



Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Εξάμηνο 7ο 2021-2022
Τελική εργασία στο μάθημα Τεχνολογία Πολυμέσων

Ονοματεπώνυμο: **Στυλιανός Κανδυλάκης**

Αριθμός Μητρώου: **03117088**

Ημερομηνία: 11/1/2022

1. Σχεδιασμός και υλοποίηση λογικής

Η εργασία μου αποτελείται από 3 ενότητες αρχείων-κλάσεις, στο καθένα αντιστοιχεί υλοποίηση μίας ξεχωριστής λειτουργίας της εφαρμογής.

Έτσι έχω:

- Parsecode.class
- thefinalhangman.class
- tuple.class
- Main.class
- FXMLhangmanController.class
- FXMLhangman.fxml

A. Δημιουργία λεξικού

Η Parsecode περιέχει methods για τη δημιουργία-επεξεργασία των αρχείων με ό,τι αυτό συνεπάγεται(επεξεργασία κειμένου, διαγραφή punctuation ώστε να είναι έγκυρο το λεξικό και exceptions).

B.1 Εκτέλεση παιχνιδιού

Η thefinalhangman αποτελεί το «κορμό» της όλης υλοποίησης αφού περιέχει τις μεθόδους που υλοποιούν τις απαιτήσεις του παιχνιδιού.

Η tuple.class είναι μια υποκλάση της hangman, έγινε προσθήκη της γιατί χρειαζόμουν tuples.

B.2. Δημιουργία γραφικής διεπαφής

Η Main χρειάζεται για να εκκινήσω το GUI.

Στη FXMLhangmanController έχω υλοποιήσει όλους τους Action/EventHandlers για το GUI.

Η FXMLhangman.fxml είναι το τελικό fxml αρχείο που προέκυψε με χρήση του scene builder.

Υλοποίησα κάθε απαίτηση της εκφώνησης και προσπάθησα να τηρώ όσο περισσότερο γίνεται τα OOP πρότυπα. Δεν χρησιμοποίησα subclasses που να κληρονομούν ιδιότητες καθώς η ίδια η εφαρμογή δεν είναι τόσο πολύπλοκη. Εντούτοις, έγινε χρήση private, public μεθόδων και static μεταβλητών ώστε να είναι πλήρης η εργασία. Τέλος, χρησιμοποίησα Javadoc σήμανση σε όλες τις κύριες μεθόδους.

Σχόλια

Χρησιμοποίησα eclipse version 4.22.0 με Java SE-11 και .jar αρχεία που έκανα download από libraries(συγκεκριμένα το 2.8.5.jar).

Έχω ένα ξεχωριστό φάκελο images, όπου φόρτωσα τα στιγμιότυπα της κρεμάλας.

Μαζί με τους παραδοτέους κώδικες για πληρότητα πρόσθεσα στο φάκελο project και τη finalehangman.class που είναι η αρχική υλοποίηση του game χωρίς GUI.

Παραδοχές

Όταν ο παίκτης βρίσκει ένα συγκεκριμένο γράμμα αποκαλύπτεται μόνο εκείνο της αντίστοιχης θέσης(ας υπάρχει το ίδιο και σε άλλη θέση), έτσι ώστε να γίνεται σωστά η διαγραφή των αντίστοιχων λέξεων στο λεξικό σύμφωνα με τον αλγόριθμο.

Το OPEN-LIBRARY-ID αντιστοιχεί σε json αρχείο όπου το value ανήκει στο αντικείμενο description όπως τα **OL31390631M**, **OL107192W**, **OL33891821M**. Διαφορετικά δεν θα λειτουργήσει.

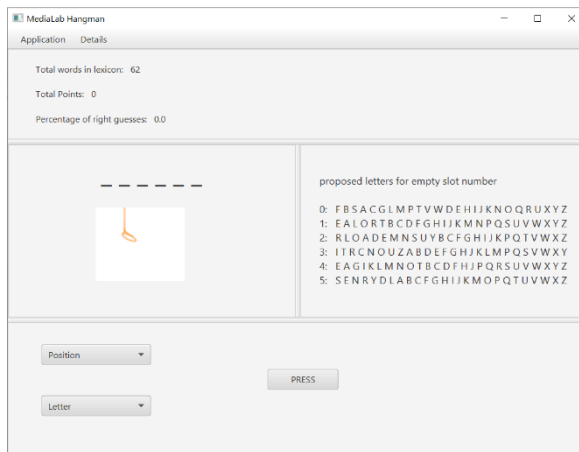
Κάθε φορά που ο χρήστης θέλει να φορτώσει νέο game πρέπει να κάνει πρώτα LOAD το txt αρχείο με το αντίστοιχο DICTIONARY-ID και στη συνέχεια να πατήσει START για να εκκινήσει το παιχνίδι.

Για να εμφανιστούν τα στατιστικά των 5 τελευταίων games χρειάζεται να έχει τερματίσει(επιτυχώς ή ανεπιτυχώς) τουλάχιστον 5 games. Έχει προστεθεί και ένα exception σε αυτή τη περίπτωση.

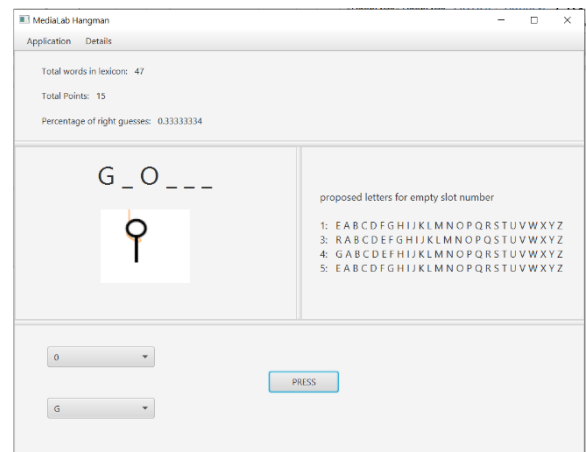
Πηγές

- διαφάνειες εργαστηρίου
- stackoverflow
- geeksforgeeks
- <https://docs.oracle.com/javase/8/javase-clienttechnologies.htm>
- <https://www.youtube.com/watch?v=q5A-qW2eGKs>
- java libraries

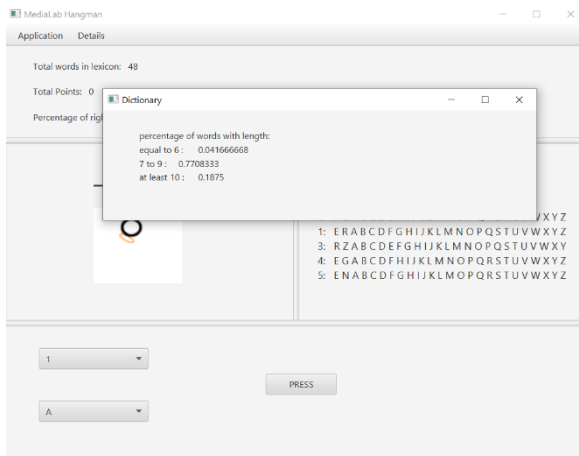
GUI screenshots



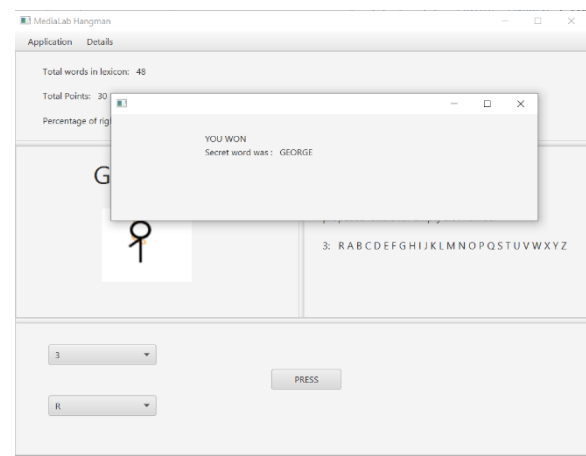
Εικόνα 1. Σημείο εκκίνησης επιλεγμένου παιχνιδιού



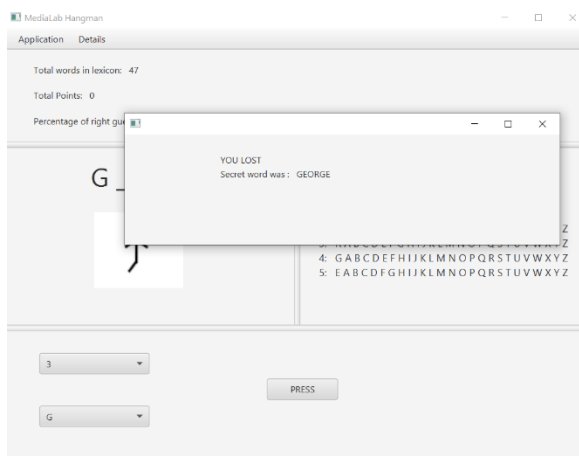
Εικόνα 2. Κατά τη διάρκεια του παιχνιδιού



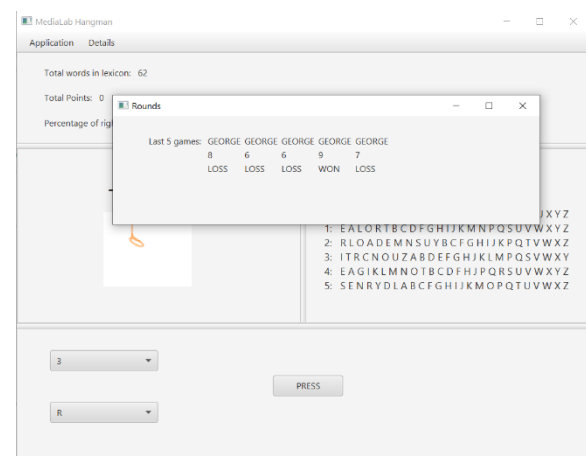
Εικόνα 3. Το dictionary pop-up παράθυρο



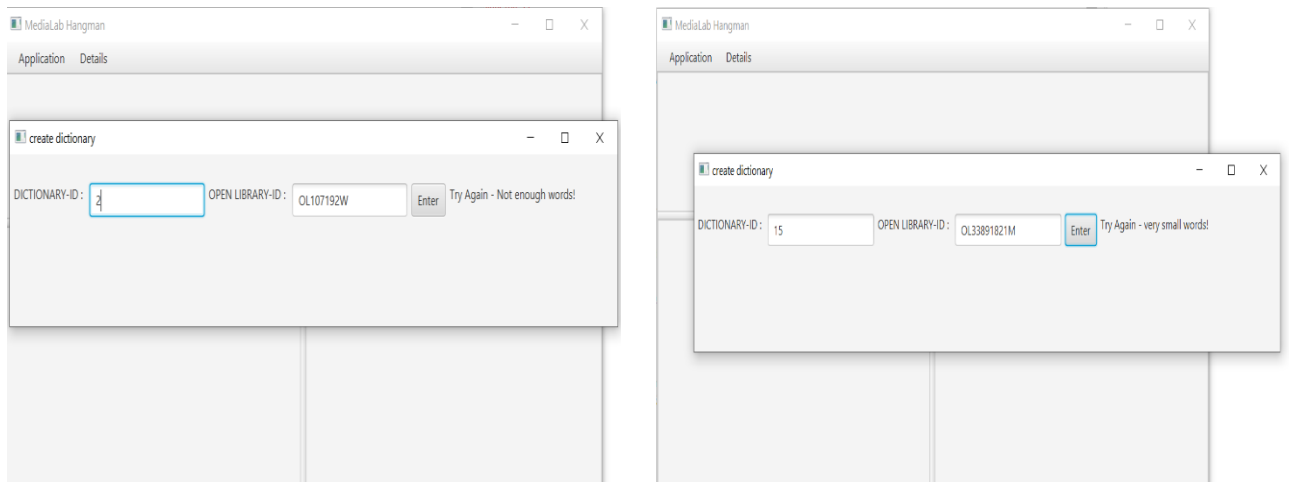
Εικόνα 4. Σήμανση επιτυχίας



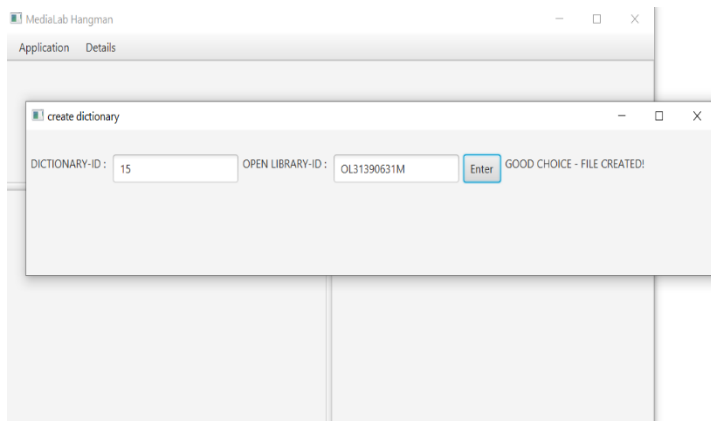
Εικόνα 5. Σήμανση αποτυχίας



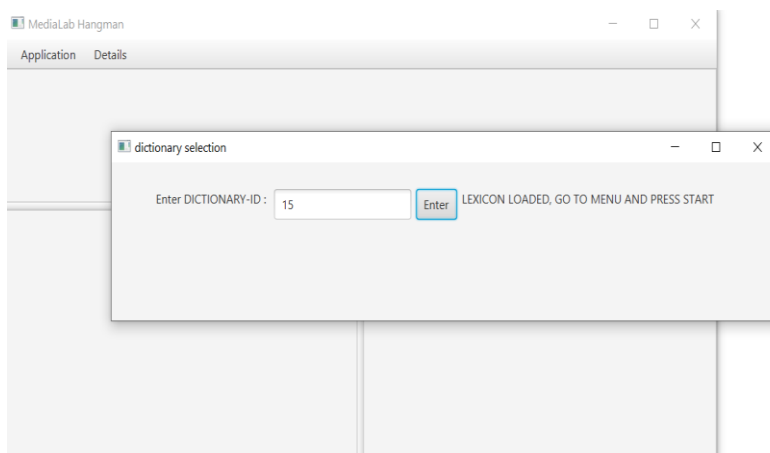
Εικόνα 6. Το Rounds pop-up παράθυρο



Εικόνα 7-8. Αποτυχία δημιουργίας dictionary



Εικόνα 9. Επιτυχής δημιουργία αρχείου-dictionary



Εικόνα 10. Επιτυχής επιλογή dictionary-φόρτωση παιχνιδιού

Καλή χρονιά!