
Τεχνολογία Πολυμέσων Τελική Εργασία

Παπαβασιλόπουλος Παναγιώτης-Φίλιππος (el17444)

Στα πλαίσια της εργασίας θα υλοποιήθηκε μια έκδοση του κλασικού παιχνιδιού «Κρεμάλα», όπου ο χρήστης θα παίζει με αντίπαλο τον υπολογιστή. Χρησιμοποιήθηκαν Java για τις λειτουργίες, JavaFX για το GUI και org.json για τη διαχείριση των json data από το open library.

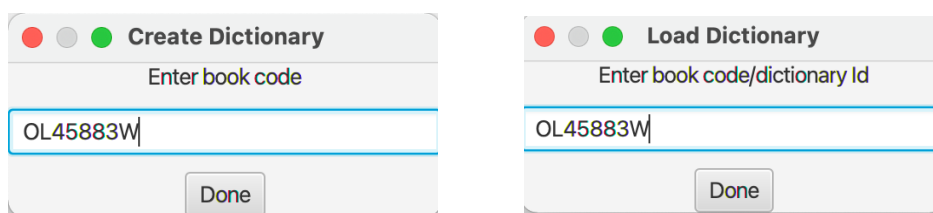
Σημείωση:

Υλοποιήθηκαν όλες οι ζητούμενες λειτουργίες της εκφώνησης ωστόσο δεν προστέθηκαν όσα exceptions χρειάζονται στον κώδικα και πολλά σημεία του κώδικα θα μπορούσαν να χωριστούν σε περισσότερες κλάσεις (OOP design principles). Τεκμηριωμένες σύμφωνα με τις προδιαγραφές του javadoc είναι οι public μέθοδοι: `createCharArrFromFile()`, `isDictGood()`, `hasGoodPercentage()`, `hasMoreThanSix()`, `noDupprl()` όπου ιδανικά θα έπρεπε να βρίσκονται σε δική τους κλάση καθώς σχετίζονται με το loading του .txt file και την εξαγωγή των λέξεων για το παιχνίδι.

Μέρος 1: Δημιουργία και φόρτωση λεξικού

Μέσω της επιλογής στο μενού Application -> Create, εμφανίζεται ένα popup παράθυρο στο οποίο ο χρήστης δίνει το DICTIONARY-ID του λεξικού που θέλει να χρησιμοποιήσει π.χ. OL45883W. Με το API της open library παίρνουμε τα δεδομένα τα οποία για την επεξεργασία τους, προσθέσαμε το .jar αρχείο που περιλαμβάνεται στον φάκελο της εργασίας (org.json). Άμα πάμε να φορτώσουμε 2η φορά τον ίδιο κωδικό στο console log του ide δείχνει το μήνυμα: File already exists. Το κείμενο φορτώνεται σε ένα αρχείο .txt όπου έπειτα το κάνουμε load και το επεξεργαζόμαστε με την επιλογή στο μενού Application -> Load.

GUI:



Console:

```
/Library/Java/JavaVirtualMachines/jdk-13.0.2.jdk/Contents/Home/bin/java ...  
File already exists.
```

Αλλιώς:

```
File created: hangman_OL45883W.txt  
Successfully wrote to the file.  
Book id : OL45883W
```

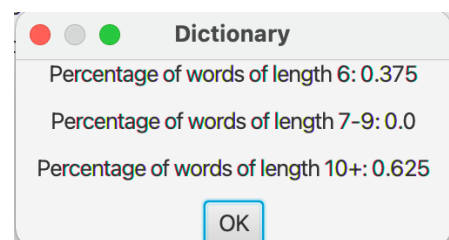
Αν δοθεί κωδικός που δεν ισχύει δεν γίνεται handle το exception.

Με τις συναρτήσεις: noDuppl, hasMoreThanSix, hasGoodPercentage, isDictGood προσαρμόζουμε το λεξικό ώστε να μην έχει πάνω από δύο φορές μία λέξη, να κρατήσουμε μόνο τις λέξεις με περισσότερα από 6 γράμματα, να δούμε αν 20% και πάνω των λέξεων έχει πάνω από 9 γράμματα και αν όσες λέξεις μένουν είναι πάνω από 20.

GUI:

- Εμφανίζεται το πλήθος λέξεων στο λεξικό που μπορούν να χρησιμοποιηθούν για το παιχνίδι
- Με την επιλογή από το μενού Details-> Dictionary εμφανίζονται πληροφορίες για τα ποσοστά των λέξεων του λεξικού ανάλογα με το μήκος τους

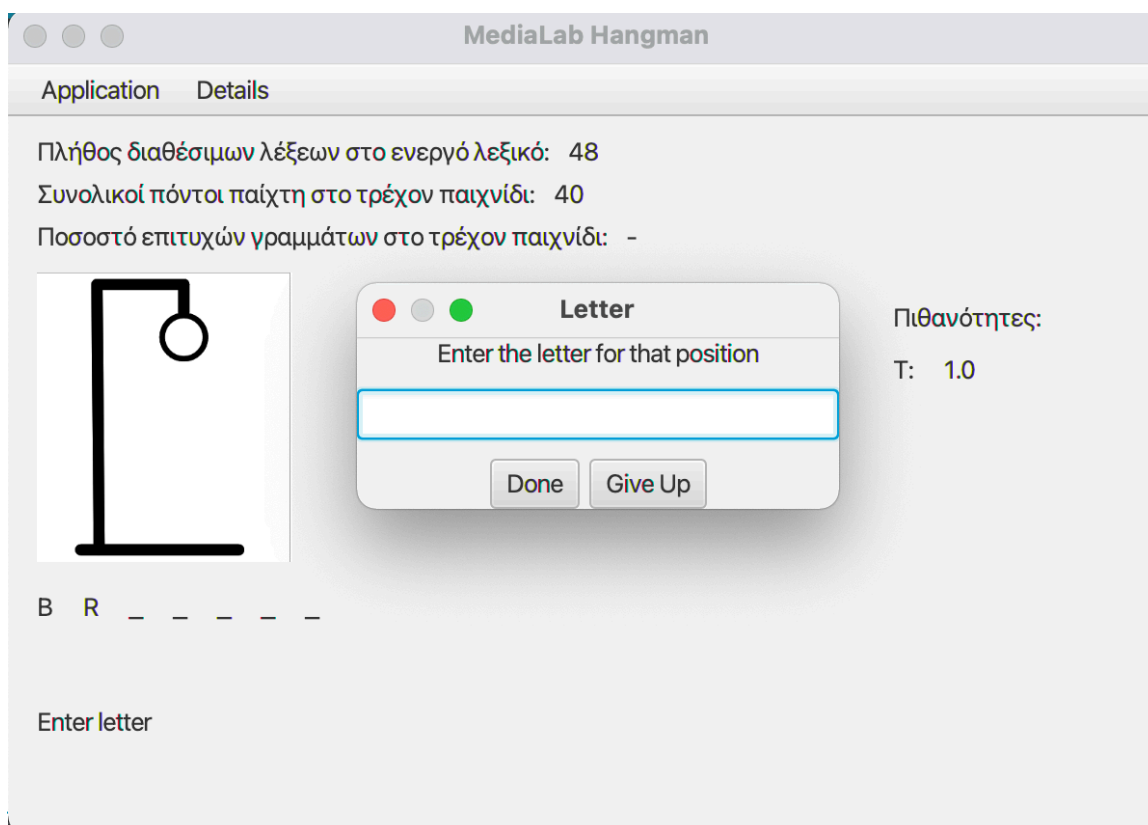
Πλήθος διαθέσιμων λέξεων στο ενεργό λεξικό: 48



Μέρος 2: Παιχνίδι

Παραδοχές:

1. Αντί να επιλέγει ο παίχτης τη θέση του γράμματος και το γράμμα στο κάτω μέρος του παραθύρου τα επιλέγει μέσω popup παραθύρων. Στο κάτω μέρος υπάρχει ένα label που εμφανίζονται μηνύματα προς τον χρήστη.
2. Η επιλογή Details -> Solution στο μενού έχει αντικατασταθεί με το κουμπί "Give Up" στα παραπάνω popup παράθυρα με την ίδια λειτουργία



ΣΗΜΕΙΩΣΗ

Καθώς άλλα json από το open library είχαν στο πεδίο description το value και άλλες όχι, στην συνάρτηση parse() που βρίσκεται στο τέλος της κλάσης Controller, έφτιαξα και τις 2 επιλογές απλά κάθε φορά που θέλω να γίνει χρήση της μίας κάνω comment τον κώδικα για την άλλη (1 γραμμή). Στην περίπτωση OL45883W δεν έχουμε value άρα κάνω comment την αντίστοιχη γραμμή.

Αυτό θα μπορούσε να γίνει με radio button στο popup create όπου ο χρήστης θα μπορούσε να επιλέξει (αν φυσικά το γνώριζε) τι μορφής θα είναι τα δεδομένα που θα βάλει (με το πεδίο value ή χωρίς).

Στην παραδοτέα είναι **by default** στην συνάρτηση **parse()** να δέχεται αρχεία που έχουν το πεδίο **description χωρίς το value**.

Κωδικοί βιβλίων:

Μόνο με πεδίο description

Mr Fox: OL45883W

The Elements of Style: OL38285W

*Desperation: OL81611W

*Misery: OL81634W

Με το πεδίο value μέσα στο description:

*The Lord of the Rings: OL27448W

*The Green Mile: OL81629W

Lisey's Story: OL81589W

*κάποιες λέξεις μπορεί να έχουν στην αρχή ή στο τέλος τους τους χαρακτήρες []