



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ  
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

“Σχεδιασμός και ανάπτυξη διαδικτυακής διαδραστικής εφαρμογής ανίχνευσης  
γνωστικών λειτουργιών παιδιών προσχολικής ηλικίας”

Καράγκος Ευάγγελος

A.M. 2113106

ΕΠΙΒΛΕΠΟΝ ΚΑΘΗΓΗΤΗΣ

Γεώργιος Σταμούλης

ΣΥΝΕΠΙΒΛΕΠΟΝΤΕΣ ΚΑΘΗΓΗΤΕΣ

Νικόλαος Ζυγούρης

Αντώνιος Δαδαλιάρης

Λαμία, Ιούλιος 2018

«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις (1), που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάστηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.

2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.

3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια

4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Θα ήθελα να εκφράσω τις ιδιαίτερες ευχαριστίες μου στον επιβλέποντα καθηγητή κ. Σταμούλη Γεώργιο και στους συνεπιβλέποντες καθηγητές κ. Νικόλαο Ζυγούρη και κ. Δαδαλιάρη Αντώνιο για την πολύτιμη καθοδήγηση και υποστήριξη τους καθόλη την διάρκεια εκπόνησης αυτής της εργασίας. Ακόμα θα ήθελα να ευχαριστήσω ιδιαίτερα την οικογένεια μου καθώς και τους φίλους μου για την στήριξή τους σε όλη την διάρκεια των ακαδημαϊκών σπουδών μου με κάθε δυνατό τρόπο.

# ΠΕΡΙΕΧΟΜΕΝΑ

|  |    |
|--|----|
| 1. ΕΙΣΑΓΩΓΗ.....                           | 5  |
| 1.1 Στόχος της πτυχιακήςεργασίας.....      | 5  |
| 1.2 Δομή της πτυχιακής εργασίας.....       | 5  |
| 2. ΜΕΛΕΤΗ.....                             | 7  |
| 2.1 Ανάπτυξη στην προσχολική ηλικία.....   | 7  |
| 2.1.1 Σωματική ανάπτυξη.....               | 7  |
| 2.1.2 Τύποι μνήμης.....                    | 8  |
| 2.1.3 Γνωστικές λειτουργίες.....           | 9  |
| 2.2 Μεθοδολογία.....                       | 10 |
| 2.2.1 Ερευνητικές υποθέσεις.....           | 10 |
| 2.2.2 Συμμετέχοντες.....                   | 11 |
| 2.2.3 Διαδικτυακή εφαρμογή ανίχνευσης..... | 11 |
| 3. ΣΧΕΔΙΑΣΜΟΣ.....                         | 13 |
| 3.1 Προδιαγραφές.....                      | 13 |
| 3.2 Τεχνολογίες.....                       | 14 |
| 3.2.1 HTML.....                            | 14 |
| 3.2.2 CSS.....                             | 18 |
| 3.2.3 JavaScript.....                      | 19 |
| 3.2.4 PHP.....                             | 20 |
| 3.2.4 MySQL.....                           | 22 |
| 3.3 Εμφάνιση.....                          | 23 |
| 4. Ανάπτυξη της εφαρμογής.....             | 24 |
| 4.1 Βάση Δεδομένων.....                    | 24 |
| 4.2 Διεπαφή.....                           | 29 |
| 4.2.1 Γενικά.....                          | 29 |
| 4.2.2 Δοκιμασία 1.....                     | 33 |
| 4.2.3 Δοκιμασία 2.....                     | 34 |
| 4.2.4 Δοκιμασία 3.....                     | 34 |
| 4.2.5 Δοκιμασία 4.....                     | 36 |
| 4.2.6 Δοκιμασία 5.....                     | 37 |
| 4.2.7 Δοκιμασία 6.....                     | 37 |
| 4.2.8 Δοκιμασία 7.....                     | 38 |
| 4.3 Κώδικας.....                           | 39 |
| 4.3.1 Δοκιμασία 1.....                     | 39 |
| 4.3.2 Δοκιμασία 2.....                     | 41 |
| 4.3.3 Δοκιμασία 3.....                     | 43 |
| 4.3.4 Δοκιμασία 4.....                     | 46 |
| 4.3.5 Δοκιμασία 5.....                     | 48 |
| 4.3.6 Δοκιμασία 6.....                     | 50 |
| 4.3.7 Δοκιμασία 7.....                     | 52 |
| 5. ΑΠΟΤΕΛΕΣΜΑΤΑ.....                       | 54 |
| 6. ΣΥΜΠΕΡΑΣΜΑΤΑ.....                       | 57 |
| ΒΙΒΛΙΟΓΡΑΦΙΑ.....                          | 59 |

## ΕΙΚΟΝΕΣ

|  |    |
|--|----|
| Εικόνα 1. LAMP structure.....  | 16 |
| Εικόνα 2. HTML5 logo.....  | 17 |
| Εικόνα 3. Στοιχείο τύπου img.....  | 18 |
| Εικόνα 4. Στοιχείο τύπου button.....   | 18 |
| Εικόνα 5. Στοιχείο τύπου table.....  | 19 |
| Εικόνα 6. Στοιχείο τύπου video.....  | 20 |
| Εικόνα 7. CSS3 logo.....   | 21 |
| Εικόνα 8. JavaScript logo.....   | 23 |
| Εικόνα 9. Φόντο δοκιμασιών.....  | 27 |
| Εικόνα 10. Οθόνη εισόδου.....  | 35 |
| Εικόνα 11. Μενού εφαρμογής.....  | 36 |
| Εικόνα 12. Οθόνη ονόματος.....   | 36 |
| Εικόνα 13. Οθόνη φύλου.....  | 37 |
| Εικόνα 14. Οθόνη ηλικίας.....  | 37 |
| Εικόνα 15. Αρχική οθόνη δοκιμασιών.....  | 38 |
| Εικόνα 16. Βίντεο οδηγιών.....   | 38 |
| Εικόνα 17. Οθόνη έναρξης δοκιμασιών.....   | 39 |
| Εικόνα 18. Δοκιμασία 1-Πρώτη κατάσταση.....                                      | 40 |
| Εικόνα 19. Δοκιμασία 1-Δεύτερη κατάσταση.....                                    | 40 |
| Εικόνα 20. Δοκιμασία 3-Βιβλίο.....   | 40 |
| Εικόνα 21. Δοκιμασία 3-Μολύβι.....   | 40 |
| Εικόνα 22. Δοκιμασία 3-ψαλίδι.....   | 40 |
| Εικόνα 23. Δοκιμασία 3-Φάση πρώτη, εμφάνιση του μήλου 5 από την λίστα (5,3)..... | 41 |
| Εικόνα 24. Δοκιμασία 3-Φάση πρώτη, εμφάνιση του μήλου 3 από την λίστα (5,3)..... | 41 |
| Εικόνα 25. Δοκιμασία 3-Οθόνη δεύτερης φάσης.....                                 | 41 |
| Εικόνα 26. Δοκιμασία 3-Φάση δεύτερη, εισαγωγή αριθμών.....                       | 41 |
| Εικόνα 27. Δοκιμασία 4-Οθόνη πρώτης φάσης.....                                   | 42 |
| Εικόνα 28. Δοκιμασία 4-Οθόνη δεύτερης φάσης.....                                 | 43 |
| Εικόνα 29. Δοκιμασία 4-Φάση δεύτερη, εισαγωγή αριθμών.....                       | 43 |
| Εικόνα 30. Δοκιμασία 5-Οθόνη δεύτερης φάσης.....                                 | 44 |
| Εικόνα 31. Δοκιμασία 6-Οθόνη δεύτερης φάσης.....                                 | 45 |
| Εικόνα 32. Οθόνη δοκιμασίας 7.....   | 46 |
| Εικόνα 33. Γράφημα 1.....  | 64 |

## ΠΙΝΑΚΕΣ

|                               |    |
|-------------------------------|----|
| Πίνακας-1,Questions_1.....    | 29 |
| Πίνακας-2,Questions_2.....    | 29 |
| Πίνακας-3,Questions_3.....    | 30 |
| Πίνακας-4,Questions_4.....    | 31 |
| Πίνακας-5,Αποτελέσματα-1..... | 63 |
| Πίνακας-6,Αποτελέσματα-2..... | 65 |

# 1. ΕΙΣΑΓΩΓΗ

## 1.1 Στόχος της πτυχιακής εργασίας

Στόχος της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη μιας διαδραστικής εφαρμογής με την χρήση της οποίας θα γίνει συλλογή πληροφοριών για τις γνωστικές λειτουργίες παιδιών προσχολικής ηλικίας. Βασικές προδιαγραφές που θα πρέπει να ικανοποιεί η σχεδίαση της εφαρμογής είναι η ομαλή χρήση και η εύκολη κατανόηση από παιδιά μικρής ηλικίας, να χρησιμοποιεί τον ελάχιστο αριθμό πόρων για την βέλτιστη απόδοση και να είναι δυνατή η μελλοντική επεκτασιμότητα καθώς και η τροποποίηση της. Ένα πολύ σημαντικό στοιχείο της εφαρμογής είναι η δημιουργία λογισμικού ανεξάρτητο πλατφόρμας, έτσι για την ανάπτυξη χρησιμοποιήθηκαν τεχνολογίες όπως HTML, CSS, JavaScript, PHP, MySQL. Στην σχεδίαση αρχικά έγιναν κάποιες υλοποιήσεις, αλλά η ανατροφοδότηση που λάβαμε από παιδιά που χρησιμοποίησαν την εφαρμογή οδήγησε σε αλλαγές, σε βελτιστοποιήσεις και στην τελική σχεδίαση μετά τις κατάλληλες τροποποιήσεις. Δημιουργήθηκε μια βάση δεδομένων με σκοπό να καλύψει τις τωρινές ανάγκες του λογισμικού αλλά και πιθανές μελλοντικές προσθήκες. Με γνώμονα όλα αυτά δημιουργήθηκαν 7 δοκιμασίες που στόχο έχουν να συλλέξουν πληροφορίες σχετικά με την οπτική μνήμη, την ακουστική μνήμη, την γραφοφωνολογική διάκριση, την φωνολογική ενημερότητα και την οπτικοχωρική αντίληψη των παιδιών προσχολικής ηλικίας. Μετά την υλοποίηση της εφαρμογής έγινε μια μελέτη με την χορήγηση των δοκιμασιών σε ένα δείγμα παιδιών και συλλογή σημαντικών αποτελεσμάτων.

## 1.2 Δομή της πτυχιακής εργασίας

Στο **κεφάλαιο 2** περιγράφεται η ανάπτυξη των παιδιών προσχολικής ηλικίας, παρουσιάζονται οι υποθέσεις και εξηγείται η μεθοδολογία που χρησιμοποιήθηκε για την διεξαγωγή της μελέτης.

Στο **κεφάλαιο 3** παρουσιάζονται οι προδιαγραφές της εφαρμογής μας, αναλύονται οι τεχνολογίες που χρησιμοποιήθηκαν με όλα τα χαρακτηριστικά τους και σχολιάζεται η εμφάνιση της εφαρμογής αυτής.

Στο **κεφάλαιο 4** γίνεται αναλυτική παρουσίαση της ανάπτυξης της εφαρμογής. Περιγράφεται η βάση δεδομένων με όλα τα χαρακτηριστικά της, παρουσιάζεται η διεπαφή των δοκιμασιών και τέλος σχολιάζεται ο κώδικας για την υλοποίηση της εφαρμογής αυτής.

Στο **κεφάλαιο 5** παρουσιάζονται και αναλύονται τα αποτελέσματα από την μελέτη που έγινε με την χρήση της εφαρμογής.

Στο **κεφάλαιο 6** αναλύονται τα συμπεράσματα που προέκυψαν και αναφέρονται πιθανές μελλοντικές προεκτάσεις.

## **2. ΜΕΛΕΤΗ**

### **2.1 Ανάπτυξη στην προσχολική ηλικία**

Στην προσχολική ηλικία το παιδί διανύει μία από τις πιο κρίσιμες φάσεις της ζωής του. Θα μπορούσε να πει κανείς ότι είναι μια φάση προετοιμασιών και προσδοκιών για την ένταξη στην επίσημη σχολική εκπαίδευση. Παρ' όλα αυτά όταν ακούμε προσχολική ηλικία δεν πρέπει να το εκλαμβάνουμε κυριολεκτικά διότι η φάση της ζωής του παιδιού αυτή δεν είναι σε καμία περίπτωση μια μεταβατική περίοδος. Μεγάλες αλλαγές στην σωματική και στην γνωστική ανάπτυξη του συμβαίνουν σε σύντομο χρονικό διάστημα, γεγονός που δεν θα πρέπει να παραβλέπεται.

#### **2.1.1 Σωματική ανάπτυξη**

Αν σκεφτεί κανείς ότι τα παιδιά αυτά πριν μερικά χρόνια δεν μπορούσαν ούτε να περπατήσουν βλέπει ότι έχουν πολύ μεγάλη πρόοδο στις σωματικές τους δυνατότητες σε τόσο σύντομο χρονικό διάστημα.

Το σώμα των παιδιών αναπτύσσεται ραγδαία, σε ηλικία 2 ετών το παιδί προβλέπεται να ζυγίζει περίπου 11-13 κιλά με ύψος 90-92 εκατοστά ενώ στα 6 έτη νά ζυγίζει 21 κιλά περίπου με ύψος 115 εκατοστά. Στην ηλικία των 2 ετών η διαφορά βάρους και ύψους ανάμεσα σε αγόρια και κορίτσια είναι μικρή, αρχίζει να μεγαλώνει στην προσχολική ηλικία, με τα αγόρια να είναι ψηλότερα και βαρύτερα. Διαφορές υπάρχουν ακόμα και ανάμεσα σε παιδιά από οικονομικά πιο ανεπτυγμένες χώρες. Η καλύτερη φαρμακευτική περίθαλψη και διατροφή έχει ως αποτέλεσμα μεγαλύτερη ανάπτυξη, ένα παιδί που ζει στην Σουηδία και είναι στην ηλικία 4 ετών υπολογίζεται ότι έχει το ύψος ενός παιδιού 6 ετών που ζει στο Μπαγκλαντές (United Nations, 1991· Leathers & Foster, 2004). Ακόμα σύμφωνα με έρευνες στις Ηνωμένες Πολιτείες, παιδιά από οικογένειες που ζουν κάτω από το όριο της φτώχειας είναι πιο πιθανόν να είναι ασυνήθιστα κοντά σε σχέση με παιδιά που μεγαλώνουν σε ευκατάστατες οικογένειες (Barrett & Frank, 1987· Ogden et al., 2002).

Κατά την διάρκεια της προσχολικής ηλικίας τα παιδιά καίνε το βρεφικός τους λίπος, αυξάνεται η μυϊκή τους δύναμη, τα πόδια και τα χέρια μακραίνουν και οι αναλογίες του



σώματος τους μοιάζουνε με αυτών των ενηλίκων.

Ο εγκέφαλος είναι το πιο ταχύτατα αναπτυσσόμενο τμήμα του σώματος, στα 2 έτη το βάρος του εγκεφάλου του παιδιού είναι το 75% περίπου του βάρους ενός ενήλικα σε σύγκριση με την ηλικία των 5 ετών όπου το βάρος του εγκεφάλου του παιδιού είναι το 90% του εγκεφάλου του μέσου ενήλικα (Schuster & Ashburn, 1986· Lowrey, 1986· Nihart, 1993).

Αρχίζουν να εμφανίζονται διαφοροποιήσεις στα δύο ημισφαίρια του εγκεφάλου, η εγκεφαλική πλευρίωση γίνεται όλο και πιο αισθητή με τον διαμοιρασμό κάποιων λειτουργιών σε ένα από τα δυο ημισφαίρια. Στην πλειοψηφία των παιδιών, για λειτουργίες όπως ανάγνωση και ομιλία χρησιμοποιείται κυρίως το αριστερό ημισφαίριο, ενώ το δεξί ημισφαίριο χρησιμοποιείται σε δυνατότητες όπως η κατανόηση σχημάτων και σχεδίων (McAuliffe & Knowlton, 2001· Koivisto & Revonsuo, 2003· Pollak, Holt, & Wismer Fries, 2004).

Στην διάρκεια της προσχολικής φάσης τα παιδιά αποκτούν και τελειοποιούν τις κινητικές τους δεξιότητες. Αυτό συμβαίνει γιατί αποκτούν καλύτερο μυϊκό έλεγχο. Στα 3 έτη μπορούν να ανεβαίνουν μόνα τους σκάλες, στα 4 έτη κάνουν μεγάλα άλματα και στα 5 έτη σταματούν και ξεκινούν την κίνηση τους με μεγάλη αποτελεσματικότητα (Clark & Humphrey, 1985).

Επιπλέον τα παιδιά κατακτούν και κάποιες λεπτές κινητικές δεξιότητες σε αυτή την περίοδο, όπως την ικανότητα να ζωγραφίσουν, να χρησιμοποιήσουν ένα μολύβι, να κόψουν με ένα ψαλίδι καθώς και να είναι σε θέση να εκτελέσουν εντολές με την χρήση του ποντικού ενός ηλεκτρονικού υπολογιστή.

### **2.1.2 Τύποι μνήμης**

Οι διάφοροι τύποι μνήμης, σύμφωνα με τους Shiffrin & Atkinson, είναι η αισθητηριακή, η βραχύχρονη και η μακρόχρονη. Οι διαφορές τους εμφανίζονται κυρίως στον χρόνο που χρειάζεται για να διατηρηθεί η πληροφορία που λήφθηκε. Η αισθητηριακή μνήμη χωρίζεται στην κιναισθητική, την οπτική και την ακουστική. Σχετίζεται με τις αισθήσεις και οι πληροφορίες που λαμβάνονται από αυτήν διατηρούνται

εως 1,5 δευτερόλεπτο. Η βραχύχρονη μνήμη έχει μικρή χωρητικότητα και οι πληροφορίες που αποθηκεύονται σε αυτήν διατηρούνται για περίπου 30 δευτερόλεπτα ή και λεπτά ανάλογα με τον αριθμό επαναλήψεων του ερεθίσματος. Για παράδειγμα χρησιμοποιείται στην απομνημόνευση αριθμών ή πληροφοριών κατά την διάρκεια μιας διαδικασίας(πχ το τεστ μας) ή συζήτησης. Το πλήθος των πληροφοριών που μπορούν να διατηρηθούν στην μνήμη αυτή διαφέρει από άνθρωπο σε άνθρωπο. Η μακρόχρονη μνήμη είναι ο μεγαλύτερος χώρος αποθήκευσης πληροφοριών του ανθρώπου. Μπορεί να αποθηκευτεί ένας σημαντικά πολύ μεγάλος αριθμός πληροφοριών με μία εξήσου μεγάλη διάρκεια ζωής. Το 1996, διακρίθηκε ακόμη μια μορφή μνήμης, η οποία προήλθε από τον Beddeley. Ο ρόλος της είναι να μας βοηθήσει να παραμείνουμε σε κάτι συγκεντρωμένο για μεγαλύτερο χρονικό διάστημα καθώς και να κρατάμε στο μυαλό μας περισσότερες πληροφορίες όταν εκτελούμε μια νοητική εργασία. Υπάρχουν δύο τύποι, η ακουστική και η οπτικοχωρική εργαζόμενη μνήμη. Η ακουστική εργασιακή μνήμη χρησιμοποιείται από τα παιδιά για να κατανοήσουν ένα πρόβλημα που θα τους δοθεί. Τα παιδιά χρησιμοποιούν την οπτικοχωρική εργαζόμενη μνήμη για να θυμούνται εικόνες, ακολουθίες στοιχείων. (Ζυγούρης, 2017)

### **2.1.3 Γνωστικές λειτουργίες**

Μαζί με την ανάπτυξη του εγκεφάλου του παιδιού προσχολικής ηλικίας συμβαίνουν και αλλαγές σχετικά με τις γνωστικές λειτουργίες του. Η γνωστική ανάπτυξη των παιδιών εμφανίζεται με αλλαγές σε λειτουργίες όπως μνήμη, γλώσσα, αντίληψη, σκέψη και άλλες. (Στασινός, 2009)

Με τον όρο αντίληψη εννοούμε τις διαδικασίες τις οποίες χρειάζεται να εκτελέσει ο εγκέφαλος του παιδιού για να προσλάβει, οργανώσει και αναγνωρίσει τα ερεθίσματα του περιβάλλοντος. Τα ερεθίσματα που υπάρχουν στο περιβάλλον είναι άγνωστα για τον ανθρώπινο οργανισμό. Έτσι η αντίληψη βασίζεται στον διαχωρισμό των κοινών στοιχείων που έχουν τα αντικείμενα. Μια κύρια μορφή των αντικειμένων διατηρείται στην μνήμη και προστίθενται έπειτα σε αυτήν τα διαφορετικά χαρακτηριστικά τους. (Hayes, 1998, Σαμαρτζή, 1995)

Η μνήμη ως γνωστική λειτουργία βοηθάει το παιδί να συγκρατεί τις ποικίλες

πληροφορίες που αρχικά προσλαμβάνει και έπειτα επεξεργάζεται ώστε να γίνουν πιθανές γνώσεις. Έτσι οι γνώσεις που διαθέτουμε στηρίζονται κατά ένα βαθμό στην μνημονική μας ικανότητα, η οποία είναι εμφανής σε διαδικασίες μάθησης. Οι οπτικές ή ακουστικές πληροφορίες λαμβάνονται ως ερέθισμα από τους αντίστοιχους υποδοχείς. Έπειτα μεταφέρονται στην βραχύχρονη μνήμη, η οποία ως εργαζόμενη τις χρησιμοποιεί για τις ανάλογες διαδικασίες. Διακρίνονται τρεις κύριες λειτουργίες της μνήμης. Η κωδικοποίηση, η αποθήκευση και η ανάσυρση. Για την διατήρηση μιας πληροφορίας αρχικά χρειάζεται η κωδικοποίηση των βασικών στοιχείων της, έπειτα η αποθήκευση τους και τέλος η ανάσυρσή τους όταν χρειαστεί ξανά. (Στασινός, 2009, Craig & Baucum, 2007, Sternberg, 2007, Siegler, 2002)

Η σκέψη είναι μια πολύπλοκη διαδικασία η οποία στηρίζεται στην αναπαραγωγή και στον συνδιασμό πληροφοριών που αποτελούν γνώσεις κάποιου. Η σκέψη εκτελείται με την ανάσυρση πληροφοριών από την μνήμη, την συσχέτιση τους και την λήψη αποφάσεων. Τα παιδιά μπορούν να σκέφτονται μόνο ένα πράγμα την φορά. Όταν ο εγκέφαλος έχει λάβει ένα ερέθισμα οπτικό ή ακουστικό δεν ασχολείται με τα υπόλοιπα ζητήματα εκείνη την συγκεκριμένη στιγμή. Έτσι τα παιδιά αντιλαμβάνονται μόνο ότι έχει σχέση με το ερέθισμα που την προκάλεσε. (Russell & Dewar, 1992)

Η γλώσσα αποτελεί ένα απαραίτητο κομμάτι των γνωστικών λειτουργιών καθώς μέσω της οποίας τα παιδιά λαμβάνουν καινούργιες γνώσεις. Χρησιμοποιείται για την κατανόηση αλλά και παραγωγή των πληροφοριών της. Σύμφωνα με μελέτες η κατάκτηση της γλώσσας παρουσιάζει μια έμφυτη τάση, η οποία ανάλογα τις συνθήκες και το περιβάλλον διαμορφώνεται ανάλογα. (Βότσος, 1992, Sternberg, 2007)

Η κατανόηση των αριθμών είναι μία από τις αλλαγές στις γνωστικές ικανότητες των παιδιών προσχολικής ηλικίας, καθώς φαίνεται να έχουν μια ανεπτυγμένη αλλά όχι τελείως ακριβής κατανόηση των αριθμών. Η πλειοψηφία των παιδιών στα 4 έτη μπορούν να κατανοήσουν τους αριθμούς, να ξεχωρίσουν ότι ένα αντικείμενο είναι μια μονάδα, να λύσουν απλά μαθηματικά προβλήματα με απλές πράξεις όπως η πρόσθεση και η αφαίρεση καθώς και να εκτελέσουν συγκρίσεις μεταξύ συνόλων με διαφορετικά αντικείμενα (Donlan, 1998).

Η γλωσσική ανάπτυξη εμφανίζει βαθμιαία βελτίωση κατά την ανάπτυξη του παιδιού. Η κατάκτηση της γλώσσας είναι μια σύνθετη διαδικασία γιατί δεν υπάρχει η συνεχόμενη διδασκαλία της. Το παιδί καλείται να γίνει σταδιακά κυρίαρχο της γλώσσας καθώς την χρησιμοποιεί από την αρχή της ζωής του με καθημερινή χρήση. Μαθαίνει την γλώσσα για να επικοινωνήσει και να εκφραστεί.

Η ικανότητα του παιδιού να αναγνωρίζει και να συνθέτει φωνήματα, συλλαβές και λέξεις ονομάζεται φωνολογική ενημερότητα. Ένα παιδί που έχει κατακτήσει και αναπτύξει κατάλληλα την φωνολογική ενημερότητα μπορεί να ξεχωρίσει ότι οι λέξεις αποτελούνται από ξεχωριστά γράμματα με τον δικό του ήχο το καθένα. τον και έχει ιδιαίτερη σημασία καθώς είναι σημαντικό να έχει κατακτηθεί πριν την διαδικασία της εκμάθησης της γραφής και της ανάγνωσης. (Ζακοπούλου, 2001, Παντελιάδου, 2000)

Στην προσχολική ηλικία εμφανίζονται και οι βασικές δομές της νοητικής λειτουργίας μαζί με την ανάπτυξη ενός μεγάλου μέρους της γενικής νοημοσύνης. Η ανάπτυξη της νοημοσύνης των παιδιών αυτής της ηλικίας καθορίζεται από τα ερεθίσματα τα οποία δίνονται σε αυτό.

## **2.2 Μεθοδολογία**

### **2.2.1 Ερευνητικές υποθέσεις**

Όπως προαναφέρθηκε βασικός στόχος εκπόνησης του παρόντος ερευνητικού πρωτοκόλλου ήταν η κατασκευή και πιλοτική εφαρμογή μιας συστοιχίας δοκιμασιών ανίχνευσης των γνωστικών λειτουργιών παιδιών προσχολικής ηλικίας που χορηγείται μέσω της χρήσης διαδικτύου. Η βασική ερευνητική υπόθεση ήταν ότι τα παιδιά προσχολικής ηλικίας που θεωρήθηκε από τους εκπαιδευτικούς προσχολικής εκπαίδευσης ότι παρουσιάζουν γνωστικές αδυναμίες θα παρουσιάσουν λιγότερες σωστές απαντήσεις σε σύγκριση με τα παιδιά που συμμετείχαν στην ομάδα ελέγχου στο σύνολο των δοκιμασιών της συστοιχίας. Επίσης, ερευνητική υπόθεση ήταν το γεγονός ότι τα παιδιά προσχολικής ηλικίας που θεωρήθηκε από τους εκπαιδευτικούς ότι παρουσιάζουν γνωστικές αδυναμίες θα χρειαστούν μεγαλύτερο χρόνο προκειμένου να ολοκληρώσουν τις δοκιμασίες σε σύγκριση με τα παιδιά της ομάδας ελέγχου.

### **2.2.2 Συμμετέχοντες**

Στην παρούσα έρευνα συμμετείχαν 79 παιδιά (43 αγόρια και 36 κορίτσια) ηλικίας 4 και 5 ετών που φοιτούσαν σε νηπιαγωγεία της Λαμίας. Από το σύνολο των παιδιών 17 παιδιά κρίθηκε σύμφωνα με τη γνώμη των εκπαιδευτικών ότι παρουσίαζαν ελαφριά αναπτυξιακή αδυναμία σε σύγκριση με τα υπόλοιπα 62. Στο σημείο αυτό αξίζει να αναφερθεί το γεγονός ότι για το σύνολο των συμμετεχόντων τηρήθηκε απόλυτη ανωνυμία και η διεξαγωγή της έρευνας πραγματοποιήθηκε αυστηρά με την τήρηση των κανόνων της συνθήκης του Helsinki για έρευνα σε ανθρώπους.

### **2.2.3 Διαδικτυακή εφαρμογή ανίχνευσης.**

Για την ανίχνευση των γνωστικών λειτουργιών των παιδιών κατασκευάστηκαν και χορηγήθηκαν 7 δοκιμασίες. Αναλυτικότερα, στα παιδιά χορηγήθηκε μία δοκιμασία εξάσκησης κατά την οποία έπρεπε να επιλέξουν ένα αντικείμενο. Στη συνέχεια χορηγήθηκε μία δοκιμασία ανίχνευσης της εκτελεστικής ικανότητας, στην οποία έπρεπε να επιλέξουν από ένα πλήθος αντικειμένων που παρουσιάζονταν τυχαία ένα συγκεκριμένο αντικείμενο – στόχο. Για την αξιολόγηση της μνήμης χορηγήθηκε μία δοκιμασία αξιολόγησης της εργαζόμενης οπτικής μνήμης και μία δοκιμασία αξιολόγησης της εργαζόμενης ακουστικής μνήμης. Στις δύο δοκιμασίες έβλεπαν και άκουγαν αντίστοιχα σειρές αριθμών. Στη συνέχεια έπρεπε να ανακαλέσουν από τη μνήμη τους τους αριθμούς που είχαν χορηγηθεί. Στο σημείο αυτό αξίζει να επισημανθεί ότι οι προαναφερθείσες δοκιμασίες ελέγχου της εργαζόμενης μνήμης διακόπτονταν μετά από δύο διαδοχικά λάθη των παιδιών. Στη συνέχεια χορηγήθηκε μία δοκιμασία γραφοφωνολογικής διάκρισης, κατά την οποία τα παιδιά έπρεπε να επιλέξουν ποιο γράφημα αντιστοιχεί στο φώνημα που είχαν ακούσει, μία δοκιμασία ανίχνευσης της φωνολογικής ενημερότητας, κατά την οποία οι συμμετέχοντες έπρεπε να εντοπίσουν την λέξη την οποία είχαν ακούσει. Στο σημείο αυτό αξίζει να επισημανθεί το γεγονός ότι παρουσιάστηκαν συχνόχρηστες δυσύλλαβες λέξεις της μορφής σ/φ/ ή φ/σ. Τέλος, χορηγήθηκε μία δοκιμασία οπτικοχωρικής αντίληψης κατά την οποία στα παιδιά παρουσιάζονταν ένα μοτίβο από το οποίο απουσίαζε ένα μέρος και έπρεπε να επιλέξουν ποιο μέρος από τα 3 πιθανά που τους παρουσιάζονταν ολοκλήρωνε το μοτίβο. Οι πληροφορίες που αφορούν τον προγραμματιστικό σχεδιασμό της εφαρμογής περιλαμβάνονται πλήρως στην αντίστοιχη υποενότητα.

Μερικά από τα πλεονεκτήματα χρήσης λογισμικών ηλεκτρονικού υπολογιστή για την διεξαγωγή τεστ αξιολόγησης σε σύγκριση με τα τεστ που υλοποιούνται με χαρτί και μολύβι είναι:

- Τα στοιχεία παρουσιάζονται σε όλους με τον ίδιο τρόπο
- Η δοκιμασία είναι εντελώς αντικειμενική
- Ακρίβεια στις μετρήσεις
- Καλύτερη διαχείριση
- Άμεσα αποτελέσματα
- Πιο ενδιαφέρον για τα παιδιά

## 3. ΣΧΕΔΙΑΣΜΟΣ

### 3.1 Προδιαγραφές

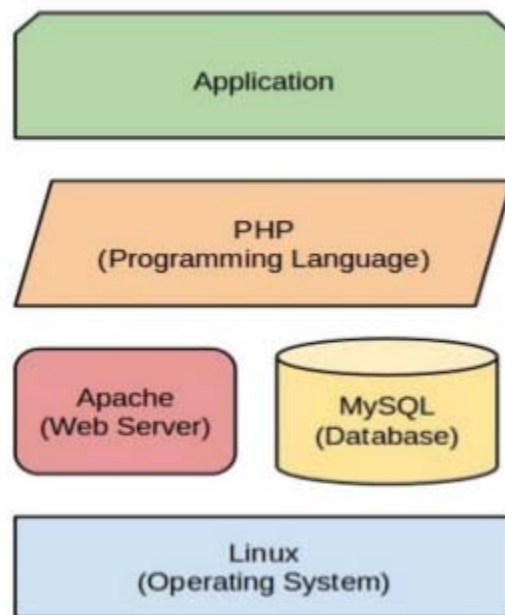
Για την σχεδίαση και την λειτουργία των δοκιμασιών επιλέξαμε να δημιουργήσουμε μια διαδικτυακή εφαρμογή η οποία θα φιλοξενείται σε κάποιον διακομιστή (server). Ο χρήστης (client) θα έχει πρόσβαση σε αυτή μέσω ενός προγράμματος περιήγησης ιστού (browser), με την χρήση ενός μοναδικού URL ή μια διεύθυνσης IP για την εμφάνιση του περιεχομένου της εφαρμογής. Οι λόγοι που επιλέξαμε να δημιουργήσουμε μια διαδικτυακή εφαρμογή και όχι μια τυπική εφαρμογή που χρειάζεται εγκατάσταση στον υπολογιστή του χρήστη είναι:

- Η ευκολία πρόσβασης στην εφαρμογή από οποιονδήποτε υπολογιστή με μοναδική προϋπόθεση την ύπαρξη ενός browser για την πρόσβαση στο διαδίκτυο.
- Η παροχή λογισμικού ανεξάρτητο πλατφόρμας (cross-platform), ο χρήστης μπορεί να χρησιμοποιεί Linux, macOS και Windows.
- Η απουσία διαδικασιών όπως λήψη και εγκατάσταση της εφαρμογής καθώς και αναβάθμιση στην νεότερη έκδοση της από τον χρήστη, ο οποίος απλά χρησιμοποιεί πάντα την τελευταία έκδοση της εφαρμογής.

Για την λειτουργία της εφαρμογής χρησιμοποιήθηκε το πακέτο προγραμμάτων LAMP το οποίο αποτελείται από τέσσερις δωρεάν και ανοιχτού κώδικα τεχνολογίες:

- Linux, είναι ένα λειτουργικό σύστημα το οποίο αναπτύσσεται από κοινότητες χρηστών σε όλο τον κόσμο καθώς είναι ένα ελεύθερο και ανοιχτού κώδικα λογισμικό. Ο ρόλος του είναι να παρέχει ένα περιβάλλον ικανό να υποστηρίξει όλες τις απαραίτητες εφαρμογές που χρειάζονται την δημιουργία και την παραμετροποίηση του web server.
- Apache, είναι ένας από τους κυριότερους web server που χρησιμοποιείται στο διαδίκτυο για την φιλοξενία ιστοσελίδων. Αναπτύσσεται και συντηρείται από ανοιχτές κοινότητες προγραμματιστών. Ο ρόλος του είναι να παρέχει ένα λογισμικό για την φιλοξενία της διαδικτυακής εφαρμογής και την εξυπηρέτηση των αιτημάτων των χρηστών προς αυτή.

- MySQL, είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων. Ο ρόλος του είναι να διατηρεί μια βάση δεδομένων με όλες τις απαραίτητες πληροφορίες της εφαρμογής.
- PHP, είναι μια γλώσσα προγραμματισμού γενικού σκοπού σχεδιασμένη για την δημιουργία εφαρμογών δικτύου. Ο ρόλος της είναι να στέλνει πληροφορίες από και προς τον server και να παρέχει το κατάλληλο περιεχόμενο προς εμφάνιση στον χρήστη.



Εικόνα 1. LAMP structure

Έτσι με την εγκατάσταση των παραπάνω λογισμικών σε έναν υπολογιστή εγκαθίστανται τα κατάλληλα εργαλεία μέσω των οποίων δημιουργείται ένας λειτουργικός web server στον οποίο μπορούν να συνδεθούν χρήστες (clients) μέσω δικτύου για την εκτέλεση των φιλοξενούμενων εφαρμογών.



## 3.2 Τεχνολογίες

### 3.2.1 HTML



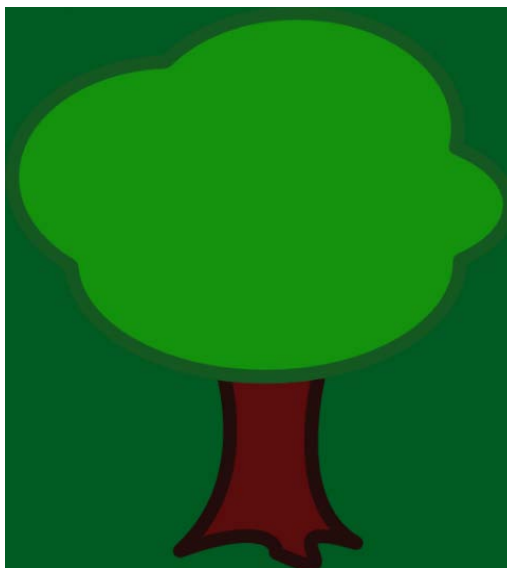
Εικόνα 2. HTML5 logo

Η HTML (Hyper Text Markup Language) είναι μια γλώσσα σήμανσης η οποία χρησιμοποιείται για την κατασκευή ιστοσελίδων. Χρησιμοποιείται για τον καθορισμό των βασικών/κύριων στοιχείων των διαδικτυακών εφαρμογών. Αποτελείται από στοιχεία τα οποία δομούν τον κορμό των σελίδων. Τα στοιχεία αυτά χαρακτηρίζονται από ετικέτες (tags) οι οποίες περιγράφουν τον τύπο του στοιχείου καθώς και το περιεχόμενο του. Ο

φυλλομετρητής (browser) λαμβάνει τα έγγραφα HTML αναλύει τα στοιχεία συνθέτει το περιεχόμενο τους και εμφανίζει την σελίδα όπως αυτή έχει καθοριστεί. Η HTML παρέχει στοιχεία όπως κείμενο, τίτλους, κεφαλίδες, εικόνες, παραγράφους, λίστες, φόρμες, κουμπιά και άλλα πολλά για την δημιουργία δομημένων σελίδων στον παγκόσμιο ιστό. Η νεότερη έκδοση, HTML5, έγινε διαθέσιμη το 2014 και είναι η βασικότερη τεχνολογία στην κατασκευή ιστοσελίδων. Εισάγει καινούργια στοιχεία και πολλές βελτιώσεις αλλά ενσωματώνει και όλα τα χαρακτηριστικά των προηγούμενων εκδόσεων. Παρέχει την ενσωματωμένη χρήση και διαχείριση πολυμέσων όπως βίντεο, ήχος και γραφικά κάτι πολύ σημαντικό καθώς δεν απαιτείται η χρήση εξωτερικών λογισμικών όπως συνέβαινε στις προηγούμενες εκδόσεις. (για αυτό fast Like js?). Μερικά από τα πιο σημαντικά στοιχεία της HTML, τα οποία χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής, είναι:

Το στοιχείο `<img>` χρησιμοποιείται για να καθορίσει μια εικόνα μέσα στον κώδικα της html. Έχει δύο απαραίτητες παραμέτρους:

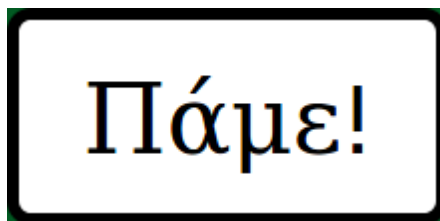
- `src`: που καθορίζει τον σύνδεσμο όπου βρίσκεται η εικόνα αποθηκευμένη στον server.
- `alt`: που αποτελεί μια λεκτική περιγραφή της εικόνας και του περιεχόμενου της.



Εικόνα 3. Στοιχείο τύπου *img*

Το στοιχείο `<button>` χρησιμοποιείται για να καθορίσει ένα κουμπί στον κώδικα της html. Το κουμπί αυτό μπορεί να περιέχει μια ετικέτα κειμένου ή ακόμα και μια εικόνα κάτι που το κάνει ιδιαίτερα χρήσιμο σε πολλές περιπτώσεις. Μερικές από τις παραμέτρους του είναι:

- `type`: καθορίζει τον τύπο του κουμπιού, κάτι που χρειάζεται να γνωρίζει ο browser για την σωστή δημιουργία του.
- `name`: καθορίζει ένα όνομα για το κουμπί.
- `value`: καθορίζει μια αρχική τιμή για το κουμπί η οποία μπορεί να χρησιμοποιηθεί σε κάποιο onclick event.



Εικόνα 4. Στοιχείο τύπου *button*

Το στοιχείο `<table>` χρησιμοποιείται για την δημιουργία ενός πίνακα σε μια σελίδα html. Ο πίνακας αυτός έχει έναν συγκεκριμένο αριθμό γραμμών και στηλών που μπορούν

να περιέχουν διάφορα αντικείμενα όπως κείμενο, εικόνες ή κουμπιά. Για να καθοριστούν οι στήλες και οι γραμμές του πίνακα, το στοιχείο `<table>` χρησιμοποιεί:

- `<tr>`: στοιχείο που καθορίζει τις γραμμές σε έναν πίνακα.
- `<td>`: στοιχείο που καθορίζει τις στήλες σε έναν πίνακα.
- `<th>`: στοιχείο που καθορίζει τις κελιά ως επικεφαλίδες σε έναν πίνακα.

|   |   |   |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

Εικόνα 5. Στοιχείο τύπου `table`

Το στοιχείο `<video>` δίνει την δυνατότητα στις ιστοσελίδες να προσθέσουν πολυμεσικό περιεχόμενο όπως βίντεο κλιπ. Για τον ορισμό της διεύθυνσης της πηγής του βίντεο χρησιμοποιείται το στοιχείο `<source>` που έχει ως παραμέτρους:

- `src`: καθορίζει τον σύνδεσμο όπου είναι αποθηκευμένο το βίντεο κλιπ στον server.
- `type`: καθορίζει την μορφή του βίντεο κλιπ που θα φορτωθεί.



Εικόνα 6. Στοιχείο τύπου video

Το στοιχείο `<audio>` επιτρέπει την προσθήκη κομματιών ήχου σε σελίδες. Για τον ορισμό της διεύθυνσης της πηγής του ηχητικού κομματιού χρησιμοποιείται το στοιχείο `<source>` που έχει ως παραμέτρους:

- `src`: καθορίζει τον σύνδεσμο όπου είναι αποθηκευμένο το κομμάτι ήχου στον server.
- `type`: καθορίζει την μορφή του ηχητικού κομματιού που θα φορτωθεί.

Το στοιχείο `<form>` καθορίζει μια φόρμα μέσα στην html ή οποία όταν αποστέλλεται μεταφέρει το περιεχόμενό της, το οποίο είναι ορισμένο/καθορισμένο από στοιχεία τύπου `input`, από τον πελάτη (client) στον διακομιστή (server). Μερικές από τις κύριες παραμέτρους του είναι:

- `action`: καθορίζει το που θα σταλούν τα στοιχεία της φόρμας όταν υποβληθεί.
- `method`: καθορίζει την μέθοδο του πρωτοκόλλου HTTP που θα χρησιμοποιηθεί για να γίνει αποστολή των πληροφοριών.

Το στοιχείο `<input>` περιέχεται μέσα σε φόρμες και καθορίζει τα στοιχεία στα οποία ο χρήστης μπορεί να εισάγει δεδομένα τα οποία θα σταλούν στον server. Μερικές από τις κύριες παραμέτρους του είναι:

- type: καθορίζει τον τύπο των δεδομένων που θα εισαχθούν. Μπορεί να είναι κείμενο, αριθμοί, ημερομηνία, κωδικοί, email κ.ά.
- name: χαρακτηρίζει το όνομα του στοιχείου.
- value: αρχικοποιεί μια τιμή στο στοιχείο.
- src: καθορίζει την διεύθυνση όταν το στοιχείο αυτό είναι εικόνα.

Το στοιχείο <div> χρησιμοποιείται για την επιλογή ενός συγκεκριμένο κομματιού στο κώδικα της HTML στο οποίο θέλουμε να εισάγουμε κάποια συγκεκριμένη μορφοποίηση με την χρήση της CSS ή για να καθορίσουμε ένα τμήμα στο οποίο θα συμβούν κάποια γεγονότα μέσω της JavaScript.

Όσον αφορά την εφαρμογή μας, με την χρήση της HTML καθορίζουμε την μορφή που θα έχει η εφαρμογή μας, τα κείμενα που θα χρησιμοποιηθούν, οι εικόνες, τα audio clips και τα video clips. Δημιουργούμε τα κουμπιά τα οποία χρειάζεται να εμφανίζονται στην οθόνη για να είναι λειτουργική η σελίδα μας. Ακόμα καθορίζουμε τις φόρμες οι οποίες θα στέλνουν δεδομένα στον server για αποθήκευση στην βάση δεδομένων.

### 3.2.2 CSS



Εικόνα 7. CSS3 logo

Η CSS (Cascading Style Sheets) είναι μια γλώσσα φύλλων στυλ που χρησιμοποιείται για την περιγραφή της μορφοποίησης ενός εγγράφου που έχει συνταχθεί σε γλώσσα σήμανσης όπως η HTML. Παρέχει ευέλικτο έλεγχο συγκεκριμένων παρουσιαστικών χαρακτηριστικών του περιεχομένου των ιστοσελίδων. Είναι σχεδιασμένη για να τροποποιεί τα χαρακτηριστικά της εμφάνισης των στοιχείων της σελίδας όπως τα χρώματα, το φόντο, την στοίχιση, την

γραμματοσειρά, το μέγεθος και την ορατότητα αυτών. Για την μορφοποίηση δημιουργούνται αρχεία με κατάληξη “.css” τα οποία είναι ξεχωριστά από τα αρχεία της HTML. Έτσι παρέχεται στον προγραμματιστή μια πολύ σημαντική δυνατότητα καθώς δεν χρειάζεται να γίνουν αλλαγές σε όλα τα αρχεία μιας εφαρμογής για την τροποποίηση της εμφάνισης της. Ακόμα η χρήση κοινών αρχείων CSS για την ομαδική μορφοποίηση όμοιων σελίδων μειώνει σε μεγάλο βαθμό τον χρόνο συντήρησης μιας ιστοσελίδας. Η

CSS3 που αποτελεί την τελευταία έκδοση, επεκτείνει τις δυνατότητες των προηγούμενων εκδόσεων εισάγοντας νέες δυνατότητες όπως σκιές, κινούμενες μεταβάσεις, μίξεις χρωμάτων, στρογγυλεμένες γωνίες και εικόνες πλαίσια. Ακόμα εισάγονται νέες διατάξεις όπως multi-columns, flexbox και grid layout. Κάποια από τα πιο σημαντικά χαρακτηριστικά της CSS, που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής είναι:

- Τα CSS Buttons όπου παρέχεται η δυνατότητα στον προγραμματιστή να τροποποιήσει τα προκαθορισμένα κουμπιά της HTML αλλάζοντας χαρακτηριστικά όπως το μέγεθος, το χρώμα, το σχήμα καθώς και διαφορά εφέ τα οποία μπορεί να έχει το κουμπί.
- Το CSS Background όπου τροποποιείται η εμφάνιση του παρασκήνιου της οθόνης είτε με την επιλογή κάποιου χρώματος ή με την εισαγωγή κάποιας εικόνας.
- Τα CSS Rounded Corners με την χρήση των οποίων αντικείμενα όπως κουμπιά, περιγράμματα κειμένων και αντικειμένων αποκτούν στρογγυλεμένες γωνίες.
- Το CSS Grid, που είναι ένα καινούριο στυλ διάταξης, προσφέρει την δυνατότητα δημιουργίας διάταξης πλέγματος και την αυτόματη στοίχιση των στοιχείων της σελίδας σε στήλες και γραμμές. Αυτό κάνει ευκολότερο τον σχεδιασμό σελίδων χωρίς να σπαταλάτε πολύ χρόνο από τον προγραμματιστή για την κατάλληλη στοίχιση του περιεχομένου της σελίδας.
- CSS Flexbox, που είναι ένα νέο στυλ διάταξης το οποίο κάνει ευκολότερη την δημιουργία για responsive web design σε σχέση με παλιότερες μεθόδους.

Ένας αρκετά συνηθισμένος τρόπος γραφής κώδικα CSS σε εφαρμογές είναι μέσω της χρήσης κάποιας εξωτερικής βιβλιοθήκης (framework) η οποία παρέχει κάποια έτοιμα στοιχεία και κάποιες έτοιμες δυνατότητες/λειτουργίες. Η χρήση όμως τέτοιων βιβλιοθηκών φορτώνει την λειτουργία των σελίδων και μπορεί να οδηγήσει σε προβλήματα καθυστέρησης. Για αυτό τον λόγο σε μικρές εφαρμογές, όπως η δική μας, συστήνεται η χρήση pure CSS.

Η HTML και η CSS είναι δύο αλληλένδετες τεχνολογίες καθώς συναντώνται μαζί στην πληθώρα των εφαρμογών του διαδικτύου. Θεωρείται αναπόσπαστο κομμάτι των διαδικτυακών σελίδων και εφαρμογών που δημιουργούνται με σκοπό να παρέχει κάποιου είδους μορφοποίηση για την παρουσίαση του περιχομένου προς τον τελικό χρήστη.

Με την χρήση της CSS στην εφαρμογή μας, μορφοποιούμε το περιεχόμενο που έχουμε καθορίσει με την HTML. Δίνουμε την κατάλληλη οπτική μορφή στα στοιχεία που έχει η σελίδα μας, την κατάλληλη διαμόρφωση αλλά και στοίχιση. Επίσης καθορίζουμε τα χρώματα που θα έχει γενικά η εφαρμογή μας, όπως για παράδειγμα το παρασκήνιο, αλλά και τα επιμέρους χρώματα των διαφόρων στοιχείων.

### 3.2.3 JavaScript



## JavaScript

Εικόνα 8. JavaScript logo

Η JavaScript είναι μια διερμηνευμένη υψηλού επιπέδου γλώσσα προγραμματισμού. Χρησιμοποιείται σε διαδραστικές ιστοσελίδες και για αυτό είναι ένα από τα απαραίτητα κομμάτια μιας διαδικτυακής εφαρμογής. Είναι μια γλώσσα που εκτελείται στον φυλλομετρητή (browser) στην πλευρά του χρήστη. Δίνει την δυνατότητα στον προγραμματιστή να επεξεργαστεί τα στοιχεία που υπάρχουν στην HTML, να δημιουργήσει καινούργια να διαγράψει τα παλιά και να τροποποιήσει τα χαρακτηριστικά τους όπως η εμφάνιση που έχει καθοριστεί μέσω της CSS. Σε συνδυασμό με την υποστήριξη συναρτήσεων, τις δυνατότητες αντικειμενοστρέφειας και τον χειρισμό γεγονότων η JavaScript αποτελεί ένα ισχυρό εργαλείο για την υλοποίηση πολύπλοκων λειτουργιών στον browser. Η JavaScript είναι μία από τις 3 κύριες τεχνολογίες του παγκόσμιου ιστού μαζί με την HTML και την CSS.

Με τον όρο Vanilla JavaScript εννοούμε την γλώσσα JavaScript χωρίς την χρήση κάποιου εξωτερικού framework ή επιπρόσθετων βιβλιοθηκών. Αυτό έχει ως αποτέλεσμα την παραγωγή γρήγορου κώδικα χωρίς μεγάλες ανάγκες για υπολογιστικούς πόρους και χωρίς εξαρτήσεις από άλλα λογισμικά.

Όταν φορτωθεί μια HTML σελίδα, ο browser δημιουργεί ένα HTML DOM (Document Object Model) μοντέλο. Το μοντέλο αυτό δίνει την δυνατότητα στην JavaScript να:

- Αλλάζει όλα τα HTML στοιχεία της εφαρμογής
- Αλλάζει όλα τα χαρακτηριστικά των HTML στοιχείων την εφαρμογής
- Αλλάζει τα στυλ μορφοποίησης που έχει καθορίσει η CSS

- Διαγράφει και να εισάγει HTML στοιχεία και χαρακτηριστικά
- Διαγράφει και να εισάγει στυλ μορφοποίησης τύπου CSS

Ο κώδικας της JavaScript μπορεί είτε να βρίσκεται εσωτερικά μαζί με τον κώδικα της HTML στο ίδιο αρχείο είτε εξωτερικά σε ξεχωριστό αρχείο. Τα αρχεία της JavaScript έχουν κατάληξη “.js”. Γενικά θεωρείται καλή τεχνική η διατήρηση του κώδικα σε εξωτερικά αρχεία, εκτός εξαιρέσεων, για λόγους καλύτερης διαχείρισης του αλλά και για κοινή χρήση σε όμοιες εφαρμογές με ίδιες λειτουργίες. Μια εξαίρεση στην οποία χρειάζεται να υπάρχει κώδικας μέσα σε αρχεία HTML είναι για την μεταφορά δεδομένων από την βάση δεδομένων που βρίσκεται στον server σε κάποια μεταβλητή της JavaScript για περαιτέρω επεξεργασία.

Με την χρήση της JavaScript διαχειριζόμαστε τα στοιχεία της εφαρμογής μας στα οποία επιθυμούμε να γίνουν κάποιες αλλαγές ανάλογα τις ανάγκες της κάθε δοκιμασίας. Δημιουργούμε καινούργια όπου χρειάζεται και σβήνουμε παλιά. Επίσης τροποποιούμε την ορατότητα στοιχείων κρύβοντας τα για μερικά δευτερόλεπτα ή μέχρι την έναρξη κάποιου γεγονότος.

### **3.2.4 PHP**

Η PHP (PHP: Hypertext Preprocessor) είναι μια ανοιχτού κώδικα (open source) γενικού σκοπού γλώσσα προγραμματισμού σχεδιασμένη για την δημιουργία εφαρμογών δικτύου. Ο κώδικας της PHP μπορεί να είναι ενσωματωμένος στον κώδικα της HTML ή να βρίσκεται σε εξωτερικό αρχείο. Ο τρόπος εκτέλεσης της γίνεται στη πλευρά του διακομιστή (server) με την χρήση του διερμηνέα της μηχανής PHP και το αποτέλεσμα επιστρέφεται στο πρόγραμμα περιήγησης σε μορφή HTML.

Χρησιμοποιείται για την υποστήριξη ιστοσελίδων δυναμικού περιεχομένου, την λήψη και το ανέβασμα αρχείων από και προς τον server καθώς και την επεξεργασία αυτών, για την επικοινωνία με βάσεις δεδομένων όπως η MySQL και για αποστολή ηλεκτρονικού ταχυδρομείου (e-mail).

Η τελευταία κύρια έκδοση είναι η PHP7. Τα καινούργια χαρακτηριστικά της έκδοσης



αυτής είναι, ο βελτιωμένος επανασχεδιασμός, της μηχανής PHPNG της PHP, προσφέροντας σημαντική βελτίωση στον χρόνο εκτέλεσης του κώδικα και στην χρήση της μνήμης, η καλύτερη διαχείριση των σφαλμάτων, καινούργιοι τελεστές, ανώνυμες κλάσεις και ακριβείς δηλώσεις τύπων.

Όταν ο χρήστης συμπληρώνει μια φόρμα στην HTML και την αποστέλλει, τα δεδομένα της φόρμας αυτής στέλνονται στην PHP για επεξεργασία μέσω είτε της μεθόδου GET ή POST. Στην αποστολή πληροφοριών με την μέθοδο GET οι πληροφορίες αυτές περνάνε ως παράμετροι στο URL. Έτσι είναι εμφανής στην μπάρα αναζήτησης του browser. Αντιθέτως στην αποστολή με την μέθοδο POST οι πληροφορίες αποστέλλονται μέσω HTTP αιτήματος και δεν μπορεί να τις δει ο χρήστης. Η PHP παρέχει έναν μηχανισμό δημιουργίας και διαχείρισης Session. Με την χρήση του αποθηκεύονται πληροφορίες κατά την περιήγηση μεταξύ των σελίδων της εφαρμογής. Οι πληροφορίες αυτές δεν αποθηκεύονται μόνιμα καθώς χάνονται με τον κλείσιμο του browser ή με την καταστροφή του Session. Η αποθήκευση τους γίνεται προσωρινά σε global μεταβλητές τύπου Session. Έτσι τα προγράμματα μπορούν να χρησιμοποιούν τις μεταβλητές αυτές για να αποθηκεύουν προσωρινά πληροφορίες που εισάγονται από τον χρήστη όπως το όνομα του, το φύλλο του ή τις προτιμήσεις του. Ένας ακόμα πολύ σημαντικός μηχανισμός της PHP είναι η σύνδεση της με βάσεις δεδομένων. Η PHP μπορεί να ανοίξει μια σύνδεση με μια βάση MySQL και να λάβει ή να στείλει δεδομένα προς αυτήν για αποθήκευση. Έτσι μπορούμε να καλύψουμε τις ανάγκες των εφαρμογών για μόνιμη αποθήκευση πληροφοριών.

Οι σημαντικότεροι λόγοι χρήσης της PHP είναι:

- Είναι μια πολύ σταθερή και γρήγορη γλώσσα που παρέχει ένα μεγάλο εύρος λειτουργιών στον προγραμματιστή.
- Είναι μια γλώσσα ανεξάρτητη πλατφόρμας καθώς μπορεί να εγκατασταθεί σε λειτουργικά (Linux, macOS, Windows).
- Ο συνδυασμός με τον Apache server προσφέρει ένα ολοκληρωμένο περιβάλλον για τις ανάγκες των σύγχρονων εφαρμογών.
- Είναι σχετικά εύκολη και παρέχοντα πολλά βοηθητικά διαδικτυακά έγγραφα με οδηγίες και παραδείγματα.

Με την χρήση της PHP στην εφαρμογή μας μεταφέρουμε πληροφορίες από την βάση δεδομένων μας προς στην σελίδα που βλέπει ο χρήστης. Πληροφορίες όπως λίστες με αριθμούς, γράμματα και λέξεις μεταφέρονται από την βάση για να παρουσιαστούν ως περιεχόμενο των δοκιμασιών που εκτελεί ο χρήστης. Ακόμα η χρήση της PHP μας δίνει την δυνατότητα να στέλνουμε πληροφορίες που εισάγει ή παράγει ο χρήστης, όπως τα στοιχεία που συμπληρώνει πριν την έναρξη των δοκιμασιών ή τα αποτελέσματα που προκύπτουν από αυτά, προς την βάση δεδομένων για μόνιμη αποθήκευση.

### 3.2.4 MySQL

Η MySQL είναι ένα ανοιχτού κώδικα (open source) σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων. Τα δεδομένα που αποθηκεύονται είναι δομημένα σε πίνακες, με στήλες ως χαρακτηριστικά και γραμμές που αντιπροσωπεύουν τις εγγραφές δεδομένων. Η SQL (Structured Query Language) είναι μια γλώσσα προγραμματισμού που χρησιμοποιείται για τον σχεδιασμό και την διαχείριση της βάσης δεδομένων. Στέλνει ερωτήματα προς την βάση και παίρνει ως απάντηση τα δεδομένα που προέκυψαν από την εκτέλεση της ερώτησης. Κάποια από τα ερωτήματα είναι SELECT για την επιλογή στοιχείων από την βάση, INSERT για την εισαγωγή στοιχείων στην βάση, UPDATE για την αντικατάσταση ήδη υπάρχων στοιχείων, και DELETE για την διαγραφή στοιχείων από την βάση.

Οι λόγοι που χρησιμοποιούμε βάσεις δεδομένων σε σύγκριση με την αποθήκευση των πληροφοριών σε αρχεία απλού κειμένου είναι:

- Η ύπαρξη μια γενικά αποδεκτής δομής για την διατήρηση της ακεραιότητας των δεδομένων
- Η δυνατότητα δημιουργίας συσχετίσεων ανάμεσα στα δεδομένα που αποθηκεύονται για την βέλτιστη διαχείριση τους.
- Η δυνατότητα για την δομημένη αποθήκευση μεγάλου όγκου πληροφοριών.
- Η εύκολη πρόσβαση και γρήγορη επεξεργασία των δεδομένων μέσω των διαδικασιών που παρέχονται.

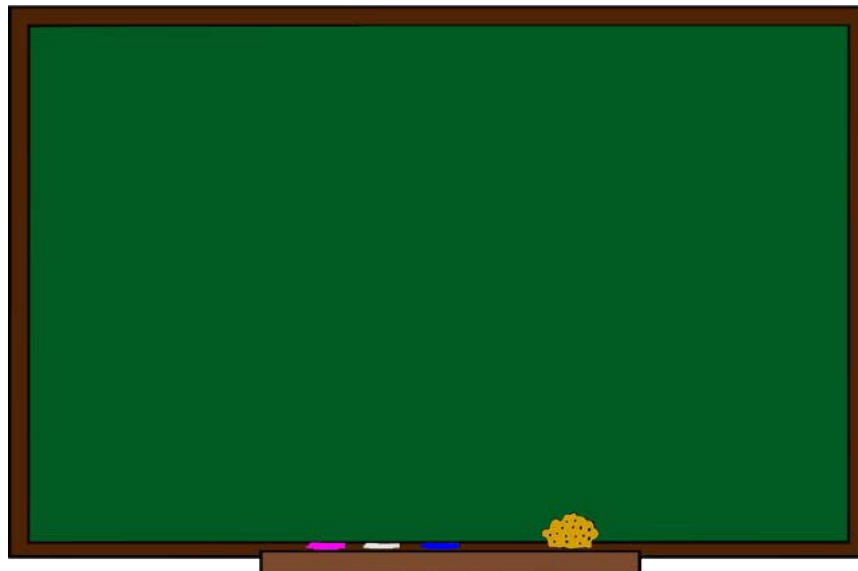
Στην εφαρμογή μας με την χρήση της MySQL έχουμε δημιουργήσει μια βάση δεδομένων στην οποία κατά την διάρκεια των δοκιμασιών αποθηκεύονται μόνιμα πληροφορίες όπως όνομα, φύλλο και ηλικία παιδιού όπως και ο αριθμός σωστών

απαντήσεων σε κάθε δοκιμασία. Ακόμα στην βάση αυτή υπάρχουν αποθηκευμένες οι απαραίτητες πληροφορίες για την κάθε δοκιμασία, η λίστα με τα σύνολα αριθμών που εμφανίζονται στα μήλα στην 1η δοκιμασία, η λίστα με τα σύνολά των αριθμών που ακούει το παιδί στην 2η δοκιμασία καθώς και οι λίστες με τα γράμματα και τις λέξεις για την 3η και 4η δοκιμασία αντίστοιχα.

### 3.3 Εμφάνιση

Έχοντας ως γνώμονα τις ηλικίες των χρηστών της εφαρμογής δόθηκε ιδιαίτερη σημασία στην εμφάνιση των δοκιμασιών. Επειδή η γνώση χρήσης του ηλεκτρονικού υπολογιστή σε προσχολικές ηλικίες δεν είναι βέβαιη έπρεπε η εφαρμογή να είναι απλή και να παρέχει μόνος τις απαραίτητες δυνατότητες στα παιδιά. Έτσι χρησιμοποιήθηκαν μεγάλα κουμπιά για την εκτέλεση των δοκιμασιών, λέξεις μεγάλης γραμματοσειράς για να είναι ευανάγνωστες, χρωματιστά περιεχόμενα σε έντονα χρώματα για να τραβάνε το ενδιαφέρον των παιδιών και αντικείμενα με μεγάλους αριθμούς και γράμματα. Έτσι όλα αυτά συμβάλουν στην ομαλότερη χρήση της εφαρμογής από τα παιδιά.

Ακόμα στις 7 δοκιμασίες χρησιμοποιήθηκε ως φόντο οθόνης ένας πράσινος πίνακας καθώς αποτελεί ένα καθημερινό σχολικό αντικείμενο για τα περισσότερα παιδιά.



Εικόνα 9. Φόντο δοκιμασιών

## 4. ΑΝΑΠΤΥΞΗ

### 4.1 Βάση Δεδομένων

Όπως αναφέρθηκε και νωρίτερα/πριν για της ανάγκες μόνιμης αποθήκευσης δεδομένων στην εφαρμογή δημιουργήθηκε μια βάση δεδομένων. Η βάση αυτή περιέχει 12 πίνακες/τραπέζια για την αποθήκευση τόσο των πληροφοριών που παράγει ο χρήστης όσο και των δεδομένων που χρειάζεται για να εκτελέσει την εφαρμογή.

Ποιό αναλυτικά η βάση χωρίζεται σε 2 ειδών πίνακες. Σε αυτούς που τα δεδομένων έχουν εισαχθεί κατά την δημιουργία της εφαρμογής από τον προγραμματιστή και δεν τροποποιούνται κατά την διάρκεια εκτέλεσης της εφαρμογής αλλά προσφέρουν στοιχεία όπως τις ερωτήσεις των δοκιμασιών και χρησιμοποιούνται ως πίνακες ανάγνωσης και σε αυτούς που δημιουργήθηκαν κενοί με σκοπό να διατηρούμε τις απαντήσεις που δίνει ο χρήστης, τον αριθμό των σωστών απαντήσεων, και τα χαρακτηριστικά στοιχεία του όπως το ψευδώνυμο, το φύλο και την ηλικία του και χρησιμοποιούνται ως πίνακες εγγραφής.

Υπάρχουν 4 πίνακες ανάγνωσης που καλύπτουν τις πληροφορίες που χρειάζονται οι 4 δοκιμασίες (3-6). Όλοι τους έχουν πανομοιότυπη μορφή κάτι που εξυπηρετεί στην καλύτερη διαχείριση της εφαρμογής.

Ο πίνακας Questions\_1 χρησιμοποιείται στην αποθήκευση των αριθμών που χρειάζονται για την εμφάνιση των μήλων στην δοκιμασία 3. Έχει 2 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών του και το Question που χρησιμοποιείται στην αποθήκευση των λιστών με τους αριθμούς. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT και το Question τύπου varchar(20). Κατά την δημιουργία του ο πίνακας συμπληρώθηκε με 12 εγγραφές.

| <b>ID</b> | <b>Question</b> |
|-----------|-----------------|
| 1         | 5,3             |
| 2         | 4,2             |
| 3         | 8,7,1           |
| 4         | 6,9,1           |
| 5         | 2,4,6,9         |
| 6         | 7,8,3,5         |
| 7         | 1,6,4,5,9       |
| 8         | 2,9,7,6,3       |
| 9         | 7,6,3,5,1,4     |
| 10        | 1,5,4,9,8,2     |
| 11        | 4,2,3,8,9,1,7   |
| 12        | 1,6,3,7,9,4,8   |

*Πίνακας 1. Questions\_1*

Ο πίνακας Questions\_2 χρησιμοποιείται στην αποθήκευση των αριθμών που χρειάζεται να ακουστούν στην δοκιμασία 4. Έχει 2 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών του και το Question που χρησιμοποιείται στην αποθήκευση των λιστών με τους αριθμούς. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT και το Question τύπου varchar(2). Κατά την δημιουργία του ο πίνακας συμπληρώθηκε με 16 εγγραφές.

| <b>ID</b> | <b>Question</b> |
|-----------|-----------------|
| 1         | 2,9             |
| 2         | 4,6             |
| 3         | 3,8,6           |
| 4         | 6,1,2           |
| 5         | 3,4,1,7         |
| 6         | 6,1,5,8         |
| 7         | 8,4,2,3,9       |
| 8         | 5,2,1,8,6       |

|    |                   |
|----|-------------------|
| 9  | 3,8,9,1,7,4       |
| 10 | 7,9,6,4,8,3       |
| 11 | 5,1,7,4,2,3,8     |
| 12 | 9,8,5,2,1,6,3     |
| 13 | 1,6,4,5,9,7,6,3   |
| 14 | 2,9,7,6,3,1,5,4   |
| 15 | 5,3,8,7,1,2,4,6,9 |
| 16 | 4,2,6,9,1,7,8,3,5 |

*Πίνακας 2. Questions\_2*

Ο πίνακας Questions\_3 χρησιμοποιείται στην αποθήκευση των γραμμάτων που χρειάζεται να εμφανιστούν και να ακουστούν στην δοκιμασία 5. Έχει 3 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών του, το Question που χρησιμοποιείται στην αποθήκευση των λιστών με τα γράμματα, και το Correct\_Answer που χρησιμοποιείται στην αποθήκευση του γράμματος που έχει οριστεί ως σωστή απάντηση. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το Question τύπου varchar(10) και το Correct\_Answer τύπου varchar(10) και αυτό. Κατά την δημιουργία του ο πίνακας συμπληρώθηκε με 10 εγγραφές.

| <b>ID</b> | <b>Question</b> | <b>Correct_Answer</b> |
|-----------|-----------------|-----------------------|
| 1         | Π,Δ,Α           | Δ                     |
| 2         | Ζ,Μ,Σ           | Σ                     |
| 3         | Ν,Ε,Β           | Β                     |
| 4         | Λ,Η,Γ           | Λ                     |
| 5         | Σ,Ε,Ρ           | Ρ                     |
| 6         | Κ,Ξ,Ν           | Ξ                     |
| 7         | Τ,Υ,Μ           | Τ                     |
| 8         | Χ,Φ,Γ           | Φ                     |
| 9         | Ο,Ε,Α           | Ο                     |
| 10        | Ξ,Τ,Ψ           | Ψ                     |

*Πίνακας 3. Questions\_3*

Ο πίνακας Questions\_4 χρησιμοποιείται στην αποθήκευση των λέξεων που χρειάζεται να εμφανιστούν και να ακουστούν στην δοκιμασία 6. Έχει 3 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών του, το Question που χρησιμοποιείται στην αποθήκευση των λιστών με τις λέξεις, και το Correct\_Answer που χρησιμοποιείται στην αποθήκευση της λέξεις που έχει οριστεί ως σωστή απάντηση. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το Question τύπου varchar(10) και το Correct\_Answer τύπου varchar(10) και αυτό. Κατά την δημιουργία του ο πίνακας συμπληρώθηκε με 10 εγγραφές.

| ID | Question    | Correct_Answer |
|----|-------------|----------------|
| 1  | γάτα,βίδα   | γάτα           |
| 2  | χέρι,παπί   | παπί           |
| 3  | μαμά,νερό   | μαμά           |
| 4  | σάκα,γάλα   | γάλα           |
| 5  | πίτα,ρόδα   | πίτα           |
| 6  | σπίτι,μήλο  | μήλο           |
| 7  | τυρί,μάτι   | μάτι           |
| 8  | αυγό,γόμα   | αυγό           |
| 9  | ραβδί,σώμα  | σώμα           |
| 10 | χαρτί,κουτί | χαρτί          |

*Πίνακας 4. Questions\_4*

Υπάρχουν 8 πίνακες εγγραφής που διατηρούν τις πληροφορίες που παράγει ο χρήστης κατά την χρήση της εφαρμογής. Ένας πίνακας είναι υπεύθυνος να κρατάει τα ψευδώνυμα των χρηστών, τις ηλικίες και το φύλο του καθενός και οι υπόλοιποι 7 κρατάνε τα αποτελέσματα των δοκιμασιών, 1 πίνακας για κάθε δοκιμασία. Και οι 7 πίνακες έχουν ίδια δομή κάτι που τους κάνει εύκολα διαχειρίσιμους και βοηθάει στην καλύτερη εξαγωγή των αποτελεσμάτων.

Ο πίνακας Kids χρησιμοποιείται στην αποθήκευση των παιδιών που εκτελούν τις δοκιμασίες. Έχει 4 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα

και χρησιμοποιείται στην αρίθμηση των εγγραφών του, το User που χρησιμοποιείται στην αποθήκευση του ψευδώνυμου που εισάγει το παιδί, το Gender που χρησιμοποιείται στην αποθήκευση του φύλλου που εισάγει το παιδί και το Age που χρησιμοποιείται στην αποθήκευση της ηλικίας που εισάγει το παιδί. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το User είναι τύπου varchar(20), το Gender τύπου varchar(5) και το Age τύπου int(2). Όταν δημιουργήθηκε ο πίνακας έμεινε κενός καθώς οι πληροφορίες που θα αποθηκευτούν σε αυτόν θα παραχθούν απ τον χρήστη.

Ο πίνακας Results1 χρησιμοποιείται στην αποθήκευση των αποτελεσμάτων στην δοκιμασία. Έχει 3 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών, το User που χρησιμοποιείται στην αποθήκευση του ψευδωνύμου του παιδιού για το οποίο γίνεται η εγγραφή και το Correct\_Answers που χρησιμοποιείται στην αποθήκευση του αριθμού των σωστών απαντήσεων που έδωσε το παιδί. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το User είναι τύπου varchar(20) και το Correct\_Answers είναι τύπου int(2). Όταν δημιουργήθηκε ο πίνακας έμεινε κενός καθώς οι πληροφορίες που θα αποθηκευτούν σε αυτόν θα παραχθούν απ τον χρήστη.

Ο πίνακας Results2 χρησιμοποιείται στην αποθήκευση των αποτελεσμάτων στην δοκιμασία. Έχει 4 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών, το User που χρησιμοποιείται στην αποθήκευση του ψευδωνύμου του παιδιού για το οποίο γίνεται η εγγραφή, το Correct\_Answers που χρησιμοποιείται στην αποθήκευση του αριθμού των σωστών απαντήσεων που έδωσε το παιδί και το Wrong\_Answers που χρησιμοποιείται στην αποθήκευση του αριθμού των λάθος απαντήσεων που έδωσε το παιδί. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το User είναι τύπου varchar(20), το Correct\_Answers είναι τύπου int(2) και το Wrong\_Answers είναι τύπου int(2). Όταν δημιουργήθηκε ο πίνακας έμεινε κενός καθώς οι πληροφορίες που θα αποθηκευτούν σε αυτόν θα παραχθούν απ τον χρήστη.

Ο πίνακας Results\_3 χρησιμοποιείται στην αποθήκευση των αποτελεσμάτων στην δοκιμασία. Έχει 3 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα



και χρησιμοποιείται στην αρίθμηση των εγγραφών, το User που χρησιμοποιείται στην αποθήκευση του ψευδωνύμου του παιδιού για το οποίο γίνεται η εγγραφή και το Correct\_Answers που χρησιμοποιείται στην αποθήκευση του αριθμού των σωστών απαντήσεων που έδωσε το παιδί. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το User είναι τύπου varchar(20) και το Correct\_Answers είναι τύπου int(2). Όταν δημιουργήθηκε ο πίνακας έμεινε κενός καθώς οι πληροφορίες που θα αποθηκευτούν σε αυτόν θα παραχθούν απ τον χρήστη.

Ο πίνακας Results\_4 χρησιμοποιείται στην αποθήκευση των αποτελεσμάτων στην δοκιμασία. Έχει 3 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών, το User που χρησιμοποιείται στην αποθήκευση του ψευδωνύμου του παιδιού για το οποίο γίνεται η εγγραφή και το Correct\_Answers που χρησιμοποιείται στην αποθήκευση του αριθμού των σωστών απαντήσεων που έδωσε το παιδί. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το User είναι τύπου varchar(20) και το Correct\_Answers είναι τύπου int(2). Όταν δημιουργήθηκε ο πίνακας έμεινε κενός καθώς οι πληροφορίες που θα αποθηκευτούν σε αυτόν θα παραχθούν απ τον χρήστη.

Ο πίνακας Results\_5 χρησιμοποιείται στην αποθήκευση των αποτελεσμάτων στην δοκιμασία. Έχει 4 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών, το User που χρησιμοποιείται στην αποθήκευση του ψευδωνύμου του παιδιού για το οποίο γίνεται η εγγραφή, το Correct\_Answers που χρησιμοποιείται στην αποθήκευση του αριθμού των σωστών απαντήσεων που έδωσε το παιδί και το Time που χρησιμοποιείται στην αποθήκευση του συνολικού χρόνου που χρειάστηκε το παιδί για την ολοκλήρωση της δοκιμασίας. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το User είναι τύπου varchar(20), το Correct\_Answers είναι τύπου int(2) και το Time είναι τύπου varchar(4). Όταν δημιουργήθηκε ο πίνακας έμεινε κενός καθώς οι πληροφορίες που θα αποθηκευτούν σε αυτόν θα παραχθούν απ τον χρήστη.

Ο πίνακας Results\_6 χρησιμοποιείται στην αποθήκευση των αποτελεσμάτων στην δοκιμασία. Έχει 4 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών, το User που χρησιμοποιείται στην

αποθήκευση του ψευδωνύμου του παιδιού για το οποίο γίνεται η εγγραφή, το Correct\_Answers που χρησιμοποιείται στην αποθήκευση του αριθμού των σωστών απαντήσεων που έδωσε το παιδί και το Time που χρησιμοποιείται στην αποθήκευση του συνολικού χρόνου που χρειάστηκε το παιδί για την ολοκλήρωση της δοκιμασίας. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το User είναι τύπου varchar(20), το Correct\_Answers είναι τύπου int(2) και το Time είναι τύπου varchar(4). Όταν δημιουργήθηκε ο πίνακας έμεινε κενός καθώς οι πληροφορίες που θα αποθηκευτούν σε αυτόν θα παραχθούν απ τον χρήστη.

Ο πίνακας Results\_7 χρησιμοποιείται στην αποθήκευση των αποτελεσμάτων στην δοκιμασία. Έχει 4 χαρακτηριστικά, το ID που είναι το πρωτεύον κλειδί του πίνακα και χρησιμοποιείται στην αρίθμηση των εγγραφών, το User που χρησιμοποιείται στην αποθήκευση του ψευδωνύμου του παιδιού για το οποίο γίνεται η εγγραφή, το Correct\_Answers που χρησιμοποιείται στην αποθήκευση του αριθμού των σωστών απαντήσεων που έδωσε το παιδί και το Time που χρησιμοποιείται στην αποθήκευση του συνολικού χρόνου που χρειάστηκε το παιδί για την ολοκλήρωση της δοκιμασίας. Το ID είναι τύπου int(2) με όρισμα AUTO\_INCREMENT, το User είναι τύπου varchar(20), το Correct\_Answers είναι τύπου int(2) και το Time είναι τύπου varchar(4). Όταν δημιουργήθηκε ο πίνακας έμεινε κενός καθώς οι πληροφορίες που θα αποθηκευτούν σε αυτόν θα παραχθούν απ τον χρήστη.

## **4.2 Διεπαφή**

Η εφαρμογή χωρίζεται σε δυο κύρια κομμάτια. Στο πρώτο, στο οποίο ανοίγουμε την εφαρμογή ως επιβλέπων και ετοιμάζουμε την διαδικασία των δοκιμασιών και το δεύτερο κομμάτι στο οποίο το παιδί συμπληρώνει τα στοιχεία του, κάτι στο οποίο μπορεί να παρέμβει και ο επιβλέπων για βοήθεια εάν το παιδί δεν είναι σε θέση να τα εισάγει μόνο του, και στην συνέχεια περνάει στην διαδικασία των δοκιμασιών.

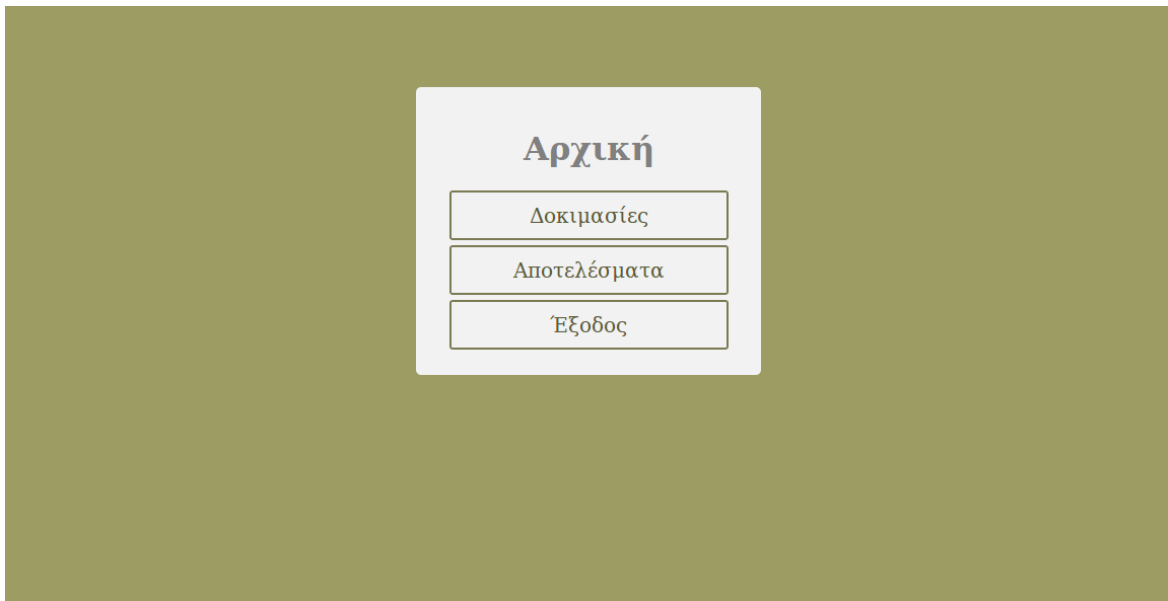
### **4.2.1 Γενικά**

Αρχικά απαιτείται ή είσοδος στην εφαρμογή με την χρήση ψευδώνυμου (username ) και κωδικού πρόσβασης (password).



Εικόνα 10. Οθόνη εισόδου

Μετά την επιτυχής επαλήθευση των στοιχείων αυτών εμφανίζεται ένα μενού με τις επιλογές “Δοκιμασίες”, “Αποτελέσματα” και “Έξοδος”. Με την επιλογή “Δοκιμασίες” μεταβαίνουμε στο δεύτερο κομμάτι της εφαρμογής που ξεκινάει η διαδικασία των δοκιμασιών από το παιδί. Η επιλογή “Αποτελέσματα” δημιουργεί και κάνει λήψη ενός αρχείου που περιέχει όλα τα αποτελέσματα των παιδιών που έχουν εκτέλεση τις δοκιμασίες. Και τέλος η επιλογής “Έξοδος” αποσυνδέει τον χρήστη και επιστρέφει την εφαρμογή στην αρχική κατάσταση εισόδου.



Εικόνα 11. Μενού εφαρμογής

Πριν την έναρξη των δοκιμασιών το παιδί εισάγει κάποιες πληροφορίες που θα βοηθήσουν τόσο στην σωστά δομημένη αποθήκευση των πληροφοριών όσο και στον προσδιορισμό σημαντικών προσωπικών στοιχείων. Αρχικά ζητείται από το παιδί να εισάγει το όνομα από τον αγαπημένο του ήρωα έτσι ώστε τα στοιχεία που αποθηκεύονται στην βάση δεδομένων να έχουν ως κοινό χαρακτηριστικό το όνομα αυτό για ευκολία στην ομαδοποίηση τους.



Εικόνα 12. Οθόνη ονόματος

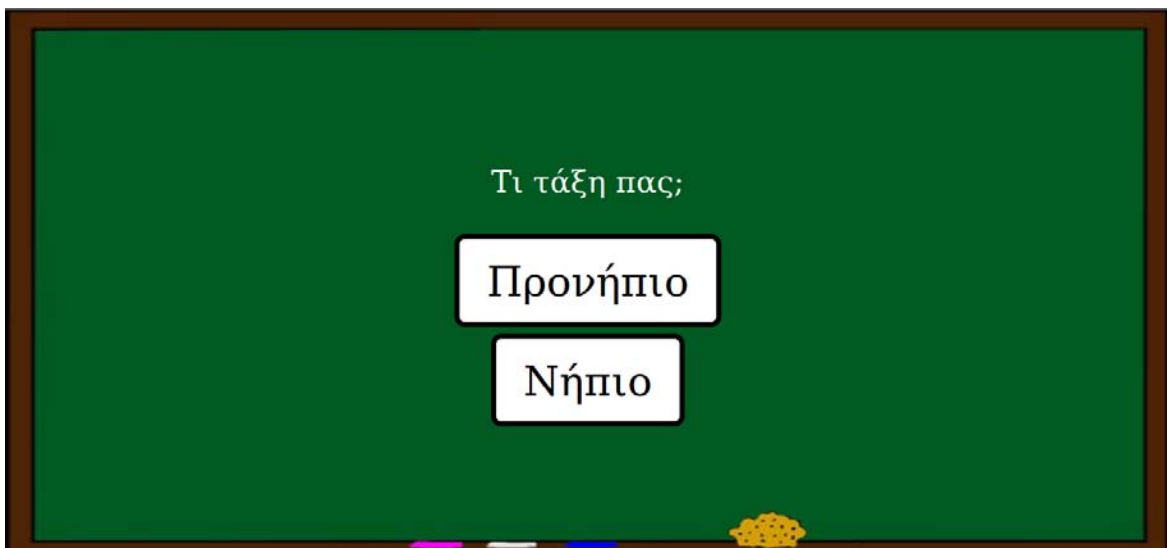
Στην συνέχεια εμφανίζεται στην οθόνη ένα σχήμα αγοριού με μπλε χρώμα και ένα

σχήμα κοριτσιού με ροζ χρώμα. Το παιδί καλείται να επιλέξει το φύλο του κάνοντας κλικ σε ένα από τα δύο.



Εικόνα 13. Οθόνη φύλου

Και τέλος την ηλικία του με την ερώτηση “Τι τάξη πηγαίνεις”.



Εικόνα 14. Οθόνη ηλικίας

Εάν το παιδί δεν γνωρίζει να πληκτρολογεί ή δεν είναι σε θέση να εισάγει τα στοιχεία αυτά παρέχεται βοήθεια από τον επιβλέπων.

Πριν την έναρξη κάθε δοκιμασίας εμφανίζονται στην οθόνη δύο κουμπιά, το ένα κάτω από το άλλο. Το πρώτο είναι πορτοκαλί και πάνω του αναγράφεται η λέξη “Παράδειγμα”

ενώ το δεύτερο είναι άσπρο και έχει πάνω του την λέξη “Πάμε!”.



Εικόνα 15. Αρχική οθόνη δοκιμασιών

Όταν το παιδί κάνει κλικ στο πορτοκαλί κουμπί εμφανίζεται στην οθόνη ένα παράθυρο μέσα στο οποίο γίνεται η αναπαραγωγή ενός βίντεο που εξηγεί στο παιδί τι πρέπει να κάνει στην δοκιμασία που ακολουθεί.



Εικόνα 16. Βίντεο οδηγιών

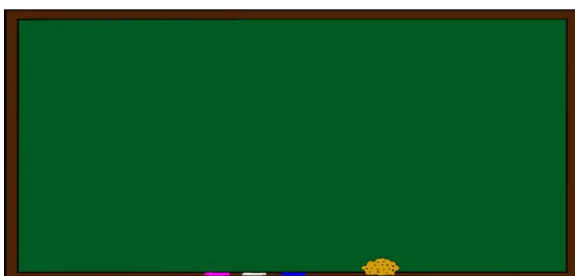
Όταν τελειώσει το βίντεο, το παράθυρο εξαφανίζεται μαζί με το πορτοκαλί κουμπί. Έτσι στην οθόνη μένει το άσπρο κουμπί το οποίο όταν το κλικάρει το παιδί ξεκινάει η δοκιμασία.



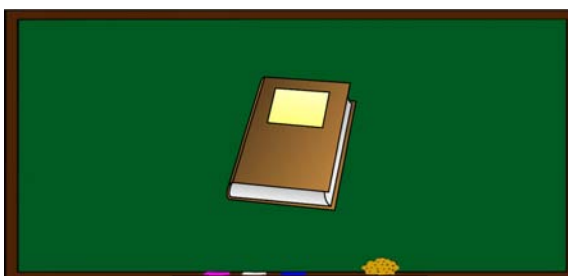
Εικόνα 17. Οθόνη έναρξης δοκιμασιών

#### 4.2.2 Δοκιμασία 1

Η δοκιμασία αυτή χωρίζεται σε δύο καταστάσεις. Στην πρώτη εμφανίζεται στο κέντρο της οθόνης μια εικόνα ενός σχολικού βιβλίου για δύο δευτερόλεπτα. Μετά το πέρας αυτού του χρόνου, η δοκιμασία περνάει στην δεύτερη κατάσταση, το βιβλίο εξαφανίζεται και παραμένει κρυφό για άλλα δύο δευτερόλεπτα. Έπειτα επανεμφανίζεται και η διαδικασία εναλλαγής των δύο καταστάσεων του βιβλίου επαναλαμβάνεται 6 φορές συνεχόμενα. Στόχος του παιδιού είναι να κάνει κλικ πάνω στο βιβλίο όσες φορές αυτό εμφανιστεί. Εάν το παιδί κάνει κλικ πάνω στο βιβλίο όταν αυτό είναι εμφανές πριν το πέρας του χρόνου των δύο δευτερολέπτων τότε το βιβλίο εξαφανίζεται και η δοκιμασία περνάει στην δεύτερη κατάσταση, συνεχίζοντας την ροή της όπως προαναφέρθηκε. Εάν το παιδί κάνει κλικ οπουδήποτε στην οθόνη όταν δεν είναι εμφανές το βιβλίο δεν συμβαίνει τίποτα. Στη συνέχεια η δοκιμασία ολοκληρώνεται και η εφαρμογή μεταβαίνει στην επόμενη σελίδα για την δοκιμασία 2.



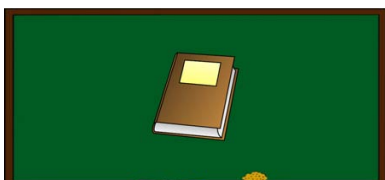
Εικόνα 18. Δοκιμασία 1-Πρώτη κατάσταση



Εικόνα 19. Δοκιμασία 1-Δεύτερη κατάσταση

### 4.2.3 Δοκιμασία 2

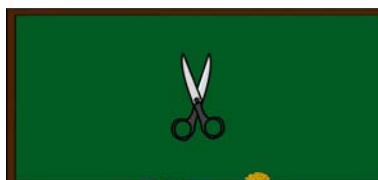
Η δοκιμασία 2 χωρίζεται σε δύο καταστάσεις. Στην πρώτη εμφανίζεται στο κέντρο της οθόνης μια εικόνα ενός αντικειμένου όπως βιβλίο ή ψαλίδι ή γόμα. Το αντικείμενο παραμένει εμφανές για δύο δευτερόλεπτα. Στην συνέχεια, η δοκιμασία περνάει στην δεύτερη κατάσταση, το αντικείμενο εξαφανίζεται παραμένοντας κρυφό για άλλα δύο δευτερόλεπτα. Έπειτα εμφανίζεται το επόμενο στην σειρά αντικείμενο και η ροή της δοκιμασίας συνεχίζεται κανονικά. Τα αντικείμενα εμφανίζονται ανακατεμένα αλλά σε συγκεκριμένη σειρά. Σκοπός της δοκιμασίας είναι το παιδί να κλικάρει μόνο στο βιβλίο όταν αυτό εμφανιστεί στην οθόνη όσες φορές εμφανιστεί. Εάν το παιδί κάνει κλικ στο αντικείμενο όταν αυτό είναι εμφανές στην οθόνη πριν το πέρας του χρόνου τότε το αντικείμενο εξαφανίζεται και η δοκιμασία περνάει στην δεύτερη κατάσταση της και συνεχίζεται κανονικά. Αντιθέτως αν το παιδί κλικάρει την οθόνη όταν η δοκιμασία βρίσκεται στην δεύτερη κατάσταση και δεν υπάρχει κάτι στην οθόνη δεν συμβαίνει κάτι. Στη συνέχεια η δοκιμασία 2 ολοκληρώνεται και η εφαρμογή μεταβαίνει στην επόμενη σελίδα για την δοκιμασία 3.



Εικόνα 20. Δοκιμασία 3-Βιβλίο



Εικόνα 21. Δοκιμασία 3-Μολύβι



Εικόνα 22. Δοκιμασία 3-ψαλίδι

### 4.2.4 Δοκιμασία 3

Η δοκιμασία 3 χωρίζεται σε δύο φάσεις. Αρχικά, στην πρώτη, εμφανίζεται στο



αριστερό μέρος της οθόνης ένα δέντρο μέσα στο οποίο εμφανίζονται διαδοχικά μήλα, το καθένα έχοντας έναν αριθμό πάνω του. Το κάθε μήλο παραμένει ορατό για 2 δευτερόλεπτα και κάθε φορά εμφανίζεται σε διαφορετική θέση πάνω στο δέντρο σε σχέση με το προηγούμενο. Οι αριθμοί που αναγράφονται στα μήλα έχουν συγκεκριμένη σειρά και λαμβάνονται από αντίστοιχη λίστα αποθηκευμένη στην βάση δεδομένων.



Εικόνα 23. Δοκιμασία 3-Φάση πρώτη, εμφάνιση του μήλου 5 από την λίστα (5,3)



Εικόνα 24. Δοκιμασία 3-Φάση πρώτη, εμφάνιση του μήλου 3 από την λίστα (5,3)

Αφού εμφανιστούν όλα τα μήλα, η δοκιμασία περνάει στην δεύτερη φάση της και εμφανίζεται στο δεξί μέρος της οθόνης ένα αριθμητικό πληκτρολόγιο με τους αριθμούς (1-9), μια άσπρη οθόνη και ένα κουμπί συνέχειας. Το παιδί πρέπει να κλικάρει τα νούμερα που εμφανίστηκαν στα μήλα στην πρώτη φάση με την σειρά που εμφανίστηκαν. Τα νούμερα που επιλέγει εμφανίζονται στην οθόνη που βρίσκεται πάνω από πληκτρολόγιο. Όταν σχηματίσει την ακολουθία αριθμών που προηγήθηκαν κάνει κλικ στο κουμπί συνέχειας για να συνεχίσει.



Εικόνα 25. Δοκιμασία 3-Οθόνη δεύτερης φάσης



Εικόνα 26. Δοκιμασία 3-Φάση δεύτερη, εισαγωγή αριθμών

Έτσι η δοκιμασία ξαναπερνάει στην πρώτη φάση της και επαναλαμβάνεται με τον ίδιο τρόπο αλλά με τα επόμενα νούμερα της λίστας που προαναφέρθηκε. Τα μήλα, δηλαδή οι αριθμοί, που καλείται το παιδί να θυμηθεί αυξάνονται. Στην αρχή εμφανίζονται δύο μήλα δύο συνεχόμενες φορές, μετά τρία μήλα δύο συνεχόμενες φορές μετά τέσσερα μήλα δύο

φορές με μέγιστο αριθμό μήλων τα εφτά. Η δοκιμασία τελειώνει είτε με την επιτυχής απομνημόνευση και των δεκατεσσάρων συνόλων με μήλα από το παιδί είτε απαντώντας λάθος δύο συνεχόμενες ή τρεις συνολικά φορές καθ' όλη την διάρκεια της δοκιμασίας. Στη συνέχεια η δοκιμασία 3 ολοκληρώνεται και η εφαρμογή μεταβαίνει στην επόμενη σελίδα για την δοκιμασία 4.

#### 4.2.5 Δοκιμασία 4

Η δοκιμασία 4 χωρίζεται σε δύο φάσεις. Στην πρώτη φάση, το παιδί ακούει ένα σύνολο κομματιών ήχου όπου καθένα περιέχει την ηχητική αναπαράσταση ενός αριθμού από το ένα μέχρι το εννέα. Τα σύνολα των αριθμών προέρχονται από μια λίστα αποθηκευμένη στην βάση δεδομένων.

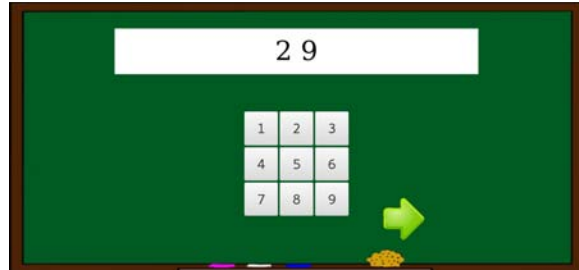


Εικόνα 27. Δοκιμασία 4-Οθόνη πρώτης φάσης

Αφού τους ακούσει, στην δεύτερη φάση, εμφανίζεται στην οθόνη ένα αριθμητικό πληκτρολόγιο με τους αριθμούς (1-9), μια άσπρη οθόνη και ένα κουμπί συνέχειας. Το παιδί πρέπει να κλικάρει στα νούμερα που άκουσε με την σειρά που τα άκουσε στην πρώτη φάση. Τα νούμερα στα οποία κάνει κλικ εμφανίζονται στην οθόνη που βρίσκεται πάνω από πληκτρολόγιο. Όταν σχηματίσει το σύνολο αριθμών που άκουσε κάνει κλικ στο κουμπί συνέχειας.



Εικόνα 28. Δοκιμασία 4-Οθόνη δεύτερης φάσης

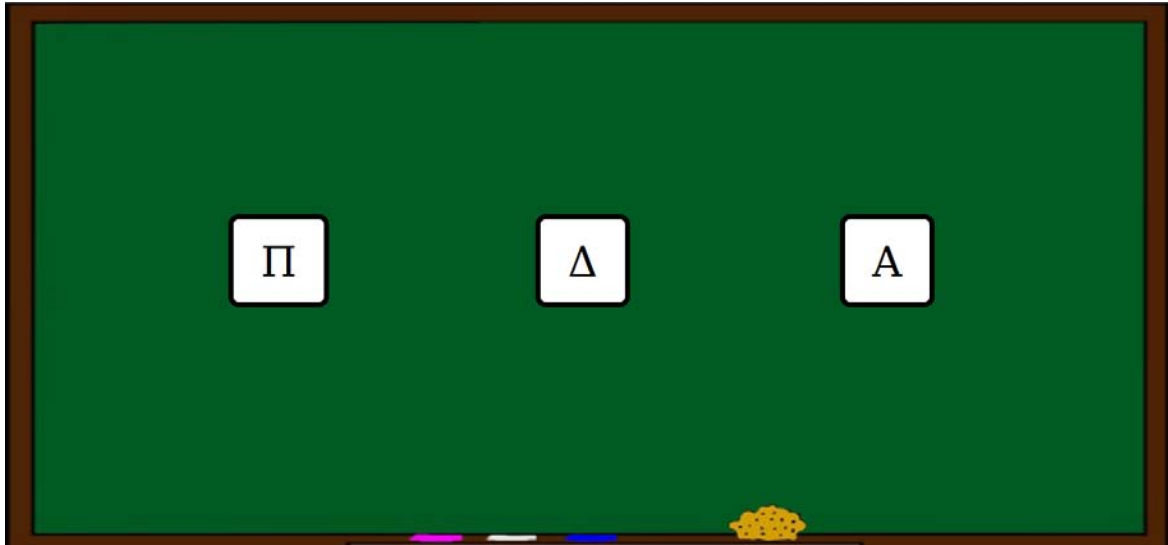


Εικόνα 29. Δοκιμασία 4-Φάση δεύτερη, εισαγωγή αριθμών

Η δοκιμασία επαναλαμβάνεται, περνώντας στην πρώτη φάση, με τον ίδιο τρόπο αλλά με το επόμενο κάθε φορά σύνολο αριθμών από την λίστα που αναφέρθηκε πριν. Το πλήθος των αριθμών σε κάθε σύνολο αυξάνετε κάθε 2η επανάληψη. Στην αρχή ακούγονται δύο αριθμοί δύο συνεχόμενες φορές, μετά τρεις αριθμοί δύο συνεχόμενες φορές, μετά τέσσερις αριθμοί δύο φορές με μέγιστο πλήθος αριθμών τους εννέα. Η δοκιμασία ολοκληρώνεται είτε με την επιτυχής απομνημόνευση και των δεκαέξι αριθμητικών συνόλων από το παιδί είτε απαντώντας λάθος δύο συνεχόμενες ή τρεις συνολικά φορές καθ' όλη την διάρκεια της δοκιμασίας. Στη συνέχεια η δοκιμασία 4 τερματίζεται και η εφαρμογή μεταβαίνει στην επόμενη σελίδα για την δοκιμασία 5.

#### 4.2.6 Δοκιμασία 5

Η δοκιμασία 5 χωρίζεται σε δύο φάσεις. Αρχικά, στην πρώτη, το παιδί ακούει ένα κομμάτι ήχου που περιέχει την ηχητική αναπαράσταση ενός γράμματος του Ελληνικού αλφάβητου. Στην συνέχεια, στην δεύτερη φάση, εμφανίζονται στην οθόνη τρία τετράγωνα κουμπιά που μέσα τους περιέχουν το καθένα από ένα γράμμα. Το ηχητικό κομμάτι καθώς και τα γράμματα αυτά βρίσκονται αποθηκευμένα σε μια λίστα στην βάση δεδομένων. Το παιδί καλείται να κάνει κλικ σε αυτό το γράμμα που πιστεύει ότι είναι το γράμμα που άκουσε.



Εικόνα 30. Δοκιμασία 5-Οθόνη δεύτερης φάσης

Όταν επιλέξει την απάντηση του τα κουμπιά εξαφανίζονται από την οθόνη, η δοκιμασία επιστρέφει στην πρώτη φάση και η διαδικασία επαναλαμβάνεται απ την αρχή με τα επόμενα ηχητικά και τις επόμενες επιλογές που περιέχει η λίστα της βάσης δεδομένων. Η διαδικασία αυτή εκτελείται συνολικά δέκα φορές και η δοκιμασία ολοκληρώνεται ανεξαρτήτως σωστών και λάθος απαντήσεων από το παιδί. Στη συνέχεια η δοκιμασία 5 τερματίζεται και η εφαρμογή μεταβαίνει στην επόμενη σελίδα για την δοκιμασία 6.

#### 4.2.7 Δοκιμασία 6

Η δοκιμασία 6 χωρίζεται σε δύο φάσεις. Αρχικά στην πρώτη το παιδί ακούει ένα κομμάτι ήχου που περιέχει την ηχητική αναπαράσταση μιας δυσύλλαβης λέξης. Στην συνέχεια, στην δεύτερη φάση, εμφανίζονται στην οθόνη δύο λέξεις σε μορφή κουμπιών. Το ηχητικό κομμάτι καθώς και οι λέξεις αυτές βρίσκονται αποθηκευμένα σε μια λίστα στην βάση δεδομένων. Το παιδί καλείται να κάνει κλικ σε αυτή που πιστεύει ότι είναι η λέξη που άκουσε στην πρώτη φάση.

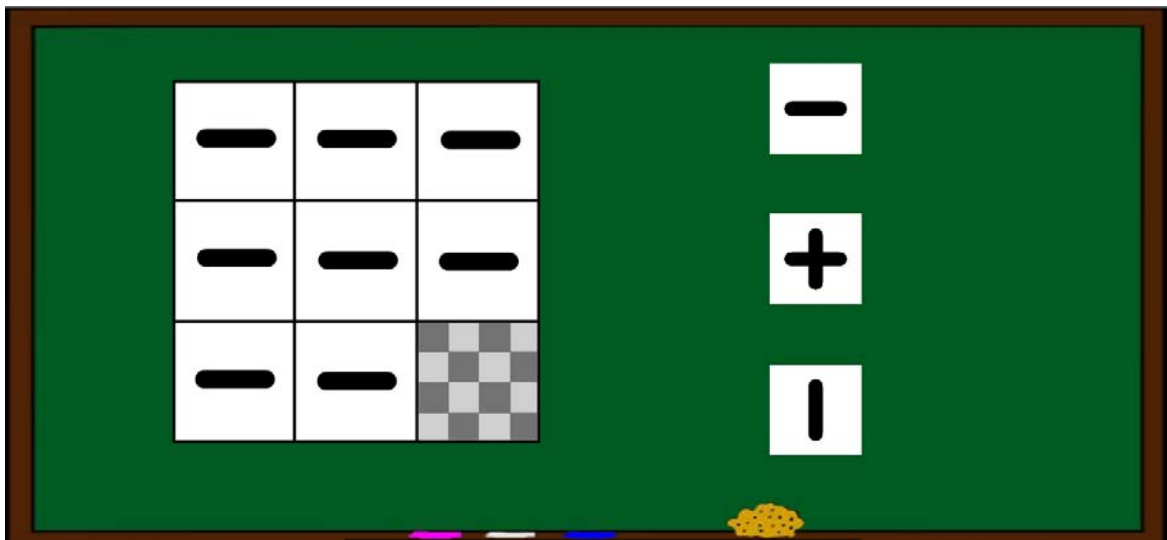


Εικόνα 31. Δοκιμασία 6-Οθόνη δεύτερης φάσης

Όταν επιλέξει την απάντηση του οι λέξεις εξαφανίζονται από την οθόνη, η δοκιμασία επιστρέφει στην πρώτη φάση και η διαδικασία επαναλαμβάνεται από την αρχή με τις επόμενες λέξεις και τα επόμενα ηχητικά που περιέχει η λίστα της βάσης δεδομένων. Η διαδικασία αυτή εκτελείται συνολικά δέκα φορές και η δοκιμασία ολοκληρώνεται ανεξαρτήτως σωστών και λάθος απαντήσεων από το παιδί. Στη συνέχεια η δοκιμασία 6 ολοκληρώνεται και η εφαρμογή μεταβαίνει στην επόμενη σελίδα για την δοκιμασία 7.

#### 4.2.8 Δοκιμασία 7

Στην δοκιμασία 7 εμφανίζεται στο αριστερό μέρος της οθόνης μια κυρίως εικόνα. Σε αυτή απεικονίζεται ένα σχήμα στο οποίο λείπει ένα κομμάτι στην κάτω δεξιά γωνία του για να ολοκληρωθεί. Στο δεξί μέρος της οθόνης εμφανίζονται τρεις μικρές εικόνες, η μια κάτω από την άλλη, που απεικονίζουν διαφορετικά κομμάτια.



Εικόνα 32. Οθόνη δοκιμασίας 7

Το παιδί καλείται να διακρίνει πιο κομμάτι από τα τρία πιθανά συμπληρώνει σωστά την κύρια εικόνα και να κλικάρει πάνω του. Στην συνέχεια εμφανίζεται μια καινούργια κυρίως εικόνα με τρεις καινούργιες επιλογές και η διαδικασία επαναλαμβάνεται. Η δοκιμασία ολοκληρώνεται μετά από δέκα διαφορετικές εικόνες ασχέτως σωστών και λάθος απαντήσεων από το παιδί. Στη συνέχεια η δοκιμασία 5 ολοκληρώνεται και η εφαρμογή μεταβαίνει στην επόμενη σελίδα για την δοκιμασία 6.

### 4.3 Κώδικας

Κάθε δοκιμασία έχει δημιουργηθεί ως SPA (single page application) ξεχωριστά. Αυτό σημαίνει ότι η κάθε δοκιμασία αλληλεπιδρά με τον χρήστη τροποποιώντας δυναμικά το περιχόμενο της χωρίς να χρειάζεται η συνεχής επαναφόρτωση της. Έτσι η εφαρμογή προσφέρει στον χρήστη μια συνεχής ροή χωρίς διακοπές,

#### 4.3.1 Δοκιμασία 1

Το βιβλίο της δοκιμασίας είναι ένα στοιχείο τύπου `img` το οποίο έχει ένα `onclick` event που καλεί την συνάρτηση `select_1`

`select_1()`:

```
1. function select_1(){  
2.   correct++;
```

```

3.   bk.style.display="none";
4.   i++;
5. }
6.

```

η οποία κάθε φορά που το παιδί κάνει κλικ στο βιβλίο αυξάνει την μεταβλητή correct κατά ένα εξαφανίζει το βιβλίο και αυξάνει τον μετρητή κατά ένα. Για να έχει όμως το βιβλίο την διαδραστικότητα της απόκρυψης και εμφάνισης του, χρησιμοποιείται η συνάρτηση game\_1

**game\_1():**

```

1. function game_1(){
2.   setTimeout(function(){
3.     if(i<6*2){
4.       if(bk.style.display=="inline"){
5.         bk.style.display="none";
6.       }
7.       else{
8.         bk.style.display="inline";
9.       }
10.      i++;
11.      game_1();
12.    }
13.    else{
14.      exit(1);
15.    }
16.  }, 2000);
17. }

```

η οποία ελέγχει εάν το βιβλίο είναι φανερό στην οθόνη, αν είναι τότε το αποκρύπτει αλλιώς το κάνει εμφανές. Αυξάνεται ο μετρητής κατά ένα και η συνάρτηση περιμένει 2000 χιλιοστά του δευτερολέπτου, δηλαδή 2 δευτερόλεπτα, και επαναλαμβάνει τον έλεγχο αναδρομικά. Αυτή η διαδικασία επαναλαμβάνεται συνολικά 12 φορές, 6 για να εμφανίσει το βιβλίο και 6 για να το αποκρύψει. Αφού ολοκληρωθούν όλες οι επαναλήψεις η διαδικασία ολοκληρώνεται καλώντας αρχικά την συνάρτηση exit\_1 με παράμετρο 1.

**exit\_1():**

```

1. function exit_1(input){
2.   switch(input){

```

```

3.   case 1:
4.   document.getElementById("correctAnswers")
      .value=correct;
5.   break;
6.   case 2:
7.   document.getElementById("correctAnswers")
      .value=correct;
8.   document.getElementById("wrongAnswers")
      .value=wrong;
9.   }
10. document.getElementById("myForm").submit();
11. }

```

Η συνάρτηση θέτει σε ένα κρυμμένο στοιχείο τύπου input με Id correctAnswers την τιμή της μεταβλητής correct η οποία κρατάει τις σωστές απαντήσεις που έχει το παιδί. Τέλος υποβάλλει την φόρμα με Id myForm η οποία περιέχει το στοιχείο correctAnswers που αναφέραμε. Αφού υποβληθεί η φόρμα, ο server εκτελεί τον κώδικα write\_1

**write\_1():**

```

1. $correct=$_POST["correctAnswers"];
2. include("db.php");
3. $sql="INSERT INTO Results_1 (User,Correct_Answers)
  VALUES ('$_SESSION["user"]."', $correct)";
4. mysqli_query($db_server, $sql);
5. mysqli_close($db_server);
6. header("Location: game02.php");

```

ο οποίος ανοίγει μια σύνδεση με την βάση δεδομένων καλώντας το αρχείο db.php, δημιουργεί ένα ερώτημα προς τη βάση στην μεταβλητή sql, το εκτελεί με την συνάρτηση mysqli\_query, κλείνει την σύνδεση με την βάση με την συνάρτηση mysqli\_close και τέλος μεταβαίνει στην επόμενη σελίδα με όνομα game2.php για την επόμενη δοκιμασία.

#### 4.3.2 Δοκιμασία 2

Τα τρία αντικείμενα της δοκιμασίας, το βιβλίο, το ψαλίδι και η γόμα είναι όλα αντικείμενα τύπου img με id book, pencil και scissors αντίστοιχα. Και τα τρία έχουν ένα onclick event που καλεί την συνάρτηση select\_2 με παραμέτρους 1,0,0 αντίστοιχα.



**select\_2():**

```
1. function select_2(input){
2.   if(input){
3.     correct++;
4.   }
5.   elements[i].style.display="none";
6.   i++;
7. }
```

Η συνάρτηση ελέγχει εάν το αντικείμενο στο οποίο έγινε κλικ έχει ως παράμετρο την τιμή 1, δηλαδή αν είναι βιβλίο, και αυξάνει την τιμή της μεταβλητή correct κατά ένα. Στην συνέχεια κρύβεται το αντικείμενο, το οποίο υποδηλώνεται από έναν πίνακα elements ο οποίος περιέχει τα αντικείμενα με την σειρά που εμφανίζονται, και αυξάνεται η τιμή του μετρητή κατά ένα. Για να έχουν τα αντικείμενα της δοκιμασίας διαδραστικότητα, να εμφανίζονται και να εξαφανίζονται χρησιμοποιείται η συνάρτηση game\_2.

**game\_2():**

```
1. function game_2(){
2.   setTimeout(function(){
3.     if(i<8){
4.       if(elements[i].style.display=="inline"){
5.         elements[i].style.display="none";
6.         i++;
7.       }
8.       else{
9.         elements[i].style.display="inline";
10.      }
11.      game_2();
12.    }
13.    else{
14.      exit(2);
15.    }
16.  }, 2000);
17. }
```

Αυτή ελέγχει αν ένα αντικείμενο από την λίστα μετά αντικείμενα είναι φανερό στην οθόνη. Αν είναι τότε το αποκρύπτει αλλιώς, αν είναι ήδη κρυμμένο, το κάνει εμφανές. Αυξάνεται η τιμή της μεταβλητής του μετρητή i κατά ένα, η συνάρτηση περιμένει 2000

χιλιοστά του δευτερολέπτου, δηλαδή 2 δευτερόλεπτα, και επαναλαμβάνει τον έλεγχο αναδρομικά. Η διαδικασία αυτή επαναλαμβάνεται οχτώ φορές, όσα και τα αντικείμενα στον πίνακα elements. Όταν η τιμή του μετρητή ξεπεράσει το οχτώ, η διαδικασία τερματίζεται καλώντας αρχικά την συνάρτηση exit\_1 με παράμετρο 0. Η συνάρτηση αυτή είναι ίδια με την αντίστοιχη συνάρτηση της δοκιμασίας 1. Η συνάρτηση θέτει σε ένα κρυμμένο στοιχείο τύπου input με Id correctAnswers την τιμή της μεταβλητής correct η οποία κρατάει τις σωστές απαντήσεις που έχει το παιδί. Τέλος υποβάλει την φόρμα με Id myForm η οποία περιέχει το στοιχείο correctAnswers που αναφέραμε. Αφού υποβληθεί η φόρμα, ο server εκτελεί τον κώδικα write\_2.

**write\_2():**

```
1. $correct=$_POST["correctAnswers"];
2. $wrong=$_POST["wrongAnswers"];
3. include("db.php");
4. $sql="INSERT INTOResults_01
   (User,Correct_Answers,Wrong_Answers)
   VALUES ('".$_SESSION["user"]."', $correct, $wrong)";
5. mysqli_query($db_server, $sql);
6. mysqli_close($db_server);
7. header("Location: game1.php");
```

Ο κώδικας ανοίγει μια σύνδεση με την βάση δεδομένων καλώντας το αρχείο db.php, δημιουργεί ένα ερώτημα προς τη βάση στην μεταβλητή sql, το εκτελεί με την συνάρτηση mysqli\_query, κλείνει την σύνδεση με την βάση με την συνάρτηση mysqli\_close και τέλος μεταβαίνει στην επόμενη σελίδα με όνομα game3.php για την επόμενη δοκιμασία.

### 4.3.3 Δοκιμασία 3

Πριν την έναρξη της δοκιμασίας η PHP εκτελεί τον κώδικα read\_1 και διαβάζει από την βάση δεδομένων MySQL τον πίνακα Questions\_1 που περιέχει τις λίστες με τους αριθμούς των μήλων και τους αποθηκεύει σε έναν πίνακα questions.

**read\_1():**

```
1. include("db.php");
2. $query="SELECT Question FROM Questions_1";
3. $result=mysqli_query($db_server, $query);
```

```

4. $questions=array();
5. if(mysqli_num_rows($result)>0){
6.     while($row=mysqli_fetch_assoc($result)){
7.         array_push($questions,$row["Question"]);
8.     }}
9.     mysqli_close($db_server);

```

Ο πίνακας αυτός μεταφέρεται από την PHP στην JavaScript με το κομμάτι κώδικα της HTML move\_1.

**move\_1():**

```

1. <script>
2. var questions=<?php echo json_encode($questions) ?>;
3. </script>

```

Τα ποία σημαντικά στοιχεία της δοκιμασίας είναι το δέντρο με τα μήλα, οι αριθμοί με την οθόνη και το βέλος συνέχειας. Το δέντρο με τα μήλα είναι στοιχεία τύπου img τα οποία ορίζονται στην HTML και εμφανίζονται με την JavaScript. Τα μήλα εμφανίζονται κάθε φορά με την κλήση της συνάρτησης createApple.

**createApple():**

```

1. function createApple(){
2.     if(i<questions[counter].length){
3.         do{
4.             x=randomNumber(0,6);
5.         }while(x==x_old);
6.         x_old=x;
7.         tmp=document.getElementById("apple_"+
            questions[counter][i]);
8.         tmp.style.left=pos[x][0];
9.         tmp.style.top=pos[x][1];
10.        tmp.style.display="inline";
11.        setTimeout(function(){
12.            tmp.style.display="none";
13.            i++;
14.            createApple();
15.        },2000);
16.        }
17.        else{
18.            cnt.style.display="inline";
19.        }}

```

Η συνάρτηση επαναλαμβάνει τόσες φορές όσοι είναι και οι αριθμοί της λίστας στον πίνακα με τις ερωτήσεις τα εξής βήματα. Αρχικά επιλέγεται ένας τυχαίος αριθμός από το 0 μέχρι το 6, με την προϋπόθεση ότι αυτός ο αριθμός δεν είναι ίδιος με τον προηγούμενο. Επιλέγετε ως στοιχείο tmp ένα από τα μήλα που έχει τον αριθμό που δείχνει η λίστα από τον πίνακα ερωτήσεων. Στο στοιχείο αυτό δίνετε μια τυχαία επιλογή θέσης από έναν πίνακα pos με συγκεκριμένες θέσεις, διαφορετική από του προηγούμενου αν υπάρχει. Το στοιχείο αυτό μένει εμφανές για 2000 χιλιοστά του δευτερολέπτου, δηλαδή δύο δευτερόλεπτα, έπειτα εξαφανίζεται ο μετρητής αυξάνεται κατά ένα και η συνάρτηση καλείται αναδρομικά. Όταν ολοκληρωθεί η εμφάνιση και του τελευταίου μήλου της λίστας γίνετε εμφανές το στοιχείο cnt που περιέχει την οθόνη τα κουμπιά με τους αριθμούς και το βέλος συνέχειας. Όταν γίνει κλικ σε έναν από τα κουμπιά εκτελείται η συνάρτηση select\_3.

**select\_3():**

```
1. function select_3(button){
2.   if(button=="enter"){
3.     answers[counter]=screenArray.join("");
4.     if(screenArray.toString()==questions[counter]){
5.       // correct answer
6.       correct++;
7.       tmp_1=0;
8.     }
9.     else{
10.      // not correct answer
11.      tmp_1++;
12.      tmp_2++;
13.    }
14.    if(tmp_1==2 || tmp_2==3){
15.      exit();
16.      return;
17.    }
18.    counter++;
19.    if(counter>=questions.length){
20.      exit();
21.      return;
22.    }
23.    else{
24.      game();
25.    }}
26.    else{
27.      screenArray.push(button);
```

```
28.     scr.innerHTML=screenArray.join(" ");
29.     }}
```

η οποία ελέγχει τι κουμπί πατήθηκε. Αν ήταν αριθμός τον αποθηκεύει σε έναν προσωρινό πίνακα `screenArray` και εμφανίζει τον πίνακα αυτόν στην οθόνη. Αν ήταν το βέλος συνέχειας τότε διαγράφεται το περιεχόμενο της οθόνης, ελέγχεται αν η απάντηση είναι σωστή και επαληθεύει τους περιορισμούς λανθασμένων απαντήσεων της δοκιμασίας ή αν δεν ήταν η τελευταία επανάληψη και συνεχίζεται η δοκιμασία με την συνάρτηση `game`. Αν κάτι από αυτά ισχύει η δοκιμασία τελειώνει με την κλήση της συνάρτησης `exit_2`

**exit\_2():**

```
1. function exit_2(){
2.     document.getElementById("main_container")
       .style.display="none";
3.     document.getElementById("correctAnswers")
       .value=correct;
4.     document.getElementById("myForm").submit();
5. }
```

η οποία καθαρίζει την οθόνη και περνάει τις σωστές απαντήσεις στην PHP μέσω της φόρμας της HTML. Η PHP με την σειρά της εκτελεί τον κώδικα `write_3`

**write\_3():**

```
1. $correct=$_POST["correctAnswers"];
2. include("db.php");
3. $sql="INSERT INTO Results_3 (User,Correct_Answers)
       VALUES ('$_SESSION["user"]."', $correct)";
4. mysqli_query($db_server, $sql);
5. mysqli_close($db_server);
6. header("Location: game4.php");
```

όπου ανοίγει μια σύνδεση με την βάση δεδομένων και αποθηκεύει τα αποτελέσματα στον αντίστοιχο πίνακα. Στην συνέχεια η εφαρμογή περνάει στην επόμενη δοκιμασία

#### 4.3.4 Δοκιμασία 4

Πριν την έναρξη της δοκιμασίας η PHP εκτελεί τον κώδικα `read_2` και διαβάζει από την βάση δεδομένων MySQL τον πίνακα `Questions_2` που περιέχει τις λίστες με τους

αριθμούς των ηχητικών κομματιών και τους αποθηκεύει σε έναν πίνακα questions

**read\_2():**

```
1. include("db.php");
2. $query="SELECT Question FROM Questions_2";
3. $result=mysqli_query($db_server,$query);
4. $questions=array();
5. if(mysqli_num_rows($result)>0){
6.     while($row=mysqli_fetch_assoc($result)){
7.         array_push($questions,$row["Question"]);
8.     }
9. mysqli_close($db_server);
```

ο οποίος μεταφέρεται από την PHP στην JavaScript με HTML κώδικα σαν τον κώδικα move\_1. Τα πιο σημαντικά στοιχεία της δοκιμασίας αυτής είναι οι αριθμοί που πρέπει να ακούσει το παιδί και τα στοιχεία που χρησιμοποιούνται για την εισαγωγή των απαντήσεων όπως είναι η οθόνη ο πίνακας με τα κουμπιά των αριθμών και το κουμπί συνέχειας. Στην αρχή της δοκιμασίας το παιδί ακούει κάποιους αριθμούς μέσω της συνάρτησης audio

```
1. function audio(){
2.     aud.src="audio/game2/"+tmp[j]+".mp3";
3.     aud.load();
4.     aud.play();
5.     aud.onended=function(){
6.         j++;
7.         if(j>=tmp.length){
8.             screenArray=[];
9.             scr.innerHTML="";
10.            m_cnt.style.display="block";
11.        }
12.        else{
13.            audio();
14.        }}}
```

η οποία ορίζει στο κομμάτι ήχου της HTML την διεύθυνση στον server του αριθμού της λίστας που χρειάζεται να αναπαραχθεί, φορτώνει το κομμάτι ήχου, το αναπαράγει και αν υπάρχουν άλλοι αριθμοί για αναπαραγωγή στην λίστα ξανά καλεί τον εαυτό της αναδρομικά. Στην αντίθετη περίπτωση προετοιμάζει τις μεταβλητές για την δεύτερη φάση της δοκιμασίας, εμφανίζοντας το κομμάτι με τα στοιχεία που χρειάζονται. Όταν γίνει κλικ

σε έναν από τα κουμπιά εκτελείται η συνάρτηση `select_3`, η οποία ελέγχει τι κουμπί πατήθηκε. Αν ήταν αριθμός τον αποθηκεύει σε έναν προσωρινό πίνακα `screenArray` και εμφανίζει τον πίνακα αυτόν στην οθόνη. Αν ήταν το βέλος συνέχειας τότε διαγράφεται το περιεχόμενο της οθόνης, ελέγχεται αν η απάντηση είναι σωστή και επαληθεύει τους περιορισμούς λανθασμένων απαντήσεων της δοκιμασίας ή αν δεν ήταν η τελευταία επανάληψη και συνεχίζεται η δοκιμασία με την συνάρτηση `playAudio()`. Αν κάτι από αυτά ισχύει η δοκιμασία τελειώνει με την κλήση της συνάρτησης `exit()` η οποία καθαρίζει την οθόνη και περνάει τις σωστές απαντήσεις στην PHP μέσω της φόρμας της HTML. Η PHP με την σειρά της εκτελεί τον κώδικα `write_4`

**write\_4():**

```
1. $correct=$_POST["correctAnswers"];
2. include("db.php");
3. $sql="INSERT INTO Results_2 (User,Correct_Answers)
   VALUES ('".$_SESSION["user"]."', $correct)";
4. mysqli_query($db_server,$sql);
5. mysqli_close($db_server);
6. header("Location: game5.php");
```

όπου ανοίγει μια σύνδεση με την βάση δεδομένων και αποθηκεύει τα αποτελέσματα στον αντίστοιχο πίνακα. Στην συνέχεια η εφαρμογή περνάει στην επόμενη δοκιμασία.

Για τις ανάγκες αποθήκευσης του χρόνου ολοκλήρωσης των επόμενων δοκιμασιών έχει δημιουργηθεί ένας χρονομετρητής. Ο χρονομετρητής αυτός εκτελεί τις εξής λειτουργίες.

- Εκκίνηση της χρονομέτρησης με την συνάρτηση `startTimer()`

**startTimer():**

```
1. function startTimer(){
2.   myTimer=setInterval(function(){time++;},1000);
3. }
```

- Παύση της χρονομέτρησης με την συνάρτηση `pauseTimer()`

**pauseTimer():**

```
1. function pauseTimer(){
2.   clearInterval(myTimer);
3. }
```

- Και διακοπή της χρονομέτρησης με την συνάρτηση stopTimer η οποία αποθηκεύει στην μεταβλητή time τον συνολικό χρόνο σε μορφή κειμένου με την μορφή Λεπτά: Δευτερόλεπτα.

**stopTimer():**

```
1. function stopTimer() {
2.   clearInterval(myTimer);
3.   var m=Math.floor(time/60);
4.   var s=time%60;
5.   time=m+": "+s;
6. }
```

### 4.3.5 Δοκιμασία 5

Πριν την έναρξη της δοκιμασίας η PHP εκτελεί τον κώδικα read\_5 όπου διαβάζει από την βάση δεδομένων MySQL τον πίνακα Questions\_3 που περιέχει τις λίστες με τα γράμματα των ηχητικών κομματιών μαζί με τις σωστές απαντήσεις της κάθε ερώτησης και τα αποθηκεύει σε πίνακες questions και answers αντίστοιχα.

**read\_3():**

```
1. include("db.php");
2. $query="SELECT Question,Correct_Answer
   FROM Questions_3";
3. $result=mysqli_query($db_server,$query);
4. $questions=array();
5. $answers=array()
6. if(mysqli_num_rows($result)>0){
7.   while($row=mysqli_fetch_assoc($result)){
8.     array_push($questions,$row["Question"]);
9.     array_push($answers,$row["Correct_Answer"]);
10.  }
11.  mysqli_close($db_server);
```

Οι πίνακες αυτοί μεταφέρονται από την PHP στην JavaScript με το κομμάτι κώδικα της HTML move\_2.

**move\_2:**

```
1. <script>
2. var questions=<?php echo json_encode($questions) ?>;
3. var corrAnsw=<?php echo json_encode($answers) ?>;
```



```
4. </script>
```

Τα κύρια στοιχεία της δοκιμασίας αυτής είναι τα γράμματα που πρέπει να ακούσει το παιδί και τα 3 κουμπιά που εμφανίζονται ως επιλογές. Στην αρχή της δοκιμασίας το παιδί ακούει ένα γράμμα με την συνάρτηση `game_1`

**game\_1():**

```
1. function game_1(){
2.   m_cnt.style.display="none";
3.   aud.src="audio/game3/"+corrAnsw[counter-1]+".mp3";
4.   aud.load();
5.   aud.play();
6.   aud.onended=function(){
7.     tmp=questions[counter-1].split(",");
8.     btn_1.innerHTML=tmp[0];
9.     btn_2.innerHTML=tmp[1];
10.    btn_3.innerHTML=tmp[2];
11.    m_cnt.style.display="grid";
12.    startTimer();
13.  }}
```

η οποία καθαρίζει την οθόνη, ορίζει στο κομμάτι ήχου της HTML την κατάλληλη διεύθυνση, το φορτώνει, το αναπαράγει και εμφανίζει στην οθόνη τα κουμπιά με τις πιθανές απαντήσεις της αντίστοιχης ερώτησης. Πριν την έξοδο από την συνάρτηση αυτή, ενεργοποιείται και ο χρονομετρητής. Όταν το παιδί επιλέξει την απάντηση του κάνοντας κλικ σε ένα από τα 3 κουμπιά καλείται η συνάρτηση `select_4`

**select\_4():**

```
1. function select_4(input){
2.   pauseTimer();
3.   if(input==corrAnsw[counter-1]){
4.     // correct answer
5.     correct++;
6.   }
7.   counter++;
8.   if(counter>questions.length){
9.     stopTimer();
10.    exit();
11.    return;
12.  }
13.  game();}
```

η οποία κάνει παύση τον χρονομετρητή ,ελέγχει αν η απάντηση που επέλεξε το παιδί είναι σωστή και αν αυτή ήταν η τελευταία ερώτηση του πίνακα. Αν δεν ισχύει αυτό η διαδικασία της δοκιμασίας επαναλαμβάνεται αναδρομικά με την κλήση της συνάρτησης game. Στην αντίθετη περίπτωση σταματάει ο χρονομετρητής και η δοκιμασία ολοκληρώνεται με την κλήση της συνάρτησης exit\_3

**exit\_3():**

```
1. function exit_3(){
2.   document.getElementById("main_container")
   .style.display="none";
3.   document.getElementById("correctAnswers")
   .value=correct;
4.   document.getElementById("time").value=time;
5.   document.getElementById("myForm").submit();
6. }
```

η οποία καθαρίζει την οθόνη και περνάει τις σωστές απαντήσεις και τον χρόνο ολοκλήρωσης των δοκιμασιών στην PHP μέσω της φόρμας της HTML. Η PHP στην συνέχεια εκτελεί τον κώδικα write\_5

**write\_5():**

```
1. $correct=$_POST["correctAnswers"];
2. $time=$_POST["time"];
3. include("db.php");
4. $sql="INSERT INTO Results_5 (User,Correct_Answers,Time)
   VALUES ('".$_SESSION["user"]."', $correct, '$time')";
5. mysqli_query($db_server, $sql);
6. mysqli_close($db_server);
7. header("Location: game6.php");
```

ο οποίος ανοίγει μια καινούργια σύνδεση με την βάση δεδομένων και αποθηκεύει τις σωστές απαντήσεις και τον χρόνο στο αντίστοιχο πίνακα. Έπειτα η εφαρμογή περνάει στην επόμενη δοκιμασία.

#### 4.3.6 Δοκιμασία 6

Πριν την έναρξη της δοκιμασίας η PHP εκτελεί τον κώδικα read\_4 και διαβάζει από την βάση δεδομένων MySQL τον πίνακα Questions\_4 που περιέχει τις λίστες με τις λέξεις

των ηχητικών κομματιών μαζί με τις σωστές απαντήσεις της κάθε ερώτησης και τα αποθηκεύει σε πίνακες questions και answers αντίστοιχα.

**read\_4():**

```
1. include("db.php");
2. $query="SELECT Question,Correct_Answer
   FROM Questions_4";
3. $result=mysqli_query($db_server,$query);
4. $questions=array();
5. $answers=array()
6. if(mysqli_num_rows($result)>0){
7.     while($row=mysqli_fetch_assoc($result)){
8.         array_push($questions,$row["Question"]);
9.         array_push($answers,$row["Correct_Answer"]);
10.    }
11.    mysqli_close($db_server);
```

Οι πίνακες αυτοί μεταφέρονται από την PHP στην JavaScript με τον κώδικα move\_2 της HTML. Τα κύρια στοιχεία της δοκιμασίας αυτής είναι οι λέξεις που πρέπει να ακούσει το παιδί και τα 2 κουμπιά που εμφανίζονται ως επιλογές. Στην αρχή της δοκιμασίας το παιδί ακούει μια λέξη με την συνάρτηση game\_2

**game\_2():**

```
1. function game_2(){
2.     m_cnt.style.display="none";
3.     aud.src="audio/game4/"+corrAnsw[counter-1]+".mp3";
4.     aud.load();
5.     aud.play();
6.     aud.onended=function(){
7.         tmp=questions[counter-1].split(",");
8.         btn_1.innerHTML=tmp[0];
9.         btn_2.innerHTML=tmp[1];
10.        m_cnt.style.display="grid";
11.        startTimer();
12.    }
```

η οποία αυτή καθαρίζει την οθόνη, ορίζει στο κομμάτι ήχου της HTML την κατάλληλη διεύθυνση, το φορτώνει, το αναπαράγει και εμφανίζει στην οθόνη τα κουμπιά με τις πιθανές απαντήσεις της αντίστοιχης ερώτησης. Πριν την έξοδο από την συνάρτηση αυτή, ενεργοποιείται και ο χρονομετρητής. Όταν το παιδί επιλέξει την απάντηση του κάνοντας

κλικ σε ένα από τα 2 κουμπιά καλείται η συνάρτηση `select_4` που είναι ίδια με της δοκιμασίας 5. Η συνάρτηση κάνει παύση τον χρονομετρητή, ελέγχει αν η απάντηση που επέλεξε το παιδί είναι σωστή και αν αυτή ήταν η τελευταία ερώτηση του πίνακα. Αν δεν ισχύει αυτό η διαδικασία της δοκιμασίας επαναλαμβάνεται αναδρομικά με την κλήση της συνάρτησης `game_2`. Στην αντίθετη περίπτωση σταματάει ο χρονομετρητής και η δοκιμασία ολοκληρώνεται με την κλήση της συνάρτησης `exit_3` η οποία καθαρίζει την οθόνη και περνάει τις σωστές απαντήσεις και τον χρόνο ολοκλήρωσης των δοκιμασιών στην PHP μέσω της φόρμας της HTML. Η PHP στην συνέχεια εκτελεί τον κώδικα `write_6`

**write\_6():**

```
1. $correct=$_POST["correctAnswers"];
2. $time=$_POST["time"];
3. include("db.php");
4. $sql="INSERT INTO Results_6 (User,Correct_Answers,Time)
   VALUES ('".$_SESSION["user"]."', $correct, '$time')";
5. mysqli_query($db_server, $sql);
6. mysqli_close($db_server);
7. header("Location: game7.php");
```

ο οποίος ανοίγει μια καινούργια σύνδεση με την βάση δεδομένων και αποθηκεύει τις σωστές απαντήσεις και τον χρόνο στο αντίστοιχο πίνακα. Έπειτα η εφαρμογή περνάει στην επόμενη δοκιμασία

#### 4.3.7 Δοκιμασία 7

Πριν την έναρξη της δοκιμασίας η JavaScript φορτώνει τις εικόνες που θα χρειαστούν για την δοκιμασία με την συνάρτηση `pre_load`.

```
1. function pre_load(){
2.     //images pre loading
3.     var img;
4.     for(var i=1;i<11;i++){
5.         img=new Image();
6.         img.src=("images/game5/"+"q"+i+".png");
7.         images.push(img);
8.         for(var j=1;j<4;j++){
9.             img=new Image();
10.            img.src=("images/game5/"+"q"+i+"_a"+j+".png");
11.            images.push(img);
```

```
12.     }}}
```

Τα πιο σημαντικά στοιχεία της δοκιμασίας αυτής είναι οι 4 εικόνες, η πρώτη ως ερώτηση και οι άλλες 3 σε μορφή κουμπιών ως απαντήσεις. Στην αρχή της δοκιμασίας εκτελείται η συνάρτηση `game_3` και εμφανίζονται στην οθόνη οι 4 πρώτες εικόνες του πίνακα `images`, στον οποίο είναι αποθηκευμένες συνολικά όλες.

**game\_3():**

```
1. function game_3(){
2. // load images
3. tmp=(counter-1)*4;
4. document.getElementById("main_container")
   .style.display="none";
5. setTimeout(function(){
6. document.getElementById("main_container")
   .style.display="grid";
7. }, 1000);
8.     qstn.src=images[tmp].src;
9.     btn_1.src=images[tmp+1].src;
10.    btn_2.src=images[tmp+2].src;
11.    btn_3.src=images[tmp+3].src;
12. }
```

Όταν το κλικάρει σε μια από τις 3 πιθανές απαντήσεις εκτελείται η συνάρτηση `select_5`

```
1. function select_5(input){
2.     if(input==corrAnsw[counter-1]){
3.         // correct answer
4.         correct++;
5.     }
6.     counter++;
7.     if(counter>corrAnsw.length){
8.         stopTimer();
9.         exit();
10.        return;
11.    }
12.    game();
13. }
```

η οποία ελέγχει αν η απάντηση που έδωσε το παιδί είναι σωστή και αν αυτή ήταν η τελευταία ερώτηση της δοκιμασίας. Αν δεν είναι η τελευταία, τότε καλείται η συνάρτηση

game\_3 η οποία εξαφανίζει τις εικόνες για 1 δευτερόλεπτο, φορτώνει τις κατάλληλες εικόνες για την επόμενη ερώτηση και μετά το πέρας του χρόνου αυτού εμφανίζει τις καινούργιες πλέον εικόνες. Αν αντιθέτως η ερώτηση είναι η τελευταία της δοκιμασίας σταματάει ο χρόνος και καλείται η συνάρτηση exit\_3. Η συνάρτηση καθαρίζει την οθόνη και στέλνει τις σωστές απαντήσεις και τον χρόνο ολοκλήρωσης των δοκιμασιών στην PHP μέσω της φόρμας της HTML. Η PHP με την σειρά της εκτελεί τον κώδικα write\_7

**write\_7():**

```
1. $correct=$_POST["correctAnswers"];
2. $time=$_POST["time"];
3. include("db.php");
4. $sql="INSERT INTO Results_7(User,Correct_Answers,Time)
VALUES ('$SESSION["user"].','$correct','$time)";
5. mysqli_query($db_server,$sql);
6. mysqli_close($db_server);
7. header("Location: end.php");
```

ο οποίος ανοίγει μια καινούργια σύνδεση με την βάση δεδομένων και αποθηκεύει τις σωστές απαντήσεις και τον χρόνο στο αντίστοιχο πίνακα. Έπειτα η εφαρμογή ολοκληρώνεται με την μετάβαση της στην τελευταία σελίδα για την εμφάνιση του αντίστοιχου μηνύματος τέλους.

## 5. ΑΠΟΤΕΛΕΣΜΑΤΑ

Για την ανάλυση των αποτελεσμάτων του συνόλου των δοκιμασιών της διαδικτυακής εφαρμογής ανίχνευσης πραγματοποιήθηκε περιγραφική στατιστική ανάλυση με στόχο την παρουσίαση των μέσων όρων των απαντήσεων των συμμετεχόντων στις επτά δοκιμασίες. Επιπλέον, για τη σύγκριση των αποτελεσμάτων σε παιδιά με τυπική ανάπτυξη και παιδιά που κρίθηκε από τους εκπαιδευτικούς τους ότι παρουσιάζουν ψυχοκινητική καθυστέρηση διεξήχθη ανάλυση διακύμανσης (ANOVA). Τα αποτελέσματα παρουσιάζονται στον πίνακα 5.

*Πίνακας 5. Παρουσιάζονται οι μέσοι όροι και οι τυπικές αποκλίσεις των απαντήσεων των συμμετεχόντων καθώς και οι διαφορές που παρουσιάζονται.*

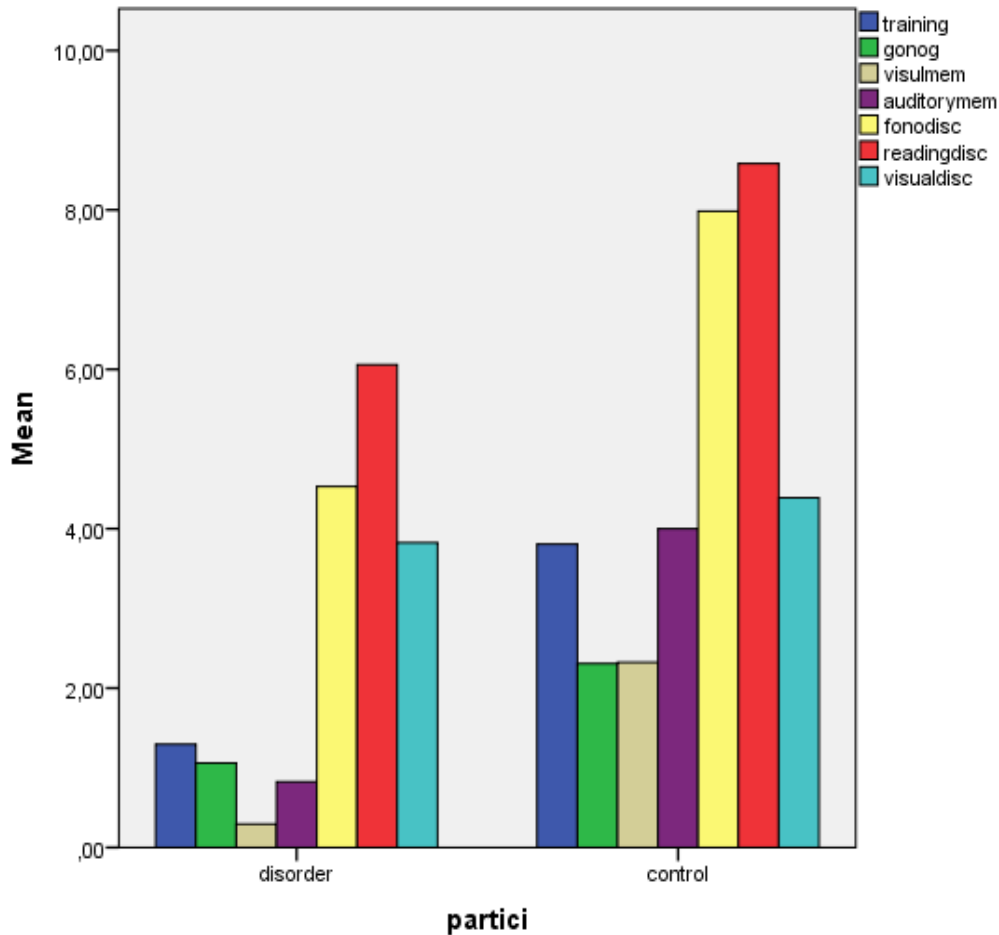
| Δοκιμασίες                 | Ομάδες                              |      |               |      | F        |
|----------------------------|-------------------------------------|------|---------------|------|----------|
|                            | Παιδιά με σύσταση των εκπαιδευτικών |      | Ομάδα ελέγχου |      |          |
|                            | Mean                                | SD   | Mean          | SD   |          |
| Εξάσκηση                   | 1.29                                | 1.92 | 3.80          | 2.06 | 20.314** |
| Εκτελεστικές λειτουργίες   | 1.06                                | 1.24 | 2.30          | 0.98 | 19.010** |
| Οπτική εργαζόμενη μνήμη    | 0.23                                | 0.59 | 2.32          | 2.31 | 12.691   |
| Ακουστική εργαζόμενη μνήμη | 0.82                                | 0.95 | 4.00          | 2.21 | 33.173** |
| Γραφοφωολογική διάκριση    | 4.52                                | 1.59 | 7.98          | 1.95 | 44.870** |
| Φωολογική ενημερότητα      | 6.06                                | 1.60 | 8.59          | 1.82 | 22.772** |
| Οπτικοχωρική αντίληψη      | 3.82                                | 1.13 | 4.39          | 1.42 | 2.279    |

Σημείωση: \*\* $p < 0.01$

Από την παρουσίαση των αποτελεσμάτων γίνεται κατανοητό το γεγονός ότι τα παιδιά με γνωστικές αδυναμίες παρουσίασαν στατιστικά σημαντικά χαμηλότερους μέσους όρους σωστών απαντήσεων ( $p < 0.01$ ) σε έξι από τις 7 δοκιμασίες της συστοιχίας ανίχνευσης σε σύγκριση με τα παιδιά της ομάδας ελέγχου. Στη δοκιμασία οπτικοχωρικής αντίληψης τα παιδιά με γνωστικές αδυναμίες παρόλο που παρουσίασαν λιγότερο αριθμό σωστών απαντήσεων σε σύγκριση με τα παιδιά της ομάδας ελέγχου δεν παρουσίασαν στατιστικά

σημαντικές διαφορές ( $p>0.05$ ). Οι μέσοι όροι των απαντήσεων των συμμετεχόντων στο σύνολο των δοκιμασιών απεικονίζονται στο γράφημα 1.

*Γράφημα 1. Μέσοι όροι σωστών απαντήσεων παιδιών με γνωστικές αδυναμίες και παιδιών της ομάδας ελέγχου στο σύνολο των δοκιμασιών.*



Επιπλέον, περιγραφική στατιστική ανάλυση χρησιμοποιήθηκε για την εύρεση των μέσων όρων και των τυπικών αποκλίσεων στο χρόνο ολοκλήρωσης της συστοιχίας των δοκιμασιών. Προκειμένου να εντοπιστούν διαφορές στο χρόνο συμπλήρωσης της δοκιμασίας μεταξύ παιδιών με γνωστικές αδυναμίες και παιδιών της ομάδας ελέγχου χρησιμοποιήθηκε ανάλυση διακύμανσης (ANOVA). Τα αποτελέσματα της ανάλυσης παρουσιάζονται στον πίνακα 6.



**Πίνακας 6. Παρουσιάζονται οι μέσοι όροι και οι τυπικές αποκλίσεις στο χρόνο απάντησης των συμμετεχόντων καθώς και οι διαφορές που παρουσιάζονται.**

| <i>Δοκιμασίες</i>        | <i>Ομάδες</i>                              |      |                      |      |       |
|--------------------------|--|------|----------------------|------|-------|
|                          | <i>Παιδιά με σύσταση των εκπαιδευτικών</i> |      | <i>Ομάδα ελέγχου</i> |      |       |
|                          | Mean                                       | SD   | Mean                 | SD   | F     |
| Γραφοφωνολογική διάκριση | 0.66                                       | 2.50 | 0.51                 | 0.39 | 1.745 |
| Φωνολογική ενημερότητα   | 0.55                                       | 0.56 | 0.49                 | 0.42 | 0.157 |
| Οπτικοχωρική αντίληψη    | 1.47                                       | 1.81 | 1.27                 | 0.58 | 0.305 |

Από την ανάλυση των αποτελεσμάτων στον πίνακα 2 καταδεικνύεται το γεγονός ότι το σύνολο των παιδιών με γνωστικές αδυναμίες χρειάστηκε περισσότερο χρόνο προκειμένου να ολοκληρώσει τις δοκιμασίες στην οποίες μετρήθηκε ο χρόνος σε σύγκριση με τα παιδιά της ομάδας ελέγχου. Ωστόσο, δεν παρουσιάστηκαν στατιστικά σημαντικές διαφορές ( $p>0.05$ ). Στο σημείο αυτό κρίνεται απαραίτητο να υπενθυμίσουμε ότι στην δοκιμασία εξάσκησης, στη δοκιμασία ελέγχου των εκτελεστικών λειτουργιών και στις δοκιμασίες εργαζόμενης ακουστικής και οπτικής μνήμης δεν υπολογίστηκε ο χρόνος απαντήσεων των συμμετεχόντων.

Τέλος, κρίθηκε αναγκαίος ο υπολογισμός της αξιοπιστίας των δοκιμασιών της συστοιχίας ανίχνευσης γνωστικών δυνατοτήτων και αδυναμιών των παιδιών προσχολικής ηλικίας. Τα αποτελέσματα από τον υπολογισμό του βαθμού αξιοπιστίας όπως παρουσιάζεται από το Cronbach's alpha κρίνεται ικανοποιητικός ( $\alpha=0.767$ ) για το σύνολο των δοκιμασιών που χρησιμοποιήθηκαν.

## 6. ΣΥΜΠΕΡΑΣΜΑΤΑ

Τα αποτελέσματα του συγκεκριμένου ερευνητικού πρωτοκόλλου υποστηρίζουν την πρώτη ερευνητική υπόθεση που αναφέρει ότι τα παιδιά με γνωστικές αδυναμίες θα παρουσιάσουν χαμηλότερους μέσους όρους στις δοκιμασίες της διαδικτυακής συστοιχίας ανίχνευσης. Ωστόσο, κρίνεται αναγκαίο να επισημανθεί το γεγονός ότι τα παιδιά που παρουσίασαν γνωστική αδυναμία, σύμφωνα με τις εκτιμήσεις των εκπαιδευτικών, σημείωσαν στατιστικά σημαντικό ( $p < 0.01$ ) μικρότερο αριθμό σωστών απαντήσεων, στο σύνολο των δοκιμασιών που χορηγήθηκαν εκτός από τη δοκιμασία ελέγχου της οπτικοχωρικής αντίληψης. Στη συγκεκριμένη υποδοκιμασία παρόλο που παρουσίασαν χαμηλότερους μέσους όρους απαντήσεων σε σύγκριση με την ομάδα ελέγχου δεν παρουσίασαν στατιστικά σημαντικές διαφορές ( $p > 0.05$ ).

Η δεύτερη ερευνητική υπόθεση του παρόντος ερευνητικού πρωτοκόλλου μερικών επαληθεύεται. Αναλυτικότερα, τα παιδιά με γνωστικές αδυναμίες, σύμφωνα με τις εκτιμήσεις των εκπαιδευτικών, παρουσίασαν μεγαλύτερους μέσους όρους ολοκλήρωσης των υποδοκιμασιών σε σύγκριση με τα παιδιά της ομάδας ελέγχου. Ωστόσο, ο μεγαλύτερος χρόνος που χρειάστηκε να ολοκληρώσουν τις υποδοκιμασίες δεν παρουσίασε στατιστικά σημαντικές διαφορές ( $p > 0.05$ ) σε σύγκριση με το χρόνο που χρειάστηκαν τα παιδιά της ομάδας ελέγχου.

Επιπλέον, αξίζει να επισημανθεί το γεγονός ότι από την ανάλυση αξιοπιστίας των υποδοκιμασιών που σχεδιάστηκαν και εφαρμόστηκαν στο παρόν ερευνητικό πρωτόκολλο προέκυψε το γεγονός ότι συγκέντρωσαν ικανοποιητικό βαθμό εφόσον ο βαθμός αξιοπιστίας, όπως υπολογίζεται από το Cronbach's alpha ήταν  $\alpha = 0.767$ . Υπενθυμίζουμε ότι στη συγκεκριμένη στατιστική ανάλυση τα εργαλεία που χρησιμοποιούνται θεωρούνται αξιόπιστα, εφόσον το  $\alpha$  είναι μεγαλύτερο από το 0.70.

Βασική αδυναμία του παρόντος ερευνητικού πρωτοκόλλου αποτελεί το γεγονός ότι η συστοιχία ανίχνευσης των γνωστικών αδυναμιών των παιδιών προσχολικής ηλικίας ήταν ότι χρησιμοποιήθηκε μικρός αριθμός συμμετεχόντων ( $N = 79$ ). Ωστόσο, η συγκεκριμένη αδυναμία μερικώς αναιρείται εξαιτίας των σημαντικών διαφορών στο μέσο όρο των απαντήσεων των παιδιών που συμμετείχαν στην έρευνα ( $p < 0.01$ ). Επιπλέον, περιορισμό

των αποτελεσμάτων της έρευνας αποτελεί το γεγονός ότι για την συμμετοχή των παιδιών στις ερευνητικές ομάδες διαδραμάτισε σημαντικό ρόλο η περιγραφή των παιδιών και η εκτίμηση του εκπαιδευτικού προσχολικής εκπαίδευσης.

Στο σημείο αυτό κρίνεται αναγκαίο να επισημανθεί ότι συγκεκριμένη συστοιχία δοκιμασιών εφόσον σταθμιστεί σε μεγάλο δείγμα του γενικού πληθυσμού και παρουσιάσει αντίστοιχα αποτελέσματα μπορεί να αποτελέσει έναν τρόπο ανίχνευσης των γνωστικών δυνατοτήτων και αδυναμιών των παιδιών προσχολικής ηλικίας, προκειμένου να παραπεμφθούν στη διεπιστημονική ομάδα προκειμένου να πραγματοποιηθεί η διάγνωση και ο σχεδιασμός του προγράμματος αποκατάστασης. Τέλος, η διαδικτυακή δοκιμασία που σχεδιάστηκε και εφαρμόστηκε στο παρόν ερευνητικό πρωτόκολλο, μπορεί να αποτελέσει οδηγό για τους παιδαγωγούς στη διαμόρφωση εξατομικευμένου προγράμματος ενίσχυσης των γνωστικών δυνατοτήτων των παιδιών προσχολικής ηλικίας.

## **ΒΙΒΛΙΟΓΡΑΦΙΑ**

BARRET, D. E & FRANK, D. A. (1987). The effects of undernutrition on children's behavior. New York: Gordon & Breach.

CLARK, J. E., & HUMPHREY, L. H. (Eds.) (1985). Motor development: Current selected research. Princeton, NJ: Princeton Book Company.

Craig, G. J., & Baucum, D. (2007). Η Ανάπτυξη του Ανθρώπου (9η Αμερικανική εκδ., Τόμ. Γ'). (ΒορριάΠ., Επιμ., & ΙωαννίδουΑ., Μεταφρ.) Αθήνα: Παπαζήση

Hayes,N. (1998). Εισαγωγή στην Ψυχολογία (9η εκδ., Τόμ. Γ'). (Κωσταρίδου-ΕυκλείδηΑ., Επιμ., ΣπανούδηςΓ., & ΣύρμαληΚ., Μεταφρ.) Αθήνα: Ελληνικά Γράμματα.

KOIVISTO, M. & REVONSUO, A. (2003). Object recognition in the cerebral hemispheres as revealed by visual field experiments. *Laterality: Asymmetries of Body, Brain & Cognition*, 8, 135-153.

LEATHERS, H. D & FOSTER, P. (2004). U.S. rate of sexual sideades highest in developed world. *The New York Times*, p. C1

LOWREY, G. H. (1986). Growth and development of children (8th ed.). Chicago: Year Book Medical Publishers.

MCAULIFFE, S. P. & KNOWLTON, B. J. (2001). Hemispheric differences in object identification. *Brain & Cognition*, 45, 119-128.

NIHART, M. A. (1993). Growth and development of the brain. *Journal of the Child and Adolescent Psychiatric and Mental Health Nursing*, 6, 39-40.

OGDEN, C. L., KUCZMARSKI, R. J., FLEGAL, K. M., MEI, Z., GUO, S., WEI, R., GRUMMER-STRAWN, L. M., CURTIN, L. R., ROCHE, A. F., & JOHNSON, C. L. (2002). Center for Disease Control and Prevention 2000 growth charts for the United States: Improvements to the 1977 National Center for Health Statistics Version. *Pediatrics*, 109, 45-60.

POLLAK, S., HOLT, L., & WISMER FRIES, A. (2004). Hemispheric asymmetries in children's perception of nonlinguistic human affective sounds. *Developmental Science*, 7, 10-18.

Russell, W. R., & Dewar, A. (1992). Εξηγώντας τον εγκέφαλο. (ΚιντήΑ., Επιμ., & ΚουσουλάκουΒ., Μεταφρ.) Αθήνα: Τροχαλία.

SCHUSTER, C. S. & ASHBURN, S. S. (1986). The process of human development (2nd ed.). Boston: Little, Brown.

Siegler, R. S. (2002). Πως Σκέφτονται τα Παιδιά. (ΒοσνιάδουΣ., Επιμ., & ΚουλεντιανούΖ., Μεταφρ.) Αθήνα: Gutenberg.

Sternberg, R. J. (2007). Γνωστική Ψυχολογία (1η εκδ.). (ΞανθάκουΓ., ΚαΐλαΜ., Επιμ., ΒραχωρίτουΙ., Μεταφρ.) Αθήνα: Ατραπός.

United Nations (1991). Declaration of the world summin for children. New York: Author.

Βότσος, Ι. (1992). Η Γλωσσική Εξέλιξη στο Παιδί. Θεσσαλονίκη: ΖΗΤΗ.

Ζυγούρης, Ν. (2016). Πανεπιστημιακές σημειώσεις σε ηλεκτρονική μορφή. Εγκεφαλικές περιοχές και Μνήμη. Λαμία

Ζυγούρης, Ν. (2016). Πανεπιστημιακές σημειώσεις σε ηλεκτρονική μορφή. Μνήμη και Μάθηση. Λαμία

Κεντερλής, Π. Δ. (2009). ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΩΝ ΕΦΑΡΜΟΓΩΝ Θεωρία και Πράξη: Λύχνος

Σαμαρτζή, Σ. (1995). Εισαγωγή στις Γνωστικές Λειτουργίες. Αθήνα: Παπαζήση.

Στασινός, Δ. Π. (2009). Ψυχολογία του Λόγου και της Γλώσσας, Ανάπτυξη και αθολογία, Δυσλεξία και Λογοθεραπεία (1η εκδ.). Αθήνα: Gutenberg