



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

**Σχεδιασμός και ανάπτυξη διαδραστικής εφαρμογής για άτομα με
νευρογνωστικές διαταραχές**

Διπλωματική Εργασία

Δήμου Ευθύμιος

Επιβλέπων: Τσαλαπάτα Χαρίκλεια

Σεπτέμβριος 2023



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

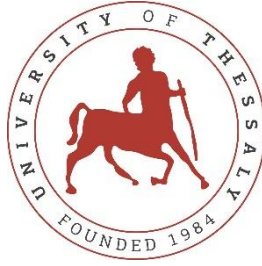
**Σχεδιασμός και ανάπτυξη διαδραστικής εφαρμογής για άτομα με
νευρογνωστικές διαταραχές**

Διπλωματική Εργασία

Δήμου Ευθύμιος

Επιβλέπων: Τσαλαπάτα Χαρίκλεια

Σεπτέμβριος 2023



UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

**Design and development of an interactive application for people
with neurocognitive disorders**

Diploma Thesis

Efthymios Dimou

Supervisor: Harikleia Tsalapata

September 2023

Εγκρίνεται από την Επιτροπή Εξέτασης:

Επιβλέπων/πουσα **Τσαλαπάτα Χαρίκλεια**

Ε.ΔΙ.Π, Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών
Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Μέλος **Σταμούλης Γεώργιος**

Καθηγητής, Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών
Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

Μέλος **Δασκαλοπούλου Ασπασία**

Αναπληρώτρια Καθηγήτρια, Τμήμα Ηλεκτρολόγων Μηχανικών και
Μηχανικών Υπολογιστών, Πανεπιστήμιο Θεσσαλίας

**ΥΠΕΥΘΥΝΗ ΔΗΛΩΣΗ ΠΕΡΙ ΑΚΑΔΗΜΑΪΚΗΣ ΔΕΟΝΤΟΛΟΓΙΑΣ ΚΑΙ ΠΝΕΥΜΑΤΙΚΩΝ
ΔΙΚΑΙΩΜΑΤΩΝ**

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ρητά ότι η παρούσα διπλωματική εργασία, καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας, αποτελούν αποκλειστικά προϊόν προσωπικής μου εργασίας, δεν προσβάλλουν οποιασδήποτε μορφής δικαιώματα διανοητικής ιδιοκτησίας, προσωπικότητας και προσωπικών δεδομένων τρίτων, δεν περιέχουν έργα/εισφορές τρίτων για τα οποία απαιτείται άδεια των δημιουργών/δικαιούχων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον και πληρούν τους κανόνες της επιστημονικής παράθεσης. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή/και πηγές άλλων συγγραφέων αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Δηλώνω επίσης ότι τα αποτελέσματα της εργασίας δεν έχουν χρησιμοποιηθεί για την απόκτηση άλλου πτυχίου. Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Ο/Η Δηλών/ούσα

Δήμου Ευθύμιος

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I also declare that the results of the work have not been used to obtain another degree. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism.

The Declarant

Efthymios Dimou

Ευχαριστίες ή Σχόλια

Θα ήθελα να ευχαριστήσω πολλούς ανθρώπους που, χωρίς αυτούς, δεν θα κατάφερνα να ολοκληρώσω την εργασία αυτήν.

Αρχικά, θα ήθελα να ευχαριστήσω την οικογένεια μου, οι οποίοι με στήριξαν κάθε στιγμή στο πολύχρονο ταξίδι που είχα για την απόκτηση του πτυχίου μου και με βοήθησαν με ό,τι ήταν δυνατό για εκείνους, μέχρι την τελευταία στιγμή.

Επιπλέον, ευχαριστώ τους φίλους μου, που για μένα πάντα ήταν στήριγμα ακόμα και στις πιο δύσκολες εποχές την φοιτητικής ζωής μου και μπορούσα να στραφώ σε αυτούς για οτιδήποτε με απασχολούσε εκείνες τις περιόδους.

Τέλος, θα ήθελα να ευχαριστήσω ονομαστικά την ψυχίατρο-ψυχοθεραπεύτρια Κρεμέτη Δήμητρα, τη νευρολόγο Πανταζή Εύα και τη νευροψυχολόγο Ζούπα Έλλη, οι οποίες μου δώσανε πολύτιμες συμβουλές για την διεκπεραίωση της εφαρμογής που είναι μέρος την διπλωματικής εργασίας.

Κλείνοντας, θα ήθελα να αφιερώσω την διπλωματική εργασία στον παππού μου, Ευθύμιο Δήμου, που η πάθηση που είχε στα τελευταία χρόνια της ζωής του με ενέπνευσε να ασχοληθώ με αυτό τον τομέα υγείας σαν μέρος της εργασίας αυτής.

Σχεδιασμός και ανάπτυξη διαδραστικής εφαρμογής για άτομα με νευρογνωστικές διαταραχές

Δήμου Ευθύμιος

Περίληψη

Η παρούσα διπλωματική εργασία έχει σκοπό την δημιουργία μιας εφαρμογής για άτομα με νευρογνωστικές διαταραχές. Για την σωστότερη δημιουργία της εφαρμογής αυτής, θα πρέπει πρώτα να ορίσουμε τις νευρογνωστικές διαταραχές, οι οποίες είναι διαταραχές του εγκεφάλου που οδηγούν σε μείωση των πνευματικών ικανοτήτων και περιλαμβάνουν ασθένειες όπως το Alzheimer και την νόσο του Parkinson. Επιπλέον θα πρέπει να ορίσουμε και τρόπους μάθησης, όπως την ενεργό μάθηση και την μάθηση με βάση το πρόβλημα, καθώς και να ανακαλύψουμε τι είναι μια εκπαιδευτική εφαρμογή και από τι αποτελείται μια τέτοια εφαρμογή. Τέλος, πριν ξεκινήσουμε την διαδικασία της δημιουργίας θα πρέπει να μελετήσουμε παρόμοιες εφαρμογές στον τομέα και ποια είναι τα σημεία στα οποία υστερούν και μπορούμε να τα καλύψουμε με την δική μας εφαρμογή.

Λέξεις-κλειδιά:

Νευρογνωστικές διαταραχές, τρόποι μάθησης, εκπαιδευτικά παιχνίδια, λογισμικό Android, εκπαιδευτικές δραστηριότητες

Design and development of an interactive application for people with neurocognitive disorders

Efthymios Dimou

Abstract

This thesis aims to create an application for people with neurocognitive disorders. To properly create this application, we should first define neurocognitive disorders, which are brain disorders that lead to a decline in mental abilities and include diseases such as Alzheimer's and Parkinson's disease. In addition, we should also define ways of learning, such as active learning and problem-based learning, as well as discover what an educational application is and what constitutes such an application. Finally, before starting the creation process, we should study similar applications in the field and what are the points where they fall short, and we can cover them with our own application.

Keywords:

Neurocognitive disorders, ways of learning, educational games, Android software, educational activities

Πίνακας περιεχομένων

<i>Ευχαριστίες ή Σχόλια.....</i>	<i>xiii</i>
<i>Περίληψη.....</i>	<i>xv</i>
<i>Abstract</i>	<i>xvii</i>
<i>Πίνακας περιεχομένων.....</i>	<i>xix</i>
<i>Κατάλογος εικόνων.....</i>	<i>xxi</i>
<i>Συντομογραφίες.....</i>	<i>xxiii</i>
Κεφάλαιο 1 Εισαγωγή	1
1.1 Αντικείμενο της διπλωματικής	1
1.1.1 Συνεισφορά.....	1
1.2 Οργάνωση του τόμου	2
Κεφάλαιο 2 Νευρογνωστικές Διαταραχές.....	3
2.1 Ορισμός των Νευρογνωστικών Διαταραχών.....	3
2.2 Κατηγοριοποίηση Νευρογνωστικών Διαταραχών	3
2.3 Κατηγορίες Νευρογνωστικών Διαταραχών.....	4
2.3.1 Παραλήρημα.....	4
2.3.2 Ήπια Νευρογνωστική Διαταραχή	5
2.3.3 Μείζονα Νευρογνωστική Διαταραχή.....	5
2.4 Συνηθισμένες νευρογνωστικές διαταραχές	6
2.4.1 Νόσος Alzheimer.....	6
2.4.2 Μετωπιοκροταφικός Λοβιακός Εκφυλισμός	6
2.4.3 HIV Λοίμωξη.....	7
2.4.4 Νόσος του Σώματος Lewy.....	7
2.4.5 Νόσος του Πάρκινσον	7
2.4.6 Νόσος Prion	8
2.5 Τομείς Ασκήσεων για τις Νευρογνωστικές Διαταραχές	8
Κεφάλαιο 3 Εκπαιδευτικά Παιχνίδια και Τρόποι Μάθησης.....	11

3.1 Εκπαιδευτικά Παιχνίδια.....	11
3.2 Δημιουργία ενός Εκπαιδευτικού Παιχνιδιού.....	12
3.3 Τρόποι Μάθησης.....	14
3.3.1 Ενεργή και Συνεργατική Μάθηση.....	15
3.3.2 Μάθηση με Βάση το Πρόβλημα.....	17
Κεφάλαιο 4 Παρόμοιες Εφαρμογές	21
4.1 Dakim Brain Fitness.....	21
4.1.1 Τεχνολογία NuroLogic™	22
4.2 Tactus Therapy	23
4.3 Lumosity.....	25
4.4 Constant Therapy.....	26
4.5 DiAnoia.....	28
Κεφάλαιο 5 Ανάλυση της Μαθησιακής Εφαρμογής	31
5.1 Αρχικός Σχεδιασμός	31
5.1.1 Περιβάλλον Υλοποίησης της Εφαρμογής	32
5.1.2 Προγραμματιστική Γλώσσα	33
5.2 Βασικές Λειτουργίες	36
5.2.1 Βασικός Σκελετός της Εφαρμογής	36
5.3 Δραστηριότητες	41
5.3.1 Δραστηριότητα Μνήμης	42
5.3.2 Δραστηριότητα Παρατηρητικότητας-Προσοχής.....	46
5.3.3 Δραστηριότητα Λόγου	49
5.3.4 Δραστηριότητα Ταχύτητας Επεξεργασίας	52
5.3 Ολοκλήρωση της εφαρμογής	55
5.4 Μελλοντικές Υλοποιήσεις	56
Βιβλιογραφία.....	58

Κατάλογος εικόνων

Εικόνα 4.1 Dakim Brain Fitness	21
Εικόνα 4.2 Tactus Therapy.....	23
Εικόνα 4.3 Lumosity.....	25
Εικόνα 4.4 Constant Therapy	26
Εικόνα 4.5 DiAnoia.....	28
Εικόνα 5.1 Περιβάλλον εργασίας του Android Studio	32
Εικόνα 5.2 Αρχική οθόνη της εφαρμογής	37
Εικόνα 5.3 Οθόνη σύνδεσης της εφαρμογής.....	38
Εικόνα 5.4 Η οθόνη στην οποία φαίνονται όλες οι δραστηριότητες της εφαρμογής.....	40
Εικόνα 5.5 Η οθόνη που αναφέρει το πώς κερδίζεται η δραστηριότητα μνήμης.....	41
Εικόνα 5.6 Δραστηριότητα μνήμης	43
Εικόνα 5.7 Δραστηριότητα Παρατηρητικότητας-Προσοχής.....	46
Εικόνα 5.8 Δραστηριότητα Λόγου	50
Εικόνα 5.9 Δραστηριότητα Ταχύτητας Επεξεργασίας	53

Συντομογραφίες

βλπ βλέπε

κ.λπ. και λοιπά

κ.ο.κ και ούτω καθεξής

Κεφάλαιο 1 Εισαγωγή

Οι νευρογνωστικές διαταραχές είναι διαταραχές που προσβάλλουν την σωστή εγκεφαλική λειτουργία ενός ατόμου, με αποτέλεσμα να δημιουργούνται προβλήματα στην κινησιολογία του, την φρασεολογία του, ακόμα και στην μνήμη του. Από όταν παρατηρήθηκαν και κατηγοριοποιήθηκαν οι διαταραχές αυτές, οι επιστήμονες προσπάθησαν να βρουν τρόπους αντιμετώπισης τους. Οι τρόποι αυτοί βασίζονται ακόμα κυρίως σε πιο συμβατικά μέσα, όπως, παραδείγματος χάρη, μολύβι και χαρτί.

1.1 Αντικείμενο της Διπλωματικής

Στην διπλωματική αυτήν, προσπαθούμε να βρούμε μια λύση στους συμβατικούς τρόπους που χρησιμοποιούνται ακόμα στην θεραπεία των ατόμων με νευρογνωστικές διαταραχές. Παρόλο που κάποιες εφαρμογές υπάρχουν στο εξωτερικό που μεταφέρουν τις ασκήσεις σε ηλεκτρονική μορφή, κάποιες από αυτές είναι αρκετά εξειδικευμένες, και σχεδόν όλες παρουσιάζουν τις ασκήσεις σε αγγλική μορφή και πολλές από αυτές προσφέρονται μόνο σε PC και όχι σε περιβάλλοντα Android ή iOS. Επομένως υπάρχει μια έλλειψη για εφαρμογές που προσφέρουν δραστηριότητες σε ελληνική μορφή και σε περιβάλλον Android, κάτι που η εφαρμογή που υλοποιήθηκε σαν μέρος της διπλωματικής εργασίας προσπαθεί να καλύψει.

1.1.1 Συνεισφορά

1. Μελετήθηκαν ήδη υπάρχουσες εφαρμογές του εξωτερικού, αλλά και του εσωτερικού, και παρατηρήθηκαν οι υλοποιήσεις τους και οι τυχόν ελλείψεις που έχουν.
2. Υπήρξε συνάντηση με την κ. Έλλη Ζούπα, νευροψυχολόγου του Κέντρου Ημέρας Άνοιας Λάρισας- ΕΠΑΨΥ, η οποία μού παρέθεσε τις κατηγορίες των δραστηριοτήτων που χρησιμοποιείται στο κέντρο, ποιες είναι οι βασικές δραστηριότητες που ζητούνται κυρίως και επιθεώρησε την τελική εφαρμογή.
3. Μέσω της παρατήρησής παρόμοιων εφαρμογών και των κατηγοριών που συζητήθηκαν στην συνάντηση που προαναφέρθηκε, δημιουργήθηκε η εφαρμογή που παρουσιάζεται σε μεταγενέστερο κεφάλαιο.

1.2 Οργάνωση του Τόμου

Στο Κεφάλαιο 2 γίνεται μια εισαγωγή στις νευρογνωστικές διαταραχές, αναφέρονται κάποιες από τις πιο συνηθισμένες διαταραχές, τις περιοχές που προσβάλουν, τα προβλήματα που δημιουργούνται στους πάσχοντες και τις κατηγορίες ασκήσεων που χρησιμοποιούν οι θεραπευτές για τα άτομα αυτά.

Στο **Error! Reference source not found.** αναφέρονται τα μαθησιακά παιχνίδια και τι καθιστά ένα παιχνίδι μαθησιακό, καθώς και οι τρόποι μάθησης οι οποίοι βασίζονται σε πιο ενεργή προσέγγιση.

Στο **Error! Reference source not found.** παρουσιάζονται εφαρμογές που απευθύνονται στα άτομα με νευρογνωστικές διαταραχές, σε ποιες κατηγορίες θεραπείας στοχεύουν και τα τυχόν πλεονεκτήματα και μειονεκτήματα που έχουν.

Τέλος, στο **Error! Reference source not found.** παρουσιάζεται η ανάπτυξη και ο σχεδιασμός της εφαρμογής, από την αρχική υλοποίηση του κορμού της μέχρι τις δραστηριότητες που περιέχει και τον τρόπο που δημιουργήθηκε η κάθε δραστηριότητα.

Κεφάλαιο 2 Νευρογνωστικές Διαταραχές

2.1 Ορισμός των Νευρογνωστικών Διαταραχών

Οι νευρογνωστικές διαταραχές αντιπροσωπεύουν την υποκείμενη εγκεφαλική παθολογία που οδηγεί σε μείωση των πνευματικών ικανοτήτων[1]. Τα κλινικά συμπτώματα των νευρογνωστικών διαταραχών χωρίζονται σε δύο κατηγορίες, τις γνωστικές και τις νευροψυχιατρικές. Οι γνωστικές διαταραχές περιλαμβάνουν συμπτώματα όπως βλάβες στη μνήμη, τη γλώσσα, τον προσανατολισμό, την αναγνώριση, και εκτελεστικές λειτουργίες, ενώ τα υπόλοιπα συμπτώματα περιλαμβάνονται στις νευροψυχιατρικές διαταραχές[2].

Οι νευρογνωστικές διαταραχές δεν αφορούν αναπτυξιακές διαταραχές ή ψυχικές ασθένειες. Ο λόγος που υπάρχει περίπτωση κάποιος να τις συνδέει λανθασμένα, έγκειται στο γεγονός ότι παρουσιάζουν παρόμοια συμπτώματα με κάποιες σοβαρές ψυχικές ασθένειες όπως η σχιζοφρένεια, η κατάθλιψη και η ψύχωση[2].

Για την εμφάνισή των διαταραχών αυτών, υπάρχουν πολλοί παράγοντες που έχουν σαν αποτέλεσμα την εγκεφαλική βλάβη σε περιοχές του εγκεφάλου. Ενδεικτικά, κάποιες από τις περιοχές που προσβάλλουν είναι η περιοχή που επηρεάζουν τη μάθηση και τη μνήμη, τον προγραμματισμό και τη λήψη αποφάσεων, την ικανότητα χρήσης και κατανόησης της γλώσσας και της ικανότητας δράσης μέσα στα κοινωνικά πρότυπα, όπως η προσωπική περιποίηση, η ενσυναίσθηση ή ακόμα και η εκτέλεση καθημερινών δραστηριοτήτων[1]. Η διάγνωση των συμπτωμάτων μπορεί να γίνει από έναν νευροψυχολόγο και η διαχείριση της πάθησης μπορεί να κατευθυνθεί από έναν νευρολόγο ή γηριατρικό ψυχίατρο μετά από κλινική και απεικονιστική συνεκτίμηση των ευρημάτων[1].

2.2 Κατηγοριοποίηση Νευρογνωστικών Διαταραχών

Η ονοματολογία των νευροψυχιατρικών διαταραχών έχει μια αμφιλεγόμενη ιστορία, με περιοδικές προσπάθειες να επιτευχθεί συνοχή σε αυτό το ποικίλο και ανόμοιο πεδίο. Ως οργανισμός με επιρροή σε αυτόν τον τομέα, η Αμερικανική Ψυχιατρική Εταιρεία

προσπάθησε να δημοσιεύσει ένα γλωσσάρι για τις ψυχικές διαταραχές το 1952 ως την πρώτη έκδοση του Diagnostic and Statistical Manual of Mental Disorders (DSM-I). Αυτό το εγχειρίδιο έχει υποστεί έκτοτε πολλές αναθεωρήσεις, με την τελευταία θεώρηση (DSM-V) να ξεκινά το 1999 και να δημοσιεύεται, μετά από πολλές μελέτες και αναθεωρήσεις, το 2014[3].

Ως το επίσημο σύστημα ταξινόμησης της Αμερικανικής Ψυχιατρικής Εταιρείας, τα άτομα στα οποία απευθύνεται κυρίως το DSM είναι επαγγελματίες ψυχικής υγείας που εδρεύουν στις ΗΠΑ, οι οποίοι το χρησιμοποιούν κυρίως για την διάγνωση των ασθενών τους. Το DSM χρησιμοποιείται επίσης εκτενώς από ψυχιατρικούς ερευνητές για κριτήρια επιλογής συμμετεχόντων, μέτρα αποτελέσματος και αξιόπιστη επικοινωνία της εργασίας τους. Η χρήση του DSM, ωστόσο, ξεπερνά τα επαγγελματικά και εθνικά όρια, με ευρεία χρήση από κλινικούς γιατρούς και ερευνητές σε ποικίλα περιβάλλοντα διεθνώς[3].

2.3 Κατηγορίες Νευρογνωστικών Διαταραχών

Το σύμπλεγμα των νευρογνωστικών διαταραχών περιέχει τρία σύνδρομα, το καθένα με μια σειρά πιθανών αιτιολογιών: το παραλήρημα, την ήπια νευρογνωστική διαταραχή και την μείζονα νευρογνωστική διαταραχή.

2.3.1 Παραλήρημα

Το παραλήρημα σαν διαταραχή χαρακτηρίζεται ως η διαταραχή της προσοχής, η οποία δυσκολεύει το άτομο να κατευθύνει, να διατηρήσει και να αλλάξει την εστίασή του. Ως εκ τούτου, το άτομο είναι πιθανό να έχει μειωμένο προσανατολισμό στο περιβάλλον τους ή ακόμα και στον ίδιο τους τον εαυτό. Το παραλήρημα έχει μερικές φορές αναφερθεί ως «μειωμένο επίπεδο συνείδησης», αν και η διαταραχή της επίγνωσης είναι πιο ακριβής περιγραφή. Η διαταραχή της επίγνωσης τείνει να αναπτύσσεται από ώρες έως ημέρες και συνήθως παρουσιάζει διακυμάνσεις κατά τη διάρκεια της ημέρας, ενώ παρατηρείται συχνά επιδείνωση κατά τις βραδινές ώρες. Το παραλήρημα μπορεί να προκληθεί από μια υποκείμενη ιατρική κατάσταση, χρήση ή στέρηση ουσιών, έκθεση σε τοξίνες ή οποιονδήποτε συνδυασμό αυτών των παραγόντων. Οι ασθενείς μπορεί να είναι υπερκινητικοί, υποκινητικοί ή να έχουν μικτό επίπεδο δραστηριότητας[3].

2.3.2 Ήπια Νευρογνωστική Διαταραχή

Το DSM-5 περιγράφει το επίπεδο της γνωστικής έκπτωσης στην ήπια νευρογνωστική διαταραχή ως «μέτριο», αφήνοντας στον διαγνωστικό να κάνει την τελική κρίση για τη σοβαρότητα. Το DSM-5 δεν προσδιορίζει ποιες ή πόσες δοκιμές θα πρέπει να χορηγούνται ανά γνωστικό τομέα. Ελλείψει επίσημης νευροψυχολογικής αξιολόγησης, ο κλινικός ιατρός μπορεί να βασιστεί σε αξιολογήσεις, αλλά η αντικειμενική επίδειξη των γνωστικών ελλειμμάτων είναι απαραίτητη. Στην πραγματικότητα, επειδή η ήπια νευρογνωστική διαταραχή πρέπει να διακρίνεται τόσο από τη φυσιολογική γνωστική γήρανση όσο και από τη μείζονα νευρογνωστική διαταραχή (ή άνοια), απαιτείται ακόμη μεγαλύτερη εξάρτηση από τη νευροψυχολογική αξιολόγηση στην ήπια παρά στη μείζονα νευρογνωστική διαταραχή. Σειριακές αξιολογήσεις μπορεί να είναι απαραίτητες για την τεκμηρίωση της παρακμής, αλλά τα αποτελέσματα πρέπει να ερμηνεύονται προσεκτικά εν όψει των επιπτώσεων της πρακτικής, της μεταβλητής αξιοπιστίας του τεστ-επανεξέτασης και της έλλειψης κανονιστικών δεδομένων για τη γνωστική έκπτωση[3].

2.3.3 Μείζονα Νευρογνωστική Διαταραχή

Ο προσδιορισμός της «σημαντικής» γνωστικής έκπτωσης, δηλαδή της έκπτωσης επαρκής για τη διάγνωση της μείζονος νευρογνωστικής διαταραχής, βασίζεται στην ανησυχία που εκφράζεται από ένα άτομο ή από έναν πληροφοριοδότη ή έναν κλινικό ιατρό που γνωρίζει το άτομο, καθώς και στην επίδειξη ουσιαστικά μειωμένης απόδοσης σε αντικειμενικό γνωστικό μέτρο. Μια γνωστική ανησυχία μπορεί να μην εκφράζεται αυθόρμητα και μπορεί να χρειαστεί να προκληθεί από προσεκτική ερώτηση του ασθενούς ή/και των σημαντικών άλλων. Η απαίτηση ενός αντικειμενικού μέτρου ικανοποιείται καλύτερα με την επίσημη νευροψυχολογική αξιολόγηση, με την απόδοση να συγκρίνεται με κανονιστικά δεδομένα κατάλληλα για την ηλικία, το μορφωτικό επίπεδο και το πολιτιστικό-γλωσσικό υπόβαθρο του ασθενούς. Όπως και στην ήπια νευρογνωστική διαταραχή, οι ασθενείς για τους οποίους δεν είναι εφικτός ο νευροψυχολογικός έλεγχος ή δεν υπάρχουν κατάλληλοι κανόνες, μπορούν να υποβληθούν σε μια σύντομη αξιολόγηση δίπλα στο κρεβάτι από τον κλινικό γιατρό για την παροχή των αντικειμενικών δεδομένων που είναι απαραίτητα για τη διάγνωση. Η επαρκής ερμηνεία της απόδοσης του τεστ είναι απαραίτητη και μπορεί να υποβοηθηθεί από προηγούμενη χορήγηση του ίδιου τεστ, ώστε να μπορεί να αξιολογηθεί η πτώση[3].

2.4 Συνηθισμένες νευρογνωστικές διαταραχές

Παρόλο που η πλειονότητα των ατόμων που πάσχουν από κάποια μορφή νευρογνωστικών διαταραχών πάσχουν από την νόσο του Alzheimer, υπάρχουν στην πραγματικότητα συνολικά οκτώ, σύμφωνα με το DSM-V, ιατρικές καταστάσεις που επηρεάζουν με παρόμοιο τρόπο τις ψυχικές λειτουργίες, ενώ δεν αποκλείεται κάποιες φορές να επηρεάζεται και το κινητήριο σύστημα[1]. Από αυτές τις οκτώ, οι συνηθέστερες καταστάσεις περιλαμβάνουν:

2.4.1 Νόσος Alzheimer

Η νόσος Alzheimer, γνωστή απλά και ως Alzheimer, είναι μια χρόνια νευροεκφυλιστική νόσος, η οποία αρχίζει με αργούς ρυθμούς και επιταχύνει με τα χρόνια[1]. Η νόσος Alzheimer είναι μη θεραπεύσιμη και θανατηφόρα αποτελώντας την αιτία του 60 με 70 τοις εκατό των περιπτώσεων άνοιας. Η νόσος Alzheimer περιεγράφηκε αρχικά από τον Γερμανό ψυχίατρο και νευροπαθολόγος Aloysius Alzheimer το 1906. Το πιο κοινό πρόωρο σύμπτωμα της νόσου είναι η απώλεια βραχυπρόθεσμης μνήμης ενώ, καθώς η ασθένεια προχωρά, τα συμπτώματα μπορεί να περιλαμβάνουν προβλήματα στη γλώσσα, στον προσανατολισμό, στις διακυμάνσεις της διάθεσης, στην απώλεια κινήτρων, στη μη αυτοεξυπηρέτηση και σε θέματα συμπεριφοράς[5].

2.4.2 Μετωπιοκροταφικός Λοβιακός Εκφυλισμός

Ο Μετωπιοκροταφικός λοβιακός εκφυλισμός (ΜΛΕ-FTLD) είναι ένα κλινικά και παθολογικά ετερογενές σύνδρομο, που χαρακτηρίζεται από προοδευτική μείωση της συμπεριφοράς ή της γλώσσας που σχετίζεται με τον εκφυλισμό του μετωπιαίου και πρόσθιου κροταφικού λοβού[6]. Ενώ οι αρχικές περιπτώσεις περιεγράφηκαν στις αρχές του 20ου αιώνα, ο FTLD έχει μόλις πρόσφατα εκτιμηθεί ως η κύρια αιτία άνοιας, ιδιαίτερα σε ασθενείς που εμφανίζονται πριν από την ηλικία των 65 ετών. Τρεις διακριτές κλινικές παραλλαγές του FTLD έχουν περιγραφεί: η μετωποκροταφική άνοια παραλλαγής συμπεριφοράς, που χαρακτηρίζεται από αλλαγές στη συμπεριφορά και την προσωπικότητα σε συνδυασμό με τον μετωπιαίο-κυρίαρχο εκφυλισμό του φλοιού, η σημασιολογική άνοια, ένα σύνδρομο προοδευτικής απώλειας γνώσης για λέξεις και αντικείμενα που σχετίζονται με απώλεια πρόσθιας κροταφικής νευρώνας και η προοδευτική μη ρευστή αφασία, που

χαρακτηρίζεται από επίπονη γλωσσική απόδοση, απώλεια γραμματικής και κινητικής ομιλίας στο πλαίσιο της ατροφίας του αριστερού περισylvιακού φλοιού[6].

2.4.3 HIV Λοίμωξη

Οι ιοί της ανθρώπινης ανοσοανεπάρκειας (HIV) είναι δύο είδη Lentivirus (υποομάδα ρετροϊών) που μολύνουν τον άνθρωπο. Με την πάροδο του χρόνου, προκαλούν σύνδρομο επίκτητης ανοσοανεπάρκειας (AIDS), μια κατάσταση κατά την οποία η προοδευτική ανεπάρκεια του ανοσοποιητικού συστήματος επιτρέπει την ανάπτυξη απειλητικών για τη ζωή ευκαιριακών λοιμώξεων και καρκίνων. Χωρίς θεραπεία, ο μέσος χρόνος επιβίωσης μετά τη μόλυνση με HIV εκτιμάται ότι είναι 9 έως 11 χρόνια, ανάλογα με τον υποτύπο HIV[7].

2.4.4 Νόσος του Σώματος Lewy

Η άνοια με σωμάτια Lewy (Lewy Body Disease Dementia) είναι ένας τύπος άνοιας που χαρακτηρίζεται από αλλαγές στον ύπνο, τη συμπεριφορά, τη γνωστική λειτουργία, την κίνηση και τη ρύθμιση των αυτόματων σωματικών λειτουργιών. Η απώλεια μνήμης δεν είναι πάντα ένα πρώιμο σύμπτωμα. Η ασθένεια επιδεινώνεται με την πάροδο του χρόνου και συνήθως διαγιγνώσκεται όταν η γνωστική εξασθένηση παρεμβαίνει στη φυσιολογική καθημερινή λειτουργία. Μαζί με την άνοια της νόσου του Πάρκινσον, η LBD είναι μία από τις δύο άνοιες των σωματίων Lewy. Είναι μια κοινή μορφή άνοιας, και η ασθένεια περιεγράφηκε για πρώτη φορά από τον Kenji Kosaka το 1976[8].

2.4.5 Νόσος του Πάρκινσον

Η νόσος του Πάρκινσον είναι μια μακροχρόνια διαταραχή του νευρικού συστήματος και επηρεάζει κυρίως το κινητικό σύστημα. Τα συμπτώματα συνήθως εμφανίζονται αργά και καθώς η νόσος επιδεινώνεται, τα μη κινητικά συμπτώματα γίνονται πιο συχνά. Τα πιο εμφανή πρώιμα συμπτώματα είναι ο τρόμος, η ακαμψία, η βραδύτητα της κίνησης και η δυσκολία στο περπάτημα. Γνωστικά και συμπεριφορικά προβλήματα μπορεί επίσης να εμφανιστούν με κατάθλιψη, άγχος και απάθεια που εμφανίζονται σε πολλά άτομα με Πάρκινσον. Η άνοια από τη νόσο του Πάρκινσον γίνεται συχνή στα προχωρημένα στάδια της νόσου. Όσοι πάσχουν από Πάρκινσον μπορεί επίσης να έχουν προβλήματα με τον ύπνο και τα αισθητηριακά τους συστήματα. Τα κινητικά συμπτώματα της νόσου

προκύπτουν από τον θάνατο των κυττάρων στη μέλαινα ουσία, μια περιοχή του μεσαίου εγκεφάλου, που οδηγεί σε έλλειψη ντοπαμίνης. Η αιτία αυτού του κυτταρικού θανάτου είναι ελάχιστα κατανοητή, αλλά περιλαμβάνει τη συσσώρευση λανθασμένων πρωτεϊνών σε σωματίδια Lewy στους νευρώνες. Συλλογικά, τα κύρια κινητικά συμπτώματα είναι επίσης γνωστά ως παρκινσονικό σύνδρομο[9].

2.4.6 Νόσος Prion

Τα Prion είναι πρωτεΐνες που έχουν διπλωθεί λανθασμένα και έχουν την ικανότητα να μεταδίδουν το λανθασμένο σχήμα τους σε κανονικές παραλλαγές της ίδιας πρωτεΐνης. Χαρακτηρίζουν αρκετές θανατηφόρες και μεταδοτικές νευροεκφυλιστικές ασθένειες στον άνθρωπο και σε πολλά άλλα ζώα. Δεν είναι γνωστό τι προκαλεί την κακή αναδίπλωση της κανονικής πρωτεΐνης, αλλά η μη φυσιολογική τρισδιάστατη δομή είναι ύποπτη ότι προσδίδει μολυσματικές ιδιότητες, καταρρέοντας τα γειτονικά μόρια πρωτεΐνης στο ίδιο σχήμα. Η λέξη πρίον προέρχεται από το "πρωτεϊνικό μολυσματικό σωματίδιο". Ο υποτιθέμενος ρόλος μιας πρωτεΐνης ως μολυσματικού παράγοντα έρχεται σε αντίθεση με όλους τους άλλους γνωστούς μολυσματικούς παράγοντες όπως ιοειδή, ιούς, βακτήρια, μύκητες και παράσιτα, τα οποία περιέχουν όλα νουκλεϊκά οξέα (DNA, RNA ή και τα δύο)[10].

Εκτός από τις παραπάνω κατηγορίες, που είναι οι σημαντικότερες και οι πιο κοινές, νευρογνωστικές διαταραχές μπορούν, σύμφωνα με το DSM-V, να εμφανιστούν από χρήση ουσιών ή/και φαρμάκων, μετα από κάποιου είδους τραυματική εγκεφαλική βλάβη, από αγγειακή νόσο, ενώ κάποιες φορές η αιτία είναι απροσδιόριστη[2].

2.5 Τομείς Ασκήσεων για τις Νευρογνωστικές Διαταραχές

Καθώς οι νευρογνωστικές διαταραχές δημιουργούν προβλήματα σε συγκεκριμένα σημεία του εγκεφάλου και πλήττουν συγκεκριμένες λειτουργίες, οι ασκήσεις που χρησιμοποιούν οι ειδικοί για την πρόληψη και την καταπολέμηση των διαταραχών εντείνεται στους εξής τομείς[4]:

1. Εκτελεστική λειτουργία, όπου εξετάζονται, εκτός των άλλων, η λήψη απόφασης, η μνήμη και η απάντηση σε σχόλια.

2. Η προσοχή και οι υποκατηγορίες της, όπως η σύνθετη προσοχή, η διαρκής προσοχή, η επιλεκτική προσοχή και η ταχύτητα επεξεργασίας.
3. Η κατηγορία της μάθησης και της μνήμης, με τις υποκατηγορίες της ανάκλησης, την μνήμη αναγνώρισης και την βραχυπρόθεσμη και μακρόχρονη μνήμη.
4. Την κατηγορία της γλώσσας, με τις υποκατηγορίες της οι οποίες είναι η ονομασία αντικειμένου, η εύρεση λέξεων, η ευφράδεια και η γραμματική και σύνταξη.
5. Η κατηγορία των κινήσεων με την αντιληπτική και κινητική λειτουργία, την οπτική αντίληψη και τον αντιληπτικό και κινητικό συντονισμό να είναι οι σημαντικότερες υποκατηγορίες της.
6. Τέλος τον κοινωνικό τομέα, στον οποίο εμβαθύνουν στην κοινωνική γνώση, την αναγνώριση συναισθημάτων και την διορατικότητα.

Κεφάλαιο 3 Εκπαιδευτικά Παιχνίδια και Τρόποι Μάθησης

3.1 Εκπαιδευτικά Παιχνίδια

Τα εκπαιδευτικά παιχνίδια έχουν γίνει όλο και περισσότερα και πιο διαδεδομένα τα τελευταία χρόνια. Σε γενικό βαθμό, υπάρχει αφθονία από παιχνίδια σχεδιασμένα για εκπαιδευτικό σκοπό. Υπάρχουν επίσης πολλά ψυχαγωγικά παιχνίδια που χρησιμοποιούνται για προπόνηση ή εκπαίδευση[11].

Τα εκπαιδευτικά παιχνίδια, σαν ορισμός, θεωρούνται τα σοβαρά παιχνίδια που χρησιμοποιούνται ειδικά για την εκπαίδευση. Τα σοβαρά παιχνίδια, με τη σειρά τους, είναι μια έννοια με πολλούς ορισμούς. Με μια ευρεία έννοια ο όρος σοβαρά παιχνίδια αναφέρεται στην ιδέα της χρήσης παιχνιδιών για σκοπούς πέρα από καθαρούς ψυχαγωγία[11].

Υπάρχουν πολλοί ορισμοί των PC games, όπως υπάρχουν επίσης ευρέως χρησιμοποιούμενες γενικές αντιλήψεις. Σε έρευνα που πραγματοποιήθηκε, στην οποία έγινε σύνοψη από οκτώ διαφορετικούς ορισμούς και, εστιάζοντας σε διαφορετικά στοιχεία, υπήρξε ως αποτέλεσμα οι ερευνητές να καταλήγουν σε έναν αρκετά συμπυκνωμένο ορισμό: «Ένα παιχνίδι είναι ένα σύστημα στο οποίο οι παίκτες συμμετέχουν σε μία τεχνητή σύγκρουση, που ορίζεται από κανόνες, η οποία καταλήγει σε ένα μετρήσιμο αποτέλεσμα.»[12]. Ο ορισμός είναι περιορισμένος, ιδίως όσον αφορά το μετρήσιμο αποτέλεσμα και, σε πολλές περιπτώσεις, ένας πιο ανοιχτός χαρακτηρισμός, ο οποίος συμπεριλαμβάνει κανόνες, στόχους, αποτελέσματα, συγκρούσεις, ανταγωνισμό, πρόκληση, αντίθεση, αλληλεπίδραση και αναπαράσταση μιας ιστορίας ιστορία είναι χρησιμότερος. Η έννοια της συνεργασίας είναι επίσης μια σημαντική πτυχή των παιχνιδιών και υπάρχουν πράγματι πολλά παιχνίδια που δεν εστιάζουν σε ανταγωνιστικά στοιχεία ή στη νίκη[11].

Τα σοβαρά παιχνίδια, τα οποία υπάρχουν περισσότερο σαν όρος-ομπρέλα, περιλαμβάνουν διαφορετικούς τύπους εκπαιδευτικών παιχνιδιών καθώς και παιχνιδιών για άλλους σκοπούς όπως αποκατάσταση, μάρκετινγκ και κοινωνική βελτίωση. Ο όρος

σοβαρά παιχνίδια ορίζεται ως εξής: «Ένα σοβαρό παιχνίδι είναι ένας διανοητικός διαγωνισμός, ο οποίος παίζεται με την βοήθεια υπολογιστή σύμφωνα με συγκεκριμένους κανόνες, και χρησιμοποιεί την ψυχαγωγία για να προωθήσει την κυβερνητική ή εταιρική κατάρτιση, εκπαίδευση, υγεία, δημόσια πολιτική και στρατηγικούς επικοινωνιακούς στόχους»[14]. Αυτός ο ορισμός προσπαθεί να είναι χωρίς αποκλεισμούς, αλλά η εστίασή του στην ψυχαγωγία μερικές φορές έρχεται σε σύγκρουση με αυτό που πλάσάζεται στην αγορά ως σοβαρά παιχνίδια.

Τα εκπαιδευτικά παιχνίδια έχουν σχεδιαστεί ειδικά με σκοπό να διδάξουν τον κόσμο σχετικά με ένα συγκεκριμένο θέμα, να επεκτείνουν τις έννοιες, να ενισχύσουν την ανάπτυξη και να τους βοηθήσει στην εκμάθηση μιας καινούργιας δεξιότητας. Τα τελευταία χρόνια, τα εκπαιδευτικά παιχνίδια έχουν χρησιμοποιηθεί ως μια καινοτόμος εκπαιδευτική στρατηγική προκειμένου να επιτευχθεί πιο αποτελεσματική μάθηση σε υψηλότερα επίπεδα διάφορων τομέων γνώσης, όπως παραδείγματος χάρη τα μαθηματικά, η γλώσσα, οι επιχειρήσεις, η υγεία και η ψυχαγωγία μεταξύ άλλων[15].

Επιπλέον, πιστεύεται ότι έχουν ως αποτέλεσμα ένα ευρύ φάσμα πλεονεκτημάτων, όπως αύξηση της μαθησιακής αποτελεσματικότητας, του ενδιαφέροντος και των κινήτρων καθώς και μείωση του χρόνου εκπαίδευσης και του φόρτου του εκπαιδευτή. Θεωρείται ότι είναι ένα διασκεδαστικό και ασφαλές περιβάλλον, στο οποίο οι μαθητές μπορούν να δοκιμάσουν εναλλακτικές σκέψεις και να δουν τις συνέπειες, μαθαίνοντας από τις δικά τους λάθη και πρακτικές εμπειρίες. Με αυτόν τον τρόπο, αναμένεται να είναι μία αποτελεσματική και αποδοτική εκπαιδευτική στρατηγική για τη διδασκαλία και τη μάθηση. Παρόλα αυτά, οι αξιώσεις αυτές δεν έχουν τεκμηριωθεί αυστηρά και για να τα χρησιμοποιηθούν τα εκπαιδευτικά παιχνίδια αποτελεσματικά είναι απαραίτητο να αξιολογηθούν συστηματικά προκειμένου να αποκτηθούν έγκυρα στοιχεία για τον αντίκτυπό τους[15].

3.2 Δημιουργία ενός Εκπαιδευτικού Παιχνιδιού

Για να δημιουργηθεί ένα μαθησιακό παιχνίδι το οποίο είναι διδακτικό αλλά και ευχάριστο θεωρείται αρκετά απαιτητικό. Σε έρευνα που πραγματοποιήθηκε, οι ερευνητές αποφάνθηκαν ότι ένα πλαίσιο, το οποίο αποτελείται από 3 μέρη και μπορεί να βοηθήσει

τους σχεδιαστές να εντοπίσουν και να αναλύσουν το σχέδιο και τις επιλογές που πρέπει να γίνουν κατά τη δημιουργία ενός εκπαιδευτικού παιχνιδιού[16].

Το πρώτο συστατικό του πλαισίου είναι ο καθορισμός των μαθησιακών στόχων. Ένα ξεκάθαρος προσδιορισμός των μαθησιακών στόχων βοηθά τους σχεδιαστές να βεβαιωθούν ότι το παιχνίδι που θέλουν να δημιουργήσουν πληροί και συνεκτικό σύνολο εκπαιδευτικών στόχων. Τους βοηθά να αποφύγουν δημιουργώντας παιχνίδια που απευθύνονται μόνο σε απλές περιπτώσεις, σε στοχευμένες γνωστικές δεξιότητες ή παιχνίδια που προϋποθέτουν προηγουμένως γνώση που δεν είναι πιθανό να έχει ο μαθητής/παίκτης. Επιπλέον, βοηθάει έναν σχεδιαστή να δημιουργήσει ένα παιχνίδι που είναι πραγματικά αποτελεσματικό εκπαιδευτικά με το να θέσει σαφές εκπαιδευτικούς στόχους που θέλει να θέσει[16].

Οι εκπαιδευτικοί αυτοί στόχοι θέτονται με τον δημιουργό να χρειάζεται να δώσει απάντηση σε τρεις ερωτήσεις:

1. Τι γνώσεις ή δεξιότητες οι παίκτες πρέπει να έχουν πριν ξεκινήσουν το παιχνίδι.
2. Τι είδους γνώσεις ή δεξιότητες που αναμένεται να μάθουν από το παιχνίδι οι παίκτες.
3. Ποιες γνώσεις και δεξιότητες μπορεί να μάθουν οι οποίες ξεπερνούν αυτό που συνάντησαν στο παιχνίδι[16].

Το δεύτερο συστατικό του πλαισίου που προτάθηκε είναι ένα πλαίσιο από μόνο του, ονόματι MDA (Mechanics, Dynamics, Aesthetics ή αλλιώς Μηχανισμοί, Δυναμική και Αισθητική). Το πλαίσιο αυτό βοηθά τους αναλυτές και τους σχεδιαστές να σκέφτονται τα παιχνίδια με τρία αλληλεξαρτώμενα στρώματα. Οι Μηχανισμοί του παιχνιδιού είναι τα βασικά συστατικά από τα οποία το παιχνίδι χτίζεται: κανόνες, στόχους, βασικές κινήσεις και επιλογές ελέγχου που είναι διαθέσιμες στους παίκτες. Η Δυναμική του παιχνιδιού είναι το αποτέλεσμα της εφαρμογή των μηχανισμών του παιχνιδιού σε συνδυασμό με τις επιλογές του παίκτη κατά τη διάρκεια του παιχνιδιού και είναι στο χέρι του σχεδιαστή να εφεύρει τους όρους και έννοιες που χρειάζονται για τον χαρακτηρισμό της δυναμικής ενός παιχνιδιού. Τέλος, η Αισθητική ενός παιχνιδιού, αποτυπώνει την εμπειρία του παίκτη και τα συναισθήματα που παιχνίδι είναι σχεδιασμένο να προκαλεί. Το πλαίσιο παρέχει μια ταξινόμηση στοιχείων που στοχεύουν συνήθως τα παιχνίδια, και περιλαμβάνονται 8 στοιχεία:

1. Αίσθηση.
2. Φαντασία.
3. Αφήγηση.
4. Πρόκληση.
5. Συναναστροφή.
6. Ανακάλυψη.
7. Έκφραση.
8. Υποβολή.

Αυτή η ταξινόμηση παρέχει έναν συγκεκριμένο και διακριτικό τρόπο ομιλίας, τις απολαύσεις που προκύπτουν από τα παιχνίδια, σε μια προσπάθεια να προχωρήσει πέρα από αόριστους όρους όπως «διασκέδαση» και «παιχνίδι»[16].

Το τρίτο συστατικό του πλαισίου που προτείνεται στην έρευνά τους, είναι το χρήση αρχών που βασίζονται στην έρευνα του διδακτικού σχεδιασμού. Υπάρχουν πολλές τέτοιες συλλογές και γίνεται ένας όλο και πιο δημοφιλής τρόπος με τον οποίο οι ερευνητές συνοψίζουν και ανακοινώνουν τα αποτελέσματα της έρευνάς τους σε εκπαιδευτικούς σχεδιαστές και εκπαιδευτές. Βασιζόμενοι σε μια σειρά από αρχές που έχουν δημοσιευθεί ευρέως ως τρόπος της εισαγωγής των μαθησιακών επιστημών στο σχεδιασμό του εκπαιδευτικά παιχνίδια, ανακάλυψαν ότι μια βασική υπόθεση είναι ότι οι αρχές του εκπαιδευτικού σχεδιασμού που έχουν εγκατεστημένοι σε άλλους τύπους μαθησιακών περιβαλλόντων μπορεί να μεταφερθεί στον σχεδιασμό εκπαιδευτικών παιχνιδιών και να βοηθήσει στη δημιουργία παιχνιδιών που να είναι πιο εκπαιδευτικά[16].

3.3 Τρόποι Μάθησης.

Οι γνωστικοί ψυχολόγοι μελετούν τη μνήμη, την προσοχή και την αντίληψη. Με το να αποκτούν μια κατανόηση αυτών των βασικών διαδικασιών, είναι σε θέση να αναπτύξουν στρατηγικές διδασκαλίας και μάθησης που μεγιστοποιούν τη διατήρηση και τη μεταφορά γνώσεων. Μετά από δεκαετίες έρευνας, οι γνωστικοί ψυχολόγοι έχουν εντοπίσει έξι στρατηγικές με σημαντικά πειραματικά στοιχεία που υποστηρίζουν τη χρήση τους:

1. Η πρώτη από τις έξι στρατηγικές είναι η πρακτική σε απόσταση, η οποία είναι μια στρατηγική στην οποία οι συνεδρίες διαχωρίζονται χρονικά αντί να συγκεντρώνεται σε μία μόνο συνεδρία.
2. Η παρεμβολή συχνά περιλαμβάνει κάποια φυσική απόσταση, αλλά προσθέτει ένα επιπλέον όφελος από την αλλαγή της σειράς με την οποία τα στοιχεία μελετώνται ή επανεξετάζονται.
3. Η επεξεργασία, ή πιο συγκεκριμένα η αναλυτική ανάκριση, περιλαμβάνει την υποβολή ερωτήσεων «πώς» και «γιατί».
4. Η χρήση παραδειγμάτων, στα οποία δίνονται στους μαθητές ποικίλα παραδείγματα χρησιμοποιώντας συγκεκριμένες πληροφορίες που είναι πιο κατανοητές και από τα αφηρημένες έννοιες.
5. Η διπλή κωδικοποίηση, στην οποία συνδυάζονται λεκτικές και οπτικές πληροφορίες για τη βελτίωση της κωδικοποίησης.
6. Η πρακτική ανάκτησης αυξάνει τη διατήρηση πέρα από την επαναλαμβανόμενη μελέτη πληροφοριών[17].

Αυτές οι στρατηγικές είναι πιο αποτελεσματικές σαν τρόποι μάθησης με φυσική παρουσία κάποιου ειδικού, εκπαιδευτικού ή ιατρού, ο οποίος καθοδηγεί και μεταλαμπαδεύει τις γνώσεις του με την βοήθεια των στρατηγικών αυτών. Όμως, υπάρχουν και άλλες στρατηγικές για την μεταφορά των γνώσεων, όπου οι ειδικοί συνεργάζονται με τα άτομα που διδάσκουν, εκπαιδεύουν ή φροντίζουν, ώστε τα άτομα αυτά να αποκτούν πιο ενεργητικό ρόλο στην απόκτηση γνώσεων. Δύο από τους βασικότερους τρόπους που χρησιμοποιούνται είναι η ενεργητική (και συνεργατική) μάθηση, και η διδασκαλία μέσω επίλυσης προβλημάτων.

3.3.1 Ενεργή και Συνεργατική Μάθηση

Ενεργή μάθηση ονομάζεται η διαδικασία συμμετοχής των μαθητών σε κάποια δραστηριότητα ενεργά που τους αναγκάζει να αναλογιστούν ιδέες και πώς χρησιμοποιούν αυτές τις ιδέες. Οι μαθητές χρειάζεται να αξιολογούν τον δικό τους βαθμό κατανόησης και ικανότητας χειρισμού εννοιών ή προβλημάτων σε έναν συγκεκριμένο κλάδο σε τακτικό βαθμό[18].

Η ενεργή μάθηση συνήθως συνδέεται με την μαθητοκεντρική διδασκαλία. Η μαθητοκεντρική διδασκαλία είναι μια διδακτική προσέγγιση στην οποία οι μαθητές

επηρεάζουν το περιεχόμενο, τις δραστηριότητες, το υλικό και τον ρυθμό μάθησης. Το μοντέλο αυτό τοποθετεί τον μαθητή στο κέντρο της διαδικασίας, ενώ ο εκπαιδευτικός υπάρχει με σκοπό να παρέχει στους μαθητές ευκαιρίες να μαθαίνουν ανεξάρτητα ο ένας από τον άλλο και να τους καθοδηγεί στις δεξιότητες που χρειάζονται για να το κάνουν αποτελεσματικά. Η προσέγγιση αυτή περιλαμβάνει διάφορες τεχνικές, με μερικές από αυτές να είναι ανάθεση προβλημάτων που απαιτούν κριτική ή δημιουργική σκέψη και τα οποία δεν μπορούν να λυθούν με το να ακολουθηθούν παραδείγματα κειμένου, να εμπνεύσει ο εκπαιδευτικός τους μαθητές να εμπλακούν σε προσομοιώσεις και παιχνίδια ρόλων και να χρησιμοποιήσει συνεργατική μάθηση. Μια σωστά εφαρμοσμένη μαθητοκεντρική προσέγγιση μπορεί να οδηγήσει σε αυξημένα κίνητρα για μάθηση, μεγαλύτερη διατήρηση του γνώση, βαθύτερη κατανόηση και πιο θετικές στάσεις προς το αντικείμενο που διδάσκεται[18]. Μερικοί τρόποι διαδραστικής μάθησης που ακολουθούν την μαθητοκεντρική προσέγγιση περιλαμβάνουν:

1. Μάθηση βάσει προβλημάτων ή περιπτώσεων.
2. Συνεργατική εργασία.
3. Στρατηγικές εννοιολογικής αλλαγής.
4. Μάθηση βάσει διερεύνησης.
5. Ανακαλυπτική μάθηση.
6. Εκμάθηση ενισχυμένη με τεχνολογία[19].

Επιπλέον, στην ίδια μελέτη[19], αναφέρονται τα βασικά ευρήματα που πρέπει να ενσωματωθούν ως προς την λήψη αποφάσεων σχετικά με τη φύση της διδασκαλίας σε οποιοδήποτε εκπαιδευτικό επίπεδο.

1. Η μάθηση περιλαμβάνει την ενεργό κατασκευή του νοήματος από τον μαθητή.
2. Το να μαθαίνεις γεγονότα και το να μαθαίνεις να κάνεις κάτι είναι δύο διαφορετικές διαδικασίες.
3. Μερικά πράγματα που μαθαίνονται είναι συγκεκριμένα για τον τομέα ή το πλαίσιο που διδάχτηκαν, ενώ άλλα πράγματα μεταφέρονται πιο εύκολα σε άλλους τομείς.
4. Τα άτομα είναι πιθανό να μάθουν περισσότερα όταν μαθαίνουν με άλλα άτομα παρά όταν μαθαίνουν μόνοι τους.
5. Η μάθηση με νόημα διευκολύνεται με την άρθρωση εξηγήσεων, είτε στον εαυτό σου, στους συνομηλίκους σου ή στους ίδιους τους δασκάλους[19].

Μια ακόμα κατηγορία που μπορεί να εφαρμοστεί παράλληλα με την ενεργή μάθηση είναι η συνεργατική μάθηση. Η συνεργατική μάθηση βασίζεται στην προϋπόθεση ότι η συνεργασία είναι πιο αποτελεσματική για την παραγωγή θετικότερων αποτελεσμάτων από τον ανταγωνισμό μεταξύ των μαθητών, όπως και στην ιδέα ότι η συνεργασία προάγει τις προσωπικές σχέσεις, βελτιώνει την κοινωνική υποστήριξη και ενισχύει την αυτοεκτίμηση.

3.3.2 Μάθηση με Βάση το Πρόβλημα.

Το πρώτο βήμα για να προσδιοριστεί μια εκπαιδευτική προσέγγιση αν λειτουργεί πρέπει πρώτα να διευκρινιστεί ποια είναι αυτή η προσέγγιση. Όμως, ενώ οι ερευνητές συμφωνούν στον γενικό ορισμό της μάθησης με βάση το πρόβλημα, η υλοποίηση της μάθησης διαφέρει αναλόγως το άτομο που την εφαρμόζει. Επομένως, λόγω της μεγάλης ποικιλίας στον συγκεκριμένο τρόπο μάθησης, η ανάλυση της αποτελεσματικότητάς του γίνεται πιο πολύπλοκη. Παρόλα αυτά, υπάρχει ένα αποδεκτό εύρημα που προκύπτει από τις έρευνες που έχουν γίνει, το οποίο είναι ότι ο συγκεκριμένος τρόπος μάθησης προσεγγίζεται θετικά από τους μαθητές[20].

3.3.2.1 Γενικά

Η επίλυση προβλημάτων μπορεί να οριστεί ως μια συμπεριφορική διαδικασία, η οποία είναι είτε φανερή είτε γνωστική όσον αφορά την φύση, η οποία καθιστά διαθέσιμη μια ποικιλία από δυνητικά αποτελεσματικές εναλλακτικές λύσεις αντίδρασης για την αντιμετώπιση της προβληματικής κατάστασης και αυξάνει την πιθανότητα επιλογής του η πιο αποτελεσματική απάντηση μεταξύ των διαφόρων εναλλακτικών. Όπως υπονοεί ο ορισμός, η επίλυση προβλημάτων θεωρείται ότι περικλείει δραστηριότητες που περιλαμβάνουν εναλλακτικές απαντήσεις και λήψεις αποφάσεων ή συμπεριφορά επιλογής. Σε ένα πραγματικό περιβάλλον, η πτυχή της λήψης αποφάσεων είναι ιδιαίτερα σημαντική, καθώς τα περισσότερα προβλήματα της ζωής έχουν περισσότερο από μία «σωστή» λύση, δηλαδή μπορούν να αντιμετωπιστούν επαρκώς με πολλούς και διαφορετικούς τρόπους, ορισμένοι από τους οποίους μπορεί να είναι πιο αποτελεσματικοί ή κατάλληλοι από άλλους. Σύμφωνα με τα παραπάνω, μπορεί επίσης να περιγράψει η επίλυση προβλημάτων ως διαδικασία μάθησης[21].

Μια αποτελεσματική απάντηση σε μία προβληματική κατάσταση μπορεί να οριστεί ως απόκριση ή μοτίβο απαντήσεων που αλλάζει την κατάσταση ώστε να μην είναι πλέον προβληματική στο άτομο, ενώ ταυτόχρονα μεγιστοποιεί τις θετικές συνέπειες και ελαχιστοποιεί τις αρνητικές. Οι συνέπειες αυτές, τόσο οι θετικές όσο και οι αρνητικές, αναφέρονται σε πιθανές, βραχυπρόθεσμες, μακροπρόθεσμες, προσωπικές και κοινωνικές επιπτώσεις της ανταπόκρισης του ατόμου στην κατάσταση[21].

3.3.2.2 Κλινικά

Ένα βασικό ερώτημα που προκύπτει είναι εάν, οι δεξιότητες για την επίλυση προβλημάτων στον γνωστικό και διαπροσωπικό τομέα μπορούν να διδαχθούν μέσω συνειδητής προσπάθειας και επαναλαμβανόμενης πρακτικής[22].

Στην αρχική τους εννοιοποίηση, θεωρήθηκε ότι η μάθηση μέσω επίλυσης προβλημάτων είναι μία πρόοδος πέντε σταδίων τα οποία είναι τα εξής:

1. Προσανατολισμός προβλημάτων ή «συνόλου»,
2. Ορισμός και διατύπωση προβλήματος ,
3. Δημιουργία εναλλακτικών λύσεων,
4. Λήψη αποφάσεων και
5. Εφαρμογή και επαλήθευση λύσης[23].

Μεταγενέστερες μελέτες όμως[24], προτείνουν τα εξής κύρια τμήματα για την μάθηση μέσω επίλυσης προβλημάτων:

1. Μια σειρά από ξεχωριστές γνωστικές δεξιότητες που είναι κρίσιμες για αποτελεσματική λειτουργία σε διαπροσωπικές καταστάσεις, οι οποίες περιλαμβάνουν την ικανότητα σκέψης πολλών επιλογών πριν την ενέργεια ή να γίνει μια εκτίμηση των πιθανών συνεπειών μιας πράξης.
2. Διαφορετικοί συνδυασμοί αυτών των δεξιοτήτων είναι σημαντικοί για την προσαρμογή κατά τη διάρκεια διαφορετικών φάσεων ανάπτυξης, όπως π.χ. πρώιμη και μέση παιδική ηλικία, εφηβεία κλπ.
3. Αυτές οι δεξιότητες, αν και γνωστικού χαρακτήρα, κατευθύνονται προς το διαπροσωπικό τομέα και είναι ψυχομετρικά διαφορετικές από αυτό που συνήθως υποθέτουμε σχετικά με την αξιολόγηση μέσω συμβατικών τεστ νοημοσύνης.

4. Μέσω ειδικά αναπτυγμένων μεθόδων εκπαίδευσης είναι δυνατό τα άτομα που στερούνται αυτές τις δεξιότητες να τις αποκτήσουν, με βελτιώσεις στη διαπροσωπική προσαρμογή[24].

Η αξιολόγηση των δραστηριοτήτων και των δεξιοτήτων που μπορούν να υιοθετηθούν και να οριστούν ξεχωριστά σε αυτόν τον τομέα έχει δημιουργήσει επαναλαμβανόμενες δυσκολίες. Στο διάστημα που μεσολάβησε από την αρχική έρευνα του τομέα, έχουν καθιερωθεί πιο σταθερά μέτρα για την αξιολόγηση της επίλυσης προβλημάτων και χωρίζονται συμβατικά σε δύο υποομάδες, με την μία να είναι συνδεδεμένη με τη διαδικασία και την άλλη με το αποτέλεσμα. Η υποομάδα της διαδικασίας έχει σχεδιαστεί να αποκτά πρόσβαση σε γενικές γνωστικές και συμπεριφορικές δραστηριότητες που διευκολύνουν την επίλυση προβλημάτων, ενώ αντίθετα η υποομάδα αποτελέσματος, αξιολογεί την απόδοση ή την ικανότητα επίλυσης προβλημάτων, όπως κρίνεται από τα προϊόντα της δραστηριότητας[22].

3.3.2.3 Θεραπεία μέσω Επίλυσης Προβλημάτων

Η Θεραπεία μέσω Επίλυσης Προβλημάτων (Θ.Ε.Π.) αναπτύχθηκε με σκοπό την παροχή ψυχοθεραπείας μέσω διδασκαλίας των ψυχοκοινωνικών δεξιοτήτων του πελάτη[23].

Η Θ.Ε.Π. περιλαμβάνει τη διδασκαλία σε έναν πελάτη σχετικά με το πώς να χρησιμοποιεί μια διαδικασία επίλυσης προβλημάτων βήμα προς βήμα. Η διαδικασία αυτή μπορεί να χωριστεί σε δύο κύρια μέρη:

1. Την εφαρμογή ενός τρόπου ζωής που σχετίζεται με την επίλυση προβλημάτων.
2. Την χρήση ορθολογικών δεξιοτήτων για την επίλυση των προβλημάτων αυτών.

Εφαρμόζοντας συνήθως έναν τρόπο ζωής ο οποίος είναι σχετικός με την επίλυση προβλημάτων περιλαμβάνουμε την αξιολόγηση τους ως προκλήσεις, με τη σκέψη ότι τα προβλήματα μπορούν να λυθούν και τη συνειδητοποίηση ότι μία αποτελεσματική επίλυση τους, τείνει να απαιτεί χρόνο και συστηματική προσπάθεια[25].

Οι δεξιότητες ορθολογικής επίλυσης προβλημάτων περιλαμβάνουν:

1. Προσπάθεια εντοπισμού ενός προβλήματος όταν παρουσιάζεται.
2. Ορισμός προβλήματος.
3. Προσπάθεια κατανόησης του προβλήματος.
4. Καθορισμός στόχων σχετικών με το πρόβλημα.

5. Δημιουργία εναλλακτικών λύσεων.
6. Αξιολόγηση και επιλογή των καλύτερων εναλλακτικών λύσεων.
7. Εφαρμογή των επιλεγμένων εναλλακτικών.
8. Την αξιολόγηση της αποτελεσματικότητας της προσπάθειας για την επίλυση του προβλήματος[25].

Η Θ.Ε.Π. συνήθως περιλαμβάνει προφορική και γραπτή παρουσίαση των βημάτων από τον θεραπευτή, μαζί με καθοδήγηση πρακτική, τόσο στη συνεδρία όσο και ως εργασίες στο σπίτι, στην επίλυση πραγματικών προβλημάτων, με σύσταση οι πελάτες να λαμβάνουν 8 με 16 συνεδρίες των 1,5 έως 2 ωρών η καθεμία[24].

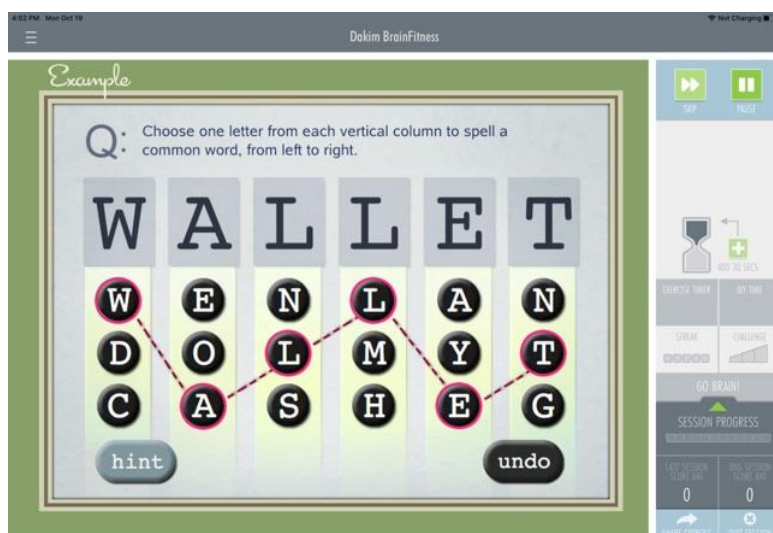
Στην έρευνα που πραγματοποιήθηκε[25], στην ανάλυση των αποτελεσμάτων των πειραμάτων έδειξε ότι σε όλες τις μελέτες της Θ.Ε.Π. είχαν σημαντικό μέγεθος επίδρασης. Η Θ.Ε.Π. ήταν κατά ένα σημαντικότερο ποσοστό αποτελεσματικότερη από μη θεραπεία, την θεραπεία προσοχής/εικονικό φάρμακο και θεραπεία που χρησιμοποιείται συνήθως, αλλά όχι σημαντικά πιο αποτελεσματική από άλλες καλόπιστες θεραπείες με τις οποίες έχει συγκριθεί.

Κεφάλαιο 4 Παρόμοιες Εφαρμογές

Στην προσπάθεια των ειδικών να καταπολεμήσουν τις νευρογνωστικές διαταραχές, χρησιμοποιούνται ασκήσεις οι οποίες έχουν ως σκοπό την εξάσκηση του εγκεφάλου, ώστε να διατηρηθεί σε ένα λειτουργικό επίπεδο ή ακόμα και να καταφέρει να γίνει πιο λειτουργικός μέσω των ασκήσεων. Αυτές οι ασκήσεις στοχεύουν στην εξάσκηση της μνήμης, της προσοχής ή ακόμα και την εξάσκηση της γλώσσας και των λέξεων μέσω της ορθογραφίας.

Μέχρι πρότινος, αυτές οι ασκήσεις γινόταν με την βοήθεια του χαρτιού και του μολυβιού, η σε κάποιες περιπτώσεις κλινικών με απτά εργαλεία, ειδικά σε ασκήσεις που έχουν ως σκοπό την κίνηση των άκρων. Τα τελευταία χρόνια όμως, η άνοδος της τεχνολογίας και η εύκολη προσβασιμότητα σε Smart συσκευές, όπως είναι τα smartphone, τα tablets κλπ., έχουν οδηγήσει στην άνθηση διαφόρων εφαρμογών οι οποίες στοχεύουν στο να βοηθήσουν άτομα με νευρογνωστικές διαταραχές. Παρακάτω, αναφέρονται κάποιες από αυτές τις εφαρμογές, καθώς και τις εκάστοτε διαταραχές που στοχεύουν και τους σκοπούς που προσπαθούν να επιτύχουν.

4.1 Dakim Brain Fitness



Εικόνα 4.1 Dakim Brain Fitness

Το Dakim Brain Fitness είναι ένα δομημένο, αυτοματοποιημένο πρόγραμμα εκπαίδευσης εγκεφάλου, το οποίο εκπαιδεύει άτομα με περισσότερες από 400 ασκήσεις σε έξι γνωστικούς τομείς (βραχυπρόθεσμη και μακροπρόθεσμη μνήμη, γλώσσα, οπτική χωρική επεξεργασία, συλλογιστική / επίλυση προβλημάτων και δεξιότητες υπολογισμού) σε συνεδρίες διάρκειας 25 λεπτών (6 με 10 ασκήσεις ανά συνεδρία)[26].

Η εταιρία ξεκίνησε το 1999 από τον Dan Michael, οπού δημιουργούσε ασκήσεις για τον πατέρα του που είχε Alzheimer, συνέχισε το 2006, με το πρόγραμμα αυτό να στέλνεται σε διαφόρους οίκους ευγηρίας ανά την Αμερική, ενώ το 2010 είναι η πρώτη φορά που το Brain Fitness γίνεται λογισμικό για χρήστες PC και Mac. Σε μια έρευνα που έγινε το 2013 φάνηκε ότι τα άτομα που χρησιμοποίησαν τον εν λόγω λογισμικό βελτίωσαν σημαντικά τη μνήμη (άμεση και καθυστερημένη), καθώς και τις και τις γλωσσικές ικανότητες[27].

Η αποτελεσματικότητα του προγράμματος εκπαίδευσης εγκεφάλου της Dakim βασίζεται σε ένα καινοτόμο πρωτόκολλο εκπαίδευσης εγκεφάλου που αναπτύχθηκε από την Dakim, με την καθοδήγηση από την ομάδα των επιστημονικών συμβούλων της, με επικεφαλής τον Gary Small, M.D., Διευθυντή του UCLA Longevity Centre. Το πρωτόκολλο του Dakim παρέχει μια γνωστική προπόνηση ευρείας βάσης που ασκεί τις περιοχές του εγκεφάλου που μας επιτρέπουν να εκτελούμε, σε υψηλό επίπεδο, τις ποικίλες γνωστικές και διανοητικές εργασίες που είναι απαραίτητες για την καριέρα και την καθημερινή μας ζωή. Για να γίνει αυτό, το Dakim Brain Fitness εκπαιδεύει τον εγκέφαλο σε έξι βασικούς γνωστικούς τομείς (μακροπρόθεσμη μνήμη, βραχυπρόθεσμη μνήμη, γλώσσα, υπολογισμός, οπτικοχωρικός προσανατολισμός και εκτελεστική λειτουργία) σε κάθε προπόνηση εγκεφάλου[26].

4.1.1 Τεχνολογία NuroLogic™

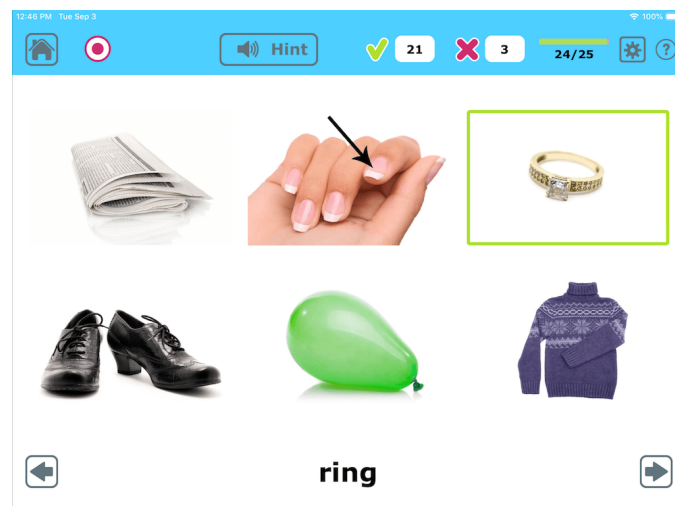
Η διαχείριση κάθε πτυχής της διαδικασίας εκπαίδευσης του εγκεφάλου είναι η κατοχυρωμένη με δίπλωμα ευρεσιτεχνίας Τεχνολογία NuroLogic™ της Dakim. Σε πραγματικό χρόνο, καθώς ο χρήστης προχωρά σε μια συνεδρία, η NuroLogic Technology τροποποιεί την εστίαση και τη σειρά των ασκήσεων που παρουσιάζονται, σε έξι γνωστικούς τομείς, και προσαρμόζει το επίπεδο πρόκλησης κάθε άσκησης, ώστε να ταιριάζει με ακρίβεια στις απαιτήσεις και την απόδοση του χρήστη. Το NuroLogic ακολουθεί τα παρακάτω βήματα:

1. Αναλύει ιστορικά στοιχεία της απόδοσης του χρήστη.

2. Ορίζει τον επόμενο γνωστικό τομέα που θα εκπαιδευτεί (π.χ. Βραχυπρόθεσμη Μνήμη).
3. Προσαρμόζει το επίπεδο πρόκλησης με βάση την απόδοση του χρήστη μέχρι εκείνη τη στιγμή (π.χ. Επίπεδο 1).
4. Επιλέγει από τον επιλεγμένο τομέα, την κατηγορία νευροψυχικών που θα εκπαιδεύσει (π.χ. Καθυστερημένη ανάκληση).
5. Επιλέγει τον κατάλληλο τύπο άσκησης Dakim από αυτήν την κατηγορία (π.χ. Broken News).
6. Επιλέγει τη συγκεκριμένη άσκηση από αυτόν τον τύπο άσκησης Dakim για παρουσίαση[26].

Τέλος, θα πρέπει να αναφερθούν ενδεικτικά πλεονεκτήματα και μειονεκτήματα της εφαρμογής. Ως πλεονέκτημα θεωρείται το γεγονός ότι είναι εύκολη εφαρμογή και διαχείριση από το προσωπικό, ότι προσφέρει εξελιγμένο περιεχόμενο προσαρμοσμένο σε ενήλικες άνω των 65 ετών, καθώς και παρέχει αναφορές παρακολούθησης χρήσης και απόδοσης. Επίσης είναι σχετικά φθηνό για οίκους ευγηρίας, κοστίζοντας 3.500 δολάρια ανά έτος για μέσους ή μεγάλους οίκους ευγηρίας, ενώ για κοινότητες με λιγότερα από 50 κρεβάτια, το ετήσιο κόστος είναι 35 δολάρια ανά κρεβάτι. Το μεγαλύτερο μειονέκτημα του προγράμματος αυτό είναι ότι απευθύνεται κατά κύριο λόγο σε οίκους ευγηρίας, καθώς και ότι τα προγράμματα του είναι για PC ή Mac.

4.2 Tactus Therapy



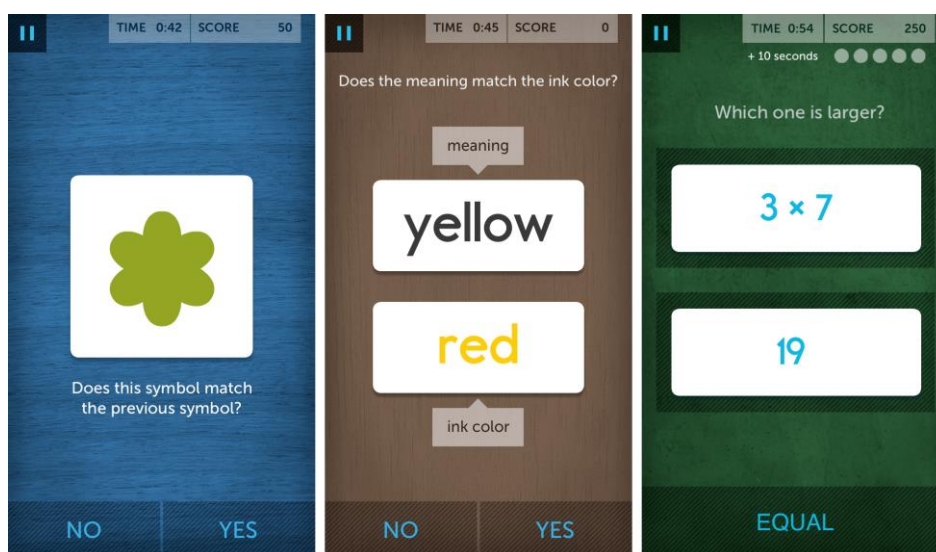
Εικόνα 4.2 Tactus Therapy

Το Tactus Therapy είναι μία σειρά εφαρμογών που απευθύνονται κυρίως σε άτομα με αφασία. Η εταιρεία δημιουργήθηκε από τους Megan Sutton και Ben Carter στο Βανκούβερ του Καναδά στην δεύτερη δεκαετία του 21ου αιώνα, και έχουν την ίδια έδρα έως και σήμερα[28].

Όπως προαναφέρθηκε, οι εφαρμογές απευθύνονται σε άτομα με αφασία. Η αφασία είναι μια διαταραχή που επηρεάζει τον τρόπο επικοινωνίας. Μπορεί να επηρεάσει την ομιλία, καθώς και τον τρόπο που κάποιος γράφει και κατανοεί τόσο τον προφορικό όσο και τον γραπτό λόγο. Επομένως οι εφαρμογές που παρέχονται αφορούν κυρίως γλωσσικές ασκήσεις, καθώς και ασκήσεις προσοχής και ομιλίας. Οι εφαρμογές είναι φτιαγμένες για Android και iOS λογισμικά, και επειδή δεν είναι web-based εφαρμογές, αλλά εφαρμογές που αποθηκεύονται στην εκάστοτε συσκευή, δεν χρειάζονται ίντερνετ για να τρέξουν. Επιπλέον, με τον τρόπο που είναι δομημένη η εταιρία και οι εφαρμογές που δημιουργούν δεν βασίζονται σε συνδρομές, αλλά αντιθέτως κάθε εφαρμογή έχει μια τιμή, η οποία είναι ανάλογη της έκτασης της και αφού την αγοράσει ο χρήστης την διατηρεί εφ' όρου ζωής χωρίς κάποια επιπλέον χρέωση. Οι εφαρμογές αυτές απευθύνονται τόσο σε προσωπικό όσο και σε επαγγελματικό επίπεδο και με το Tactus App Finder wizard, στο οποίο απαντώντας μερικές ερωτήσεις το πρόγραμμα προτείνει εφαρμογές που απευθύνονται στο εκάστοτε πρόβλημα που υπάρχει, είναι αρκετά εύκολο να βρεθούν οι εφαρμογές που θα είναι πιο χρήσιμες σε προσωπικό επίπεδο[28].

Το Tactus Therapy με την πολιτική που έχει εφαρμόσει, έχει πολλά θετικά, όπως πόσο εύκολα εντοπίζει εφαρμογές καθώς και ότι είναι φτιαγμένες σε Android και iOS περιβάλλοντα, κάνοντας τα πιο προσβάσιμα. Το μόνο αρνητικό που θα μπορούσε να προσάψει κάποιος είναι ότι είναι υπέρ εξειδικευμένο σε μία μόνο υποκατηγορία των νευρογνωστικών διαταραχών, επομένως άτομα τα οποία έχουν αδυναμία σε άλλους τομείς από αυτούς που προσφέρεται, θα πρέπει να στραφούν σε άλλες εφαρμογές για να καλύψουν τις ανάγκες τους.

4.3 Lumosity



Εικόνα 4.3 Lumosity

Το Lumosity είναι μια εφαρμογή, η οποία δημιουργήθηκε από την Lumos Labs. Η Lumos Labs ιδρύθηκε το 2005, εμπνευσμένη από την απλή υπόθεση ότι ο καθένας, οπουδήποτε, θα μπορούσε να χρησιμοποιήσει την ψηφιακή εκπαίδευση του εγκεφάλου για να ασκήσει βασικές νοητικές λειτουργίες ή δεξιότητες και να ανακαλύψει τι μπορεί να κάνει το μυαλό του[29]. Το Lumosity πρωτοεμφανίστηκε στην αγορά το 2007, μετά από δυο χρόνια έρευνας και ανάπτυξης, ενώ το 2013 επεκτάθηκε σε Android και iOS συσκευές. Το Lumosity είναι μία από τις τρεις εφαρμογές που έχουν δημιουργήσει, με τις άλλες δύο να επικεντρώνονται σε άλλες πτυχές του εγκεφάλου από την εκπαίδευση του. Η Lumos Labs χρησιμοποιεί τις εφαρμογές και τα αποτελέσματα που παράγονται για να προχωρήσει την επιστήμη και τις έρευνες που γίνονται σχετικά με τον εγκέφαλο, με απώτερο σκοπό να αναπτυχθεί ψηφιακή ιατρική για νευρολογικές διαταραχές[29].

Έχουν υπάρξει πάνω από 20 δημοσιεύσεις σε ακαδημαϊκά περιοδικά που χρησιμοποιούν το Lumosity. Σε μια μελέτη, οι επιστήμονες μας διεξήγαγαν μια τυχαioποιημένη δοκιμή στην οποία συμμετείχαν 4.715 συμμετέχοντες, προκειμένου να μελετήσουν εάν η γνωστική απόδοση βελτιώνεται μετά την προπόνηση με το Lumosity. Η ομάδα δοκιμής εκπαιδεύτηκε με το Lumosity, ενώ η ομάδα ελέγχου με σταυρόλεξα. Και οι δύο ομάδες προπονούσαν πέντε ημέρες την εβδομάδα, για δεκαπέντε λεπτά την ημέρα. Μετά από δέκα εβδομάδες, η ομάδα του Lumosity βελτιώθηκε στην απόδοση σε μια σειρά

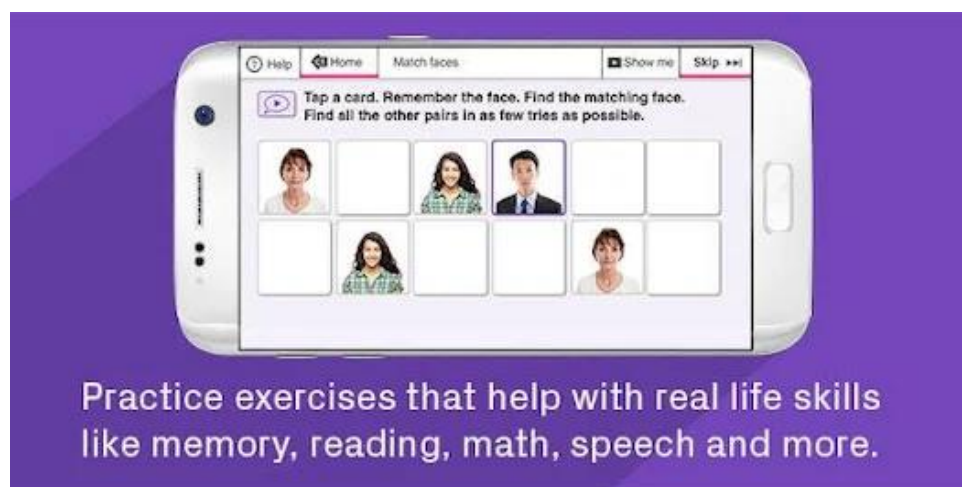
γνωστικών αξιολογήσεων. Στην πραγματικότητα, βελτιώθηκαν περισσότερο από δύο φορές περισσότερο από την ομάδα ελέγχου[30].

Όσον αφορά την εφαρμογή, το Lumosity έχει δεκάδες δραστηριότητες, οι οποίες υπάγονται σε 7 διαφορετικές κατηγορίες (ταχύτητας, μνήμης, προσοχής, ευελιξίας, επίλυσης προβλημάτων, λέξεων και μαθηματικών). Κάθε δραστηριότητα έχει tutorials πριν την έναρξη τους, ώστε να είναι βέβαιο ότι ο χρήστης θα είναι σίγουρος για το τι ζητάει κάθε δραστηριότητα, όπως επίσης υπάρχει και scoring, με σκοπό ο χρήστης να έχει μια γενική εικόνα στο πόσο καλά τα πήγε σε κάθε δραστηριότητα. Τέλος, το πρόγραμμα συγκεντρώνει τα αποτελέσματα και τα συγκρίνει με άλλα αποτελέσματα που έχει συγκεντρώσει από χρήστες οι οποίοι ανήκουν στην ίδια ηλικιακή ομάδα.

Το Lumosity προσφέρει κάποιες δωρεάν δραστηριότητες, αλλά για να χρησιμοποιήσεις όλες τις δραστηριότητες που προσφέρει θα πρέπει να πληρώνεις συνδρομή, η οποία μπορεί να είναι είτε μηνιαία είτε ετήσια, με την ετήσια συνδρομή να είναι οικονομικότερη της μηνιαίας. Επίσης προσφέρει μια περίοδο δύο εβδομάδων που μπορείς να χρησιμοποιήσεις την εφαρμογή με όλα τις υπηρεσίες που προσφέρει.

Το Lumosity έχει κάνει πολλά θετικά βήματα στον νοητικό τομέα, και συνεχίζει να εξελίσσεται με εφαρμογές και την σύνδεση τους με τις έρευνες που πραγματοποιεί η επιστημονική τους ομάδα. Το μόνο αρνητικό που μπορεί να εντοπιστεί είναι η συνδρομή, που σημαίνει ότι δεν μπορείς να αγοράσεις τις δραστηριότητες και να τις έχεις εφ' όρου ζωής.

4.4 Constant Therapy



Εικόνα 4.4 Constant Therapy

Η εταιρία Constant Therapy Health ιδρύθηκε με την ιδέα ότι η ψηφιοποίηση της θεραπείας μπορεί να συμβάλλει στη βελτίωση της φροντίδας, στη μείωση του κόστους και να βοηθήσει τους ανθρώπους να φτάσουν σε νέα επίπεδα ανάρρωσης. Ο πρωταρχικός της στόχος είναι να βοηθήσει τους επιζώντες εγκεφαλικού τραυματισμού, εγκεφαλικού ή ακόμα και τους ανθρώπους που ζουν με αφασία ή άνοια να βελτιώσουν το επίπεδο ζωής τους. Ένας επιπλέον στόχος της είναι να βοηθήσει τους κλινικούς ιατρούς να χρησιμοποιήσουν το πρόγραμμα και τα δεδομένα του για να εξελίσουν την συνεργασία τους με τους ασθενείς τους. Το μέσο που χρησιμοποιούν για να επιτύχουν τους στόχους τους είναι το NeuroPerformance Engine[31].

Το NeuroPerformance Engine είναι ένα κατοχυρωμένο με δίπλωμα ευρεσιτεχνίας AI που χρησιμοποιείται σε όλα την τα προϊόντα. Αναλύει δεδομένα από τον πραγματικό κόσμο που δημιουργούνται από τις αλληλεπιδράσεις των χρηστών με το προϊόν για να παρέχει ασκήσεις βασισμένες σε στοιχεία προσαρμοσμένες για κάθε ασθενή. Προκειμένου να παρέχει τις πιο αποτελεσματικές, ασκήσεις για τον τραυματισμό ή τη διαταραχή που θέλουν να αντιμετωπίσουν οι χρήστες, το NeuroPerformance Engine αξιολογεί πρώτα τις ικανότητες των χρηστών σε διάφορους γνωστικούς τομείς. Στη συνέχεια εξετάζει τον τρόπο με τον οποίο οι χρήστες εκτελούν τις ασκήσεις Constant Therapy για να καθορίσει ακριβώς πώς να προσαρμόσουν το πρόγραμμα ώστε να ταιριάζει στις μοναδικές και μεταβαλλόμενες ανάγκες κάθε χρήστη, διασφαλίζοντας ότι σημειώνουν την πιο αποτελεσματική και αποτελεσματική πρόοδο προς τους στόχους τους. Και πολλές ερευνητικές μελέτες έχουν δείξει ότι το NPE λειτουργεί[31].

Το AI του NeuroPerformance αλληλοεπιδρά με τον χρήστη του με εξής βήματα:

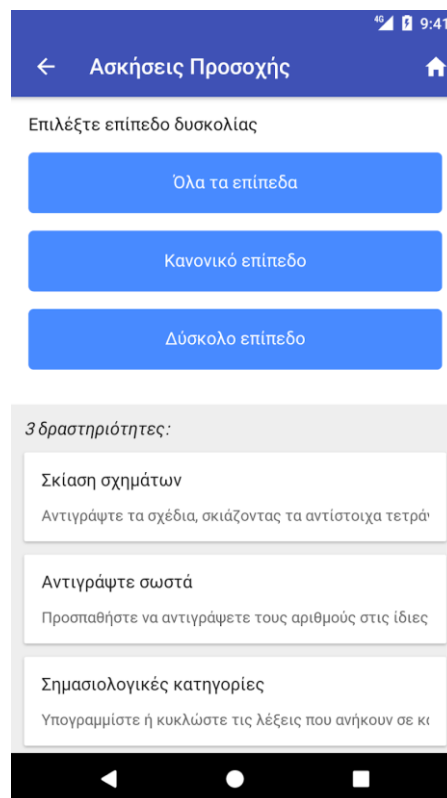
1. Οι νέοι ασθενείς που εργάζονται με έναν κλινικό ιατρό ή μόνοι τους κατεβάζουν το Constant Therapy σε smartphone, Chromebook ή tablets και συμπληρώνουν δημογραφικά και διαγνωστικά στοιχεία.
2. Με βάση το προφίλ του ασθενούς, το NPE συνιστά αυτόματα συγκεκριμένες ασκήσεις σε διαφορετικές κατηγορίες και επίπεδα δυσκολίας που ταιριάζουν καλύτερα στις ανάγκες του.
3. Ο κλινικός ιατρός και ο ασθενής μπορούν να αποφασίσουν εάν θα αποδεχτούν ή θα προσαρμόσουν το προτεινόμενο πρόγραμμα άσκησης και, στη συνέχεια, το πρόγραμμα αρχίζει να παρέχει τις εξατομικευμένες θεραπευτικές ασκήσεις στον ασθενή.

4. Καθώς ο ασθενής ολοκληρώνει τις θεραπευτικές του ασκήσεις, το NPE μετρά την απόδοση του ασθενούς σε διαφορετικές διαστάσεις όπως η ακρίβεια, η ταχύτητα και η συνέπεια.
5. Στη συνέχεια, το NPE προσαρμόζει τις θεραπευτικές ασκήσεις για τον ασθενή με βάση την ατομική πρόοδο απόδοσης και τις βαθμολογίες πληθυσμού.

Καθώς το NPE συλλέγει πιο αναλυτικά δεδομένα και εφαρμόζει όλο και πιο εξελιγμένες τεχνικές μηχανικής μάθησης, γίνεται καλύτερο και πιο ακριβές στο να προτείνει το σωστό θεραπευτικό σχήμα και δοσολογία για κάθε ασθενή, έτσι ώστε να προσαρμόζεται ειδικά στις ανάγκες του[31].

Με την αυξανόμενη υιοθέτηση αυτών των τεχνολογιών, η Constant Therapy βλέπει τη δυνατότητα για βελτιωμένη κοινή χρήση κλινικών δεδομένων σε όλη τη συνέχεια της περίθαλψης και μεγαλύτερη δέσμευση με την κατάσταση της υγειονομικής περίθαλψης των ασθενών εκτός κλινικών πλαισίων. Τελικά, αυτό θα επιτρέψει στην Constant Therapy να παρέχει φάρμακα ακριβείας σε όλους σε κλίμακα, οδηγώντας σε χαμηλότερο κόστος και καλύτερη υγεία του πληθυσμού[31].

4.5 DiAnoia



Εικόνα 4.5 DiAnoia

Η εφαρμογή DiAnoia δημιουργήθηκε από την sciFY, μια Αστική Μη Κερδοσκοπική Εταιρεία που ιδρύθηκε το 2012, και σήμερα απασχολεί 8 εργαζόμενους και 10 εθελοντές ενώ έχει λάβει αρκετές διακρίσεις και βραβεία, όπως από τον Πρόεδρο της Ελληνικής Δημοκρατίας Κ. Κάρολο Παπούλια το 2013[32] και η αρχική της έκδοση αναπτύχθηκε με την επιμέλεια και την συνεργασία της Εταιρείας Alzheimer Αθηνών, ενός Μη Κερδοσκοπικού Οργανισμού, που ιδρύθηκε το 2002, από άτομα με άνοια, συγγενείς τους και επαγγελματίες υγείας που ασχολούνται με την άνοια και τη νόσο Alzheimer[33].

Η εφαρμογή απευθύνεται σε φροντιστές ατόμων με άνοια, επαγγελματίες υγείας που δραστηριοποιούνται στο χώρο της άνοιας αλλά και σε κλινικές που ασχολούνται με την φροντίδα ανθρώπων με άνοια[34]. Η εφαρμογή παρέχεται για Android και iOS και είναι δωρεάν στο να την κατεβάσει κάποιος.

Η εφαρμογή έχει διάφορες δραστηριότητες για αρκετές κατηγορίες, όπως προσοχής και μνήμης, καθώς και διαφορετικά επίπεδα δυσκολίας, καθώς και έχει και προτάσεις για τους φροντιστές σχετικά με το πως να περνάν δημιουργικά τον χρόνο τους με τους ασθενείς. Οι δραστηριότητες που έχει σαν εφαρμογή είναι έτοιμες σελίδες τις οποίες μπορείς να εκτυπώσεις και να δώσεις στον ασθενή να ενασχοληθεί.

Στα θετικά της εφαρμογής είναι το ότι η χρήση της είναι δωρεάν , αλλά το κυριότερο είναι ότι είναι η μοναδική ελληνική εφαρμογή που υπάρχει για Android και iOS. Το μεγαλύτερο αρνητικό που έχει είναι ότι παρόλο που υπάρχει στα κινητά για να την αποκτήσεις, πρακτικά η εφαρμογή είναι απλά μια αποθήκη ασκήσεων, οι οποίες χρειάζονται χαρτί και μολύβι για να επιλυθούν.

Κεφάλαιο 5 Ανάλυση της Μαθησιακής Εφαρμογής

Λαμβάνοντας υπ' όψιν όλα όσα παρατηρήθηκαν και σχολιάστηκαν στα προηγούμενα κεφάλαια, αποφασίστηκε, σαν μέρος της διπλωματικής εργασίας, να δημιουργηθεί μία εφαρμογή η οποία θα είναι σχεδιασμένη για να βοηθήσει άτομα που πάσχουν από νευρογνωστικές διαταραχές.

5.1 Αρχικός Σχεδιασμός

Η αρχική ιδέα για την υλοποίηση της εφαρμογής ήταν να υλοποιηθεί σαν web-based application, όπως το Dakim Brain Fitness, που αναλύσαμε στο προηγούμενο κεφάλαιο. Όμως, κατά την διάρκεια της αρχικής υλοποίησης της, υπήρξε μια συνάντηση με την νευροψυχολόγο Έλλη Ζούπα και, κατά την διάρκεια της συνάντησης ενώ αναφέρθηκε η αρχική ιδέα για την πραγματοποίηση της εφαρμογής, υπήρξε η αντιπρόταση να πραγματοποιηθεί σε περιβάλλον Android, καθώς οι ασθενείς που υπάρχουν στην συγκεκριμένη δομή που εργάζεται βρίσκουν αρκετά βοηθητικό το touch screen που έχουν τα tablets. Η συζήτηση αυτή, σε συνδυασμό με την μεταγενέστερη έρευνα και τον εντοπισμό των υπόλοιπων εφαρμογών που αναφέρθηκαν στο προηγούμενο κεφάλαιο και λειτουργούν σε περιβάλλον Android, με ώθησαν να υλοποιηθεί την εφαρμογή σε περιβάλλον Android.

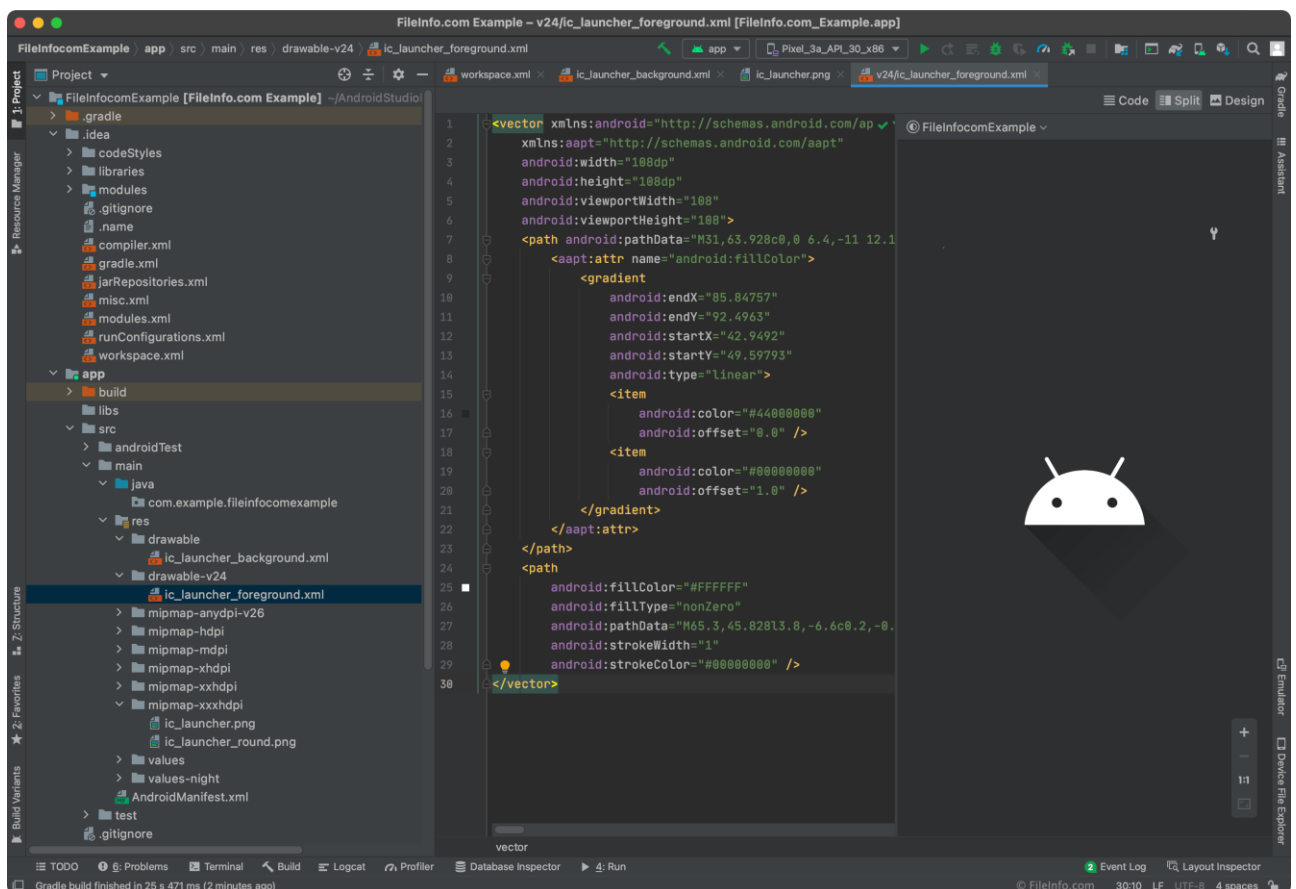
Προτού ξεκινήσει η αρχική υλοποίηση της εφαρμογής, εξετάστηκε η ιδέα να δημιουργηθεί αρχικά σε περιβάλλον iOS. Η βασική γλώσσα προγραμματισμού που χρησιμοποιείται στο iOS είναι η Swift, μια υψηλού επιπέδου μεταγλωττισμένη γλώσσα προγραμματισμού γενικής χρήσης, πολλαπλών παραδειγμάτων, που αναπτύχθηκε από την Apple Inc. και την κοινότητα ανοιχτού κώδικα. Η Swift κυκλοφόρησε για πρώτη φορά το 2014 και αναπτύχθηκε ως αντικατάσταση της προηγούμενης γλώσσας προγραμματισμού της Apple, η οποία ήταν η Objective-C, καθώς ήταν σε μεγάλο βαθμό αμετάβλητη από τις αρχές της δεκαετίας του 1980 και δεν διέθετε σύγχρονες γλωσσικές δυνατότητες [35]. Για την διευκόλυνση των προγραμματιστών στην δημιουργία εφαρμογών σε iOS υπάρχει το Apple Developer, που είναι ο ιστότοπος της Apple Inc. για εργαλεία ανάπτυξης λογισμικού, διεπαφές προγραμματισμού εφαρμογών (API) και τεχνικούς πόρους. Επίσης περιέχει

πόρους που βοηθούν τους προγραμματιστές λογισμικού να γράφουν λογισμικό για τις πλατφόρμες macOS, iOS, iPadOS, watchOS και tvOS[36]. Όμως, επειδή δεν υπήρχε εμπειρία ούτε χρήση του περιβάλλοντος και του λογισμικού της Apple, επανήλθε η αρχική ιδέα, δηλαδή η δημιουργία της εφαρμογής για Android συσκευές.

5.1.1 Περιβάλλον Υλοποίησης της Εφαρμογής

Για να υλοποιηθεί η εφαρμογή σε περιβάλλον Android χρειάζεται το σωστό λογισμικό για να γραφτεί ο κώδικας για την υλοποίηση του. Όπως στο λογισμικό iOS υπάρχει το Apple Developer για την ευκολότερη δημιουργία iOS εφαρμογών, αντίστοιχα υπάρχει το Android Studio για την δημιουργία Android εφαρμογών.

Το Android Studio είναι το επίσημο περιβάλλον ανάπτυξης για το λειτουργικό σύστημα Android της Google, το οποίο βασίζεται στο λογισμικό IntelliJ IDEA της JetBrains και έχει σχεδιαστεί ειδικά για ανάπτυξη Android και είναι διαθέσιμο για λήψη σε λειτουργικά συστήματα που βασίζονται σε Windows, Mac OS και Linux. Είναι μια αντικατάσταση του Eclipse Android Development Tools ως το κύριο ολοκληρωμένο περιβάλλον ανάπτυξης για την ανάπτυξη εφαρμογών Android[37].



Εικόνα 5.1 Περιβάλλον εργασίας του Android Studio

Το Android Studio ανακοινώθηκε στις 16 Μαΐου 2013, όπου και βρισκόταν σε στάδιο early access με την έκδοση 0.1. Τον Ιούνιο του 2014 εισήλθε σε στάδιο beta ,με την έκδοση 0.8, ενώ η πρώτη σταθερή έκδοση κυκλοφόρησε τον Δεκέμβριο του 2014. Στα τέλη του 2015, η Google διέκοψε την υποστήριξη για το Eclipse ADT, καθιστώντας το Android Studio το μόνο επίσημα υποστηριζόμενο IDE για ανάπτυξη Android[37].

Στις 7 Μαΐου 2019, η Kotlin αντικατέστησε την Java ως προτιμώμενη γλώσσα της Google για την ανάπτυξη εφαρμογών, ενώ η Java εξακολουθεί να υποστηρίζεται, όπως και η C++[37].

Το βασικότερο και σημαντικότερο χαρακτηριστικό που προσφέρει το Android Studio είναι το Android Emulator. Το Android Emulator προσομοιώνει συσκευές Android στον υπολογιστή, ώστε να μπορεί ο χρήστης να δοκιμάσει την εφαρμογή σε διάφορες συσκευές και επίπεδα Android χωρίς να χρειάζεται να έχετε κάθε φυσική συσκευή[38]. Επομένως, κατά την διάρκεια της υλοποίησης της εφαρμογής ήταν αρκετά εύκολο, μέσω του Android Emulator, να ελέγχεται ο κώδικας και να διορθώνονται τυχόν συντακτικά ή λογικά λάθη με αρκετή άνεση, μιας και το τρέξιμο της εφαρμογής στο Emulator είναι ταχύτατο.

5.1.2 Προγραμματιστική Γλώσσα

Από την στιγμή που επιλέχθηκε το Android Studio σαν το λογισμικό στο οποίο θα υλοποιούνταν ο κώδικας της εφαρμογής, το επόμενο βήμα ήταν να επιλεγεί και η προγραμματιστική γλώσσα με την οποία θα γραφόταν ο κώδικας της εφαρμογής. Διαβάζοντας για τις γλώσσες προγραμματισμού που υποστηρίζει το Android Studio, υπήρχαν τρεις βασικές υποψήφιες για την υλοποίηση του κώδικα: Η C++, Η Java, και η Kotlin.

Η C++ είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου, γενικής χρήσης που δημιουργήθηκε από τον Δανό επιστήμονα υπολογιστών Bjarne Stroustrup. Κυκλοφόρησε για πρώτη φορά το 1985 ως επέκταση της γλώσσας προγραμματισμού C, από τότε έχει επεκταθεί σημαντικά με την πάροδο του χρόνου. Η σύγχρονη C++ διαθέτει αντικειμενοστραφή χαρακτηριστικά, αλλά και εγκαταστάσεις για χειρισμό μνήμης χαμηλού επιπέδου. Εφαρμόζεται σχεδόν πάντα ως μεταγλωττισμένη γλώσσα ενώ πολλοί προμηθευτές παρέχουν μεταγλωττιστές C++, όπως π.χ. η Free Software Foundation, η Microsoft, η Intel, αλλά και η Oracle[39].

Η C++ σχεδιάστηκε με προγραμματισμό συστημάτων και ενσωματωμένο λογισμικό με περιορισμένους πόρους και μεγάλα συστήματα κατά νου, με την απόδοση, την αποτελεσματικότητα και την ευελιξία στη χρήση ως κύρια σημεία του σχεδιασμού της. Η C++ έχει επίσης φανεί χρήσιμη και σε άλλα πλαίσια, με βασικότερα πλεονεκτήματα της να είναι η υποδομή λογισμικού και οι εφαρμογές με περιορισμένους πόρους, συμπεριλαμβανομένων εφαρμογών για PC, βιντεοπαιχνιδιών και Server[39].

Η Java είναι μια υψηλού επιπέδου, βασισμένη σε κλάσεις, αντικειμενοστραφή γλώσσα προγραμματισμού που έχει σχεδιαστεί για να έχει όσο το δυνατόν λιγότερες εξαρτήσεις υλοποίησης. Είναι μια γλώσσα προγραμματισμού γενικής χρήσης που προορίζεται να επιτρέπει στους προγραμματιστές το λεγόμενο write once, run anywhere (WORA), που σημαίνει ότι ο μεταγλωττισμένος κώδικας Java μπορεί να εκτελεστεί σε όλες τις πλατφόρμες που υποστηρίζουν Java χωρίς να χρειάζεται να γίνει εκ νέου μεταγλώττιση. Οι εφαρμογές Java συνήθως μεταγλωττίζονται σε bytecode που μπορεί να εκτελεστεί σε οποιαδήποτε εικονική μηχανή Java (JVM) ανεξάρτητα από την υποκείμενη αρχιτεκτονική του υπολογιστή. Η σύνταξη της Java είναι παρόμοια με τη C και τη C++, αλλά έχει λιγότερες λειτουργίες χαμηλού επιπέδου από κάθε μία από αυτές. Ο χρόνος εκτέλεσης Java παρέχει δυναμικές δυνατότητες που συνήθως δεν είναι διαθέσιμες σε παραδοσιακές μεταγλωττισμένες γλώσσες. Το 2019, η Java ήταν μια από τις πιο δημοφιλείς γλώσσες προγραμματισμού σε χρήση σύμφωνα με το GitHub, ιδιαίτερα για εφαρμογές client-server, με 9 εκατομμύρια προγραμματιστές να την χρησιμοποιούν[40].

Η Java αναπτύχθηκε αρχικά από τον James Gosling στη Sun Microsystems και κυκλοφόρησε πρώτη φορά το 1995 ως βασικό στοιχείο της πλατφόρμας Java της Sun Microsystems. Οι αρχικοί μεταγλωττιστές Java και η εφαρμογή αναφοράς, οι εικονικές μηχανές και οι βιβλιοθήκες κλάσεων κυκλοφόρησαν αρχικά από τη Sun με ιδιόκτητες άδειες. Από τον Μάιο του 2007, η Sun είχε αδειοδοτήσει εκ νέου τις περισσότερες από τις τεχνολογίες Java της με την άδεια μόνο GPL-2.0. Η Oracle προσφέρει το δικό της HotSpot Java Virtual Machine, ωστόσο η επίσημη υλοποίηση αναφοράς είναι το OpenJDK JVM, το οποίο είναι δωρεάν λογισμικό ανοιχτού κώδικα και χρησιμοποιείται από τους περισσότερους προγραμματιστές και είναι το προεπιλεγμένο JVM για όλες σχεδόν τις διανομές Linux[40].

Η Kotlin είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου γενικού σκοπού, στατικά πληκτρολογημένα, πολλαπλών πλατφόρμων[41]. Η Kotlin έχει σχεδιαστεί για να

λειτουργεί πλήρως με την Java, και η έκδοση JVM της τυπικής βιβλιοθήκης του Kotlin εξαρτάται από τη βιβλιοθήκη Java Class, με το type inference που έχει επιτρέπει τη σύνταξή της να είναι πιο συνοπτική. Η Kotlin στοχεύει κυρίως το JVM, αλλά μεταγλωττίζεται και σε JavaScript ή άλλον εγγενή κώδικα μέσω του LLVM[41], ένα σύνολο τεχνολογιών μεταγλωττιστή και αλυσίδας εργαλείων που μπορούν να χρησιμοποιηθούν για την ανάπτυξη ενός frontend για οποιαδήποτε γλώσσα προγραμματισμού και ενός backend για οποιαδήποτε αρχιτεκτονική συνόλου εντολών[42]. Το κόστος ανάπτυξης γλώσσας βαρύνει την JetBrains, ενώ το Ίδρυμα Kotlin προστατεύει το εμπορικό σήμα Kotlin[41].

Στις 7 Μαΐου 2019, η Google ανακοίνωσε ότι η γλώσσα προγραμματισμού Kotlin ήταν η πλέον προτιμώμενη γλώσσα για τους προγραμματιστές εφαρμογών Android. Από την κυκλοφορία του Android Studio 3.0 τον Οκτώβριο του 2017, η Kotlin έχει συμπεριληφθεί ως εναλλακτική λύση στον τυπικό μεταγλωττιστή της Java. Ο μεταγλωττιστής Android Kotlin παράγει Java 8 bytecode από προεπιλογή, αλλά επιτρέπει στον προγραμματιστή να επιλέξει Java 9-20, για βελτιστοποίηση ή επιτρέπει περισσότερες δυνατότητες. Επίσης, έχει αμφίδρομη υποστήριξη λειτουργικότητας κλάσης εγγραφής για JVM, που εισήχθη στην Java 16 και θεωρείται σταθερή από το Kotlin 1.5[41].

Διαβάζοντας για τις τρεις γλώσσες και παρακολουθώντας κάποια εκπαιδευτικά video, υπήρξε η τελική απόφαση να χρησιμοποιηθεί η Kotlin ως γλώσσα προγραμματισμού της εφαρμογής. Όμως η Kotlin είναι υπεύθυνη μόνο για την υλοποίηση του κώδικα, και το πως αλληλοεπιδρά η εφαρμογή με τον χρήστη. Για το πώς η εφαρμογή εμφανίζεται στον χρήστη θα πρέπει να χρησιμοποιηθεί το LayoutEditor.

Το Layout Editor δίνει τη δυνατότητα να δημιουργούνται γρήγορα διατάξεις σύροντας στοιχεία διεπαφής χρήστη σε ένα πρόγραμμα επεξεργασίας οπτικού σχεδιασμού αντί να γράφετε διάταξη XML. Το πρόγραμμα επεξεργασίας σχεδίασης μπορεί να κάνει προεπισκόπηση της διάταξής σας σε διαφορετικές συσκευές και εκδόσεις Android και μπορείτε να αλλάξετε δυναμικά το μέγεθος της διάταξης για να βεβαιωθείτε ότι λειτουργεί σωστά σε διαφορετικά μεγέθη οθόνης.

Το Layout Editor είναι ιδιαίτερα ισχυρό όταν δημιουργείτε μια διάταξη με το ConstraintLayout. Το ConstraintLayout επιτρέπει να δημιουργείτε μεγάλες, σύνθετες διατάξεις με ιεραρχία επίπεδης προβολής—χωρίς ένθετες ομάδες προβολής. Είναι παρόμοιο με το RelativeLayout, καθώς όλες οι προβολές διατάσσονται σύμφωνα με τις

σχέσεις μεταξύ των προβολών αδελφών και της γονικής διάταξης, αλλά είναι πιο ευέλικτο από το RelativeLayout και είναι πιο εύκολο στη χρήση με το πρόγραμμα επεξεργασίας διάταξης του Android Studio.

5.2 Βασικές Λειτουργίες

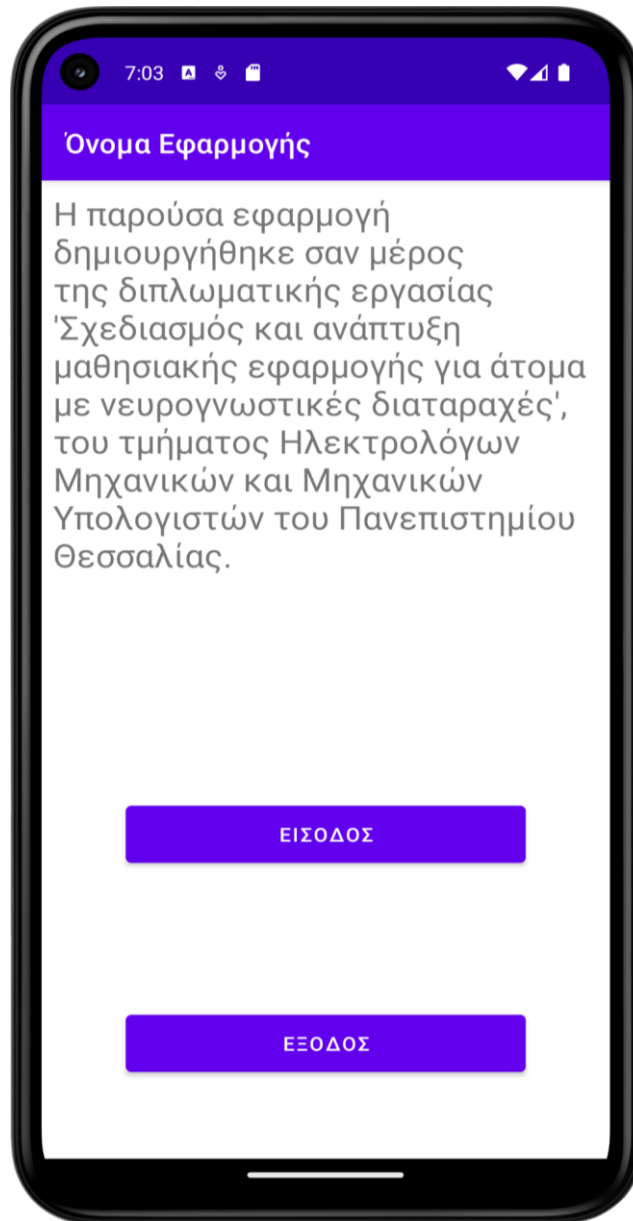
Η εφαρμογή είναι, σε πρακτικό επίπεδο, απλή στον σχεδιασμό και την υλοποίηση της, το οποίο ήταν και ο στόχος, λαμβάνοντας υπ' όψιν το κοινό στο οποίο κατά κύριο λόγο προορίζεται, δηλαδή άτομα προχωρημένης ηλικίας τα οποία πάσχουν ήδη ή παρατηρούνται συμπτώματα από κάποια από τις διαταραχές που αναλύθηκαν σε προηγούμενο κεφάλαιο.

5.2.1 Βασικός Σκελετός της Εφαρμογής

Πριν την υλοποίηση των δραστηριοτήτων της εφαρμογής, θα έπρεπε να δημιουργηθεί ο βασικός σκελετός της, ώστε να έχω μια βασική ιδέα το τι χρειάζεται η εφαρμογή για να είναι λειτουργική και ευανάγνωστη.

Το Android Studio λειτουργεί με Activities. Το Activity class είναι ένα κρίσιμο στοιχείο μιας εφαρμογής Android και ο τρόπος με τον οποίο ξεκινούν και δημιουργούνται οι δραστηριότητες είναι ένα βασικό μέρος του μοντέλου εφαρμογής της πλατφόρμας. Σε αντίθεση με τα παραδείγματα προγραμματισμού στα οποία οι εφαρμογές ξεκινούν με μια μέθοδο main(), το σύστημα Android εκκινεί τον κώδικα σε ένα Activity επικαλώντας συγκεκριμένες μεθόδους επανάκλησης που αντιστοιχούν σε συγκεκριμένα στάδια του κύκλου ζωής του. Οι εφαρμογές για κινητά διαφέρουν από τις αντίστοιχες για επιτραπέζιους υπολογιστές στο γεγονός ότι η αλληλεπίδραση ενός χρήστη με την εφαρμογή δεν ξεκινά πάντα από το ίδιο σημείο αλλά, αντίθετα, συχνά μη ντετερμινιστικά[43]. Το Activity class έχει σχεδιαστεί για να διευκολύνει αυτό το παράδειγμα. Όταν μια εφαρμογή καλεί μια άλλη, η εφαρμογή κλήσης επικαλείται ένα Activity στην άλλη εφαρμογή, αντί για την εφαρμογή ως ατομικό σύνολο. Με αυτόν τον τρόπο, το Activity χρησιμεύει ως το σημείο εισόδου για την αλληλεπίδραση μιας εφαρμογής με τον χρήστη. Ένα Activity παρέχει το παράθυρο στο οποίο η εφαρμογή σχεδιάζει τη διεπαφής χρήστη της. Αυτό το παράθυρο συνήθως γεμίζει την οθόνη, αλλά μπορεί να είναι μικρότερο από την οθόνη. Γενικά, ένα Activity υλοποιεί μια οθόνη σε μια εφαρμογή. Οι περισσότερες εφαρμογές περιέχουν πολλές οθόνες, πράγμα που σημαίνει

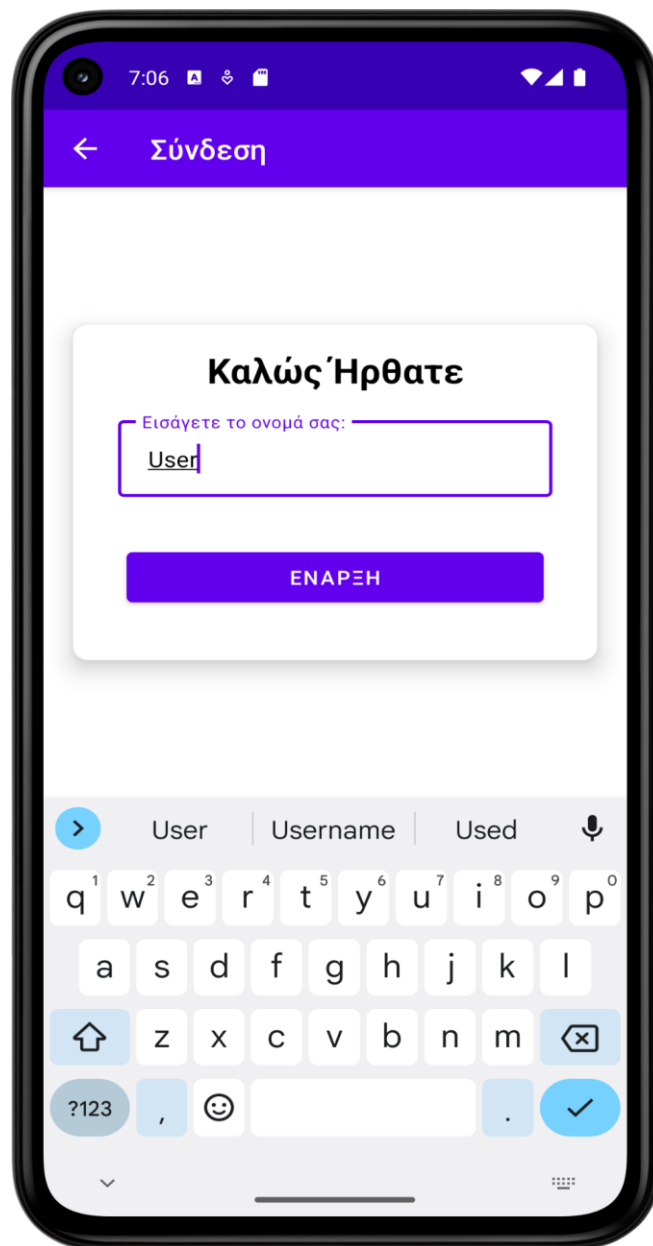
ότι περιλαμβάνουν πολλαπλά activities. Συνήθως, ένα Activity σε μια εφαρμογή καθορίζεται ως η mainactivity, η οποία είναι η πρώτη οθόνη που εμφανίζεται όταν ο χρήστης εκκινεί την εφαρμογή. Κάθε δραστηριότητα μπορεί στη συνέχεια να ξεκινήσει μια άλλη δραστηριότητα για να εκτελέσει διαφορετικές ενέργειες[43].



Εικόνα 5.2 Αρχική οθόνη της εφαρμογής

Όπως αναφέρθηκε στην προηγούμενη παράγραφο, συνήθως η πρώτη οθόνη που εμφανίζεται στον χρήστη όταν εισέρχεται σε μία εφαρμογή θεωρείται το main Activity. Στην περίπτωση της εφαρμογής μου, παρόλο που η πρώτη οθόνη είναι ονομασμένη σαν main Activity, δεν είναι η βασική οθόνη στην οποία βρίσκονται οι δραστηριότητες.

Αντιθέτως, είναι μία απλή οθόνη, στην οποία υπάρχει ένα TextView στο οποίο αναγράφονται δύο λόγια σχετικά με την υλοποίηση της εφαρμογής, και δυο buttons, το ένα για να εισέλθει ο χρήστης στην εφαρμογή, ένα αντίθετα να εξέλθει με το άλλο. Στην αρχική υλοποίηση της εφαρμογής υπήρχαν 3 κουμπιά μόνο και το τρίτο κουμπί σε μετέφερε σε ένα καινούριο Activity το οποίο είχα ονομάσει «about». Σε εκείνο το Activity υπήρχε αρχικά το TextView το οποίο αναφέρεται στην εφαρμογή και το οποίο μεταφέρθηκε τελικά στην αρχική οθόνη σε δεύτερο χρόνο, για να κρατήσω την εφαρμογή σε πιο απλά επίπεδα.



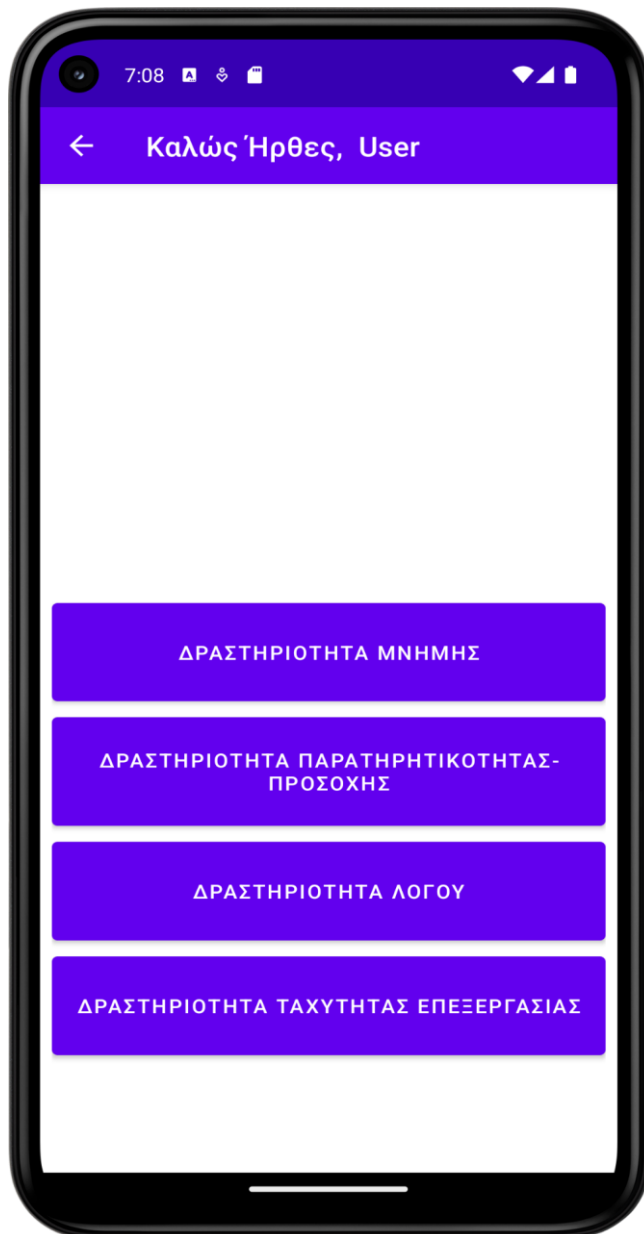
Εικόνα 5.3 Οθόνη σύνδεσης της εφαρμογής

Εάν ο χρήστης πατήσει το κουμπί εισόδου, τότε τον μεταφέρει στο επόμενο Activity, το οποίο είναι ένα login screen. Αρκετά απλό στην υλοποίηση του, ζητάει το όνομα του χρήστη και, εάν ο χρήστης το δώσει, τον μεταφέρει στο επόμενο Activity. Ένα σχετικό ενδιαφέρον για αυτό το Activity παρουσιάζει ο κάτωθι κώδικας:

```
if(enteredName.isEmpty()){  
    Toast.makeText(this, "Παρακαλώ, Γράψτε το όνομά σας!", Toast.LENGTH-  
    SHORT).show  
}else{  
    val sharedPreferences = getSharedPreferences("MyPrefs", Context.MODE-  
    PRIVATE)  
    val editor = sharedPreferences.edit()  
    editor.putString("userName", enteredName)  
    editor.apply()  
    val intent=Intent(this, ActivitiesMainScreen::class.java){  
        startActivity(intent),  
    }  
}  
},
```

τον οποίο το Activity επεξεργάζεται το string που δίνει ο χρήστης για όνομα. Εάν το string είναι κενό, τότε με το `Toast.makeText().show()` εμφανίζεται το κείμενο "Παρακαλώ, Γράψτε το όνομά σας!", και όταν ο χρήστης εισάγει το όνομά του και πατήσει το κουμπί "Εναρξη" τότε το όνομα το οποίο εισήγαγε αποθηκεύεται και μεταφέρεται στο επόμενο Activity.

Το επόμενο Activity είναι η κύρια οθόνη στην οποία υπάρχουν 4 buttons, όπου το καθένα αντιστοιχεί σε μία από τις 4 δραστηριότητες που έχουν υλοποιηθεί σαν μέρος της εργασίας. Επίσης, στο πάνω μέρος της οθόνης υπάρχει ένα κείμενο το οποίο γράφει "Καλώς Ήρθες, userName" όπου το username είναι το όνομα το οποίο έδωσε ο χρήστης στο login screen. Ο κώδικας που χρησιμοποιήθηκε για να λάβει αυτό το Activity το όνομα είναι ο κάτωθι:



Εικόνα 5.4 Η οθόνη στην οποία φαίνονται όλες οι δραστηριότητες της εφαρμογής

```
val sharedPreferences =getSharedPreferences("MyPrefs", Context.MODE_PRIVATE)
```

```
val userName = sharedPreferences.getString("userName", "DefaultName")
```

Ενώ ο κώδικας που εμφανίζει το "Καλώς Ήρθες, username", είναι:

```
supportActionBar?.title = "Καλώς Ήρθες, $userName"
```

Οι δραστηριότητες, όπως προαναφέρθηκε είναι 4, μία για κάθε τομέα που ήθελα να εκπαιδεύονται οι χρήστες. Επομένως υπάρχει μία δραστηριότητα μνήμης, μία δραστηριότητα προσοχής-παρατηρητικότητας, μία δραστηριότητα λόγου και τέλος, μία δραστηριότητα ταχύτητας επεξεργασίας. Επίσης τα activities για κάθε δραστηριότητα είναι δημιουργημένα με την ίδια λογική: πατώντας το κουμπί για κάθε δραστηριότητα

υπάρχει πρώτα ένα Activity στο οποίο ο χρήστης βλέπει ένα κείμενο που περιγράφει την εκάστοτε δραστηριότητα και τί πρέπει να κάνει ανάλογα την δραστηριότητα που πρέπει να δοκιμάσει και ένα κουμπί έναρξης το οποίο μόλις πατηθεί, μεταφέρεται ο χρήστης στην δραστηριότητα.



Εικόνα 5.5 Η οθόνη που αναφέρει το πώς κερδίζεται η δραστηριότητα μνήμης

5.3 Δραστηριότητες

Οι δραστηριότητες, όπως προαναφέρθηκε είναι 4, μία για κάθε τομέα που ήθελα να εκπαιδεύονται οι χρήστες. Επομένως υπάρχει μία δραστηριότητα μνήμης, μία δραστηριότητα παρατηρητικότητας-προσοχής, μία δραστηριότητα λόγου και τέλος, μία

δραστηριότητα ταχύτητας επεξεργασίας. Επιπλέον τα activities για κάθε δραστηριότητα είναι δημιουργημένα με την ίδια λογική: πατώντας το κουμπί για κάθε δραστηριότητα υπάρχει πρώτα ένα Activity στο οποίο ο χρήστης βλέπει ένα κείμενο που περιγράφει την εκάστοτε δραστηριότητα και τί πρέπει να κάνει ανάλογα την δραστηριότητα που πρέπει να δοκιμάσει και ένα κουμπί έναρξης το οποίο μόλις πατηθεί, μεταφέρεται ο χρήστης στην δραστηριότητα.

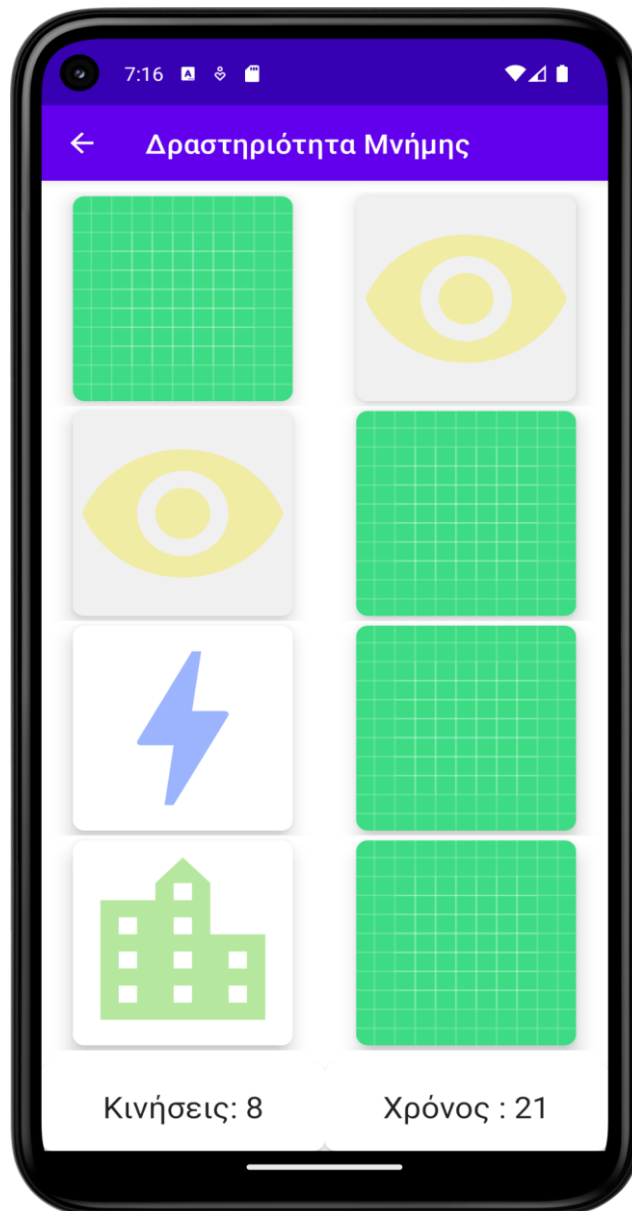
5.3.1 Δραστηριότητα Μνήμης

Η πρώτη δραστηριότητα στοχεύει στην κατηγορία της μνήμης, που θεωρείται από τις σημαντικότερες αν όχι η σημαντικότερη κατηγορία, καθώς οι περισσότερες παθήσεις των νευρογνωστικών διαταραχών έχουν σαν αποτέλεσμα την σταδιακή αποδυνάμωση της ικανότητας του πάσχοντα να θυμάται. Ειδικότερα, η δραστηριότητα αυτή έχει ως σκοπό την προπόνηση του στην βραχυπρόθεσμη μνήμη.

Σαν βασική λειτουργία, η δραστηριότητα παρουσιάζει ένα παιχνίδι μνήμης, το οποίο σαν δραστηριότητα είναι αρκετά γνωστή. Υπάρχουν διάφορα image buttons, κουμπιά τα οποία το καθένα έχει και μια εικόνα, τα οποία αρχικά εμφανίζονται γυρισμένα. Τα κουμπιά αυτά έχουν ένα σχέδιο και το κάθε σχέδιο εμφανίζεται δύο φορές, δηλαδή υπάρχει ένα μοναδικό ζεύγος για την κάθε εικόνα. Σκοπός της εφαρμογής είναι ο παίκτης να βρει τα ζευγάρια με όσο το δυνατόν λιγότερες κινήσεις, οι οποίες εμφανίζονται στην οθόνη, καθώς και στον λιγότερο δυνατό χρόνο, που εμφανίζεται και αυτός στην οθόνη.

Παρόλο που η δραστηριότητα φαίνεται αρκετά απλή σαν σκέψη, η υλοποίηση της ήταν η δυσκολότερη συγκριτικά με τις άλλες τρεις. Αυτό έγκειται στο γεγονός ότι έπρεπε να αρχικοποιηθούν σωστά οι κάρτες, η οθόνη στην οποία τρέχει η δραστηριότητα καθώς και να υλοποιηθεί σωστά η λογική που αναποδογυρίζει τις κάρτες και να ελεγχθεί αν ο χρήστης ανακάλυψε κάποιο ζευγάρι ή όχι.

Ξεκινώντας από τις κάρτες, δημιουργήθηκε ένα data class ονόματι MemoryCard, και μέσω αυτού αρχικοποιήθηκε ότι χρειάζεται μία από τις κάρτες. Πιο συγκεκριμένα, κάθε κάρτα έχει ένα identifier, για να ξεχωρίζει από τις υπόλοιπες και δυο τιμές boolean, όπου η isFaceup ελέγχει σε ποιά πλευρά είναι η κάρτα ενώ η isMatched εάν έχει βρεθεί το ζευγάρι της.



Εικόνα 5.6 Δραστηριότητα μνήμης

Συνεχίζοντας θα πρέπει να αναφερθούμε στο πως αρχικοποιείται η οθόνη στην οποία πραγματοποιείται η δραστηριότητα. η οθόνη, η αλλιώς το board υλοποιείται με τον κάτωθι κώδικα:

```
override fun onCreateView(parent: ViewGroup, viewType: Int): ViewHolder {
    val cardWidth=parent.width/2 - (2* MARGIN_SIZE)
    val cardHeight =parent.height/4 - (2* MARGIN_SIZE)
    val cardSideLength =min (cardWidth,cardHeight)
    val view = LayoutInflater.from(context).inflate(R.layout.memory_card,parent,false)
```

```

        val layoutParams= view.findViewById<CardView>(R.id.cardView).layoutParams as
        ViewGroup.MarginLayoutParams
        layoutParams.width=cardSideLength
        layoutParams.height=cardSideLength
        layoutParams.setMargins(MARGIN_SIZE,      MARGIN_SIZE,      MARGIN_SIZE,
        MARGIN_SIZE)
        return ViewHolder(view)
    }

```

Αρχικά παρατηρούμε την τιμή `MARGIN_SIZE`, η οποία έχει μία απόλυτη και αυθαίρετη τιμή και ο σκοπός της υπάρχει για να δημιουργεί αποστάσεις στα όρια της οθόνης, καθώς και ανάμεσα στις κάρτες, όπως φαίνεται παρακάτω. Οι τιμές `cardWidth` και `cardHeight` χωρίζουν την οθόνη σε κομμάτια, και η δραστηριότητα αυτήν χωρίζεται σε έναν πίνακα 2×4 , για να υπάρχει μία ομοιομορφία και συμμετρία ανάμεσα στις κάρτες. Οι επόμενες δύο γραμμές του κώδικα χρησιμοποιούνται για να μεταφέρουμε τα δεδομένα που έχουμε δώσει στο `MemoryCard` στην οθόνη και να θέσουμε τα όρια κάθε κάρτας σαν αυτά που έχουμε αρχικοποιήσει στο xml του `MemoryCard`.

Τέλος θα αναφέρουμε τον κώδικα που χρησιμοποιείται όσο τρέχει η δραστηριότητα. Τα δύο βασικά κομμάτια κώδικα είναι τα:

```

init {
    val images=(DEFAULT_ICONS + DEFAULT_ICONS).shuffled()
    cards=images.map { MemoryCard(it) }
},

```

στο οποίο το πρόγραμμα παίρνει δεδομένα από την λίστα `DEFAULT_ICONS`, η οποία περιέχει τα σχήματα που χρησιμοποιούνται στο παιχνίδι, και αφού τα ανακατέψει, αντιστοιχεί μία εικόνα σε κάθε κουμπί, ενώ το δεύτερο κομμάτι κώδικα:

```

fun flipCard(position: Int):Boolean {
    CardFlips++
    val card= cards[position]
    var foundMatch =false
    if(indexOfSingleSelectedCard==null) {
        restoreCards()
        indexOfSingleSelectedCard = position
    }
}

```

```

    }else{
        foundMatch= checkforMatch(indexOfSingleSelectedCard!! , position)
        indexOfSingleSelectedCard=null
    }
    card.isFaceup= !card.isFaceup
    return foundMatch
},

```

ασχολείται με την λογική της δραστηριότητας. Ειδικότερα, χρησιμοποιείται η τιμή card και η boolean τιμή foundMatch. Με το που μία κάρτα γυρίσει χρησιμοποιείται η τιμή foundMatch, η οποία βλέπει την συνάρτηση checkforMatch. Η συνάρτηση checkforMatch παίρνει σαν όρισμα τις δύο θέσεις των καρτών που είναι γυρισμένες, και εάν έχουν το ίδιο identifier τότε η boolean τιμή τους isMatched γίνεται true και η συνάρτηση επιστρέφει true, ενώ αν δεν είναι ίδια επιστρέφει false. Ανεξάρτητα από εάν ο παίκτης μάντεψε σωστά η λανθασμένα, στο τρίτο συνεχόμενο πάτημα το πρόγραμμα χρησιμοποιεί την συνάρτηση restoreCards(), η οποία επιστρέφει στην αρχική τους θέση, δηλαδή γυρίζει, τις κάρτες που η isMatched τιμή τους είναι false. Εάν ο παίκτης προσπαθήσει να πατήσει ένα ήδη γυρισμένο κουμπί, τότε θα εμφανιστεί ένα μήνυμα που θα τον ειδοποιεί ότι δεν μπορεί να πατήσει ένα ήδη πατημένο κουμπί και το πάτημα αυτό δεν μετρείται ούτε για τον μετρητή CardFlips, που μετράει πόσες φορές μια κάρτα έχει γυρίσει, ούτε για την συνάρτηση restoreCards. Τέλος, σε ένα κομμάτι κώδικα που δεν αναφέρεται εδώ, όταν ο χρήστης βρει όλα τα ζευγάρια, το πρόγραμμα σταματάει να μετράει τα δευτερόλεπτα και επιστρέφει στην προηγούμενη οθόνη με την πάροδο ενός δευτερολέπτου.

5.3.2 Δραστηριότητα Παρατηρητικότητας-Προσοχής



Εικόνα 5.7 Δραστηριότητα Παρατηρητικότητας-Προσοχής

Ένας ακόμα τομέας που οι φροντιστές συνήθως ελέγχουν και προσφέρουν δραστηριότητες στους ασθενείς του είναι ο τομέας της προσοχής. Όπως αναφέραμε και στο 1ο κεφάλαιο ο τομέας αυτός αποτελείται από υποκατηγορίες, με μερικές από αυτές να είναι η σύνθετη προσοχή, η επιλεκτική προσοχή και η ταχύτητα επεξεργασίας.

Η δραστηριότητα αυτή έχει σκοπό να ελέγξει την σύνθετη προσοχή, καθώς και την ταχύτητα επεξεργασίας. Η ιδέα της δραστηριότητας είναι ότι εμφανίζονται 4 εικόνες στον παίκτη, και θα πρέπει να καταλάβει ποια από τις 4 εικόνες είναι αταίριαστη με τις υπόλοιπες 3. Όταν ο χρήστης αποφανθεί ποια εικόνα του φαίνεται αταίριαστη τότε απλά πατάει το κουμπί της εικόνας και μετά το κουμπί υποβολής. Εάν η εικόνα είναι η σωστή τότε το φόντο θα γίνει πράσινο και θα πάει στο επόμενο σετ εικόνων, ενώ αν είναι λάθος πάλι θα πάει στο επόμενο σετ αλλά πρώτα το φόντο θα γίνει κόκκινο και θα φωτιστεί η σωστή απάντηση. Επίσης υπάρχει και μία γραμμή προόδου, η οποία δείχνει σε ποιά ερώτηση βρίσκεσαι καθώς και τις συνολικές ερωτήσεις που έχει η δραστηριότητα

Σε αυτήν την δραστηριότητα υπήρχαν αρκετοί τρόποι προσέγγισης, με έναν από αυτούς να είναι 4 Imageviews, 4 εικόνες πρακτικά, και κάτω από κάθε εικόνα να υπάρχει ένα radio button, αλλά μετά από πειραματισμούς αποφασίστηκε να χρησιμοποιηθούν 4 Image Buttons, 4 κουμπιά δηλαδή που το καθένα φέρει από μία εικόνα για χάριν απλοποίησης του κώδικα, καθώς και να γίνει η περιοχή που ο παίκτης χρειάζεται να πατήσει μεγαλύτερη, για να είναι πιο βολικό και εύκολο σε άτομα προχωρημένης ηλικίας.

Η διάταξη της εφαρμογής έχει ελάχιστη πολυπλοκότητα και είναι δομημένη γύρω από τρία linear layouts, τα δυο σε οριζόντιο προσανατολισμό και το άλλο σε κάθετο. Το πρώτο linear layout είναι σε οριζόντιο προσανατολισμό και έχει τον ρόλο του φόντου, καθώς η κυριότερη λειτουργία του είναι να περικλείει τα άλλα δύο layouts καθώς και το κουμπί της υποβολής της ερώτησης. Το δεύτερο κατά σειρά είναι κάθετο και μέσα περικλείεται ένα progressbar και ένα TextView, με το πρώτο να είναι ένας οπτικός οδηγός του ποσοστού της εφαρμογής που έχει διεκπεραιώσει ο χρήστης και το TextView δείχνει με αριθμούς πόσες ερωτήσεις έχει απαντήσει ο χρήστης και πόσες είναι οι συνολικές. Τέλος το τρίτο linear layout είναι σε οριζόντια μορφή, και ο σκοπός του είναι το να περιέχει τα 4 image buttons και να τα κρατάει σε μία συμμετρική απόσταση καθώς και σε ένα συμμετρικό μέγεθος.

Όσον αφορά τον κώδικα της εφαρμογής, αποτελείται από δύο καίρια σημεία. Το πρώτο είναι οι λίστες που έχουν τα δεδομένα και το δεύτερο είναι το πώς χειρίζεται η εφαρμογή όταν ο παίκτης κλικάρει ένα από τα κουμπιά. ο τρόπος που έχω διαχωρίσει τα δεδομένα είναι ότι υπάρχουν 5 λίστες, μία για κάθε ερώτηση που έχω υλοποιήσει, στις οποίες μέσα υπάρχουν 4 εικόνες, μία για κάθε κουμπί. Επιπλέον, υπάρχει και ακόμα μια λίστα, στην

οποία είναι αποθηκευμένες οι σωστές απαντήσεις και επειδή οι λίστες είναι δομημένες με θέσεις, οι απαντήσεις είναι σε μορφή Integer, ο οποίος αντιστοιχεί στην θέση της σωστής απάντησης στην λίστα. Παρακάτω αναφέρεται και ο αντίστοιχος κώδικας:

```
private val buttonCombinations: List<Array<Int>> = listOf(  
    arrayOf(R.drawable.ic_1a, R.drawable.ic_1b, R.drawable.ic_1c, R.drawable.ic_1d ),  
    arrayOf(R.drawable.ic_2a, R.drawable.ic_2b, R.drawable.ic_2c, R.drawable.ic_2d),  
    arrayOf(R.drawable.ic_3a, R.drawable.ic_3b, R.drawable.ic_3c, R.drawable.ic_3d),  
    arrayOf(R.drawable.ic_4a, R.drawable.ic_4b, R.drawable.ic_4c, R.drawable.ic_4d),  
    arrayOf(R.drawable.ic_5a, R.drawable.ic_5b, R.drawable.ic_5c, R.drawable.ic_5d)  
)
```

```
private val correctAnswers: List<Int> = listOf(3, 1, 0, 3, 2),
```

οπού τα R.drawable.ic_1a, ic_1b κλπ. είναι οι εικόνες που χρησιμοποιώ στην δραστηριότητα.

Όλες οι εικόνες που χρησιμοποιήθηκαν πηγάζουν από το ίδιο το Android Studio και είναι διάφορα εικονίδια που μπορεί κάποιος να παρατηρήσει σε διάφορες εφαρμογές Android. Αφότου η δραστηριότητα κάνει τις αρχικοποιήσεις που χρειάζεται για τα buttons και τις μπάρες προόδου χρησιμοποιεί την συνάρτηση setCombination, η οποία έχει ως σκοπό να ελέγχει εάν υπάρχουν επόμενες ερωτήσεις, και εάν υπάρχουν να αντιστοιχεί εικόνες σε κουμπιά ενώ αν δεν υπάρχουν να τερματίζει την δραστηριότητα. Όταν ο παίκτης πιέσει κάποιο από τα κουμπιά, τότε το πρόγραμμα "κλειδώνει" την απάντηση του και, αφότου πιεστεί το κουμπί "ΥΠΟΒΟΛΗ", ελέγχει για την ορθότητα της απάντησης.

Ο έλεγχος αυτός γίνεται με την συνάρτηση checkAnswer και ο κώδικας της είναι ο κάτωθι:

```
private fun checkAnswer() {  
    val selectedAnswerIndex = imageButtons.indexOfFirst { it.isSelected }  
    val correctAnswerIndex = correctAnswers[currentCombinationIndex]  
    if (selectedAnswerIndex == correctAnswerIndex) {  
        savedProgress[currentCombinationIndex] = true  
        imageButtons[selectedAnswerIndex].setBackgroundColor(Color.GREEN)  
        prog +=1  
    } else {  
        savedProgress[currentCombinationIndex] = false  
        imageButtons[selectedAnswerIndex].setBackgroundColor(Color.RED)  
    }  
}
```

```

        imageButtons[correctAnswerIndex].setBackgroundColor(Color.GREEN)
        prog +=1
    }
    imageButtons.forEach { button ->
        button.isEnabled = false
    }
    imageButtons[0].postDelayed({
        imageButtons[correctAnswerIndex].setBackgroundColor(Color.WHITE)
        imageButtons[selectedAnswerIndex].setBackgroundColor(Color.WHITE)
        currentCombinationIndex = (currentCombinationIndex + 1)
        val currentprogress=(currentCombinationIndex)*100/5
        progressbar.progress=currentprogress
        progress.text= " $prog / $total"
        setCombination()
    }, 1500)
}

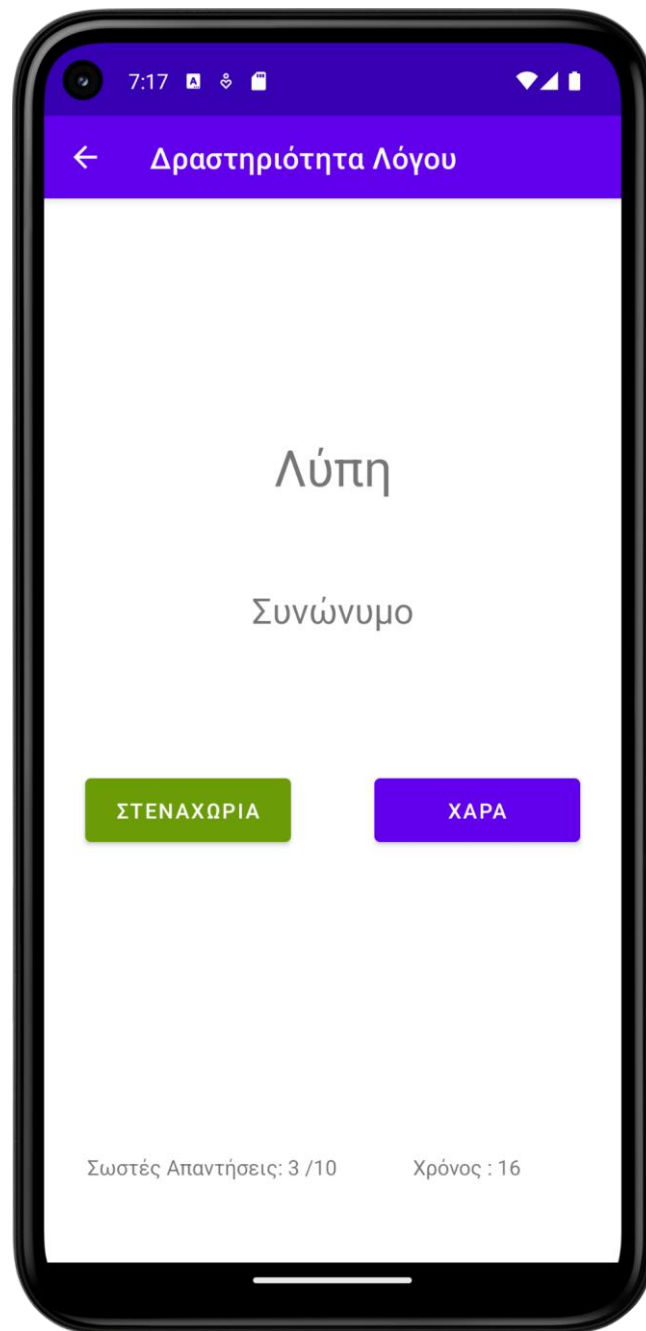
```

Το value `selectedAnswerIndex` είναι πρακτικά το κουμπί που επέλεξε ο χρήστης, ενώ το `correctAnswerIndex` είναι το σωστό κουμπί. Εάν και τα δύο είναι τα ίδια τότε όταν με το που πατηθεί το κουμπί "ΥΠΟΒΟΛΗ" τότε το φόντο της απάντησης γίνεται πράσινο, ενώ αν είναι διαφορετικά το φόντο της απάντησης κοκκινίζει και πρασινίζει η σωστή απάντηση. Ανεξάρτητα της απάντησης υπάρχει ο ακέραιος `prog`, ένας μετρητής ο οποίος υπάρχει για το progress bar, ώστε να δείχνει σωστά το ποσοστό διεκπεραίωσης της εφαρμογής. Τέλος, οι τελευταίες γραμμές του κώδικα υπάρχουν για να επαναφέρουν τα χρώματα του φόντου, να ετοιμάσουν το επόμενο σετ από εικόνες και να προχωρήσουν την πρόοδο του progress bar, και χρησιμοποιώ το `.postDelayed` σαν μια συνάρτηση wait ώστε να μην γίνει η αλλαγή ακαριαία, αλλά αντιθέτως μετα από ένα χρονικό διάστημα που επιλέγεται αυθαίρετα, και φαίνεται στον κώδικα με τον αριθμό 1500, η αλλιώς 1.5 δευτερόλεπτα.

5.3.3 Δραστηριότητα Λόγου

Ακόμα ένας σημαντικός τομέας που παρατηρούνται προβλήματα στους πάσχοντες των νευρογνωστικών διαταραχών και αναφέρθηκε στο 1ο κεφάλαιο είναι ο γλωσσικός τομέας. Επομένως θεωρήθηκε σημαντικό να προσφέρεται και μία δραστηριότητα αυτού του

τομέα για τους ασθενείς που υστερούνται σε αυτόν. Όπως και στις υπόλοιπες κατηγορίες, υπάρχουν διάφορες υποκατηγορίες, με την δραστηριότητα θα εμβαθύνει στην εύρεση λέξεων κατά κύριο στόχο και λιγότερο στην γραμματική και σύνταξη.



Εικόνα 5.8 Δραστηριότητα Λόγου

Η βασική ιδέα της δραστηριότητας είναι να εμφανίζονται διάφορες λέξεις στην οθόνη και κάτω από αυτές τις λέξεις να εμφανίζεται είτε η λέξη "συνώνυμο" είτε η λέξη "αντώνυμο". Επίσης υπάρχουν δύο κουμπιά, όπου το ένα περιέχει ένα συνώνυμο της λέξης, ενώ το

άλλο ένα αντώνυμο. Με το που ο χρήστης πατήσει ένα από τα δύο, το πρόγραμμα του εμφανίζει αν η επιλογή ήταν σωστή ή λάθος και προχωράει στην επόμενη λέξη.

Όπως και στις προηγούμενες δραστηριότητες, έτσι και σε αυτήν υπήρχαν αρκετές αρχικές σκέψεις για την υλοποίηση, με μία από αυτές απλά να δίνεται η λέξη και να ζητάται από τον χρήστη είτε να γράψει ο ίδιος ένα συνώνυμο είτε ένα αντώνυμο. Παρόλο που με αυτόν τον τρόπο θα ελεγχόταν σε μεγαλύτερο βαθμό η εύρεση των λέξεων, επιλέχθηκε τελικά μία πιο απλουστευμένη ιδέα της και να ελέγχω κατά κύριο λόγο την γραμματική του παίκτη.

Όσοι παρατηρήσουν τον κώδικα, θα παρατηρήσουν ότι χρησιμοποιήθηκε παρόμοια λογική με την Δραστηριότητα 2 όσον αφορά τον σχεδιασμό των λιστών, ενώ το βασικότερο σημείο της εφαρμογής είναι το πώς αντιλαμβάνεται το πρόγραμμα ποιά λέξη από τις δύο που εμφανίζονται στα κουμπιά είναι η σωστή ανάλογα την ερώτηση. Αυτό το αναλαμβάνει η συνάρτηση `checkAnswer` και ο σημαντικότερο κώδικας της είναι ο κάτωθι:

```
private fun checkAnswer(isSynonym: Boolean) {
    val correctType = corWorList[currentIndex]
    if (correctType == "Συνώνυμο" && isSynonym || correctType == "Αντώνυμο"
        && !isSynonym) {
        if (selectedAnswer == correctAnswer) {
            selectedButton.setBackgroundColor(correctColor)
            corrCounter +=1
            currentIndex += 1
        }
        if (currentIndex < endOfList) {
            Handler(Looper.getMainLooper()).postDelayed({
                selectedButton.setBackgroundColor(original_color)
                correctAnswersCounter.text="Σωστές Απαντήσεις: \${corrCounter} / \${endOfList}"
                updateButtonWord()
            }, 1000)
        }
    }
}
```

Αναλύοντας τον κώδικα βλέπουμε πρώτα την κλάση `isSynonym`, η οποία είναι `Boolean`, και χρησιμοποιείται στο `onClickListener` των κουμπιών, το πώς συμπεριφέρονται τα κουμπιά με το που τα επιλέξει ο χρήστης, με το `SynonymButton` να επιστρέφει `true` και το `AntonymButton` να επιστρέφει `false`. Αμέσως μετά παρατηρούμε την τιμή `correctType`, η οποία βλέπει στο `corWorList`, μία λίστα που κάθε κελί της περιέχει είτε την λέξη συνώνυμο, είτε την λέξη αντωνυμο.

Η γραμμή `if (correctType == "Συνώνυμο" && isSynonym || correctType == "Αντώνυμο" && !isSynonym)` υπάρχει για να ελέγξει εάν ο χρήστης πάτησε το σωστό κουμπί. Εάν έχει πατήσει το σωστό κουμπί τότε το φόντο του κουμπιού γίνεται πράσινο με το `selectedButton.setBackgroundColor(correctColor)`, και ανεβαίνουν κατά ένα οι δύο μετρητές που έχω για να εμφανίζονται στην οθόνη της δραστηριότητας, με τον πρώτο να είναι ο μετρητής των σωστών απαντήσεων και ο δεύτερος να είναι ο μετρητής που χρησιμοποιείται για να δείξω στο πρόγραμμα σε ποιά κελιά των λιστών να βλέπει. Η επόμενη λογική `if(currentIndex < endOfList){ } else{ }`, ο σκοπός της οποίας είναι να κοιτάει αν ο μετρητής έχει γίνει ίσος με την τιμή `endOfList`, η οποία είναι η ποσότητα των ερωτήσεων στην δραστηριότητα. Προτίμησα να χρησιμοποιώ την τιμή `endOfList` αντί κάποιου `Integer`, ώστε να είναι ευκολότερη η αλλαγή της τιμής, μιας και χρειάζεται απλά να αλλάξω τον αριθμό στο σημείο που την αρχικοποιώ, αντί να αλλάζω οποιοδήποτε σημείο χρειάζομαι το `endOfList`. Εάν δεν έχει φτάσει στο τέλος τότε μετα την πάροδο ενός δευτερολέπτου, το οποίο επιτυγχάνεται με τον `handler`, επιστρέφει το κουμπί στο αρχικό του χρώμα, ενημερώνει τις σωστές απαντήσεις στην οθόνη και καλεί την συνάρτηση `updateButtonWord()`, πού είναι υπεύθυνη για το ποιες λέξεις εμφανίζονται στην οθόνη, ενώ αν έχει φτάσει ειδοποιεί τον παίκτη ότι τελείωσε η δραστηριότητα και τερματίζει το πρόγραμμα μετά από ένα δευτερόλεπτο.

Σε παρόμοια λογική εάν ο χρήστης δεν πάτησε το σωστό κουμπί, τότε το φόντο γίνεται κόκκινο, ο μετρητής `corrCounter` δεν αυξάνεται, και υπάρχει η ίδια `if (currentIndex < endOfList) else { }` για να ελέγχεται εάν το πρόγραμμα έχει φτάσει στο τέλος του ή όχι.

5.3.4 Δραστηριότητα Ταχύτητας Επεξεργασίας

Η τελευταία δραστηριότητα εστιάζει στην κατηγορία των αντιληπτικών και κινητικών λειτουργιών και συγκεκριμένα στην υποκατηγορία του αντιληπτικού-κινητικού συντονισμού. Παρόλο που δεν είναι από τις βασικότερες κατηγορίες που εξετάζονται στις

συγκεκριμένες περιπτώσεις, θεωρήθηκε ευεργετικό να υπάρχει και μια δραστηριότητα αυτού του τομέα, για ποικιλία και για την περίπτωση που κάποιο άτομο έχει ανάγκη την δραστηριότητα αυτή.



Εικόνα 5.9 Δραστηριότητα Ταχύτητας Επεξεργασίας

Η δραστηριότητα αυτήν έχει ως σκοπό την εξέταση του ατόμου ως προς το πόσο γρήγορα αντιλαμβάνεται και αντιδρά. Για αυτόν τον λόγο η δραστηριότητα αυτήν έχει κουμπιά διαφορετικού χρώματος, και στο πάνω μέρος της οθόνης εμφανίζεται σαν λέξη ένα από τα χρώματα που αντιστοιχούν στα κουμπιά αυτά. Ο σκοπός της δραστηριότητας είναι ο

παίκτης να πατήσει όσο το δυνατόν περισσότερα και σωστά κουμπιά μπορεί στον χρόνο που του δίνεται.

Υπήρχαν αρκετές σκέψεις με ποιον τρόπο θα υλοποιούνταν η δραστηριότητα, με μία από αυτές να είναι σε ίδια λογική με την προηγούμενη δραστηριότητα, δηλαδή να υπάρχει μία λίστα ευσεβούς μεγέθους με χρώματα και να ελέγχεται εάν πατήθηκε το σωστό, μετρώντας το ποσοστό επιτυχίας και τον χρόνο που χρειάστηκε. Τελικά, η δραστηριότητα υλοποιήθηκε με μία λίστα χρωμάτων που εμφανίζονται τυχαία, ενώ υπάρχει ένα χρονόμετρο που μετράει αντίστροφα και δυο μετρητές, ένας των σωστών απαντήσεων και ένας των συνολικών απαντήσεων.

Πιο αναλυτικά, υπάρχει η λίστα `colorList`, στην οποία είναι αποθηκευμένα τα χρώματα και για κάθε χρώμα που αποθηκεύτηκε στην λίστα, δημιουργήθηκε και ένα κουμπί αντιστοίχου χρώματος. Τα επόμενα δύο κομμάτια κώδικα είναι αυτά που θεωρούνται είναι τα πιο σημαντικά στην δραστηριότητα αυτήν.

Το πρώτο είναι το:

```
private fun startTimer() {  
    DownTimer = object : CountdownTimer(countdownMillis, 1000) {  
        override fun onTick(millisUntilFinished: Long) {  
            val secondsLeft = millisUntilFinished / 1000  
            timer.text = "Υπολειπόμενος χρόνος: $secondsLeft seconds"  
        }  
        override fun onFinish() {  
            timer.text = "Τέλος Χρόνου!"  
            DownTimer.cancel()  
            Handler(Looper.getMainLooper()).postDelayed({  
                finish()  
            }, 1000)  
        }  
    }  
    DownTimer.start()  
},
```

στο οποίο υλοποιείται το αντίστροφο μέτρημα του χρόνου και με το που ο χρόνος φτάσει στο μηδέν σταματάει το μέτρημα και εξέρχεται ο χρήστης από την δραστηριότητα.

Το δεύτερο και σημαντικότερο κομμάτι κώδικα είναι το:

```
private fun checkAnswer(selectedColor: String) {  
    val displayedColor = colorTextView.text.toString()  
    if (selectedColor == displayedColor) {  
        correct +=1  
        total +=1  
        corAnsw.text="Σωστές Απαντήσεις: $correct / $total"  
    }else {  
        total+=1  
        corAnsw.text="Σωστές Απαντήσεις: $correct / $total"  
    }  
    showRandomColor()  
},
```

στο οποίο υλοποιείται η λογική. Πιο συγκεκριμένα κάθε κουμπί αντιστοιχεί σε ένα χρώμα και κάθε κουμπί δίνει σαν string το χρώμα αυτό όταν πατιέται, με την βοήθεια του selectedColor. Στην συνέχεια έχουμε το displayedColor, που μετατρέπει σε string το χρώμα που εμφανίζεται στην οθόνη με το colorTextView. Από εκεί και πέρα η λογική είναι αρκετά απλή, καθώς το μόνο που χρειάζεται είναι να συγκρίνουμε τα δυο αυτά strings αν είναι ίδια και εάν είναι ίδια να αυξήσουμε τις σωστές και συνολικές απαντήσεις, με την βοήθεια των μετρητών correct και total, ενώ αν είναι λάθος αυξάνεται μόνο ο μετρητής total. Ανεξάρτητα από το εάν ο παίκτης επέλεξε σωστά ή όχι καλούμε την συνάρτηση showRandomColor() η οποία παίρνει με το val randomColor τυχαία ένα από τα χρώματα που έχουμε βάλει στο val colorList και το εμφανίζει με την βοήθεια του colorTextView.

5.3 Ολοκλήρωση της εφαρμογής

Με το πέρας της υλοποίησης των δραστηριοτήτων και την διόρθωση των bugs που δημιουργήθηκαν κατά τον αρχικό κώδικα στην εφαρμογή, έπρεπε να διορθωθούν τυχόν συντακτικά και γραμματικά λάθη που υπήρχαν στην αρχική συγγραφή των κειμένων, όπως και αλλαγή κάποιων από των κειμένων με σκοπό να είναι πιο κατανοητά σε άτομα μεγαλύτερων ηλικιών. Τέλος έπρεπε να δημιουργηθεί ένα .apk αρχείο, το οποίο επιτρέπει σε άτομα που έχουν λογισμικό Android να εγκαταστήσουν και να τρέξουν την

εφαρμογή στο κινητό τους. Η δημιουργία του αρχείου αυτού είναι εύκολη, καθώς το Android Studio κάνει την δουλειά αυτή αυτόματα, μέσω των εντολών Build > Build Bundle(s)/APK(s) > Build APK(s).

Επιπλέον, ο κώδικας υπάρχει ανεβασμένος στο GitHub. Ο σύνδεσμος για τον κώδικα είναι:

https://github.com/EfthymiosDimou/Diploma_app

5.4 Μελλοντικές Υλοποιήσεις

Παρόλο που η εφαρμογή είναι υλοποιημένη σε ένα σημαντικό βαθμό, υπάρχουν αρκετά περιθώρια βελτίωσής της σε αρκετούς από τους τομείς της.

Πιο συγκεκριμένα, η εφαρμογή κατά την είσοδο της ζητάει το όνομα του χρήστη και το χρησιμοποιεί στο activities main screen. Μια βασική μελλοντική υλοποίηση θα μπορούσε να είναι να ζητάται το όνομα σε μορφή χρήστη, δηλαδή να υπάρχει μια ολοκληρωμένη υλοποίηση ενός login και sign up, καθώς και τις βάσεις δεδομένων που θα χρειάζεται το πρόγραμμα για να αποθηκεύει αυτές τις πληροφορίες.

Επιπλέον, παρόλο που η εφαρμογή εκτελείται ήδη από 4 δραστηριότητες, θα ήταν θεμιτό να υπάρξει εμπλουτισμός της ποσότητας των δραστηριοτήτων, έτσι ώστε να υπάρχει μεγαλύτερη ποικιλία από δραστηριότητες, οι οποίες θα εξετάζουν και διαφορετικές κατηγορίες ή υποκατηγορίες των νευρογνωστικών διαταραχών. Αυτό θα αυξήσει εκτός των άλλων και την δυνατότητα επανάληψης αναπαραγωγής της δραστηριότητας.

Αναφερόμενοι στην δυνατότητα επανάληψης, παρόλο που οι δραστηριότητες που έχουν ήδη υλοποιηθεί είναι ολοκληρωμένες και πλήρως λειτουργικές, έχουν και οι ίδιες περιθώρια βελτίωσης. Αναφορικά, σε όλες θα μπορούσαν να προσφέρονται διαφορετικά επίπεδα δυσκολίας, και στο τέλος της κάθε εφαρμογής να υπάρχει μια οθόνη η οποία θα δείχνει πιο αναλυτικά το σκορ. Επιπλέον, θα ήταν θεμιτό να υπάρχει μια βάση δεδομένων στην οποία θα αποθηκεύονται τα σκορ αυτά και θα εμφανίζονται στην καρτέλα του εκάστοτε χρήστη, ώστε να είναι ευκολότερο τόσο για τους ίδιους τους χρήστες όσο και για τους φροντιστές τους να παρακολουθούν την πρόοδο του.

Συνεχίζοντας, κάθε δραστηριότητα θα μπορούσε να βελτιωθεί σε ειδικότερο επίπεδο. Για την δραστηριότητα Μνήμης, θα μπορούσε να υπάρχουν περισσότερες καρτέλες και περισσότερα σχήματα, ώστε να γίνεται πιο απαιτητικό στον χρήστη να θυμάται τις θέσεις των σχημάτων.

Στην δραστηριότητα Προσοχής-Παρατηρητικότητας, θα μπορούσε να υπάρχουν περισσότερες ερωτήσεις, για να ανεβαίνει το επίπεδο δυσκολίας ώστε να αντιλαμβάνεται ο χρήστης την εικόνα που δεν ταιριάζει με τις υπόλοιπες, ή ακόμα και να υπάρχει ένας αντίστροφος μετρητής σε κάθε ερώτηση που να πρέπει να απαντήσει πριν φτάσει ο μετρητής στο μηδέν.

Στην δραστηριότητα Λόγου, μια μελλοντική υλοποίηση θα ήταν κάθε ερώτηση να ζητάει τυχαία ένα συνώνυμο ή ένα αντώνυμο αντί να «κοιτάει» μία λίστα για το τι πρέπει να ζητήσει, ή ακόμα και κάθε ερώτηση να λαμβάνεται από μια τυχαία θέση της λίστας από το να ζητάται με γραμμική σειρά.

Τέλος, στην δραστηριότητα Ταχύτητας Επεξεργασίας, θα μπορούσαν να υλοποιηθούν περισσότερα κουμπιά και χρώματα, καθώς και να ζητούνται παραπάνω από ένα χρώμα κάθε φορά ώστε ο χρήστης να τα πατήσει με την σειρά που δίνονται.

Βιβλιογραφία

1. Παπαευθυμίου Α, Τσίγκα Π. Ο ρόλος του λογοθεραπευτή στην διαχείριση των διαταραχών επικοινωνίας & σίτισης σε ασθενείς με μείζονες νευρογνωστικές διαταραχές, Πανεπιστήμιο Ιωαννίνων, 2020
2. Management of Neuropsychiatric Symptoms in Neurocognitive Disorders, Benalfew Legesse, M.D., Baktash Babadi, M.D., Ph.D., Brent Forester, M.D., M.Sc
3. Ganguli, M., Blacker, D., Blazer, D. G., Grant, I., Jeste, D. V., Paulsen, J. S., Sachdev, P. S. (2011). Classification of Neurocognitive Disorders in DSM-5: A Work in Progress. The American Journal of Geriatric Psychiatry, 19(3), 205–210. doi:10.1097/jgp.0b013e3182051ab4
4. Sachdev, P. S., Blacker, D., Blazer, D. G., Ganguli, M., Jeste, D. V., Paulsen, J. S., & Petersen, R. C. (2014). *Classifying neurocognitive disorders: the DSM-5 approach*. *Nature Reviews Neurology*, 10(11), 634–642. doi:10.1038/nrneurol.2014.181
5. https://el.wikipedia.org/wiki/Νόσος_Αλτσχάιμερ
6. Rabinovici GD, Miller BL. Frontotemporal lobar degeneration: epidemiology, pathophysiology, diagnosis and management. *CNS Drugs*. 2010;24(5):375-398. doi:10.2165/11533100-000000000-00000
7. <https://en.wikipedia.org/wiki/HIV>
8. https://en.wikipedia.org/wiki/Dementia_with_Lewy_bodies
9. https://en.wikipedia.org/wiki/Parkinson%27s_disease
10. <https://en.wikipedia.org/wiki/Prion>
11. Backlund, P., & Hendrix, M. (2013). Educational games - Are they worth the effort? A literature survey of the effectiveness of serious games. 2013 5th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES). doi:10.1109/vs-games.2013.6624226
12. K. Salen and E. Zimmerman, Rules of play: Game design fundamentals. MIT press, 2003.
13. M. Zyda, 'From visual simulation to virtual reality to games', Computer, vol. 38, no. 9, pp. 25–32, 2005.
14. Petri, Giani & Gresse von Wangenheim, Christiane. (2016). How to Evaluate Educational Games: a Systematic Literature Review. *Journal of Universal Computer Science*. 22. 992.
15. Alevan, V., Myers, E., Easterday, M., & Ogan, A. (2010). Toward a Framework for the Analysis and Design of Educational Games. 2010 Third IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning. doi:10.1109/digitel.2010.55

16. Nebel, C. (2020). Considerations for Applying Six Strategies for Effective Learning to Instruction. *Medical Science Educator*, 30(S1), 9–10.
<https://doi.org/10.1007/s40670-020-01088-8>
17. Collins, J., & O'Brien, N. (Eds.). (2003). *The Greenwood Dictionary of Education*. Westport, CT: Greenwood.
18. Michael, J. (2006). Where's the evidence that active learning works? *Advances in Physiology Education*, 30(4), 159–167. doi:10.1152/advan.00053.2006
19. Prince, M. (2004). Does Active Learning Work? A Review of the Research. *Journal of Engineering Education*, 93(3), 223–231. doi:10.1002/j.2168-9830.2004.tb00809.x
20. D'Zurilla, T. J., & Goldfried, M. R. (1971). Problem solving and behavior modification. *Journal of Abnormal Psychology*, 78(1), 107–126.
doi:10.1037/h0031360
21. McMurran, M., & McGuire, J. (Eds.). (2005). *Social Problem Solving and Offending*. doi:10.1002/9780470713488
22. D'Zurilla, T. J., & Goldfried, M. R. (1971). Problem solving and behavior modification. *Journal of Abnormal Psychology*, 78(1), 107–126.
doi:10.1037/h0031360
23. Spivack, G., Platt, J. J., & Shure, M. B. (1976). *The problem solving approach adjustment*. San Francisco, CA: Jossey-Bass.
24. MALOUFF, J., THORSTEINSSON, E., & SCHUTTE, N. (2007). The efficacy of problem solving therapy in reducing mental and physical health problems: A meta-analysis. *Clinical Psychology Review*, 27(1), 46–57. doi:10.1016/j.cpr.2005.12.005
25. D'Zurilla, T. J., & Nezu, A. M. (1999). *Problem-solving therapy: A social competence approach to clinical intervention*. (2nd ed.). New York
26. <https://www.dakim.com/>
27. Miller, K. et al. (2013) Effect of a Computerized Brain Exercise Program on Cognitive Performance in Older Adults.
28. <https://tactustherapy.com> <https://www.lumoslabs.com/about>
29. <https://www.lumosity.com/en/about/>
30. Hardy, J. L., Nelson, R. A., Thomason, M. E., Sternberg, D. A., Katovich, K., Farzin, F., Scanlon, M. (2015). Enhancing Cognitive Abilities with Comprehensive Training: A Large, Online, Randomized, Active-Controlled Trial. *PLOS ONE*, 10(9), e0134467. doi:10.1371/journal.pone.0134467
31. <https://constanttherapyhealth.com/science/technology/>
32. <http://www.scify.gr/site/el/about/scify>
33. <https://alzheimerathens.gr/etaireia/pioi-eimaste/>
34. <http://www.scify.gr/site/el/impact-areas/assistive-technologies/dianoia>
35. <https://en.wikipedia.org/wiki/Android-Studio>
36. <https://developer.android.com/studio/run/emulator>
37. <https://en.wikipedia.org/wiki/C++>

38. [https://en.wikipedia.org/wiki/Java\(programming-language\)](https://en.wikipedia.org/wiki/Java(programming-language))
39. [https://en.wikipedia.org/wiki/Kotlin-\(programming-language\)](https://en.wikipedia.org/wiki/Kotlin-(programming-language))
40. <https://en.wikipedia.org/wiki/LLVM>
41. <https://developer.android.com/guide/components/activities/intro-activities>
42. [https://en.wikipedia.org/wiki/Swift_\(programming language\)](https://en.wikipedia.org/wiki/Swift_(programming_language))
43. https://en.wikipedia.org/wiki/Apple_Developer