φορές περισσότερο. Τα σκούρα γκρι (μέτωπο, εσωτερικά του μάγουλου) συνεισφέρουν μια φορά και τα μαύρα (μύτη, εξωτερικό του μάγουλου) απορρίπτονται και συνεισφέρουν μηδέν φορές.

Οι τιμές στάθμισης που αναφέρθηκαν βρέθηκαν πειραματικά από τους Ahonen, Hadid και Pietikäinen εκτελώντας αλγορίθμους συντονισμού υπερπαραμέτρων πάνω από τα δεδομένα εκπαίδευσης και επικύρωσης – δοκιμής. Τα σταθμισμένα LBP ιστογράμματα συνδέονται μεταξύ τους σχηματίζοντας το τελικό διάνυσμα χαρακτηριστικών.

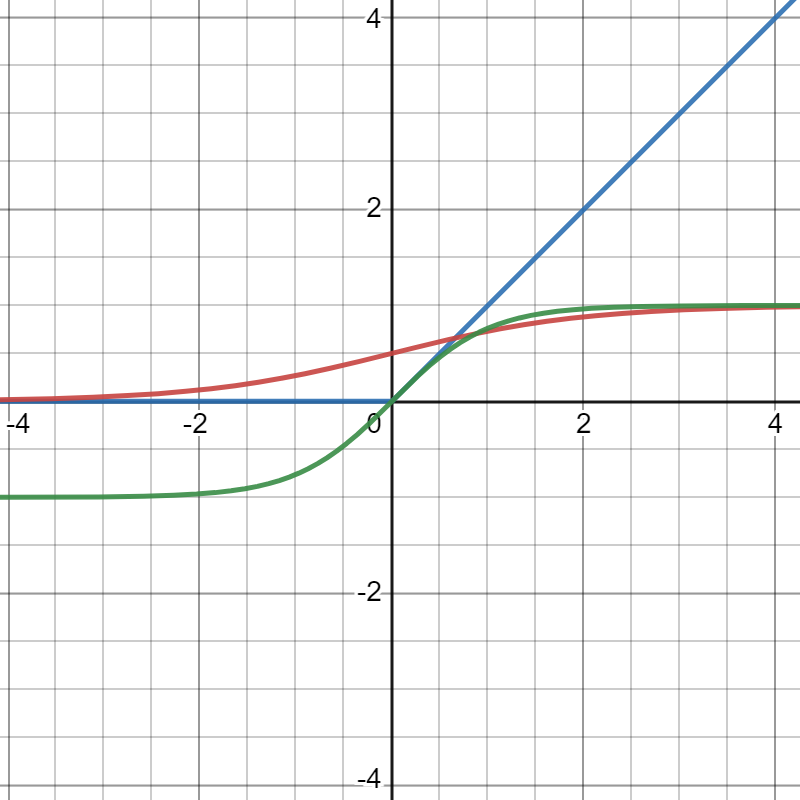
Η αναγνώριση προσώπου πραγματοποιείται χρησιμοποιώντας την απόσταση και έναν ταξινομητή πλησιέστερου γείτονα (nearest neighbor classifier) ως εξής: Αρχικά, παρουσιάζεται ένα πρόσωπο στο σύστημα και εξάγονται τα LBPs. Στη συνέχεια, ζυγίζονται και συνενώνονται με τον ίδιο τρόπο όπως τα δεδομένα εκπαίδευσης. Έπειτα, με εκτελείται ο k-NN με την απόσταση για να βρεθεί το πιο κοινό (όμοιο) πρόσωπο στα δεδομένα εκπαίδευσης. Τέλος, επιλέγεται το όνομα του ανθρώπου που σχετίζεται με το πρόσωπο που έχει την μικρότερη απόσταση[24].

#### 1.3.1.5 Αναγνώριση προσώπου με CNN

Ένα CNN (Convolutional Neural Network) είναι ένα νευρωνικό δίκτυο το οποίο έχει κρυφά επίπεδα (hidden layers) συνέλιξης και ομαδοποίησης (pooling) συνήθως εναλλάξ και προτάθηκε από τον μεταδιδακτορικό ερευνητή Yann LeCun την δεκαετία του 1980s χρησιμοποιώντας ως έμπνευση το neocognitron, ένα απλοϊκό νευρωνικό για αναγνώριση εικόνων που σχεδιάστηκε από τον Ιάπωνα ερευνητή Kunihiko Fukusima[25]. Το ποσοστό επιτυχίας του νευρωνικού δικτιού πέρα από την δομή που έχει σχετίζεται και με την συνάρτηση ενεργοποίησης (activation function) που έχει υιοθετηθεί. Η πιο συχνά χρησιμοποιούμενη συνάρτηση είναι η ReLU (Rectified Linear Unit), η οποία για όλες τις αρνητικές τιμές και το μηδέν επιστρέφει μηδέν ενώ για τις θετικές επιστρέφει δεκαδικό από το μηδέν μέχρι το 1 και δίνεται από των τύπο:

.

Πέρα από την ReLU αρκετά CNNs χρησιμοποιούν είτε την σιγμοειδή (sigmoid) ή την υπερβολική εφαπτομένη (hyperbolic tangent – tanh) με τύπους αντίστοιχα:



Σχήμα 1.15: Συναρτήσεις ενεργοποίησης ReLU (μπλε), Sigmoid (κόκκινη), Tanh (πράσινη)

Για την εκπαίδευση του νευρωνικού χρησιμοποιείται ο αλγόριθμος οπισθοδρόμησης (back propagation) ο οποίος με βάση μια συνάρτηση λάθους (error function) συνήθως το λάθος μέσου τετραγώνου (mean square error – MSE) ενημερώνει τα βάρη στις συνδέσεις του δικτύου ώστε να μπορεί να δώσει πιο ακριβή και σωστά αποτελέσματα. Ο τύπος για την MSE είναι:

όπου με *W* συμβολίζονται ο πίνακας βαρών, με *B* ο πίνακας προκαταλήψεων (bias) του nn (neural network), *aiL* και *tiL* υποδεικνύουν την πραγματική τιμή και την τιμή του *iοστού* νευρώνα στο επίπεδο εξόδου με *NL* νευρώνες που έδωσε το nn αντίστοιχα. Η έξοδος του *i-οστού* νευρώνα στον *l-οστό* επίπεδο μπορεί να υπολογισθεί ως:

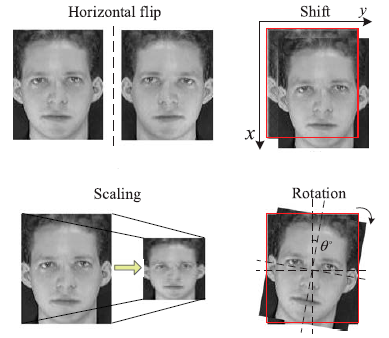
όπου *fil* είναι η συνάρτηση ενεργοποίησης, *bil* είναι το bias του *ail.* Tα βάρη σύνδεσης (connection weight) μεταξύ του *i-οστού* νευρώνα του *l-οστού* επίπεδου και το *j-οστού* νευρώνα του προηγούμενου επιπέδου συμβολίζεται ως *wijl*. Τo νούμερο του *L-*επιπέδου του νευρωνικού συμβολίζεται με *l*. *Nl* είναι το πλήθος των νευρώνων του επιπέδου *l*. Αν χρησιμοποιηθεί η μέθοδος εμπροσθοδρόμησης (forward propagation) τα βάρη του νευρωνικού δίνονται:

Στη λειτουργία συνέλιξης, η λειτουργία μηδενισμού ακμών εφαρμόζεται για την εικόνα εισόδου ώστε να διασφαλιστεί η ορθολογικότητα των αποτελεσμάτων, δηλαδή τα στοιχεία της εικόνας που εισάχθηκε και του πυρήνα συνέλιξης μπορούν να σταθμιστούν (weight) και να αθροιστούν (summated) διαδοχικά. Το CNN μπορεί να θεωρηθεί ως μια κοινή διαδικασία εξαγωγής χαρακτηριστικών όπως τα περισσότερα νευρωνικά δίκτυα που χρησιμοποιούνται για την εξαγωγή χαρακτηριστικών.

Έστω, πως ως εισάγουμε μια τετραγωνική εικόνα με διαστάσεις 100x100 στο νευρωνικό, το οποίο έχει στο κρυφό επίπεδο 100 νευρώνες. Αυτό σημαίνει πως θα υπάρχουν 100x100x100 βάρη σύνδεσης μεταξύ του επιπέδου εισόδου και του πρώτου κρυφού επιπέδου αν όλα τα εικονοστοιχεία συνδέονται με όλους τους νευρώνες. Ο τεράστιος υπολογιστικός όγκος, που σε αυτή την περίπτωση αναπόφευκτα θα προκύψει, θα καθυστερήσει τον χρόνο εκπαίδευσης και πιθανότατα θα μειώσει την απόδοση του νευρωνικού. Αντίθετα, αν κάθε κρυφός νευρώνας είναι συνδεδεμένος σε ένα τοπικό πεδίο της εικόνας, διαστάσεων για παράδειγμα 10x10, τα βάρη σύνδεσης θα είναι 10x10x100. Τα βάρη σύνδεσης μπορούν να μειωθούν και άλλο καθώς όλοι οι νευρώνες ενός πυρήνα συνέλιξης έχουν τα ίδια βάρη οδηγώντας σε περαιτέρω μείωση του υπολογιστικού φόρτου και αύξηση της απόδοσης.

Τα στρώματα ομαδοποίησης βρίσκονται συνήθως μετά από τα επίπεδα συνέλιξης και χρησιμοποιούνται κυρίως για τη συμπίεση των δεδομένων χαρακτηριστικών εξόδου των συνελικτικών στρωμάτων. Μετά το επίπεδο συγκέντρωσης, τα βελτιωμένα αποτελέσματα εξόδου μπορούν να μειώσουν την πιθανότητα υπερεκπαίδευσης (over-fitting) στο νευρωνικό δίκτυο. Ουσιαστικά, η συγκέντρωση είναι μια διαδικασία μείωσης της εικόνας, η οποία μπορεί να ταξινομηθεί ως συγκέντρωση μέσης τιμής (mean-pooling), μέγιστης συγκέντρωσης (max-pooling), επικάλυψης συγκέντρωσης (overlapping-pooling), στοχαστικής συγκέντρωσης (stochastic-pooling) και συνολικής μέσης συγκέντρωσης (global average pooling).

Το πρόβλημα που εμφανίζεται στο CNN είναι πως απαιτούν αρκετές φωτογραφίες – δεδομένα για να εκπαιδευτούν κάνοντας τα δύσχρηστα για απλές χρήσεις όπου δεν μπορούν ή δεν επιτρέπεται η συλλογή πολλών δεδομένων. Λύση στο πρόβλημα αυτό δίνεται από τις μεθόδους αύξησης δεδομένων (Σχήμα 1.16), όπως είναι για παράδειγμα η μετακίνηση (shift), η αλλαγή μεγέθους (scaling), η περιστροφή κατά γωνία (rotation) και η οριζόντια αντιστροφή (horizontal flip).



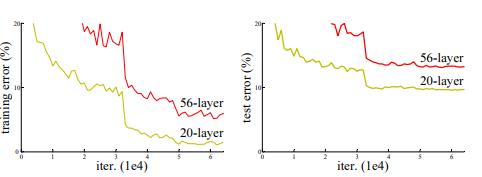
Σχήμα 1.16: Η τέσσερις μέθοδοι αύξησης δεδομένων

Ένα CNN μπορεί να πετύχει μέχρι και 99.5% ακρίβεια στα δεδομένα ελέγχου[26].

#### 1.3.1.6 Αναγνώριση προσώπου με ResNet

Τα Residual Network (ResNet) είναι ένας ειδικός τύπος βαθιών νευρωνικών δικτύων που δημιουργήθηκε το 2015 από τους Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun στην ερευνητική εργασία με τίτλο “Deep Residual Learning for Image Recognition”[31]. Τα δίκτυα αυτά αποδείχτηκαν αρκετά αποδοτικά καθώς διακρίθηκαν με την πρώτη θέση στον διαγωνισμό ILSVRC (ImageNet Large Scale Visual Recognition Challenge) του 2015 με ποσοστό λάθους μόλις 3.57%[42].

Για να λυθεί ένα πρόβλημα με το μεγαλύτερο δυνατό ποσοστό επιτυχίας και με καλύτερη απόδοση με την μέθοδο των νευρωνικών δικτύων συνήθως προστίθενται επιπλέον convolutional επίπεδα με σκοπό το κάθε επίπεδο να εξειδικευτεί στον εντοπισμό ενός συγκεκριμένου χαρακτηριστικού, όπως για παράδειγμα η εύρεση ακμών και η αναγνώριση υφών. Στο paper που δημοσίευσαν οι δημιουργοί του ResNet αποδεικνύουν πως η ανεξέλεγκτη προσθήκη επιπέδων δεν είναι η πιο αποδοτική μέθοδος. Συγκεκριμένα, όπως φαίνεται στο σχήμα 1.17, σύγκριναν τα ποσοστά λάθους ενός νευρωνικού με 20 και ενός με 56 επίπεδα. Το δεύτερο δίκτυο κατά την διαδικασία της εκπαίδευσης παρουσίασε μεγαλύτερο λάθος κάτι που επιβεβαιώθηκε και από τον αντίστοιχο δείκτη κατά την διαδικασία της επαλήθευσης (test).



Σχήμα 1.17: Ποσοστά λάθους κατά την εκπαίδευση και επαλήθευση ενός δικτύου 20 και 56 επιπέδων[31]

Το πρόβλημα αυτό εξισορροπείτε με την ανάπτυξη των ResNet τα οποία αποτελούνται από Residual Blocks.

Η πιο βασική διαφορά που παρατηρείται είναι πως στα ResNet υπάρχει μια άμεση σύνδεση που παρακάμπτει κάποια ενδιάμεσα επίπεδα και ονομάζεται σύνδεση παράλειψης και είναι ο πυρήνας των residual block. Εξαιτίας της σύνδεσης αυτής η έξοδος του επιπέδου δεν είναι η ίδια καθώς η είσοδος δεν πολλαπλασιάζεται με τα βάρη του στρώματος ακολουθούμενα από την προσθήκη του όρου πόλωσης (bias term) και στην συνέχεια χρησιμοποιείται ως είσοδος στην συνάρτηση ενεργοποίησης οπότε ως τελική έξοδο έχουμε: . Με την εφαρμογή της σύνδεσης παράληψης η έξοδος είναι . Η σύνδεση παράληψης των ResNet επιλύει το πρόβλημα της εξαφάνισης της κλίσης (vanishing gradient) επιτρέποντας την εναλλακτική διαδρομή για την κλίση.

Τα δίκτυα αυτά έχουν δομή 34 επιπέδων εμπνευσμένη από το VGG-19 και έπειτα χρησιμοποιείται η σύνδεση παράληψης.

## 1.4 User-friendly εφαρμογές

### 1.4.1 Διεπαφή χρήστη

Ο όρος διεπαφή χρήστη (User Interface - UI) αναφέρεται στη γραφική διάταξη μιας εφαρμογής και αποτελείται από όλα εκείνα τα στοιχεία με τα οποία αλληλεπιδρά ο χρήστης, όπως για παράδειγμα τα κουμπιά, τη διάταξη των οθονών και κάθε είδους οπτικού στοιχείου αλληλεπίδρασης.

Τη δημιουργία ενός καλού UI την αναλαμβάνουν οι UI designers, οι οποίοι είναι σχεδιαστές γραφικών, ασχολούνται με την αισθητική και εναπόκειται σε αυτούς να βεβαιωθούν ότι η διεπαφή της εφαρμογής είναι ελκυστική, έχει το κατάλληλο θέμα ώστε να ταιριάζει με τον σκοπό και αποφασίζουν για την εμφάνιση της εφαρμογής, επιλέγοντας για παράδειγμα συνδυασμούς χρωμάτων, ή/και τη γραμματοσειρά που θα φαίνεται το κείμενο. Επιπλέον, πρέπει να αποφεύγουν τα περιττά στοιχεία και να είναι ευανάγνωστη η γλώσσα που χρησιμοποιείται στις ετικέτες και στα μηνύματα έτσι ώστε η διεπαφή να διατηρηθεί απλή και κατανοητή. Δημιουργούν μια διεπαφή που χαρακτηρίζεται από συνέπεια (consistency), αυτό επιτυγχάνεται με την χρήση κοινών στοιχείων τα οποία κάνουν τους χρήστες να αισθάνονται πιο άνετα και καθιστώντας τους πιο αποτελεσματικούς. Μόλις ένας χρήστης μάθει πώς να κάνει κάτι, θα πρέπει να μπορεί να μεταφέρει αυτήν την ικανότητα σε άλλα μέρη του συστήματος (της εφαρμογή ή του ιστοτόπου). Εξετάζουν τις χωρικές σχέσεις μεταξύ των στοιχείων της σελίδας και δομούν τη σελίδα με βάση τη σημασία του περιεχομένου. Η προσεκτική τοποθέτηση των αντικειμένων μπορεί να επιστήσει την προσοχή του χρήστη στις πιο σημαντικές πληροφορίες και μπορεί να βοηθήσει στη σάρωση και την αναγνωσιμότητα της εφαρμογής[27].

### 1.4.2 Εμπειρία χρήστη

Η εμπειρία χρήσης, θετική ή αρνητική που θα έχει ένας χρήστης, (User Experience - UX) από την εφαρμογή καθορίζεται από τον τρόπο που αυτός αλληλεπιδρά με την εφαρμογή. Η εμπειρία χρήστη καθορίζεται από το πόσο εύκολη ή δύσκολη είναι η αλληλεπίδραση με τα στοιχεία της διεπαφής χρήστη που έχουν δημιουργήσει οι σχεδιαστές UI.

Οι σχεδιαστές UX εστιάζουν στην κατανόηση των αναγκών, τις ικανότητες και τους περιορισμούς των χρηστών με απώτερο σκοπό τη δημιουργία μιας εύκολης και απλής για αυτούς διεπαφής. Επιπλέον, λαμβάνουν υπόψη τους επιχειρηματικούς στόχους της εταιρίας που διαχειρίζεται το έργο. Οι βέλτιστες πρακτικές UX προωθούν τη βελτίωση της ποιότητας της αλληλεπίδρασης του χρήστη και της αντίληψης του για την εφαρμογή και οποιεσδήποτε σχετικές υπηρεσίες[28].

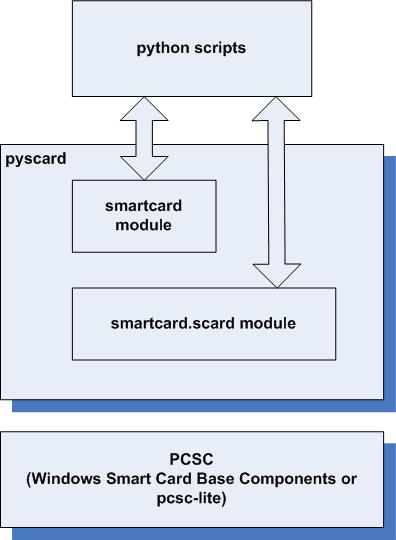
## 1.5 Επιλογές

Για τις ανάγκες αυτής της πτυχιακής εργασίας θα χρησιμοποιηθούν παθητικές ετικέτες RFID καθώς δεν χρειάζεται να εκπέμπουν συνέχεια ή σε μεγάλες αποστάσεις το σήμα τους. Σημαντική σημείωση: οι ετικέτες που χρησιμοποιήθηκαν είναι NFC που είναι ένα υποσύνολο της τεχνολογίας RFID με συχνότητα λειτουργίας στα 13.56MHz (RFID HF)[30]. Επιπλέον, για την ανάγνωση – εγγραφή των καρτών έχει αγοραστεί έτοιμη λύση USB RFID/NFC scanner για ηλεκτρονικό υπολογιστή. Τέλος, την αναγνώριση προσώπου την αναλαμβάνει ένα μοντέλο ResNet με 27 επίπεδα συνέλιξης το οποίο βασίζεται στο ResNet-34 που αναπτύχθηκε στο paper “Deep residual learning for image recognition”[31].

# Κεφάλαιο 2 Μέθοδοι, πρωτόκολλα, αλγόριθμοι, τεχνικές και βιβλιοθήκες

## 2.1 Σύνδεση RFID/NFC reader με την python

Με σκοπό να ενσωματωθούν οι δυνατότητες, σύνδεσης του προγράμματος με έναν αναγνώστη RFID/NFC και της ανάγνωσης πληροφοριών από κατάλληλη κάρτα χρησιμοποιήθηκε μια βιβλιοθήκη pyscard[32] της Python.



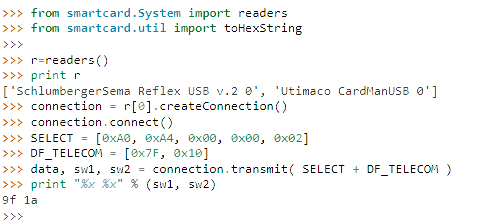
Εικόνα 2.1: Η δομή της βιβλιοθήκης pyscard

H βιβλιοθήκη αυτή, χρησιμοποιώντας το πρωτόκολλο PCSC παρέχει δυνατότητες αναγνώρισης, ανάγνωσης, και εγγραφής έξυπνων καρτών και ετικετών RFID/NFC σε προγράμματα, που δημιουργήθηκαν με Python[33]. Η ομάδα εργασίας PCSC έχει ορίσει ένα API (Application Programming Interface) για τη διασύνδεση smartcards και των συσκευών ανάγνωσης τους με τους υπολογιστές. Το λειτουργικό σύστημα της Microsoft ενσωματώνει την υποστήριξη έξυπνων καρτών, που συνήθως ονομάζεται PCSC. Για διανομές Linux καθώς επίσης και για το MacOS έχει υλοποιηθεί παρόμοιο λογισμικό υποστήριξης με όνομα PCSC-lite. Το API έχει υλοποιηθεί στη γλώσσα C και παρέχονται αρκετές γέφυρες (bridges) καθιστώντας εφικτή την πρόσβαση σε αυτό από διαφορετικές γλώσσες όπως η Java, η Visual Basic, η Python και άλλες.

Η μονάδα που γεφυρώνει και επιτρέπει τη μεταφορά πληροφοριών από τον αναγνώστη στις κάρτες και αντίστροφα καλείται APDU (Application Protocol Data Unit) και ορίζεται σύμφωνα με το πρότυπο ISO/IEC 7816-4[34, 35]. Υπάρχουν τέσσερις τρόποι αποστολής πληροφοριών ο reader – centric approach, answer to reset (ATR), card – centric approach και object – centric approach.

### 2.1.1 Reader – centric approach

Στην αναγνωστοκεντρική προσέγγιση δημιουργείται μια σύνδεση με την έξυπνη κάρτα μέσω του αναγνώστη και χρησιμοποιώντας την σύνδεση αυτή στέλνονται οι εντολές APDU.



Σχήμα 2.1: Reader – centric approach

Με την εντολή ανακτώνται οι συνδεδεμένοι με τον υπολογιστή αναγνώστες. Έπειτα, δημιουργείται η σύνδεση για τον πρώτο στη λίστα αναγνώστη με την εντολή

και στη συνέχεια ανοίγει η σύνδεση . Τέλος, με την μπορούν να μεταδοθούν APDU εντολές στην κάρτα.

Τα βασικά μειονεκτήματα αυτής της προσέγγισης είναι:

* Ο δείκτης (index) του αναγνώστη ή το όνομα του πρέπει να είναι δηλωμένα στο script και πρέπει να αλλαχτούν μέσω κώδικα για να ταιριάζουν τις ανάγκες του κάθε χρήστη,
* Με τη μέθοδο αυτή δεν υπάρχει η δυνατότητα γνώσης αν υπάρχει κάποια κάρτα στον αναγνώστη, θα πρέπει να εκτελεστεί το script ώστε να πιάσουμε (catch) μια εξαίρεση τύπου CardConnectionException, η οποία και υποδεικνύει τη μη ύπαρξη κάρτας RFID/NFC στον αναγνώστη,
* Τέλος, δεν μπορεί να ελεγχθεί αν ο τύπος της κάρτας που εισάχθηκε στον αναγνώστη είναι του τύπου αναμέναμε.

### 2.1.2 Answer To Reset (ATR)

ATR είναι η πρώτη απάντηση μιας κάρτας σε έναν αναγνώστη, σκοπός της είναι να περιγράψει τις υποστηριζόμενες παραμέτρους επικοινωνίας και περιγράφεται από το πρότυπο ISO/IEC7816-3. Ο αναγνώστης, ο οδηγός του αναγνώστη (smart card reader driver) καθώς και το λειτουργικό σύστημα θα χρησιμοποιήσουν τις παραμέτρους αυτές για να πραγματοποιήσουν την επικοινωνία με την κάρτα. Τα πρώτα bytes του ATR περιγράφουν τη τάση (άμεση ή αντίστροφη) , τα επόμενα περιγράφουν τις διαθέσιμες διεπαφές επικοινωνίας και τις παραμέτρους τους. Έπειτα, υπάρχουν τα ιστορικά bytes (Historical Bytes), τα οποία δεν είναι τυποποιημένα και χρησιμοποιούνται για τη μετάδοση πληροφοριών όπως ο τύπος της κάρτας, η έκδοση του ενσωματωμένου λογισμικού και η κατάσταση της κάρτας. Τέλος, υπάρχει ένα byte αθροίσματος ελέγχου (checksum).

### 2.1.3 Card – centric approach

Στη καρτοκεντρική προσέγγιση δημιουργούμε ένα αίτημα (request) για ένα συγκεκριμένο τύπο κάρτας και περιμένουμε μέχρι μια κάρτα που ταιριάζει στον τύπο να εισαχθεί. Μόλις εισαχθεί η κατάλληλη κάρτα δημιουργείται αυτόματα μια σύνδεση και είμαστε σε θέση να στείλουμε APDU εντολές.

### 2.1.4 Object – centric approach

Στην αντικειμενοκεντρική προσέγγιση συσχετίζεται ένα αντικείμενο υψηλού επιπέδου (high – level object) με ένα σύνολο από έξυπνες κάρτας που υποστηρίζονται από το αντικείμενο.

### 2.1.5 Τι χρησιμοποιήθηκε

Όπως φαίνεται και από το σχήμα 2.2 η προσέγγιση που χρησιμοποιήθηκε είναι η αναγνωστοκεντρική.



Σχήμα 2.2: Η συνάρτηση readuid()

Παρά τα εμφανή μειονεκτήματα αυτής της προσέγγισης προτιμήθηκε έναντι των υπόλοιπων καθώς προσφέρει τον πιο απλό και γρήγορο τρόπο επικοινωνίας με τον αναγνώστη και κατ’ επέκταση με την κάρτα. Επιπλέον, δεν θα υπάρχει δεύτερος αναγνώστης συνδεδεμένος ταυτόχρονα στον υπολογιστή. Μέσω της Python o χρήστης ενημερώνεται με μήνυμα λάθους σε περίπτωση που δεν υπάρχει κάποια κάρτα στον αναγνώστη.

## 2.2 Αναγνώριση προσώπου (Face recognition)

Τη λειτουργικότητα της αναγνώριση προσώπου την αναλαμβάνει η βιβλιοθήκη face\_recognition της Python, η οποία σύμφωνα με το documentation[36] της, στηρίζεται στο μοντέλο αναγνώρισης προσώπου της dlib με deep learning, το οποίο στο benchmark του MIT Labeled Faces in the Wild[37] έκανε σωστή πρόβλεψη στο 99.38% των περιπτώσεων[38, 39].

To module αυτό για να εντοπίσει αν και που υπάρχουν πρόσωπα σε μια εικόνα χρησιμοποιεί τον αλγόριθμο HOG (Histogram of Oriented Gradients). O HOG είναι ένας αλγόριθμος περιγραφής χαρακτηριστικών και χρησιμοποιείται στην υπολογιστική όραση για να εντοπίζει αντικείμενα. Η τεχνική αυτή μετράει την ύπαρξη της κατεύθυνσης του gradient σε ένα τοπικό σημείο της εικόνας. O περιγραφέας ΗΟG επικεντρώνεται στη δομή ή το σχήμα ενός αντικειμένου[45]. Το γεγονός πως για να δημιουργήσει ένα ιστόγραμμα για κάθε περιοχή της εικόνας χρησιμοποιεί το μέγεθος καθώς και τη γωνία της κλίσης καθιστά τον αλγόριθμο αυτό καλύτερο από οποιοδήποτε περιγραφέα ακμών. Για να εξαχθούν τα χαρακτηριστικά μιας εικόνας με HOG πρώτα πρέπει η εικόνα να πάρει τις κατάλληλες διαστάσεις (128x64 pixels). Έπειτα, υπολογίζεται η κλίση της εικόνας λαμβάνοντας υπόψη το μέγεθος και τη γωνία. Από ένα μπλοκ 3x3 pixels υπολογίζονται τα *Gx* και *Gy* για κάθε pixel με τους παρακάτω τύπους:

όπου *r*, *c* συμβολίζει της γραμμές και τις στήλες αντίστοιχα. Αφού ολοκληρωθεί ο υπολογισμός των *Gx* και *Gy* μπορούν να υπολογισθούν το μέγεθος (magnitude) και η γωνία (angle) κάθε pixel

Αφού ληφθεί η κλίση για κάθε εικονοστοιχείο οι πίνακες διαβάθμισης (gradient matrices) χωρίζονται σε κελιά 8x8 για να σχηματίσουν ένα μπλοκ. Για κάθε μπλοκ υπολογίζεται ένα ιστόγραμμα 9 σημείων. Ένα ιστόγραμμα 9 σημείων αναπτύσσει ένα ιστόγραμμα με 9 bins το κάθε ένα από τα bins έχει εύρος γωνίας 20º. Καθώς ένα block αποτελείται από 64 διαφορετικές τιμές, για κάθε ένα σετ (magnitude gradient) πραγματοποιείται ο ακόλουθος υπολογισμός:

Κάθε j-στό bin θα έχει όρια που δίνονται:

και τιμή κέντρου ίση με:

Για κάθε κελί ενός μπλοκ υπολογίζεται πρώτα το j-στό bin και έπειτα η τιμή που θα δοθεί σε αυτό και στο επόμενο bin σύμφωνα με τους παρακάτω τύπους:

Ένας πίνακας λαμβάνεται ως bin για ένα μπλοκ και οι τιμές των και εντάσσονται στον πίνακα στον δείκτη j και (j+1) που υπολογίζονται για κάθε pixel. Ο πίνακας που δημιουργείται μετά τους υπολογισμούς αυτούς έχει σχήμα 16x8x9. Όταν έχουν υπολογισθεί όλα τα ιστογράμματα για όλα τα μπλοκ, 4 μπλοκ από τον πίνακα συγχωνεύονται για να σχηματίσουν ένα νέο μπλοκ 2x2. Η ομαδοποίηση αυτή, γίνεται με αλληλεπικαλυπτόμενο τρόπο και με βηματισμό των 8 pixel. Για τα 4 κελιά σε ένα νέο πλέον μπλοκ, ενώνονται και τα ιστογράμματα των 9 σημείων για κάθε συστατικό κελί και παράγεται ένα διάνυσμα 36 χαρακτηριστικών.

Οι τιμές του κάθε μπλοκ κανονικοποιούνται από τον κανόνα L2:

όπου *ε* είναι μια μικρή τιμή της τάξεως 1e-05 και χρησιμοποιείται για να αποφευχθεί η διαίρεση με το 0. Για την κανονικοποίηση, η τιμή του k υπολογίζεται πρώτα με τους ακόλουθους τύπους:

Αυτή η κανονικοποίηση γίνεται για να μειωθεί η επίδραση των αλλαγών στην αντίθεση μεταξύ των εικόνων του ίδιου αντικειμένου. Από κάθε μπλοκ συλλέγεται ένα διάνυσμα χαρακτηριστικών 36 σημείων. Στην οριζόντια κατεύθυνση υπάρχουν 7 μπλοκ και στην κατακόρυφη κατεύθυνση υπάρχουν 15 μπλοκ. Έτσι, το συνολικό μήκος των χαρακτηριστικών HOG θα είναι: 7 x 15 x 36 = 3780.

# Κεφάλαιο 3 Υλικό

## 3.1 Η κατασκευή

Η κατασκευή αποτελείται από έναν προσωπικό υπολογιστή τύπου laptop, ο οποίος έχει ενσωματωμένη webcam και έναν εξωτερικό αναγνώστη RFID/NFC, ο οποίος συνδέεται μέσω USB θύρας με τον υπολογιστή όπως φαίνεται στο Σχήμα 3.1.



Σχήμα 3.1: Η συνδεσμολογία

## 3.2 Σύστημα ανάπτυξης

Το πρόγραμμα αναπτύχθηκε σε προσωπικό υπολογιστή τύπου laptop. Πιο συγκεκριμένα το laptop είναι το Dell G5-5587 το οποίο έχει τα παρακάτω specs:

Πίνακας 3.1: Χαρακτηριστικά συστήματος

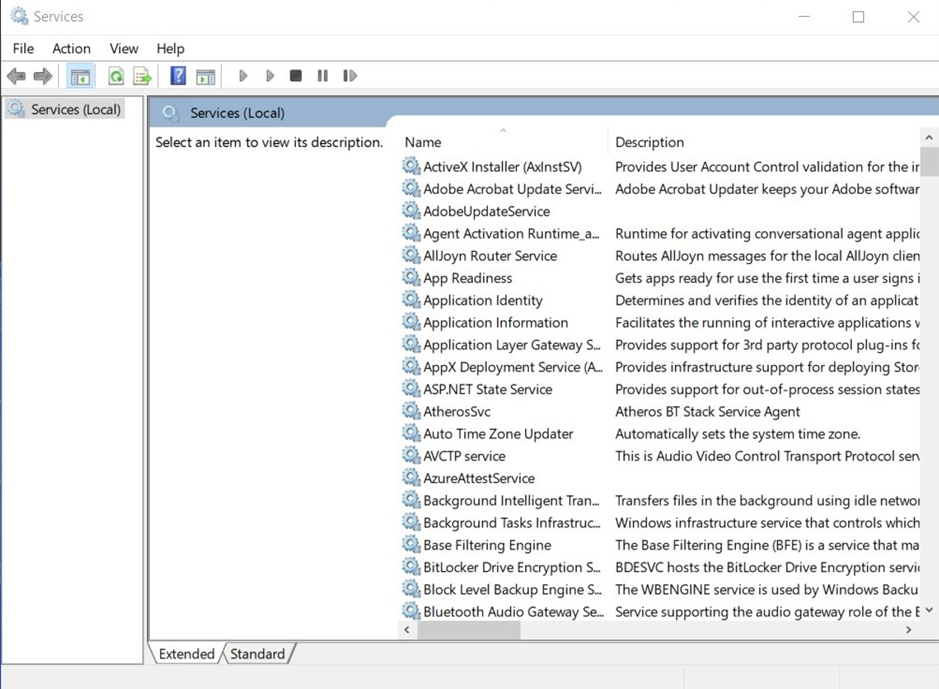
|  |  |
| --- | --- |
| CPU | Intel Core i7-8750H @ 2.20 GHz |
| GPU | Intel UHD Graphics 630 & NVIDIA GeForce GTX 1050 Ti 4 GB |
| RAM | 32 GB |
| Primary Storage | M.2 NVMe SSD 1 TB |
| Secondary Storage | HDD 1 TB |

Η κάμερα που χρησιμοποιήθηκε είναι η ενσωματωμένη κάμερα του συστήματος ανάπτυξης του προγράμματος και είναι σε θέση να βιντεοσκοπήσει σε ανάλυση 720 pixel με αναλογία 16:9 στα 30 καρέ το δευτερόλεπτο.

## 3.3 Αναγνώστης RFID

Για την υλοποίηση αυτής της πτυχιακής εργασίας αγοράστηκε από το κατάστημα “HELLAS digital” ο αναγνώστης “ACR112U USB NFC Reader” της εταιρίας Advanced Card Systems Ltd. Προτιμήθηκε ο συγκεκριμένος έναντι άλλων αναγνωστών καθώς υποστηρίζει το πρωτόκολλο CCID (Chip Card Interface Device) το οποίο αποτελεί ένα USB πρωτόκολλο που επιτρέπει σε μία έξυπνα κάρτα να διαβαστεί από ένα σύστημα απλά μέσω της διεπαφής USB και υπάρχει στα Windows από την έκδοση των Windows 2000. Επιπλέον, υποστηρίζει το PC/SC (Personal Computer/Smart Card) API το οποίο υποστηρίζεται και από την Microsoft μεταξύ άλλων εταιριών. Προσφέρει ταχύτητα ανάγνωσης εγγραφής μέχρι και 424 Kbps, υποστηρίζει τα ISO 14443 Type A και B και όλους τους τύπους NFC ετικετών όπως αυτοί ορίζονται στο πρότυπο ISO/IEC 18092. Τέλος, ένα αρκετά σημαντικό χαρακτηριστικό είναι πως επιτρέπει την ανάγνωση μόνο μιας ετικέτας την φορά ώστε να αποφευχθούν πιθανές συγκρούσεις (anti-collision feature).

Έπειτα από δοκιμές που πραγματοποιήθηκαν με το λογισμικό της εταιρίας “GoToTags” εντοπίστηκε ένα πρόβλημα που εμπόδιζε την ορθή λειτουργία του αναγνώστη με το λειτουργικό σύστημα των Windows.



Σχήμα 3.2: Windows services

Τα Windows διαθέτουν κάποιες υπηρεσίες (services) ώστε να λειτουργούν σωστά, μια από τις οποίες είναι η “Certificate Propagation”. Η συγκεκριμένη υπηρεσία ήταν η αιτία ώστε η συσκευή ανάγνωσης να χάνει συχνά την σύνδεση με τον υπολογιστή και να μην λειτουργεί όπως θα έπρεπε[31]. Η συγκεκριμένη υπηρεσία είναι ο πρόγονος του σύγχρονου “Windows Hello!” και χρησιμοποιούνταν μέχρι την εμφάνιση αυτού για να βοηθά στην πιο εύκολη, γρήγορη και προπαντός ασφαλή σύνδεση του χρήστη σε ένα προστατευμένο υπολογιστή (password protected computer) αντιγράφοντας τα πιστοποιητικά του χρήστη και της ρίζας (root certificates) από έξυπνες κάρτες (RFID/NFC cards) στο χώρο αποθήκευσης πιστοποιητικών χρήστη (user’s certificate store), και ανίχνευε πότε μια έξυπνη κάρτα διαβαζόταν από έναν αναγνώστη, και αν ήταν αναγκαίο εγκαθιστούσε τον αντίστοιχο driver smart card Plug&Play. Η συγκεκριμένη υπηρεσία πρέπει να απενεργοποιηθεί στα σύγχρονα συστήματα για να δουλέψουν σωστά οι περισσότεροι αναγνώστες/εγγραφείς RFID (RFID readers/writers)[32].

## 3.4 Εικόνες προσώπων

Με σκοπό να είναι όσο το δυνατό πιο ρεαλιστικό το σύστημα, δημιουργήθηκε ένα python script το οποίο έχει την δυνατότητα να αντλήσει έναν μεγάλο όγκο από face images από το site “This person does not exist” και βρίσκεται στο “Παράρτημα 1”.

Η ιστοσελίδα κατασκευάστηκε από τον μηχανικό λογισμικού Philip Wang χρησιμοποιώντας τον αλγόριθμο GAN (Generate Adversarial Network), που δημιουργήθηκε από την Nvidia, είναι σε θέση να παράγει μια αρκετά αληθοφανή εικόνα προσώπου ενός ανθρώπου που δεν είναι υπαρκτό πρόσωπο όπως φαίνεται στο σχήμα 3.3[43].



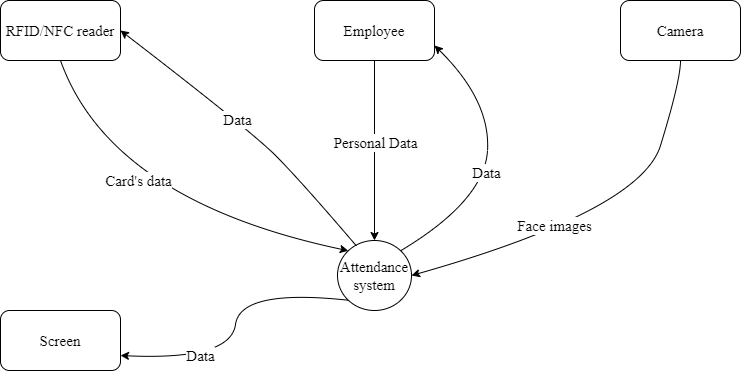
Σχήμα 3.3: Εικόνα προσώπου από το site <https://thispersondoesnotexist.com/>

Ο τρόπος εκπαίδευσης του δικτυού αυτού είναι βάζοντας δύο δίκτυα να ανταγωνιστούν μεταξύ τους το ένα καλείται γεννήτρια (generator) και το άλλο διάκριση (discriminator). Στην γεννήτρια δίνονται αληθινές φωτογραφίες τις οποίες το δίκτυο προσπαθεί να αναπαράγει όσο πιο πιστά μπορεί, ενώ το δίκτυο της διάκρισης μαθαίνει να ξεχωρίζει την αληθινή από την ψεύτικη. Με παρόμοιο τρόπο κατασκευάζονται και τα deepfake videos τα οποία έχουν την δυνατότητα να παραπλανήσουν τους περισσότερους[44]. Ο Wang κατασκεύασε το site για να γνωστοποιήσει σε όσο το δυνατόν περισσότερους ανθρώπους την τεχνολογία αυτή και τις δυνατότητες της. Για παράδειγμα, ένας κακόβουλος θα μπορούσε να διαδώσει ένα βίντεο ή μια εικόνα που δημιουργήθηκε από ένα GAN το οποίο απεικονίζει ένα ψεύτικο γεγονός ή τον Πρόεδρο/Πρωθυπουργό της χώρας να λέει στον λαό κάτι το οποίο θα μπορούσε να υποκινήσει ταραχές, διαμαρτυρίες ή άλλες δυνητικά βίαιες αντιδράσεις στο διαδίκτυο αλλά και στον πραγματικό κόσμο. Δεδομένου ότι η διαδικασία είναι όλη αυτοματοποιημένη, το μόνο που θα χρειαζόταν κάποιος θα ήταν πρόσβαση σε μια σειρά μονάδων επεξεργασίας γραφικών ή καρτών γραφικών (GPU) και σε ένα σύνολο δεδομένων εικόνων για να αρχίσει να βγάζει ψεύτικα videos και εικόνες.

# Κεφάλαιο 4 Λογισμικό

## 4.1 Λειτουργικές απαιτήσεις

### 4.1.1 Διάγραμμα γενικού πλαισίου



Σχήμα 4.1: Διάγραμμα περιβάλλοντος

### 4.1.2.1 Ανάγνωση – Εγγραφή κάρτας RFID

Περιγραφή: Το σύστημα είναι αναγκαίο να μπορεί να διαβάσει τα στοιχεία που βρίσκονται αποθηκευμένα σε μια κάρτα RFID. Επιπλέον πρέπει να είναι σε θέση να μπορεί να εγγράψει στοιχεία στις κάρτες.

Αιτιολόγηση: To σύστημα θα χρησιμοποίει την κάρτα RFID ως ένα πρώτο στάδιο για την είσοδο των υπαλλήλων στον χώρο εργασία τους.

### 4.1.2.2 Αποθήκευση των στοιχείων των υπαλλήλων

Περιγραφή: Η εφαρμογή θα αποθηκεύει στην βάση δεδομένων τα παρακάτω στοιχεία για κάθε χρήστη:

* UID (της RFID κάρτας που έχει)
* Ονοματεπώνυμο (θα είναι στα αγγλικά)
* Θέση εργασίας
* Τηλέφωνο
* Email
* Τομέας εργασίας
* Ημερομηνία γέννησης
* Ημερομηνία πρόσληψης

Αιτιολόγηση: Πρέπει να κρατούνται οι συγκεκριμένες πληροφορίες ώστε να μπορεί να γίνεται η αναγνώριση, η ενημέρωση τους και άλλες λειτουργίες που αφορούν τους χρήστες-εργαζομένους.

### 4.1.2.3 Εικόνα προσώπου των υπαλλήλων

Περιγραφή: Το σύστημα πρέπει να συλλέγει, αποθηκεύει και αν είναι αναγκαίο να βελτιστοποιεί την εικόνα του προσώπου του κάθε εργαζομένου.

Αιτιολόγηση: Οι παραπάνω λειτουργίες, συλλογής, αποθήκευσης και βελτιστοποίησης του προσώπου, είναι απαραίτητες για να είναι σε θέση η εφαρμογή να μπορεί να πραγματοποιήσει την αναγνώριση/ταυτοποίηση των εργαζομένων.

### 4.1.2.4 Εγγραφή εργαζομένου

Περιγραφή: Οι εργαζόμενοι θα πρέπει να εγγραφούν στο σύστημα παρέχοντας τα προσωπικά τους στοιχεία όπως αυτά αναγράφονται στην ενότητα 4.3.2 και μια φωτογραφία τύπου jpg η οποία θα λαμβάνεται από το σύστημα.

Αιτιολόγηση: Για να μπορεί να εισέρχεται ένας εργαζόμενος είναι υποχρεωτικό να ακολουθήσει τη διαδικασία της εγγραφής.

### 4.1.2.5 Ταυτοποίση εργαζομένου

Περιγραφή: Το σύστημα θα πρέπει να μπορεί να ταυτοποιεί τον εργαζόμενο κατά την είσοδο του στον χώρο εργασίας χρησιμοποιώντας τα στοιχεία από την κάρτα RFID και την εικόνα προσώπου.

Αιτιολόγηση: Για να μπορεί να εισέρχεται ένας εργαζόμενος είναι υποχρεωτικό να ακολουθήσει αυτή τη διαδικασία.

### 4.1.2.6 Καταγραφή στατιστικών εργαζόμενου

Περιγραφή: To σύστημα να μπορεί να καταγράψει τις παρουσίες των εργαζομένων, τους χρόνους εισόδου – εξόδου τους και τον χρόνο παραμονής στους χώρους της εταιρίας καθώς επίσης και τον τομέα στον οποίον δουλεύουν σε αρχεία τύπου csv. Ο τύπος για τον υπολογισμό του χρόνου παραμονής είναι:

Το αρχείο csv θα έχει την παρακάτω μορφή:

* UID χρήστη (λειτουργική απαίτηση 4.3.2)
* Ονοματεπώνυμο (λειτουργική απαίτηση 4.3.2)
* Τομέας εργασίας (λειτουργική απαίτηση 4.3.2)
* Ημερομηνία εισόδου
* Έτος
* Βδομάδα
* Ώρα εισόδου
* Ώρα εξόδου
* Χρόνος παραμονής

Αιτιολόγηση: Τα στατιστικά αυτά στοιχεία κρίνονται απαραίτητα για την ορθή λειτουργεία της επιχείρησης.

### 4.1.2.7 Έκδοση στατιστικών στοιχείων

Περιγραφή: Το σύστημα θα πρέπει να είναι σε θέση να εξάγει τα στατιστικά που κατέγραψε στην λειτουργική απαίτηση 4.3.6 συγκεντρωτικά και όχι για τον κάθε υπάλληλο ξεχωριστά σε βάθος βδομάδας, μήνα, χρόνου.

Αιτιολόγηση: Η εξαγωγή των στοιχείων είναι αναγκαία για τη σωστή λειτουργία της επιχείρησης.

##### 4.1.2.7.1 Υπολογισμοί ανά βδομάδα

όπου Χρόνος παραμονής είναι ο χρόνος παραμονής όπως αυτός ορίζεται στην λειτουργική απαίτηση 4.3.6

##### **4.1.2.7.2 Υπολογισμοί ανά μήν**α

##### 4.1.2.7.3 Υπολογισμοί ανά χρόνο

## 4.2 Προδιαγραφές

### 4.2.1 Προδιαγραφή 1

|  |  |
| --- | --- |
| Υπηρεσία ή Λειτουργία | Ανάγνωση – Εγγραφή κάρτας RFID |
| Περιγραφή | Το σύστημα πρέπει να μπορεί να διαβάσει τα αποθηκευμένα στοιχεία μιας RFID κάρτας |
| Δεδομένα εισόδου | Το μοναδικό αναγνωριστικό (UID) της κάρτας |
| Προέλευση | Αναγνώστης RFID |
| Δεδομένα εξόδου | Το μοναδικό αναγνωριστικό (UID) της κάρτας |
| Προορισμός | Βάση δεδομένων |
| Ενέργεια | O χρήστης πλησιάζει/ακουμπά την κάρτα πάνω στον αναγνώστη ώστε να αναγνώσει ή να εγγράψει τα απαραίτητα δεδομένα |
| Απαίτηση | Να λειτουργούν σωστά οι κάρτες και ο αναγνώστης |
| Προϋπόθεση ή Προσυνθήκη | Να είναι συνδεδεμένος και λειτουργικός αναγνώστης RFID/NFC με το σύστημα |
| Αποτέλεσμα | Εκτέλεση πρώτου βήματος για την ταυτοποίηση του χρήστη |
| Πλευρικά φαινόμενα | Μη δυνατότητα εισόδου και καταγραφής της παρουσίας του εργαζομένου |

### 4.2.2 Προδιαγραφή 2

|  |  |
| --- | --- |
| Υπηρεσία ή Λειτουργία | Αποθήκευση των στοιχείων των υπαλλήλων |
| Περιγραφή | Η εφαρμογή θα αποθηκεύει τις προσωπικές πληροφορίες του χρήστη στην βάση δεδομένων |
| Δεδομένα εισόδου | * UID (της RFID κάρτας) * Ονοματεπώνυμο * Θέση εργασίας * Τηλέφωνο * Email * Τομέας εργασίας * Ημερομηνία γέννησης * Ημερομηνία πρόσληψης |
| Προέλευση | Πληκτρολόγιο |
| Δεδομένα εξόδου | Φόρμα εγγραφής |
| Προορισμός | Βάση δεδομένων |
| Ενέργεια | Αφού ο χρήστης εισάγει τα δεδομένα, αυτά θα αποθηκευτούν στην βάση δεδομένων για διάφορες άλλες διεργασίες της εφαρμογής, όπως μελλοντική τροποποίηση ή διαγραφή τους |
| Απαίτηση | Να συμπληρωθούν σωστά τα πεδία της φόρμας |
| Προϋπόθεση ή Προσυνθήκη | Για κάθε χρήστη να υπάρχουν όλα τα απαιτούμενα στοιχεία |
| Αποτέλεσμα | Αποθήκευση των δεδομένων για κάθε νέο χρήστη ως νέα εγγραφή στην βάση δεδομένων |
| Πλευρικά φαινόμενα |  |

### 4.2.3 Προδιαγραφή 3

|  |  |
| --- | --- |
| Υπηρεσία ή Λειτουργία | Εικόνα προσώπου των υπαλλήλων |
| Περιγραφή | Το σύστημα θα συλλέγει, αποθηκεύει και αν είναι αναγκαίο θα βελτιστοποιεί την εικόνα του προσώπου κάθε εργαζομένου |
| Δεδομένα εισόδου | Φωτογραφία προσώπου εργαζομένου |
| Προέλευση | Κάμερα |
| Δεδομένα εξόδου |  |
| Προορισμός | Βάση δεδομένων |
| Ενέργεια | Ο χρήστης κάθεται μπροστά από την κάμερα και βγαίνει φωτογραφία |
| Απαίτηση | Ο χρήστης να δεχτεί να αποθηκευτεί η φωτογραφία του |
| Προϋπόθεση ή Προσυνθήκη | Να υπάρχει συνδεδεμένη και λειτουργική κάμερα |
| Αποτέλεσμα | Λήψη εικόνας του προσώπου του υπάλληλου και αποθήκευση της στο σύστημα για την πραγματοποίηση της ταυτοποίησης |
| Πλευρικά φαινόμενα | Μη δυνατότητα εισόδου και καταγραφής της παρουσίας του εργαζομένου |

### 4.2.4 Προδιαγραφή 4

|  |  |
| --- | --- |
| Υπηρεσία ή Λειτουργία | Εγγραφή εργαζομένου |
| Περιγραφή | Οι εργαζόμενοι θα πρέπει να εγγραφούν στο σύστημα παρέχοντας τα προσωπικά δεδομένα τους |
| Δεδομένα εισόδου | * UID (της RFID κάρτας) * Ονοματεπώνυμο * Θέση εργασίας * Τηλέφωνο * Email * Τομέας εργασίας * Ημερομηνία γέννησης * Ημερομηνία πρόσληψης * Εικόνα προσώπου |
| Προέλευση | Πληκτρολόγιο, κάμερα |
| Δεδομένα εξόδου | Φόρμα εγγραφής |
| Προορισμός | Βάση δεδομένων |
| Ενέργεια | Ο χρήστης εισάγει τα απαραίτητα στοιχεία και έπειτα βγάζει φωτογραφία το πρόσωπο του |
| Απαίτηση | Να δεχτεί ο χρήστης πως τα δεδομένα του θα τα διαχειρίζεται η εταιρία |
| Προϋπόθεση ή Προσυνθήκη | Να είναι συνδεδεμένα και λειτουργικά ένας αναγνώστης RFID/NFC και μια κάμερα |
| Αποτέλεσμα | Αποθήκευση νέου υπαλλήλου ως μια νέα εγγραφή στην βάση δεδομένων |
| Πλευρικά φαινόμενα | Μη δυνατότητα εισόδου και καταγραφής της παρουσίας του εργαζομένου |

### 4.2.5 Προδιαγραφή 5

|  |  |
| --- | --- |
| Υπηρεσία ή Λειτουργία | Ταυτοποίηση εργαζομένου |
| Περιγραφή | Το σύστημα λαμβάνοντας ως είσοδο τον μοναδικό αριθμό της κάρτας και το πρόσωπο του κάθε χρήστη θα κρατάει παρουσίες και στατιστικά στοιχεία |
| Δεδομένα εισόδου | * UID (της RFID κάρτας) * Εικόνα προσώπου * Δεδομένα από την βάση δεδομένων |
| Προέλευση | Πληκτρολόγιο, κάμερα |
| Δεδομένα εξόδου | Δεδομένα ταυτοποίησης |
| Προορισμός | Οθόνη εισόδου |
| Ενέργεια | Ο χρήστης αφήνει/ακουμπά την κάρτα στον αναγνώστη και κάθεται μπροστά από την κάμερα ώστε να αναγνωριστεί το πρόσωπο του |
| Απαίτηση | Να δεχτεί ο χρήστης πως τα δεδομένα του θα τα διαχειρίζεται η εταιρία |
| Προϋπόθεση ή Προσυνθήκη | Να είναι συνδεδεμένα και λειτουργικά ένας αναγνώστης RFID/NFC και μια κάμερα |
| Αποτέλεσμα | Είσοδος και καταγραφή παρουσίας τους εργαζομένου |
| Πλευρικά φαινόμενα | Μη δυνατότητα εισόδου και καταγραφής της παρουσίας του εργαζομένου |

### 4.2.6 Προδιαγραφή 6

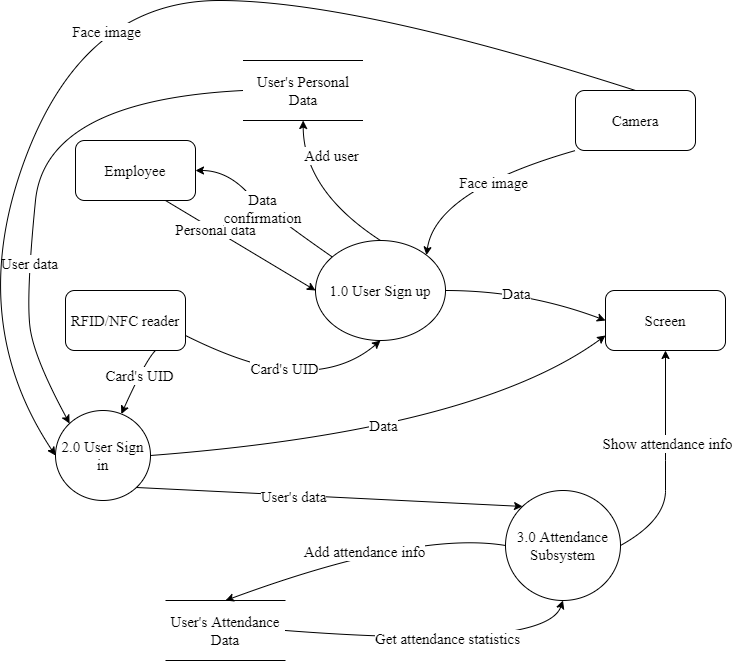
|  |  |
| --- | --- |
| Υπηρεσία ή Λειτουργία | Καταγραφή στατιστικών εργαζομένου |
| Περιγραφή | Το σύστημα θα μπορεί να καταγράψει τις παρουσίες των εργαζομένων, τους χρόνους εισόδου – εξόδου και τον χρόνο παραμονής των εργαζομένων στην εταιρία |
| Δεδομένα εισόδου |  |
| Προέλευση | Κάρτα RFID, βάση δεδομένων, σύστημα |
| Δεδομένα εξόδου | * UID χρήστη * Ονοματεπώνυμο * Τομέας εργασίας * Ημερομηνία εισόδου * Έτος * Βδομάδα * Ώρα εισόδου * Ώρα εξόδου * Χρόνος παραμονής |
| Προορισμός | Αρχείο csv |
| Ενέργεια | Καταγραφή παρουσιών και χρόνου παραμονής στους χώρους της εταιρίας |
| Απαίτηση | Να υπάρχουν όλα τα απαραίτητα στοιχεία |
| Προϋπόθεση ή Προσυνθήκη |  |
| Αποτέλεσμα | Δημιουργία και εξαγωγή παρουσιολογίου σε μορφή csv |
| Πλευρικά φαινόμενα | Πιθανότητα απώλειας στοιχείων |

### 4.2.7 Προδιαγραφή 7

|  |  |
| --- | --- |
| Υπηρεσία ή Λειτουργία | Έκδοση στατιστικών στοιχείων |
| Περιγραφή | Το σύστημα θα εξάγει τα στατιστικά στοιχεία που έχει καταγράψει |
| Δεδομένα εισόδου | Παρουσιολογίου σε μορφή csv |
| Προέλευση | Σύστημα |
| Δεδομένα εξόδου | Γραφήματα |
| Προορισμός | Οθόνη στατιστικών στοιχείων |
| Ενέργεια | Το σύστημα με χρήση των τύπων που παρουσιάζονται από την λειτουργική απαίτηση 4.4.7.1 μέχρι την 4.4.7.3 και με δεδομένα από τα csv αρχεία θα παράγει γραφήματα |
| Απαίτηση | Να έχουν δομηθεί σωστά τα csv |
| Προϋπόθεση ή Προσυνθήκη | Να υπάρχουν csv αρχεία |
| Αποτέλεσμα | Παρουσίαση συγκεντρωτικών στατιστικών στοιχείων σε γραφήματα |
| Πλευρικά φαινόμενα | Πιθανότητα απώλειας στατιστικών στοιχείων |

## 4.3 Διαγράμματα Ροής Δεδομένων

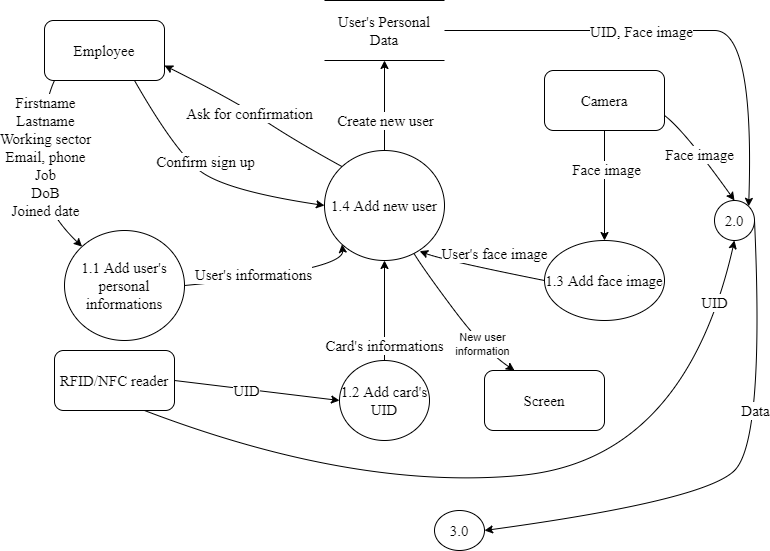
### 4.3.1 Διάγραμμα ροής δεδομένων 1ου επιπέδου



Σχήμα 4.2: Διάγραμμα ροής δεδομένων 1ου επιπέδου

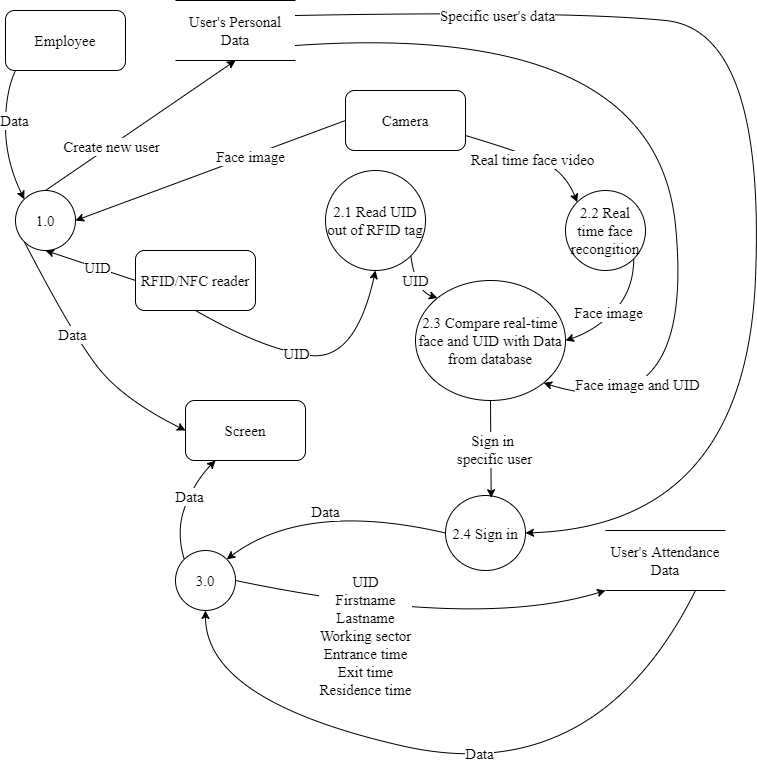
### 4.3.2 Διαγράμματα ροής δεδομένων 2ου επιπέδου

#### 4.3.2.1 Διάγραμμα ροής δεδομένων 2ου επιπέδου 1η διεργασία



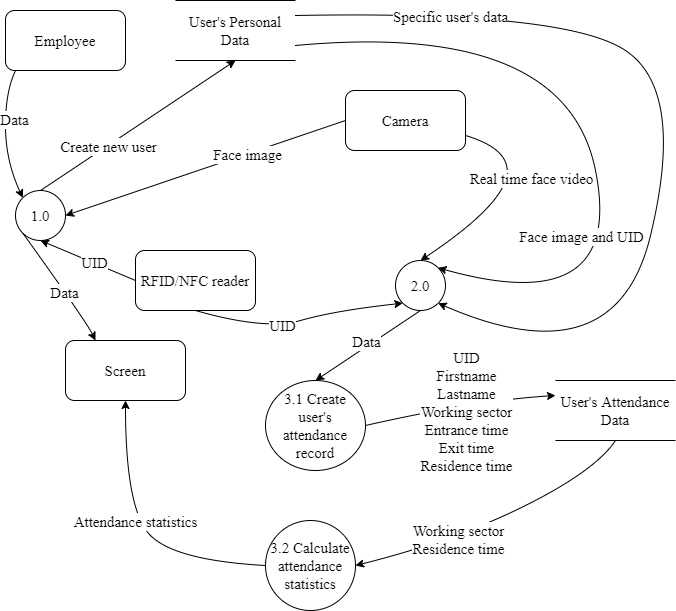
Σχήμα 4.3: Διάγραμμα ροής δεδομένων 2ου επιπέδου για εγγραφή χρήστη

#### 4.3.2.2 Διάγραμμα ροής δεδομένων 2ου επιπέδου 2η διεργασία



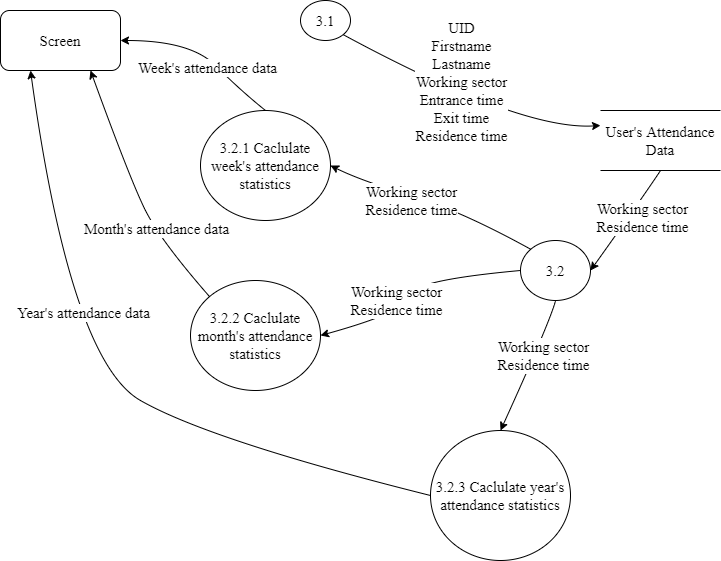
Σχήμα 4.4: Διάγραμμα ροής δεδομένων 2ου επιπέδου για είσοδο χρήστη στο σύστημα

#### 4.3.2.1 Διάγραμμα ροής δεδομένων 2ου επιπέδου 3η διεργασία



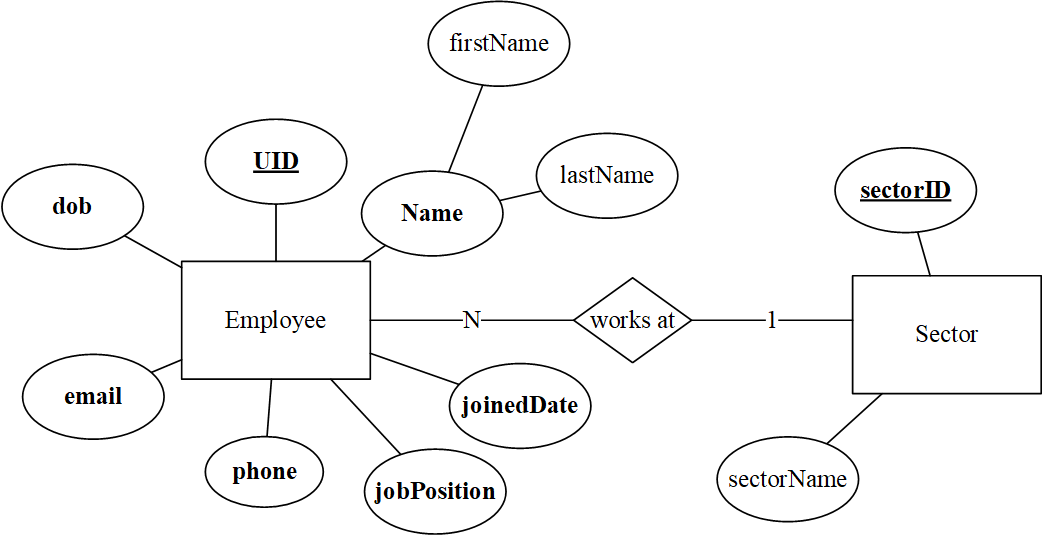
Σχήμα 4.5 Διάγραμμα ροής δεδομένων 2ου επιπέδου για παρουσιολόγιο

### 4.3.3 Διάγραμμα ροής δεδομένων 3ου επιπέδου



Σχήμα 4.6: Διάγραμμα ροής δεδομένων 3ου επιπέδου για προβολή στατιστικών

## 4.4 Διάγραμμα οντοτήτων – συσχετίσεων (ERD)



Σχήμα 4.7: Η βάση δεδομένων

### 4.4.1 Λεξικό δεδομένων

Πίνακας 4.1: Λεξικό δεδομένων

|  |  |  |  |
| --- | --- | --- | --- |
| Όρος | Τύπος | Περιγραφή | Ρόλος |
| Sector | Οντότητα | Περιέχει τους τομείς εργασίας |  |
| sectorID | Χαρακτηριστικό | Μοναδικός αριθμός τομέα | Κύριο κλειδί |
| sectorName | Χαρακτηριστικό | Όνομα τομέα | Απλή στήλη |
| Employee | Οντότητα | Περιέχει τους υπαλλήλους |  |
| UID | Χαρακτηριστικό | Κωδικός κάρτας RFID | Κύριο κλειδί |
| Name | Σύνθετο χαρακτηριστικό | Ονοματεπώνυμο υπαλλήλου |  |
| firstName | Χαρακτηριστικό | Όνομα υπαλλήλου | Απλή στήλη |
| lastName | Χαρακτηριστικό | Επώνυμο υπαλλήλου | Απλή στήλη |
| dob | Χαρακτηριστικό | Ημερομηνία γέννησης υπαλλήλου | Απλή στήλη |
| joinedDate | Χαρακτηριστικό | Ημερομηνία πρόσληψης | Απλή στήλη |
| phone | Χαρακτηριστικό | Τηλέφωνο υπαλλήλου | Απλή στήλη |
| email | Χαρακτηριστικό | Εταιρικό email | Απλή στήλη |
| jobPosition | Χαρακτηριστικό | Θέση εργασίας | Απλή στήλη |

### 4.4.2 Διατύπωση επιχειρησιακών κανόνων

Ένας τομέας έχει μοναδικό κωδικό (sectorID) και όνομα (sectorName).

Ένας υπάλληλος έχει μοναδικό κωδικό (UID), ένα ονοματεπώνυμο (firstName, lastName), ένα τηλέφωνο (phone), μια εταιρική διεύθυνση email (email), ημερομηνία γέννησης (dob), ημερομηνία που προσλήφθηκε στην εταιρία (joinedDate) και θέση εργασίας (jobPosition).

Ένας υπάλληλος μπορεί να δουλεύει σε έναν τομέα. Σε έναν τομέα εργάζονται πολλοί υπάλληλοι.

### 4.4.3 Σχήμα σχέσεων

Sector (sectorID, sectorName)

Κύριο κλειδί: sectorID

Employee (uid, firstName, lastName, phone, email, dob, joinedDate, jobPosition, sectorID)

Κύριο κλειδί: uid

Ξένο κλειδί: sectorID

### 4.4.4 Αντιπροσωπευτικό δείγμα

Πίνακας 4.2: Πίνακας τομέων

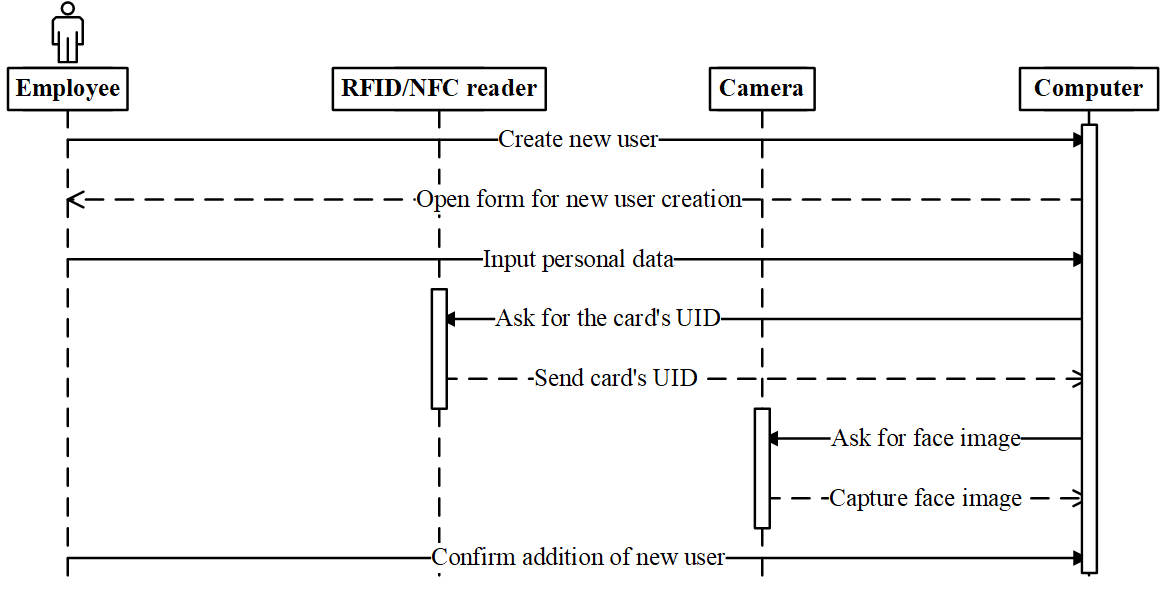
|  |  |
| --- | --- |
| sectorID | sectorName |
| 1 | Sector A |
| 2 | Sector B |
| 3 | Sector C |
| 4 | Sector D |

Πίνακας 4.3: Πίνακας υπαλλήλων

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| uid | firstName | lastName | phone | email | dob | joinedDate | jobPosition | sectorID |
| E05261A3 | Panagiotis | Sklidas | 6982523864 | pasklid@xyz.com | 14/09/2000 | 18/07/2022 | Software Engineer | 1 |
| 45F80C54 | John | Brown | 6974522174 | jobrown@xyz.com | 20/05/1990 | 28/02/2020 | Project Manager | 2 |
| EBBBC69E | Joe | Scott | 6920547856 | joscott@xyz.com | 30/11/1995 | 19/05/2019 | Developer | 1 |
| AC6F0A74 | Alex | Simons | 6999587120 | alsimon@xyz.com | 31/01/2001 | 05/05/2022 | Tester | 4 |

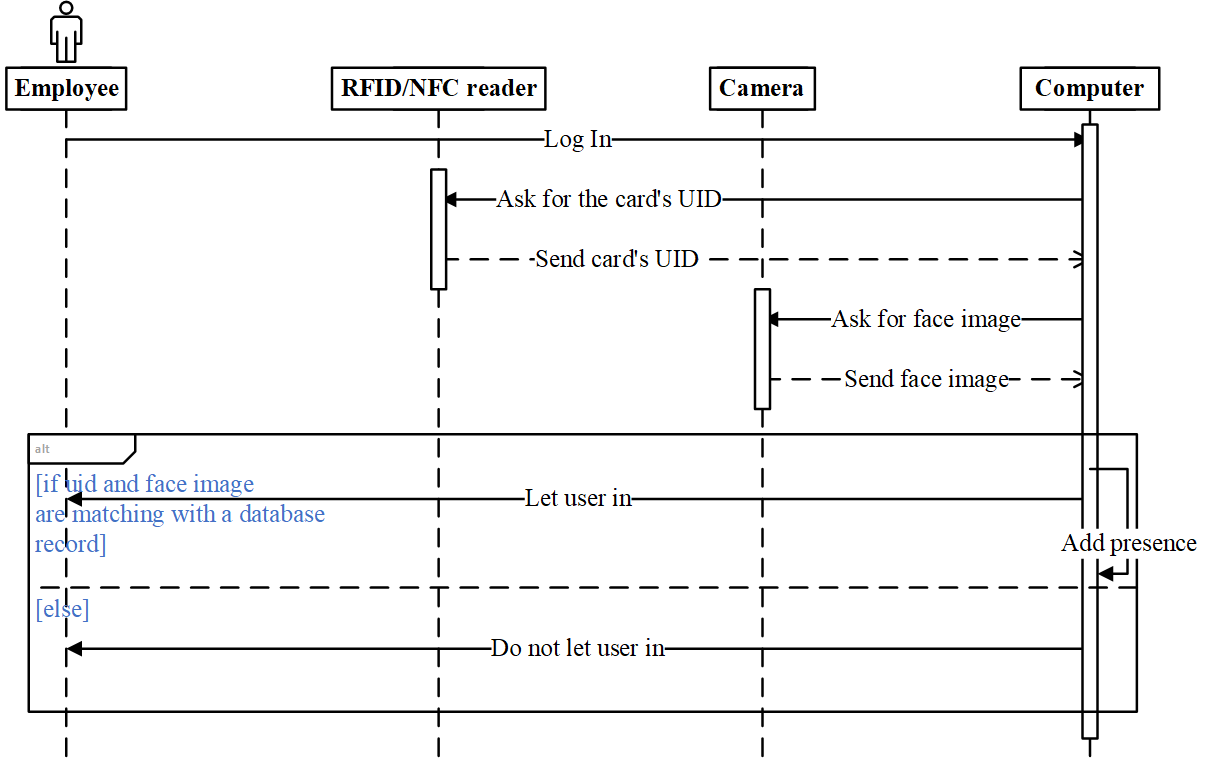
## 4.5 Διαγράμματα ακολουθίας

### 4.5.1 Διάγραμμα ακολουθίας για εγγραφή νέου χρήστη



Σχήμα 4.8: Διάγραμμα ακολουθίας για δημιουργία χρήστη

### 4.5.2 Διάγραμμα ακολουθίας για είσοδο χρήστη



Σχήμα 4.9: Διάγραμμα ακολουθίας εισόδου υπαρκτού χρήστη

## 4.6 Μη λειτουργικές απαιτήσεις

### 4.6.1 Λειτουργικό σύστημα

Περιγραφή: Η εφαρμογή θα υλοποιηθεί με γνώμονα την βέλτιστη απόδοση στις πιο πρόσφατες εκδόσεις των Windows, πιο συγκεκριμένα οι χρήστες που διαθέτουν Windows 10 ή νεότερη έκδοση του λειτουργικού της Microsoft δεν θα πρέπει να έχουν κανένα πρόβλημα κατά την εγκατάσταση και τη λειτουργία του προγράμματος.

Αιτιολόγηση: Το λειτουργικό σύστημα της Microsoft είναι το πιο διαδεδομένο στην αγορά υπολογιστών τόσο για προσωπική όσο και για επαγγελματική χρήση.

### 4.6.2 Δυνατότητα εγκατάστασης

Περιγραφή: Η εφαρμογή θα πρέπει να είναι έτσι δομημένη ώστε να μπορεί να εγκατασταθεί σε οποιοδήποτε υπολογιστικό σύστημα είτε είναι desktop είτε laptop. Για να επιτευχθεί αυτό η εφαρμογή θα συνοδεύεται από προσεκτικά και σωστά σχεδιασμένο λογισμικό εγκατάστασης και απεγκατάστασης.

Αιτιολόγηση: Παρέχοντας ένα αρχείο εγκατάστασης και απεγκατάστασης μπορεί κάθε χρήστης να χρησιμοποιήσει το πρόγραμμα όποιο και αν είναι το επίπεδο των γνώσεων του στους υπολογιστές.

### 4.6.3 Γλώσσα Διεπαφής

Περιγραφή: Η γλώσσα διεπαφής του προγράμματος θα είναι η αγγλική.

Αιτιολόγηση: Η εφαρμογή έχει αρκετά απλό UI και με την χρήση της αγγλικής γλώσσας μπορεί να χρησιμοποιηθεί από τους περισσότερους.

### 4.6.4 Ελάχιστη συχνότητα επεξεργαστή

Περιγραφή: H ελάχιστη απαιτούμενη συχνότητα επεξεργαστή είναι τα 2GHz. Η συγκεκριμένη συχνότητα είναι χαμηλότερη από αυτή που αναγράφεται στα συνοδευτικά έγγραφα σχεδόν κάθε σύγχρονου επεξεργαστή που βρίσκεται στο λιανικό εμπόριο, δίνοντας έτσι την δυνατότητα να δημιουργηθεί ένα σχετικά φτηνό σύστημα για την εγκατάσταση του λογισμικού.

Αιτιολόγηση: Η συχνότητα του επεξεργαστή είναι αναγκαία καθώς με χαμηλότερη υπάρχει η περίπτωση ο χρήστης να έρθει αντιμέτωπος με κολλήματα στην εφαρμογή ή να παρατηρήσει ότι το σύστημα του είναι αργό.

### 4.6.5 Ελάχιστη απαιτούμενη μνήμη RAM

Περιγραφή: H ελάχιστη απαιτούμενη μνήμη RAM που πρέπει να έχει το σύστημα είναι τα 4Gb. H εφαρμογή απαιτεί τουλάχιστον 4Gb διότι ταυτόχρονα με την ανάγνωση της κάρτας RFID το σύστημα θα χρησιμοποιεί αλγόριθμους για την ταυτοποίηση του εργαζομένου μέσω του προσώπου του, με το συγκεκριμένο μέγεθος μνήμης αναμένεται το πρόγραμμα αλλά και το σύστημα να λειτουργεί ομαλά.

Αιτιολόγηση: Η μνήμη αυτή είναι απαραίτητη ώστε να μην παρατηρήσει μεγάλες καθυστερήσεις κατά την εκτέλεση του προγράμματος.

### 4.6.6 Γλώσσα προγραμματισμού

Περιγραφή: Η εφαρμογή θα αναπτυχθεί στην προγραμματιστική γλώσσα Python έκδοση 3.9.6 στο PyCharm CE 2022.1.3.

Αιτιολόγηση: Η Python παρέχει την δυνατότητα γρήγορης ανάπτυξης κώδικα και επιπλέον σε αυτή είναι γραμμένοι αρκετοί αλγόριθμοι machine learning.

### 4.6.7 Βάση δεδομένων

Περιγραφή: Το πρόγραμμα θα αποθηκεύει τα δεδομένα σε βάση δεδομένων καθώς είναι αρκετά, επιπλέον με τον τρόπο αυτό ελαχιστοποιούνται οι πιθανότητες απώλειας τους.

Αιτιολόγηση: Η χρήση βάσης δεδομένων διευκολύνει τους χρήστες αφού παρέχουν τη δυνατότητα άμεσης προσθήκης, τροποποίησης και διαγραφής στοιχείων από την εφαρμογή.

### 4.6.8 Προστασία προσωπικών δεδομένων

Περιγραφή: Η εφαρμογή πρέπει να προστατεύει τα προσωπικά δεδομένα των χρηστών σύμφωνα με τις οδηγίες τις Ευρωπαϊκής Ένωσης όπως αυτές έχουν καθοριστεί και θεσμοθετηθεί στο έγγραφο GDPR (Κανονισμός ΕΕ 2016/679).

Αιτιολόγηση: Η εφαρμογή θα διαχειρίζεται τα προσωπικά δεδομένα των υπαλλήλων της εταιρίας και σύμφωνα με τους ισχύοντες κανονισμούς τις Ευρωπαϊκής Ένωσης για προγράμματά που έχουν πρόσβαση σε προσωπικές πληροφορίες είναι υπεύθυνη για την προστασία αυτών.

### 4.6.9 Συντηρησημότητα

Περιγραφή: Η εφαρμογή θα είναι δομημένη με τρόπο τέτοιο ώστε να επιτρέπει την αναβάθμιση και την συντήρηση της χωρίς να χρειάζεται πολύς χρόνος για τις ενέργειες αυτές.

Αιτιολόγηση: Μια επαγγελματική εφαρμογή πρέπει να είναι δομημένη με τρόπο τέτοιο ώστε να μπορεί να συντηρηθεί όχι μόνο από την εταιρία δημιουργίας της αλλά και από τρίτες σε περίπτωση που κάτι τέτοιο κριθεί αναγκαίο για παράδειγμα κλείσει η εταιρία ή λήξη το συμβόλαιο παροχής υπηρεσιών.

## 4.7 Code snippets



Σχήμα 4.10: Δημιουργία φακέλων και αρχείων

Στο παραπάνω κομμάτι κώδικα γίνεται η αρχικοποίηση της εφαρμογής. Η εντολή ελέγχει αν υπάρχει ο βασικός φάκελος της εφαρμογής σε περίπτωση που δεν υπάρχει δημιουργεί τον φάκελο, τον υποφάκελο στον οποίο θα αποθηκεύονται οι εικόνες προσώπου και το αρχείο παρουσιών.



Σχήμα 4.11: Encode of face images

Στο σχήμα 4.11 φαίνεται ο κώδικας ο οποίος είναι υπεύθυνος για την κωδικοποίηση των εικόνων. Συγκεκριμένα η εντολή:

φορτώνει την εικόνα προσώπου που αντιστοιχεί στον κωδικό της κάρτας που τοποθετήθηκε στον αναγνώστη. Στην επόμενη γραμμή καλείται η συνάρτηση για την εικόνα που φορτώθηκε η οποία υπολογίζει ένα πίνακα 128 διαστάσεων ο οποίος αποτελεί την κωδικοποιημένη εικόνα.



Σχήμα 4.12: Σύγκριση εικόνας εισόδου με κωδικοποιημένη εικόνα

Το σχήμα 4.12 είναι ο κώδικας της αναγνώριση προσώπου μέσω της webcam του συστήματος, ο οποίος εκτελείται κάθε φορά που ένας χρήστης θέλει να κάνει είσοδο στην εφαρμογή και κατ’ επέκταση στον “προστατευόμενο” χώρο. Η μέθοδος ξεκινάει με την κλήση της συνάρτησης

όπου cuid είναι ο μοναδικός κωδικός της έξυπνης κάρτας η οποία πρέπει να διαβαστεί από τον αναγνώστη και σχετίζεται άμεσα με την εικόνα προσώπου καθώς αποτελεί το όνομα της και με τον εργαζόμενο καθώς είναι ο κωδικός που τον ξεχωρίζει από τους υπόλοιπους για το σύστημα.

Έπειτα, περνάει τον πίνακα της κωδικοποιημένης εικόνας στην λίστα faces και το όνομα του αρχείο στην λίστα names. Με τις εντολές:

λαμβάνεται η εικόνα προσώπου από την webcam. Επειδή θέλουμε οι πράξεις όπως η κωδικοποίηση να γίνονται γρήγορα χρησιμοποιήθηκε η μέθοδος της βιβλιοθήκης OpenCV, η οποία παίρνει ως μεταβλητές την αρχική εικόνα (frame) και τις νέες διαστάσεις που θέλουμε να έχει (fx=0.25, fy=0.25) οι τιμές fx και fy δηλώνουν το ποσοστό που θα αλλάξει η αρχική εικόνα στην περίπτωση αυτή θα μικρύνει κατά 3/4 κατά ύψος και πλάτος. Η εντολή που ακολουθεί είναι αρκετά σημαντική καθώς χωρίς αυτή δεν θα μπορεί να γίνει η αναγνώριση. Η βιβλιοθήκη face\_recognition χρησιμοποιεί εικόνες σε RGB ενώ η OpenCV παράγει την εικόνα (frame) σε BGR, για τον λόγο αυτό χρησιμοποιείται η συνάρτηση

η οποία μετατρέπει μια εικόνα από BGR σε RGB. Στη συνέχεια εντοπίζονται τα πρόσωπα με τη μέθοδο

και κωδικοποιείται το πρόσωπο με την

.

Αφού ολοκληρωθεί η κωδικοποίηση της εικόνας γίνεται η σύγκριση της εικόνας από την webcam με την εικόνα που αντλήθηκε από τον φάκελο αποθηκευμένων εικόνων και πραγματοποιεί την θετική αναγνώριση αν η απόσταση μεταξύ των δύο εικόνων είναι κάτω του 0.6. Στην εντολή μπορεί να μπει και μια τρίτη μεταβλητή η οποία είναι η ελάχιστη απόσταση που πρέπει να έχουν οι δύο εικόνες για να πραγματοποιηθεί θετική αναγνώριση. Προτιμήθηκε να μην αλλαχθεί η προκαθορισμένη (default) απόσταση (0.6) διότι παρότι το σύστημα θα ήταν πιο ασφαλές με μια τιμή μεταξύ 0.4 και 0.5 δεν θα μπορούσε να ταυτοποιεί εικόνες από κινητό ή γενικότερα από κάποια οθόνη κάποιας άλλης συσκευής, το οποίο για την εξέλιξη της πτυχιακής κρίθηκε αναγκαίο. Ένας επιπλέον λόγος για να μην αλλαχθεί η ελάχιστη απόσταση δύο εικόνων είναι πως η αναγνώριση επηρεάζεται άμεσα από τον φωτισμό του χώρου.

Τέλος, ελευθερώνει την webcam ώστε να μπορεί να χρησιμοποιηθεί και από άλλες εφαρμογές ή άλλες λειτουργίες της εφαρμογής.

# Κεφάλαιο 5 Πειραματικά αποτελέσματα

## 5.1 Αναγνώριση έξυπνων καρτών

Η αναγνώριση των ετικετών RFID δουλεύει στο 100% των περιπτώσεων σωστά αρκεί να τηρούνται οι οδηγίες του κατασκευαστή καθώς και του πρωτοκόλλου NFC. Περιπτώσεις όπως είναι η τοποθέτηση πολλαπλών ετικετών ταυτόχρονα πάνω στον αναγνώστη ή η κάρτα να βρίσκεται σε απόσταση μεγαλύτερη των 4 εκατοστών από την επιφάνεια το αναγνώστη θα οδηγήσουν το ποσοστό επιτυχούς αναγνώρισης/ανάγνωσης από 100% σε 0%.



Σχήμα 5.1 : Τοποθέτηση 2 καρτών στον αναγνώστη

## 5.2 Αναγνώριση προσώπου

Για να πραγματοποιηθεί η αναγνώριση προσώπου σε ένα dataset η πιο συνηθισμένη πρακτική είναι η κωδικοποίηση όλων των εικόνων και έπειτα η σύγκριση της κωδικοποιημένης εικόνας εισόδου με όλες τις κωδικοποιημένες γνωστές εικόνες. Αυτό σημαίνει πρακτικά πως όσο μεγαλύτερο είναι το σύνολο των γνωστών εικόνων, τόσο περισσότερο χρόνο θα χρειαστεί το σύστημα για να ολοκληρώσει την αναγνώριση. Για παράδειγμα στο πρόγραμμα που δημιουργήθηκε, δημιουργήθηκε ένα dataset περίπου 100 φωτογραφιών (βλέπε Ενότητα 3.5 για τον τρόπο δημιουργίας του dataset) και έτρεξα την μέθοδο αναγνώρισης. Το σύστημα με αναγνώρισε ναι μεν με επιτυχία αλλά σχετικά αργά συγκεκριμένα χρειάστηκε 1 λεπτό 27 δευτερόλεπτα και 709 εκατοστά (00:01:27.709) για να ολοκληρώσει τη διαδικασία από τη στιγμή που κλήθηκε η συνάρτηση μέχρι τη στιγμή που με αναγνώρισε. Με σκεπτικό πως ένα dataset ενός εργοστασίου ή ακόμα και ενός εργαστηριακού μαθήματος σε μια σχολή θα μπορούσε να περιέχει από μερικές δεκάδες μέχρι και μερικές εκατοντάδες εικόνες φανερώθηκε πως υπάρχουν πολλά περιθώρια βελτίωσης στον χρόνο αναγνώρισης. Έτσι τροποποιήθηκε η διαδικασία αναγνώρισης και αποδείχθηκε πως θα ήταν πιο αποδοτικό αν χρησιμοποιούνταν ο μοναδικός κωδικός κάθε έξυπνης κάρτας για να κωδικοποιείται μια εικόνα από το dataset την φορά. Με την συγκεκριμένη υλοποίηση ο χρόνος που χρειαζόταν η εφαρμογή για να αναγνωρίσει ένα πρόσωπο μειώθηκε δραματικά. Μετά από μετρήσεις με ίδιο μέγεθος δεδομένων όπως και πριν, ο χρόνος που χρειάστηκε το σύστημα για να κάνει την ταυτοποίηση στην χειρότερη περίπτωση άγγιζε τα 9 με 10 δευτερόλεπτα τις περισσότερες όμως φορές ήταν στα 6 με 7 (00:00:07.377) δευτερόλεπτα.

# Κεφάλαιο 6 Συμπεράσματα

Η ανάπτυξη κώδικα στα πλαίσια υλοποίησης της παρούσας πτυχιακής εργασίας με έκανε να διαπιστώσω πόσο αποτελεσματική και γρήγορη είναι η δημιουργία κώδικα σε Python καθώς μέσα σε λίγες μόλις γραμμές κώδικα είναι σε θέση να κάνει τις ίδιες λειτουργίες, οι οποίες σε άλλες γλώσσες όπως η C# ή η Java θα χρειαζόταν παραπάνω. Το γεγονός αυτό, οφείλεται στο ότι η Python χρησιμοποιείται από ομάδες χρηστών όπως οι αναλυτές δεδομένων (data analyst), οι οποίες δεν έχουν την ίδια ευχέρεια στην συγγραφή κώδικα όπως οι προγραμματιστές και θέλουν ένα εργαλείο το οποίο να μπορεί σε λίγες γραμμές να λύσει το πρόβλημα τους. Παρόλα αυτά, η δημιουργία μιας ολοκληρωμένης εφαρμογής με UI στην γλώσσα αυτή αποδείχθηκε πιο δύσκολη από το αναμενόμενο. Σε αντίθεση με την C# που μέσω του Visual Studio μπορεί να παράγει τον απαιτούμενο κώδικα για το UI καθώς ο IDE έχει τη λειτουργία του designer, όπου ο προγραμματιστής μπορεί να σύρει σε μια οθόνη όλα τα απαραίτητα στοιχεία του UI και το πρόγραμμα να δημιουργήσει τον απαραίτητο κώδικα αυτόματα. Στην Python ο προγραμματιστής πρέπει να σχεδιάσει μόνος του και εξολοκλήρου από το μηδέν το UI της εφαρμογής σε κώδικα μέσω κάποιον module που διατίθενται για αυτή τη δουλεία όπως το Tkinter ή το PyQt.

Τα τελευταία δύο χρόνια η καθημερινότητα μας άλλαξε σε αρκετά σημεία. Ένα από αυτά είναι ο τρόπος πρόσβασης μας σε διάφορους κλειστούς χώρους και η ανάγκη για την καταγραφή του χρόνου παραμονής μας σε αυτούς. Έτσι, το σύστημα αυτό θα μπορούσε να χρησιμοποιηθεί στην είσοδο πανεπιστημίων, γυμναστηρίων και χώρων εργασίας ώστε να καταγράφεται ο χρόνος που βρισκόταν μέσα στον χώρο ο κάθε χρήστης και σε περίπτωση κάποιου κρούσματος να ενημερωθούν μόνο όσοι ήταν τις ίδιες ώρες.

Μελλοντικά η εφαρμογή θα μπορούσε να δείχνει περισσότερα στατιστικά στοιχεία όπως ποιος χρήστης παρέμεινε τον περισσότερο χρόνο στον χώρο, πόσες φορές κάποιος μπήκε ή βγήκε από αυτόν. Επιπλέον, θα μπορούσε να γίνει πολύ γλωσσική ώστε να είναι ακόμα πιο εύκολη η χρήση της εφαρμογής από χρήστες που πιθανών να μην γνωρίζουν καλά αγγλικά. Τέλος, θα μπορούσε η εφαρμογή να επανασχεδιαστεί ώστε να είναι native για το λειτουργικό σύστημα των Windows χρησιμοποιώντας την C# αφού οι περισσότερες εταιρίες χρησιμοποιούν το λειτουργικό της Microsoft.

# Παράρτημα

## Παράρτημα 1 Script άντλησης εικόνων προσώπου



## Παράρτημα 2 User manual

### 1 Εισαγωγή

Διαβάστε αυτό το εγχειρίδιο πριν χρησιμοποιήσετε την εφαρμογή λογισμικού για να διασφαλίσετε την ορθή λειτουργία της.

Ζητήθηκε η δημιουργία μιας windows desktop εφαρμογής λογισμικού για την διαχείριση της εισόδου – εξόδου στον χώρο εργασίας και την προβολή στατιστικών στοιχείων για το σύνολο των εργαζομένων μιας εταιρίας. Η λύση η οποία υλοποιήθηκε για το έργο αυτό ήταν η παραγωγή της εφαρμογής λογισμικού “Advanced Attendance System” (από εδώ και στο εξής η «Εφαρμογή»). Η εφαρμογή είναι εξολοκλήρου στα αγγλικά με απλό και μοντέρνο γραφικό περιβάλλον, με τον εξής περιορισμό: ανήκει στην κατηγορία desktop application που σημαίνει πως οι πληροφορίες της θα αποθηκεύονται τοπικά στον υπολογιστή που έχει εγκατασταθεί και δεν θα είναι προσβάσιμες μέσω διαδικτύου ή άλλου υπολογιστή.

Για να μπορεί να εγκατασταθεί η εφαρμογή και να έχει την βέλτιστη απόδοση θα πρέπει ο υπολογιστής στον οποίο θα εγκατασταθεί να πληροί τα παρακάτω τεχνικά χαρακτηριστικά:

Ελάχιστα:

Λειτουργικό σύστημα: Windows 10 x64

Επεξεργαστής: Intel Core i3-7100 @ 3.90GHz

Κάρτα γραφικών: Αδιάφορο

RAM: 4.00 GB

Αποθηκευτικός Χώρος: 128 GB

Προτεινόμενα:

Λειτουργικό σύστημα: Windows 10 x64

Επεξεργαστής: Intel Core i7-8750h @ 2.20GHz

Κάρτα γραφικών: Nvidia GTX 1050Ti

RAM: 8.00 GB

Αποθηκευτικός Χώρος: 512 GB

Σημειώσεις, και προειδοποιήσεις

 Σημείωση: Μια σημείωση υποδηλώνει μια σημαντική πληροφορία που θα σας βοηθήσει να κάνετε καλύτερη χρήση της εφαρμογής.

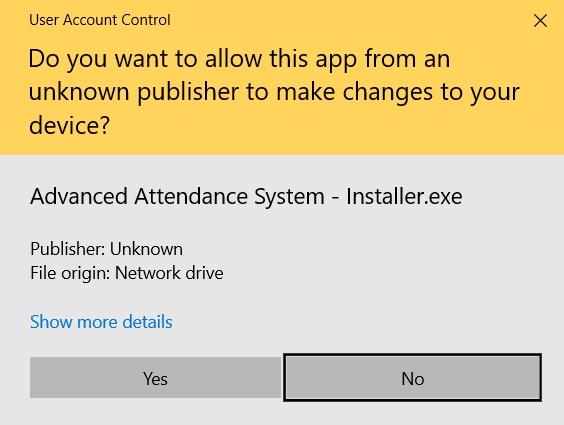
 Προειδοποίηση: Μια προειδοποίηση υποδηλώνει κάποια πιθανή απώλεια δεδομένων και προτείνει πιθανή λύση.

### 2 Εγκατάσταση

Τα αρχεία τα οποία παρέχουμε ως πακέτο, είναι δύο: α) το πρόγραμμα εγκατάστασης και β) το πρόγραμμα απεγκατάστασης.

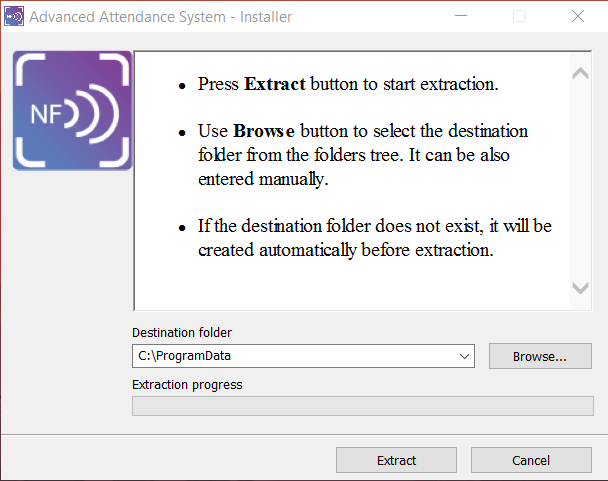
Βήμα 1: Για να εγκαταστήσετε το προϊόν μας θα χρειαστεί να τρέξετε το πρόγραμμα εγκατάστασης το οποίο ονομάζεται “Advanced Attendance System - Installer.exe”.

Βήμα 2: Θα σας ζητηθεί να παρέχετε δικαιώματα διαχειριστή για την εγκατάσταση. Επιλέξτε “Ναι” για να συνεχίσει η εγκατάσταση.



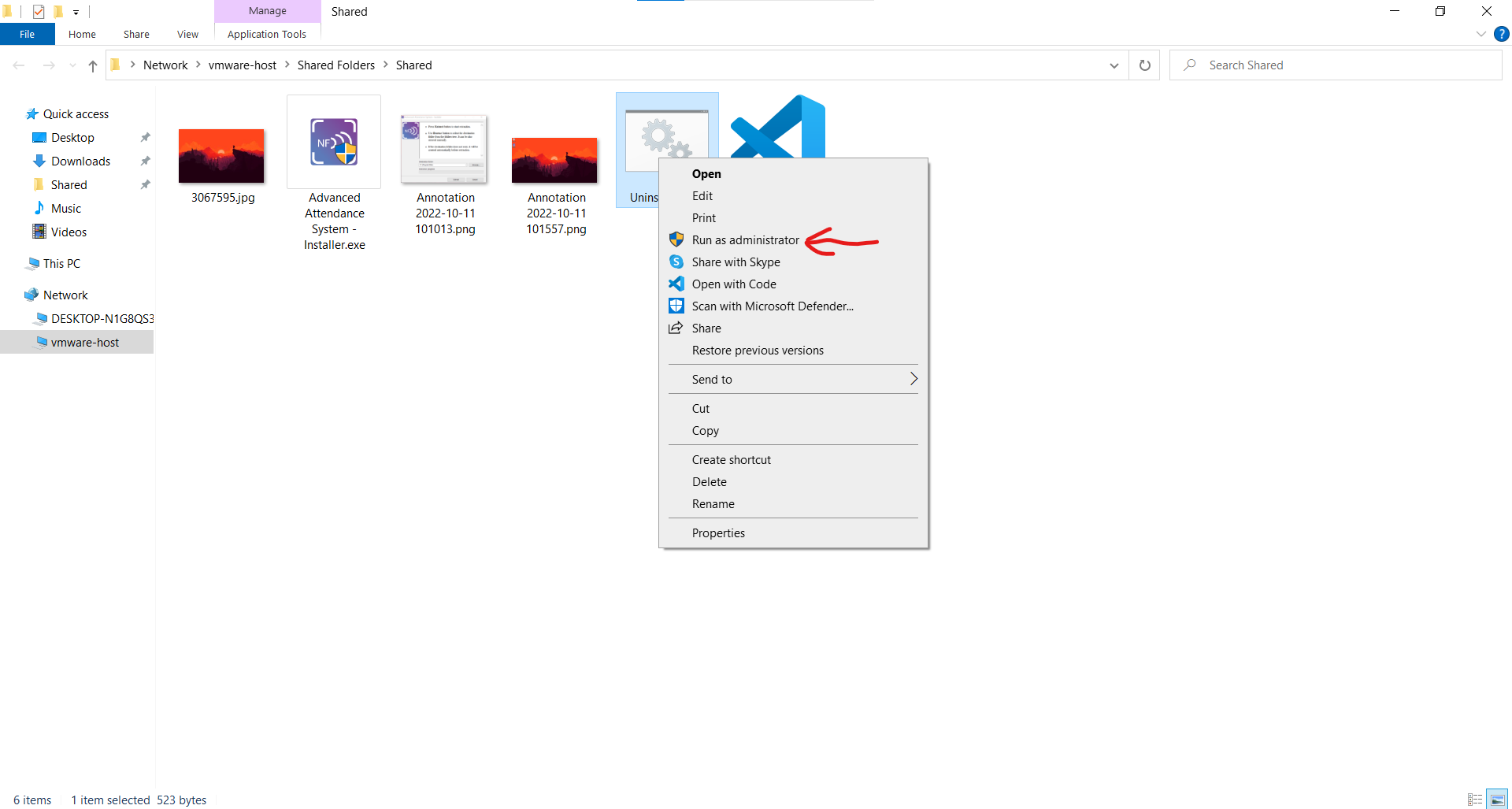
Σχήμα 2.1: Παραχώρηση δικαιωμάτων διαχειριστή

Βήμα 3: Έπειτα θα σας εμφανιστεί ένα νέο παράθυρο το οποίο είναι ο “Installation Wizard”. Πατήστε εγκατάσταση για να ολοκληρωθεί η εγκατάσταση του προγράμματος. Αν δεν εμφανιστεί κάποιο μήνυμα λάθους σημαίνει πως η εγκατάσταση έχει ολοκληρωθεί επιτυχώς. Το πρόγραμμα εγκατάστασης δημιουργεί αυτόματα συντομεύσεις στην επιφάνεια εργασίας και έναν φάκελο στον τοπικό δίσκο C ώστε να αποθηκεύει την βάση δεδομένων, τις εικόνες προσώπου και το έγγραφο παρουσιών.



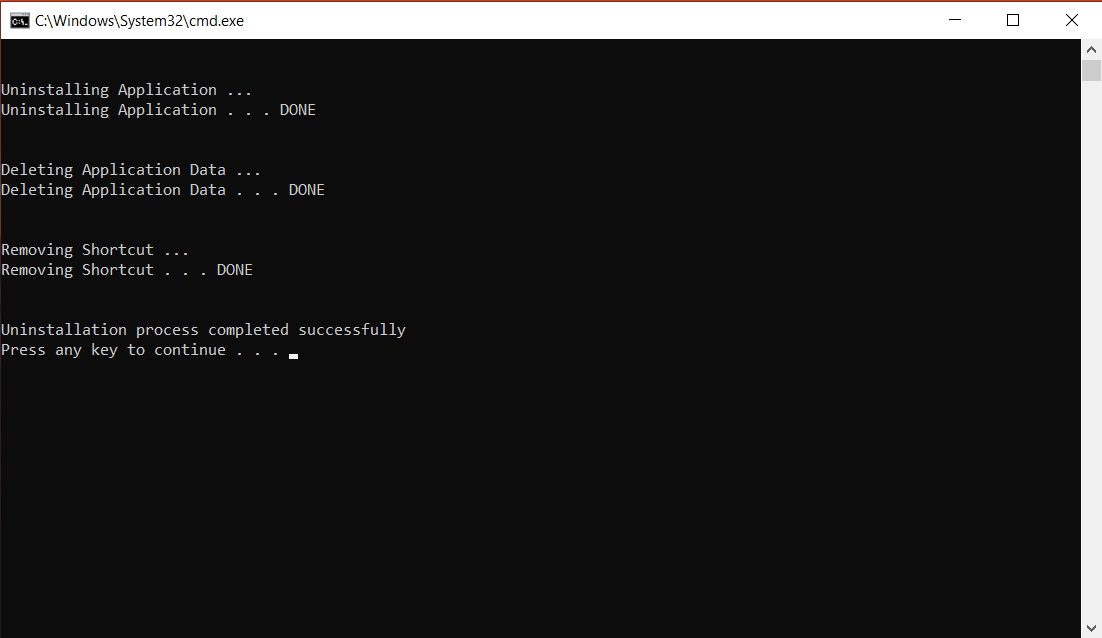
Σχήμα 2.2: Εγκατάσταση της εφαρμογής

### 3 Απεγκατάσταση

Βήμα 1: Για να εκτελέσετε το πρόγραμμα απεγκατάστασης με όνομα “Uninstaller.bat” θα κάνετε δεξί κλικ πάνω στο αρχείο και έπειτα “Εκτέλεση ως διαχειριστής”. 

Σχήμα 3.1: Απεγκατάσταση εφαρμογής

Βήμα 2: Όλα θα γίνουν αυτόματα και απλά περιμένετε να εμφανίσει το μήνυμα “Uninstallation process completed successfully”. Όταν εμφανιστεί αυτό το μήνυμα μπορείτε να κλείσετε το παράθυρο καθώς η απεγκατάσταση θα έχει ολοκληρωθεί με επιτυχία.



Σχήμα 3.1: Επιτυχής απεγκατάσταση εφαρμογής

### 4 Πως να χρησιμοποιήσετε την εφαρμογή

Η εφαρμογή έχει δύο κατηγορίες χρηστών, από τις οποίες μόνο οι χρήστες που ανήκουν στο τμήμα ανθρώπινου δυναμικού (Human Resources) των εταιριών έχουν την δυνατότητα να επεξεργαστούν (τροποποιήσουν, διαγράψουν) άλλους εργαζομένους, να επεξεργαστούν (τροποποιήσουν, διαγράψουν, προσθέσουν) τους τομείς (sectors) εργασίας της εταιρίας και τέλος έχουν την δυνατότητα να δουν τα στατιστικά στοιχεία. Η άλλη κατηγορία περιέχει όλους τους υπόλοιπους εργαζομένους, οι οποίοι θα μπορούν μόνο να κάνουν είσοδο και έξοδο από τον εργασιακό τους χώρο.

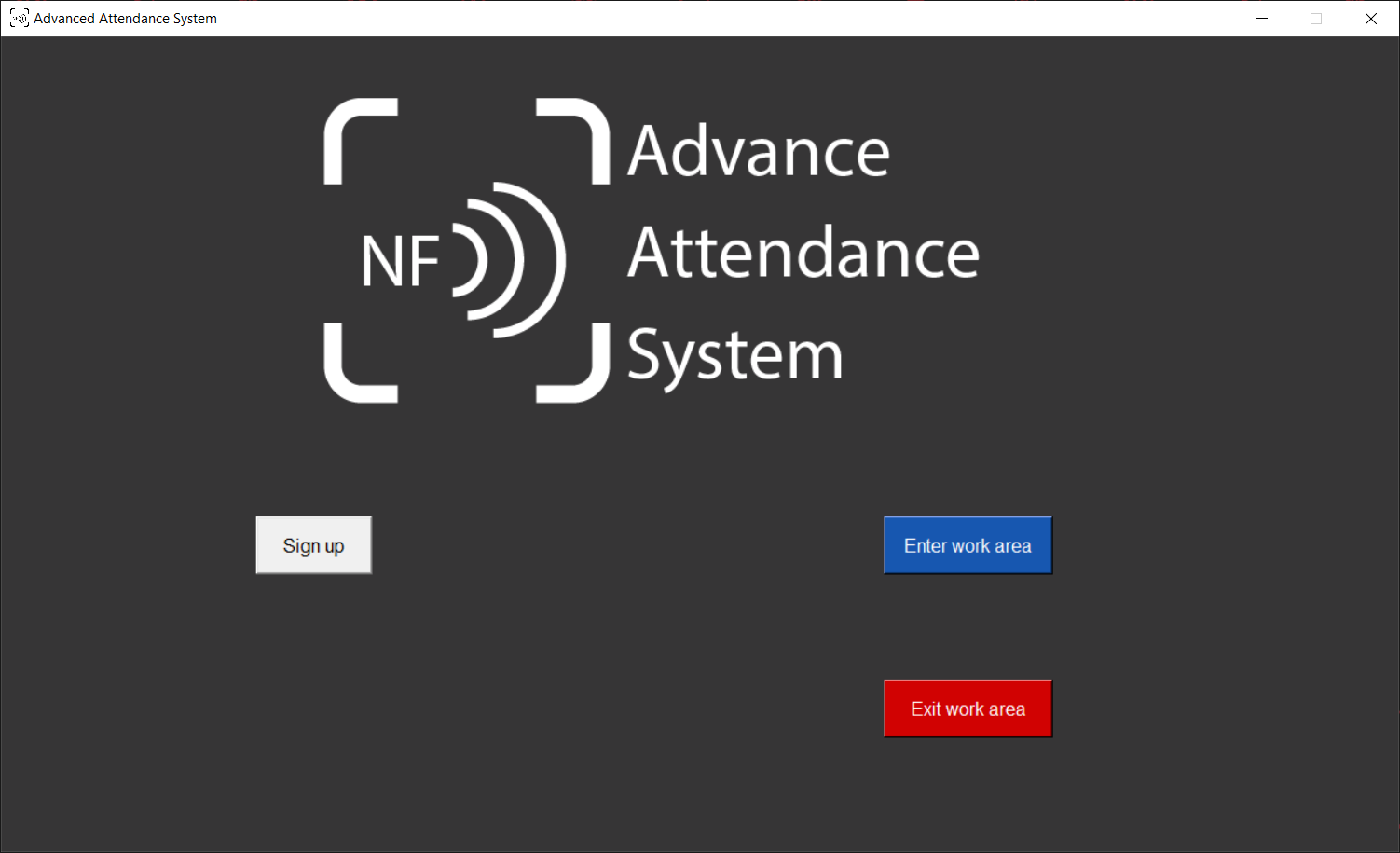
#### 4.1 Άνοιγμα εφαρμογής

Βήμα 1: Επιλέξτε το εικονίδιο της εφαρμογής που θα βρείτε στην επιφάνεια εργασίας του υπολογιστή σας.



Σχήμα 4.1.1: Άνοιγμα εφαρμογής

Βήμα 2: Ο χρήστης καλωσορίζεται από την αρχική οθόνη του προγράμματος



2

3

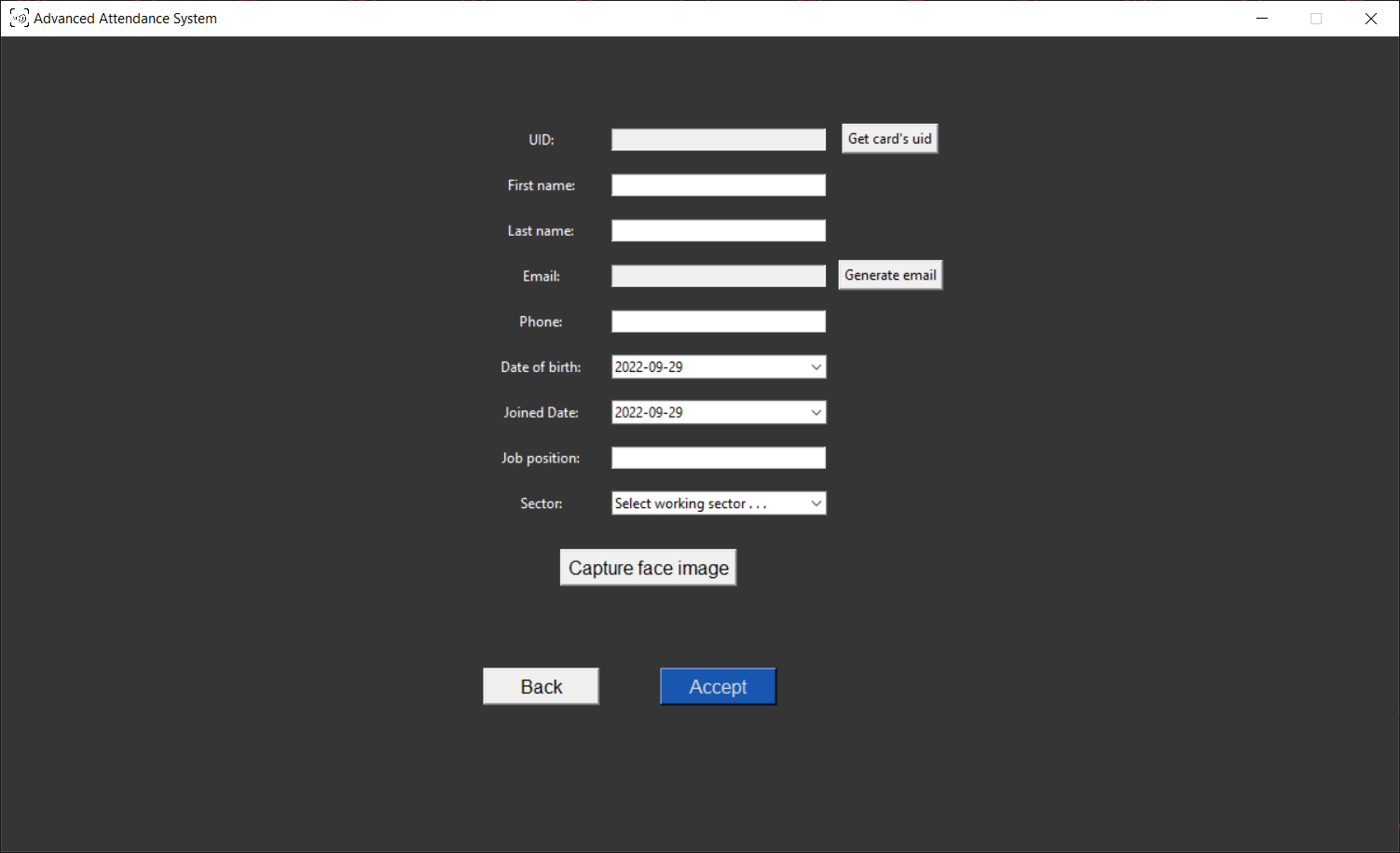
1

Σχήμα 4.1.2: Αρχική οθόνη

#### 4.2 Εγγραφή νέου χρήστη

Για να εγγραφή ο εργαζόμενος στην εφαρμογή πρέπει να ακολουθήσει τα εξής βήματα:

Βήμα 1: Επιλέγει με το ποντίκι το κουμπί 1 (λευκό κουμπί “Sign up”) του σχήματος 4.1.2 με αυτόν το τρόπο θα οδηγηθεί στην οθόνη εγγραφής που φαίνεται στο σχήμα 4.2.1



3

2

1

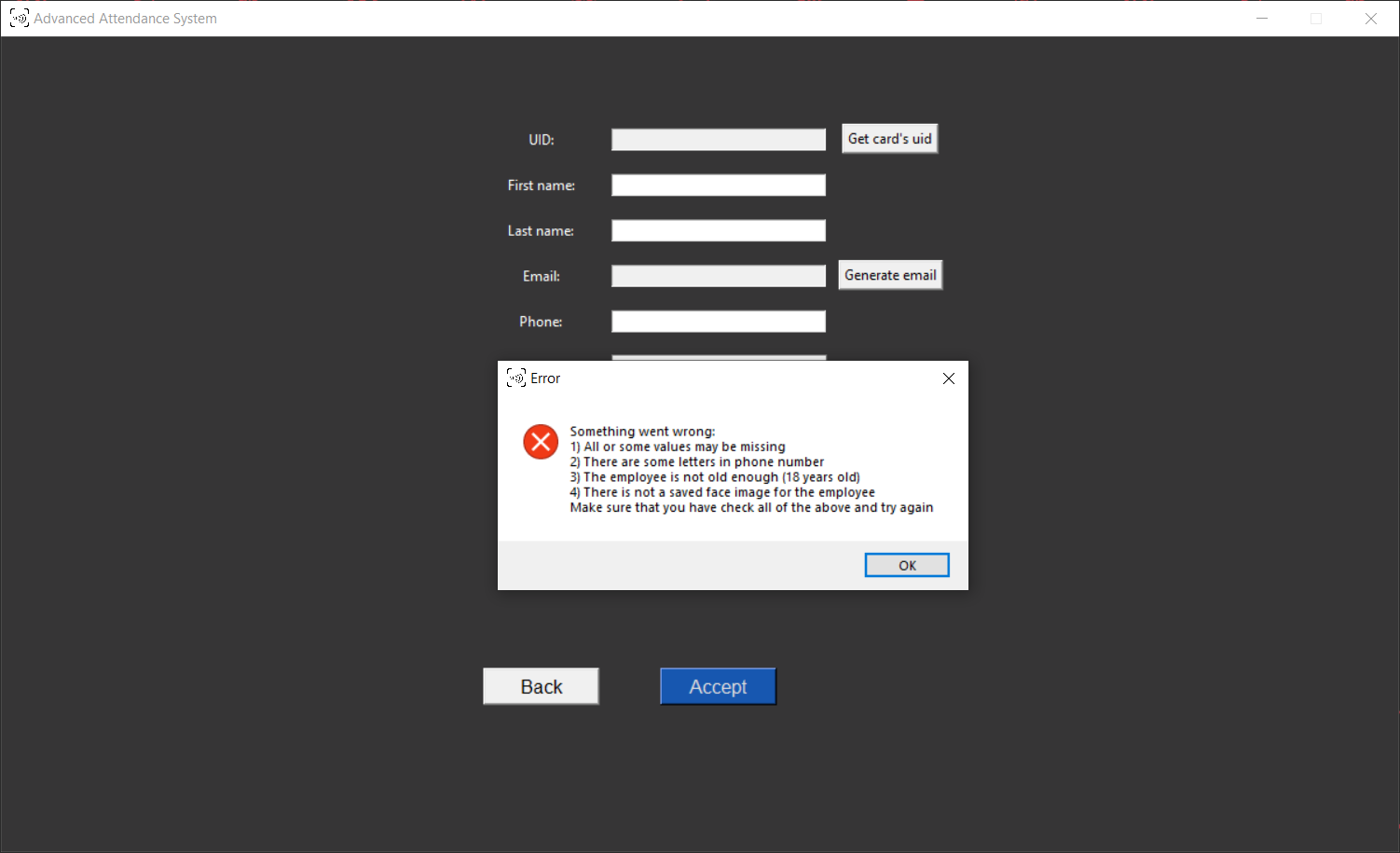
Σχήμα 4.2.1: Οθόνη εγγραφής

Βήμα 2: Αφού εναποθέσει την κάρτα RFID, που θα του δοθεί, πάνω στον ειδικό αναγνώστη πρέπει να πατήσει το κουμπί 1 (σχήμα 4.2.1 “Get card’s uid”).

Βήμα 3: Ο χρήστης συμπληρώνει τις προσωπικές του πληροφορίες στα πεδία.

 Σημείωση 4.2.1: Αφού συμπληρώσει το όνομα και το επίθετο του ο χρήστης πρέπει να πατήσει το κουμπί 2 (σχήμα 4.2.1 “Generate email”) ώστε η εφαρμογή να δημιουργήσει αυτόματα το εταιρικό του email.

 Σημείωση 4.2.2: Η εφαρμογή κάνει αυτόματο έλεγχο για το αν ο χρήστης/εργαζόμενος είναι τουλάχιστον 18 χρονών. Ως ενήλικας για την εφαρμογή ορίζεται ένας χρήστης αν η διαφορά της ημερομηνίας πρόσληψης (Joined Date) είναι μεταγενέστερη κατά 18 χρόνια από την ημερομηνία γέννησης (Date of birth) που θα δηλωθεί στο πρόγραμμα. Σε περίπτωση που η προϋπόθεση αυτή δεν ισχύει το σύστημα βγάζει ένα μήνυμα λάθους και δεν ολοκληρώνει την εγγραφή σχήμα 4.2.2.



Σχήμα 4.2.2: Σφάλμα κατά την εγγραφή

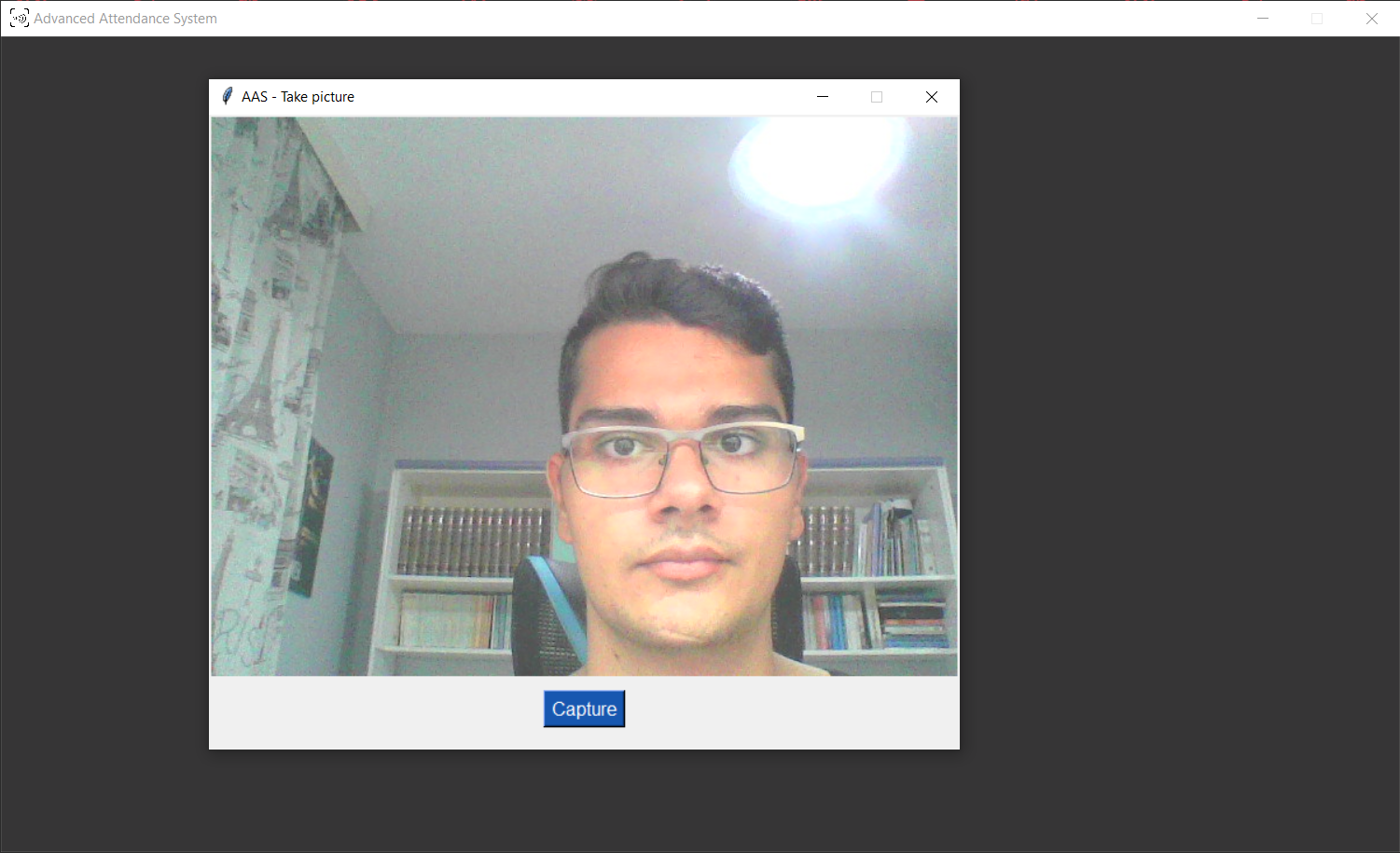
Βήμα 4: Ο χρήστης αφού συμπληρώσει όλα τα προσωπικά του στοιχεία πρέπει να επιλέξει το κουμπί 3 (στο σχήμα 4.2.1 “Capture face image”) ώστε να αποθηκεύσει το πρόγραμμα την εικόνα προσώπου του με την οποία θα γίνεται η ταυτοποίηση.

 Προειδοποίηση 4.2.1: Για να ξεκινήσει η λήψη της φωτογραφίας πρέπει ο χρήστης να έχει ολοκληρώσει επιτυχώς το βήμα 2. Σε διαφορετική περίπτωση θα του εμφανιστεί το παρακάτω μήνυμα (σχήμα 4.2.3).



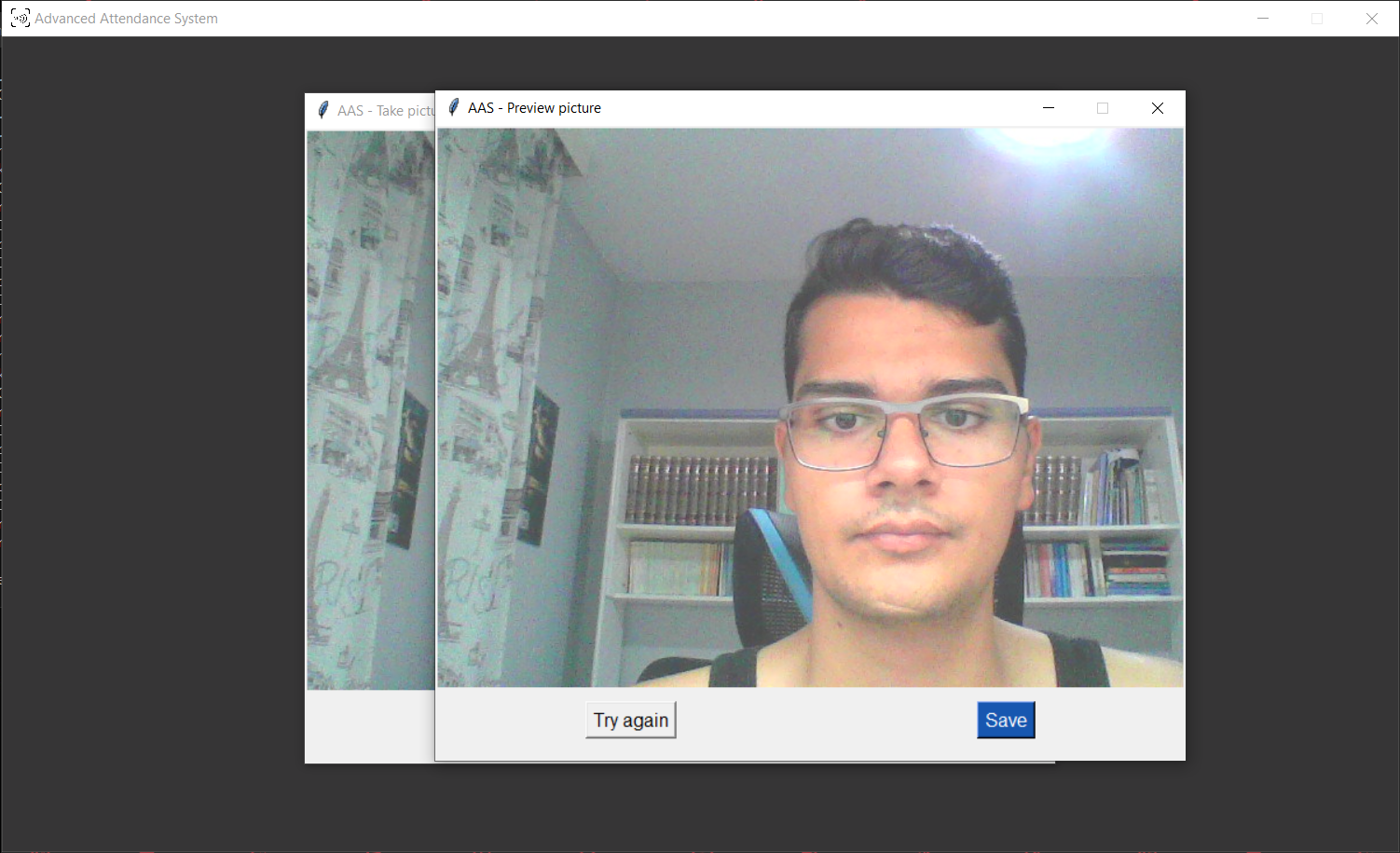
Σχήμα 4.2.3: Σφάλμα κατά την λήψη της φωτογραφίας

Η εφαρμογή θα ανοίξει ένα αναδυόμενο παράθυρο (pop-up window σχήμα 4.2.4) στο οποίο ο χρήστης θα βλέπει την κάμερα και θα μπορεί να επιλέξει “Capture” ώστε να βγει η φωτογραφία.



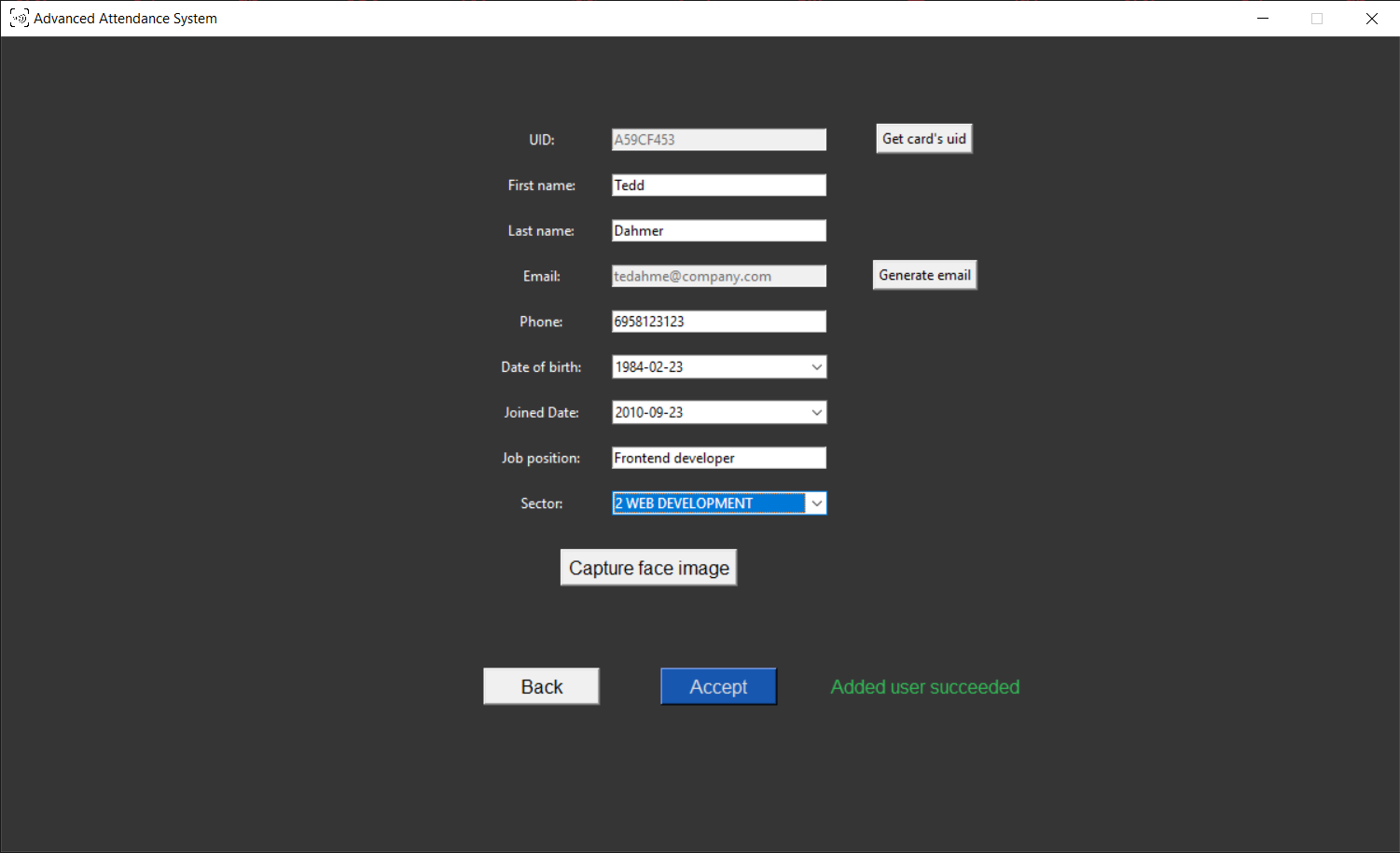
Σχήμα 4.2.4: Camera feed

Έπειτα, η εφαρμογή θα προβάλει την φωτογραφία που έβγαλε και επιτρέπει στον χρήστη να την αποθηκεύσει πατώντας το κουμπί “Save” στο σχήμα 4.2.5 ή αν θέλει να βγάλει άλλη πατώντας το “Try again” κουμπί.



Σχήμα 4.2.5: Προβολή φωτογραφίας

Βήμα 5: Αφού ολοκληρωθούν τα βήματα 2 μέχρι 4 με επιτυχία ο χρήστης μπορεί να επιλέξει το μπλε κουμπί του σχήματος 4.2.1 (“Accept”) ώστε να ολοκληρωθεί η εγγραφή του (σχήμα 4.2.6). Σε περίπτωση που για κάποιο λόγο δεν είναι επιτυχής η εγγραφή θα εμφανιστεί το μήνυμα του σχήματος 4.2.2.

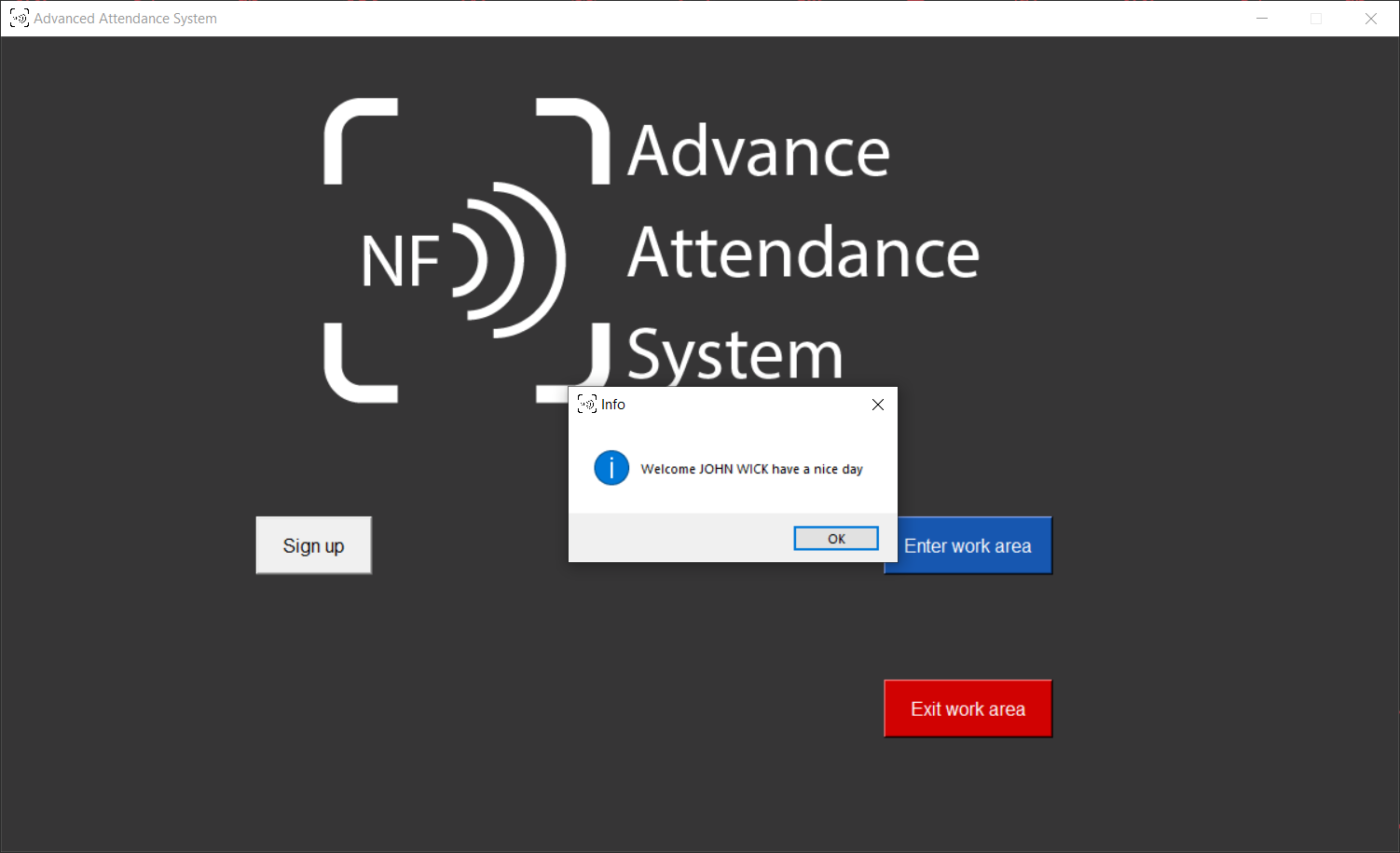


Σχήμα 4.2.6: Επιτυχής αποθήκευση χρήστη

#### 4.3 Είσοδος στον χώρο εργασίας

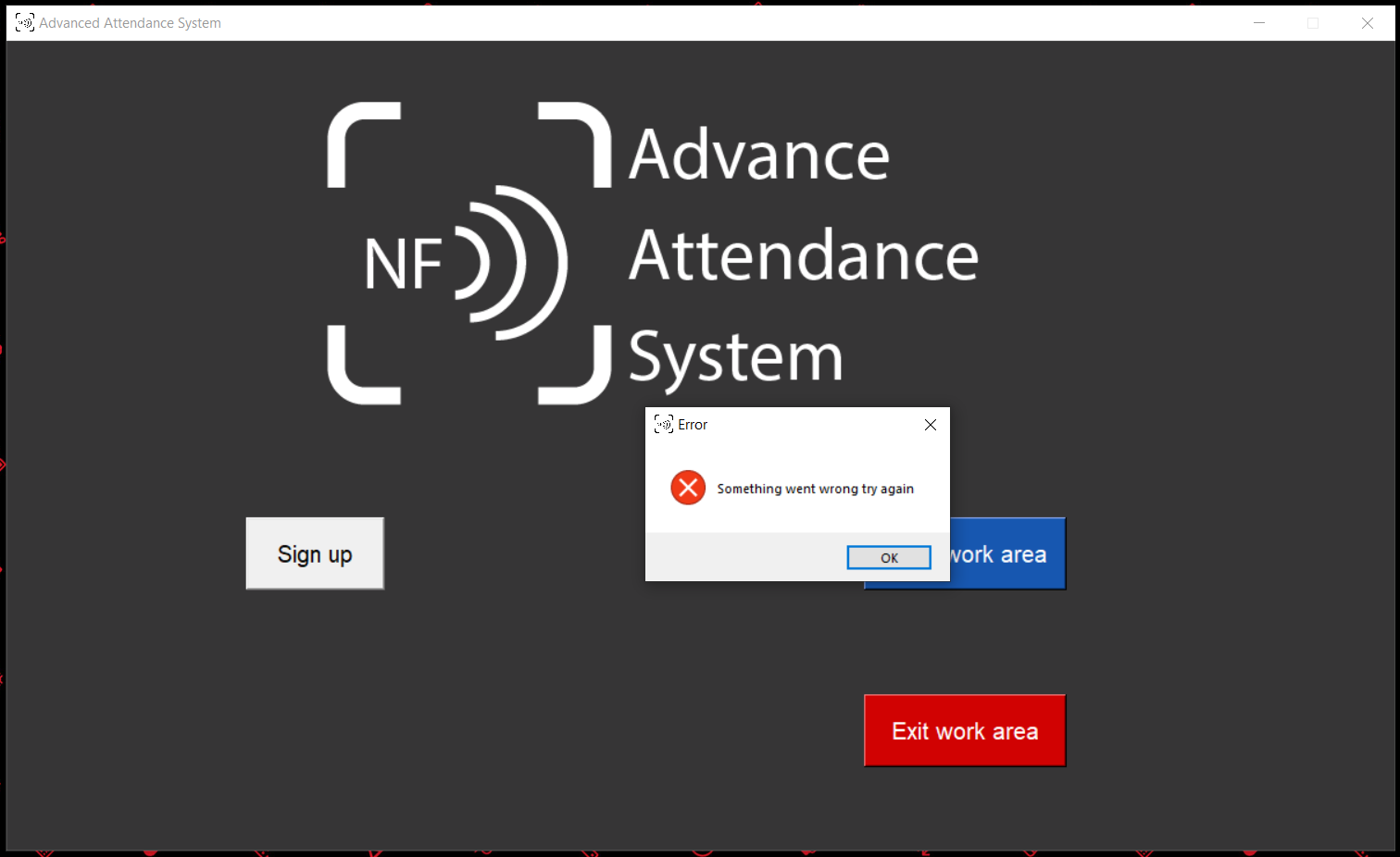
Για να καταγραφή η ώρα εισόδου ο εργαζόμενος πρέπει να ακολουθήσει τα εξής βήματα:

Βήμα 1: Αφού εναποθέσει την κάρτα RFID πάνω στον ειδικό αναγνώστη επιλέγει με το ποντίκι το κουμπί 2 (μπλε κουμπί “Enter work area”) του σχήματος 4.1.2, η εφαρμογή χρησιμοποιώντας την κάμερα του συστήματος μέσα σε 6 με 11 δευτερόλεπτα θα ταυτοποιήσει τον χρήστη με την εικόνα που έχει αποθηκευμένη για την συγκεκριμένη κάρτα. Το σύστημα στην συνέχεια θα τον ενημερώσει για πιθανό πρόβλημα ή αν έχει καταχωρηθεί η είσοδος του με κατάλληλο μήνυμα σχήμα 4.3.1-3.



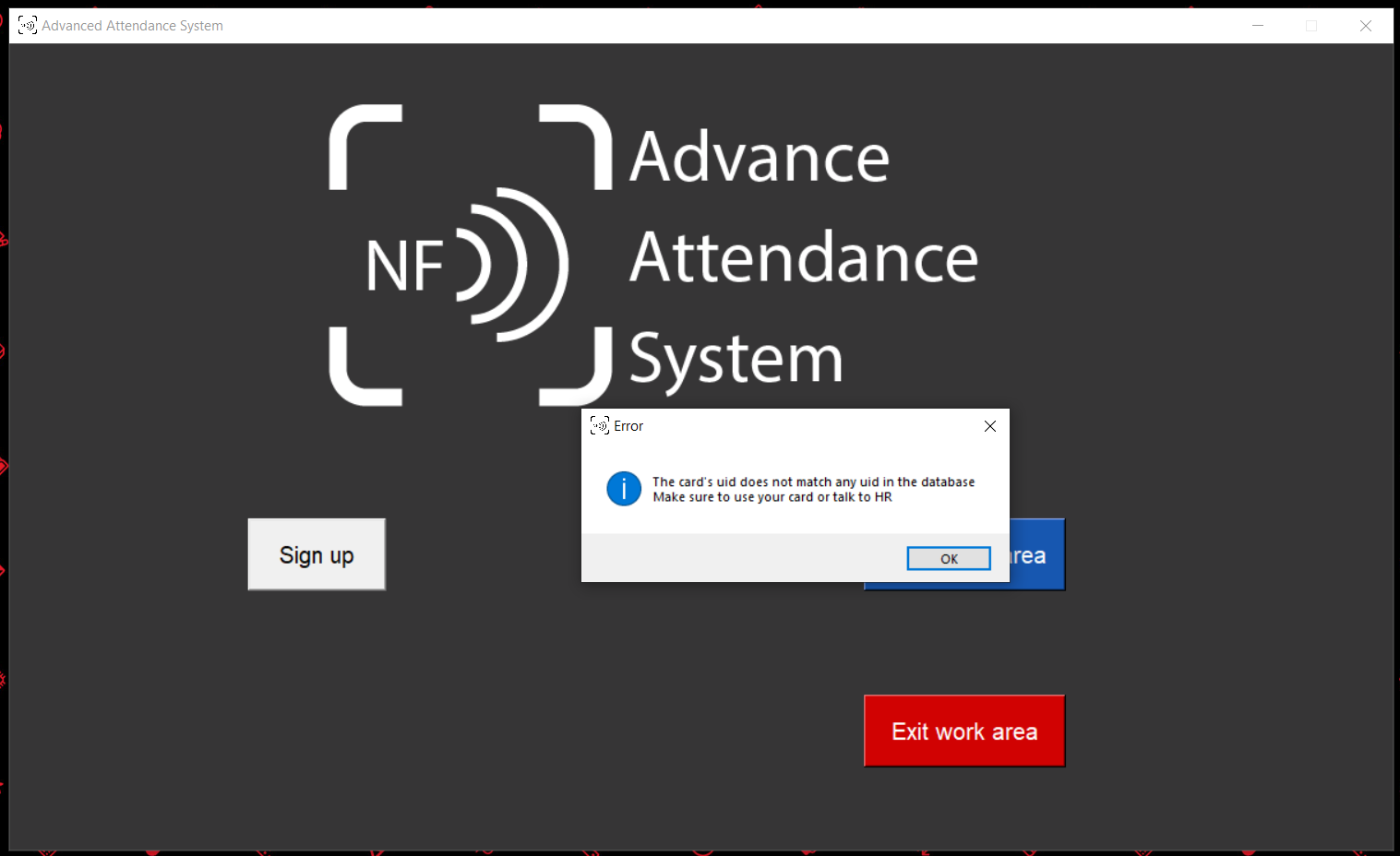
Σχήμα 4.3.1: Επιτυχής καταγραφή εισόδου

 Σημείωση 4.3.1: Σε περίπτωση που είτε η κάμερα ή ο αναγνώστης δεν είναι συνδεδεμένος με το σύστημα είτε για οποιονδήποτε άλλο λόγο δεν είναι δυνατή η ταυτοποίηση από την εφαρμογή θα εμφανιστεί ένα μήνυμα λάθους όπως στο σχήμα 4.3.2.



Σχήμα 4.3.2: Μη επιτυχής ταυτοποίηση

 Σημείωση 4.3.2: Σε περίπτωση που χρησιμοποιηθεί διαφορετική κάρτα από άλλο εργαζόμενο θα εμφανιστεί ένα μήνυμα λάθους όπως στο σχήμα 4.3.3

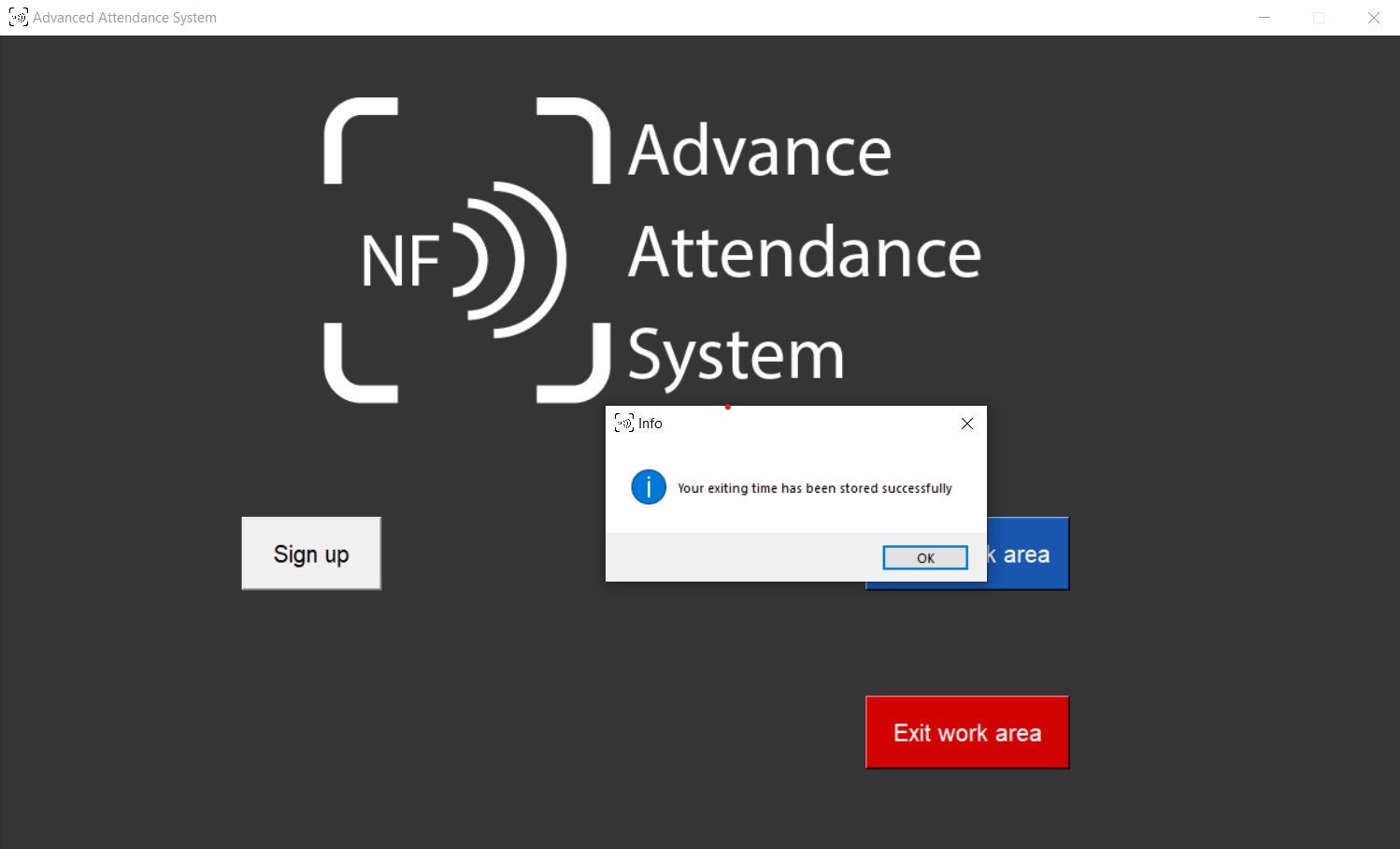


Σχήμα 4.3.3: Λάθος κάρτα RFID

#### 4.4 Έξοδος από τον χώρο εργασίας

Για να καταγραφή η ώρα εξόδου ο εργαζόμενος πρέπει να ακολουθήσει τα εξής βήματα:

Βήμα 1: Αφού εναποθέσει την κάρτα RFID πάνω στον ειδικό αναγνώστη επιλέγει με το ποντίκι το κουμπί 3 (κόκκινο κουμπί “Exit work area”) του σχήματος 4.1.2. Το σύστημα έπειτα θα τον ενημερώσει για πιθανό πρόβλημα ή αν έχει καταχωρηθεί η ώρα εξόδου του με κατάλληλο μήνυμα σχήμα 4.4.1.

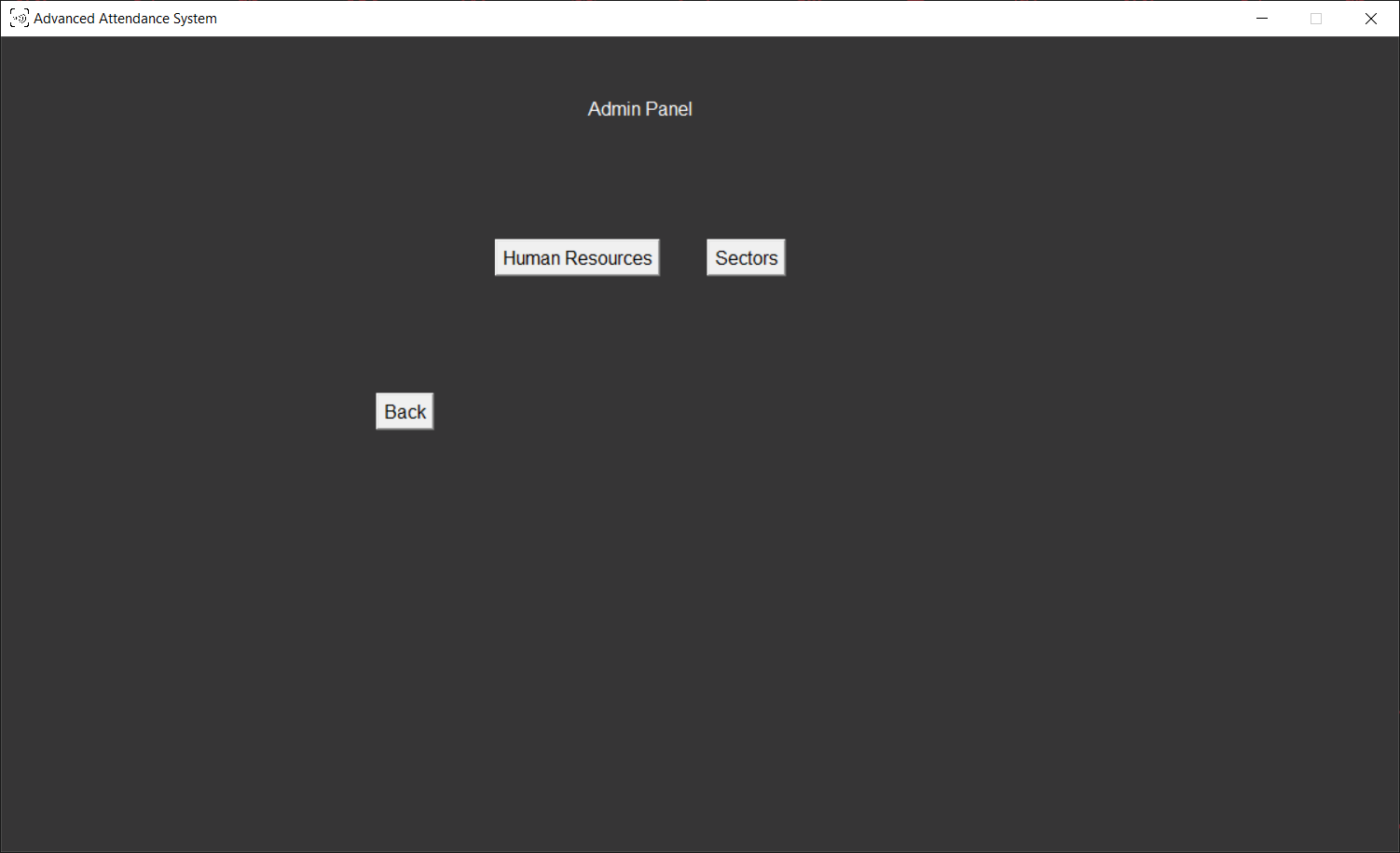


Σχήμα 4.4.1: Επιτυχής καταγραφή εξόδου

 Σημείωση 4.4.1: Σε περίπτωση που η εφαρμογή αδυνατεί να βρει κάποια εγγραφή εισόδου του υπαλλήλου για την συγκεκριμένη μέρα θα εμφανιστεί ένα μήνυμα λάθους όπως στο σχήμα 4.3.2

#### 4.5 Admin panel

Αν κατά την είσοδο (βλέπε Ενότητα 4.3) η εφαρμογή αναγνωρίσει πως ο χρήστης είναι εργαζόμενος στον τομέα του HR (Human Resources/Ανθρώπινου Δυναμικού) τότε μόλις καταγράψει την είσοδο του θα ανοίξει το “Admin Panel” σχήμα 4.5.1.



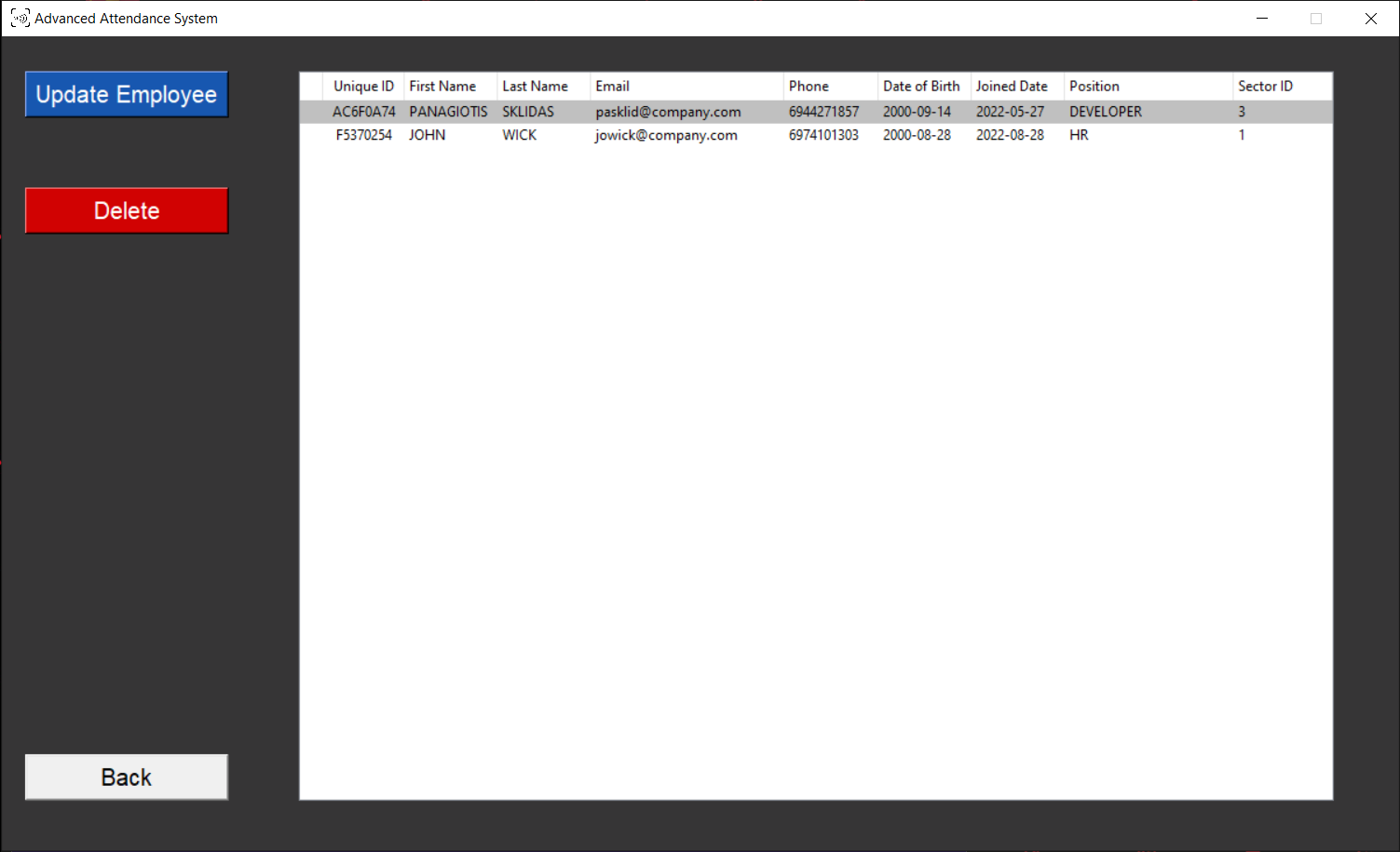
2

1

Σχήμα 4.5.1: Admin panel

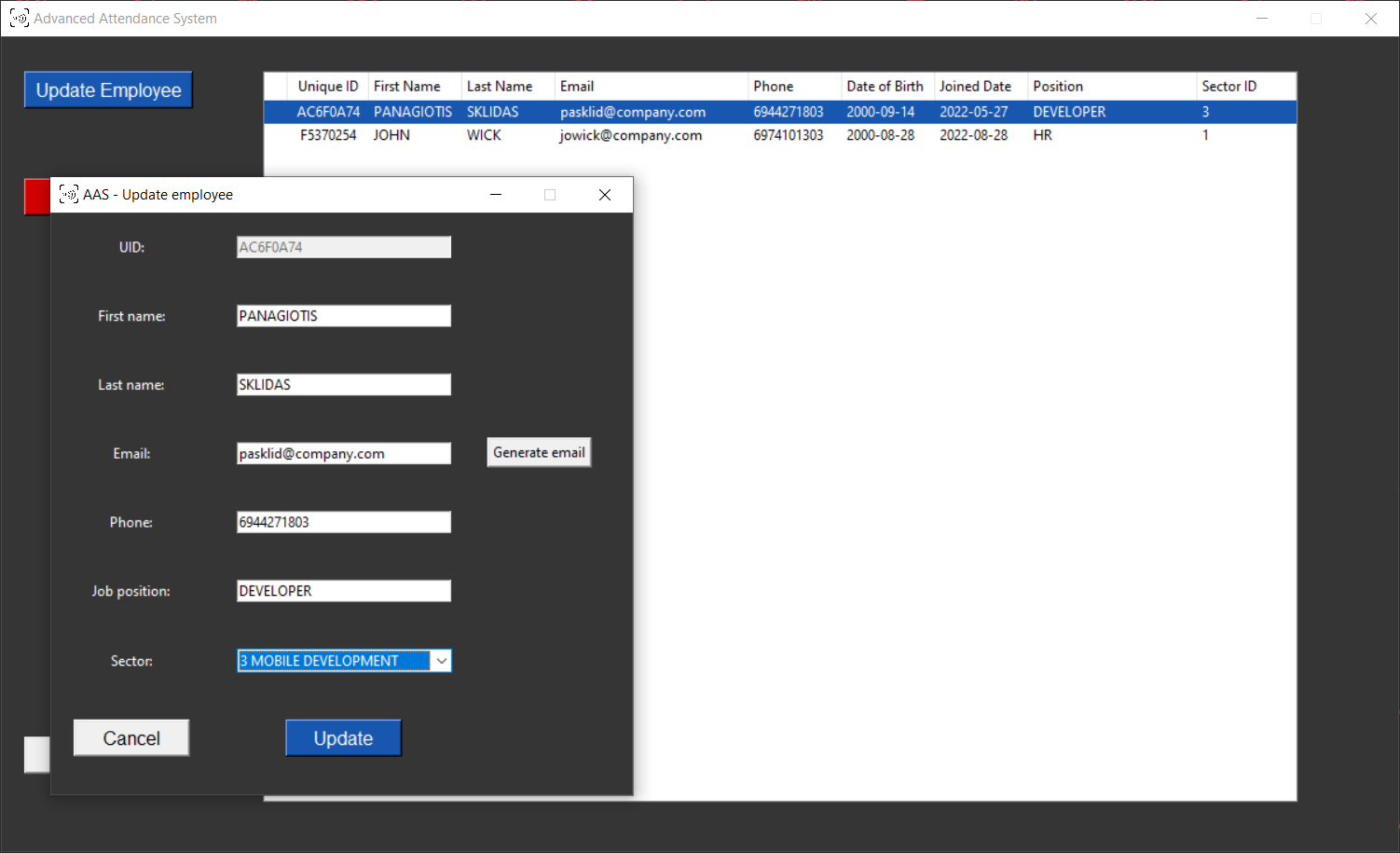
Αν ο χρήστης/διαχειριστής επιθυμεί να δει, να ενημερώσει ή να διαγράψει τα στοιχεία ενός χρήστη τότε αρκεί να πατήσει το κουμπί “Human Resources” (κουμπί 1) ώστε να μεταφερθεί στην αντίστοιχη οθόνη. Αν θέλει να δει στοιχεία που αφορούν του τομείς εργασίας τότε επιλέγει το κουμπί “Sectors” (κουμπί 2).

##### 4.5.1 Human Resources



Σχήμα 4.5.2: Οθόνη ανθρώπινου δυναμικού

Αν ο χρήστης/διαχειριστής θέλει να ενημερώσει κάποια στοιχεία ενός χρήστη αρκεί απλά να επιλέξει τον χρήστη που επιθυμεί να επεξεργαστεί και να πατήσει το μπλε κουμπί “Update Employee” και θα εμφανιστεί ένα αναδυόμενο παράθυρο στο οποίο θα μπορεί να επεξεργαστεί τις πληροφορίες του χρήστη σχήμα 4.5.3



Σχήμα 4.5.3: Ενημέρωση στοιχείων χρήστη

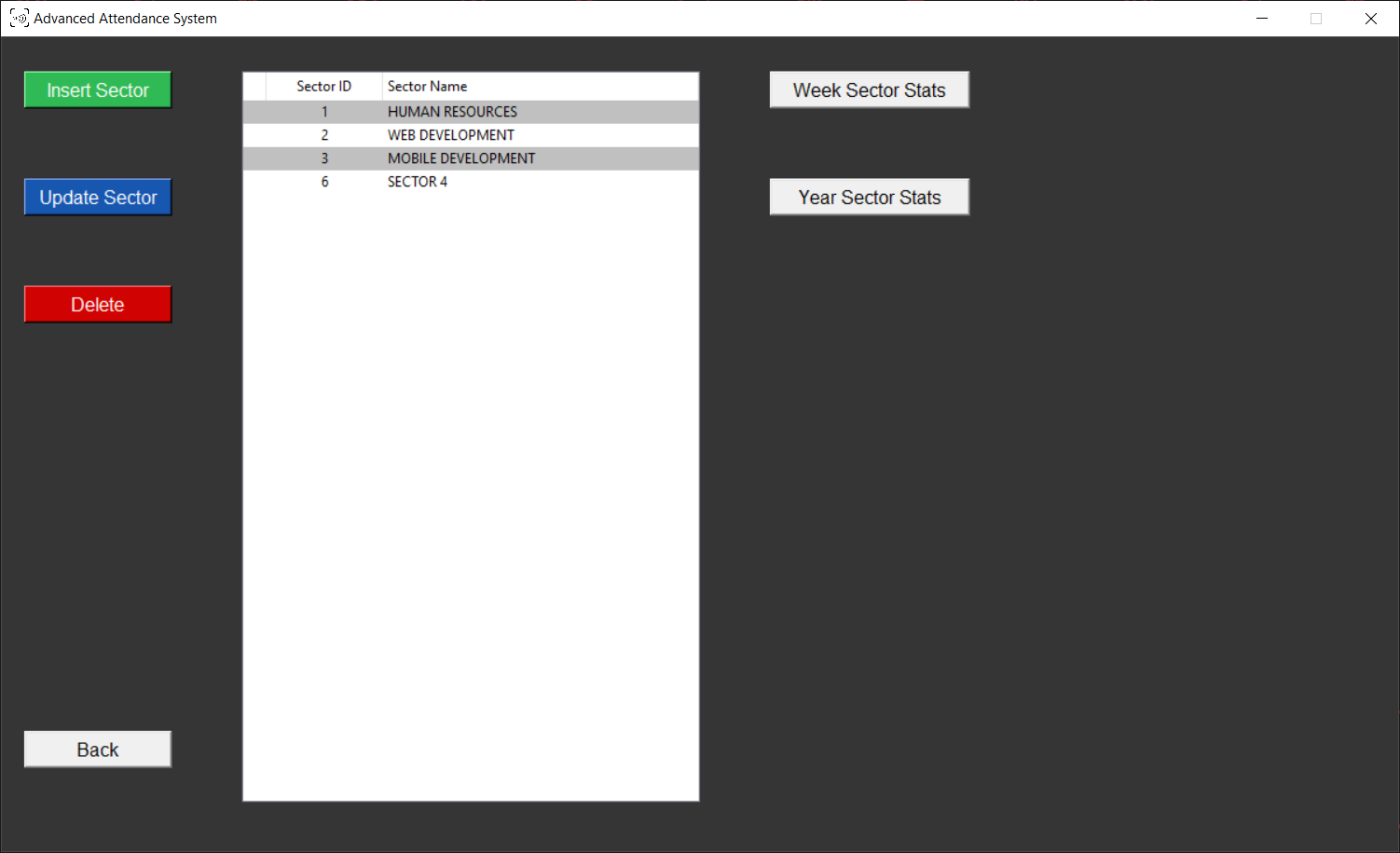
Αφού έχει ολοκληρώσει τις αλλαγές ο διαχειριστής επιλέγει το “Update”.

 Σημείωση 4.5.1: Ο πίνακας ενημερώνεται αυτόματα

Αν ο χρήστης/διαχειριστής θέλει να διαγράψει ένα χρήστη αρκεί απλά να επιλέξει τον χρήστη που επιθυμεί και να πατήσει το κουμπί “Delete”

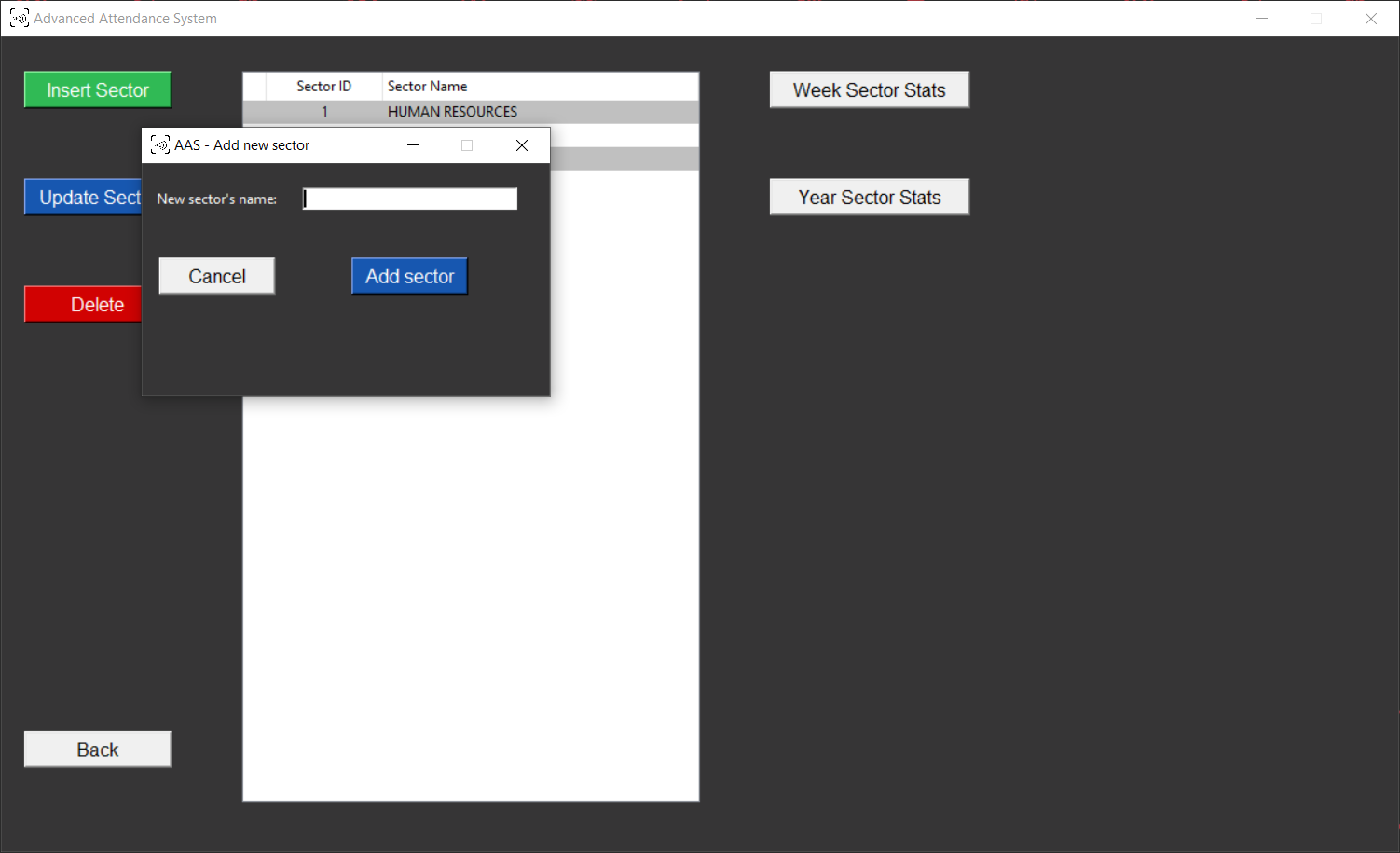
 Σημείωση 4.5.2: Ο πίνακας ενημερώνεται αυτόματα

##### 4.5.2 Sectors



Σχήμα 4.5.4: Οθόνη τομέων εργασίας

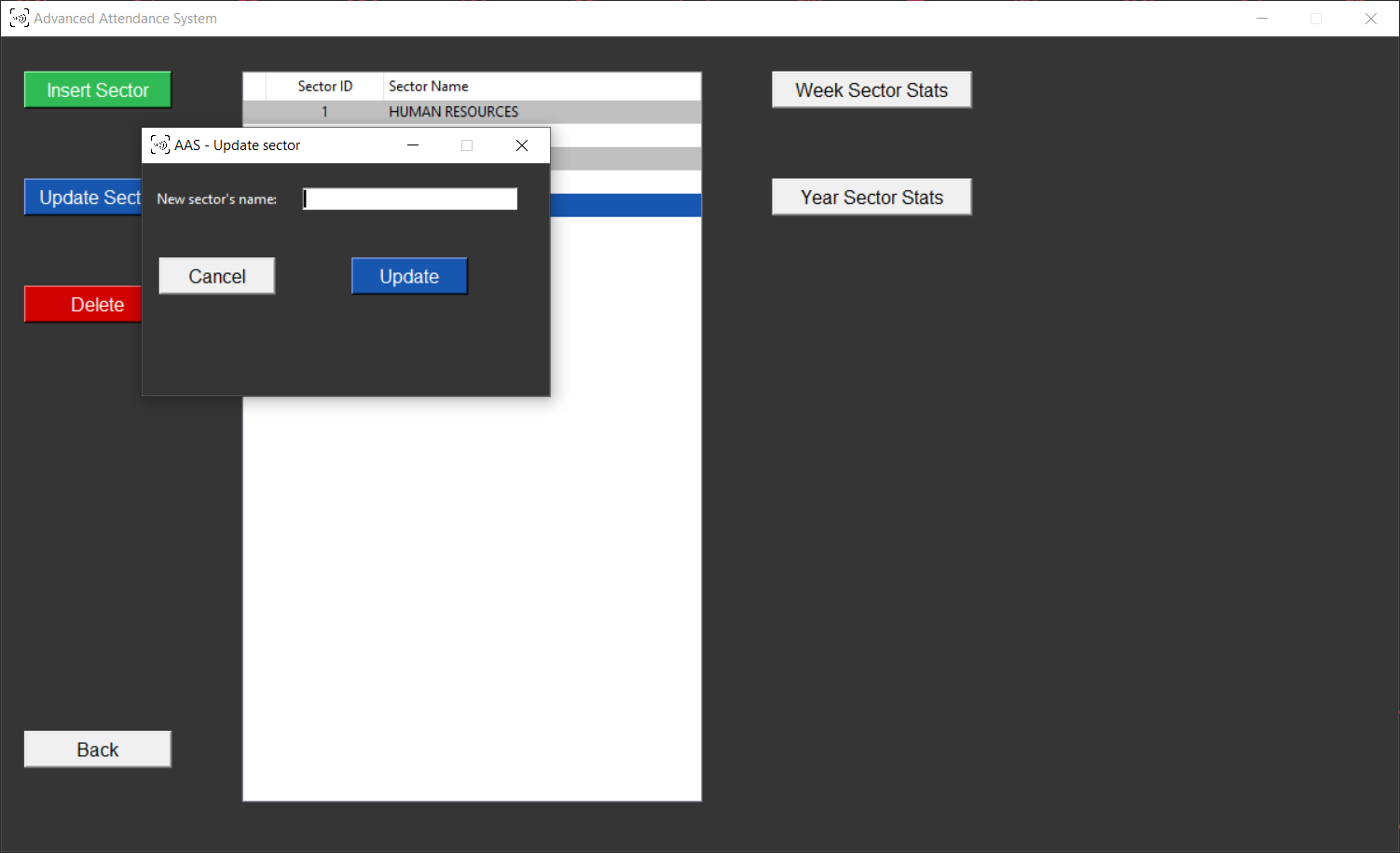
Αν ο χρήστης/διαχειριστής θέλει να δημιουργήσει ένα νέο τομέα αρκεί να πατήσει το πράσινο κουμπί “Insert Sector” και θα εμφανιστεί ένα αναδυόμενο παράθυρο στο οποίο θα μπορεί να γράψει το όνομα του νέου τομέα σχήμα 4.5.5.



Σχήμα 4.5.5: Δημιουργία νέου τομέα

 Σημείωση 4.5.2: Ο πίνακας ενημερώνεται αυτόματα

Αν ο χρήστης/διαχειριστής θέλει να ενημερώσει το όνομα ενός τομέα αρκεί απλά να επιλέξει τον τομέα που επιθυμεί να τροποποιήσει και να πατήσει το μπλε κουμπί “Update Sector” και θα εμφανιστεί ένα αναδυόμενο παράθυρο στο οποίο θα μπορεί να γράψει το νέο όνομα του τομέα σχήμα 4.5.6



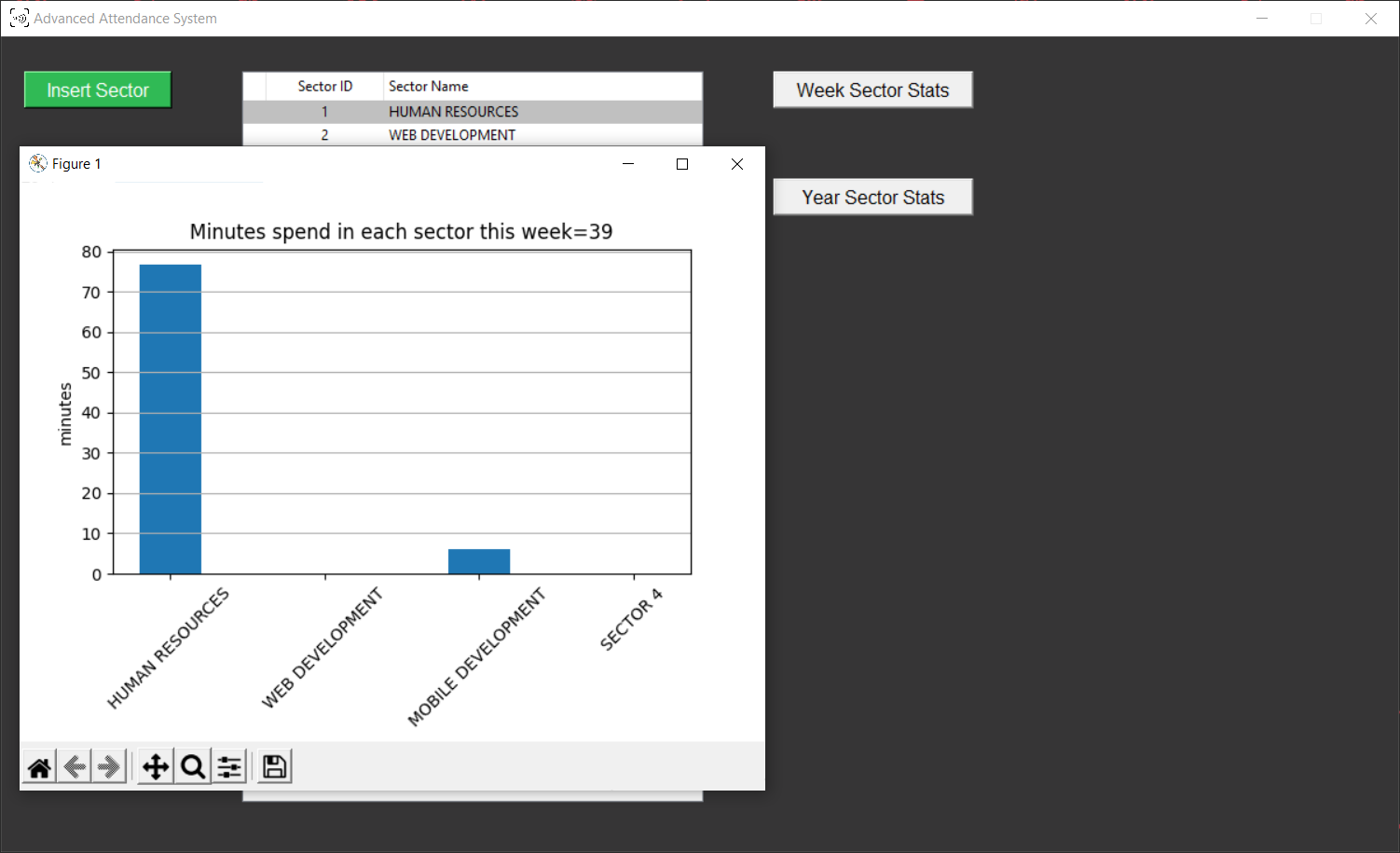
Σχήμα 4.5.6: Ενημέρωση τομέα

 Σημείωση 4.5.2: Ο πίνακας ενημερώνεται αυτόματα

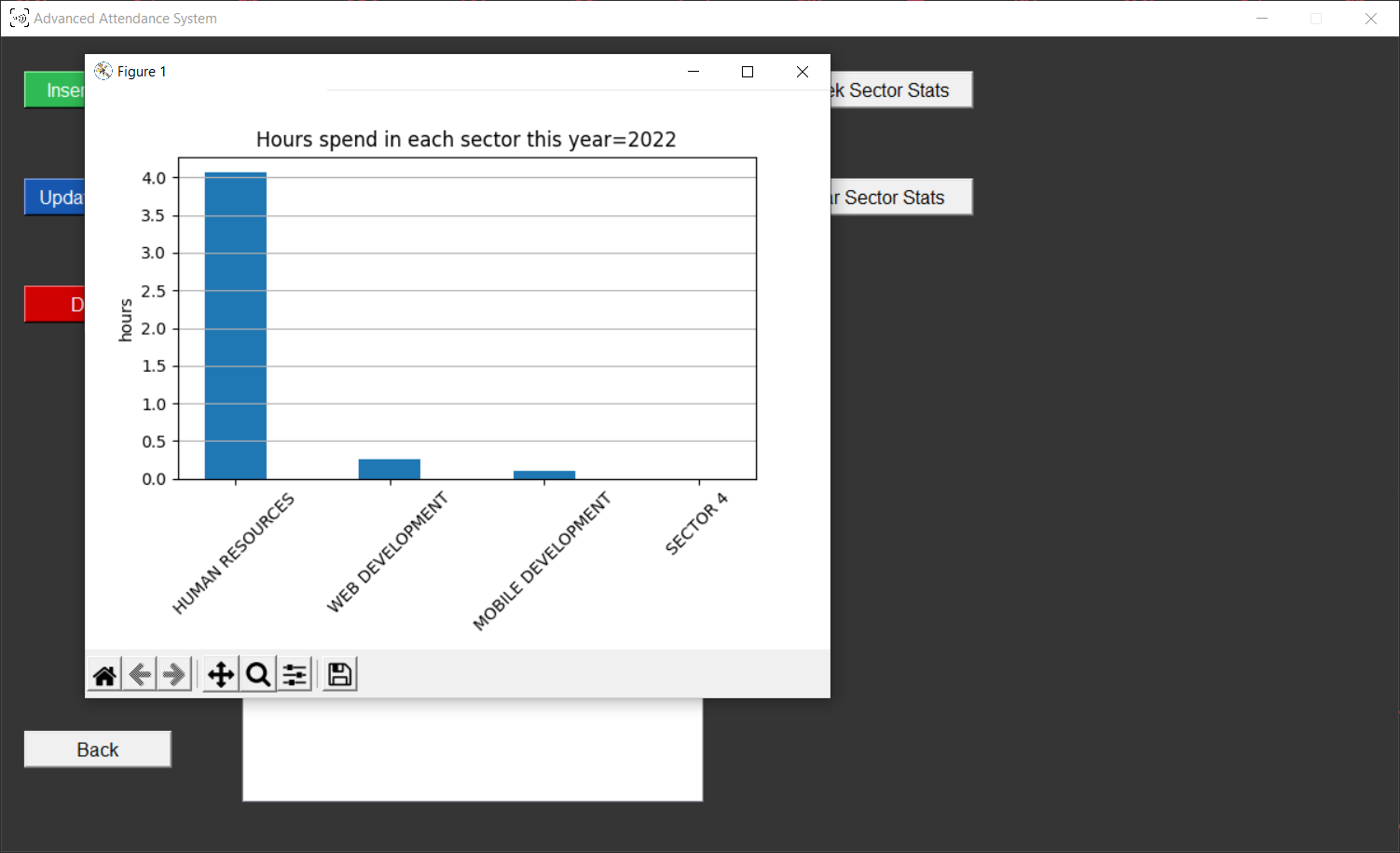
Αν ο χρήστης/διαχειριστής θέλει να διαγράψει ένα τομέα αρκεί απλά να επιλέξει τον τομέα που επιθυμεί και να πατήσει το κουμπί “Delete”

 Σημείωση 4.5.2: Ο πίνακας ενημερώνεται αυτόματα

Τέλος, αν ο χρήστης/διαχειριστής θέλει να δει τα στατιστικά στοιχεία τις εβδομάδας και του χρόνου των τομέων αρκεί να επιλέξει το “Week Sector Stats” (Σχήμα 4.5.7) ή “Year Sector Stats” (Σχήμα 4.5.8) αντίστοιχα.



Σχήμα 4.5.7: Week stats



Σχήμα 4.5.8: Year stats

#### 4.6 Συχνές ερωτήσεις

1. Σε περίπτωση που ξεχάσω/χάσω την κάρτα πρόσβασης ως διαχειριστής τι μπορώ να κάνω;

Σε περίπτωση που χάσατε την κάρτα RFID θα πρέπει να ζητήσετε από κάποιον άλλο διαχειριστή (υπάλληλο στο τμήμα HR) να σας δώσει μια καινούργια κάρτα και να σας διαγράψει από το σύστημα ώστε να έχετε την δυνατότητα να κάνετε εκ νέου εγγραφή στο σύστημα. Αν την ξεχάσατε δεν μπορείτε να κάνετε κάτι και θα πρέπει να έρθετε πάλι σε επικοινωνία με ένα διαχειριστή ώστε να κάνει την διαδικασία ταυτοποίησης και εισόδου όπως θεωρεί αυτός πιο σωστά.

1. Σε περίπτωση που ξεχάσω/χάσω την κάρτα πρόσβασης ως απλός χρήστης τι μπορώ να κάνω;

Σε περίπτωση που χάσατε την κάρτα RFID θα πρέπει να ζητήσετε από κάποιον διαχειριστή (υπάλληλο στο τμήμα HR) να σας δώσει μια καινούργια κάρτα και να σας διαγράψει από το σύστημα ώστε να έχετε την δυνατότητα να κάνετε εκ νέου εγγραφή σε αυτό. Αν την ξεχάσατε δεν μπορείτε να κάνετε κάτι και θα πρέπει να έρθετε πάλι σε επικοινωνία με ένα διαχειριστή ώστε να κάνει την διαδικασία ταυτοποίησης και εισόδου όπως θεωρεί αυτός πιο σωστά.

1. Ποιοι μπορούν να δουν τα δεδομένα μου;

Τα δεδομένα σας, όπως για παράδειγμα το ονοματεπώνυμο, το τηλέφωνο, το εταιρικό email, τον κωδικό της κάρτας RFID, την θέση εργασίας, τον τομέα στον οποίον εργάζεστε, την ημερομηνία γέννησης καθώς και την ημερομηνία πρόσληψης σας μπορούν να τα δουν μόνο οι διαχειριστής (υπεύθυνος ανθρώπινου δυναμικού) και κανένας άλλος.

1. Μπορώ να αλλάξω τα στοιχεία μου;

Τα δεδομένα σας, όπως για παράδειγμα το ονοματεπώνυμο, το τηλέφωνο, το εταιρικό email, την θέση εργασίας και τον τομέα στον οποίον εργάζεστε μπορούν να αλλαχτούν αποκλειστικά από κάποιον διαχειριστή του συστήματος, συνεπώς θα πρέπει να έρθετε σε επικοινωνία με τον υπεύθυνο προσωπικού και να του αναφέρετε τι αλλαγές θέλετε να γίνουν.

## Παράρτημα 3 Άδειες χρήσης βιβλιοθηκών

### 3.1 Pyscard[46]

MIT License

MIT License Copyright (c) 2021, Adam Geitgey Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### 3.2 Face\_Recognition[47]

GNU Lesser General Public License v2.1

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

GNU LESSER GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you". A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library. You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful. (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it. Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library. In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License. However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License.

Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions).

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already presents on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable. It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library. If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances. It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice. This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

# Βιβλιογραφία

1. Cardullo, M., & Parks, W. (1973). *Transponder apparatus and system* (U.S. Patent No. 3,713,148). U.S Patent and Trademark Office. <https://patentimages.storage.googleapis.com/4a/63/c4/1e14dedfdb7bd2/US3713148.pdf>
2. A. (2020, April 9). *Genesis of the Versatile RFID Tag*. RFID JOURNAL. <https://www.rfidjournal.com/genesis-of-the-versatile-rfid-tag>
3. Lagkas, T. (2021, September). *Wireless Sensor Network* [Slides]. Dropbox. <https://www.dropbox.com/sh/lqcltjb5smwyvbq/AADr9rdDo9LZzJI1iwtvTvaya?dl=0&preview=JAUNTY_O2_3_Wireless_Sensor_Networks.pptx>
4. Playful Technology. (2018, December 5). *RFID Roundup!* [Video]. YouTube. <https://www.youtube.com/watch?v=98GXrixOM4c>
5. Amsler, S., & Shea, S. (2021, March 31). *RFID (radio frequency identification)*. IoT Agenda. <https://internetofthingsagenda.techtarget.com/definition/RFID-radio-frequency-identification>
6. JT308 RFID ID Card Reader. (n.d.). Hellas Digital. Retrieved April 15, 2022, from <https://www.hellasdigital.gr/electronics/sensors/rfid-sensors/jt308-rfid-id-card-reader/>
7. RFID Features: *Key Features of RFID* - Corerfid. (2020, December 20). RFID Systems for Manufacturing, Assets, Lifting & Logistics. <https://www.corerfid.com/rfid-technology/what-is-rfid/rfid-features/>
8. Taylor, M. (2021, December 14). *Computer Vision with Convolutional Neural Networks - The Startup*. Medium. <https://medium.com/swlh/computer-vision-with-convolutional-neural-networks-22f06360cac9>
9. W. Hao, W. Yizhou, L. Yaqin and S. Zhili, *"The Role of Activation Function in CNN,"* 2020 2nd International Conference on Information Technology and Computer Application (ITCA), 2020, pp. 429-432, doi: 10.1109/ITCA52113.2020.00096.
10. Rizzoli, A. (2022, February 7). *27+ Most Popular Computer Vision Applications in 2022*. V7. <https://www.v7labs.com/blog/computer-vision-applications>
11. Cohen, J. (2019, November 25). *Computer Vision applications in Self-Driving Cars.* Becoming Human: Artificial Intelligence Magazine. <https://becominghuman.ai/computer-vision-applications-in-self-driving-cars-610561e14118>
12. IBM. (n.d.). *What is Computer Vision?* | IBM. Retrieved February 27, 2022, from <https://www.ibm.com/topics/computer-vision>
13. Face Recognition. (2021, February 15). *Electronic Frontier Foundation*. <https://www.eff.org/pages/face-recognition>
14. RECFACES. (2021, March 25). *Understanding Facial Recognition Algorithms*. <https://recfaces.com/articles/facial-recognition-algorithms>
15. P. Viola, & M. Jones. (2001). *Rapid Object Detection using a Boosted Cascade of Simple Features*. Conference on Computer Vision and Pattern Recognition. <https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf>
16. OpenCV. (n.d.). *OpenCV: Cascade Classifier*. <https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html>
17. M. Turk and A. Pentland. (1991). *Face Recognition Using Eigenfaces.* Conference on Computer Vision and Pattern Recognition. <https://sites.cs.ucsb.edu/~mturk/Papers/mturk-CVPR91.pdf>
18. L. Sirovich and M. Kirby. (1986, November 10). *Low-dimensional procedure for the characterization of human faces*. <https://pyimagesearch.com/wp-content/uploads/2021/05/kirby_1987.pdf>
19. Wang, P. (n.d.). *This Person Does Not Exist*. This Person Does Not Exist. <https://thispersondoesnotexist.com/>
20. L. Smith. (2002, February 26). *A tutorial on Principal Components Analysis*. <http://www.iro.umontreal.ca/~pift6080/H09/documents/papers/pca_tutorial.pdf>
21. Rosebrock, A. (2021, May 12). *OpenCV Eigenfaces for Face Recognition*. PyImageSearch. <https://pyimagesearch.com/2021/05/10/opencv-eigenfaces-for-face-recognition/>
22. R. A. Fisher. (1936). *The use of multiple measurements in taxonomic problems*. <https://www.comp.tmu.ac.jp/morbier/R/Fisher-1936-Ann._Eugen.pdf>
23. T. Ahonen, A. Hadid & M. Pietikäinen. (2004). *Face Recognition with Local Binary Patterns*. 3021. 469-481. 10.1007/978-3-540-24670-1\_36. <https://www.researchgate.net/publication/221304831_Face_Recognition_with_Local_Binary_Patterns>
24. Rosebrock, A. (2021a, May 8). *Face Recognition with Local Binary Patterns (LBPs) and OpenCV*. PyImageSearch. <https://pyimagesearch.com/2021/05/03/face-recognition-with-local-binary-patterns-lbps-and-opencv/>
25. Dickson, B. (2020, January 6). *What are convolutional neural networks (CNN)?* TechTalks. <https://bdtechtalks.com/2020/01/06/convolutional-neural-networks-cnn-convnets>
26. Peng Lu, Baoye Song & Lin Xu. (2021). *Human face recognition based on convolutional neural network and augmented dataset*. Systems Science & Control Engineering, 9:sup2, 29-37, DOI: [10.1080/21642583.2020.1836526](https://doi.org/10.1080/21642583.2020.1836526)
27. (n.d.). *User Interface Design Basics* | Usability.gov. Usability. <https://www.usability.gov/what-and-why/user-interface-design.html>
28. (n.d.). *User Experience Basics* | Usability.gov. Usability. <https://www.usability.gov/what-and-why/user-experience.html>
29. OpenCV. (n.d.). OpenCV: *Face Recognition with OpenCV*. <https://docs.opencv.org/3.4/da/d60/tutorial_face_main.html>
30. (2020, April 21). *The Difference Between NFC and RFID*. Global Payments Integrated. <https://www.globalpaymentsintegrated.com/en-us/blog/2020/04/21/what-is-the-difference-between-nfc-and-rfid>
31. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
32. Rousseau, L. (n.d.-a). *GitHub - LudovicRousseau/pyscard: pyscard smartcard library for python*. GitHub. <https://github.com/LudovicRousseau/pyscard>
33. Rousseau, L. (n.d.). *Pyscard user’s guide — Pyscard 2.0.4 documentation*. GitHub. <https://pyscard.sourceforge.io/user-guide.html#pyscard-user-guide>
34. Wikipedia contributors. (2022, July 7). *Smart card application protocol data unit*. Wikipedia. <https://en.wikipedia.org/wiki/Smart_card_application_protocol_data_unit>
35. (2020). *Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange*. ISO. <https://www.iso.org/obp/ui/#iso:std:iso-iec:7816:-4:ed-4:v1:en:sec:intro>
36. Geitgey, A. (n.d.). GitHub - ageitgey/face\_recognition: The world’s simplest facial recognition api for Python and the command line. GitHub. Retrieved September 6, 2022, from <https://github.com/ageitgey/face_recognition#readme>
37. MIT. (n.d.). LFW Face Database : Main. Retrieved September 8, 2022, from <http://vis-www.cs.umass.edu/lfw/>
38. LFW : Results. (n.d.-b). Retrieved September 8, 2022, from <http://vis-www.cs.umass.edu/lfw/results.html>
39. LFW : Results. (n.d.). Retrieved September 8, 2022, from <http://vis-www.cs.umass.edu/lfw/results.html#dlib>
40. D. (2021, October 1). Certificate Propagation Service (Windows) - Windows security. Microsoft Docs. <https://docs.microsoft.com/en-us/windows/security/identity-protection/smart-cards/smart-card-certificate-propagation-service>
41. GoToTags. (2021, September 29). Windows Certificate Propagation Service. GoToTags Learning Center. <https://learn.gototags.com/nfc/software/windows/certificate-propagation-service>
42. Great Learning Team. (2022, March 22). *What is Resnet or Residual Network | How Resnet Helps?* GreatLearning Blog: Free Resources What Matters to Shape Your Career! Retrieved September 17, 2022, from <https://www.mygreatlearning.com/blog/resnet/#sh1>
43. Paez, D. (2019, February 21). *“This Person Does Not Exist” Creator Reveals His Site’s Creepy Origin Story*. Inverse. Retrieved September 21, 2022, from <https://www.inverse.com/article/53414-this-person-does-not-exist-creator-interview>
44. <https://www.youtube.com/watch?v=cQ54GDm1eL0>
45. Tyagi, M. (2022, January 6). *HOG (Histogram of Oriented Gradients): An Overview - Towards Data Science*. Medium. Retrieved September 24, 2022, from <https://towardsdatascience.com/hog-histogram-of-oriented-gradients-67ecd887675f>
46. Rousseau, L. (n.d.). *GitHub - LudovicRousseau/pyscard: pyscard smartcard library for python*. GitHub. Retrieved October 12, 2022, from <https://github.com/LudovicRousseau/pyscard>
47. Geitgey, A. (n.d.-b). *GitHub - ageitgey/face\_recognition: The world’s simplest facial recognition api for Python and the command line*. GitHub. Retrieved October 12, 2022, from <https://github.com/ageitgey/face_recognition>