



**ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ**  
**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ**  
**ΥΠΟΛΟΓΙΣΤΩΝ**  
**ΑΚΑΔΗΜΑΪΚΗ ΠΕΡΙΟΔΟΣ 2024-25**

**ΤΕΛΙΚΗ ΕΡΓΑΣΙΑ ΜΑΘΗΜΑΤΟΣ ΤΕΧΝΟΛΟΓΙΑΣ**  
**ΠΟΛΥΜΕΣΩΝ**

Ονοματεπώνυμο: Αλεξιάδης Αντώνιος  
Αριθμός Μητρώου: 03120167

Σχεδιασμός Υλοποίησης

Για την υλοποίηση της εφαρμογής ακολουθήθηκε το MVC (Model, View, Controller) design pattern, μια σχεδιαστική τεχνική που ευθυγραμμίζεται με τις αρχές του Object Oriented Programming. Πιο συγκεκριμένα, στο directory models, βρίσκουμε την αναπαράσταση των Categories, Tasks, Reminders και Priorities ως δεδομένων. Στο directory controllers υπάρχουν τα controllers αυτά που χειρίζονται το business logic για τις διάφορες λειτουργίες της εφαρμογής, ενώ τέλος η γραφική διεπαφή έχει χωριστεί σε Panes (παράθυρα) τα οποία αποτελούν τα αντίστοιχα Views.

Δομή Πληροφορίας σε JSON αρχεία

Η κατάσταση της εφαρμογής αποθηκεύεται σε αρχεία JSON μέσω JSON Objects. Η πληροφορία για την κατάσταση ανακτάται από τα αρχεία κατά την αρχικοποίηση της εφαρμογής και τα αρχεία αυτά ανανεώνονται μόνο κατά τον τερματισμό της, με όλες τις ενδιαμέσες λειτουργίες να λαμβάνουν χώρα στην μνήμη του προγράμματος.

To data schema είναι το ακόλουθο:

Task {id, title, description, category, priority, deadline, status}

Category {name}

Priority {name}

Reminder {id, task\_id, reminder\_date, description}

Στο directory medialab υπάρχουν και τα αντίστοιχα JSON files που αποθηκεύουν τις παραπάνω πληροφορίες. Για τα categories και priorities, έγινε η παραδοχή στο data schema ότι ο χρήστης δεν θα μπορεί να εισάγει 2 categories ή priorities με το ίδιο όνομα και για αυτόν τον λόγο ένα πεδίο name μαζί με το hashcode του είναι αρκετό για την μοναδική ταυτοποίηση. Στα Reminders χρειαζόμαστε μοναδικό id καθώς ένας χρήστης μπορεί να θέσει πολλαπλά reminders για ένα task και φυσικά το task\_id για να μπορούμε να διακρίνουμε σε ποιο task αναφέρεται το εκάστοτε Reminder. Τα υπόλοιπα πεδία, τόσο στο Task όσο και στο Reminder είναι απαραίτητα για να διασφαλιστούν οι απαιτήσεις της εφαρμογής.

### Παραδοχές και λειτουργικότητες

Έχουν υλοποιηθεί όλες οι λειτουργικότητες που ζητώνται. Οι παραδοχές που έχουν γίνει είναι οι ακόλουθες:

- Ο Χρήστης δεν μπορεί να επιλέξει την κατάσταση Delayed καθώς αυτή δίνεται αυτόματα μόνο σε Tasks που το deadline τους έχει ξεπεράσει την σημερινή ημερομηνία.
- Στα upcoming Tasks θεωρούμε ότι συγκαταλέγονται και τα Tasks με σημερινή ημερομηνία deadline.
- Όταν μεταβάλλουμε ένα Task σε completed διαγράφονται όλα τα Reminders που σχετίζονται με αυτό το Task.
- Δεν επιτρέπεται ο Χρήστης να θέσει Reminders σε Delayed Tasks.

- Ο Χρήστης μπορεί να θέσει πολλαπλά Reminders για το ίδιο Task με την ίδια ημερομηνία.
- Τα Reminders των οποίων το Reminder\_date είναι πριν τη σημερινή ημερομηνία διαγράφονται αυτόματα.
- Ο Χρήστης μπορεί να θέσει το Status ενός Task από Completed σε κάτι άλλο.
- Ένα Task μπορεί να δημιουργηθεί χωρίς προυπάρχουσα κατηγορία.

**Στο αρχείο ReminderController.java όλες οι public μέθοδοι είναι τεκμηριωμένες με βάση το εργαλείο Javadoc, όπως αναφέρει η εκφώνηση.**