



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ  
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΜΗΜΑ  
ΠΛΗΡΟΦΟΡΙΚΗΣ**

**Πτυχιακή Εργασία**

Τίτλος Πτυχιακής Εργασίας	Εκμάθηση αλφάβητου (Εκπαιδευτική τηλεόραση: ΕΝΑ ΓΡΑΜΜΑ, ΜΙΑ ΙΣΤΟΡΙΑ)  Learning the alphabet (educational TV: A LETTER, A STORY)
Όνοματεπώνυμο Φοιτητή	ΕΥΑΓΓΕΛΟΣ ΓΑΒΡΙΛΗΣ
Πατρώνυμο	ΚΩΝΣΤΑΝΤΙΝΟΣ ΓΑΒΡΙΛΗΣ
Αριθμός Μητρώου	Π17193
Επιβλέπων	ΠΑΝΑΓΙΩΤΗΣ ΤΣΑΚΩΝΑΣ

**ΠΕΙΡΑΙΑΣ, ΙΟΥΝΙΟΣ 2024**

---

## Copyright ©

---

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν αποκλειστικά τον συγγραφέα και δεν αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πειραιώς.

Ως συγγραφέας της παρούσας εργασίας δηλώνω πως η παρούσα εργασία δεν αποτελεί προϊόν λογοκλοπής και δεν περιέχει υλικό από μη αναφερόμενες πηγές.

---

## Ευχαριστίες

---

Θα ήθελα να εκφράσω τη βαθύτατη ευγνωμοσύνη μου στον καθηγητή μου, Τσάκωνα Παναγιώτη για τη συνεχή υποστήριξη και καθοδήγησή του κατά τη διάρκεια της συγγραφής της διατριβής. Οι πολύτιμες συνεισφορές του και η συνεχής ενθάρρυνση αποτέλεσαν τον ακρογωνιαίο λίθο της επιτυχίας αυτής της έρευνας. Θα ήθελα επίσης να εκφράσω την ειλικρινή μου ευγνωμοσύνη στο Πανεπιστήμιο Πειραιώς για τις ευκαιρίες και τους πόρους που συνέβαλαν τόσο πολύ στην ακαδημαϊκή και προσωπική μου ανάπτυξη.

## Περίληψη

---

### Επισκόπηση εφαρμογής Εκμάθηση αλφαβήτου

Η εφαρμογή «Μαθαίνοντας το αλφάβητο» αποτελεί μέρος του εκπαιδευτικού προγράμματος A LETTER A STORY. Στόχος της εφαρμογής είναι να ενισχύσει την εκμάθηση του ελληνικού αλφαβήτου από τα παιδιά προσχολικής ηλικίας μέσω διαδραστικών και διασκεδαστικών δραστηριοτήτων. Η εφαρμογή περιλαμβάνει τέσσερις διαφορετικές κατηγορίες παιχνιδιών: «Βρες τα γράμματα », «Βρες τις εικόνες» ,«Κυνήγι θησαυρού» και «Βρες τα ζώα» ,τα οποία προσφέρουν διαφορετικές μεθόδους μάθησης και ενισχύουν την αναγνώριση των γραμμάτων και τη συσχέτιση με εικόνες και λέξεις.

Σχετικά με το έργο: Το έργο είναι ένα από τα σημαντικότερα παιχνίδια που έχουν σχεδιαστεί για να γράφουν και να διαβάζουν τα γράμματα της ελληνικής αλφαβήτου.

Η εργασία αυτή αφορά την ανάπτυξη μιας εκπαιδευτικής εφαρμογής για παιδιά προσχολικής ηλικίας για την εκμάθηση των ελληνικών γραμμάτων. Η εφαρμογή περιλαμβάνει τέσσερα διαφορετικά παιχνίδια που στοχεύουν στην ενίσχυση της αναγνώρισης γραμμάτων, της οπτικής μνήμης και των γλωσσικών συσχετισμών :στο παιχνίδι «Βρες το γράμμα », τα παιδιά αναγνωρίζουν γράμματα από ήχους και εικόνες, στο «Βρες την εικόνα »,τα παιδιά αντιστοιχούν γράμματα με τις αντίστοιχες εικόνες. Το «Κυνήγι θησαυρού »παρέχει μια πιο περιπετειώδη μαθησιακή εμπειρία, ενώ το «Βρες τα ζώα »συνδυάζει γράμματα και ζώα και οικοδομεί την κατανόηση μέσω του παιχνιδιού .Η εφαρμογή στοχεύει στην ανάπτυξη γλωσσικών δεξιοτήτων και στην προετοιμασία των παιδιών για την πρώτη δημοτικού με διασκεδαστικό και διαδραστικό τρόπο.

### Λέξεις Κλειδιά:

*Εκμάθηση αλφαβήτου, Εκπαιδευτική ραδιοτηλεόραση, ΕΝΑ ΓΡΑΜΜΑ ΜΙΑ ΙΣΤΟΡΙΑ, Προσχολική ηλικία, Διαδραστική μάθηση, Παιχνίδια εκπαίδευσης, Βρες το γράμμα, Βρες την εικόνα, Κυνήγι θησαυρού, Βρες το ζώο, Αναγνώριση γραμμάτων, Οπτική μνήμη, Λεκτική σύνδεση, Γλωσσικές δεξιότητες, Διασκεδαστική εκπαίδευση.*

---

## Αφιερώσεις

---

Αφιερώνω αυτήν την εργασία στην οικογένειά μου για την αμέριστη αγάπη και υποστήριξή τους καθ' όλη τη διάρκεια των σπουδών μου. Επίσης, ευχαριστώ τον αγαπημένο μου φίλο Χοακίν για την αδιάκοπη ενθάρρυνση και τις πολύτιμες συμβουλές του. Τέλος, είμαι ευγνώμων στους δασκάλους και καθηγητές μου, των οποίων η καθοδήγηση και η εμπειρία με οδήγησαν σε αυτό το σημαντικό επίτευγμα.

# Πίνακας Περιεχομένων

---

Copyright.....	1
Ευχαριστίες.....	1
Περίληψη.....	2
Λέξεις κλειδιά:.....	2
Αφιερώσεις.....	3
Πίνακας Περιεχομένων.....	4
Εισαγωγή.....	5
1.Βρες την Εικόνα.....	8
1.1 Σκοπός και Στόχοι .....	9
1.2 Μέθοδος Παιχνιδιού .....	10
1.3 Εκπαιδευτικά Οφέλη .....	11
2.Βρες το Γράμμα .....	13
2.1 Σκοπός και Στόχοι .....	13
2.2 Μέθοδος Παιχνιδιού .....	15
2.3 Εκπαιδευτικά Οφέλη .....	16
3.Κυνήγι Θησαυρού .....	17
3.1 Σκοπός και Στόχοι .....	18
3.2 Μέθοδος Παιχνιδιού .....	19
3.3 Εκπαιδευτικά Οφέλη.....	20
4.Βρες το Ζώο .....	21
4.1 Σκοπός και Στόχοι .....	21
4.2 Μέθοδος Παιχνιδιού .....	22
4.3 Εκπαιδευτικά Οφέλη .....	24
5.Αναλυτική Περιγραφή Υλοποίησης.....	26
Συμπεράσματα .....	44
Βιβλιογραφία.....	45

---

## Εισαγωγή

---

Η εκπαίδευση στην πρώιμη παιδική ηλικία αποτελεί το θεμέλιο της ανάπτυξης ενός παιδιού και το πρώτο βήμα προς τη μελλοντική επιτυχία στην εκπαίδευση και τη ζωή. Στο πλαίσιο αυτό, η εκμάθηση του αλφαβήτου διαδραματίζει καθοριστικό ρόλο, καθώς αποτελεί το πρώτο βήμα για την απόκτηση της γλώσσας και την ανάπτυξη της ανάγνωσης. Η παρούσα εργασία επικεντρώνεται στην ανάπτυξη μιας εκπαιδευτικής εφαρμογής με τίτλο «Εκμάθηση αλφαβήτου (Εκπαιδευτική εκπομπή: A LETTER A STORY)», η οποία έχει ως στόχο να ενισχύσει την εκμάθηση του ελληνικού αλφαβήτου από παιδιά προσχολικής ηλικίας μέσω διαδραστικών και διασκεδαστικών δραστηριοτήτων. Η εφαρμογή περιλαμβάνει τέσσερα διαφορετικά παιχνίδια.

Η εφαρμογή περιλαμβάνει τέσσερα διαφορετικά παιχνίδια. Τα παιχνίδια αυτά έχουν σχεδιαστεί για να παρέχουν στα παιδιά ποικίλες μεθόδους μάθησης, να αυξάνουν την αναγνώριση γραμμάτων, να συνδέουν τα γράμματα με εικόνες και λέξεις και να αναπτύσσουν την οπτική μνήμη και τις γλωσσικές δεξιότητες. Η παρούσα εργασία αναλύει τη σημασία και τα οφέλη κάθε παιχνιδιού, την παιδαγωγική προσέγγιση που ακολουθήθηκε και αξιολογεί την αποτελεσματικότητα της εφαρμογής μέσω ποιοτικών και ποσοτικών δεδομένων.

Ελπίζεται ότι η εφαρμογή αυτή θα συμβάλει σημαντικά στην ενίσχυση της εκπαίδευσης στην πρώιμη παιδική ηλικία, παρέχοντας στα παιδιά μια διασκεδαστική και διαδραστική μαθησιακή εμπειρία, και θα αποτελέσει τη βάση για μελλοντικές εκπαιδευτικές πρωτοβουλίες και έρευνες. Για την παρακάτω εφαρμογή χρησιμοποιήθηκαν τα εξής εργαλεία HTML, CSS, JAVASCRIPT, REACT.

### Ανάλυση Τεχνολογιών :

#### HTML

Η HTML περιγράφεται με τη μορφή στοιχείων HTML, τα οποία αποτελούνται από ετικέτες. Οι ετικέτες περιβάλλονται από τα σύμβολα «μεγαλύτερο από» και «μικρότερο από» (π.χ. <html>) μέσα στο περιεχόμενο μιας ιστοσελίδας- οι ετικέτες HTML αποτελούνται συνήθως από δύο ζεύγη (π.χ. <h1>και</h1>), όπου η πρώτη ετικέτα ονομάζεται «ετικέτα έναρξης» και η δεύτερη ετικέτα «ετικέτα λήξης» (αλλιώς αντίστοιχα «έναρξη»). tag» και «end tag» αντίστοιχα). Μεταξύ των ετικετών, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κ.λπ.

Ο σκοπός ενός προγράμματος περιήγησης στο διαδίκτυο είναι να διαβάσει ένα έγγραφο HTML και να το συνθέσει σε μια σελίδα που μπορεί να διαβαστεί ή να ακουστεί. Οι φυλλομετρητές δεν εμφανίζουν τις ετικέτες HTML, αλλά τις χρησιμοποιούν για να εμφανίσουν το περιεχόμενο της σελίδας. Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων σε μια σελίδα και την εμφάνιση διαδραστικών φορμών. Παρέχει έναν τρόπο δημιουργίας δομημένων

---

εγγράφων (δηλαδή εγγράφων που αποτελούνται από περιεχόμενο και τον κώδικα μορφοποίησης αυτού του περιεχομένου), καθορίζοντας δομικά σημαντικά στοιχεία του κειμένου, όπως επικεφαλίδες, παραγράφους, λίστες, συνδέσμους και παραπομπές. Τα σενάρια εντολών μπορούν επίσης να ενσωματωθούν σε γλώσσες όπως η JavaScript, οι οποίες επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML, μεταβάλλοντας τις από στατικές σε διαδραστικές.

Οι φυλλομετρητές ιστού μπορούν επίσης να αναφέρονται σε στυλ μορφής CSS για να καθορίζουν την εμφάνιση και τη διάταξη του κειμένου και άλλου υλικού. η Κοινοπραξία του Παγκόσμιου Ιστού, η οποία δημιουργεί και διατηρεί πρότυπα για την HTML και την CSS, έχει αναπτύξει μια σειρά από πρότυπα με σκοπό την παρουσίαση περιεχομένου, συμπεριλαμβανομένης της χρήσης διαφόρων στοιχείων της HTML Ενθαρρύνει τη χρήση της CSS αντί αυτής.

## **CSS**

Η CSS (Cascading Style Sheets) είναι μια γλώσσα υπολογιστών που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ που χρησιμοποιούνται για τον έλεγχο της εμφάνισης εγγράφων γραμμένων σε γλώσσες σήμανσης. Η CSS είναι μια γλώσσα υπολογιστών που στοχεύει στην ανάπτυξη ιστοσελίδων με στιλιστικό τρόπο. Δηλαδή, δίνει περισσότερες δυνατότητες από την html για τον καθορισμό περισσότερων χαρακτηριστικών, χρωμάτων και ρυθμίσεων.

## **JAVASCRIPT**

Η JavaScript (JS) είναι μια διερμηνευόμενη γλώσσα προγραμματισμού υπολογιστών . Αρχικά αποτελούσε μέρος της υλοποίησης ενός προγράμματος περιήγησης στο διαδίκτυο, το οποίο επέτρεπε στα σενάρια από την πλευρά του πελάτη να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται .

Η JavaScript είναι μια γλώσσα σεναρίων βασισμένη σε πρωτότυπα, η οποία είναι δυναμική, ασθενώς τυποποιημένη και διαθέτει συναρτήσεις ως αντικείμενα πρώτης κατηγορίας. Η σύνταξή της είναι επηρεασμένη από τη γλώσσα C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο γλώσσες δεν σχετίζονται και έχουν πολύ διαφορετική σημασιολογία. Οι σχεδιαστικές αρχές της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme που προέρχονται από και είναι μια γλώσσα πολλαπλών παραδειγμάτων που υποστηρίζει αντικειμενοστραφή , προστακτική και συνειρμική γλώσσα προγραμματισμού.

Η JavaScript χρησιμοποιείται επίσης σε μη διαδικτυακές εφαρμογές, π.χ. σε έγγραφα PDF, σε ειδικούς φυλλομετρητές ιστοτόπων, σε μικρά widgets επιφάνειας εργασίας κ.λπ. από νέες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript (όπως το Node.js), Η JavaScript γίνεται επίσης δημοφιλής για την ανάπτυξη εφαρμογών ιστού από την πλευρά του διακομιστή.

## **REACT**

Το React (επίσης γνωστό ως React.js ή ReactJS) είναι μια δωρεάν και ανοικτού κώδικα βιβλιοθήκη JavaScript για την ανάπτυξη διεπαφών χρήστη βασισμένων σε συστατικά, που παρέχεται από την Facebook Inc. από την Meta (πρώην Facebook Inc.) και συντηρείται από μια κοινότητα μεμονωμένων προγραμματιστών και εταιρειών.

Το React μπορεί να χρησιμοποιηθεί για την ανάπτυξη εφαρμογών μίας σελίδας, για κινητά τηλέφωνα ή για την απόδοση σε διακομιστή με χρήση πλαισίων όπως το Next.js. Καθώς το

---

React χειρίζεται μόνο την απόδοση της διεπαφής χρήστη και των στοιχείων στο DOM, το React εφαρμογές συχνά βασίζονται σε βιβλιοθήκες για δρομολόγηση και άλλες λειτουργίες πελάτη- το κύριο πλεονέκτημα του React είναι ότι αποφεύγεται η περιττή επανεκτέλεση αμετάβλητων στοιχείων DOM με επαναληπτική απόδοση μόνο των αλλαγμένων τμημάτων της σελίδας.

### ΑΡΧΙΚΗ ΣΕΛΙΔΑ ΕΦΑΡΜΟΓΗΣ



Εικόνα 1



---

# 1. Βρες την εικόνα

## 1.1 Σκοπός και στόχοι

Στόχοι και επιδιώξεις του παιχνιδιού Βρες την εικόνα

Ο κύριος στόχος του παιχνιδιού «Βρες την εικόνα» είναι να βελτιώσει την ικανότητα των παιδιών να αναγνωρίζουν και να διακρίνουν το ελληνικό αλφάβητο. Μέσα από μια διαδραστική και διασκεδαστική εμπειρία, το παιχνίδι στοχεύει στην εμπέδωση της γραφής των γραμμάτων και της αναγνώρισης του λόγου και στην προετοιμασία των παιδιών για την εκμάθηση της ανάγνωσης και της γραφής.

Στόχοι.

- Αναγνώριση γραμμάτων: να είναι σε θέση να αναγνωρίζουν τα γράμματα του αλφαβήτου με την όραση και τον ήχο.
- Μάθηση συσχετίσεων φωνημάτων: να ενισχυθούν οι συσχετίσεις γραμμάτων-ήχων και να βελτιωθεί η φωνητική επίγνωση.
- Ανάπτυξη οπτικής μνήμης: βελτίωση της απομνημόνευσης και αναγνώρισης γραμμάτων μέσω της επανάληψης και των διασκεδαστικών προκλήσεων.
- Απόλαυση και ενθάρρυνση: παρέχει μια διασκεδαστική και παρακινητική εμπειρία όπου τα παιδιά μπορούν να συμμετέχουν ενεργά και να απολαμβάνουν τη διαδικασία μάθησης.
- Αυτοεκτίμηση και αυτοπεποίθηση: ενισχύει την αυτοεκτίμηση και την αυτοπεποίθηση των παιδιών αναγνωρίζοντας και επιβραβεύοντάς τα όταν τα καταφέρνουν καλά κατά τη διάρκεια του παιχνιδιού.

Για την επίτευξη αυτών των στόχων, το παιχνίδι «Βρες το γράμμα» στοχεύει να παρέχει μια ισχυρή βάση για να μάθουν τα παιδιά το αλφάβητο και να προχωρήσουν στο επόμενο στάδιο των ακαδημαϊκών τους σπουδών.

## 1.2 Μέθοδος παιχνιδιού

Γενική περιγραφή.

Το παιχνίδι «Βρες την εικόνα» έχει σχεδιαστεί για να βοηθήσει τα παιδιά να αναγνωρίσουν και να αντιστοιχίσουν το ελληνικό αλφάβητο μέσα από διασκεδαστικές και διαδραστικές δραστηριότητες. Το παιχνίδι διατίθεται σε δύο επίπεδα δυσκολίας, Εύκολο και Δύσκολο, για να προκαλεί τα παιδιά ανάλογα με τις ικανότητες και τις ανάγκες τους.

## Εύκολη έκδοση

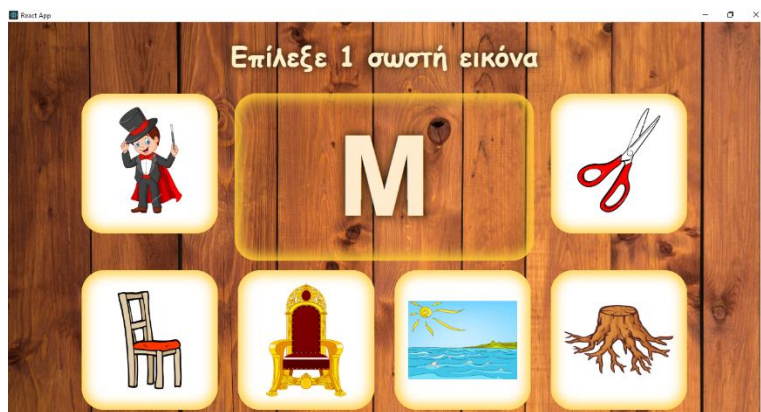
### Περιγραφή

ΜΕΘΟΔΟΣ: Στην οθόνη εμφανίζονται έξι εικόνες, εκ των οποίων μόνο μία συνδέεται σωστά με το ζητούμενο γράμμα. Οι υπόλοιπες εικόνες είναι τυχαίες ή άσχετες.

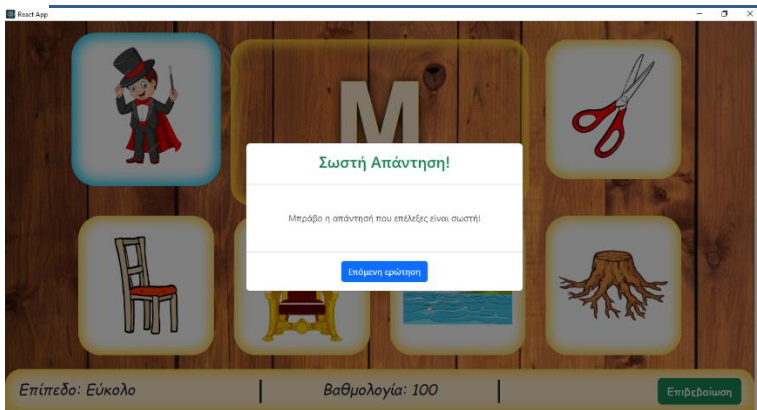
Στόχος: Το παιδί πρέπει να αναγνωρίσει την εικόνα που αντιστοιχεί στο συγκεκριμένο γράμμα και να επιλέξει τη σωστή εικόνα από τις έξι διαθέσιμες εικόνες.



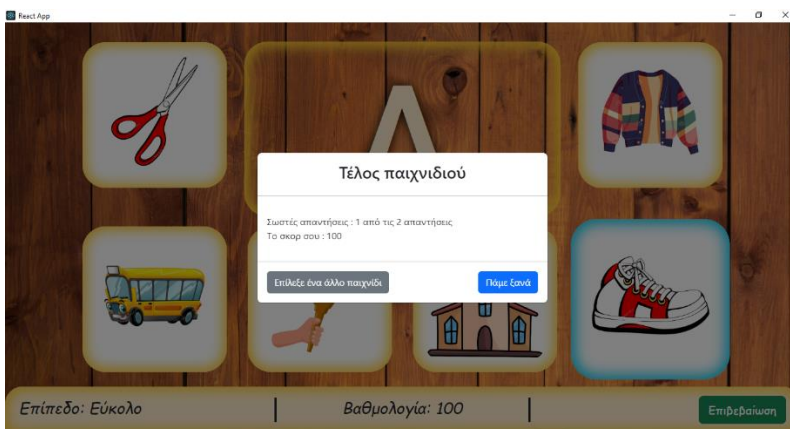
Εικόνα 1.1.2



Εικόνα 1.1.3



Εικόνα 1.1.4



Εικόνα 1.1.5

### Δύσκολη έκδοση

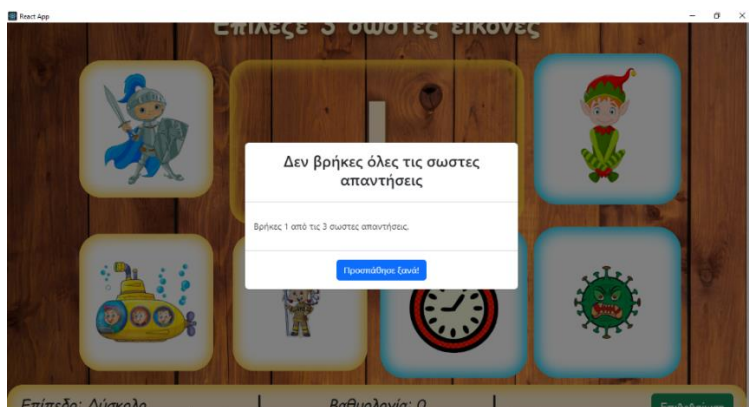
**ΠΕΡΙΓΡΑΦΗ:** Η δύσκολη έκδοση καθιστά το παιχνίδι πιο δύσκολο, προκειμένου να ενισχυθούν περαιτέρω οι γνωστικές δεξιότητες του παιδιού.

**ΜΕΘΟΔΟΣ:** Εδώ, το παιδί καλείται να βρει την εικόνα που αντιστοιχεί σε ένα δεδομένο γράμμα, αλλά ο αριθμός των σωστών εικόνων που εμφανίζονται είναι μεγαλύτερος από ό,τι στην εύκολη έκδοση (π.χ. 2-3).

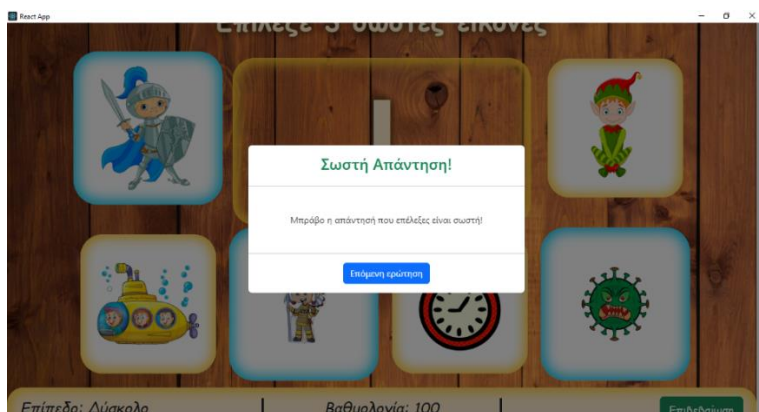
**Στόχος:** Το παιδί πρέπει να βρει και να επιλέξει τις σωστές εικόνες από έναν μεγάλο αριθμό επιλογών. Αυτό αυξάνει το επίπεδο δυσκολίας και την ικανότητα αναγνώρισης γραμμάτων από πιο συγκεχυμένες εικόνες.



Εικόνα 1.2.1



Εικόνα 1.2.2



Εικόνα 1.2.3

---

### 1.3 Εκπαιδευτικά οφέλη

Το παιχνίδι «Βρες την εικόνα » έχει σημαντικά εκπαιδευτικά οφέλη, καθώς ενισχύει την αναγνώριση γραμμάτων και φωνημάτων. Στην εύκολη έκδοση, βοηθάει τους συσχετισμούς γραμμάτων-εικόνων και βελτιώνει την οπτική μνήμη. Η πιο δύσκολη έκδοση, από την άλλη πλευρά, βελτιώνει τη συγκέντρωση, την κριτική σκέψη και την κρίση βοηθώντας τους μαθητές να αναγνωρίσουν γράμματα από πιο συγκεχυμένες εναλλακτικές λύσεις. Σε κάθε έκδοση, το παιχνίδι παρέχει μια διασκεδαστική και εκπαιδευτική εμπειρία που ενισχύει την αυτοπεποίθηση και την αυτοεκτίμηση και προετοιμάζει τα παιδιά για μελλοντικές ακαδημαϊκές προκλήσεις.

---

## 2. Βρες το Γράμμα

---

### 2.1 Σκοπός και Στόχοι

Στόχοι και σκοποί του παιχνιδιού Βρες το Γράμμα.

Ο κύριος στόχος του παιχνιδιού «Βρες το Γράμμα» είναι να αναπτύξει την ικανότητα των παιδιών να συσχετίζουν το ελληνικό αλφάβητο με τις αντίστοιχες εικόνες. Μέσω αυτής της διαδραστικής διαδικασίας, το παιχνίδι αναπτύσσει την αναγνώριση γραμμάτων και φωνημάτων και βελτιώνει τις οπτικές και γλωσσικές δεξιότητες.

#### Στόχοι

- Συσχέτιση γραμμάτων-εικόνων: να συνδέσουν οπτικά γράμματα και εικόνες για να βελτιώσουν τη μνήμη και την κατανόηση.
- Μάθηση της αναγνώρισης: να βελτιωθεί η ικανότητα αναγνώρισης γραμμάτων μέσω του συνδυασμού γραμμάτων και σχετικών εικόνων.
- Ανάπτυξη οπτικής μνήμης: βελτίωση της οπτικής μνήμης μέσω της επανάληψης της αναγνώρισης και του συσχετισμού.
- Φωνολογική μάθηση: βοηθά τα παιδιά να κατανοήσουν τους συσχετισμούς ήχου-γραμμάτων μέσω της χρήσης εικόνων.
- Διασκεδαστικό και παρακινητικό: παρέχει μια διασκεδαστική και παρακινητική εμπειρία, ενθαρρύνοντας τα παιδιά να συμμετέχουν ενεργά και να απολαμβάνουν τη διαδικασία μάθησης.

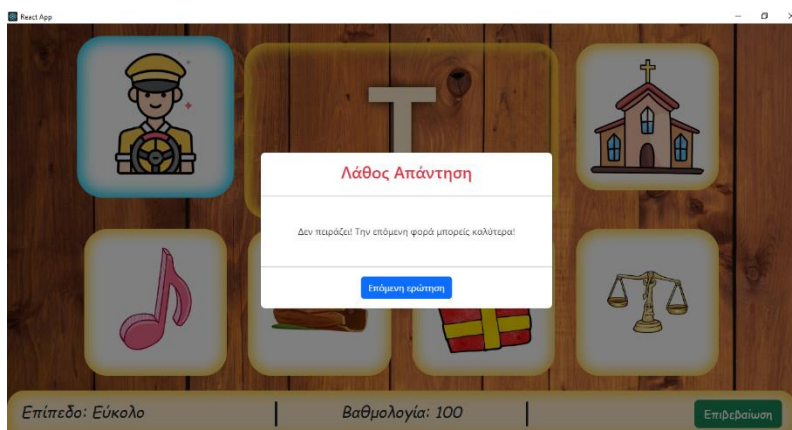
#### **Εύκολη έκδοση.**

- Μέθοδος: ένα γράμμα και έξι εικόνες εμφανίζονται στην οθόνη. Το παιδί πρέπει να επιλέξει την εικόνα που αντιστοιχεί στο γράμμα που δίνεται.

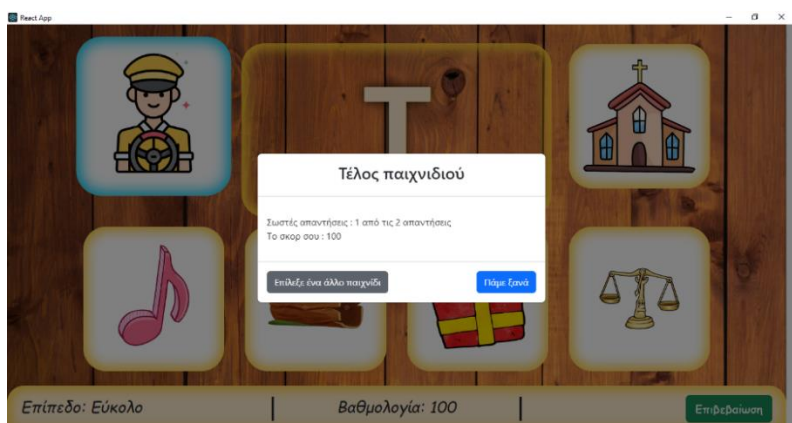
- Στόχος: να διασφαλιστεί η σωστή αντιστοίχιση γραμμάτων και εικόνων και να ενισχυθούν οι άμεσες συνδέσεις μεταξύ γραμμάτων και σχετικών αντικειμένων.



Εικόνα 2.1.1



Εικόνα 2.1.2



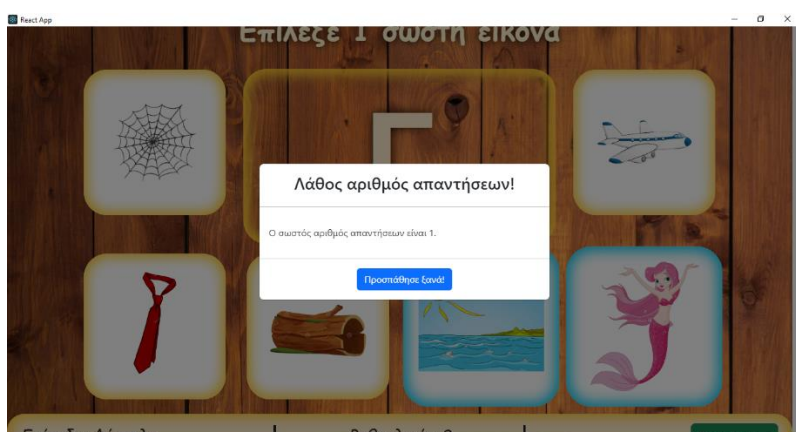
Εικόνα 2.1.3

### Δύσκολη Έκδοση

- Μέθοδος: αν δοθεί ένα γράμμα και έξι εικόνες στην οθόνη, το σωστό γράμμα αντιστοιχεί σε δύο από τις εικόνες. Το παιδί πρέπει να επιλέξει μία σωστή εικόνα.
- Στόχος: αύξηση της δυσκολίας και πρόκληση της ικανότητας των παιδιών να αναγνωρίζουν και να επιλέγουν τη σωστή εικόνα σε πιο σύνθετα πλαίσια.



Εικόνα 2.2.1



Εικόνα 2.2.2

## 2.2 Μέθοδος Παιχνιδιού

Γενική περιγραφή.

Το παιχνίδι «Αναζήτηση εικόνας» έχει σχεδιαστεί για να βοηθήσει τα παιδιά να συσχετίσουν το ελληνικό αλφάβητο με τις αντίστοιχες εικόνες. Το παιχνίδι διατίθεται σε δύο επίπεδα δυσκολίας, εύκολο και δύσκολο, για να προκαλεί τα παιδιά ανάλογα με τις ικανότητες και τις ανάγκες τους.

### Εύκολη έκδοση

**Μέθοδος:** Στην εύκολη έκδοση, το κείμενο εμφανίζεται στην οθόνη. Εμφανίζονται πέντε εικόνες, εκ των οποίων μόνο μία αντιστοιχεί στο συγκεκριμένο γράμμα. Το παιδί πρέπει να επιλέξει την εικόνα που αντιστοιχεί σωστά στο γράμμα.



---

Στόχος: Να ενισχυθεί η άμεση σύνδεση μεταξύ γραμμάτων και αντικειμένων με την αναγνώριση και αντιστοίχιση γραμμάτων και των αντίστοιχων εικόνων.

### **Δύσκολη έκδοση**

Μέθοδος: Στη δύσκολη έκδοχή, τα γράμματα εμφανίζονται επίσης στην οθόνη. Εμφανίζονται πέντε εικόνες και το σωστό γράμμα αντιστοιχεί σε δύο από αυτές. Το παιδί πρέπει να επιλέξει όσες εικόνες περιέχουν το σωστό γράμμα.

Στόχος: Να αυξηθεί το επίπεδο δυσκολίας των παιδιών και η ικανότητά τους να αναγνωρίζουν και να επιλέγουν τη σωστή εικόνα σε ένα πιο σύνθετο περιβάλλον.

## **2.2 Εκπαιδευτικά οφέλη**

Το παιχνίδι «Βρες το Γράμμα» έχει σημαντικά εκπαιδευτικά οφέλη, όπως η βελτίωση της αναγνώρισης γραμμάτων και της φωνητικής επίγνωσης, καθώς τα παιδιά συνδέουν οπτικά τα γράμματα με τις αντίστοιχες εικόνες. Στην εύκολη έκδοση, το παιχνίδι ενισχύει την άμεση σύνδεση μεταξύ γραμμάτων και εικόνων και βελτιώνει την οπτική μνήμη και την αναγνωστική κατανόηση. Στην πιο δύσκολη έκδοση, τα παιχνίδια αυξάνονται σε δυσκολία και βελτιώνουν την κριτική σκέψη, τη διάκριση και την κρίση. Και στις δύο εκδόσεις, η επιτυχής ολοκλήρωση της πρόκλησης ενισχύει την αυτοπεποίθηση και την αυτοεκτίμηση των παιδιών, δημιουργώντας μια διασκεδαστική και παρακινητική εμπειρία μάθησης.

---

## 3. Κυνήγι Θησαυρού

### 3.1 Σκοπός και Στόχοι

Στόχος: Ο κύριος στόχος του «Κυνηγιού του Θησαυρού» είναι να αναπτύξει την ικανότητα των παιδιών να αναγνωρίζουν το ελληνικό αλφάβητο συσχετίζοντας τα σχήματα των γραμμάτων με τα σχήματα των αντικειμένων. Μέσα από αυτή τη διαδραστική διαδικασία, τα παιδιά μπορούν να αναπτύξουν τις οπτικές και γλωσσικές τους δεξιότητες και να αυξήσουν τη φωνητική τους επίγνωση και την αναγνώριση γραμμάτων.

- Συνδέσεις σχημάτων: να συνδέσουν οπτικά τα σχήματα των αντικειμένων με τα αντίστοιχα γράμματα, ενισχύοντας έτσι την παρατηρητικότητα και τις συνειρμικές δεξιότητες.
- Αναγνώριση γραμμάτων: ανάπτυξη της ικανότητας των παιδιών να συσχετίζουν οικεία αντικείμενα με γράμματα του αλφαβήτου και να αναγνωρίζουν γράμματα.
- Ανάπτυξη της οπτικής μνήμης: ανάπτυξη της οπτικής μνήμης με την επανειλημμένη αναγνώριση σχημάτων και τη συσχέτισή τους με γράμματα.
- Φωνητική επίγνωση: να βοηθηθούν τα παιδιά να κατανοήσουν τη συσχέτιση μεταξύ ήχων και γραμμάτων μέσω της χρήσης σχημάτων.
- Διασκέδαση και κίνητρα: διασφαλίστε ότι τα παιδιά συμμετέχουν ενεργά και απολαμβάνουν τη διαδικασία μάθησης.
- Ενθαρρύνετε τη διασκέδαση και την ευχαρίστηση: ενθαρρύνετε την ανάπτυξη της κριτικής σκέψης και των δεξιοτήτων επίλυσης προβλημάτων με την ανάλυση και τον εντοπισμό σχημάτων.

### 3.2 Μέθοδος παιχνιδιού

Μέθοδος παιχνιδιού «Κυνήγι θησαυρού

Γενική περιγραφή.

Το «Κυνήγι Θησαυρού» είναι ένα παιχνίδι που έχει σχεδιαστεί για να βοηθήσει τα παιδιά να αναγνωρίσουν το ελληνικό αλφάβητο συσχετίζοντας τα σχήματα των αντικειμένων με τα αντίστοιχα γράμματα. Το παιχνίδι διαθέτει δύο επίπεδα δυσκολίας - εύκολο και δύσκολο - για να προκαλεί τα παιδιά ανάλογα με τις ικανότητες και τις ανάγκες τους.

#### Εύκολη έκδοση

Στην εύκολη έκδοση, ένα αντικείμενο εμφανίζεται στην οθόνη. Το παιδί πρέπει να παρατηρήσει το σχήμα του αντικειμένου και να επιλέξει το γράμμα από μια λίστα γραμμάτων που περιγράφει καλύτερα το σχήμα του αντικειμένου. Για παράδειγμα, αν το αντικείμενο είναι

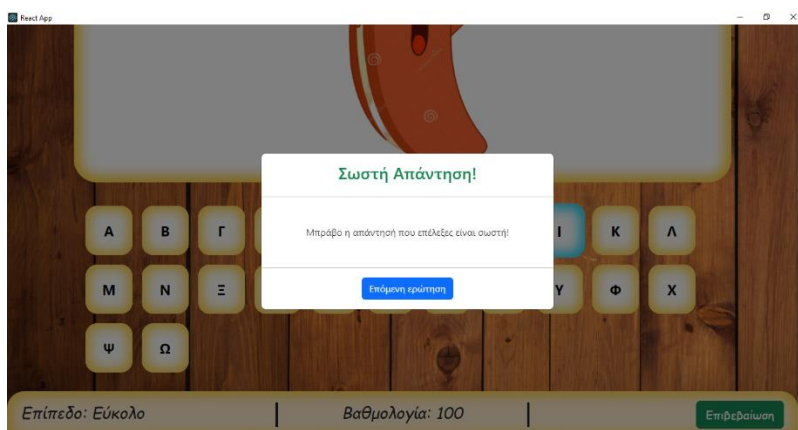
για μια μπάλα (στρογγυλό σχήμα), το σωστό γράμμα είναι το «Ο». Στόχος: να ενισχυθεί η άμεση σύνδεση μεταξύ σχημάτων και γραμμάτων βοηθώντας τα παιδιά να αναγνωρίζουν και να αντιστοιχίζουν τα σχήματα των αντικειμένων και τα αντίστοιχα γράμματα.



εικόνα 3.1.1



Εικόνα 3.1.2



Εικόνα 3.1.3

## Δύσκολη Μέθοδος

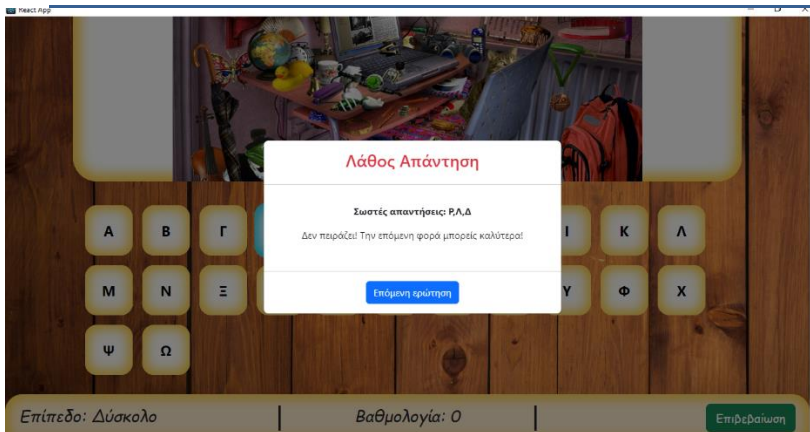
Στη δύσκολη έκδοση αυτού του παιχνιδιού, στην οθόνη εμφανίζεται μια εικόνα με διάφορα αντικείμενα.- Το παιδί πρέπει να παρατηρήσει όλα τα αντικείμενα της εικόνας και να επιλέξει το γράμμα που αντιπροσωπεύει το σχήμα του αντικειμένου. - Για παράδειγμα, αν στην εικόνα υπάρχει μια μπάλα και μια γωνία (τρίγωνο), το παιδί πρέπει να επιλέξει τα γράμματα «Ο» και «Δ».- Στόχος: αύξηση της δυσκολίας και πρόκληση της ικανότητας των παιδιών να αναγνωρίζουν και να επιλέγουν το σωστό γράμμα σε ένα πιο σύνθετο περιβάλλον.



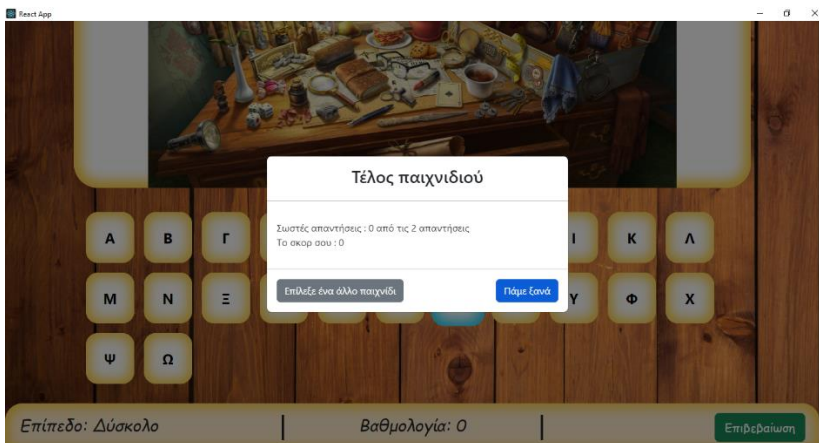
Εικόνα 3.2.1



Εικόνα 3.2.1



Εικόνα 3.2.3



Εικόνα 3.2.4

### 3.3 Εκπαιδευτικά Οφέλη

Το παιχνίδι Κυνήγι Θησαυρού έχει σημαντικά εκπαιδευτικά οφέλη, όπως η βελτίωση των δεξιοτήτων οπτικής παρατήρησης, η δημιουργία συσχετισμών μεταξύ σχημάτων και γραμμάτων και η ενθάρρυνση της αναγνώρισης του λόγου και των γραμμάτων. Μέσα από την εύκολη και τη δύσκολη εκδοχή του, το παιχνίδι βελτιώνει την κριτική σκέψη και τις δεξιότητες επίλυσης προβλημάτων και αυξάνει την αυτοπεποίθηση και το ενδιαφέρον των παιδιών για μάθηση μέσω μιας διασκεδαστικής και διαδραστικής προσέγγισης.

---

## 4. Βρες το Ζώο

### 4.1 Σκοπός και Στόχοι

Στόχοι και σκοποί του παιχνιδιού 'Βρείτε τα ζώα'

Ο κύριος στόχος του παιχνιδιού «Βρες τα ζώα» είναι να ενισχύσει την ικανότητα των παιδιών να συσχετίζουν το ελληνικό αλφάβητο με ονόματα και εικόνες ζώων. Μέσω αυτής της διαδραστικής διαδικασίας, το παιχνίδι αναπτύσσει τις γλωσσικές δεξιότητες και τη φωνητική επίγνωση των παιδιών, ενώ παράλληλα βελτιώνει τις αναγνωστικές τους δεξιότητες.

Στόχοι

- Να συνδέσουν οπτικά τα γράμματα του αλφαβήτου με ονόματα ζώων και να βελτιώσουν τις δεξιότητες μνήμης και κατανόησης.
- Να αυξήσουν τα παιδιά την αναγνώριση του αλφαβήτου συνδέοντάς το με ονόματα ζώων και εικόνες.
- Ανάπτυξη της οπτικής μνήμης: ενίσχυση της οπτικής μνήμης με την επανειλημμένη αναγνώριση των γραμμάτων και τη συσχέτισή τους με ζώα.
- Αναγνώριση του λόγου: κατανόηση των συνδέσεων ήχου-γραμμάτων διευκολύνοντας την αναγνώριση του λόγου.
- Ανάπτυξη λεξιλογίου: ανάπτυξη του λεξιλογίου των παιδιών με την εισαγωγή νέων ονομάτων ζώων και αντίστοιχων γραμματικών τύπων.
- Διασκεδαστική και παρακινητική: παρέχει διασκεδαστικές και παρακινητικές μαθησιακές εμπειρίες για να ενθαρρύνει τα παιδιά να συμμετέχουν ενεργά και να απολαμβάνουν τη διαδικασία της μάθησης.
- Ενίσχυση της κριτικής σκέψης: ενθαρρύνει την ανάπτυξη της κριτικής σκέψης και των δεξιοτήτων αναγνώρισης μέσω της αναγνώρισης και της επιλογής του σωστού ζώου για ένα δεδομένο γράμμα.

Με αυτούς τους στόχους κατά νου, το παιχνίδι «Βρες το ζώο» παρέχει μια διασκεδαστική και εκπαιδευτική εμπειρία που θα συμβάλει σημαντικά στην ανάπτυξη των αναγνωστικών και γλωσσικών δεξιοτήτων των παιδιών.

## 4.2 Μέθοδος Παιχνιδιού

### Μέθοδος παιχνιδιού «Βρείτε τα ζώα»

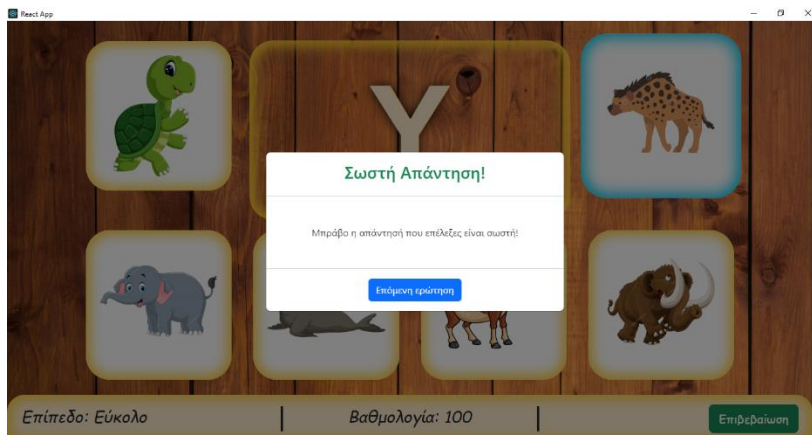
Το παιχνίδι «Βρες τα ζώα» έχει σχεδιαστεί για να βοηθήσει τα παιδιά να συσχετίσουν τα ονόματα και τις εικόνες των ζώων με το ελληνικό αλφάβητο. Το παιχνίδι διατίθεται σε δύο επίπεδα δυσκολίας, εύκολο και δύσκολο, για να προκαλεί τα παιδιά ανάλογα με τις ικανότητες και τις ανάγκες τους.

### Εύκολη έκδοση

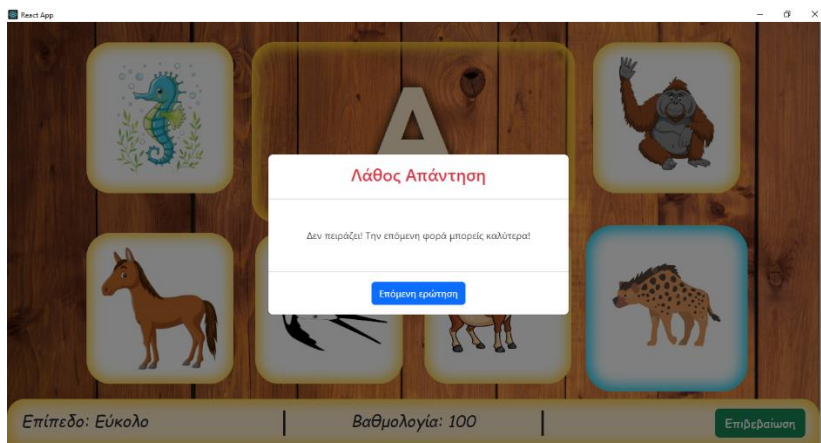
- Μέθοδος. Στην εύκολη έκδοση, στην οθόνη εμφανίζεται κείμενο. Εμφανίζονται έξι εικόνες ζώων, εκ των οποίων μόνο μία αντιστοιχεί σωστά στο γράμμα. Το παιδί πρέπει να επιλέξει την εικόνα του ζώου του οποίου το όνομα αρχίζει με το συγκεκριμένο γράμμα. Για παράδειγμα, αν το γράμμα είναι «Υ», το παιδί πρέπει να επιλέξει την εικόνα μιας Ύαινας. Στόχος: να ενισχυθεί η άμεση σύνδεση μεταξύ γράμματος και ζώου, βάζοντας το παιδί να αναγνωρίσει και να αντιστοιχίσει το όνομα και την εικόνα του ζώου με το γράμμα.



Εικόνα 4.1.1



Εικόνα 4.1.2



4.1.3

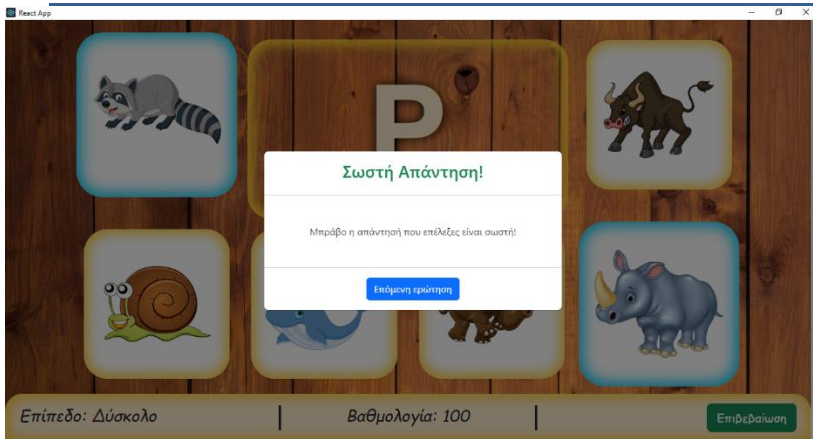
### Δύσκολη Έκδοση

- Μέθοδος. Στη δύσκολη εκδοχή, τα γράμματα εμφανίζονται ξανά στην οθόνη. Εμφανίζονται έξι εικόνες ζώων, αλλά αυτή τη φορά περισσότερες από μία εικόνες αντιστοιχούν στο συγκεκριμένο γράμμα. Το παιδί πρέπει να επιλέξει όλες τις εικόνες ζώων των οποίων το όνομα αρχίζει με το συγκεκριμένο γράμμα. Για παράδειγμα, αν το γράμμα είναι «ρ», το παιδί πρέπει να επιλέξει μια εικόνα ενός ρακούν και ενός ρινόκερου. Στόχος: να αυξηθεί η δυσκολία και η πρόκληση των παιδιών αυξάνοντας την ικανότητά τους να αναγνωρίζουν και να επιλέγουν πολλές σωστές εικόνες σε ένα πιο σύνθετο περιβάλλον.

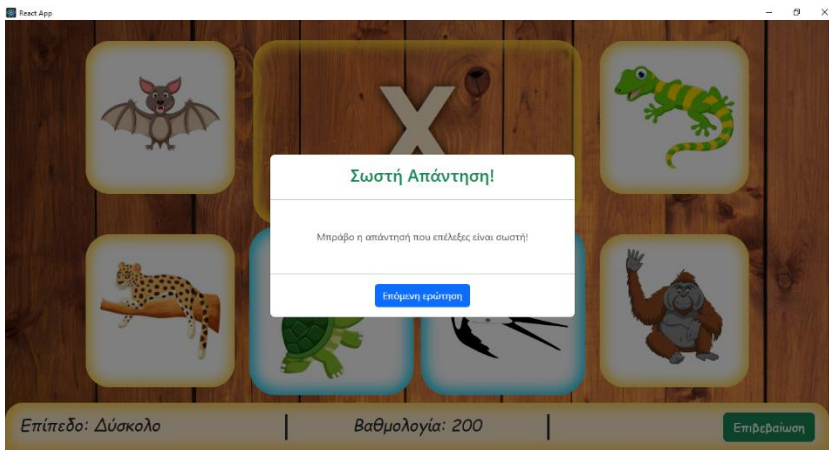


Εικόνα 4.2.1

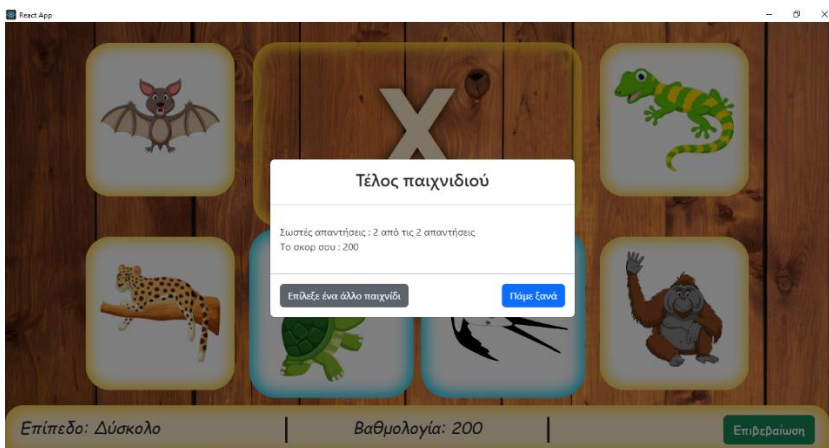




Εικόνα 4.2.2



Εικόνα 4.2.3



Εικόνα 4.2.4

---

### 4.3 Εκπαιδευτικά Οφέλη

Τα εκπαιδευτικά οφέλη της εφαρμογής Μαθαίνω το αλφάβητο είναι πολλαπλά. Στον χρήστη δίνονται τα γράμματα του αλφαβήτου και έξι εικόνες ζώων από τις οποίες πρέπει να βρει το σωστό. Πρώτον, ενισχύεται η αναγνώριση του αλφαβήτου και η συσχέτισή του με λέξεις, γεγονός που συμβάλλει στην ανάπτυξη δεξιοτήτων φωνολογικής επίγνωσης. Δεύτερον, η χρήση εικόνων ζώων κάνει τη διαδικασία μάθησης διασκεδαστική και αυξάνει το ενδιαφέρον και τη δέσμευση των παιδιών. Επιπλέον, τα παιδιά πρέπει να συγκρίνουν και να επιλέξουν τη σωστή εικόνα, γεγονός που προάγει την ανάπτυξη της οπτικής αναγνώρισης και της αντιστοίχισης. Τέλος, η επαναλαμβανόμενη εξάσκηση με διαφορετικά γράμματα και ζώα ενισχύει τη μνήμη και την αναγνώριση μοτίβων και παρέχει μια ολιστική προσέγγιση στην εκμάθηση του αλφαβήτου και τη συνολική γνωστική ανάπτυξη των παιδιών.

---

## Αναλυτική Περιγραφή Υλοποίησης

---

### App CSS

```
.App {
  text-align: center;
}

.App-logo {
  height: 40vmin;
  pointer-events: none;
}

@media (prefers-reduced-motion: no-preference) {
  .App-logo {
    animation: App-logo-spin infinite 20s linear;
  }
}

.App-header {
  background-color: #282c34;
  min-height: 100vh;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  font-size: calc(10px + 2vmin);
  color: white;
}

.App-link {
  color: #61dafb;
}

@keyframes App-logo-spin {
  from {
    transform: rotate(0deg);
  }
  to {
    transform: rotate(360deg);
  }
}
```

---

## APP JAVASCRIPT

```
import React from "react";
import GameProvider from "../providers/game";
import Main from "../pages/main";
import "../assets/style/theme.scss";

const App = () => {
  return (
    <GameProvider>
      <Main />
    </GameProvider>
  );
}

export default App;
```

## APPTTEST JAVASCRIPT

```
import { render, screen } from '@testing-library/react';
import App from './App';

test('renders learn react link', () => {
  render(<App />);
  const linkElement = screen.getByText(/learn react/i);
  expect(linkElement).toBeInTheDocument();
});
```

## INDEX CSS

```
body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto',
'Oxygen',
  'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
  sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}
```

```
code {
  font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
    monospace;
}
```

## INDEX JAVASCRIPT

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import reportWebVitals from './reportWebVitals';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <App />
);

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();
```

## APP REPORT WEB VITALS JAVASCRIPT

```
const reportWebVitals = onPerfEntry => {
  if (onPerfEntry && onPerfEntry instanceof Function) {
    import('web-vitals').then(({ getCLS, getFID, getFCP, getLCP, getTTFB }) =>
    {
      getCLS(onPerfEntry);
      getFID(onPerfEntry);
      getFCP(onPerfEntry);
      getLCP(onPerfEntry);
      getTTFB(onPerfEntry);
    });
  }
};

export default reportWebVitals;
```

## APP SETUP SETS

---

```
// jest-dom adds custom jest matchers for asserting on DOM nodes.
// allows you to do things like:
// expect(element).toHaveTextContent(/react/i)
// learn more: https://github.com/testing-library/jest-dom
import '@testing-library/jest-dom';
```

## GAME

### INDEX

```
import React, { useMemo } from "react";
import GameAreaHeader from "../../components/header/GameAreaHeader";
import GameAreaFooter from "../../components/footer/GameAreaFooter";
import SubmitAnswerModal from "../../components/modal/SubmitAnswerModal";
import NotFound from "../notFound";
import FindImageGame from "./FindImageGame";
import FindLetterGame from "./FindLetterGame";
import TreasureHuntGame from "./TreasureHuntGame";
import AnimalAlphabetGame from "./AnimalAlphabetGame";
import { useGameState } from "../../providers/game";
import { gameTypeEnum } from "../../constants/game";
import FinishGameModal from "../../components/modal/FinishGameModal";

const Game = () => {
  const gameState = useGameState();
  const isGameInvalid = useMemo(() =>
!Object.values(gameTypeEnum).includes(gameState.type), [gameState.type])

  if (isGameInvalid) {
    return <NotFound />
  }

  return (
    <div className="main">
      <div className="game-area-wrapper">
        <GameAreaHeader />

        <div className="main-content py-4 px-3">
          <div className="row g-0 justify-content-center w-100">
            <div className="col-12">
              <div className="game-round-requirements">
```

```

        {gameState.type === gameTypeEnum.TREASURE_HUNT
          ? `Επίλεξε
${gameState.currentQuestionCorrectAnswers.length}
${(gameState.currentQuestionCorrectAnswers.length > 1) ? 'σωστά γράμματα' :
'σωστό γράμμα'}`
          : `Επίλεξε
${gameState.currentQuestionCorrectAnswers.length}
${(gameState.currentQuestionCorrectAnswers.length > 1) ? 'σωστές εικόνες' :
'σωστή εικόνα'}`
        }
      </div>
    </div>

    <div className="col-12 col-xl-10 col-3xl-8 col-4xl-6">
      {gameState.type === gameTypeEnum.FIND_IMAGE &&
<FindImageGame />}

      {gameState.type === gameTypeEnum.FIND_LETTER &&
<FindLetterGame />}

      {gameState.type === gameTypeEnum.TREASURE_HUNT &&
<TreasureHuntGame />}

      {gameState.type === gameTypeEnum.ANIMAL_ALPHABET
&& <AnimalAlphabetGame />}
    </div>
  </div>
</div>

  <GameAreaFooter />

  <SubmitAnswerModal />

  <FinishGameModal />
</div>
</div>
)
};

export default Game;

```

## FIND LETTER

```

import React from "react";
import FindImageGame from "./FindImageGame";

```

---

```

const FindLetterGame = () => {

  return (
    <FindImageGame />
  )
};

export default FindLetterGame;

```

## FIND IMAGE

```

import React, { useMemo } from "react";
import TileItem from "../../components/item/TileItem";
import { useGameAction, useGameState } from "../../providers/game";

const imgSrcPathPrefix = "alphabetImages/common"

const FindImageGame = () => {
  const { handleAnswerChange } = useGameAction()
  const gameState = useGameState();
  const [firstOption, secondOption, ...restOptions] = useMemo(() =>
gameState.currentQuestionOptions, [gameState.currentQuestionOptions])

  // console.log({ gameState })

  return (
    <div className="row">
      <div className="col-12 d-md-none">
        <div className="letter-main-question text-center fw-bold">
          {gameState.currentQuestion}
        </div>
      </div>

      <div className="col-12 col-sm-6 col-md-3 mt-3 mt-md-0">
        <TileItem imgSrcPathPrefix={imgSrcPathPrefix}
id={firstOption.id} imgSrc={firstOption.imgSrc} onClick={handleAnswerChange}
/>
      </div>

      <div className="col-md-6 d-none d-md-block">

```



```

        <div className="letter-main-question text-center fw-bold">
          {gameState.currentQuestion}
        </div>
      </div>

      <div className="col-12 col-sm-6 col-md-3 mt-3 mt-md-0">
        <TileItem imgSrcPathPrefix={imgSrcPathPrefix}
id={secondOption.id} imgSrc={secondOption.imgSrc} onClick={handleAnswerChange}
/>
      </div>

      {restOptions.map(({ id, imgSrc }) => (
        <div key={id} className="col-12 col-sm-6 col-md-3 mt-3">
          <TileItem imgSrcPathPrefix={imgSrcPathPrefix} id={id}
imgSrc={imgSrc} onClick={handleAnswerChange} />
        </div>
      ))}
    </div>
  )
};

export default FindImageGame;

```

## TREASURE HUNT

```

import React from "react";
import classNames from "classnames";
import { useGameAction, useGameState } from "../../providers/game";

const imgSrcPathPrefix = "alphabetImages"

const TreasureHuntGame = () => {
  const { handleAnswerChange } = useGameAction()
  const gameState = useGameState();

  const isOptionSelected = (option) => {
    return gameState.currentQuestionUserAnswers.includes(option)
  }

  console.log({ gameState })

  return (
    <div className="treasure-hunter-game-container">
      <div className="question-container row justify-content-center">
        <div className="col-auto">

```

---

```

        <img src={imgSrcPathPrefix + "/" +
gameState.currentQuestion} alt="not available" />
      </div>
    </div>

    <div className="row justify-content-center mt-3">
      <div className="col-auto">
        <div className="options-container row g-0 mt-3">
          {gameState.currentQuestionOptions.map(o => (
            <div key={o} className="col-auto m-2">
              <button
                className={classnames("letter-option", {
"selected": isOptionSelected(o) })}
                onClick={() => handleAnswerChange({
option: o })}
              >
                {o}
              </button>
            </div>
          ))}
        </div>
      </div>
    </div>
  </div>
)
};

export default TreasureHuntGame;

```

## ANIMAL ALPHABET GAME

```

import React, { useMemo } from "react";
import TileItem from "../../components/item/TileItem";
import { useGameAction, useGameState } from "../../providers/game";

const imgSrcPathPrefix = "alphabetImages/animals"

const AnimalAlphabetGame = () => {
  const { handleAnswerChange } = useGameAction()
  const gameState = useGameState();

```

```

    const [firstOption, secondOption, ...restOptions] = useMemo(() =>
gameState.currentQuestionOptions, [gameState.currentQuestionOptions])

    console.log({ gameState })

    return (
      <div className="row">
        <div className="col-12 d-md-none">
          <div className="letter-main-question text-center fw-bold">
            {gameState.currentQuestion}
          </div>
        </div>

        <div className="col-12 col-sm-6 col-md-3 mt-3 mt-md-0">
          <TileItem imgSrcPathPrefix={imgSrcPathPrefix}
id={firstOption.id} imgSrc={firstOption.imgSrc} onClick={handleAnswerChange}
/>
        </div>

        <div className="col-md-6 d-none d-md-block">
          <div className="letter-main-question text-center fw-bold">
            {gameState.currentQuestion}
          </div>
        </div>

        <div className="col-12 col-sm-6 col-md-3 mt-3 mt-md-0">
          <TileItem imgSrcPathPrefix={imgSrcPathPrefix}
id={secondOption.id} imgSrc={secondOption.imgSrc} onClick={handleAnswerChange}
/>
        </div>

        {restOptions.map(({ id, imgSrc }) => (
          <div key={id} className="col-12 col-sm-6 col-md-3 mt-3">
            <TileItem imgSrcPathPrefix={imgSrcPathPrefix} id={id}
imgSrc={imgSrc} onClick={handleAnswerChange} />
          </div>
        ))}
      </div>
    )
  };

export default AnimalAlphabetGame;

```

---

## INDEX NOT FOUND

```
import React from "react";
import { useGameAction } from "../../providers/game";
import logo from '../../assets/images/logo.svg';
import './assets/style/notFound.scss'

const NotFound = () => {
  const { exitGame } = useGameAction()

  return (
    <div className="not-found">
      <header className="not-found-header">
        <img src={logo} className="not-found-logo" alt="logo" />

        <p>
          Game or Page Not Found
        </p>

        <button className="btn btn-primary btn-lg" onClick={exitGame}>
          Back
        </button>
      </header>
    </div>
  );
};

export default NotFound;
```

## GAME CARD

```
import React, { useState } from "react";
import classNames from "classnames"
import { useGameAction } from "../../providers/game";
import { gameLevelEnum, gameLevelOptions } from "../../constants/game";

const GameCard = ({ type, name, description, cssClasses }) => {
  const { initializeGame } = useGameAction();
  const [level, setLevel] = useState(gameLevelEnum.EASY);

  const handleLevelChange = (e) => {
    setLevel(e.target.value);
  };
};
```

```

}

const handleGameStart = () => {
  initializeGame({ type, level });
}

return (
  <div className="game-card">
    <div className={classnames("card game-card-inner", cssClasses)}>
      <div className="game-card-front">
        {/*  */}
        <div className="card-img-overlay">
          <h1 className="card-title game-card-title">{name}</h1>
          <p className="card-text mt-5 game-card-
description">{description}</p>
        </div>
      </div>

      <div className="game-card-back p-3">
        <h1 className="card-title game-card-title">{name}</h1>

        <h5 className="mt-5 game-card-back-level-title">Επιλέξτε
Επίπεδο :</h5>

        {gameLevelOptions.map(o => (
          <div key={o.id} className="form-check form-check-
inline level-check-style">
            <input
              id={`_${o.elementIdPrefix}_${type}`}
              value={o.id}
              checked={o.id === level}
              className="form-check-input"
              type="radio"
              onChange={handleLevelChange}
            />

            <label htmlFor={`_${o.elementIdPrefix}_${type}`}
className="form-check-label" >
              {o.text}
            </label>
          </div>
        ))}

        <div className="py-5 px-3">
          <button className="btn btn-light btn-lg w-100 game-
start-button" onClick={handleGameStart}>Πάμε!</button>

```

```

        </div>
      </div>
    </div>
  </div>
);
};

export default GameCard;

import React from "react";
import LobbyAreaHeader from "../../components/header/LobbyAreaHeader";
import GameCard from "./GameCard";
import { gameOptions } from "../../constants/game";

const Welcome = () => {
  return (
    <div className="main">
      <div className="lobby-area-wrapper">

        <LobbyAreaHeader />

        <div className="main-content py-4">
          <div className="row g-0 justify-content-center w-100">
            <div className="col-12 col-xl-10 col-3xl-8 col-4xl-6">
              <div className="row g-0 pe-0 pe-md-3 pe-md-0">
                {gameOptions.map(g =>
                  <div key={g.type} className="col-12 col-
md-6 mt-3 ps-3 pe-3 pe-md-0">
                    <GameCard type={g.type} name={g.name}
description={g.description} cssClasses={g.cssClasses} />
                  </div>
                )}
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  );
};

```

```
export default Welcome;
```

## PRIVIDER INDEX

```
import React, { useCallback, useContext, useReducer } from "react";
import GameStateContext, { GameActionContext } from "./context";
import gameReducer, { gameActionTypes, gameInitialState } from "./reducer";

const GameProvider = ({ children }) => {
  const [state, dispatch] = useReducer(gameReducer, gameInitialState);

  const initializeGame = useCallback(({ type, level }) => {
    dispatch({ actionType: gameActionTypes.INITIALIZE_GAME, type, level })
  }, []);

  const handleAnswerChange = useCallback(({ option }) => {
    dispatch({ actionType: gameActionTypes.ANSWER_CHANGE, option })
  }, []);

  const submitAnswer = useCallback(() => {
    dispatch({ actionType: gameActionTypes.SUBMIT_ANSWER })
  }, []);

  const nextQuestion = useCallback(() => {
    dispatch({ actionType: gameActionTypes.NEXT_QUESTION })
  }, []);

  const restartQuestion = useCallback(() => {
    dispatch({ actionType: gameActionTypes.RESTART_QUESTION })
  }, []);

  const finishGame = useCallback(() => {
    dispatch({ actionType: gameActionTypes.FINISH_GAME })
  }, []);

  const restartGame = useCallback(() => {
    dispatch({ actionType: gameActionTypes.RESTART_GAME })
  }, []);

  const exitGame = useCallback(() => {
    dispatch({ actionType: gameActionTypes.EXIT_GAME })
  }, []);

  return (
```

---

```

        <GameActionContext.Provider value={{ initializeGame,
handleAnswerChange, submitAnswer, restartQuestion, nextQuestion, finishGame,
restartGame, exitGame }}>
            <GameStateContext.Provider value={state}>
                {children}
            </GameStateContext.Provider>
        </GameActionContext.Provider>
    );
};

export const useGameAction = () => useContext(GameActionContext);

export const useGameState = () => useContext(GameStateContext);

export default GameProvider;

```

## PROVIDER REDUCER

```

import { gameGenerateNextRound } from "../../helpers/game";
import { gameLevelOptions, gameOptions } from "../../constants/game";

export const gameInitialState = {
    started: false,
    type: null,
    level: null,
    levelText: '',
    name: "",
    description: "",

    currentQuestionIndex: 1,
    currentQuestion: null,
    currentQuestionCorrectAnswers: [],
    currentQuestionOptions: [],
    currentQuestionUserAnswers: [],
    isCurrentUserAnswerCorrect: false,

    // previousQuestionHistory: [],

    totalQuestions: 2,
    totalCorrectAnswers: 0,
    score: 0,

```



```

    isSubmitAnswerModalOpen: false,
    isFinishGameModalOpen: false
  }

export const actionTypes = {
  INITIALIZE_GAME: "initializeGame",
  ANSWER_CHANGE: "answerChange",
  SUBMIT_ANSWER: "submitAnswer",
  RESTART_QUESTION: "restartQuestion",
  NEXT_QUESTION: "nextQuestion",
  FINISH_GAME: "finishGame",
  RESTART_GAME: "restartGame",
  EXIT_GAME: "exitGame"
}

const gameReducer = (state, action) => {
  switch (action.actionType) {
    case actionTypes.INITIALIZE_GAME:
      const selectedGame = gameOptions.find(({ type }) => type ===
action.type);

      const initialGameConfig = gameGenerateNextRound(action.type,
action.level);

      return {
        ...gameInitialState,
        started: true,
        type: action.type,
        level: action.level,
        levelText: gameLevelOptions.find(({ id }) => id ===
action.level)?.text || '',
        name: selectedGame?.name || "N/A",
        description: selectedGame?.description || "N/A",
        currentQuestion: initialGameConfig.question,
        currentQuestionCorrectAnswers:
initialGameConfig.correctAnswers,
        currentQuestionOptions: initialGameConfig.options
      };
    case actionTypes.ANSWER_CHANGE:
      let currentQuestionUserAnswers =
[...state.currentQuestionUserAnswers]

      // if (state.currentQuestionCorrectAnswers.length > 1) {
      const findOptionIndex =
currentQuestionUserAnswers.findIndex(id => id === action.option);

```

---

```

        if (findOptionIndex > -1) {
            currentQuestionUserAnswers.splice(findOptionIndex, 1);
        } else {
            currentQuestionUserAnswers.push(action.option);
        }
        // console.log(state)
// } else {
//     currentQuestionUserAnswers = [action.option];
// }

    return {
        ...state,
        currentQuestionUserAnswers
    };
    case gameActionTypes.SUBMIT_ANSWER:
        const isCurrentUserAnswerCorrect =
(state.currentQuestionCorrectAnswers.length ===
state.currentQuestionUserAnswers.length) &&
        (state.currentQuestionCorrectAnswers.every(x =>
state.currentQuestionUserAnswers.includes(x)))

        let correctAnswersSum = 0

        if( !isCurrentUserAnswerCorrect &&
state.currentQuestionCorrectAnswers.length > 1
            && state.currentQuestionCorrectAnswers.length ===
state.currentQuestionUserAnswers.length
            && state.type !== 'treasure_hunt' ){
            state.currentQuestionUserAnswers.forEach(item => {
                if (state.currentQuestionCorrectAnswers.includes(item)){
                    correctAnswersSum++
                }
            })
        }
        // console.log(state)

    return {
        ...state,
        isSubmitAnswerModalOpen: true,
        isCurrentUserAnswerCorrect,
        correctAnswersSum,

```

```

        score: isCurrentUserAnswerCorrect ? state.score + 100 :
state.score,
        totalCorrectAnswers: isCurrentUserAnswerCorrect ?
state.totalCorrectAnswers + 1 : state.totalCorrectAnswers
    });
    case gameActionTypes.NEXT_QUESTION:
        const nextQuestionGameConfig = gameGenerateNextRound(state.type,
state.level)

        return {
            ...state,
            isSubmitAnswerModalOpen: false,
            currentQuestionIndex: state.currentQuestionIndex + 1,
            currentQuestion: nextQuestionGameConfig.question,
            currentQuestionCorrectAnswers:
nextQuestionGameConfig.correctAnswers,
            currentQuestionOptions: nextQuestionGameConfig.options,
            currentQuestionUserAnswers: [],
            isCurrentUserAnswerCorrect: false
        };
    case gameActionTypes.RESTART_QUESTION:

        return {
            ...state,
            isSubmitAnswerModalOpen: false,
            currentQuestionIndex: state.currentQuestionIndex,
            currentQuestion: state.currentQuestion,
            currentQuestionCorrectAnswers:
state.currentQuestionCorrectAnswers,
            currentQuestionOptions: state.currentQuestionOptions,
            currentQuestionUserAnswers: [],
            isCurrentUserAnswerCorrect: false
        };
    case gameActionTypes.FINISH_GAME:
        return {
            ...state,
            isSubmitAnswerModalOpen: false,
            isFinishGameModalOpen: true
        };
    case gameActionTypes.RESTART_GAME:
        const restartedGameConfig = gameGenerateNextRound(state.type,
state.level);
        return {
            ...gameInitialState,
            started: true,
            type: state.type,
            level: state.level,

```

---

```

        levelText: state.levelText,
        name: state.name,
        description: state.description,
        currentQuestion: restartedGameConfig.question,
        currentQuestionCorrectAnswers:
restartedGameConfig.correctAnswers,
        currentQuestionOptions: restartedGameConfig.options
    };

    case gameActionTypes.EXIT_GAME:
        return { ...gameInitialState };
    default:
        throw new Error("Action not supported.");
    }
}

export default gameReducer

```

## PROVIDER CONTEXT

```

import { createContext } from "react";

export const GameActionContext = createContext(null);

export default createContext();

```

“Τα παραπάνω αποσπάσματα κώδικα περιέχουν μέρος του κώδικα της διατριβής μου. Η πλήρης έκδοση και όλα τα αρχεία της εφαρμογής βρίσκονται στο συμπιεσμένο φάκελο «ΓΑΒΡΙΛΗΣ\_Π17193» και έχουν αποσταλεί και βαθμολογηθεί από τον καθηγητή.”

---

## ΣΥΜΠΕΡΑΣΜΑΤΑ

Συμπερασματικά, η εφαρμογή «Μαθαίνω το αλφάβητο» σχεδιάστηκε και υλοποιήθηκε για να προσφέρει μια διασκεδαστική και εκπαιδευτική εμπειρία για παιδιά προσχολικής ηλικίας, χρησιμοποιώντας τεχνολογίες HTML, CSS, JavaScript και React για να καλύψει τις ανάγκες των μικρών παιδιών που μαθαίνουν το ελληνικό αλφάβητο. Δημιουργήθηκε μια διαδραστική πλατφόρμα. Η επιλογή της HTML και της CSS εξασφάλισε μια σταθερή και όμορφη βάση για την εφαρμογή, επιτρέποντας την αποτελεσματική δομή και παρουσίαση του περιεχομένου. Η JavaScript παρείχε την απαραίτητη λειτουργικότητα, επιτρέποντας τη δυναμική αλληλεπίδραση και τη δημιουργία διαφόρων κατηγοριών παιχνιδιών.

Το έργο σχεδιάστηκε επίσης για να έχει μεγάλη επιτυχία. Τέλος, η React επέτρεψε την ανάπτυξη ευέλικτων και κλιμακούμενων εφαρμογών, διευκολύνοντας τη διαχείριση καταστάσεων και την αλληλεπίδραση συστατικών. Με την ανάπτυξη τεσσάρων κατηγοριών παιχνιδιών, τα παιδιά μπορούν να μάθουν γράμματα με διασκεδαστικό και παρακινητικό τρόπο, να αναγνωρίζουν αντικείμενα και ζώα και να βελτιώνουν την αναγνώριση ομιλίας τους. Τα εκπαιδευτικά οφέλη της εφαρμογής περιλαμβάνουν την ενίσχυση της οπτικής μνήμης, της κατανόησης της ανάγνωσης και της κριτικής σκέψης.

Συνολικά, η εργασία αυτή δείχνει πώς η τεχνολογία μπορεί να χρησιμοποιηθεί για τη δημιουργία ισχυρών εκπαιδευτικών εργαλείων και να διευκολύνει τη μάθηση μέσω της διασκέδασης και της διαδραστικότητας.

---

## ΒΙΒΛΙΟΓΡΑΦΙΑ

**HTML.** (2024). Στη Wikipedia, την ελεύθερη εγκυκλοπαίδεια. Ανακτήθηκε από

<https://el.wikipedia.org/wiki/HTML>

**CSS.** (2024). Στη Wikipedia, την ελεύθερη εγκυκλοπαίδεια. Ανακτήθηκε από <https://el.wikipedia.org/wiki/CSS>

**React (JavaScript library).** (2024). Στη Wikipedia, την ελεύθερη εγκυκλοπαίδεια. Ανακτήθηκε από [https://en.wikipedia.org/wiki/React\\_\(JavaScript\\_library\)](https://en.wikipedia.org/wiki/React_(JavaScript_library))

**JavaScript.** (2024). Στη Wikipedia, την ελεύθερη εγκυκλοπαίδεια. Ανακτήθηκε από <https://el.wikipedia.org/wiki/JavaScript>