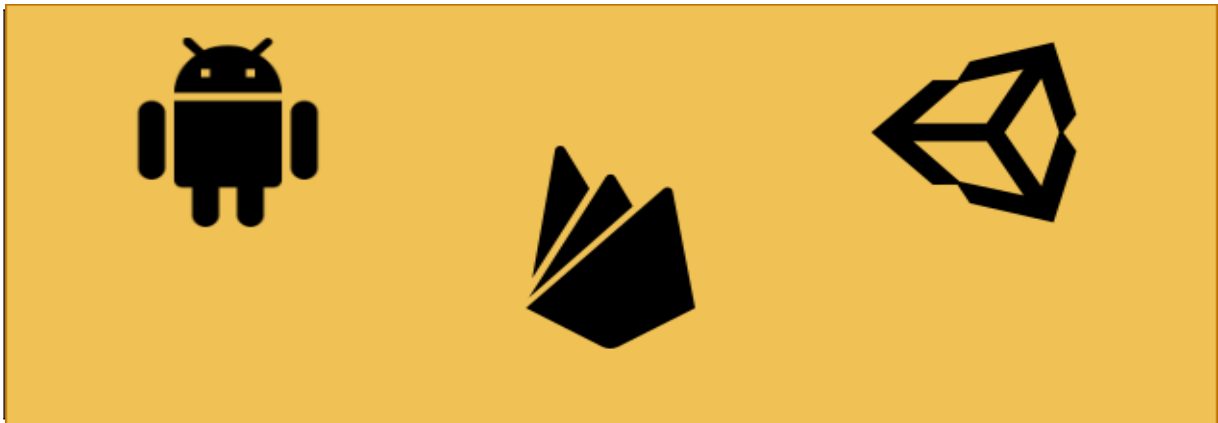




ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Άνοια: Εφαρμογή πρόληψης και υποβοήθησης σε
συσκευές Android



Της φοιτήτριας
Μπόλκα Μαρίας
Αρ. Μητρώου: 164798

Επιβλέπων
Αδαμίδης
Παναγιώτης
Καθηγητής

Ιανουάριος 2023

Τίτλος Δ.Ε. Άνοια: Εφαρμογή πρόληψης και υποβοήθησης σε συσκευές Android

Κωδικός Δ.Ε. 21358

Όνοματεπώνυμο φοιτητή Μπόλκα Μαρία

Όνοματεπώνυμο εισηγητή Αδαμίδης Παναγιώτης

Ημερομηνία ανάληψης Δ.Ε. 14-10-2021

Ημερομηνία περάτωσης Δ.Ε. 21-01-2023

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως πτυχιακή εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία της φοιτήτριας Μπόλκα Μαρίας που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητα και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Πρόλογος

Η παρούσα πτυχιακή εργασία έχει ως στόχο την ανάλυση της άνοιας και την προσπάθεια για πρόληψη της νόσου. Για τον σκοπό αυτό δημιουργήθηκε λογισμικό που ως κεντρική λειτουργία έχει, μέσω διασκεδαστικών παιχνιδιών, να εξασκεί τις πνευματικές ικανότητες των χρηστών. Υποστηρίζονται και άλλες λειτουργίες όπως η προτροπή για σωματική άσκηση (περπάτημα) και η καταχώρηση φαρμακευτικής αγωγής.

Επέλεξα το συγκεκριμένο θέμα γιατί ήθελα να αναπτύξω μία εφαρμογή που θα είναι ικανή να βοηθά στο πιο σημαντικό κομμάτι στην ζωή ενός ανθρώπου, που είναι η υγεία. Θεωρώ χρέος των επιστημόνων της τεχνολογίας να συμβάλλουν σε τέτοιου τύπου εφαρμογές. Ένας ακόμη λόγος που επέλεξα το συγκεκριμένο θέμα είναι η επιθυμία μου να αναπτύξω τις γνώσεις μου για εφαρμογές κινητών τηλεφώνων, καθώς είναι το είδος των συσκευών που χρησιμοποιούνται περισσότερο τα τελευταία χρόνια.

Περίληψη

Στα πλαίσια της παρούσας πτυχιακής εργασίας έγινε η υλοποίηση ενός λογισμικού που έχει ως στόχο να δρα προληπτικά για την ανάπτυξη της άνοιας ή και να υποβοηθά ασθενείς στα πρώτα στάδια της νόσου. Στο κείμενο αναλύεται η νόσος, οι τεχνολογίες που χρησιμοποιήθηκαν, η υλοποίηση της εφαρμογής και οι βελτιώσεις που θα έδιναν καλύτερα αποτελέσματα στους χρήστες.

Το λογισμικό που αναπτύχθηκε αφορά κινητά τηλέφωνα με λειτουργικό σύστημα Android. Για την ανάπτυξη του λογισμικού χρησιμοποιήθηκε το Android Studio, όπου έγινε χρήση της γλώσσας Java και η πλατφόρμα Unity, όπου έγινε ανάπτυξη κώδικα σε C#. Για την διαχείριση του προφίλ του χρήστη έγινε χρήση δύο υπηρεσιών της Firebase. Πιο συγκεκριμένα για την αποθήκευση των δεδομένων χρησιμοποιήθηκε η Firestore και για την αυθεντικοποίηση του χρήστη η υπηρεσία Authentication.

Dementia: Prevention and assistance application on Android devices

Maria Bolka

Abstract

For this thesis, a software was implemented that aims to act preventively for the development of dementia or to help patients in the early stages of the disease. The text analyses the disease, the technologies used, the development of the application and the improvements that would give better results to the users.

The software developed is for mobile phones with Android operating system. Android Studio was used for software development, where the Java language was used and Unity platform, where code development was done in C#. To manage the user's profile two Firebase services were used. More specifically for data storage Firestore was used and the Authentication service was used to authenticate the user.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κύριο Αδαμίδη Παναγιώτη για την καθοδήγηση του με σκοπό το καλύτερο δυνατό αποτέλεσμα. Επίσης θα ήθελα να ευχαριστήσω οικογένεια και φίλους που με βοήθησαν δίνοντας μου δύναμη ώστε να μπορέσω να ανταποκριθώ στις απαιτήσεις της πτυχιακής εργασίας αλλά και γενικότερα στην φοίτηση μου στην σχολή.

Περιεχόμενα

Πρόλογος.....	ii
Περίληψη.....	iii
Abstract.....	iv
Ευχαριστίες.....	v
Περιεχόμενα.....	vi
Κατάλογος Σχημάτων.....	ix
Συντομογραφίες.....	xi
Κεφάλαιο 1ο: Εισαγωγή.....	1
1.1 Εισαγωγή.....	1
1.2 Τεχνολογίες ανάπτυξης λογισμικού.....	1
1.3 Παρουσίαση κεφαλαίων.....	2
Κεφάλαιο 2ο: Άνοια.....	4
2.1 Εισαγωγή.....	4
2.2 Η νόσος της άνοιας.....	4
2.3 Τύποι άνοιας.....	4
2.3.1 Η νόσος Alzheimer.....	4
2.3.2 Αγγειακή άνοια.....	5
2.3.3 Άνοια σωματιδίων Lewy.....	5
2.3.4 Άνοια της νόσου Parkinson.....	5
2.3.5 Μετοποκροταφική άνοια.....	6
2.4 Κατηγοριοποίηση της άνοιας βάσει προτύπων.....	6
2.5 Εξέλιξη των συμπτωμάτων γενικά.....	7
2.6 Πνευματική εξάσκηση και πρόληψη άνοιας.....	7
2.7 Ο ρόλος της τεχνολογίας.....	8
2.7.1 Πρόληψη της νόσου με την βοήθεια παιχνιδιών σε ψηφιακή μορφή.....	8
2.7.2 Τεχνολογική αντιμετώπιση της ασθένειας.....	8
2.8 Επίλογος.....	10
Κεφάλαιο 3ο: Τεχνολογίες που χρησιμοποιήθηκαν.....	11
3.1 Εισαγωγή.....	11
3.2 Android.....	11

3.2.1 Java API framework.....	11
3.2.2 View System.....	11
3.2.3 Resource Manager.....	12
3.2.4 Activity Manager.....	12
3.3 Αρχείο AndroidManifest.....	12
3.4 Αρχείο εξαρτήσεων.....	12
3.5 Firebase.....	12
3.5.1 Authentication.....	13
3.5.2 Real-time database.....	13
3.5.3 Firestore.....	13
3.5.4 Cloud Storage.....	13
3.5.5 Firebase Security Rules.....	13
3.6 Unity.....	13
3.6.1 Asset workflow.....	14
3.6.2 Scenes.....	14
3.6.3 Game Objects.....	14
3.6.4 Prefabs.....	14
3.7 Δημιουργία εικονιδίων και σχεδιαγραμμάτων.....	14
3.8 Επίλογος.....	15
Κεφάλαιο 4ο: Σχεδίαση και υλοποίηση της εφαρμογής.....	16
4.1 Εισαγωγή.....	16
4.2 Εμπειρία χρήστη (User Experience).....	16
4.3 Αποθήκευση των δεδομένων του χρήστη.....	17
4.4 Σχεδιασμός και υλοποίηση Android εφαρμογής.....	17
4.4.1 Εγκατάσταση Firebase.....	18
4.4.2 Σύνδεση και εγγραφή χρήστη.....	18
4.4.3 Κύριο μενού.....	21
4.4.4 Οριζόντια προβολή διεπαφής χρήστη.....	22
4.4.5 Εκκίνηση Unity εφαρμογής.....	23
4.4.6 Καταμέτρηση βημάτων.....	23
4.4.7 Καθημερινή ειδοποίηση για χρήση της εφαρμογής.....	24
4.4.8 Υπολογισμός ημερησίων/συνολικών βημάτων/πόντων παιχνιδιών.....	25
4.4.9 Ενημέρωση απόδοσης χρήστη μέσω μηνύματος ηλεκτρονικού ταχυδρομείου.....	26
4.4.10 Προσθήκη φαρμακευτικής αγωγής.....	28

4.4.11 Αφαίρεση φαρμακευτικής αγωγής.....	30
4.4.12 Προβολή φαρμακευτικής αγωγής.....	30
4.4.13 Κλείδωμα ρυθμίσεων.....	31
4.4.14 Προφίλ χρήστη.....	33
4.5 Unity εφαρμογή.....	33
4.5.1 Πλοήγηση.....	33
4.5.2 Παιχνίδι αριθμών.....	34
4.5.3 Παζλ.....	37
4.5.4 Παιχνίδι καρτών.....	41
4.5.5 Ανάλυση των μηνυμάτων προς τον χρήστη.....	42
4.5.6 Προσθήκη βαθμολογίας παιχνιδιών στην Firestore.....	42
4.6 Επίλογος.....	42
Κεφάλαιο 5ο: Συμπεράσματα και προτάσεις βελτίωσης.....	44
5.1 Εισαγωγή.....	44
5.2 Συμπεράσματα.....	44
5.3 Βελτιώσεις.....	44
5.4 Επίλογος.....	45
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	46

Κατάλογος Σχημάτων

Σχήμα 1.1: Διάγραμμα γενικής λειτουργίας εφαρμογής.....	2
Σχήμα 4.2: Στιγμιότυπο της συλλογής email στην Firestore.....	17
Σχήμα 4.3: Στιγμιότυπο της συλλογής medication στην Firestore.....	17
Σχήμα 4.4: Δημιουργία λογαριασμού.....	19
Σχήμα 4.5: Στιγμιότυπο υπηρεσίας Authentication.....	20
Σχήμα 4.6: Οθόνη σύνδεσης χρήστη.....	21
Σχήμα 4.7: Βασικό μενού εφαρμογής.....	22
Σχήμα 4.8: Σύνδεση χρήστη σε οριζόντια προβολή.....	22
Σχήμα 4.9: Βασικό μενού εφαρμογής σε οριζόντια προβολή.....	23
Σχήμα 4.10: Διεπαφή καταμέτρησης βημάτων.....	24
Σχήμα 4.11: Στιγμιότυπο Notification υπενθύμισης χρήσης.....	25
Σχήμα 4.12: Στιγμιότυπο συλλογής συνολικών βημάτων.....	26
Σχήμα 4.13: Διεπαφή χρήστη εισαγωγής ηλεκτρονικής διεύθυνσης.....	27
Σχήμα 4.14: Παράδειγμα αυτοματοποιημένου ηλεκτρονικού μηνύματος.....	28
Σχήμα 4.15: Διεπαφή χρήστη προσθήκης φαρμακευτικής αγωγής καθημερινά.....	29
Σχήμα 4.16: Μέρος διεπαφής χρήστη προσθήκης φαρμακευτικής αγωγής συγκεκριμένες μέρες.....	29
Σχήμα 4.17: Εμφάνιση φαρμακευτικής αγωγής.....	31
Σχήμα 4.18: Διαχείριση κλειδώματος ρυθμίσεων.....	32
Σχήμα 4.19: Διεπαφή εισαγωγής κωδικού ρυθμίσεων.....	32
Σχήμα 4.20: Μενού πλοήγησης.....	33
Σχήμα 4.21: Διεπαφή πλοήγησης στην προηγούμενη οθόνη.....	34
Σχήμα 4.22: Διεπαφή παιχνιδιού αριθμών πριν την κάλυψη.....	34
Σχήμα 4.23: Διεπαφή παιχνιδιού αριθμών με την κάλυψη.....	35
Σχήμα 4.24: Διεπαφή παιχνιδιού αριθμών στο τρίτο λάθος.....	35
Σχήμα 4.25: Διεπαφή παιχνιδιού αριθμών (ενημέρωση χρήστη πως έχασε, αποκάλυψη αριθμών, κόκκινος χρόνος).....	37
Σχήμα 4.26: Μενού επιλογής επιπέδου στο παιχνίδι του παζλ.....	38
Σχήμα 4.27: Επίπεδο 1 του παζλ.....	38
Σχήμα 4.28: Επίπεδο 2 του παζλ.....	39
Σχήμα 4.29: Επίπεδο 3 του παζλ.....	39
Σχήμα 4.30: Ενημέρωσης νίκης στο επίπεδο του παζλ.....	41

Συντομογραφίες

ΔΙΠΙΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
Π.Ε.	Πτυχιακή Εργασία
NFTs	Neurofibrillary Tangles
REM	Rapid Eye Movement
RBD	REM Behavior Disorder
CDR	Clinical Dementia Rating
GDS	Global Deterioration Scale
SGDC	Serius Games for Dementia Caring
HAL	Hardware Abstraction Layer
ART	Android Runtime
DEX	Dalvik EXecutable
XML	Extensible Markup Language
SDK	Software Development Kit
BaaS	Backend-as-a-Service
JSON	Javascript Object Notation
GIMP	GNU Image Manipulation Program
SMTP	Simple Mail Transfer Protocol
MIME	Multipurpose Internet Mail Extensions

Κεφάλαιο 1ο: Εισαγωγή

1.1 Εισαγωγή

Στις μέρες μας, όλο ένα και περισσότερα προβλήματα λύνονται με την τεχνολογία. Οικονομικά, υγείας και άλλα προβλήματα, μελετώνται από τους επιστήμονες του κλάδου και επιλύονται. Επίσης, η τεχνολογία προσφέρει και αναβαθμίζει κάποιους τομείς όπως είναι η ψυχαγωγία, ο πολιτισμός και ο τουρισμός. Η τεχνολογία μπορεί να χρησιμοποιηθεί μέσω διάφορων συσκευών (κινητό τηλέφωνο, ηλεκτρική σκούπα, μηχάνημα ρομποτικής ιατρικής κ.α.).

Για τους περισσότερους ανθρώπους, αναπόσπαστο κομμάτι της καθημερινότητας τους είναι το κινητό τους τηλέφωνο. Στην συσκευή αυτή βρίσκονται εφαρμογές που προσφέρουν ένα πλήθος υπηρεσιών και οι οποίες βοηθούν (ημερολόγιο), καθοδηγούν (εφαρμογή γυμναστικής), εξοικονομούν χρόνο (εφαρμογή τράπεζας), ψυχαγωγούν (παιχνίδια). Ενώ μέχρι πριν από λίγα χρόνια τα κινητά χρησιμοποιούνταν περισσότερο από τους νέους ανθρώπους, όσο περνάει ο καιρός τα χρησιμοποιούν και άνθρωποι όλο και μεγαλύτερης ηλικίας.

Στα πλαίσια της τρέχουσας ΠΕ αναπτύχθηκε λογισμικό για κινητά τηλέφωνα με σκοπό την πρόληψη της άνοιας. Εκτός από πρόληψη, η χρήση της εφαρμογής μπορεί να υποβοηθήσει έναν ασθενή στα αρχικά στάδια της νόσου να κατευνάσει σε ένα βαθμό τα συμπτώματα. Η πιθανή διαδικασία υποβοήθησης περιγράφεται στο κεφάλαιο 2. Βέβαια, το λογισμικό αυτό μπορεί να χρησιμοποιηθεί από οποιονδήποτε χρήστη, με σκοπό την ψυχαγωγία και την εξάσκηση μνήμης με στόχο την καλύτερη ψυχική υγεία. Υπάρχει δυνατότητα κάποιων ρυθμίσεων στην εφαρμογή μέσω των οποίων μπορεί να ωφεληθεί ένας οποιοσδήποτε χρήστης ή ένας χρήστης μεγάλης ηλικίας του οποίου η πορεία παρακολουθείται από κάποιο οικείο πρόσωπο. Κυρίως σκοπός του λογισμικού είναι η εξάσκηση της μνήμης και η βελτίωση συγκέντρωσης του μυαλού. Η διεπαφή χρήστη είναι σχεδιασμένη με τέτοιο τρόπο ώστε να είναι απλή η χρήση του λογισμικού. Μεγάλα πλαίσια μακριά το ένα από το άλλο και μεγάλα γράμματα κάνουν την εμπειρία του χρήστη πιο εύκολη, αυξάνοντας την επιθυμία του για χρήση.

1.2 Τεχνολογίες ανάπτυξης λογισμικού

Η εφαρμογή αναπτύχθηκε με σκοπό να λειτουργεί σε Android λειτουργικό σύστημα κινητών. Το τελικό προϊόν έχει δύο μέρη (κεφάλαιο 4). Το ένα μέρος αφορά λειτουργίες όπως το προφίλ του χρήστη, την καταμέτρηση βημάτων, αποθήκευση φαρμακευτικής αγωγής και το άλλο μέρος αποτελείται από τα παιχνίδια που εξασκούν το μυαλό του χρήστη.

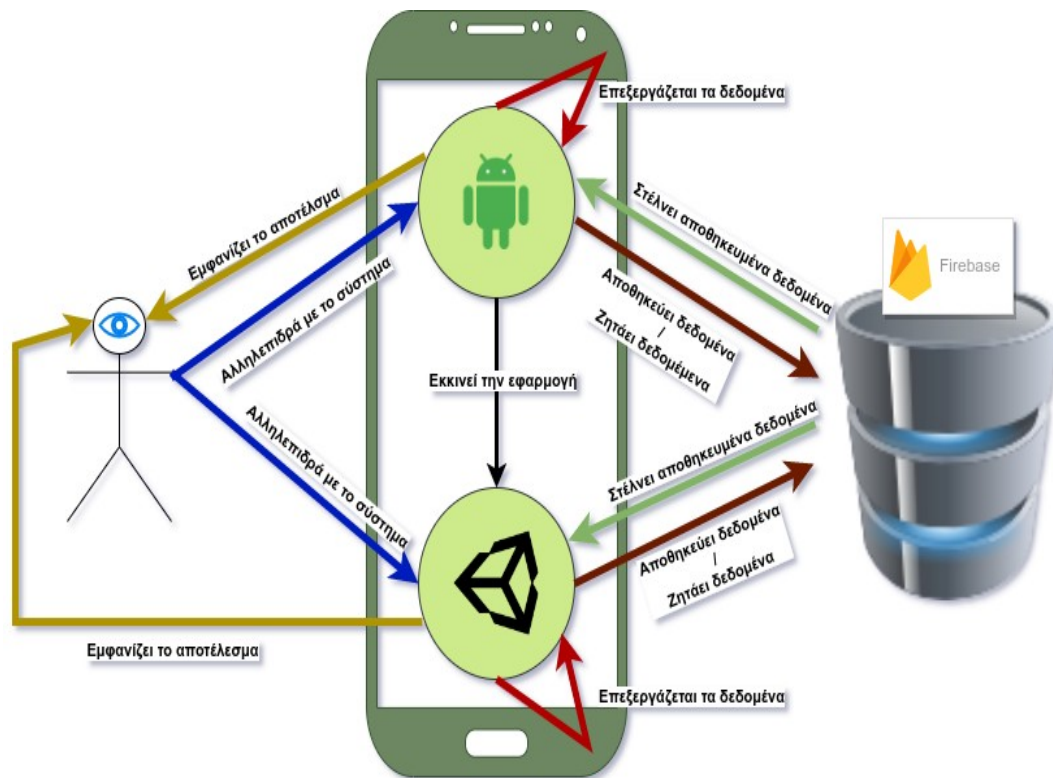
Το πρώτο μέρος προγραμματίστηκε στο Android Studio, που προσφέρει ένα εύχρηστο περιβάλλον για την ανάπτυξη κώδικα και κάνει αρκετά πιο απλή την δημιουργία διεπαφών χρήστη. Έγινε επιλογή του συγκεκριμένου λογισμικού καθώς μέσω απλών βημάτων είναι πολύ εύκολη η δημιουργία ενός νέου προγράμματος. Αυτό συμβαίνει γιατί ο προγραμματιστής αλληλεπιδρά με την διεπαφή του Android Studio και αυτό δημιουργεί όλα τα απαραίτητα αρχεία για το νέο πρόγραμμα.

Η γλώσσα που χρησιμοποιήθηκε είναι η Java. Η δυνατότητα αντικειμενοστρέφειας που δίνει η Java, βοηθά στην εύκολη διαχείριση και περιγραφή των αντικειμένων της εφαρμογής.

Για την δημιουργία των παιχνιδιών έγινε χρήση της πλατφόρμας Unity. Η πλατφόρμα αυτή είναι από τις πρώτες σε επιλογή για την ανάπτυξη παιχνιδιών παγκοσμίως. Προσφέρει ένα περιβάλλον με

πολλές δυνατότητες στον προγραμματιστή, το οποίο αντισταθμίζεται από την δύσκολη εκμάθηση της πλατφόρμας.

Για την διαχείριση των δεδομένων των χρηστών έγινε χρήση δύο υπηρεσιών της Firebase, της Authentication για την δημιουργία λογαριασμών και την αυθεντικοποίηση χρήστη, και της Firestore για την αποθήκευση δεδομένων του χρήστη. Αυτές οι τεχνολογίες επιλέχθηκαν καθώς εκτελούν στο παρασκήνιο κώδικα που επιτρέπει την εύκολη διαχείριση τους από τον πελάτη (client) που χρησιμοποιούνται. Στο σχήμα 1.1 φαίνονται οι αλληλεπιδράσεις μεταξύ του χρήστη, των δύο εφαρμογών και της Firebase.



Σχήμα 1.1: Διάγραμμα γενικής λειτουργίας εφαρμογής

1.3 Παρουσίαση κεφαλαίων

Στο επόμενο κεφάλαιο περιγράφεται αναλυτικά η νόσος. Δίνεται βαρύτητα στις οι πιο συνηθισμένες κατηγορίες, στους παράγοντες που προκαλούν την νόσο, σε τρόπους πρόληψης και στα συμπτώματα που έχει κάθε κατηγορία. Τέλος, γίνεται παρουσίαση κάποιων ερευνών που χρησιμοποίησαν τεχνολογικά μέσα για πρόληψη και καταπολέμηση της νόσου.

Στο τρίτο κεφάλαιο περιγράφονται οι τεχνολογίες που χρησιμοποιήθηκαν για την δημιουργία του λογισμικού. Πιο συγκεκριμένα, αναλύονται βασικά στοιχεία που χρησιμοποιούνται γενικότερα από την πλατφόρμα Android, κάποιες υπηρεσίες της Firebase και η Unity πλατφόρμα ανάπτυξης παιχνιδιών.

Στο τέταρτο κεφάλαιο αναλύεται η υλοποίηση της εφαρμογής της ΠΕ. Περιγράφεται η λογική σχεδίασης της διεπαφής χρήστη, οι λειτουργίες που υλοποιήθηκαν, τα παιχνίδια, προβλήματα που εντοπίστηκαν κατά την ανάπτυξη του λογισμικού και οι τρόποι που αυτά αντιμετωπίστηκαν.

Στο τελευταίο κεφάλαιο αναφέρονται συμπεράσματα που εξήχθησαν από την ανάπτυξη του λογισμικού. Τέλος, περιγράφονται κάποιες προσθήκες, βελτιώσεις ή αλλαγές που θα μπορούσαν να δημιουργήσουν μία εφαρμογή που θα είχε καλύτερα αποτελέσματα για τους χρήστες.

Κεφάλαιο 2ο: Άνοια

2.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζεται αναλυτικά η νόσος και ο τρόπος που επηρεάζει τις ζωές των ασθενών. Αναφέρονται κάποιες σημαντικές κατηγορίες της άνοιας, συμπτώματα, και τρόποι πρόληψης. Τονίζεται ο τρόπος που η τεχνολογία μπορεί να προστατέψει από την νόσο, ή να μειώσει τα συμπτώματα σε κάποιο βαθμό.

2.2 Η νόσος της άνοιας

Η άνοια, περιγράφεται ως μια πάθηση, προοδευτικών νευροεκφυλιστικών διαταραχών του εγκεφάλου [1]. Επηρεάζει λειτουργίες, όπως είναι η μνήμη, η σκέψη, ο προσανατολισμός, η κατανόηση, ο υπολογισμός, η ικανότητα της μάθησης και η κριτική σκέψη. Διαταράσσει την ζωή του ασθενή, καθώς πλήττει τόσο την σωματική όσο και την ψυχολογική του υγεία. Το χαρακτηριστικό της άνοιας είναι ότι είναι χρόνια και προοδευτική.

Οι λόγοι που μπορούν να προκαλέσουν σε ένα άτομο άνοια, μπορεί να είναι γενετικοί ή περιβαλλοντικοί. Η αύξηση της ηλικίας του ατόμου, συνεπάγεται αύξηση της πιθανότητας νόσησης. Ο τρόπος ζωής του ατόμου είναι επίσης καθοριστικός παράγοντας στην εμφάνιση και την επιδείνωση. Η παχυσαρκία, το κάπνισμα, η υψηλή πίεση, ο διαβήτης και οι μεταβολικές διαταραχές (για παράδειγμα διαταραχές που σχετίζονται με το αλκοόλ) αποτελούν πιθανά αίτια. Η εξάσκηση μνήμης, η σωματική άσκηση, το υψηλό μορφωτικό επίπεδο και IQ έχει αποδειχθεί ότι δρουν προληπτικά και αποτρεπτικά.

2.3 Τύποι άνοιας

2.3.1 Η νόσος Alzheimer

Η νόσος Alzheimer, είναι μία κατηγορία άνοιας [2]. Αντιπροσωπεύει το 70% των περιπτώσεων. Η πιθανότητα νόσησης αυξάνεται όσο αυξάνεται και η ηλικία. Έχει ως αποτέλεσμα την αποδυνάμωση της μνήμης, σε συνδυασμό με κάποια άλλη δυσλειτουργία όπως για παράδειγμα, διαταραχή σε ομιλία, προσωπικότητα ή/και κριτική σκέψη. Η ασθένεια είναι ικανή να επηρεάσει την ψυχική και σωματική υγεία του ασθενή σε πολύ μεγάλο βαθμό, ανάλογα με τις περιοχές του εγκεφάλου που πλήττονται.

Η διάγνωση γίνεται με τον εντοπισμό γεροντικών πλακών και των πρωτεϊνών Neurofibrillary Tangles (NFTs). Οι γεροντικές πλάκες είναι κατηγορία παθολογικών πρωτεϊνών, όπου γίνεται η συσσώρευση του Αβ πεπτιδίου (παράγεται από τον μεταβολισμό της πρωτεΐνης του αμυλοειδούς- amyloid precursor protein, APP). Οι NFTs ανήκουν στις πρωτεΐνες tau (πρωτεΐνες υπεύθυνες για νευροεκφυλιστικές ασθένειες), και αποδεδειγμένα εμποδίζουν την ροή πληροφοριών ανάμεσα στα τμήματα του εγκεφάλου. Η APOE4, μπορεί επίσης να καταστρέψει νευρώνες του εγκεφάλου, καθώς επηρεάζει την ομοιοστατική παραγωγή του Αβ πεπτιδίου που αναφέρθηκε παραπάνω [3]. Σε προχωρημένο στάδιο στερεί από το άτομο την ικανότητα να μπορεί να ζήσει ανεξάρτητο, και να οδηγήσει σε θάνατο [2].

2.3.2 Αγγειακή άνοια

Η Αγγειακή άνοια οφείλεται σε εγκεφαλικές και αγγειακές δυσλειτουργίες και καλύπτει περίπου το 20% των περιπτώσεων [4]. Δυσκολεύει την παροχή οξυγόνου, θρεπτικών συστατικών και σημάτων από τους νευρώνες στον εγκέφαλο. Σε κάποιες περιπτώσεις μπορεί να θεωρηθεί εξέλιξη της νόσου Alzheimer, και από ότι έχει παρατηρηθεί δημιουργεί μεγαλύτερες αλλαγές στα τμήματα του εγκεφάλου που επηρεάζει [5].

Η πορεία της νόσου, βασίζεται σε αγγειακές παθολογίες όπως είναι το λανθασμένο εγκεφαλικό επεισόδιο, εγκεφαλικό επεισόδιο φλοιού, εγκεφαλικό επεισόδιο υποφλοιού, αλλοιώσεις λευκής ουσίας (που επηρεάζει την ικανότητα μάθησης), καρδιογενείς εμβολές, υποτασικό έμφραγμα, υπερτασική εγκεφαλική αιμορραγία, λοβιακή αιμορραγία και υπαραχνοειδής αιμορραγία. Η διάγνωση προκύπτει μέσω μηχανημάτων ακτίνων X και μαγνητικής τομογραφίας τα οποία εντοπίζουν οργανικές αλλαγές στον φλοιό του εγκεφάλου και αλλοιώσεις λευκής ουσίας. Το άτομο που νοσεί εμφανίζει συχνά κατάθλιψη και/ή κρατάει απαθή στάση στην ζωή του [6]. Λιγότερο συχνά συμπτώματα αποτελούν οι παραισθήσεις. Δεν υπάρχει κάποια αποτελεσματική θεραπεία για την αγγειακή άνοια. Έχουν χρησιμοποιηθεί ως φάρμακα οι αναστολείς χολινεστεράσης και η μεμαντίνη που περιορίζουν, ως έναν βαθμό, την νευροπαθολογία των ασθενών με αγγειακή άνοια.

2.3.3 Άνοια σωματιδίων Lewy

Η άνοια σωματιδίων Lewy, είναι μία σημαντική κατηγορία άνοιας, που καλύπτει πάνω από το 20% των περιπτώσεων [7]. Δεν έχουν ακόμα αποσαφηνιστεί οι αιτίες που προκαλούν άνοια αυτής της κατηγορίας, εκτός του ότι έχει παρατηρηθεί η συχνότερη εμφάνιση σε άντρες.

Πλήττει τον φλοιό και υποφλοιό, προκαλώντας έτσι προβλήματα οπτικής αντίληψης, προσοχής και σωματικής δυσλειτουργίας. Πιο συγκεκριμένα, από αρχικά ακόμα στάδια ο ασθενής εμφανίζει γνωστικές διαταραχές, οι οποίες διακρίνονται για την μεταβολή της έντασης τους στο χρόνο. Μία πολύ συχνή επίπτωση της άνοιας σωματιδίων Lewy, που προσβάλλει το 80% των ασθενών Lewy είναι οι παραισθήσεις. Οι λόγοι για αυτό το υψηλό ποσοστό παραμένουν άγνωστοι, παρ' όλο που φαίνεται να σχετίζονται με την έλλειψη οπτικής αντίληψης και προσοχής. Μία ακόμα επίπτωση είναι οι υπνικές δυσλειτουργίες που χαρακτηρίζονται από διατάραξη του στάδιου Rapid Eye Movement (REM), και ανήκουν στην κατηγορία REM Behavior Disorder (RBD). Το RBD δημιουργεί την αίσθηση πως το όνειρο που βλέπει κάποιος είναι αληθινό και τις περισσότερες φορές τα όνειρα αυτά είναι τρομακτικά.

Ένα σημαντικό ποσοστό της τάξης του 25-50% εμφανίζει Παρακινσονισμό (εξωπυραμιδική συνδρομή), δηλαδή συμπτώματα παρόμοια με αυτά της νόσου Parkinson. Για μία επιτυχημένη αντιμετώπιση των συμπτωμάτων προτείνεται ο συνδιασμός φαρμακευτικών (Levodopa monotherapy, selegiline, amantadine catechol-O-methyltransferase, ντοπαμίνη) και μη, μεθόδων (εξάσκηση μνήμης, εκπαίδευση κ.α.).

2.3.4 Άνοια της νόσου Parkinson

Πρόκειται για την δεύτερη πιο συχνά εμφανιζόμενη νευροεκφυλιστική ασθένεια, της οποίας η εμφάνιση αυξάνεται κατακόρυφα από τα 60 έτη και έπειτα [8]. Η ασθένεια ουσιαστικά, είναι η προοδευτική απώλεια των νευρώνων της ντοπαμίνης.

Η μακροχρόνια έκθεση σε φυτοφάρμακα και βαρέα μέταλλα έχει συσχετιστεί με την ανάπτυξη της νόσου. Αλλοιώσεις στο DNA στο στάδιο της μεταγραφής, είναι επίσης πιθανά αίτια νόσησης.

Συμπτώματα αποτελούν το τρέμουλο, η βραδυκινησία, η μυϊκή δυσκαμψία, η δυσκολία στην ισορροπία και άλλα.

Αποτελεσματικός τρόπος αποκατάστασης των συμπτωμάτων, έχει αποδειχθεί η μεταμόσχευση νευρώνων παραγωγής ντοπαμίνης, το οποίο βέβαια ενέχει κάποιους κινδύνους και πιθανές αποτυχίες [9]. Ως φαρμακευτική αγωγή χρησιμοποιείται η λεβοντόπα (φάρμακο που όταν εισέρχεται στον εγκέφαλο μετατρέπεται σε ντοπαμίνη), αν και με την πάροδο του χρόνου, το φάρμακο αυτό χάνει την αποτελεσματικότητά του ακόμα και στην περίπτωση αύξησης της δοσολογίας.

2.3.5 Μετοποκροταφική άνοια

Η μετοποκροταφική άνοια προσβάλλει το μετωπιαίο και κροταφικό λοβό [10]. Προκαλεί διαφοροποίηση της προσωπικότητας, και της κοινωνικότητας του ατόμου. Μπορεί να δημιουργήσει ψυχιατρικά προβλήματα όπως είναι η κατάθλιψη. Μείωση της ευφράδειας και δυσκολία στην εκφορά του λόγου είναι δύο πιθανά συμπτώματα, τα οποία μάλιστα σχετίζονται με την ατροφία του αριστερού ημισφαιρίου. Η ατροφία του μέσου και κατώτερου κροταφικού νεοφλοιού προκαλούν την μείωση των ερεθισμάτων από εξωτερικούς παράγοντες και την μη κατανόηση ανθρώπων και αντικειμένων.

Η διάγνωση γίνεται με τον εντοπισμό ανωμαλιών στους μετοποκροταφικούς λοβούς και στην απουσία νευρολογικών σημάτων. Δυστυχώς δεν υπάρχει ακόμα κάποια μέθοδος αντιμετώπισης της μετοποκροταφικής άνοιας. Η μείωση των συμπτωμάτων περιορίζεται στην ψυχιατρική υποστήριξη των ασθενών.

2.4 Κατηγοριοποίηση της άνοιας βάσει προτύπων

Η πορεία της άνοιας και πιο πολύ για τη νόσος Alzheimer, είναι δυνατό να καταγραφεί με βάση δύο πρότυπα. Αυτά είναι το Clinical Dementia Rating (CDR) και το Global Deterioration Scale (GDS). Η αξιολόγηση είναι σημαντική καθώς με βάση αυτή γίνεται η σωστή χορήγηση φαρμάκων για την βελτίωση των συμπτωμάτων.

Το CDR αξιολογεί τα εξής: την μνήμη, τον προσανατολισμό, την κοινωνική ζωή, τις δραστηριότητες και την ικανότητα της προσωπικής φροντίδας [11]. Η αξιολόγηση της κάθε μίας λειτουργίας πρέπει να είναι όσο το δυνατόν ανεξάρτητη και να μην επηρεάζεται από τις άλλες. Η μνήμη ορίζεται ως κύρια κατηγορία και οι υπόλοιπες ως δευτερεύουσες. Έτσι λοιπόν, στην περίπτωση που τουλάχιστον τρεις δευτερεύουσες δίνουν την ίδια βαθμολογία με την μνήμη τότε ο ασθενής αξιολογείται με βάση την μνήμη (CDR=Memory). Στην περίπτωση τριών ή περισσότερων δευτερευουσών μεγαλύτερων της κύριας, τότε ο δείκτης CDR παίρνει την βαθμολογία αυτών. Με ανάλογο τρόπο προκύπτουν και τα αποτελέσματα για κάθε περίπτωση.

Το μοντέλο GDS, είναι αξιολόγηση επτά επιπέδων που είναι τα εξής: 1 ως μη γνωστική εξασθένηση, 2 πολύ ήπια γνωστική εξασθένηση, 3 ήπια γνωστική εξασθένηση, 4 μέτρια, 5 μέτρια προς σοβαρή, 6 σοβαρή, 7 πολύ σοβαρή [12]. Ανάλογα το επίπεδο, ένας ασθενής μπορεί να χαρακτηριστεί ότι βρίσκεται σε ήπια άνοια (επίπεδο 1) έως και τελικού σταδίου (επίπεδο 7). Η αξιολόγηση και τοποθέτηση σε κάποιο επίπεδο γίνεται με διαφορετικές μεθόδους. Οι έλεγχοι αυτοί γενικά είναι τύπου γνωστικής λειτουργίας, συμπεριφοράς, ακεραιότητας νευροανατομικής δομής, βαθμός αυτοεξυπηρέτησης και άλλοι.

2.5 Εξέλιξη των συμπτωμάτων γενικά

Τα συμπτώματα μπορούν να επηρεάσουν κάθε πτυχή της ζωής ενός ασθενή [13]. Γενικά, η νόσος ξεκινάει με ήπια συμπτώματα που επηρεάζουν την μνήμη και την συμπεριφορά του ατόμου, κατάσταση που οξύνεται σε βάθος χρόνου. Σταδιακά, τα συμπτώματα κάνουν την εμφάνισή τους και σε σωματικό επίπεδο. Στο επόμενο στάδιο οι ασθενείς δεν μπορούν πλέον να εξυπηρετήσουν βασικές τους ανάγκες και είναι απαραίτητη η φροντίδα από ειδικούς.

2.6 Πνευματική εξάσκηση και πρόληψη άνοιας

Όπως έχει αναφερθεί, και παραπάνω ένας σημαντικός παράγοντας για το ποσοστό νόσησης είναι ο τρόπος ζωής του ατόμου. Η εξάσκηση του μυαλού έχει προστατευτικό ρόλο απέναντι στην νόσο. Το New England Journal of Medicine, δημοσίευσε μία μελέτη σχετικά με τα παραπάνω [14]. Γενικότερα αναφέρεται πως όσοι παίζουν συχνά επιτραπέζια παιχνίδια έχουν 35% λιγότερες πιθανότητες να αναπτύξουν άνοια σε σύγκριση με ανθρώπους που παίζουν σπάνια. Επίσης, άνθρωποι μεγαλύτεροι της ηλικίας των 75, οι οποίοι στο ελεύθερο τους χρόνο ασχολούνται με δραστηριότητες που ενεργοποιούν τον εγκέφαλο, έχουν εξίσου λιγότερες πιθανότητες νόσησης.

Η εν λόγω μελέτη διήρκεσε πέντε χρόνια, συμμετείχαν σχεδόν 500 άνθρωποι, και έχει συμπεράνει, πέραν της πρόληψης, πως η ενασχόληση με αρκετές δραστηριότητες μπορεί να καθυστερήσει αποδεδειγμένα την ανάπτυξη άνοιας κατά 1.5 έτος. Η ενασχόληση με το σκάκι μπορεί να δράσει προστατευτικά, καθώς έρευνες δείχνουν ότι επηρεάζει συγκεκριμένα μέρη του εγκεφάλου, αφού δρα με πολύπλοκο τρόπο. Ακόμη παρατηρείται, πως η δραστηριότητα αυτή, και κατ' επέκταση, κάθε ανάλογη πνευματική εξάσκηση, λειτουργεί θετικά όχι μόνο προληπτικά, αλλά και μετά την την διάγνωση της ίδιας της ασθένειας στο άτομο [15]. Ως παρόμοιας λειτουργίας και αποτελέσματος τονίζονται επίσης το λύσιμο σταυρόλεξων, τα επιτραπέζια παιχνίδια καθώς και η ενασχόληση με κάποιο μουσικό όργανο.

Ανάλογα αποτελέσματα είχε και η κοορτή του Raquid που έλαβε χώρα στο Bordeaux της Γαλλίας, με συμμετέχοντες 3777 συμμετέχοντες, όπου διαπιστώθηκε πως οι 102 (2,7%) έπασχαν ήδη από άνοια, οπότε και αποκλείστηκαν [16]. Από τους υπόλοιπους 3675 συμμετέχοντες, οι 1181 (32,2%) δήλωσε την ενασχόληση του με επιτραπέζια παιχνίδια. Αυτό το ποσοστό αντιπροσωπεύει άτομα που ήταν νεότερα σε ηλικία, πιο μορφωμένα, έκαναν περισσότερους από έναν γάμο, καθώς και πως είχαν λιγότερο άγχος στην ζωή τους. Επίσης το ποσοστό των ατόμων μεγαλύτερης ηλικίας 85-89 ετών αντιπροσωπεύει δείγμα της τάξης 18%, και 12% άτομα άνω των 89. Το 18,8% είχαν ηλικία 20 έως 23 ετών, και το 10,6% μικρότεροι των 20 ετών. Αξιοσημείωτο το γεγονός πως σε αιματολογικό δείγμα από 623 ανθρώπους, βρέθηκαν παρόμοια ποσοστά λιποπρωτεΐνης E (πρωτεΐνη ύποπτη για την νόσο του Alzheimer) ανάμεσα σε άτομα που έχουν ως ενασχόληση ή όχι τα επιτραπέζια παιχνίδια, που σημαίνει την μη επιρροή των αποτελεσμάτων από αυτό τον παράγοντα. Σύμφωνα με την έρευνα που είχε διάρκεια 20 έτη διαπιστώθηκαν τα παρακάτω. Από τους τελικά 3670 συμμετέχοντες, οι 2987 (81,4%) εξετάστηκαν ξανά μέσα σε αυτό το διάστημα. Οι 142 (3,9%) πέθαναν πριν την επανεξέτασή τους. Οι 541 (14,7%) αρνήθηκαν να συμμετέχουν σε κάποια επανεξέταση. Οι 804 (27,8%) συμμετέχοντες ανέπτυξαν άνοια. Συμπερασματικά όμως τα άτομα που είχαν επαφή με παιχνίδια εξάσκησης του μυαλού, είχαν 15% λιγότερες πιθανότητες να αναπτύξουν άνοια. Συγκεκριμένα σε διάστημα τριών ετών 3% των ανθρώπων που ασχολούνταν με παιχνίδια, ανέπτυξαν άνοια, ενώ για τα άτομα που δεν έπαιζαν το ποσοστό αυτό έφτανε το 6%. Τα οφέλη αυτής της συχνής ενασχόλησης με την εξέλιξη των πνευματικών δεξιοτήτων παρατηρούνται για 10 με 20 χρόνια, παρ' όλο που φαίνεται πως μειώνονται δραστικά εν συναρτήσει ανάπτυξης κάποιου είδους κατάθλιψης, που όπως έχει

αναφερθεί είναι σημαντικός παράγοντας για την ανάπτυξη της άνοιας. Έτσι, διαπιστώνει κανείς πως τα ευχάριστα συναισθήματα που δημιουργούνται κατά την διάρκεια του κάθε παιχνιδιού, παίζουν και αυτά τον ρόλο τους στην πρόληψη.

2.7 Ο ρόλος της τεχνολογίας

Η τεχνολογία έχει εισβάλλει σε κάθε τομέα της ζωής των ανθρώπων. Σκοπός της είναι να κάνει την ζωή τους πιο εύκολη. Η εξέλιξη τα τελευταία χρόνια είναι τόσο ραγδαία που ο καθένας πλέον έχει πάντα μαζί του κάποιο υπολογιστικό σύστημα (πχ. smartphone). Η χρήση ενός υπολογιστή μπορεί να μεταβάλλει την ψυχική υγεία του ανθρώπου. Το αποτέλεσμα αυτής της μεταβολής οφείλεται πάντα στο είδος της χρήση που γίνεται. Σύμφωνα με έρευνες που ασχολούνται με την χρήση παιχνιδιών σε υπολογιστικά συστήματα, έχουν την δυνατότητα πρόληψης άνοιας και διόρθωσης μερικών συμπτωμάτων (σε κάποιο βαθμό) [17-19]. Ένα παραπάνω λόγο καλύτερης επίτευξης των παραπάνω έχουν οι συσκευές με διαδραστικές οθόνες αφής (smartphones, tablets), καθώς δικαίως θεωρούνται πιο εύχρηστες και πιο εύκολα διαχειρίσιμες από μεγαλύτερους σε ηλικία ανθρώπους.

2.7.1 Πρόληψη της νόσου με την βοήθεια παιχνιδιών σε ψηφιακή μορφή

Η εφαρμογές με παιχνίδια μνήμης μπορούν να δράσουν προληπτικά έναντι στην άνοια [17-18]. Εκκινούν λειτουργίες του εγκεφάλου και τον αποτρέπουν από την αδράνεια. Με αυτό τον τρόπο είναι μειώνονται και οι πιθανότητες νόσησης και εμφάνισης συμπτωμάτων. Πιο συγκεκριμένα, τα χαρακτηριστικά που θέλουν να διεγείρουν αυτές οι εφαρμογές είναι η μνήμη, η προσοχή, η ικανότητα επίλυσης προβλημάτων και η λήψη αποφάσεων. Για να επιτευχθεί πιο εύκολα αυτό, απαιτείται και η καθημερινή ενασχόληση του χρήστη με το λογισμικό.

Η χρήση έξυπνων συσκευών από μεγάλους σε ηλικία ανθρώπους τις περισσότερες φορές είναι δύσκολη [17]. Για τον λόγο αυτό, οι εφαρμογές που αναπτύσσονται για την άνοια πρέπει να είναι πολύ εύκολες στην χρήση. Σε αυτό πολλές φορές μπορούν να φανούν χρήσιμοι οι ήχοι καθοδήγησης χρήστη και η αυτόματη ενημέρωση της κατάστασης της εφαρμογής χωρίς την χειροκίνητη υποβολή κάποιας απάντησης. Σύμφωνα με έρευνα, σε μία καλά σχεδιασμένη εφαρμογή, περίπου το 50% ηλικιωμένων ατόμων κατάφερε να αλληλεπιδράσει και να παίξει αυτόνομα χωρίς καμία βοήθεια.

Τα παιχνίδια στο μέλλον θα αποτελούν μέρος ιατρικής φροντίδας σε άτομα με γνωστικές ασθένειες. Ενώ τα δεδομένα και τα "σκορ" που θα καταφέρνει ο χρήστης, θα αποτελούν μέρος της διάγνωσης

2.7.2 Τεχνολογική αντιμετώπιση της ασθένειας

Οι ερευνητές, προσπαθούν συνεχώς να ανακαλύψουν μεθόδους αντιμετώπισης της άνοιας. Στην προσπάθεια αυτή αναπτύχθηκε το μοντέλο Serious Games for Dementia Caring (SGDC) [19]. Ως στιγμή φαίνεται να έχει μεγαλύτερη επίδραση σε σχέση με άλλες προσπάθειες θεραπείας (πχ. ψυχολογικές και φαρμακευτικές μέθοδοι).

Οι μεγάλες κατηγορίες παιχνιδιών που υποστηρίζει η μέθοδος SGDC είναι επιτραπέζια παιχνίδια, βιντεοπαιχνίδια και παιχνίδια εικονικής πραγματικότητας. Αρχικά χρησιμοποιήθηκαν κλασικά παιχνίδια όπως Huarongdao, και παζλ. Στην συνέχεια όμως σχεδιάστηκαν παιχνίδια ειδικά για την καταπολέμηση της άνοιας στο κομμάτι της γνωστικής δυσλειτουργίας, όπως είναι τα MINWii (παιχνίδι δημιουργίας μουσικής), Big Brain Academy (σύνολο παιχνιδιών πνευματικής εξάσκησης), και άλλα. Για την αντιμετώπιση της σωματικής δυσλειτουργίας, αναπτύχθηκαν εφαρμογές όπως WiiFit και WiiSports [19]. Ο εντοπισμός της επίδρασης των παραπάνω γίνεται με την βοήθεια των φυσιολογικών σημάτων (π.χ. ηλεκτροεγκεφαλογράφημα).

Για κάθε στάδιο της νόσου, πρέπει να χρησιμοποιούνται τα κατάλληλα SGDC. Στην περίπτωση της ήπιας άνοιας, τα παιχνίδια που θα χρησιμοποιηθούν έχουν ως σκοπό την αναγνώριση του μεγέθους του προβλήματος, αλλά και την βοήθεια σε κάποιο βαθμό για την αποκατάσταση της γνωστικής δυσλειτουργίας. Για μέτριου σταδίου άνοια, επιλέγονται παιχνίδια τόσο για την πνευματική όσο και στην σωματική ζωή του ασθενή. Για προχωρημένες περιπτώσεις όμως, δεν μπορεί να αξιοποιηθεί αυτή η μέθοδος καθώς το άτομο έχει χάσει την ικανότητα του πλέον να μπορεί να αλληλεπιδράσει με τα SGDC.

Σε έρευνα που διεξήχθη το 2012-2014 σε ασθενείς με άνοια συμμετείχαν 54 ασθενείς, (24 άνδρες, 30 γυναίκες), με μέση ηλικία τα 83.5 έτη. Τα άτομα αυτά, κλήθηκαν να περάσουν χρόνο παίζοντας παιχνίδια σε κάποιο tablet. Κατά την έρευνα, μελετήθηκε ο τρόπος αλληλεπίδρασης των ασθενών με τις εφαρμογές που τους δόθηκαν [20].

Τα στοιχεία που διέκριναν τους ασθενείς, ήταν η ηλικία, το φύλο, ο βαθμός εμπειρίας με την τεχνολογία, το επάγγελμα που ασκούσαν στο παρελθόν, καθώς και ο βαθμός και ένταση που νοσούσαν. Σύμφωνα με τους ψυχολόγους και το ιατρικό προσωπικό που επέβλεπαν την έρευνα αυτή, τα παιχνίδια που δόθηκαν στους ασθενείς, προτάθηκαν από υψηλόβαθμο μέλος της μονάδας φροντίδας. Χαρακτηριστικά παραδείγματα, αποτέλεσαν παιχνίδια καρτών, εφαρμογή μουσικού οργάνου (για παράδειγμα ξυλόφωνο), παιχνίδια αλληλεπίδρασης ήχου ή εικόνας, εικονικό κατοικίδιο.

Αξίζει να σημειωθεί πως κατά την διάρκεια της ενασχόλησης του κάθε μέλους της έρευνας με το tablet, έγινε κάλυψη του διακόπτη τροφοδοσίας του μηχανήματος καθώς και της έντασης ήχου, με σκοπό να αποφευχθεί η κάθε πιθανή σύγχυση ή απώλεια προσοχής του ασθενή και κατ' επέκταση εξαγωγή λανθασμένων συμπερασμάτων. Για την έρευνα αυτή, χρησιμοποιήθηκε συγκεκριμένο πρωτόκολλο παρατήρησης των συμμετεχόντων. Πιο συγκεκριμένα, η παρατήρηση από τους ειδικούς, ξεκινούσε 15 λεπτά πριν την έναρξη της εκάστοτε ενασχόλησης και έληγε 15 λεπτά αφότου ο ασθενής ολοκλήρωνε την διάδραση. Επίσης, έγινε τοποθέτηση του ενός παρατηρητή με τέτοιο τρόπο ώστε υπάρχει μία πολύ καλή οπτική γωνία ως προς το πρόσωπο και την στάση σώματος του συμμετέχοντα. Ένας άλλος παρατηρητής ήταν δίπλα στον ασθενή, με σκοπό να αντιληφθεί τον τρόπο διάδρασης του συμμετέχοντα με το υπολογιστικό σύστημα. Η στάση τους έπρεπε να είναι φιλική και ουδέτερη, και να μην παρέμβουν σε καμία περίπτωση.

Μετά το πέρας της κάθε παρατήρησης, ο ασθενής ρωτιόταν αν του άρεσε αυτό που έκανε, αν θα το ξαναέκανε, καθώς και το πως θα το βαθμολογούσε με κλίμακα από το 1 έως το 10. Πριν αναφερθούν τα συμπεράσματα της έρευνας, αξίζει να αποδοθούν τα εξής στοιχεία. Το 88% δεν είχε κάποια προηγούμενη ενασχόληση με υπολογιστή. Από τους 54 ασθενείς, οι 23 είχαν "μέτρια προς προχωρημένη" άνοια και οι 31, "ήπια προς μέτρια". Καθ' όλη την διάρκεια της έρευνας πραγματοποιήθηκαν 177 κύκλοι παρατηρήσεων, από τους οποίους οι 17 αφορούν περιπτώσεις που τελικά ο ασθενής δεν χρησιμοποίησε καθόλου το tablet πράγμα που αφορά είτε την δημιουργία μη επιθυμίας λόγω προηγούμενης αρνητικής απόδοσης, είτε απλά την διάθεση του. Όσον αφορά τις υπόλοιπες 160 περιπτώσεις, ο μέσος χρόνος ενασχόλησης ήταν τα 20.9 λεπτά με 8.9 λεπτά απόκλιση. Πιο συγκεκριμένα παρατηρήθηκε ότι περιπτώσεις "ήπιες προς μέτριες" έκαναν χρήση για 23.6 με 8.2 λεπτά απόκλιση, ενώ οι "μέτριες προς προχωρημένες" 16.8 με 8.3 απόκλιση. Στις ερωτήσεις που διατυπώθηκαν μετά το πέρας του κάθε κύκλου παρατήρησης, κατά πλειοψηφία οι ερωτηθέντες απαντούσαν πως είχαν αίσθημα αυτοπεποίθησης, ανάγκη για περαιτέρω αυτοβελτίωση, πως ανήκουν κάπου και διεκπεραιώνουν κάποια πρόκληση. Υπήρχαν βέβαια και οι περιπτώσεις αρνητικών απαντήσεων όταν για παράδειγμα το παιχνίδι δεν τραβούσε την προσοχή, δεν κάλυπτε κάποιες προσδοκίες, ή δεν τα πήγαιναν καλά.

Κεφάλαιο 2

Ως τελικό πόρισμα εξάγεται ότι ακόμα και οι ήδη πάσχοντες από άνοια είναι σε θέση να διαχειριστούν, διασκεδάσουν εξασκήσουν τις πνευματικές τους λειτουργίες και την μνήμη τους μέσω τέτοιου είδους παιχνιδιών και να μειώσουν την όποια ψυχολογική βλάβη και έλλειψη αυτοπεποίθησης τους έχει προκληθεί μέσω της συνολικής κατάστασης, τα οποία συνεπάγονται σε κάποια βελτίωση ή σταθεροποίηση της κατάστασης της υγείας τους.

2.8 Επίλογος

Στο κεφάλαιο αυτό, έγινε κάλυψη της νόσου της άνοιας. Συμπερασματικά, προκύπτει πως μπορεί επιτευχθεί πρόληψη και ίσως βελτίωση κάποιων συμπτωμάτων με την χρήση της εφαρμογής (ή παρόμοιων εφαρμογών) που αναπτύσσεται στα πλαίσια της ΠΕ.

Κεφάλαιο 3ο: Τεχνολογίες που χρησιμοποιήθηκαν

3.1 Εισαγωγή

Στο κεφάλαιο αυτό αναλύονται οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής. Πιο συγκεκριμένα, η πλατφόρμα Android που αποτελεί τον κορμό της εφαρμογής, κάποιες υπηρεσίες (services) της Firebase και η Unity πλατφόρμα, για την δημιουργία παιχνιδιών. Τέλος, αναφέρονται οι εφαρμογές που χρησιμοποιήθηκαν για την παραγωγή εικονιδίων που χρησιμοποιήθηκαν στην διεπαφή χρήστη και για την υλοποίηση του διαγράμματος 1.1.

3.2 Android

Είναι μία ανοιχτού λογισμικού πλατφόρμα βασισμένη στον πυρήνα των Linux [21]. Πρόκειται για μία στοίβα λογισμικού (software) για την δημιουργία εφαρμογών για το Android λειτουργικό σύστημα. Ξεκινώντας από το χαμηλότερο σημείο στην στοίβα μετά τον Linux πυρήνα, βρίσκεται το Hardware Abstraction Layer (HAL) που είναι υπεύθυνο για την επικοινωνία του υλικού (hardware) με την Java διεπαφή προγραμματισμού εφαρμογών (Application Programming Interface) για το οποίο θα γίνει λόγος στην συνέχεια. Αμέσως μετά βρίσκεται το Android Runtime (ART) που μεταφράζει τον Java κώδικα σε Dalvik EXecutable (DEX) κώδικα ψηφιολέξης (byte code). Οι C/C++ βιβλιοθήκες βρίσκονται στο επόμενο επίπεδο, και διαχειρίζονται υπηρεσίες του πυρήνα. Ακολουθεί το πλαίσιο (framework) Java διεπαφής προγραμματισμού εφαρμογών και στο τελευταίο επίπεδο της στοίβας βρίσκονται οι τελικές εφαρμογές.

3.2.1 Java API framework

Για την υλοποίηση μίας εφαρμογής σε Android, χρησιμοποιείται ένα σύνολο από διεπαφές προγραμματισμού εφαρμογών. Οι διεπαφές αυτές προσφέρουν ένα σύνολο από έτοιμες κλάσεις και μεθόδους, διευκολύνουν τον προγραμματιστή καθώς μειώνουν τον χρόνο ανάπτυξης μίας εφαρμογής, την περιπλοκότητα και το μέγεθος του. Παρακάτω αναλύονται κάποιες σημαντικές διεπαφές προγραμματισμού εφαρμογών Android εφαρμογών.

3.2.2 View System

Πρόκειται για την διεπαφή χρήστη (User Interface) που δημιουργείται με XML (Extensible Markup Language) αρχεία [22]. Το αρχείο αυτό αναπαριστά την δομή της διεπαφής χρήστη. Μπορεί να του δοθεί δυναμικό περιεχόμενο προγραμματιστικά. Πρόκειται για μία ιεραρχική δομή από Android συστατικά όπως "buttons", "textfields" και ομάδες Όψεων (View Groups) όπως "ConstraintLayout", "LinearLayout" κ.α.. Η διεπαφή χρήστη θα εμφανιστεί σε κάποιο συστατικό Android (Android component) όταν αυτό δηλωθεί στην έναρξη του κύκλου ζωής του. Προγραμματιστικά, το παραπάνω συμβαίνει κατά την "onCreate" με την μέθοδο "setContentView" όπου και ορίζεται το αρχείο που θα αναπαρασταθεί. Για την μορφοποίηση των διεπαφών χρήστη αλλά και των συστατικών τους χρησιμοποιούνται διάφορα γνωρίσματα (attributes) όπως "margin", "background-color", "text-size" και άλλα.

3.2.3 Resource Manager

Ο Resource Manager διαχειρίζεται τους πόρους της εφαρμογής (drawables, layouts κ.α.) και δίνει την δυνατότητα της επεξεργασίας τους ανάλογα με τις ανάγκες της εφαρμογής (π.χ. νυχτερινή λειτουργία) [23]. Οι πηγές, ομαδοποιούνται σε ένα φάκελο (με όνομα "res") με βάση τον τύπο τους. Το αναγνωριστικό του κάθε πόρου για την χρήση του είναι η ταυτότητα (Identification) του. Η κλάση R έχει πρόσβαση σε όλους τους πόρους. Για την αναφορά ενός πόρου στο κώδικα χρησιμοποιείται η κλάση R, το όνομα κατηγορίας, και το όνομα του επιθυμητού πόρου (π.χ. R.drawable.img1), ενώ η αναφορά του σε XML γίνεται πάλι με τον ίδιο τρόπο αλλά διαφορετικό συντακτικό (π.χ. @color/red11).

3.2.4 Activity Manager

Η δήλωση όλων των Activities που χρησιμοποιούνται στην εφαρμογή γίνεται στο XML αρχείο AndroidManifest [24]. Το Activity είναι μία κλάση σε μία Android εφαρμογή που έχει τον δικό της κύκλο ζωής. Με την εκκίνηση μίας εφαρμογής ξεκινάει ο κύκλος ζωής του Activity που έχει ορισθεί ως εκκινήτης (launcher Activity).

Με την δημιουργία ενός Activity εμφανίζεται και η διεπαφή χρήστη (onCreate). Ακολουθεί η εκτέλεση της onStart μεθόδου. Η onStart εκτελείται και μετά την onRestart. Αμέσως μετά εκτελείται η onResume, όπου γίνεται η αλληλεπίδραση του χρήστη με την διεπαφή χρήστη. Η onPause (συμβαίνει μετά την onResume) ενεργοποιείται όταν χαθεί το εστίαση του Activity. Για παράδειγμα στην περίπτωση που ο χρήστης πλοηγηθεί με το κουμπί πλοήγησης προς τα πίσω. Η onStop μέθοδος εκτελείται όταν καταστρέφεται ένα Activity ή όταν εκτελεστεί η onResume κάποιου άλλου Activity και εξαφανίζεται την διεπαφή χρήστη του Activity. Μετά την onStop ακολουθεί είτε η onRestart που επαναφέρει το Activity, είτε η onDestroy όταν το Activity τερματιστεί.

3.3 Αρχείο AndroidManifest

Είναι ένα αρχείο τύπου XML που δημιουργείται σε κάθε νέο Android πρόγραμμα [25]. Περιέχει πληροφορίες όπως τα Activities της εφαρμογής και ίσως κάποιες οδηγίες για το πώς χρησιμοποιούνται, άδειες που δίνονται στην εφαρμογή, δέκτες (broadcast receivers), εικονίδιο εκκίνησης κ.α.

3.4 Αρχείο εξαρτήσεων

Πρόκειται για το αρχείο που δηλώνονται οι εξαρτήσεις (π.χ. βιβλιοθήκες) ή ρυθμίσεις όπως η Software Development Kit (SDK) έκδοση που χρησιμοποιείται στην εφαρμογή [26]. Το αρχείο αυτό ονομάζεται build.gradle. Μπορεί να είναι είτε σε γονικό φάκελο αν οι εξαρτήσεις αφορούν όλο το πρόγραμμα ή να έχει πιο εμφωλευμένη θέση στην ιεραρχία των αρχείων του προγράμματος.

3.5 Firebase

Η Firebase είναι μία υπηρεσία της Google είχε αρχικά σχεδιαστεί να λειτουργεί απλά ως βάση δεδομένων πραγματικού χρόνου, αλλά πλέον έχει πολλές ακόμα λειτουργικότητες [27]. Χρησιμοποιείται στο διαδίκτυο, σε κινητά και γενικότερα φορητές συσκευές. Χρησιμοποιεί το μοντέλο Backend-as-a-Service (BaaS) που σημαίνει πως έχει κάποιες λειτουργικότητες που με άλλη τεχνολογία θα έπρεπε να βρίσκονται στον εξυπηρετητή. Για πιο απλές εφαρμογές το BaaS αναπληρώνει τον εξυπηρετητή εξ ολοκλήρου. Στην συνέχεια αναλύονται κάποιες σημαντικές υπηρεσίες της Firebase.

3.5.1 Authentication

Είναι μία υπηρεσία της Firebase που προσφέρει μία ολοκληρωμένη υπηρεσία αυθεντικοποίησης χρήστη [28]. Χρησιμοποιεί πρωτόκολλα αυθεντικοποίησης όπως OAuth 2.0 και OpenID Connect.. Η ταυτοποίηση του χρήστη γίνεται με διαπιστευτήρια (credentials) όπως ηλεκτρονική ταχυδρομική διεύθυνση, κωδικός πρόσβασης και γραπτό μήνυμα. Καλύπτει επαναφορά κωδικού μέσω ηλεκτρονικής ταχυδρομικής διεύθυνσης. Επίσης προσφέρει δυνατότητα ταυτοποίησης μέσω Google, Facebook, Twitter, και GitHub λογαριασμών. Με την αυθεντικοποίηση του χρήστη παρέχεται στην εφαρμογή πρόσβαση στις υπόλοιπες υπηρεσίες της Firebase. Για την κάλυψη αναγκών εφαρμογών επιχειρήσεων υπάρχει η δυνατότητα αναβάθμισης σε Firebase Authentication with Identity Platform που υποστηρίζει περισσότερους παράγοντες ταυτοποίησης .

3.5.2 Real-time database

Είναι μία βάση δεδομένων τύπου NoSQL όπου τα δεδομένα αποθηκεύονται σε Javascript Object Notation (JSON) μορφή [29]. Ένα χαρακτηριστικό της Real-time βάσης δεδομένων είναι ότι ακόμα και χωρίς σύνδεση στο διαδίκτυο τα δεδομένα δεν χάνονται, αλλά αποθηκεύονται τοπικά σε μια κρυφή μνήμη και συγχρονίζονται με την βάση όταν η σύνδεση επανέλθει. Το όνομα της οφείλεται στο γεγονός ότι οι αλληλεπιδράσεις των πελατών με την βάση είναι σχεδόν πραγματικού χρόνου.

3.5.3 Firestore

Είναι μία NoSQL βάση δεδομένων [30]. Υποστηρίζει αποθήκευση δεδομένων σε documents. Τα documents σχηματίζουν collections. Από κάθε collection μπορεί να δημιουργηθεί άλλο ένα ή περισσότερα collections. Τα documents έχουν fields. Το κάθε field μπορεί να είναι τύπου string, array, number κ.α.. Αποθηκεύει στην κρυφή μνήμη τα δεδομένα ακόμα και όταν δεν υπάρχει σύνδεση στο διαδίκτυο και συγχρονίζει τα δεδομένα με την επανέναρξη της σύνδεσης. Κατά την ανάκτηση δεδομένων ανακτά τα documents που ταιριάζουν στις παραμέτρους του αιτήματος.

3.5.4 Cloud Storage

Σε κάποιες εφαρμογές είναι αναγκαία η αποθήκευση πολυμεσικών αρχείων (π.χ. φωτογραφίες, βίντεο) την ανάγκη αυτή καλύπτει η Cloud Storage [31]. Το uploading και downloading με την Cloud Storage είναι πολύ ευέλικτο και δεν επηρεάζει την διάδραση του χρήστη αρνητικά καθώς με την ανάκτηση μίας χαμένης σύνδεσης στο διαδίκτυο η ενέργεια θα συνεχιστεί από το σημείο που έμεινε. Η μεταφορά δεδομένων μπορεί να γίνει ακόμα και με χαμηλές ταχύτητες σύνδεσης.

3.5.5 Firebase Security Rules

Σε μία εφαρμογή σημαντικός παράγοντας είναι η ασφάλεια δεδομένων του χρήστη [32]. Έτσι, ο διαχειριστής της βάσης δεδομένων ορίζει κάποιους κανόνες σε μορφή JSON (Realtime Database) ή κάποια γλώσσα που ορίζουν το περιεχόμενο πρόσβασης των πελατών.

3.6 Unity

Το Unity είναι μία πλατφόρμα ανάπτυξης τρισδιάστατου περιεχομένου και είναι πολύ διαδεδομένο στον τομέα της ανάπτυξης ηλεκτρονικών παιχνιδιών [33]. Στο περιβάλλον αυτό μπορούν να δημιουργηθούν επίσης δισδιάστατα παιχνίδια, επαυξημένης ή εικονικής πραγματικότητας παιχνίδια και παιχνίδια κονσόλας. Η συμπεριφορά των συστατικών που χρησιμοποιούνται δημιουργείται με σενάρια (scripts) σε γλώσσα C#. Η πλατφόρμα αυτή, προσφέρει πολλά εργαλεία ανάπτυξης και έτσι

μπορεί να δημιουργηθούν ρεαλιστικές σκηνές στο παιχνίδι. Ένας σημαντικός παράγοντας για αληθοφανή αποτελέσματα είναι ο σωστός φωτισμός (Lights GameObject) [34]. Μπορούν να επεξεργαστούν διάφορες ιδιότητες του (χρώμα, σκιά κ.α.). Η οπτική γωνία του χρήστη ορίζεται από το αντικείμενο Camera [33]. Είναι δυνατόν να υπάρχουν παραπάνω από μία κάμερες σε μία σκηνή (Scene) να είναι στατικές, κινούμενες ή να ορίζονται από συντεταγμένες.

3.6.1 Asset workflow

Κατά την δημιουργία μίας εφαρμογής για λόγους ευκολίας, συντηρησιμότητας και επεκτασιμότητας ένας προγραμματιστής προσπαθεί να επαναχρησιμοποιεί κώδικα [36]. Αυτό ακριβώς συμβαίνει και με τα Assets σε ένα Unity πρόγραμμα αφού μπορούν να χρησιμοποιηθούν είτε έτοιμα Assets από κάποιο αποθετήριο είτε να δημιουργηθούν σε κάποιο εξωτερικό εργαλείο από τον προγραμματιστή (και να επαναχρησιμοποιηθούν). Τα Assets μπορεί να αναπαριστούν τρισδιάστατα μοντέλα, χρώματα, μουσική και άλλους τύπους .

Το Asset workflow αναφέρεται στην πορεία των Assets από την εισαγωγή τους μέχρι την αναπαράσταση του αποτελέσματος στον τελικό χρήστη. Μετά την από κάποια εξωτερική πηγή ακολουθεί το Creation. Κατά το Creation τα Assets παίρνουν την μορφή των Game Objects, τοποθετούνται σε Scenes και προγραμματίζονται με σενάρια. Το Building δημιουργεί δυαδικά αρχεία του προγράμματος που χρησιμοποιούνται για την εκτέλεση της εφαρμογής. Τέλος, η εφαρμογή μπορεί να γίνει διαθέσιμη σε κάποιο αποθετήριο ή και να εμπλουτιστεί με νέα χαρακτηριστικά.

3.6.2 Scenes

Πρόκειται για την διεπαφή χρήστη που αναπαριστά το περιβάλλον του παιχνιδιού [37]. Ένα Unity πρόγραμμα μπορεί να έχει ένα ή περισσότερα Scenes. Ένα παράδειγμα χρησιμοποίησης διαφορετικών Scenes είναι τα διαφορετικά επίπεδα δυσκολίας ενός παιχνιδιού. Σε κάθε Scene μπορεί να υπάρχει μία ιεραρχική δομή από Assets.

3.6.3 Game Objects

Αναπαριστούν διάφορα αντικείμενα μέσα στα Scenes ενός Unity προγράμματος (φωτισμός, τρισδιάστατες φιγούρες κ.α.) [38]. Για την επεξεργασία τους χρειάζεται η προσθήκη της ανάλογης ιδιότητας "component" στο αντικείμενο. Η πλατφόρμα Unity προσφέρει ποικιλία στις δυνατότητες μορφοποίησης των Game Objects (τοποθέτηση, μέγεθος κ.α.). Προγραμματίζονται με σενάρια.

3.6.4 Prefabs

Τα Prefabs είναι Game Objects που αποθηκεύονται σε μία συγκεκριμένη κατάσταση και συγκεκριμένες ιδιότητες με σκοπό την επαναχρησιμοποίησή τους [39]. Τα Prefabs συντηρούνται εύκολα καθώς η επιθυμητή αλλαγή γίνεται στο Prefab και επηρεάζει σε όλα τα "αντίγραφα". Μπορούν επίσης να χρησιμοποιηθούν στην αρχή της υλοποίησης ενός παιχνιδιού για να αναπαραστήσουν αντικείμενα που πρόκειται ολοκληρωθούν στην συνέχεια.

3.7 Δημιουργία εικονιδίων και σχεδιαγραμμάτων

Το λογότυπο και οι εικόνες για τα παιχνίδια δημιουργήθηκαν στην εφαρμογή GNU Image Manipulation Program (GIMP). Το GIMP είναι μία εφαρμογή επεξεργασίας εικόνων. Για τις ανάγκες της ΠΕ δημιουργήθηκαν απλές εικόνες με γράμματα ή αριθμούς και σε κάποιες περιπτώσεις με γραφικά στο φόντο.

Για την παραγωγή του διαγράμματος 1.1 έγινε χρήση της διαδικτυακής πλατφόρμας draw.io. Πρόκειται για μία εύχρηστη πλατφόρμα που ο χρήστης μπορεί να χρησιμοποιήσει πλήθος από έτοιμα εικονίδια, να τα μορφοποιήσει, να προσθέσει κείμενο και βελάκια ώστε να δημιουργήσει το επιθυμητό σχεδιάγραμμα.

Οι αριθμοί του παιχνιδιού "αριθμοί" που θα αναλυθεί παρακάτω, αλλά και το κουμπί περιήγησης στην προηγούμενη οθόνη που χρησιμοποιήθηκε στην Unity εφαρμογή είναι εικονίδια από το Flaticon.com

3.8 Επίλογος

Σε αυτό το κεφάλαιο έγινε λόγος για τις τεχνολογίες Android, Firebase και Unity που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής και τα βασικά χαρακτηριστικά τους. Με το πέρας του χρόνου οι τεχνολογίες αυτές εξελίσσονται, δίνουν στον προγραμματιστή περισσότερες δυνατότητες και τα αποτελέσματα αφήνουν τους χρήστες πιο ευχαριστημένους. Τέλος, αναφέρθηκαν οι εφαρμογές που χρησιμοποιήθηκαν για την δημιουργία εικονιδίων και διαγραμμάτων.

Κεφάλαιο 4ο: Σχεδίαση και υλοποίηση της εφαρμογής

4.1 Εισαγωγή

Στο κεφάλαιο αυτό περιγράφεται αναλυτικά η σχεδίαση και η υλοποίηση του πρακτικού μέρους της ΠΕ. Σκοπός του λογισμικού που αναπτύχθηκε είναι η πρόληψη της νόσου της άνοιας. Όπως αναφέρθηκε στο κεφάλαιο 2, η καθημερινή εξάσκηση του μυαλού αλλά και η σωματική άσκηση είναι δύο τρόποι που μειώνουν τις πιθανότητες νόσησης. Η υλοποίηση της ΠΕ πραγματοποιήθηκε σε δύο μέρη. Το ένα μέρος είναι η Android εφαρμογή που υποστηρίζει τις παρακάτω λειτουργίες.

- Καταμέτρηση βημάτων χρησιμοποιώντας αισθητήρες του συστήματος.
- Διαχείριση της φαρμακευτικής αγωγής του χρήστη (προσθήκη και αφαίρεση).
- Προβολή προφίλ χρήστη (καθημερινή και συνολική καταμέτρηση βημάτων, καθημερινοί και συνολικοί πόντοι παιχνιδιών).
- Καθημερινή υπενθύμιση για χρήση της εφαρμογής.
- Έναρξη της Unity εφαρμογής (αναλύεται παρακάτω)

Υλοποιήθηκαν επίσης διάφορες ρυθμίσεις της εφαρμογής που θα αναφερθούν στην συνέχεια. Προγραμματίστηκε σε γλώσσα Java, στο Android Studio. Η υλοποίηση των διεπαφών χρήστη έγινε με την χρήση XMLs αρχείων.

Το άλλο μέρος της υλοποίησης αφορά παιχνίδια που έχουν ως σκοπό την εξάσκηση της μνήμης του χρήστη και την συγκέντρωση του. Τα παιχνίδια αυτά αναπτύχθηκαν σε περιβάλλον Unity. Η διεπαφή χρήστη των παιχνιδιών έγινε στον συντάκτη Unity (τοποθέτηση, κλιμάκωση, συστατικά κ.α.). Οι συμπεριφορές των αντικειμένων που αποτελούν τα παιχνίδια ορίστηκαν σε σενάρια σε γλώσσα C#.

Η αρχική πρόθεση ήταν η Unity εφαρμογή να ενσωματωθεί μέσα στην Android και να χρησιμοποιείται από εκεί ως βιβλιοθήκη. Κάτι τέτοιο δεν έγινε, καθώς μετά την επίλυση αρκετών προβλημάτων κατά την ενσωμάτωση η εφαρμογή Android δεν ανταποκρινόταν χωρίς να εμφανίζεται κάποιο πρόβλημα κατά το χτίσιμο (build) ή το τρέξιμο (run). Η λύση που επιλέχθηκε σε αυτό το πρόβλημα ήταν να υπάρχουν δύο εγκατεστημένες εφαρμογές στην συσκευή όπου η Android εφαρμογή εκκινεί την Unity.

4.2 Εμπειρία χρήστη (User Experience)

Κατά τον σχεδιασμό της εφαρμογής λήφθηκε υπόψιν ο τύπος των χρηστών που πρόκειται να χρησιμοποιήσουν την εφαρμογή αλλά και τα προβλήματα που θα μπορούσαν να αντιμετωπίσουν. Η ευκολία χρήσης μίας εφαρμογής αυξάνει και την επιθυμία ενός χρήστη για αυτή. Για τον λόγο αυτό δημιουργήθηκε μία απλή διεπαφή χρήστη, ξεκάθαρη, με μεγάλα και καθαρά γράμματα. Ένα καλό παράδειγμα είναι το κύριο μενού όπου η κάθε επιλογή είναι μακριά η μία από την άλλη.

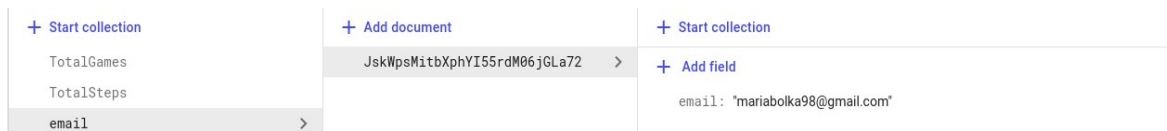
Τα κύρια χρώματα που χρησιμοποιήθηκαν είναι το άσπρο και το πορτοκαλί. Πρόκειται για χρώματα που στον συνδυασμό τους δημιουργούν αντίθεση και κατ' επέκταση ευκολία στην ανάγνωση.

4.3 Αποθήκευση των δεδομένων του χρήστη

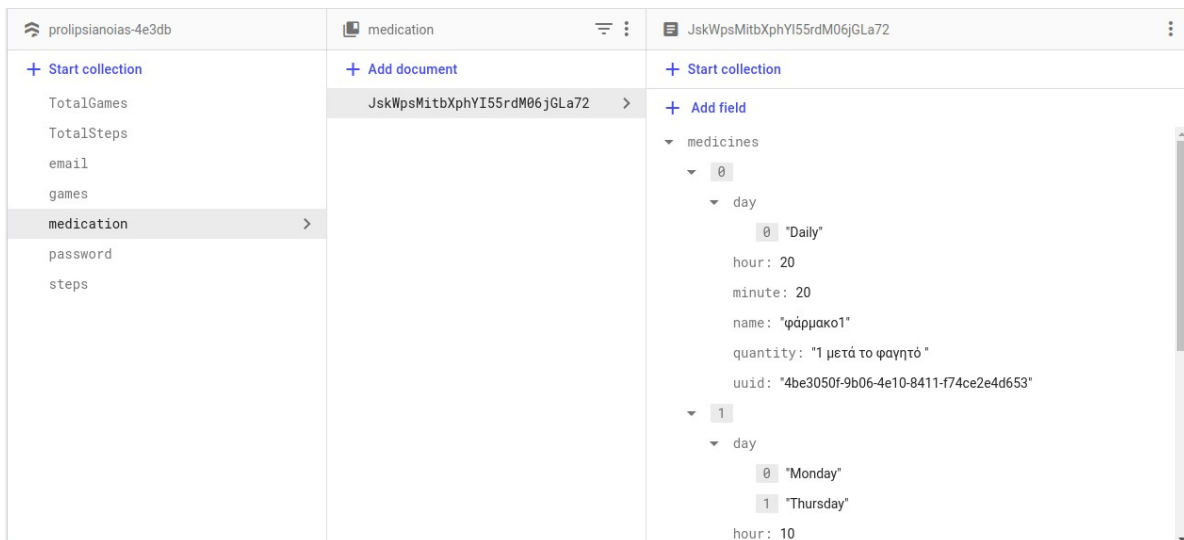
Σημαντικό ρόλο στην εφαρμογή έχει η βάση δεδομένων Firestore. Χρησιμοποιήθηκε για την αποθήκευση και ανάκτηση των δεδομένων του χρήστη όπως για παράδειγμα την προβολή των καθημερινών του βημάτων. Δημιουργήθηκε μία συγκεκριμένη δομή που χρησιμοποιήθηκε για την ευκολία των συναλλαγών με βάση δεδομένων. Η δομή αυτή περιέχει όλες τις μεθόδους που κάνουν κλήσεις στην βάση και επαναχρησιμοποιούνται από διάφορα σημεία στο πρόγραμμα. Προσφέρει ευελιξία στον προγραμματιστή, καθαρότερο και πιο εύκολα συντηρήσιμο κώδικα.

Η κάθε συλλογή (collection) στην Firestore αναπαριστά μία οντότητα. Ένα παράδειγμα είναι η συλλογή που αποθηκεύονται βήματα (steps). Η κάθε συλλογή έχει έγγραφα (documents). Το κάθε έγγραφο αναπαριστά τον κάθε χρήστη και έχει ως τιμή το καθολικά μοναδικό χαρακτηριστικό (Universally Unique Identifier) του. Κάθε έγγραφο αντιστοιχίζεται σε μηδέν ή περισσότερα πεδία (fields) με τιμές. Για παράδειγμα το πεδίο "email" με τιμή "mariabolka98@gmail.com". Το σκεπτικό είναι πως για κάθε συναλλαγή με την βάση δεδομένων, γίνεται η αναφορά της συλλογής που χρειάζεται, του καθολικά μοναδικού χαρακτηριστικού του συνδεδεμένου χρήστη (logged in user) ως έγγραφο και των επιθυμητών πεδίων. Έτσι τα δεδομένα ανάμεσα στους λογαριασμούς χρηστών παραμένουν απομονωμένα και ασφαλή κατά την διάρκεια της ανάπτυξης λογισμικού.

Στα σχήματα 4.2 και 4.3 υπάρχουν δύο στιγμιότυπα από δύο συλλογές στην Firestore για έναν λογαριασμό, όπου διακρίνεται ότι εξηγήθηκε παραπάνω.



Σχήμα 4.2: Στιγμιότυπο της συλλογής email στην Firestore



Σχήμα 4.3: Στιγμιότυπο της συλλογής medication στην Firestore

4.4 Σχεδιασμός και υλοποίηση Android εφαρμογής

Στις επόμενες ενότητες περιγράφονται αναλυτικά οι λειτουργίες που αναπτύχθηκαν στην Android εφαρμογή. Αναλύεται η αλληλεπίδραση του χρήστη με το σύστημα, οι έλεγχοι και οι διαδικασίες που

εκτελούνται από το πρόγραμμα κατά την αλληλεπίδραση και οι συναλλαγές με την βάση δεδομένων για τα δεδομένα του χρήστη.

4.4.1 Εγκατάσταση Firebase

Για να επιτευχθεί η επικοινωνία της Firebase με το πρόγραμμα του Android, πρέπει να γίνει πρώτα η μεταξύ τους σύνδεση, όπως περιγράφεται στην συνέχεια.

Αφού έχει δημιουργηθεί το Firebase έργο (project), στην κονσόλα διαχείρισης, στην καρτέλα "Project Overview" επιλέχθηκε το "add app". Εκεί προστέθηκε το όνομα πακέτου (package name) της εφαρμογής. Έτσι, δημιουργήθηκε το google-services αρχείο (περιέχει πληροφορίες για την αναγνώριση των πελάτη της Firebase) και στην συνέχεια έγινε η εισαγωγή του στο φάκελο app της εφαρμογής. Επίσης στο αρχείο build.gradle (στον γονικό φάκελο) έγινε προσθήκη εξάρτησης (dependency) για τις υπηρεσίες Google. Ακόμη, στο build.gradle (στο επίπεδο του φακέλου app) έγινε σύνδεση με τις υπηρεσίες Google. Για την χρήση των υπηρεσιών της Firebase είναι αναγκαίο να δοθεί δικαίωμα χρήσης "Internet" στο αρχείο AndroidManifest.

4.4.2 Σύνδεση και εγγραφή χρήστη

Για να χρησιμοποιηθεί η εφαρμογή από τον χρήστη πρέπει πρώτα από όλα να δημιουργηθεί λογαριασμός, ώστε τα δεδομένα να αποθηκεύονται για τον λογαριασμό αυτό. Στην συνέχεια ο χρήστης μπορεί να συνδέεται στον λογαριασμό κάθε φορά που θέλει να χρησιμοποιήσει την εφαρμογή. Για τις λειτουργίες αυτές χρησιμοποιείται η υπηρεσία Authentication της Firebase. Η σύνδεση και η εγγραφή φιλοξενούνται σε δύο ξεχωριστά αλλά παρόμοια μεταξύ τους Activities.

Κατά την εγγραφή χρειάζεται απλώς από τον χρήστη να εισάγει την διεύθυνση ηλεκτρονικού του ταχυδρομείου του και τον επιθυμητό κωδικό σύνδεσης δύο φορές για αποφυγή τυχόν λαθών κατά την πληκτρολόγηση του. Η οθόνη που εμφανίζεται στον χρήστη φαίνεται στο σχήμα 4.4. Με την συμπλήρωση των στοιχείων αυτών το σύστημα απλώς περιμένει τον χρήστη να πατήσει το κουμπί (button) "δημιουργία λογαριασμού" ώστε να κάνει τις κατάλληλες ενέργειες για την δημιουργία λογαριασμού.

PreventApp

App

ΔΗΜΙΟΥΡΓΙΑ ΛΟΓΑΡΙΑΣΜΟΥ

εισαγωγή e-mail

κωδικός πρόσβασης

επιβεβαίωση κωδικού πρόσβασης

ΔΗΜΙΟΥΡΓΙΑ ΛΟΓΑΡΙΑΣΜΟΥ

Έχετε ήδη λογαριασμό;

Σχήμα 4.4: Δημιουργία λογαριασμού

Με την εκκίνηση του Activity για την εγγραφή, καλείται η `onCreate` μέθοδος, όπου γίνεται αναφορά των συστατικών τις διεπαφής που αναφέρθηκαν παραπάνω σε μεταβλητές. Έτσι, έγινε προσθήκη ενός ακροατή (`listener`) στο γεγονός (`event`) του κλικ του κουμπιού δημιουργίας λογαριασμού. Όταν ο ακροατής ενεργοποιηθεί γίνονται κάποιοι έλεγχοι. Ένας έλεγχος είναι αν είναι κενά τα πεδία της διεύθυνσης ηλεκτρονικού ταχυδρομείου και ο κωδικός πρόσβασης. Επίσης γίνεται έλεγχος αν ο κωδικός και ο κωδικός επαλήθευσης δεν είναι ίδιοι μεταξύ τους και αν έχουν λιγότερους από έξι χαρακτήρες. Αν έστω και ένα από τα παραπάνω είναι αληθή τότε εμφανίζεται το αντίστοιχο μήνυμα στον χρήστη. Αν όλες οι εισαγωγές πληρούν τις προϋποθέσεις του συστήματος, τότε γίνεται κλήση στην υπηρεσία `Authentication` για την δημιουργία του λογαριασμού. Στο σχήμα 4.5 φαίνεται ένα στιγμιότυπο της υπηρεσίας `Authentication`, με έναν εγγεγραμμένο χρήστη.

Identifier	Providers	Created ↓	Signed In	User UID
mariabolka98@gmail.com		Nov 20, 2022	Nov 28, 2022	JskWpsMitbXphYI55rdM06jGLa72

Rows per page: 50 1 - 1 of 1

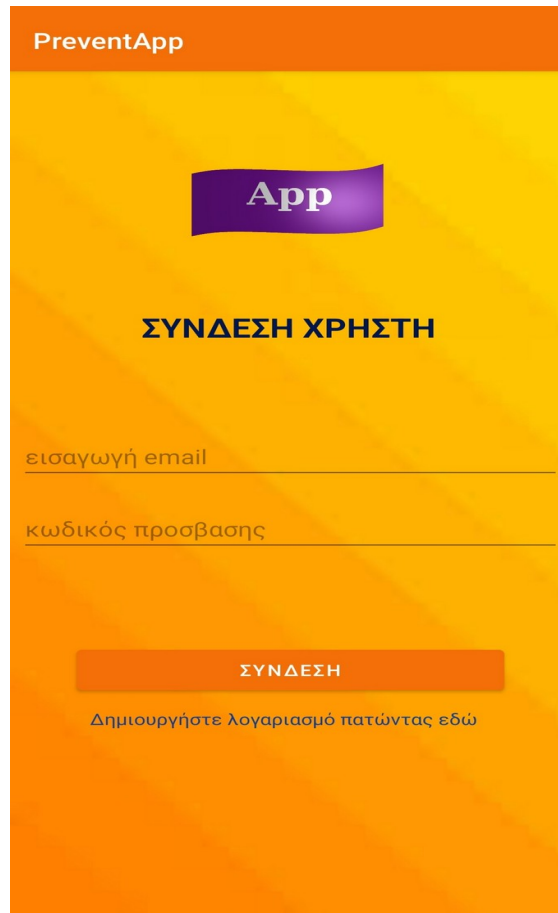
Σχήμα 4.5: Στιγμιότυπο υπηρεσίας Authentication

Το FirebaseAuth είναι αντικείμενο για την Java για την υπηρεσία Authentication της Firebase που προσφέρει ένα σύνολο έτοιμων μεθόδων για την εύκολη διαχείριση της δημιουργίας λογαριασμών και ταυτοποίησης χρήστη. Έτσι, για την δημιουργία ενός λογαριασμού πρέπει πρώτα να ληφθεί η κατάσταση (instance) της βάσης την τρέχουσα στιγμή με βάση αυτή την κατάσταση γίνεται η κλήση της μεθόδου createUserWithEmailAndPassword με την διεύθυνση ηλεκτρονικού ταχυδρομείου και τον κωδικό πρόσβασης που έδωσε ο χρήστης και ενεργοποιείται μία αλυσίδα μεθόδων στην υπηρεσία. Στην διαδικασία αυτή προστέθηκε ένας ακροατής για την διακρίβωση του αν είναι επιτυχημένη ή αποτυχημένη. Το είδος του αποτελέσματος εμφανίζει και το κατάλληλο μήνυμα στον χρήστη. Κατά τον χρόνο επεξεργασίας του αιτήματος δημιουργίας λογαριασμού μέχρι την απάντηση πορείας του αιτήματος, ο χρήστης βλέπει μία μπάρα προόδου.

Με την επιτυχή δημιουργία του λογαριασμού, δημιουργούνται οι συλλογές της Firestore με τα αντίστοιχα έγγραφα (με τιμή το καθολικά μοναδικό χαρακτηριστικό του λογαριασμού) και πεδία που θα χρησιμοποιηθούν για την μελλοντική αποθήκευση των δεδομένων του χρήστη. Ουσιαστικά συμβαίνει μία αρχικοποίηση στην βάση δεδομένων Firestore για την υποδοχή των δεδομένων του νέου χρήστη.

Η διαδικασία και η διεπαφή χρήστη κατά την σύνδεση χρήστη (σχήμα 4.6) είναι αρκετά παρόμοια με την δημιουργία νέου χρήστη. Η μέθοδος της κλάσης FirebaseAuth που κάνει ταυτοποίηση του χρήστη είναι η signInWithEmailAndPassword. Τα διαπιστευτήρια είναι η διεύθυνση ηλεκτρονικού ταχυδρομείου και ο κωδικός πρόσβασης.

Η λειτουργία αποσύνδεσης χρήστη από την εφαρμογή έγινε με ακροατή σε ένα κουμπί με το κείμενο "έξοδος" που υπάρχει στο κύριο μενού. Όταν ο ενεργοποιείται ο ακροατής καλείται η signOut μέθοδος της FirebaseAuth που ενημερώνει τον σύστημα ότι ο χρήστης δεν είναι πια συνδεδεμένος και τερματίζεται το Activity του κύριου μενού.



Σχήμα 4.6: Οθόνη σύνδεσης χρήστη

4.4.3 Κύριο μενού

Το κύριο μενού αποτελείται από διάφορα κουμπιά. Οι διαθέσιμες επιλογές είναι "προφίλ", "παιχνίδια", "φάρμακα", "βήματα", "ρυθμίσεις" και "έξοδος" (σχήμα 4.7). Κάθε κουμπί έχει και ένα εικονίδιο σχετικό με την λειτουργία που θα ακολουθήσει αν επιλεγθεί, κάνοντας πιο ευανάγνωστο το μενού, αφού είναι γνωστό πως μία εικόνα αποτυπώνεται πιο εύκολα σε σχέση με μία λέξη.

Ο χρήστης αρκεί να επιλέξει ένα από τα κουμπιά και να περιηγηθεί στην λειτουργία που θέλει. Για τον λόγο αυτό σε κάθε επιλογή του κύριου μενού έχει προστεθεί και ένας ακροατής που εκκινεί το κατάλληλο Activity.

Η ενεργοποίηση της εκκίνησης ενός Activity μπορεί να γίνει με δύο τρόπους. Ο πρώτος τρόπος είναι να δημιουργηθεί μία μέθοδος στο τρέχον Activity που εκκινεί το νέο Activity και στην συνέχεια ορίζεται στο αρχείο μορφοποίησης ότι αυτή η μέθοδος ότι θα ενεργοποιείται με το γεγονός του κλικ. Ο άλλος τρόπος είναι μέσα από το Activity να βρεθεί το συστατικό με βάση την ταυτότητα (id) του (findViewById μέθοδος) και εκεί να προστεθεί ο ακροατής που εκκινεί το άλλο Activity.

Σε κάθε περίπτωση για να ξεκινήσει ένα νέο Activity, πρέπει να δημιουργηθεί ένα νέο Intent με την κλάση που αντιπροσωπεύει το Activity που πρόκειται να ξεκινήσει και στην συνέχεια να κληθεί η μέθοδος startNewActivity με το Intent που περιέχει την κλάση.

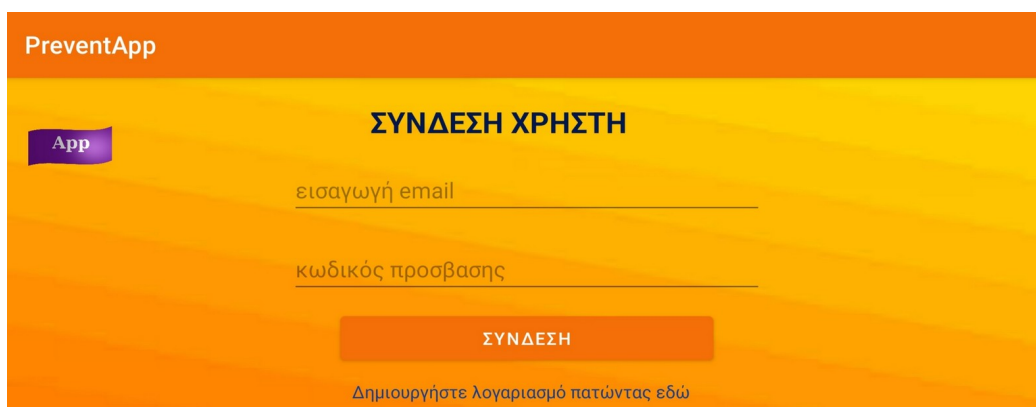


Σχήμα 4.7: Βασικό μενού εφαρμογής

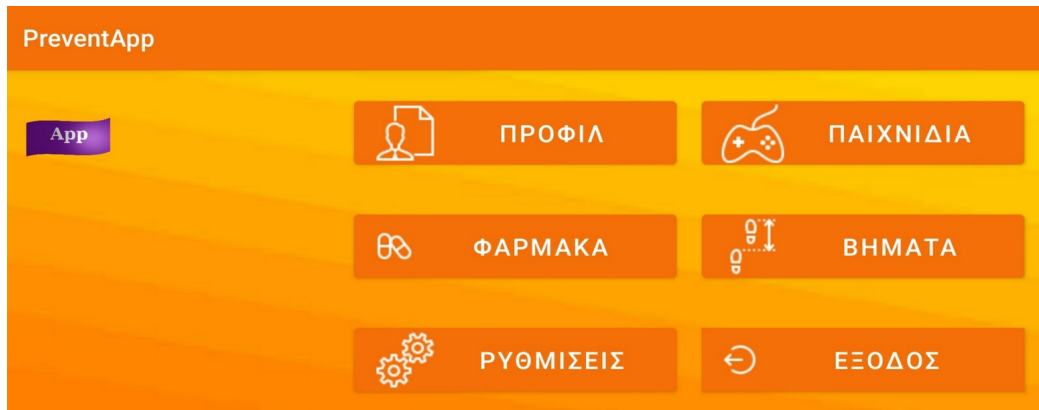
4.4.4 Οριζόντια προβολή διεπαφής χρήστη

Τέτοιου είδους εφαρμογές ενδείκνυται να χρησιμοποιούνται σε λειτουργία πορτραίτου (portrait mode) καθώς έτσι είναι πιο εύχρηστες. Η Android εφαρμογή όμως έχει σχεδιαστεί και σε οριζόντια λειτουργία (landscape mode). Έτσι ο χρήστης να μπορεί να την χρησιμοποιήσει ακόμα και με αυτό τον τρόπο, αλλά και να μην βρεθεί σε σύγχυση αν η συσκευή περιστραφεί κατά λάθος.

Ο σχεδιασμός κάθε οριζόντιας διεπαφής χρήστη έγινε από την αρχή με σκέψη πάντα την ευχρηστία της εφαρμογής. Στα σχήματα 4.8 και 4.9 φαίνονται δύο παραδείγματα οριζόντιων διεπαφών.



Σχήμα 4.8: Σύνδεση χρήστη σε οριζόντια προβολή



Σχήμα 4.9: Βασικό μενού εφαρμογής σε οριζόντια προβολή

4.4.5 Εκκίνηση Unity εφαρμογής

Για την εκκίνηση της Unity εφαρμογής από το πρόγραμμα Android χρειάζεται το όνομα πακέτου της. Στην Android εφαρμογή χρησιμοποιείται ένας ακροατής στο κουμπί του κυρίου μενού που γράφει "παιχνίδια". Εκεί δημιουργείται ένα Intent που περιέχει το όνομα πακέτου της εφαρμογής Unity. Πιο συγκεκριμένα το Intent αυτό περιέχει την μέθοδο `getPackageManager` της κλάσης `ContextWrapper` (μέθοδος που επιστρέφει τις εγκατεστημένες στην συσκευή εφαρμογές) και με την μέθοδο `getLaunchIntentForPackage` με το όνομα πακέτου της Unity εφαρμογής που αναφέρθηκε παραπάνω. Ουσιαστικά το Intent αυτό δηλώνει την πρόθεση για εκκίνηση της Unity εφαρμογής.

Κρίθηκε αναγκαίο στην Unity εφαρμογή να δίνεται το καθολικά μοναδικό χαρακτηριστικό του συνδεδεμένου χρήστη που θα χρησιμοποιηθεί στην συνέχεια για την προσθήκη των βαθμολογιών του συγκεκριμένου χρήστη στην Firestore. Έτσι στο Intent που αναφέρθηκε παραπάνω, με την μέθοδο `putExtra` προστέθηκε σε αντιστοιχία με κάποιο κλειδί (key) το καθολικά μοναδικό χαρακτηριστικό του χρήστη.

Το Intent είναι έτοιμο από θέμα πληροφορίας. Για να εκκινήσει την εφαρμογή Unity εκτελείται με την μέθοδο `startActivity`. Για να γίνει επιτρεπτό από την Android εφαρμογή να εκκινήσει μία άλλη πρέπει να δηλωθεί το όνομα πακέτου της δεύτερης στο αρχείο `AndroidManifest` μέσα σε ετικέτα (tag) "queries".

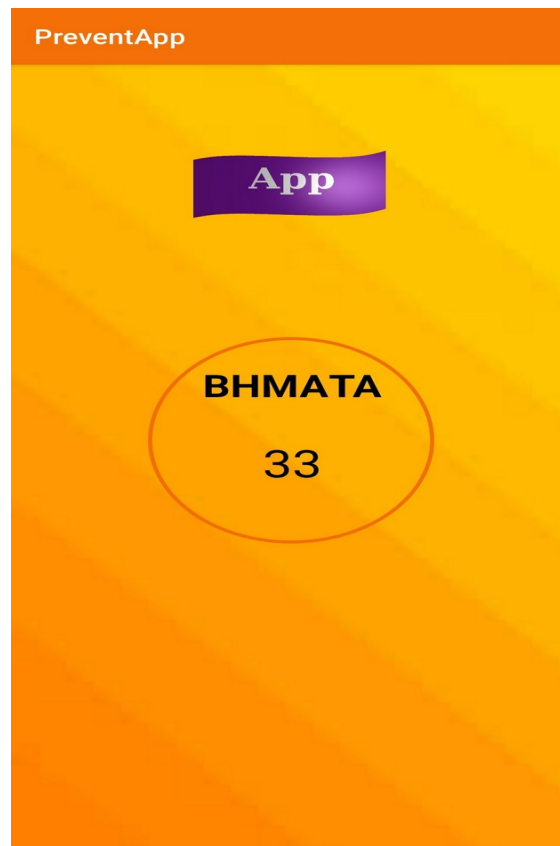
4.4.6 Καταμέτρηση βημάτων

Με την επιλογή του χρήστη "βήματα" από το κύριο μενού, γίνεται κλήση στην Firestore για το πεδίο που έχει ως τιμή τα βήματα με έγγραφο το καθολικά μοναδικό χαρακτηριστικό του συνδεδεμένου χρήστη στην συλλογή που αποθηκεύει τα βήματα που γίνονται καθημερινά (steps). Η τιμή αυτή αφορά τα βήματα που έχουν γίνει ως στιγμής την συγκεκριμένη ημέρα και εμφανίζεται στην διεπαφή χρήστη.

Για την καταγραφή των βημάτων το πρόγραμμα έχει πρόσβαση σε ένα αισθητήρα της συσκευής, το επιταχυνσιόμετρο. Το σύστημα ακούει στα γεγονότα αλλαγής (change events) και κρατά μόνο τις τιμές που αφορούν κίνηση βήματος. Η εκκίνηση του ακροατή στα γεγονότα γίνεται μόνο στην συγκεκριμένη οθόνη. Ο ακροατής απενεργοποιείται όταν ο χρήστης περηγηθεί σε άλλη οθόνη.

Κάθε βήμα προστίθεται με την τιμή των βημάτων από την Firestore που αναφέρθηκε παραπάνω και κάθε φορά γίνεται κλήση στην βάση για ενημέρωση των βημάτων. Το αποτέλεσμα είναι η σχεδόν

πραγματικού χρόνου ενημέρωση της διεπαφής χρήστη. Η οθόνη που βλέπει ο χρήστης φαίνεται στο σχήμα 4.10.



Σχήμα 4.10: Διεπαφή καταμέτρησης βημάτων

4.4.7 Καθημερινή ειδοποίηση για χρήση της εφαρμογής.

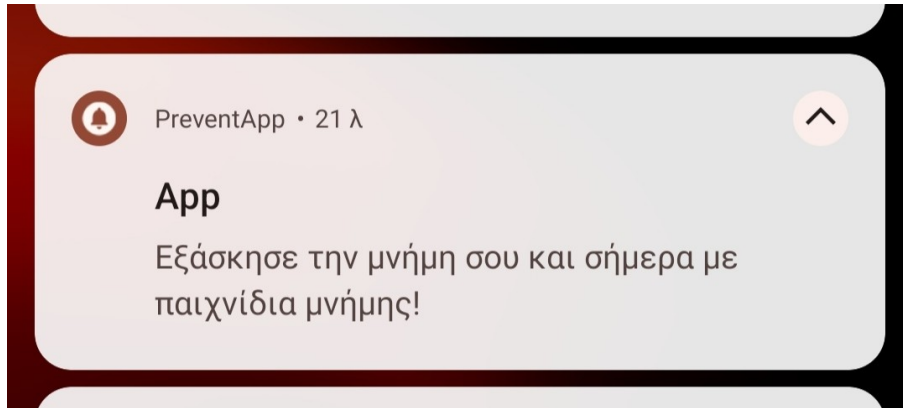
Στα πλαίσια ανάπτυξης της εφαρμογής κρίθηκε ως μία αρκετά βοηθητική λειτουργία η υπενθύμιση του χρήστη καθημερινά σε συγκεκριμένη ώρα να χρησιμοποιεί την εφαρμογή. Η υπενθύμιση γίνεται με Notification που περιέχει το αντίστοιχο μήνυμα.

Το Notification πρέπει να δημιουργείται ακόμα και όταν η εφαρμογή είναι κλειστή. Έτσι χρησιμοποιήθηκε ο AlarmManager που συνδέει την εφαρμογή με το Alarm σύστημα υπηρεσιών της συσκευής και άρα μπορεί να εκτελεί οτιδήποτε του έχει δοθεί ακόμα και όταν η εφαρμογή έχει τερματιστεί.

Αρχικά, δημιουργήθηκε μία κλάση που κληρονομεί την κλάση BroadcastReceiver και καταπατά την onReceive. Η μέθοδος αυτή ενεργοποιείται σε γεγονότα του συστήματος. Για να χρησιμοποιηθεί σωστά η κλάση αυτή από το σύστημα πρέπει να ορισθεί στο αρχείο AndroidManifest με την ετικέτα "receiver". Στην μέθοδο onReceive δημιουργείται το Notification που θα δειχθεί στον χρήστη. Στο Notification προστίθεται τίτλος, κείμενο και εικονίδιο. Στο σημείο αυτό, το Notification είναι ανενεργό στον κώδικα αλλά έτοιμο για χρησιμοποίηση από τον AlarmManager.

Με την περιήγηση του χρήστη στο κύριο μενού της εφαρμογής, προγραμματίζεται η εμφάνιση του Notification σε καθημερινή βάση. Στον κώδικα γίνεται προσθήκη της επιθυμητής ώρας για την ενεργοποίηση της ειδοποίησης. Στην συνέχεια γίνεται έλεγχος αν έχει ξεπεραστεί αυτή η ώρα την δεδομένη χρονική στιγμή. Αν έχει ξεπεραστεί, τότε η ειδοποίηση δημιουργείται για την επόμενη ημέρα, διαφορετικά δημιουργείται για την τρέχουσα ημέρα. Η ενεργοποίηση της ειδοποίησης

καθημερινά γίνεται με την `setRepeating` μέθοδο του `AlarmManager` και θα επαναλαμβάνεται με βάση τις ιδιότητες που προστίθενται (ώρα που επιλέχθηκε, καθημερινά). Στο σχήμα 4.11 φαίνεται ένα `Notification` που έχει εμφανιστεί αυτοματοποιημένα.



Σχήμα 4.11: Στιγμιότυπο `Notification` υπενθύμισης χρήσης

Τα παραπάνω έχουν ως αποτέλεσμα η ειδοποίηση να γίνεται καθημερινά και για όσο υπάρχει η εφαρμογή στην συσκευή, ακόμα και στην περίπτωση που η εφαρμογή έχει ανοίξει μόνο μία φορά. Κάθε φορά που η εφαρμογή ανοίγει γίνεται αντικατάσταση του προηγούμενου `AlarmManager`, άρα πάλι δεν υπάρχει κάποιο πρόβλημα.

Το πρόβλημα με το `AlarmManager` έρχεται κατά την απενεργοποίηση της συσκευής. Στην περίπτωση αυτή οι ειδοποιήσεις που έχουν δημιουργηθεί διαγράφονται και πρέπει η εφαρμογή να εκκινηθεί εκ νέου για να δημιουργηθούν ξανά οι καθημερινές ειδοποιήσεις. Ένα ακόμα πρόβλημα που εντοπίστηκε κατά τον πρακτικό έλεγχο είναι πως κάποιες φορές οι ειδοποιήσεις μπορεί να αργήσουν ανάλογα με τον φόρτο του λειτουργικού συστήματος της συσκευής.

4.4.8 Υπολογισμός ημερησίων/συνολικών βημάτων/πόντων παιχνιδιών

Ο χρήστης μπορεί να δει από το προφίλ του τα βήματα που έκανε και τους πόντους παιχνιδιών που συγκέντρωσε την συγκεκριμένη ημέρα αλλά και τα συνολικά βήματα και πόντους που έχει συγκεντρώσει μέχρι στιγμής.

Αυτό που πραγματικά ενδιαφέρει την εφαρμογή είναι ο χρήστης να βλέπει τα ενημερωμένα δεδομένα εφόσον αλλάξει η μέρα από την τελευταία φορά που χρησιμοποίησε την εφαρμογή και όχι τα δεδομένα όντως να αλλάζουν καθημερινά. Η ενημέρωση με τις νέες τιμές γίνεται κυριολεκτικά μόνο όταν χρειάζεται.

Η δομή στην βάση που περιέχει τις πληροφορίες που αναφέρθηκαν στην παραπάνω παράγραφο έχει ως εξής. Οι συλλογές για την αποθήκευση των ημερησίων βημάτων και πόντων (`steps`, `games`) με έγγραφο το καθολικά μοναδικό χαρακτηριστικό του χρήστη και από ένα πεδίο που συγκρατεί τις αντίστοιχες ημερησίες τιμές. Οι συλλογές που περιέχουν τα συνολικά βήματα και πόντους (`TotalSteps`, `TotalGames`) με έγγραφο το καθολικά μοναδικό χαρακτηριστικό του χρήστη και τα αντίστοιχα πεδία. Η συλλογή με τα συνολικά βήματα έχει ένα ακόμα πεδίο τύπου `Timestamp` που περιέχει την ημερομηνία από την τελευταία ενημέρωση των συνολικών βημάτων και πόντων. Ενδεικτικά στο σχήμα 4.12 φαίνεται η δομή για την συλλογή με τα συνολικά βήματα.



Σχήμα 4.12: Στιγμιότυπο συλλογής συνολικών βημάτων

Κάθε φορά που ο χρήστης περιηγείται στο κύριο μενού το σύστημα ελέγχει αν πρέπει να γίνει μηδενισμός των καθημερινών βημάτων και πόντων και ο υπολογισμός των συνολικών βημάτων και πόντων. Για να γίνει ο έλεγχος αυτός πρέπει να συγκριθεί η ημερομηνία του πεδίου της βάσης που αναφέρεται στην τελευταία χρονική στιγμή ενημέρωσης. Η ημερομηνία συγκρίνεται με την τρέχουσα ημερομηνία. Πιο συγκεκριμένα πρέπει να γίνει έλεγχος για το αν αυτές οι δύο ημερομηνίες έχουν ίδιο έτος, ίδιο μήνα και ίδια μέρα. Αν ισχύουν τα παραπάνω δεν γίνεται κάποια αλλαγή στα δεδομένα, αφού αυτό σημαίνει πως ήδη τα δεδομένα έχουν ενημερωθεί την τρέχουσα ημέρα. Αν δεν ισχύει ότι οι δύο ημερομηνίες αναφέρονται στην ίδια ημέρα τότε γίνεται έλεγχος και για το αν έχει ξεπεραστεί η ώρα 12 τα μεσάνυχτα. Αν η παραπάνω συνθήκη ισχύει τότε ενημερώνονται οι αντίστοιχες συλλογές στην βάση δεδομένων με τρόπο που αναφέρεται παρακάτω.

Για την ενημέρωση των δεδομένων γίνεται μία σειρά από κλήσεις στην Firestore. Οι τιμές των πεδίων με τις ημερήσιες μετρήσεις προστίθενται σε τοπικές μεταβλητές και στην συνέχεια ενημερώνονται τα πεδία που αναφέρονται στα σύνολα με τις μεταβλητές αυτές. Επίσης ενημερώνεται και η ημερομηνία τελευταίας ενημέρωσης. Τα πεδία των συλλογών με τις ημερήσιες μετρήσεις μηδενίζονται.

4.4.9 Ενημέρωση απόδοσης χρήστη μέσω μηνύματος ηλεκτρονικού ταχυδρομείου

Στην εφαρμογή μέσω των ρυθμίσεων είναι δυνατόν να προστεθεί κάποια διεύθυνση ηλεκτρονικού ταχυδρομείου, στην οποία αποστέλλεται ενημέρωση για την απόδοση του χρήστη από την τελευταία μέρα που χρησιμοποιήθηκε η εφαρμογή (βήματα, πόντοι παιχνιδιών). Αυτό ωφελεί την περίπτωση που ο κύριος χρήστης έχει άτομο που παρακολουθεί την εξέλιξη της υγείας του από μακριά. Υπάρχει δυνατότητα η διεύθυνση να αλλαχθεί αλλά και να αφαιρεθεί. Για τις ανάγκες αυτής της λειτουργικότητας δημιουργήθηκε ένας λογαριασμός ηλεκτρονικού ταχυδρομείου που χρησιμοποιείται αποκλειστικά για αυτό τον σκοπό ως αποστολέας.

Για να μπορεί να γίνει χρήση του λογαριασμού από την εφαρμογή Android έπρεπε να γίνουν κάποιες ρυθμίσεις στον λογαριασμό που αποστέλλει τα ηλεκτρονικά μηνύματα. Ουσιαστικά προστέθηκε στον λογαριασμό το δικαίωμα χρήσης της Android εφαρμογής, και παράχθηκε ένας ιδιαίτερος κωδικός που χρησιμοποιείται ως κωδικός πρόσβασης στον κώδικα. Ο κωδικός αυτός δεν μπορεί να χρησιμοποιηθεί για να γίνει σύνδεση στον λογαριασμό από κάποιο άλλο πελάτη. Έτσι, η εφαρμογή και λογαριασμός ηλεκτρονικού ταχυδρομείου παραμένουν χωρίς κενά ασφαλείας.

Για την διαχείριση της διεύθυνσης ηλεκτρονικού ταχυδρομείου δημιουργήθηκε ένα Fragment στο Activity των Ρυθμίσεων. Το Fragment αυτό ανοίγει με την επιλογή του από το μενού των Ρυθμίσεων. Επομένως όταν ο χρήστης επιλέξει το κουμπί "E-mail" από το μενού των ρυθμίσεων, εμφανίζεται στον χρήστη ένα πεδίο για την εισαγωγή της διεύθυνσης και δύο κουμπιά για ενημέρωση/προσθήκη διεύθυνσης και αφαίρεση. Κάθε ένα από τα δύο κουμπιά έχουν και ένα ακροατή. Και τα δύο κουμπιά κάνουν ενημέρωση στην Firestore στην συλλογή που αποθηκεύεται η διεύθυνση ηλεκτρονικού ταχυδρομείου (email). Η διαφορά τους είναι πως με το κουμπί "αφαίρεση" το πεδίο που περιέχει την πληροφορία παίρνει την τιμή "none" ενώ με το κουμπί ενημέρωση γίνονται έλεγχοι για το αν η

διεύθυνση ηλεκτρονικού ταχυδρομείου είναι ορθή και μόνο τότε ενημερώνεται του πεδίου. Η διεπαφή χρήστη φαίνεται στο σχήμα 4.13.



Σχήμα 4.13: Διεπαφή χρήστη εισαγωγής ηλεκτρονικής διεύθυνσης

Το ενημερωτικό ηλεκτρονικό μήνυμα αποστέλλεται αφού ενεργοποιηθεί η ενημέρωση των ημερησίων βημάτων και πόντων παιχνιδιών (αναλύθηκαν στην παράγραφο 4.4.8). Η πληροφορία που πρέπει να αποσταλεί είναι τα καθημερινά βήματα και πόντοι παιχνιδιών. Έτσι γίνονται κλήσεις στην Firestore για την ανάκτηση των κατάλληλων πεδίων. Έτσι σε μεταβλητές αποθηκεύονται οι τιμές των βημάτων και πόντων παιχνιδιών τα οποία χρησιμοποιούνται σαν κείμενο στο ηλεκτρονικό μήνυμα που αποστέλλεται. Επίσης γίνεται κλήση για την διεύθυνση λογαριασμού που καταχώρησε ο χρήστης στο σύστημα μέσω την διεπαφή των ρυθμίσεων.

Για την δημιουργία και αποστολή δημιουργήθηκε μία κλάση που κάνει κληρονομεί την AsyncTask κλάση η οποία τρέχει σε κάποιο νήμα (thread) στο παρασκήνιο (background). Η μέθοδος που περιέχει τον κώδικα αποστολής ηλεκτρονικού μηνύματος είναι η doInBackground που κληρονομείται από την AsyncTask. Τα παραπάνω επιλέγονται σε περιπτώσεις που κάποια λειτουργία είναι χρονοβόρα στο σύστημα, ώστε η διάδραση του χρήστη με το UI να συνεχίζει κανονικά και να μην διακόπτεται (δεν μπλοκάρεται δηλαδή το κύριο νήμα), αυτό συμβαίνει και στην περίπτωση αποστολής ηλεκτρονικού μηνύματος.

Η doInBackground μέθοδος παίρνει ως παραμέτρους ένα αλφαριθμητικό πίνακα στον οποίο περιέχεται οποία είναι η πληροφορία (αριθμός βημάτων, πόντοι παιχνιδιών και διεύθυνση ηλεκτρονικού ταχυδρομείου παραλήπτη).

Στην μέθοδο doInBackground σε δύο αλφαριθμητικές μεταβλητές αποδίδεται η διεύθυνση ηλεκτρονικού ταχυδρομείου αποστολέα και ο ιδιαίτερος κωδικός που αναφέρθηκε παραπάνω.

Για την αποστολή ηλεκτρονικών μηνυμάτων χρησιμοποιήθηκε η JavaMail διεπαφή προγραμματισμού εφαρμογών. Η διεπαφή αυτή προσφέρει έτοιμες λειτουργίες για την αποστολή και λήψη μηνυμάτων ηλεκτρονικού ταχυδρομείου. Σε ένα αντικείμενο τύπου Properties ορίζονται πληροφορίες που χρησιμοποιούνται στην συνέχεια από το Authenticator του JavaMail API. Τέτοιες πληροφορίες είναι το Simple Mail Transfer Protocol (SMTP) πρωτόκολλο, η πόρτα και ο φιλοξενητής (host). Ο Authenticator χρησιμοποιεί και τα διαπιστευτήρια (διεύθυνση ηλεκτρονικού ταχυδρομείου και ο ιδιαίτερος κωδικός πρόσβασης που αναφέρθηκε προηγουμένως) προκειμένου να γίνει η ταυτοποίηση του αποστολέα.

Στην συνέχεια δημιουργείται ένα αντικείμενο τύπου Multipurpose Internet Mail Extensions (MIME) από τη JavaMail διεπαφή όπου ορίζονται ο αποστολέας, ο παραλήπτης, το θέμα, και το κείμενο του ηλεκτρονικού μηνύματος. Με την μέθοδο send του Transport αντικειμένου (αντικείμενο της διεπαφής JavaMail) γίνεται η αποστολή του μηνύματος. Στο σχήμα 4.14 φαίνεται ένα παράδειγμα ηλεκτρονικού μηνύματος που στάλθηκε αυτοματοποιημένα.



Σχήμα 4.14: Παράδειγμα αυτοματοποιημένου ηλεκτρονικού μηνύματος

4.4.10 Προσθήκη φαρμακευτικής αγωγής

Μέσω των ρυθμίσεων μπορεί να προστεθεί η φαρμακευτική αγωγή του χρήστη ώστε να μπορεί βλέπει τις μέρες και τις ώρες που πρέπει να πάρει τα φάρμακα του. Κάθε φάρμακο έχει όνομα, ποσότητα, σχόλια, μέρες και ώρες κατανάλωσης.

Στο αρχείο που αναπαριστά την διεπαφή χρήστη εισαγωγής φαρμακευτικής αγωγής, ο χρήστης προσθέτει το όνομα του φαρμάκου και την ποσότητα με τα σχόλια. Οι κατανάλωση ενός φαρμάκου συνήθως γίνεται σε καθημερινή βάση, αλλά υπάρχει και η πιθανότητα ένα φάρμακο να πρέπει να καταναλώνεται συγκεκριμένες μέρες. Έτσι για την απόφαση αυτή υπάρχει ένα radioGroup με δύο radioButtons με τις τιμές "καθημερινά" και "μέρες". Η προκαθορισμένη τιμή του radioGroup κατά την έναρξη εισαγωγής φαρμακευτικής αγωγής είναι το "καθημερινά". Αν ο χρήστης επιλέξει την τιμή "μέρες", τότε ενεργοποιούνται επτά checkboxes, κάθε ένα έχει ως τιμή μία μέρα της εβδομάδας. Αν ο χρήστης επιλέξει ξανά το "καθημερινά" από το radioGroup, τότε τα checkboxes που αναπαριστούν κάθε μέρα ξεχωριστά απενεργοποιούνται. Έτσι ο χρήστης μπορεί να επιλέξει συγκεκριμένες μέρες (π.χ. Δευτέρα και Τετάρτη). Η ενεργοποίηση και απενεργοποίηση των checkboxes γίνεται αλλάζοντας

την ιδιότητα "enabled" των checkboxes (true και false). Η διεπαφή χρήστη για τις δύο παραπάνω περιπτώσεις φαίνεται στα σχήματα 4.15 και 4.16. Η ώρα κατανάλωσης του φαρμάκου γίνεται σε δύο εισαγωγές, μία εισαγωγή για την ώρα και μία για τα λεπτά. Για την αποθήκευση της φαρμακευτικής αγωγής στην Firestore, στην διεπαφή υπάρχει και ένα κουμπί με την τιμή "προσθήκη" και έχει ένα ακροατή στο γεγονός κλικ που θα αναλυθεί στην συνέχεια.

Σχήμα 4.15: Διεπαφή χρήστη προσθήκης φαρμακευτικής αγωγής καθημερινά

Σχήμα 4.16: Μέρος διεπαφής χρήστη προσθήκης φαρμακευτικής αγωγής συγκεκριμένες μέρες

Τα συστατικά εμφανίζονται πάνω σε άσπρα μικρά ορθογώνια πλαίσια με στρογγυλεμένες γωνίες. Η δημιουργία του πλαισίου έγινε με ένα νέο XML αρχείο. Εκεί ορίστηκε το περιθώριο σε σχέση με τα συστατικά που θα προστεθούν στο σχήμα και η στρογγύλευση των γωνιών του πλαισίου. Αυτό το κομμάτι διεπαφής χρήστη μπορεί να εισηχθεί σε συστατικά ή ομάδες συστατικών της διεπαφής.

Για την προσθήκη της φαρμακευτικής αγωγής δημιουργήθηκε αρχικά ένα αντικείμενο. Το αντικείμενο αυτό περιέχει την πληροφορία που εισάγεται στην συλλογή που αποθηκεύεται η

φαρμακευτική αγωγή (medicines) της Firestore. Πιο συγκεκριμένα έχει ως ιδιότητες (properties) ένα καθολικά μοναδικό χαρακτηριστικό που είναι το μοναδικό κλειδί αναγνώρισης του συγκεκριμένου φαρμάκου. Επίσης έχει την ονομασία και την ποσότητα/σχόλια, μία λίστα που περιέχει τις ημέρες που πρέπει να καταναλωθεί το φάρμακο. Τέλος, έχει τις ώρες και τα λεπτά χορήγησης του φαρμάκου.

Το αντικείμενο που αναφέρεται παραπάνω, περιγράφει μόνο ένα φάρμακο. Για την περιγραφή ολόκληρης της φαρμακευτικής αγωγής δημιουργήθηκε μία κλάση που έχει ως ιδιότητα, μία λίστα από φάρμακα.

Το σκεπτικό είναι το πέρασμα των δεδομένων της Firestore στην μνήμη του προγράμματος στην δομή που αναφέρθηκε προηγουμένως, η εμπλούτιση τους με τις προσθήκες του χρήστη και τέλος η ενημέρωση της συλλογής της Firestore. Το παραπάνω είναι αναγκαίο γιατί με κάθε ενημέρωση στην Firestore τα παλιά δεδομένα αντικαθίστανται.

Για την προσθήκη της φαρμακευτικής αγωγής στην Firestore προστέθηκε ένας ακροατής στο κουμπί "προσθήκη". Σε αυτό το σημείο, το σύστημα κάνει ελέγχους στα δεδομένα που έχει προσθέσει ο χρήστης. Για να είναι αποδεκτά τα δεδομένα πρέπει όλα τα πεδία να είναι συμπληρωμένα. Η ώρα να πρέπει να αντιπροσωπεύει μία τιμή μικρότερη του 24 και τα λεπτά να έχουν τιμή μικρότερη του 59. Αν οι εισαγωγές του χρήστη είναι σωστές τότε δημιουργείται το αντικείμενο που θα κάνει ενημέρωση την Firestore. Αν οι εισαγωγές δεν είναι αποδεκτές εμφανίζεται στον χρήστη το κατάλληλο μήνυμα και η συναλλαγή ακυρώνεται.

4.4.11 Αφαίρεση φαρμακευτικής αγωγής

Στις ρυθμίσεις, στην καρτέλα που γίνεται η διαχείριση της φαρμακευτικής αγωγής με την επιλογή "διαγραφή" από την διεπαφή χρήστη, το σύστημα περιμένει από τον χρήστη να πατήσει πάνω σε κάποιο στοιχείο της φαρμακευτικής αγωγής. Το κάθε μέρος της διεπαφής που αναπαριστά ένα φάρμακο αντιπροσωπεύεται προς το σύστημα με το καθολικά μοναδικό χαρακτηριστικό του. Έτσι, γίνεται ενημέρωση της λίστας της φαρμακευτικής αγωγής χωρίς το φάρμακο που επιλέχθηκε και ενημέρωση της συλλογής medicines στην Firestore με την νέα λίστα.

4.4.12 Προβολή φαρμακευτικής αγωγής

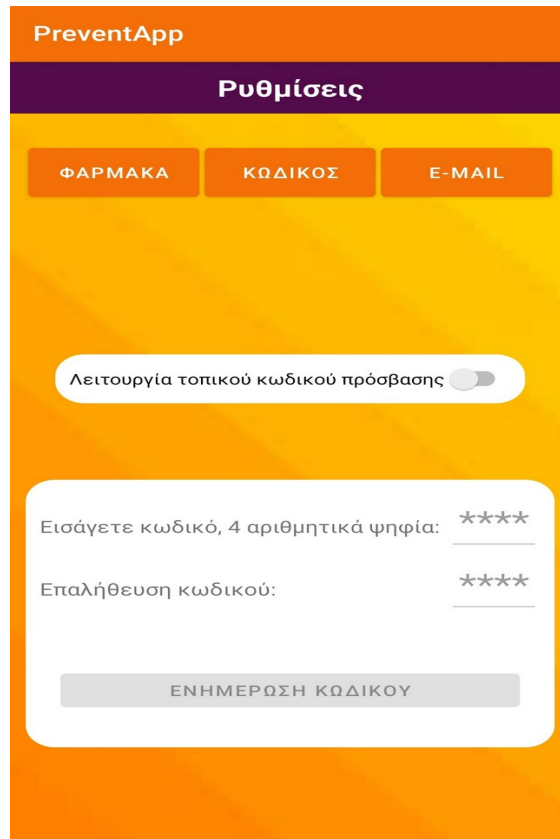
Ο χρήστης μπορεί να δει την φαρμακευτική του αγωγή επιλέγοντας από το κύριο μενού την επιλογή "φάρμακα". Με αυτή την ενέργεια εμφανίζεται στον χρήστη μία λίστα με την φαρμακευτική του αγωγή. Η διεπαφή χρήστη αποτελείται από μεγάλα γράμματα, περιέχει όλη την χρήσιμη πληροφορία (σχήμα 4.17) και υπάρχει η δυνατότητα μετακίνησης προς τα κάτω (scroll down) εφόσον τα φάρμακα που πρέπει να εμφανιστούν δεν χωράνε στην οθόνη.

PreventApp	
<p>φάρμακο1</p> <p>1 μετά το φαγητό</p>	<p>20:20</p> <p>ΚΑΘΗΜΕΡΙΝΑ</p>
<p>φάρμακο2</p> <p>2 δισκία</p>	<p>10:40</p> <p>ΔΕΥ, ΠΕ</p>
<p>φάρμακο3</p> <p>1 δισκίο</p>	<p>12:30</p> <p>ΚΑΘΗΜΕΡΙΝΑ</p>

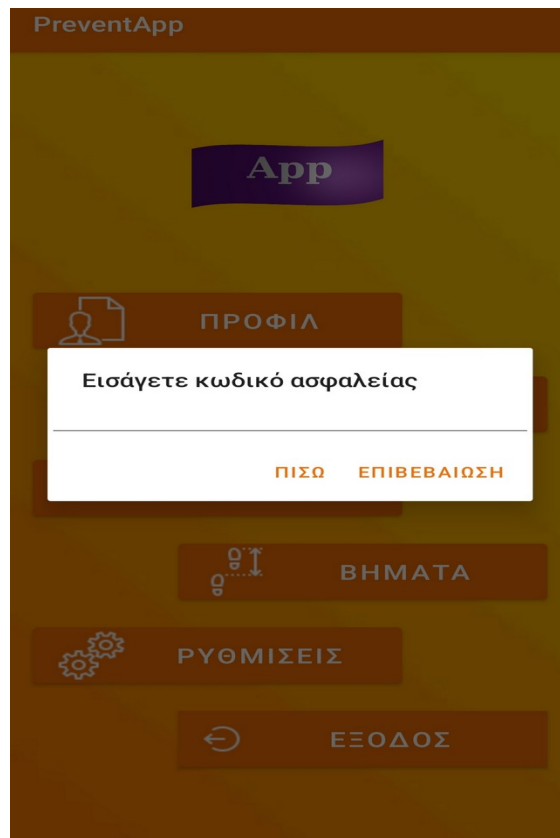
Σχήμα 4.17: Εμφάνιση φαρμακευτικής αγωγής

4.4.13 Κλείδωμα ρυθμίσεων

Όπως έχει ήδη αναφερθεί, μπορεί να υπάρχει κάποιο οικείο πρόσωπο που παρακολουθεί την πορεία του βασικού χρήστη. Αυτό το άτομο μπορεί να προσθέτει φαρμακευτική αγωγή και την ηλεκτρονική του διεύθυνση για την αποστολή ηλεκτρονικών μηνυμάτων που δείχνουν την δραστηριότητα του χρήστη όπως αναφέρθηκε. Τα παραπάνω αποτελούν κατηγορίες ρυθμίσεων. Τα δεδομένα αυτά πρέπει να μην είναι προσβάσιμα από τον άμεσο χρήστη αλλά μόνο από το οικείο πρόσωπο. Για τον λόγο αυτό είναι δυνατόν μέσω των ρυθμίσεων να ορισθεί και κάποιος κωδικός πρόσβασης όπως φαίνεται στο σχήμα 4.18 ώστε οι ρυθμίσεις να μην είναι επιρρεπείς σε αλλαγές που ο κύριος χρήστης ίσως κάνει κατά λάθος. Έτσι, αν έχει ορισθεί κωδικός, ζητείται σε κάθε προσπάθεια που ο χρήστης προσπαθεί να ανοίξει τις ρυθμίσεις (σχήμα 4.19).



Σχήμα 4.18: Διαχείριση κλειδώματος ρυθμίσεων



Σχήμα 4.19: Διεπαφή εισαγωγής κωδικού ρυθμίσεων

Η λειτουργία του τοπικού κωδικού πρόσβασης είναι δυνατόν να απενεργοποιηθεί (για τις περιπτώσεις χρήστη χωρίς την βοήθεια οικείου προσώπου). Με την δημιουργία λογαριασμού στην εφαρμογή δεν υπάρχει κανένας προκαθορισμένος τοπικός κωδικός, αλλά το σύστημα προετοιμάζεται για τον σκοπό αυτό (αρχικοποίηση των πεδίων στη βάση δεδομένων).

4.4.14 Προφίλ χρήστη

Ο χρήστης μπορεί να δει τα επιτεύγματα του οποιαδήποτε στιγμή από την επιλογή "προφίλ" του κυρίως μενού. Εκεί, εμφανίζεται το αρχικό γράμμα της ηλεκτρονικής διεύθυνσης που χρησιμοποιείται για σύνδεση πάνω σε ένα πορτοκάλι στρογγυλό πλαίσιο. Ακριβώς από κάτω μπορεί να δει τα ημερήσια/συνολικά βήματα και ημερήσιους/συνολικούς πόντους που έχει συγκεντρώσει ως εκείνη την στιγμή. Έτσι, είναι σε θέση να πάρει απόφαση ποιο από αυτά θέλει να βελτιώσει.

4.5 Unity εφαρμογή

Στην Unity εφαρμογή υλοποιήθηκαν τα παιχνίδια. Πρόκειται για μία ξεχωριστά εγκατεστημένη εφαρμογή που εκκινείται από την Android. Οι παρακάτω ενότητες περιγράφουν τις λειτουργίες, όλα τα πιθανά σενάρια και τι ακριβώς συμβαίνει στο σύστημα με κάθε ενέργεια του χρήστη.

4.5.1 Πλοήγηση

Με την εκκίνηση της Unity εφαρμογής, ο χρήστης βλέπει το μενού. Μπορεί να επιλέξει ανάμεσα σε "κάρτες", "παζλ", "αριθμοί" και "πίσω στην εφαρμογή" (σχήμα 4.20). Οι επιλογές αυτές είναι κουμπιά και κάθε κουμπί ενεργοποιείται σε κλικ γεγονότα. Οι τρεις πρώτες επιλογές οδηγούν τον χρήστη στην αντίστοιχη σκηνή. Ο χρήστης μπορεί να κλείσει την εφαρμογή Unity, και να επαναφερθεί στην Android, πατώντας το κουμπί "κλείσιμο της εφαρμογής". Ο ακροατής αυτής της ενέργειας εκτελεί μέθοδο του συστήματος που τερματίζει την εφαρμογή.



Σχήμα 4.20: Μενού πλοήγησης

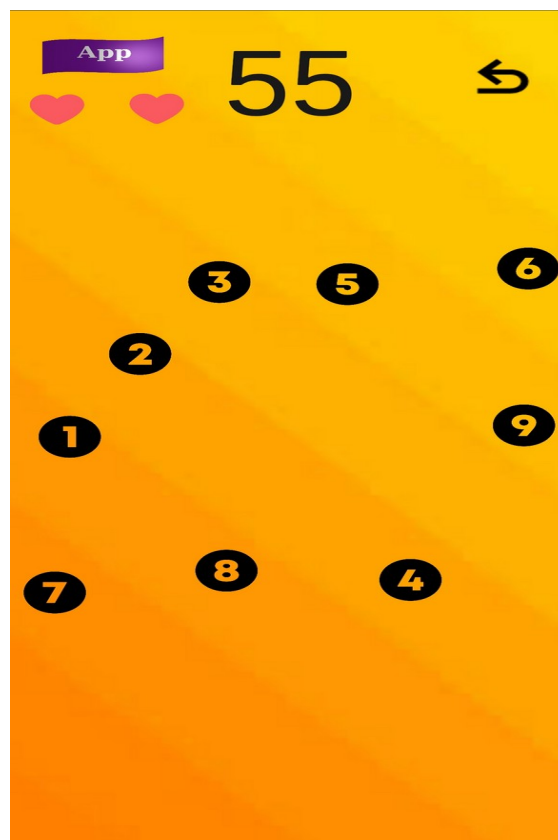
Η εφαρμογή Android ενσωματώνει τις λειτουργίες πλοήγησης της συσκευής, κάτι που δεν συμβαίνει με την εφαρμογή Unity. Για τον σκοπό αυτό δημιουργήθηκε αντικείμενο που δίνει στον χρήστη την δυνατότητα να περιηγηθεί στην προηγούμενη σκηνή. Όπως φαίνεται στο σχήμα 4.21 πρόκειται για ένα εικονίδιο. Το εικονίδιο ενεργοποιείται με το άγγιγμα του χρήστη και αντιδρά με την μέθοδο LoadScene. Η μέθοδος αυτή φέρνει στο προσκήνιο οποια σκηνή έχει οριστεί σε κάθε περίπτωση.



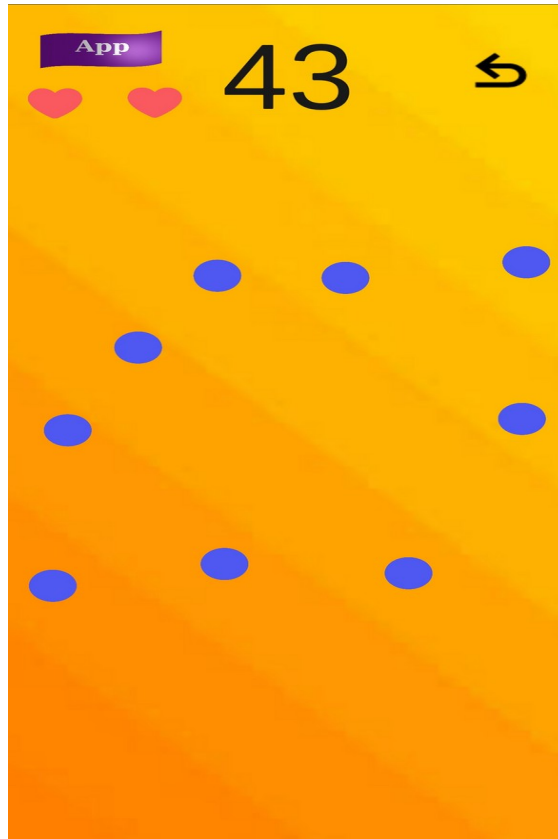
Σχήμα 4.21: Διεπαφή πλοήγησης στην προηγούμενη οθόνη

4.5.2 Παιχνίδι αριθμών

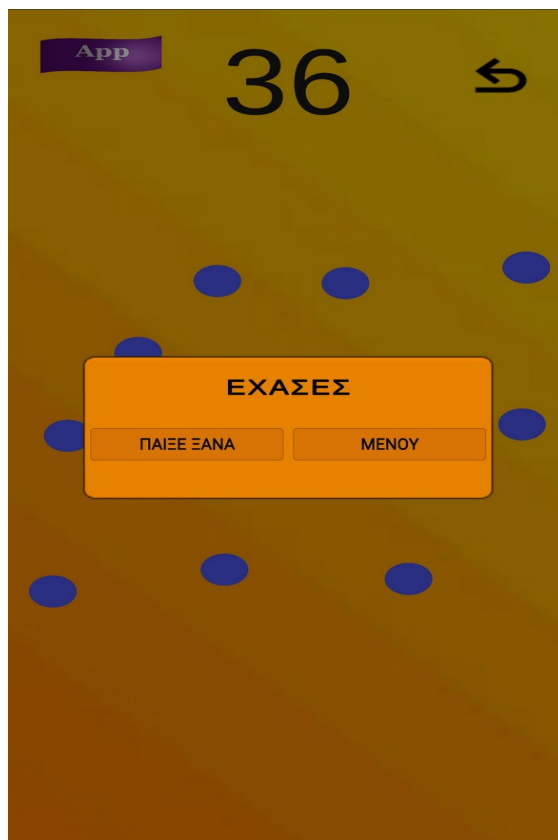
Το παιχνίδι αυτό εμφανίζει στον χρήστη 9 αριθμούς (από το ένα έως το εννέα) για δεκαπέντε δευτερόλεπτα (σχήμα 4.22). Ο χρήστης στο διάστημα αυτό πρέπει να απομνημονεύσει την θέση του κάθε αριθμού. Με το πέρας των δεκαπέντε δευτερολέπτων οι αριθμοί αυτοί κρύβονται (σχήμα 4.23). Σκοπός του παιχνιδιού είναι ο χρήστης να θυμηθεί την σειρά τους μέσα σε χρονικό διάστημα εξήντα δευτερολέπτων και να τους αποκαλύψει με αύξουσα σειρά. Έχει την δυνατότητα δύο λαθών. Με το τρίτο λάθος το παιχνίδι τερματίζεται και ο χρήστης χάνει (σχήμα 4.24). Για να νικήσει πρέπει να αποκαλύψει και τον ένατο αριθμό. Με κάθε του νίκη κερδίζει έναν "πόντο παιχνιδιών".



Σχήμα 4.22: Διεπαφή παιχνιδιού αριθμών πριν την κάλυψη



Σχήμα 4.23: Διεπαφή παιχνιδιού αριθμών με την κάλυψη



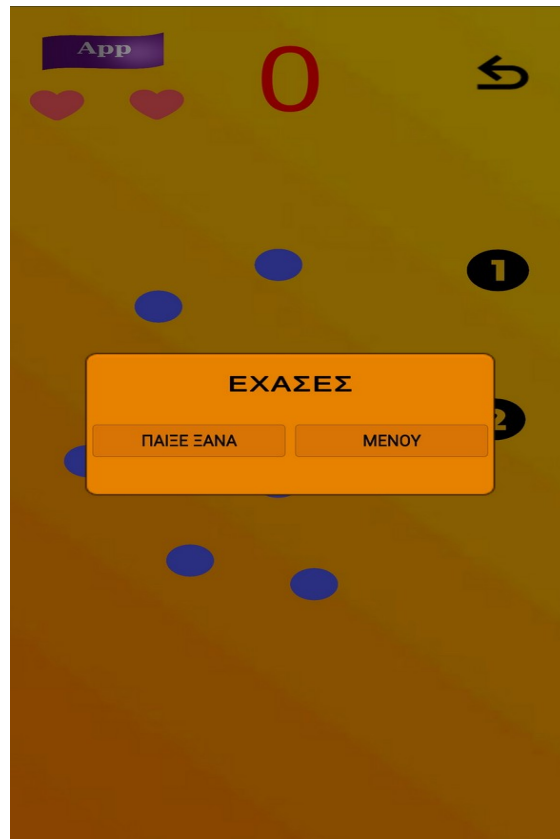
Σχήμα 4.24: Διεπαφή παιχνιδιού αριθμών στο τρίτο λάθος

Για την δημιουργία αυτού του παιχνιδιού χρειάστηκε μία σκηνή. Σε αυτή την σκηνή βρίσκονται όλα τα αντικείμενα που βλέπει και αλληλεπιδρά ο χρήστης. Πιο συγκεκριμένα αποτελείται από τις δύο ζωές, τους αριθμούς, το κουμπί περιήγησης στην προηγούμενη σκηνή, την αντίστροφη μέτρηση και δύο μηνύματα που εμφανίζονται για να ενημερώσουν τον χρήστη αν κέρδισε ή έχασε. Οι αριθμοί έχουν τοποθετηθεί στο κέντρο της οθόνης και με την έναρξη του κάθε παιχνιδιού παίρνουν τυχαίες θέσεις μέσα στην οθόνη.

Για την συνολική λειτουργικότητα του παιχνιδιού χρησιμοποιήθηκαν δύο σενάρια. Το ένα σενάριο διαχειρίζεται την αντίστροφη μέτρηση. Στο σενάριο αυτό, δηλώνονται ο τρέχων χρόνος, και ο αρχικός χρόνος και αρχικοποιούνται με το εξήντα. Δηλώνεται επίσης το αντικείμενο της σκηνής που αναπαριστά την αντίστροφη μέτρηση. Στο αντικείμενο αυτό έχει δοθεί το συστατικό "Text" με τιμή "00" και το σενάριο. Στην Update μέθοδο η μεταβλητή του τρέχων χρόνου μειώνεται κατά ένα. Αυτό συμβαίνει πολλαπλασιάζοντας το 1 με το `deltatime`, ώστε να επιτευχθεί η ανεξαρτητοποίηση του χρονομέτρου από τα πλαίσια (frames) και να γίνεται ενημέρωση του τρέχων χρόνου και εμφάνιση κάθε ένα δευτερόλεπτο ακριβώς. Όταν η αντίστροφη μέτρηση φτάσει στα τρία δευτερά ή λιγότερα η αναπαράσταση του γίνεται κόκκινη (σχήμα 4.25). Αυτό συμβαίνει αλλάζοντας το `color` πεδίο της του τρέχων χρόνου. Αν ο χρόνος φτάσει στο μηδέν εμφανίζεται μέσα στην σκηνή του παιχνιδιού μήνυμα που δηλώνει στον χρήστη ότι έχασε.

Το άλλο σενάριο διαχειρίζεται την υπόλοιπη υλοποίηση του παιχνιδιού. Με την εκκίνηση του σεναρίου, με την `Random.Range` αποδίδονται τυχαίες τιμές όσον αφορά τον οριζόντιο και κάθετο άξονα. Έτσι αλλάζοντας το πεδίο της θέσης του κάθε αντικειμένου, αυτά αλλάζουν την θέση τους στην σκηνή. Επίσης καλείται μία μέθοδος που καλύπτει τους αριθμούς δεκαπέντε δευτερόλεπτα από την έναρξη της σκηνής (οι αριθμοί κρύβονται). Τέλος, στο σενάριο υλοποιείται μία σειρά ελέγχων σε σχέση με το κατά πόσο ο χρήστης επιλέγει τους αριθμούς με την σειρά οι οποίοι ενεργοποιούνται σε κάθε κλικ γεγονός του χρήστη.

Το σκεπτικό υλοποίησης του αλγορίθμου είναι το εξής. Ελέγχεται το όνομα του αριθμού που πατιέται και αν αυτό είναι ίδιο με το όνομα του αριθμού "1" τότε αποκαλύπτεται ο αριθμός στον χρήστη. Στην συνέχεια γίνεται ο έλεγχος αν το όνομα του αντικείμενου που πατήθηκε είναι το "2", σε αυτή την περίπτωση πρέπει να ισχύει ότι και το "1" έχει αποκαλυφθεί και τότε θα αποκαλυφθεί ο αριθμός στον χρήστη. Σε περίπτωση που δεν επιλεγεί το σωστό νούμερο μειώνεται μία "ζωή" από τον χρήστη. Η αφαίρεση των "ζωών" από την σκηνή γίνεται με την μέθοδο `SetActive` και ψευδή (false) τιμή, ουσιαστικά αποκρύπτονται από την διεπαφή χρήστη. Το σενάριο ελέγχει το πότε θα αποκαλυφθεί ο τελευταίος αριθμός για να εμφανίσει στον χρήστη το κατάλληλο μήνυμα. Το ένα μήνυμα ενημερώνει τον χρήστη ότι κέρδισε, ενώ το άλλο τον ενημερώνει ότι έχασε (αυτό συμβαίνει αν χάσει όλες τις ζωές ή λήξει ο χρόνος) όπως φαίνεται στο σχήμα 4.25.



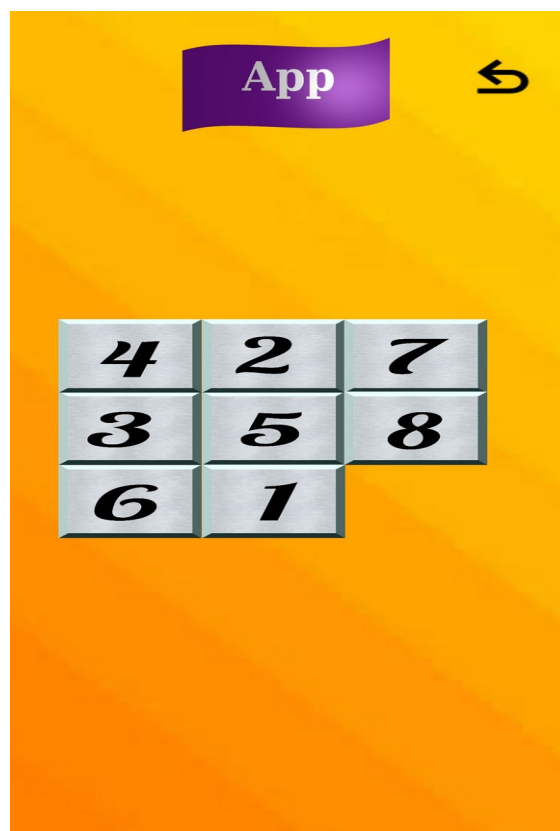
Σχήμα 4.25: Διεπαφή παιχνιδιού αριθμών (ενημέρωση χρήστη πως έχασε, αποκάλυψη αριθμών, κόκκινος χρόνος)

4.5.3 Παζλ

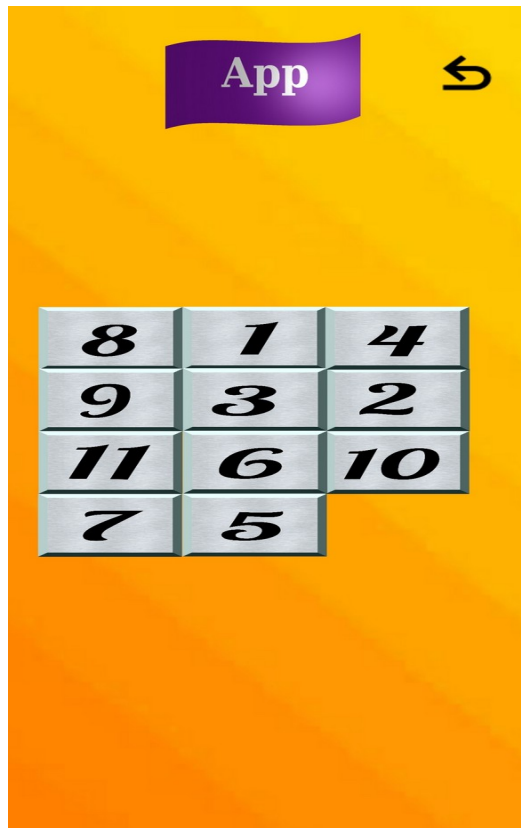
Για το παιχνίδι αυτό, ο χρήστης καλείται να επιλέξει ανάμεσα σε τρία επίπεδα δυσκολίας από ένα μενού (σχήμα 4.26). Το παιχνίδι αυτό είναι ένα παζλ που αποτελείται από κομμάτια με αριθμούς που έχουν μπερδεμένη σειρά. Στα σχήματα 4.27, 4.28, 4.29 φαίνονται οι τρεις διαφορετικές διεπαφές χρήστη ανάλογα το επίπεδο.



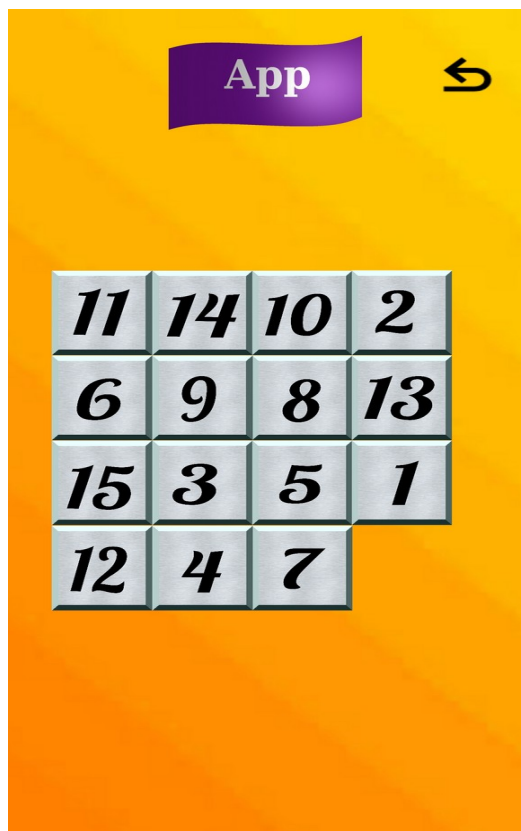
Σχήμα 4.26: Μενού επιλογής επιπέδου στο παιχνίδι του παζλ



Σχήμα 4.27: Επίπεδο 1 του παζλ



Σχήμα 4.28: Επίπεδο 2 του παζλ



Σχήμα 4.29: Επίπεδο 3 του παζλ

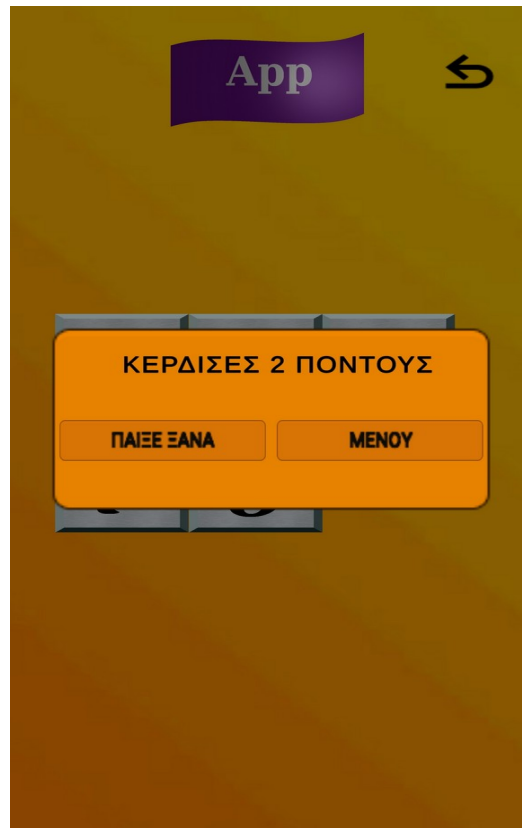
Σκοπός του χρήστη είναι να κινεί τα κομμάτια με σκοπό να τοποθετήσει τους αριθμούς που έχουν τα κομμάτια του παζλ σε αύξουσα σειρά. Για να συμβεί αυτό στην σκηνή προσφέρεται ένα άδειο πεδίο που δεν περιέχει κάποιο κομμάτι παζλ. Έτσι ο χρήστης μπορεί να μεταφέρει τα μέρη του παζλ εκμεταλλευόμενος την κενή θέση.

Η σκηνή του κάθε επιπέδου αποτελείται από τα εξής αντικείμενα. Τα κομμάτια του παζλ με τους αριθμούς, ένα κενό αντικείμενο (στις ίδιες διαστάσεις με τα κομμάτια του παζλ), ένα κουμπί επιστροφής στην προηγούμενη σκηνή και το μήνυμα που δείχνει στον χρήστη ότι κέρδισε.

Για την υλοποίηση του παιχνιδιού χρησιμοποιήθηκαν δύο σενάρια. Το ένα σενάριο έχει ορισθεί σε κάθε κομμάτι του παζλ και ελέγχει αν το τρέχων κομμάτι είναι στην σωστή θέση και ένα σενάριο που έχει ορισθεί στο αντικείμενο της κάμερας και διαχειρίζεται τις υπόλοιπες λειτουργίες του παιχνιδιού, δηλαδή το ανακάτεμα των κομματιών του παζλ και τον έλεγχο αν όλα τα κομμάτια είναι στην σωστή θέση ώστε ο χρήστης να κερδίσει.

Το σενάριο που ελέγχει το κάθε κομμάτι ξεχωριστά αν βρίσκεται στην σωστή θέση έχει ορισθεί σε κάθε αντικείμενο που αναπαριστά ένα κομμάτι του παζλ. Ελέγχει αν η θέση που κατευθύνει ο χρήστης το κομμάτι είναι η σωστή, και έτσι κρατάει σε μία μεταβλητή αληθή τιμή που χρησιμοποιείται από το άλλο σενάριο, διαφορετικά κρατάει ψευδή τιμή.

Στο δεύτερο σενάριο, κατά την Start μέθοδο γίνεται το ανακάτεμα των κομματιών του παζλ και απενεργοποιείται η εμφάνιση του μηνύματος που εμφανίζεται στον χρήστη όταν κερδίζει. Στην μέθοδο Update χρησιμοποιείται η Ray δομή δεδομένων του γεγονότος κλικ, έτσι υπάρχει πρόσβαση στις τιμές του γεγονότος. Στην συνέχεια γίνεται χρήση της συνάρτησης Raycast που επιστρέφει πληροφορίες σχετικές με το γεγονός όταν αυτό συμβεί. Με βάση αυτές τις πληροφορίες γίνεται έλεγχος αν το κομμάτι που πατήθηκε είναι δίπλα στο κενό αντικείμενο (και όχι μακριά ή διαγώνια) ώστε να μετακινηθεί στην άδεια θέση και η άδεια θέση να πάει εκεί που ήταν προηγουμένως το κομμάτι. Για τον έλεγχο χρησιμοποιήθηκε η μέθοδος Distance του Vector2 και οι αλλαγές στις θέσεις με αλλαγές στις θέσεις (positions) της ιδιότητας μετατροπής (transform) των αντικειμένων. Κατά την Update μέθοδο γίνεται επίσης ο έλεγχος αν όλα τα κομμάτια του παζλ βρίσκονται στην σωστή θέση (προκύπτει από το σενάριο που αναφέρθηκε παραπάνω). Αυτό συμβαίνει με την βοήθεια ενός μετρητή που μετράει το πλήθος των κομματιών που βρίσκονται στην σωστή θέση. Αν το πλήθος των σωστών θέσεων είναι ίσο με το πλήθος των κομματιών του παζλ τότε ενεργοποιείται και το πάνελ που δείχνει στον χρήστη ότι κέρδισε όπως φαίνεται στο σχήμα 4.30.



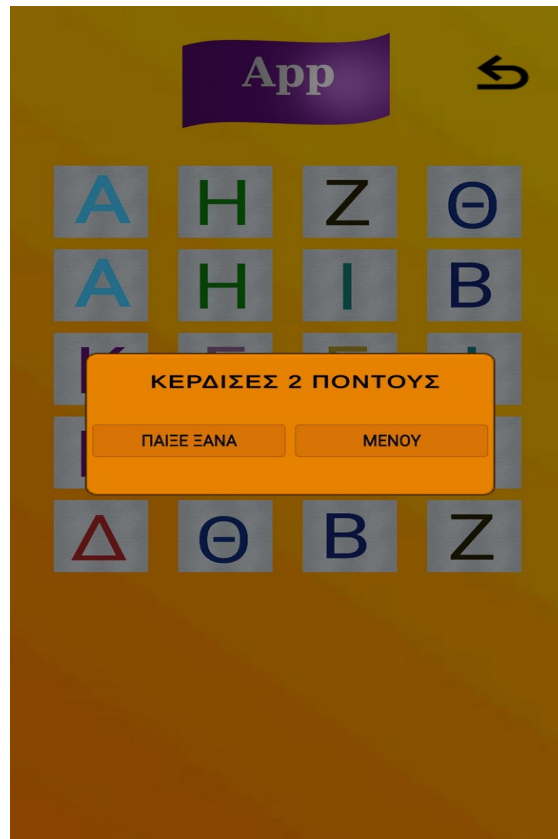
Σχήμα 4.30: Ενημέρωσης νίκης στο επίπεδο του παζλ.

4.5.4 Παιχνίδι καρτών

Στο παιχνίδι αυτό, εμφανίζονται στον χρήστη είκοσι κρυμμένες κάρτες. Στις κάρτες αυτές υπάρχουν δέκα διαφορετικά ζευγάρια (γράμματα του αλφαβήτου). Σκοπός του χρήστη είναι να αποκαλύψει όλα τα ζευγάρια. Πιο συγκεκριμένα αγγίζοντας μία κάρτα, το περιεχόμενο της αποκαλύπτεται. Όταν αγγίζει και την δεύτερη, το περιεχόμενο πάλι θα αποκαλυφθεί. Αν αυτές οι δύο επιλογές αποτελούν ένα ζευγάρι τότε ο χρήστης έχει βρει επιτυχώς ένα ζευγάρι και πρέπει να βρει και τα υπόλοιπα ώστε να κερδίσει. Αν οι δύο κάρτες που άνοιξε δεν είναι ζευγάρι, τότε το περιεχόμενο τους κρύβεται πάλι.

Για την δημιουργία των καρτών χρησιμοποιήθηκε ένα αντικείμενο παιχνιδιού. Το εν λόγω αντικείμενο έχει και την μπροστά αλλά και την πίσω όψη της κάρτας σαν πληροφορία. Το αντικείμενο αυτό χρησιμοποιήθηκε από σενάριο για την παραγωγή όλων των καρτών στην σκηνή με τη μέθοδο κλωνοποίησης αντικειμένων (clone). Οι παραγόμενες κάρτες, με κάθε νέο παιχνίδι παίρνουν τυχαίες θέσεις.

Η εναλλαγή ανάμεσα στις δύο όψεις του αντικειμένου γίνεται με ενεργοποίηση κλικ γεγονόςτος, με μέθοδο του συστήματος (SetActive). Έτσι "εξαφανίζεται" ή "εμφανίζεται" ένα μέρος του αντικειμένου και φαίνεται στον χρήστη η όψη της κάρτας που πρέπει.



Σχήμα 4.31: Διεπαφή ενημέρωσης χρήστη

4.5.5 Ανάλυση των μηνυμάτων προς τον χρήστη

Όταν ο χρήστης κερδίσει (ή χάσει στην περίπτωση του παιχνιδιού με τους αριθμούς), στην σκηνή παρεμβαίνει μια ακόμα διεπαφή που ενημερώνει πόσους πόντους κέρδισε από αυτό το παιχνίδι. Κατά την διάρκεια εμφάνισης το υπόλοιπο παιχνίδι παγώνει και περιμένει από τον χρήστη να επιλέξει ανάμεσα στο "παίξε ξανά" ή "μενού" ώστε να περιηγηθεί στην κατάλληλη σκηνή.

4.5.6 Προσθήκη βαθμολογίας παιχνιδιών στην Firestore

Γενικότερα, με κάθε νίκη γίνεται κλήση μίας συγκεκριμένης μεθόδου που παίρνει σαν παράμετρο ένα ακέραιο αριθμό (πόντοι) και είναι υπεύθυνη για να προσθέτει την βαθμολογία στην βάση δεδομένων. Η προσθήκη βαθμολογίας παιχνιδιών γίνεται στην συλλογή που αφορά τους ημερήσιους πόντους παιχνιδιών (games), με έγγραφο το μοναδικό χαρακτηριστικό του χρήστη και ως πεδίο έχει την ημερήσια βαθμολογία του χρήστη. Το καθολικά μοναδικό χαρακτηριστικό του χρήστη προέρχεται από την Android εφαρμογή. Έτσι, σε μία δομή ορίζονται οι πόντοι που έχουν ζητηθεί από την βάση δεδομένων και προστίθενται οι πόντοι σύλλεξε ο χρήστης. Τέλος γίνεται ενημέρωση την βάσης από την δομή που προκύπτει.

4.6 Επίλογος

Σε αυτό το κεφάλαιο στην αρχή περιγράφηκαν κάποιες γενικές αρχές (δομή βάσης δεδομένων, εμπειρία χρήστη) που εφαρμόστηκαν κατά την διάρκεια προγραμματισμού του λογισμικού. Το υπόλοιπο κεφάλαιο χωρίστηκε σε δύο μέρη. Το πρώτο μέρος αναφέρεται στην υλοποίηση της Android εφαρμογής, ενώ το δεύτερο στην δημιουργία της Unity εφαρμογής. Έγινε εκτενής αναφορά των

λειτουργικότητων και των διεπαφών χρήστη. Παρουσιάστηκαν επίσης κάποια προβλήματα που συναντήθηκαν και οι τρόποι που αυτά αντιμετωπίστηκαν.

Κεφάλαιο 5ο: Συμπεράσματα και προτάσεις βελτίωσης

5.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζονται τα συμπεράσματα που έχουν διεξαχθεί από την υλοποίηση του λογισμικού για την πρόληψη της άνοιας. Επίσης, αναφέρονται περαιτέρω βελτιώσεις και προσθήκες που μπορούν να υλοποιηθούν στο μέλλον για καλύτερα αποτελέσματα κατά την χρήση της εφαρμογής.

5.2 Συμπεράσματα

Στα πλαίσια της παρούσας πτυχιακής εργασίας, αναπτύχθηκε λογισμικό που έχει ως στόχο την πρόληψη και υποβοήθηση της νόσου της άνοιας. Για τον σκοπό αυτό, δημιουργήθηκαν παιχνίδια σε περιβάλλον Unity με πρόθεση την εξάσκηση των πνευματικών ικανοτήτων του χρήστη. Τα παιχνίδια αυτά φιλοξενούνται σε μία Android εφαρμογή η οποία έχει και επιπλέον λειτουργίες όπως η καταμέτρηση βημάτων και η αποθήκευση φαρμακευτικής αγωγής του χρήστη. Για αποθήκευση των δεδομένων του χρήστη και διαχείριση του λογαριασμού έγινε χρήση των υπηρεσιών Firestore και Authentication αντίστοιχα.

Κατά την διάρκεια εκμάθησης και ανάπτυξης λογισμικού έμαθα πολλά πράγματα, τόσο για τις γλώσσες προγραμματισμού Java και C# όσο και για τις πλατφόρμες Android, Unity και Firebase που χρησιμοποίησα. Με δυσκόλεψε περισσότερο η δημιουργία των παιχνιδιών, καθώς ήταν μία πρωτόγνωρη εμπειρία και δεν είχα ασχοληθεί στο παρελθόν.

Η εμπειρία που αποκόμισα από την ΠΕ με βοήθησε να καταλάβω πώς στον κλάδο ανάπτυξης λογισμικού δεν υπάρχουν όρια στις δυνατότητες ούτε στα διαθέσιμα εργαλεία. Πάντα μπορούν να γίνουν βελτιώσεις και προσθήκες για την καλύτερη εμπειρία του χρήστη με οποιαδήποτε εφαρμογή. Ακόμη, κατάλαβα πως η τεχνολογία μπορεί να συμβάλλει ή και να λύσει πολλά προβλήματα που αντιμετωπίζουν πολλοί κλάδοι με την κατάλληλη έρευνα και ανάπτυξη λογισμικού.

5.3 Βελτιώσεις

Υπάρχουν διάφορες βελτιώσεις που θα μπορούσαν να δώσουν στον χρήστη ένα καλύτερο αποτέλεσμα και μία πιο ολοκληρωμένη εμπειρία. Επίσης, σε σχέση με την υλοποίηση κώδικα θα μπορούσε να χρησιμοποιηθεί κάποιο πρότυπο σχεδίασης (Design Pattern) ώστε η επεκτασιμότητα, η ανάγνωση και η διαχείριση να είναι πιο εύκολη για τον προγραμματιστή. Το πρότυπο σχεδίασης θα προσέφερε πιο εύκολο διαμοιρασμό της εργασίας σε μία ομάδα, όπου το κάθε μέλος θα ήταν εξοικειωμένο και θα εμβάθυνε σε συγκεκριμένο μέρος της εφαρμογής.

Το λογισμικό απευθύνεται κυρίως σε ανθρώπους μεγάλης ηλικίας. Αυτό σημαίνει πως υπάρχουν πολλές πιθανότητες οι χρήστες να μην είναι εξοικειωμένοι με την τεχνολογία. Μία παρέμβαση σε αυτό το πρόβλημα θα ήταν ένας ηχητικός βοηθός που θα εξηγούσε στον χρήστη τι ακριβώς μπορεί να κάνει σε κάθε οθόνη.

Στην εφαρμογή υπάρχει η δυνατότητα καταχώρησης φαρμακευτικής αγωγής του χρήστη. Μία πολύ βοηθητική προσθήκη σε αυτή την λειτουργικότητα θα μπορούσε να είναι η ειδοποίηση του χρήστη για το φάρμακο που έχει να πάρει εκείνη την στιγμή. Αυτό θα έκανε τον χρήστη να νιώθει πιο ασφαλής στην συνέπεια του απέναντι στην φαρμακευτική του αγωγή.

Μία ακόμα βελτίωση για καλύτερα αποτελέσματα και εμπειρία χρήστη είναι η προσθήκη περισσότερων παιχνιδιών. Κάτι τέτοιο θα έκανε πιο ενδιαφέρουσα την εμπειρία του χρήστη και έτσι θα τον ωθούσε σε περισσότερη ενασχόληση του με την εφαρμογή και άρα καλύτερα αποτελέσματα.

Κατά την περιήγηση του χρήστη στο προφίλ του, μπορεί να δει την βαθμολογία του από όλα τα παιχνίδια. Μία αλλαγή σε αυτή την λειτουργία θα ήταν η πιο συγκεκριμένη ενημέρωση του, δηλαδή να μπορεί να δει πόσους πόντους έχει συλλέξει από κάθε παιχνίδι συγκεκριμένα.

Θα μπορούσαν να γίνουν προσθήκες στις ρυθμίσεις όπως για παράδειγμα δυνατότητα αλλαγής στο μέγεθος της γραμματοσειράς και αλλαγή χρωμάτων. Κάτι τέτοιο θα έκανε τον χρήστη να νιώθει πιο οικεία στο να χρησιμοποιεί το λογισμικό.

5.4 Επίλογος

Το λογισμικό που παράχθηκε για την ΠΕ μπορεί να λειτουργήσει ως μία πρώτη έκδοση υποβοήθησης και πρόληψης για την νόσο. Ωστόσο χρειάζονται κάποιες προσθήκες που αναφέρθηκαν παραπάνω που θα κάνουν μεγάλη διαφορά στην εμπειρία του χρήστη και έτσι ίσως θα μπορούσε να φτάσει σε επίπεδο να προτείνεται σε ανθρώπους από τους ειδικούς του κλάδου.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] National Collaborating Centre for Mental Health (UK). Dementia: A NICE-SCIE Guideline on Supporting People With Dementia and Their Carers in Health and Social Care. Leicester (UK): British Psychological Society; 2007. (NICE Clinical Guidelines, No. 42.) 4, DEMENTIA. Available from: <https://www.ncbi.nlm.nih.gov/books/NBK55480/>
- [2] Castellani, Rudy J., Raj K. Rolston, and Mark A. Smith. "Alzheimer disease." *Disease-a-month: DM* 56.9 (2010): 484.
- [3] Jiang, Qingguang, et al. "ApoE promotes the proteolytic degradation of A β ." *Neuron* 58.5 (2008): 681-693.
- [4] Iadecola, Costantino. "The pathobiology of vascular dementia." *Neuron* 80.4 (2013): 844-866.8
- [5] Nagata, Ken, et al. "Clinical diagnosis of vascular dementia." *Journal of the neurological sciences* 257.1-2 (2007): 44-48.
- [6] T O'Brien, John, and Alan Thomas. "Vascular dementia." *The Lancet* 386.10004 (2015): 1698-1706.
- [7] McKeith, Ian. "Dementia with Lewy bodies." *Handbook of clinical Neurology* 84 (2007): 531-548
- [8] De Lau, Lonneke ML, and Monique MB Breteler. "Epidemiology of Parkinson's disease." *The Lancet Neurology* 5.6 (2006): 525-535
- [9] Singh, Neha, Viness Pillay, and Yahya E. Choonara. "Advances in the treatment of Parkinson's disease." *Progress in neurobiology* 81.1 (2007): 29-44.
- [10] Neary, David, Julie Snowden, and David Mann. "Frontotemporal dementia." *The Lancet Neurology* 4.11 (2005): 771-780.
- [11] Morris, John C. "The clinical dementia rating (cdr): Current version and." *Young* 41 (1991): 1588-1592.
- [12] Wesson, J., and J. Luchins. "An empirical evaluation of the Global Deterioration Scale for staging Alzheimer's disease." *Am. J. Psychiatry* 149 (1992): 190-194.
- [13] Yuan, Qi, et al. "Staging dementia based on caregiver reported patient symptoms: Implications from a latent class analysis." *PloS one* 15.1 (2020): e0227857.
- [14] Lillo-Crespo, Manuel, et al. "Chess practice as a protective factor in dementia." *International journal of environmental research and public health* 16.12 (2019): 2116.
- [15] Polidori, Maria C., and Ludger Pientka. "A brief update on dementia prevention." *Zeitschrift für Gerontologie und Geriatrie* 45.1 (2012): 7-10.
- [16] Dartigues, Jean François, et al. "Playing board games, cognitive decline and dementia: a French

population-based cohort study." *BMJ open* 3.8 (2013): e002998.

[17] McCallum, Simon. "Gamification and serious games for personalized health." *pHealth*. 2012.

[18] Kang, Kiho, Eun-Jin Choi, and Young-Suk Lee. "Proposal of a serious game to help prevent dementia." *International Conference on Games and Learning Alliance*. Springer, Cham, 2016.

[19] H. Ning, R. Li, X. Ye, Y. Zhang and L. Liu, "A Review on Serious Games for Dementia Care in Ageing Societies," in *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 8, pp. 1-11, 2020, Art no. 1400411, doi: 10.1109/JTEHM.2020.2998055.

[20] Groenewoud, Hanny, et al. "People with dementia playing casual games on a tablet." *Gerontechnology* 16.1 (2017): 37-47.

[21] Google, Platform | Architecture, <https://developer.android.com/guide/platform>, 20/9/2022

[22] Google, Layouts | Android Developers, <https://developer.android.com/develop/ui/views/layout/declaring-layout>, 20/9/2022

[23] Google, App resources overview | Android Developers. <https://developer.android.com/guide/topics/resources/providing-resources>, 21/9/2022

[24] Google, Introduction to Activities | Android Developers <https://developer.android.com/guide/components/activities/intro-activities>, 20/9/2022

[25] Google, App Manifest Overview | Android Developers, <https://developer.android.com/guide/topics/manifest/manifest-intro>, 24/11/2022

[26] Google, Configure your build | Android Developers <https://developer.android.com/studio/build>, 24/11/2022

[27] Chatterjee, Nilanjan, et al. "Real-time communication application based on android using Google firebase." *Int. J. Adv. Res. Comput. Sci. Manag. Stud* 6.4 (2018).

[28] Google, Firebase Authentication, <https://firebase.google.com/docs/auth>, 24/9/2022

[29] Kumar, Ashok. *Mastering Firebase for Android Development: Build real-time, scalable, and cloud-enabled Android apps with Firebase*. Packt Publishing Ltd, 2018.

[30] Google, Firestore | Firebase, <https://firebase.google.com/docs/firestore>, 18/9/2022

[31] Google, Cloud storage for Firebase, <https://firebase.google.com/docs/storage>, 24/9/2022

[32] Google, Firebase Security Rules, <https://firebase.google.com/docs/rules>, 24/9/2022

[33] Juliani, Arthur, et al. "Unity: A general platform for intelligent agents." *arXiv preprint arXiv:1809.02627* (2018).

[34] Unity, Unity-Manual: Lights, <https://docs.unity3d.com/Manual/Lights.html>, 25/9/2022

[35] Unity, Unity-Manual: Cameras, <https://docs.unity3d.com/Manual/Cameras.html>, 25/9/2022

[36] Unity, Unity-Manual: Asset Workflow <https://docs.unity3d.com/Manual/AssetWorkflow.html>, 24/9/2022

[37] Unity, Unity-Manual: Scenes <https://docs.unity3d.com/Manual/CreatingScenes.html>, 25/9/2022

[38] Unity, Unity-Manual: GameObjects <https://docs.unity3d.com/Manual/GameObject.html>, 25/9/2022

[39] Unity, Unity-Manual: Prefabs, <https://docs.unity3d.com/Manual/Prefabs.html>, 25/9/2022