

Ανάπτυξη Εφαρμογών Κινητών Συσκευών
Εργασία εαρινό 2023



Μέλη ομάδας

Ανδρονιάδης Θεμιστοκλής ΑΕΜ: 3307

Μαρινέλα Κάλθι ΑΕΜ: 3984

Καλλιόπη Μαλλά ΑΕΜ:3979

Περιγραφή σκοπού εφαρμογής και των τυπικών χρηστών της

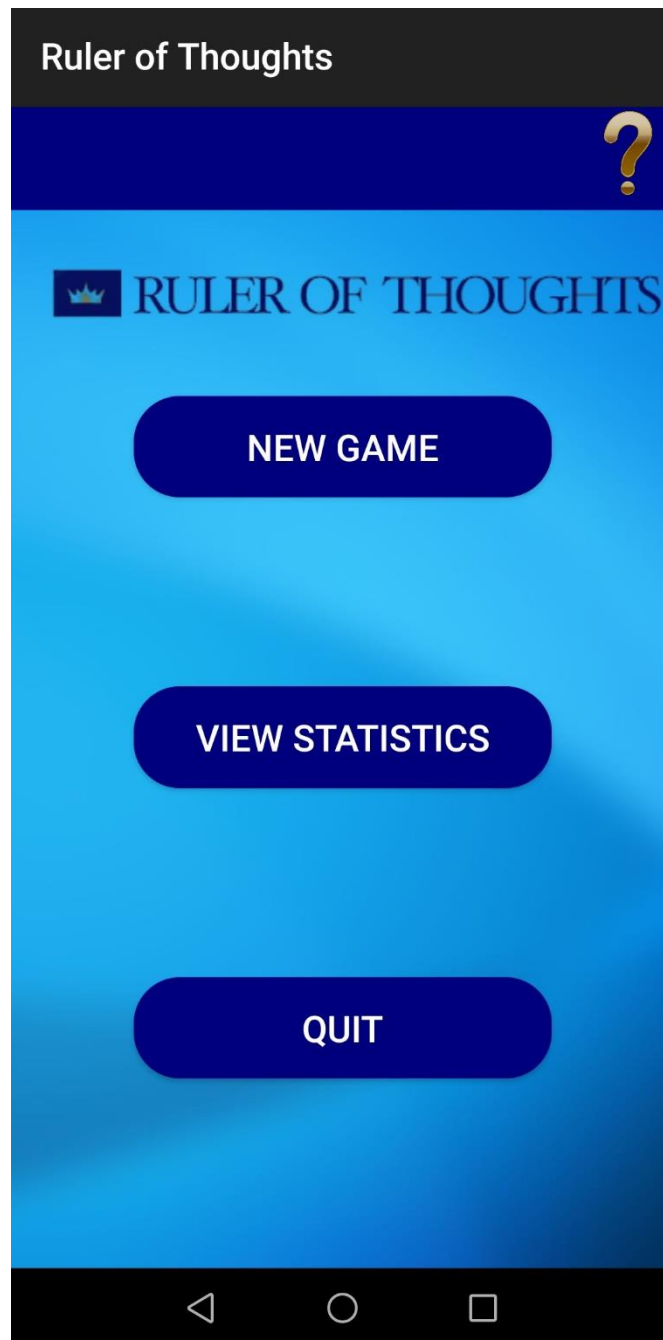
Σκοπός της εφαρμογής είναι η επαφή του χρήστη με το επιτραπέζιο παιχνίδι Master Minds «παιχνίδι με καρφάκια» της δεκαετίας του '70.

Πρόκειται για ένα επιτραπέζιο της λογικής των δύο παικτών. Στην περίπτωση μας, υπάρχει ένας τυπικός χρήστης ο οποίος παίζει το παιχνίδι με τον υπολογιστή. Ο υπολογιστής θα δημιουργήσει ένα μυστικό κωδικό χρωμάτων τον οποίο ο χρήστης θα προσπαθήσει να σπάσει. Ο παίχτης έχει 9 προσπάθειες για να μαντέψει τον κωδικό, και ο υπολογιστής δίνει διάφορες βοήθειες (hints) όπως αναλύεται επακριβώς στα Rules της εφαρμογής.

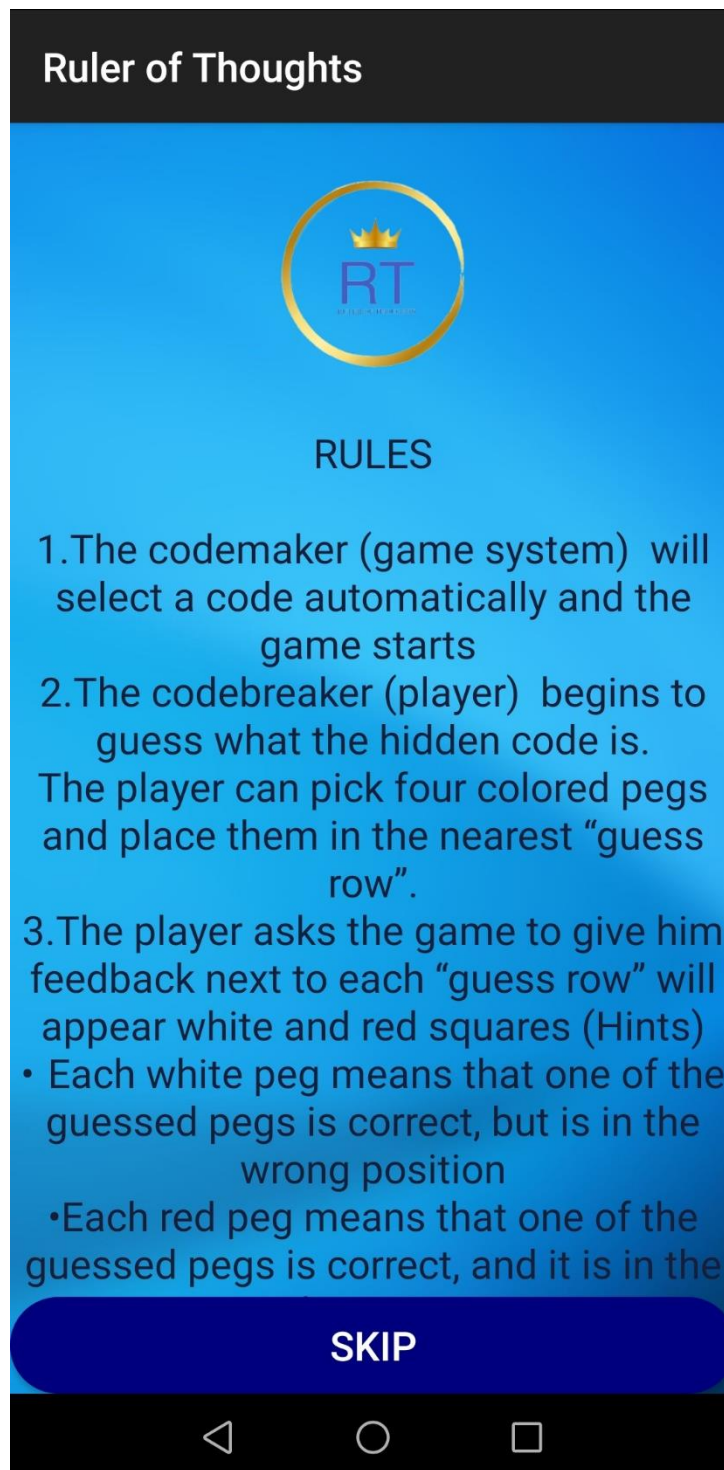


Λειτουργικότητα εφαρμογής

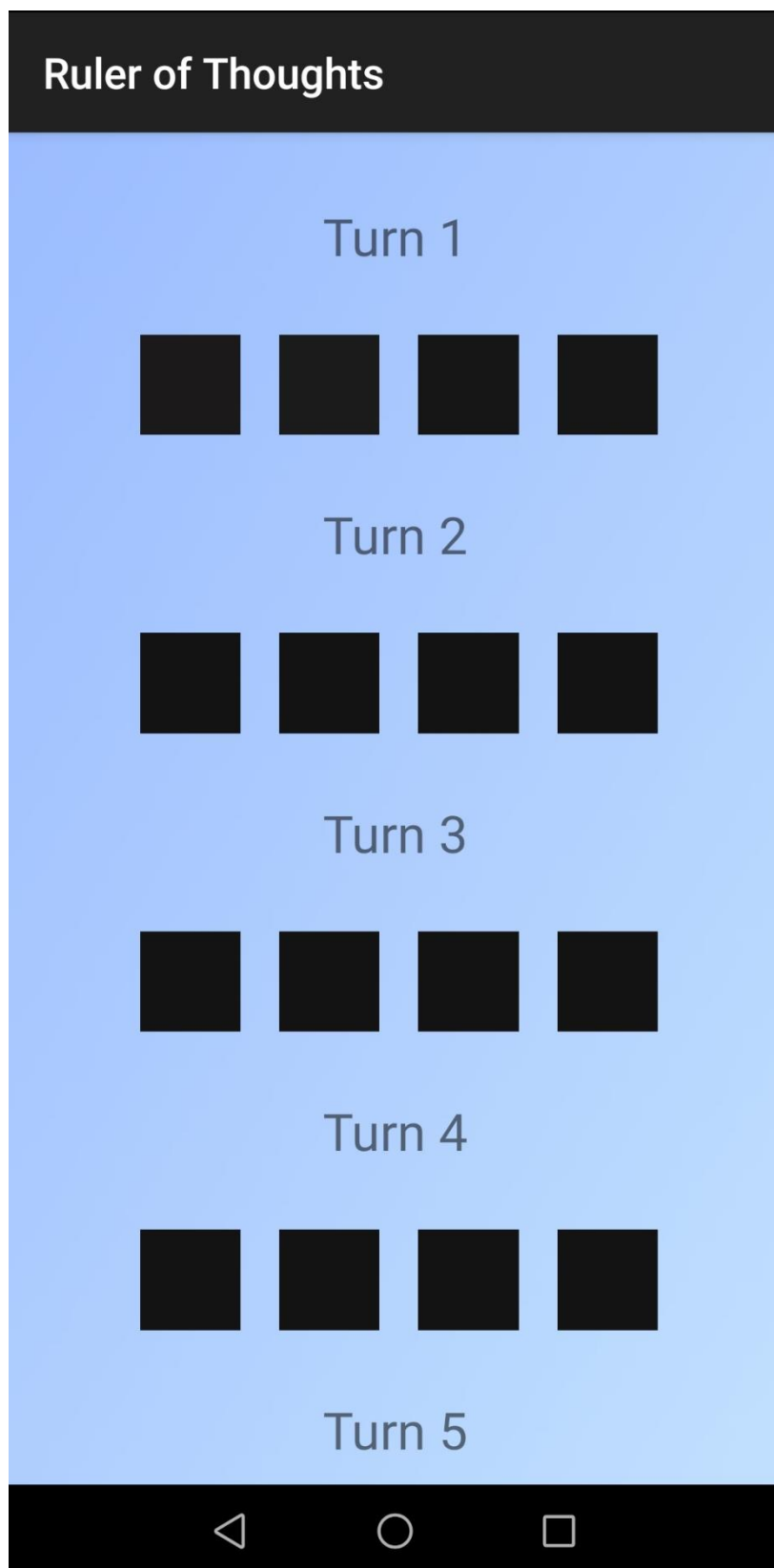
Όταν ο χρήστης ανοίξει την εφαρμογή , εμφανίζεται το μενού επιλογών όπως βλέπουμε παρακάτω. Το button «NEW GAME» ανοίγει μια νέα παρτίδα , το «VIEW STATISTICS» ανοίγει το layout με τα στατιστικά των παικτών , το «QUIT» σε βγάζει από την εφαρμογή.



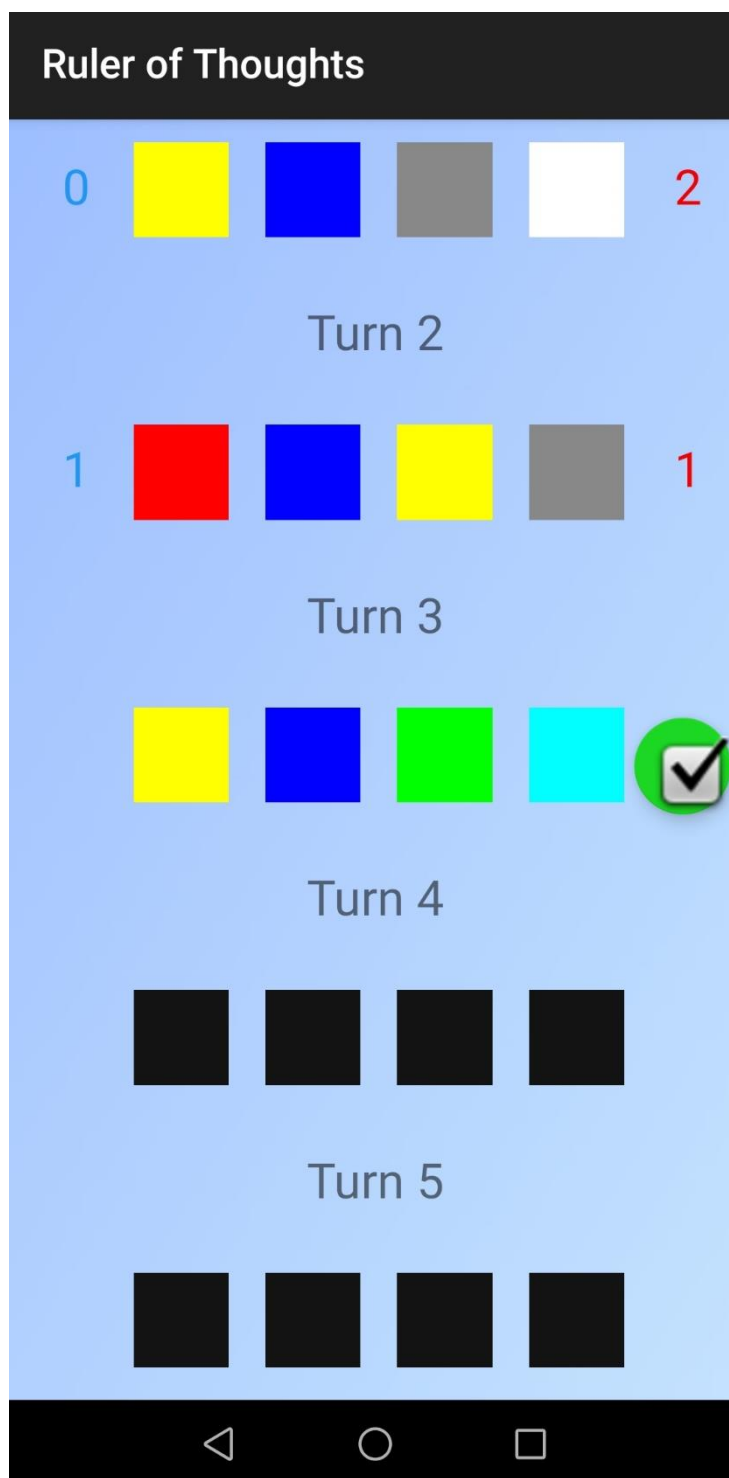
Εάν από το layout του μενού πατήσει το ImageButton «?» εμφανίζεται η σελίδα με τους κανονισμούς του παιχνιδιού.



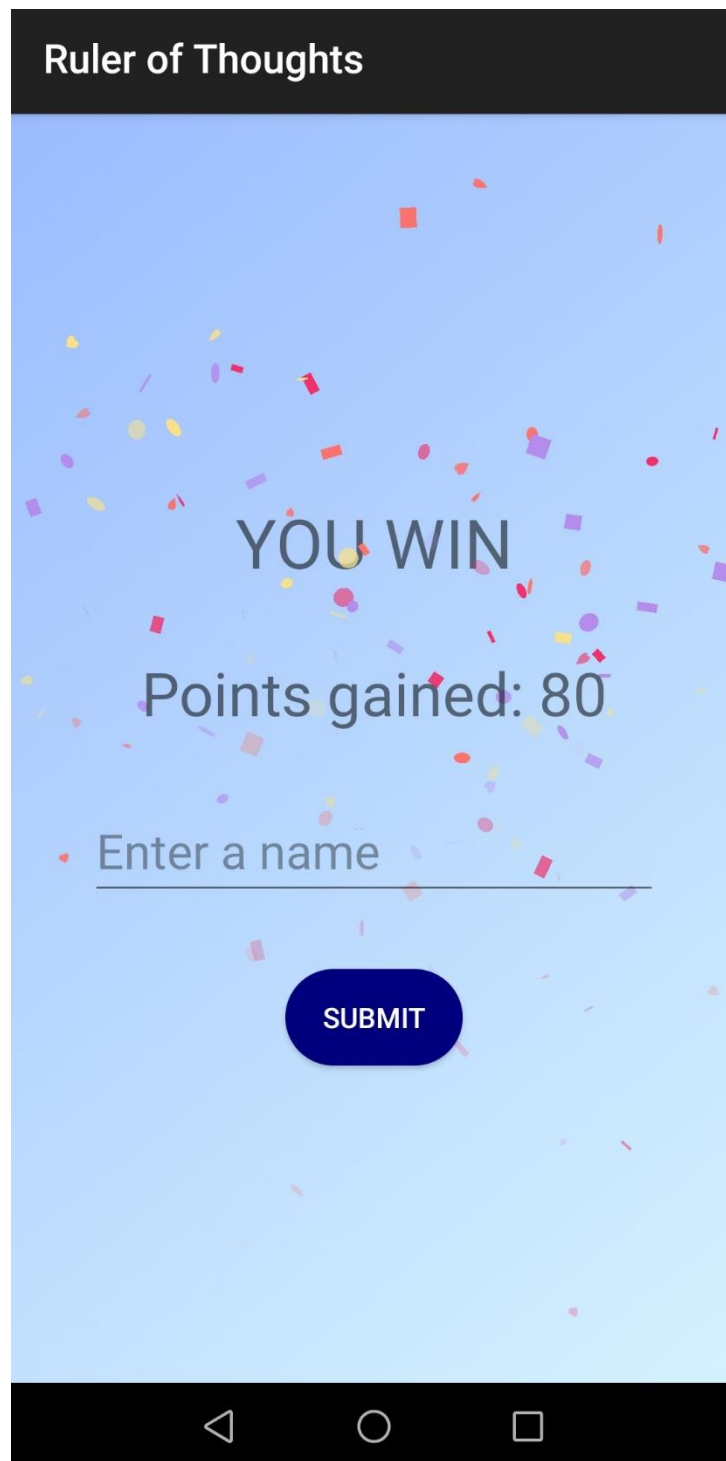
Εάν από το layout του μενού πατήσει την επιλογή «NEW GAME»
εμφανίζεται το παιχνίδι.



Πατώντας κάθε κουτί εμφανίζεται ένας κατάλογος με χρώματα και ο παίχτης μπορεί να επιλέξει το χρώμα που επιθυμεί. Συμπληρώνοντας τον συνδυασμό των τεσσάρων επιλογών εμφανίζεται το χαρακτηριστικό «τικ» που αποθηκεύει τις επιλογές του. Οι δυο ακριανοί αριθμοί είναι τα hints που θα βοηθήσουν τον παίχτη να διαλέξει την στρατηγική του και να καταλήξει στον σωστό τελικό κωδικό χρωμάτων.



Σε περίπτωση νίκης εμφανίζεται αντίστοιχο layout στο οποίο εμφανίζονται οι πόντοι της παρτίδας και ο χρήστης γράφει το όνομα του για να αποθηκευτεί στην βάση πατώντας το «submit». Να επισημανθεί πως οι πόντοι της παρτίδας υπολογίζονται με τον εξής τρόπο: εάν ο κωδικός λυθεί με την πρώτη προσπάθεια ο παίχτης κερδίζει 100 πόντους, και σε κάθε επόμενη η πόντοι μειώνονται κατά 10.



Αντίστοιχα εάν ο παίχτης δεν καταφέρει να τελειώσει την παρτίδα με επιτυχία, δεν βρει δηλαδή τον κωδικό χρωμάτων , εμφανίζεται layout «GAME OVER». Επίσης, προσθέτει το όνομα του για να μπει η προσπάθεια του στην βάση.

GAME OVER

No turns left.

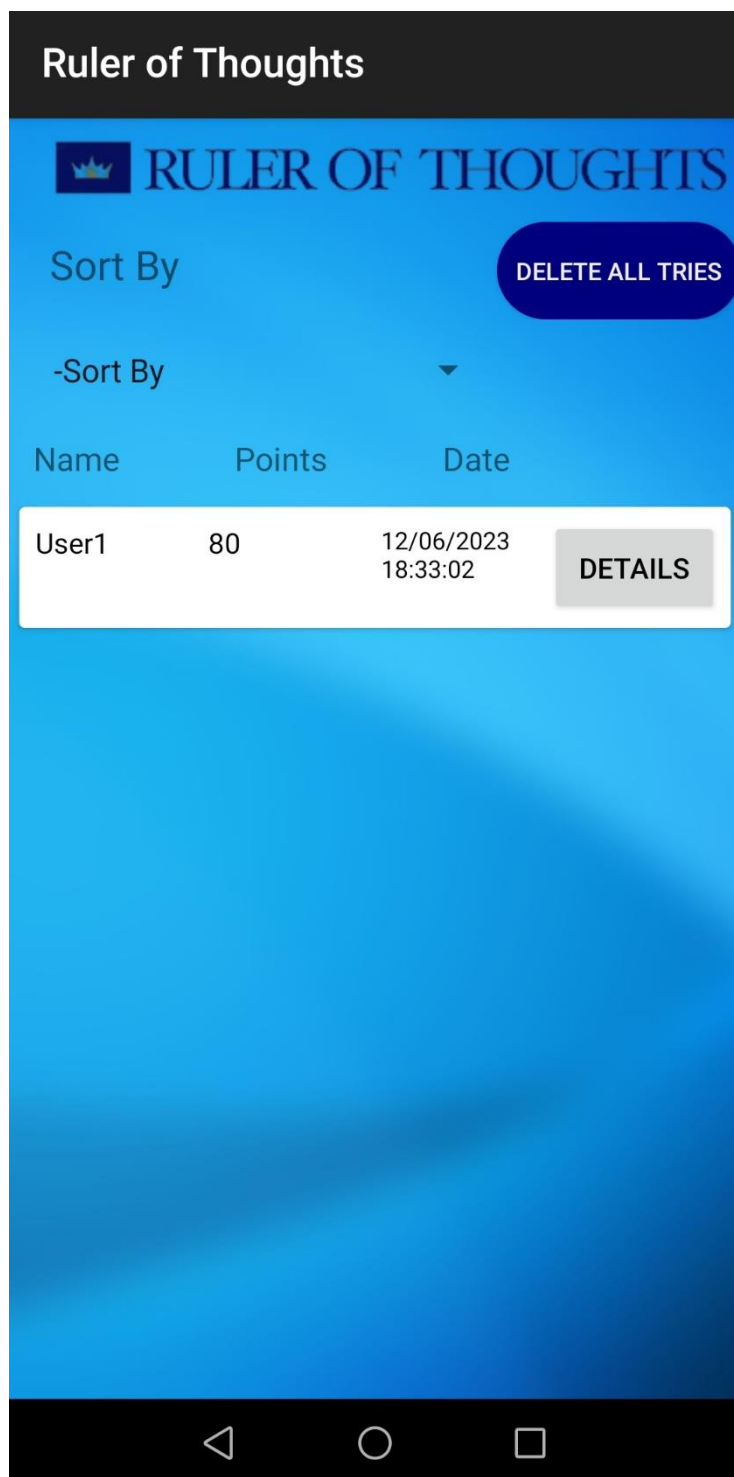
Points gained: 0

Enter a name

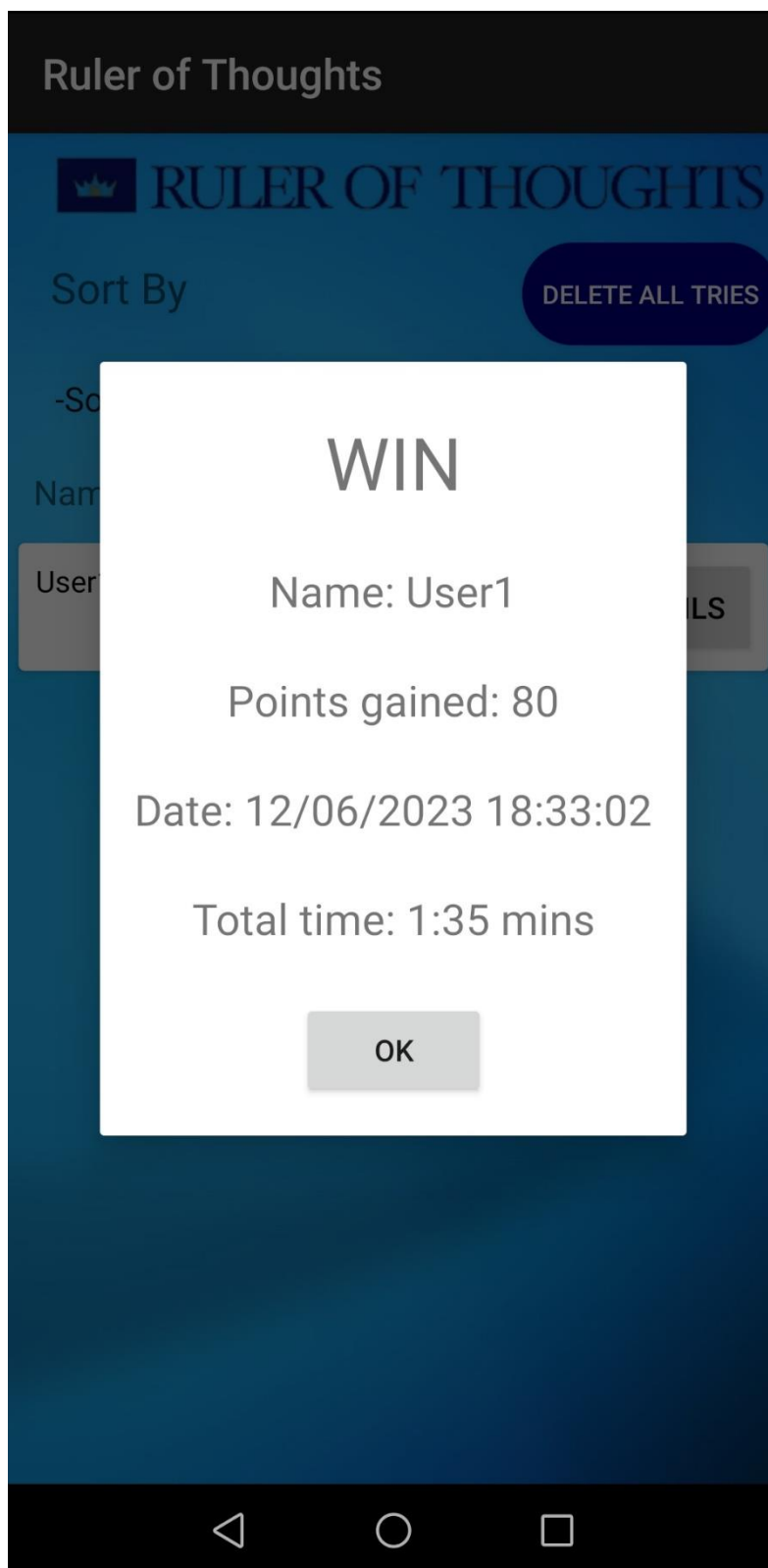
OK

Turn 5

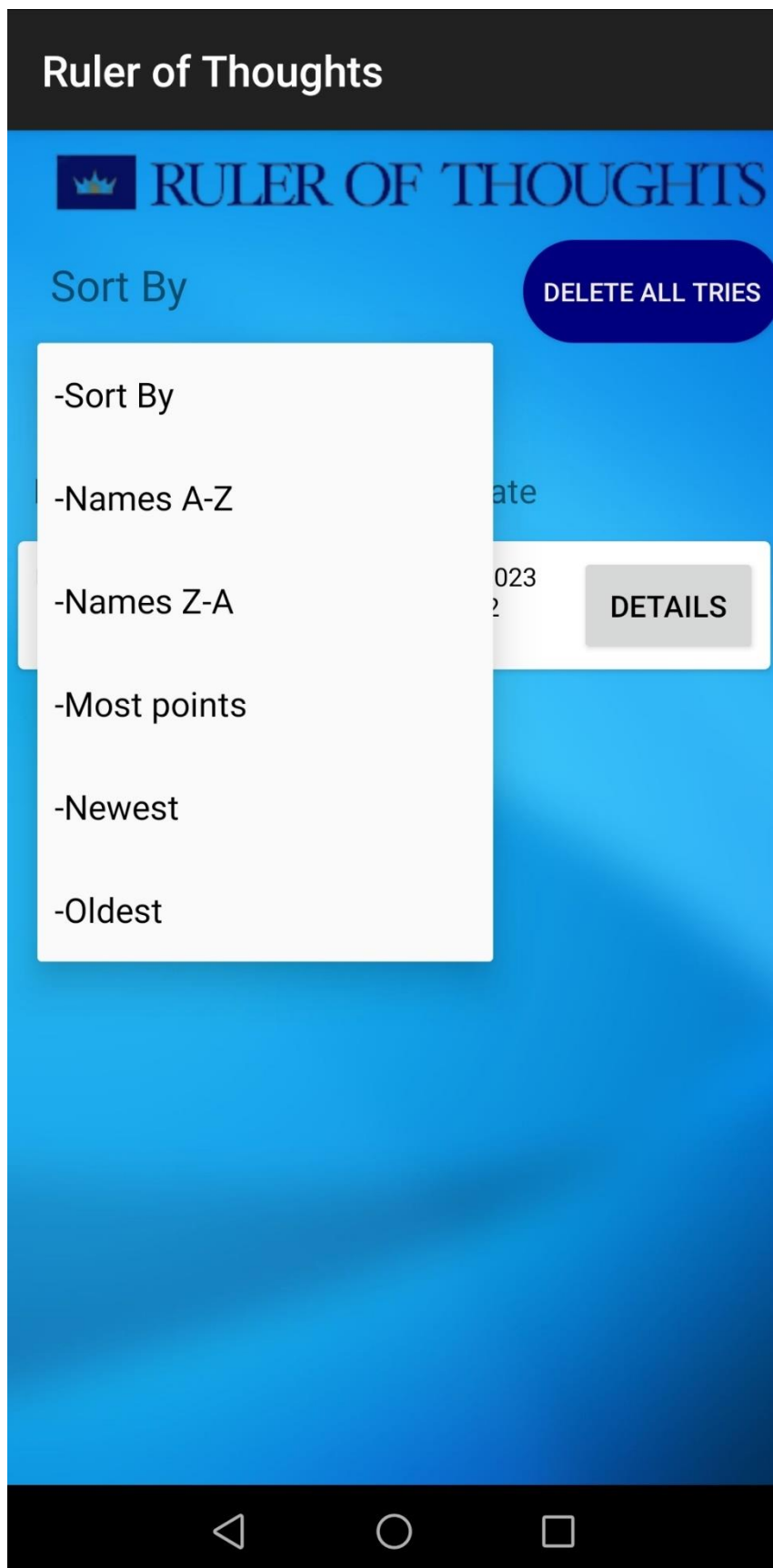
Επιστρέφοντας στο menu, εάν ο χρήστης πατήσει το button «VIEW STATISTICS» οδηγείται στην σελίδα που μπορεί να δει τους πόντους, την ημέρα και την ώρα της παρτίδας καθώς και τις προσπάθειες άλλων παιχτών. Εδώ εισήχθη και η απαιτούμενη βάση, με τον κάθε χρήστη να αποθηκεύεται με βάση το username του.



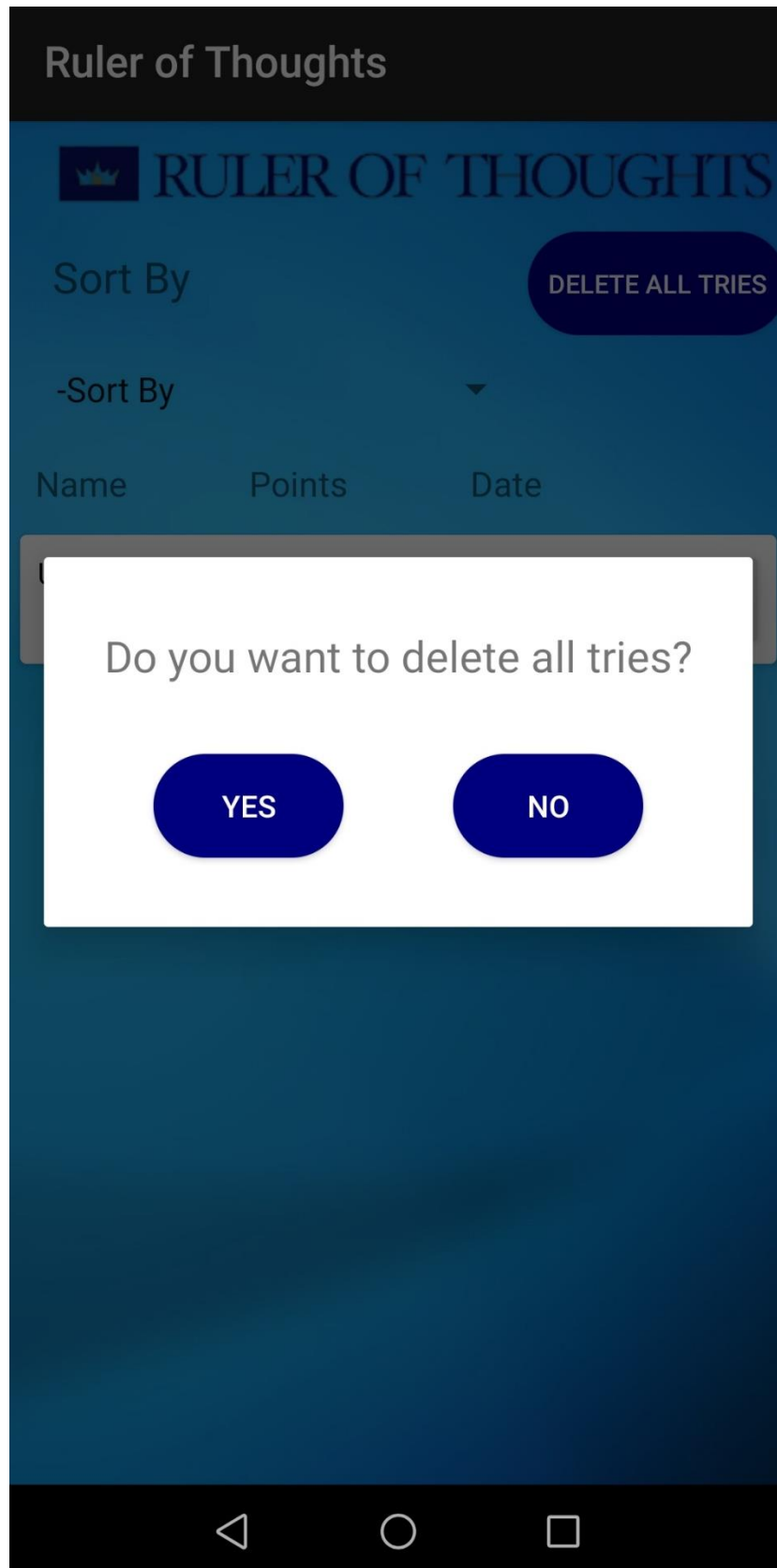
Εάν θέλει περισσότερες πληροφορίες πατώντας τα «DETAILS» μπορεί να
δει εάν κέρδισε ή έχασε ,τους πόντους, την ημερομηνία , την ώρα και
τον χρόνο που διήρκεσε η παρτίδα .




Μπορεί να γίνει και sorting με βάση διάφορες μετρικές.



Σε περίπτωση που θες να διαγράψεις όλες τις προσπάθειες , δίνεται η δυνατότητα μέσω του «DELETE ALL TRIES» και εμφανίζεται ένα παράθυρο στο οποίο επιβεβαιώνεις την επιλογή σου.



Εδώ φαίνονται και οι προσπάθειες περισσότερων από ενός χρήστη.

Ruler of Thoughts			
 RULER OF THOUGHTS			
Sort By			DELETE ALL TRIES
-Newest			
Name	Points	Date	
User2	0	12/06/2023 18:34:30	DETAILS
User1	80	12/06/2023 18:33:02	DETAILS

Περιγραφή κώδικα εφαρμογής

Λογική του κώδικα:

Σημείωση: Στην κλάση `GameActivityWithSpinner`, υπάρχουν κάποια σχόλια στις γραμμές 95-98. Αν αυτές τις γραμμές τις βγάλετε από τα σχόλια στην οθόνη με την παρτίδα της εφαρμογής θα εμφανιστούν 2 `imagebuttons`, ένα στα αριστερά και ένα στα δεξιά. Το κουμπί στα αριστερά με το όνομα `"SHOW ANSWER"`, εμφανίζει ένα `toast message` με τα χρώματα και τις θέσεις του κρυφού κώδικα. Αν βάλετε με αυτή τη σειρά εκείνα τα χρώματα κερδίζετε το παιχνίδι. Το κουμπί στα δεξιά με το όνομα `"FORCE DEFEAT"`, τελειώνει την παρτίδα με ήττα. Είναι σαν να έχετε χάσει, δηλαδή έχουν τελειώσει και οι 9 γύροι. Τα συγκεκριμένα `buttons` δημιουργήθηκαν για τον έλεγχο της εφαρμογής, αλλά μπορούν να χρησιμοποιηθούν σε περίπτωση που δε θέλετε να παίξετε όλους τους γύρους του παιχνιδιού, για να δείτε την λειτουργικότητα της εφαρμογής πιο άμεσα. Αφού έχει ξεκινήσει καινούργια παρτίδα, την λογική την αναλαμβάνει η κλάση `GameActivityWithSpinner`. Δημιουργούμε έναν διδιάστατο πίνακα που θα αποθηκεύει αντικείμενα της κλάσης `Pegs`. Αυτή η κλάση χρησιμοποιείται για να αποθηκεύσει τα `ids` των `imagebuttons` που αναπαριστούν τις επιλογές του `game` για κάθε προσπάθεια. Επίσης η κλάση χρησιμοποιείται για να αποθηκεύσει το χρώμα-επιλογή του παίκτη, έτσι ώστε όταν γίνει ο έλεγχος για το αν είναι σωστή η προσπάθεια, να μπορεί να γίνει η σύγκριση με τα χρώματα του κρυφού κωδικού. Δημιουργούνται επίσης τρεις πίνακες, όπου οι δύο αποθηκεύουν τα `ids` των `textviews` που βρίσκονται στα αριστερά και στα δεξιά των προσπαθειών και αναπαριστούν τους δείκτες σωστής επιλογής χρώματος-σωστής θέσης και σωστής επιλογής χρώματος-λάθος θέσης. Ο τρίτος πίνακας έχει τα `ids` των `floating action buttons` τα οποία είναι κρυφά εξ αρχής και εμφανίζονται όταν η συγκεκριμένη προσπάθεια έχει σε όλες τις θέσεις χρώματα, που δεν είναι το μαύρο χρώμα. Στα `imagebuttons` που αναπαριστούν τις θέσεις του παιχνιδιού, υπάρχει μία μέθοδος όπου μόλις πατηθούν ενεργοποιείται. Πρώτα από όλα, γίνεται έλεγχος για το αν η θέση που επέλεξε ο παίκτης είναι στον τρέχον γύρο. Αφού πάρουμε το `id` του συγκεκριμένου `imagebutton` που επιλέχθηκε μέσω της `getId()` της `View` κλάσης, ελέγχουμε αν βρίσκεται στη γραμμή εκείνη που δείχνει τον τρέχον γύρο μέσω της μεταβλητής `current_turn`, η οποία αυξάνεται στο τέλος του κάθε γύρου, αν δεν έχει βρεθεί ο σωστός κωδικός ή δεν έχουν τελειώσει οι γύροι. Αν η θέση ανήκει σε αυτόν το γύρο, η μέθοδος δημιουργεί ένα `dialog` το οποίο περιέχει ένα `spinner` με επιλογές χρωμάτων. Μόλις πατηθεί το `OK button`, η μέθοδος έχει υπερφορτωμένη την `setOnClickListener` μέθοδο, έτσι ώστε να κάνει έλεγχο σχετικά με το ποιο χρώμα επιλέχθηκε και αντίστοιχα, στον διδιάστατο πίνακα όπου αποθηκεύονται τα αντικείμενα στέλνουμε στην αντίστοιχη θέση που επιλέχθηκε το χρώμα που επέλεξε ο χρήστης για να αποθηκευτεί στο αντικείμενο. Η μέθοδος των `imagebuttons` πριν τελειώσει, καλεί μία άλλη μέθοδο, έτσι ώστε να γίνει ο έλεγχος των επιλογών του παίκτη. Ο πρώτος έλεγχος αναγνωρίζει το αν η συγκεκριμένη προσπάθεια είναι ολοκληρωμένη, δηλαδή δεν υπάρχει το μαύρο χρώμα σε καμία από τις 4 θέσεις.

Αν είναι συμπληρωμένη, η μέθοδος εμφανίζει το floating action button που βρίσκεται δεξιά της συγκεκριμένης προσπάθειας. Σε περίπτωση που ο παίκτης αλλάξει άποψη για κάποια επιλογή και βάλει στη θέση της το μαύρο χρώμα, η μέθοδος κρύβει το floating action button, για να μην επιλεγθεί. Αν το floating action button εμφανιστεί, το οποίο είναι ένα τικ σύμβολο, το οποίο έχει ως σκοπό να ελέγξει την προσπάθεια του παίκτη με αυτή του κρυφού κωδικού. Τα floating action buttons τα οποία βρίσκονται στα δεξιά κάθε προσπάθειας, είναι κρυφά εξαρχής. Μόλις φανερωθεί κάποιο, εκτελώντας τα παραπάνω βήματα, ο παίκτης μπορεί να το πατήσει για να γίνει ο έλεγχος. Υπάρχει μία μέθοδος για κάθε floating action button, η οποία κάνει το εξής: υπάρχουν δύο hash tables, όπου το ένα αποθηκεύει το μοναδικά χρώματα με τις αντίστοιχες εμφανίσεις στο κρυφό κωδικό, έχοντας ως κλειδί τα χρώματα (String) και τιμές τον αριθμό εμφανίσεων (Integer). Το άλλο hash table αποθηκεύει με την ίδια λογική, τις επιλογές του χρήστη για την συγκεκριμένη προσπάθεια. Δημιουργείται ένα αντίγραφο του hash table του κρυφού κωδικού για να παραμείνει αναλλοίωτο το πρωτότυπο, ενώ το hash table της προσπάθειας του παίκτη κάθε φορά διαγράφει τις τιμές, έτσι ώστε να εισαχθούν οι καινούργιες επιλογές του χρήστη στην επόμενη προσπάθεια. Στο σημείο αυτό ακολουθεί μία επαναληπτική μέθοδος, η οποία ελέγχει αν στις αντίστοιχες θέσεις του κρυφού κωδικού και της προσπάθειας του παίκτη, βρίσκονται τα ίδια χρώματα. Αν βρεθούν κάποια υπάρχει μία μεταβλητή, `red_index` η οποία αποθηκεύει τον αριθμό σωστών χρωμάτων-θέσεων, ενώ ταυτόχρονα αλλάζουν οι τιμές στο hash table του χρήστη μειώνοντας τις εμφανίσεις αυτών. Στην συνέχεια κάνουμε μία ακόμη επαναληπτική διαδικασία, όπου βλέπουμε ποια χρώματα βρίσκονται και στα δύο hash tables. Αν υπάρχει κάποιο τότε, στην μεταβλητή `white_index` αποθηκεύουμε τον μικρότερο αριθμό εμφανίσεων, και όχι το μεγαλύτερο γιατί κάποιες εμφανίσεις θα εμφανίζονται διπλές/τριπλές. Αφού τελειώσουν οι συγκρίσεις, γίνεται ο έλεγχος για το αν η μεταβλητή `red_index` έχει την τιμή 4, το οποίο σημαίνει ότι οι απαντήσεις που έδωσε ο χρήστης βρίσκονται στις αντίστοιχες θέσεις με του κρυφού κωδικού. Άρα έχει κερδίσει. Αν κερδίσει πηγαίνει στην κλάση Results, όπου θα ζητηθεί το όνομα για την προσπάθεια. Πριν ξεκινήσει η κλάση, αποθηκεύουμε σε ένα bundle ο χρόνος που έκανε ο χρήστης να ολοκληρωθεί η παρτίδα, ξεκινώντας από την εμφάνιση της οθόνης του παιχνιδιού μέχρι και όταν γίνει ο έλεγχος ότι κέρδισε. Επίσης στέλνονται και οι πόντοι με βάση το ποια προσπάθεια τελείωσε. Αν δεν έχει κερδίσει ο παίκτης, εμφανίζονται στα αριστερά και στα δεξιά οι τιμές των μεταβλητών `white_index` και `red_index` αντίστοιχα, για να βοηθηθεί ο παίκτης στις επόμενες προσπάθειες. Αυξάνεται η τιμή της μεταβλητής που αποθηκεύουν τους τρέχον γύρους (`current_turn`) και γίνεται ο έλεγχος για το αν έχει χάσει ο παίκτης δηλαδή αν οι τρέχον γύροι έχουν ξεπεράσει τους 9. Αν έχει χάσει ο παίκτης εμφανίζεται ένα dialog που δείχνει στον παίκτη ότι έχει χάσει και ζητάει το όνομα του για να αποθηκευτεί η προσπάθεια.