

**ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ**

**ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ**

**ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ, ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΓΕΝΝΗΤΡΙΑ ΕΠΙΤΡΑΠΕΖΙΩΝ ΠΑΙΧΝΙΔΙΩΝ**

**Π****έμπτη Εβδομάδα**

**Ομαδική Εργασία:**

Αναστασιάδης Αλκίνοος (20003)

Ζήνα Ελένη (20046)

Λαγιόκαπας Δημήτριος (20079)

Μακρή Στυλιανή (20060)

Επιβλέποντες:

Κουρέας Αργύριος

Λάντζος Θεόδωρος

**ΣΕΡΡΕΣ****, 8 έως 12 ΜΑΙΟΥ 2023**

Περιεχόμενα

Περίληψη - Εισαγωγή 3

1. Μεθοδολογία 3
2. Υλοποίηση 4
3. Χρονοδιάγραμμα 11
4. Αποτελέσματα - Επίλογος 13

5.Βιβλιογραφία 13

Περίληψη - Εισαγωγή

Την Πέμπτη εβδομάδα της εργασίας βελτιώσαμε τον κώδικα μας αφαιρώντας περιττές κλάσεις και κάνοντας κάποιες αλλαγές στην διαχείριση των δεδομένων μας. Αναπτύχθηκαν επίσης κλάσεις και συναρτήσεις που χρειάζονται για να υλοποιηθούν οι κανόνες και η λογική του Endless παιχνιδιού, καθώς και δημιουργήθηκαν οι κλάσεις των Τiles, και Cards.

1. Μεθοδολογία

Πλέον, έχει αλλάξει η αρχιτεκτονική του προγράμματος έτσι ώστε κάθε κλάση να έχει να κάνει μόνο με δεδομένα που την αφορούν, χωρίς υπάρχουν περιττές συσχετίσεις μεταξύ των κλάσεων, καθώς και χωρίς να υπάρχουν αμφίδρομες μεταξύ κλάσεων του παιχνιδιού.   
Παρακάτω σας παραθέτουμε ένα σχεδιάγραμμα αποτύπωσης της αρχιτεκτονικής του παιχνιδιού, έτσι όπως έχει διαμορφωθεί:

Main

Game

Player

Board

Γραφή 42

Rule

Dice

Cards

Tile

1. Υλοποίηση

Στο πρώτο βήμα της υλοποίησης κώδικα για τη παρούσα εβδομάδα, έγινε η δημιουργία μιας κλάσης, που σκοπό έχει την κατάλληλη δημιουργία κλάσεων Tiles. Σε αυτή τη κλάση, εφαρμόστηκε το πρότυπο του Factory Pattern, το οποίο, με λίγα λόγια, έχει μια μοναδική λειτουργία, και αυτή είναι η δημιουργία κλάσεων Tiles, ανάλογα με τις παραμέτρους που υπάρχουν. Στη συγκεκριμένη περίπτωση, οι παράμετροι για τη δημιουργία των αντικειμένων Tiles, βρίσκονται στο αρχείο JSON που υπάρχει στο παιχνίδι μας. Έτσι, κατά αυτόν τον τρόπο, γίνεται η δημιουργία των αντικειμένων Tiles, στη κλάση του Factory Pattern, σύμφωνα με τη παράμετρο ‘*parameter*’ στου JSON αρχείου του παιχνιδιού μας.

Το πρότυπο του Factory Pattern, μας δίνει τη ευελιξία να χειριστούμε τα διαφορετικά Τiles που έχουμε εισάγει στο παιχνίδι μας, με κατάλληλο τρόπο, έτσι ώστε να μην έχουμε έναν μεγάλο και χωρίς νόημα κώδικα κατά τη δημιουργία ενός Tile. Το θετικό του Factory Pattern, είναι το γεγονός ότι μπορούμε να έχουμε εύκολη και γρήγορη επεκτασιμότητα του κώδικα μας, χωρίς να υπάρξει σχεδόν καμία αλλαγή κώδικα στο κύριο πυρήνα του παιχνιδιού.

Η αντίστοιχη με τη παραπάνω διαδικασία, έγινε και για τα αντικείμενα τύπου Cards, καθώς και για τα αντικείμενα τύπου Board, καθώς, και οι προαναφερόμενες δύο κλάσεις, έχουν την απαίτηση της διαχείρισης πολλών και διαφορετικών ειδών η κάθε μια ξεχωριστά.

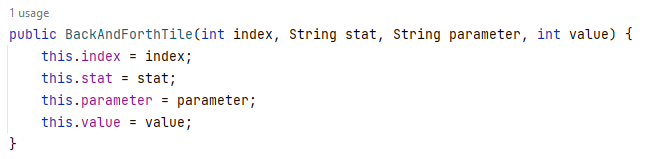


Ως προς τα είδη των Tiles που έχουμε εισάγει στο παιχνίδι μας, κάποιος, μπορεί να βρει μέσα σε αυτό, τα παρακάτω είδη: BackAndForthTile, BlankTile, CardTile, DicePointsTile, καθώς και SimpleTile.

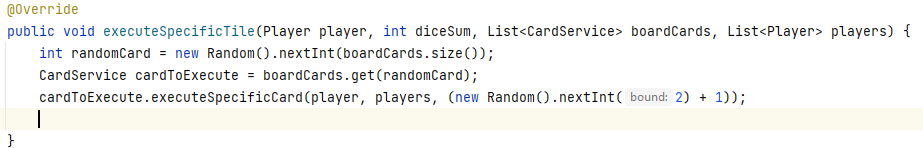
Για να μπορέσει να λειτουργήσει σωστά το Factory Pattern που υλοποιήθηκε, πρέπει όλα τα διαφορετικά είδη Tile που έχουμε στο παιχνίδι μας, να έχουν μια κοινή κλάση υλοποίησης. Αυτή η κλάση, είναι το Service που έχουμε δημιουργήσει για τα Tiles, εν ονόματι TileService.

Παρακάτω, μπορούμε να δούμε τους διαφορετικούς Constructors, που υπάρχουν στα διαφορετικά είδη Tile του παιχνιδιού.

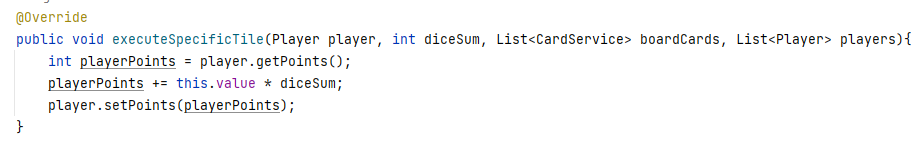
BackAndForthTile:



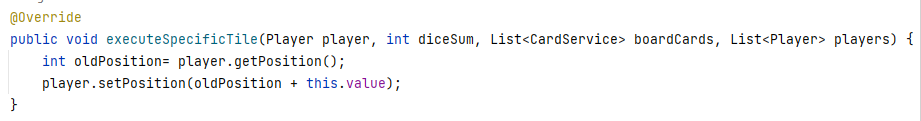
CardTile:



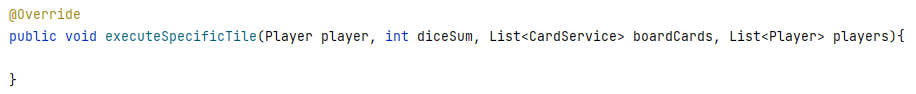
DicePointsTile:



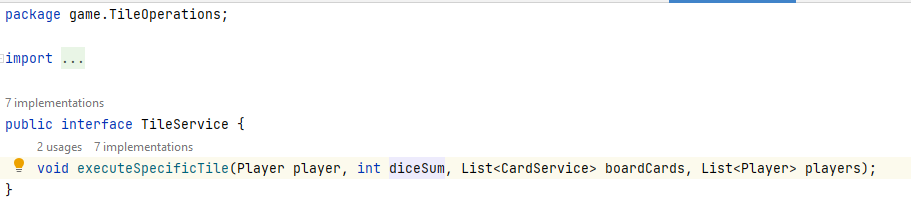
SimpleTile:



BlankTile:



Η κοινή κλάση υλοποίησης (TileService)



Όσο αναφορά τις κλάσεις που αφορούν το ταμπλό, έγιναν σημαντικές αλλαγές ως προς τις σχέσεις που υπάρχουν μεταξύ των κλάσεων του παιχνιδιού.

Η κλάση LimitedRangeBoard περιλαμβάνει τα εξής: (Με όμοιο τρόπο λειτουργεί και η EndlessBoard)

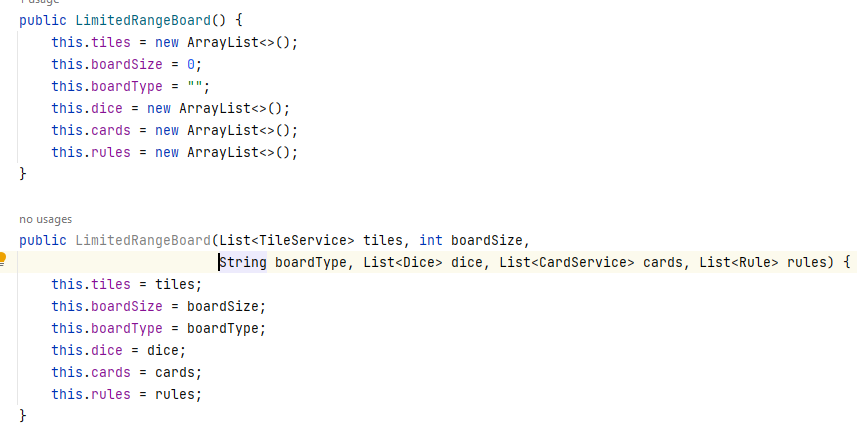
Πεδία:

* Tiles: Μια λίστα από TilesService αντικείμενα που αντιπροσωπεύουν τα τετραγωνάκια του ταμπλό.
* boardSize: Ένας ακέραιος που εκπροσωπεύει το μέγεθος του ταμπλό.
* boardType: ένα String που αντιπροσωπεύει τον τύπο του ταμπλό.
* Dice: Μία λίστα από αντικείμενα dice.
* Cards: μια λίστα από CardService αντικείμενα για τις κάρτες του παιχνιδιού.
* Rules: Μια λίστα από Rule αντικείμενα.



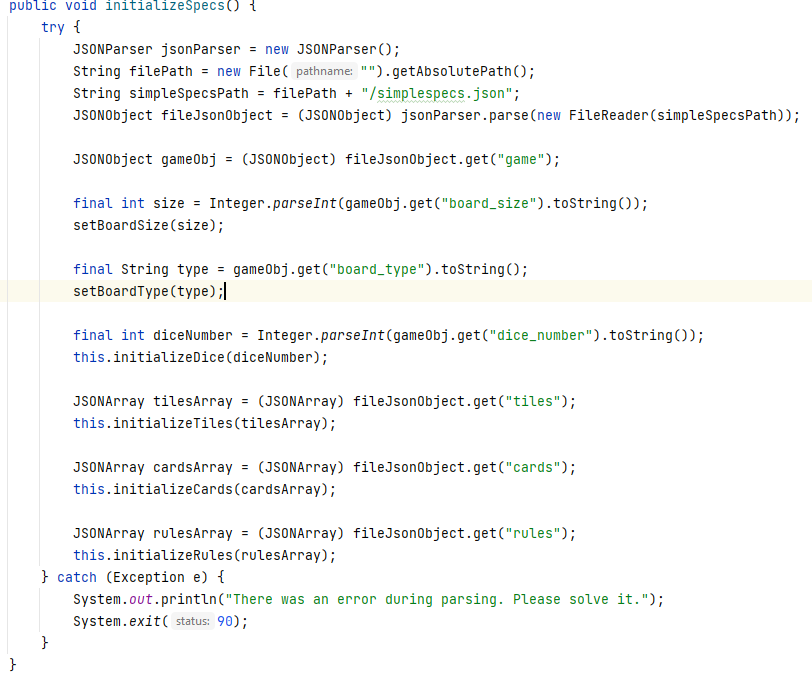
Constructor:

* LimitedRangeBoard(): Αρχικοποιεί ένα κενό LimitedRangeBoard αντικείμενο.

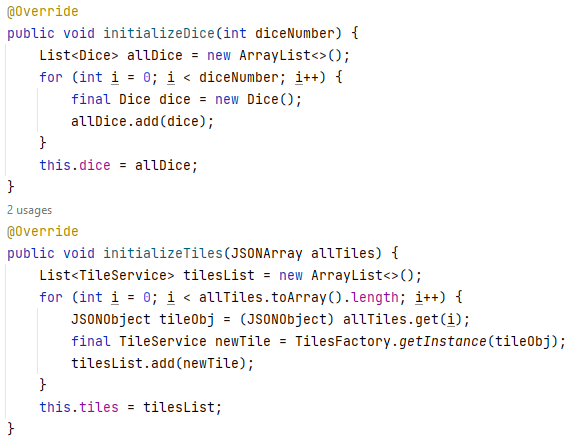


Συναρτήσεις:

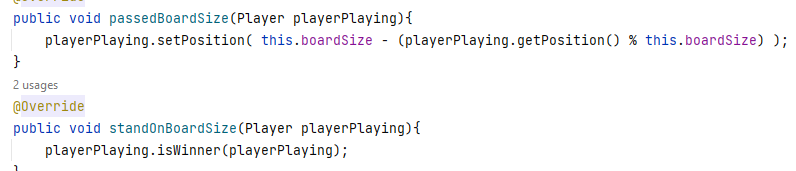
* Η initializeSpecs αρχικοποιεί τα specs του ταμπλό, δηλαδή τα board size, type, dice, tiles, cards, and rules, διαβάζοντας το json αρχείο simplespecs.json.



* Τα initializations για κάθε παράμετρο:



* Την PassedBoardSize που μετακινεί τον παίκτη πίσω σε περίπτωση που ξεφύγει από το ταμπλό, και την standOnBoardSize που ελέγχει αν το αντικείμενο Player έχει κερδίσει το παιχνίδι, αν βρίσκεται δηλαδή πάνω στο τελευταίο τετραγωνάκι.



* Getters και Setters για τα Tiles, το BoardSize, BoardType, Dice, Cards, Rules.

To BoardService: (Δεν εμφανίζεται ολόκληρο):



1. Χρονοδιάγραμμα

5/5/2023:

Ελένη:

Διαγραφή των περιττών κλάσεων του προγράμματος, όπως η κλάση RulesRecord.

Αλλαγή τρόπου εισαγωγής με JSONParser.

Διαχωρισμό της συνάντησης readGameSpecs, σε πολλές μικρές συναρτήσεις, ώστε να υπάρχουν πολλά μικρά αλλά και διαχειρίσιμα “κομμάτια” κώδικα.

6/5/2023:

Ελένη:

Δημιουργία τύπων Tiles και Cards με δικά τους interfaces καθώς και υλοποίηση των αντίστοιχων κλάσεων, που ανάλογα τον τύπο (boardtype), που το παιχνίδι κάνει αναγνώριση από JSON αρχείο, θα δημιουργείτε η κατάλληλη κλάση Tile ή/και Card, με τη χρήση του Factory Pattern που περιγράφηκε παραπάνω.

7/5/2023:

Ελένη:

Αλλαγή αρχιτεκτονικής και χρήση του Factory Pattern για τις κλάσεις των Tiles και Board, ώστε να δημιουργούμε τα κατάλληλα Tiles (που διαβάζονται από το αρχείο JSON), για το αντίστοιχο Board που έχει επιλέξει ο παίκτης, με δυνατότητα προσθήκης κι άλλων Τile.

8/5/2023:

Ελένη:

Δημιουργία Factory Pattern για τη κλάση των Cards με παρόμοια λογική όπως και η αντίστοιση υλοποίηση στη κλάση των Τiles.

9/5/2023:

Ελένη:

Δόθηκαν στην ομάδα οι κατάλληλες αρμοδιότητες σύμφωνα με την εμπειρία και τις γνώσεις του κάθε μέλους της ομάδας.

10/5/2023:

Στέλλα:

Δημιουργία συνάρτησης PrintRankAndWinner η οποία κατατάσσει τους παίκτες σε αύξουσα σειρά ανάλογα με τους πόντους μετά από 30 γύρους και εκτυπώνει τον νικητή.

Διαμόρφωση της EndlessBoard σύμφωνα με τη νέα λογική του προγράμματος.

11/5/2023:

Αλκίνοος:

Διόρθωση των συναρτήσεων που ξεκίνησε η Στέλλα (passedBoardSize, standOnBoardSize) στην κλάση EndlessBoard και κατάλληλη παραμετροποίηση της playGame της κλάσης Game.

Δημήτρης:

Σύνταξη μηνυμάτων εξόδου για τις διάφορες περιπτώσεις που μπορεί να προκύψουν κατά την εκτέλεση των ενεργειών κάθε Tile ή/και Card.

Σύνταξη μιας δομής ελέγχου που ως στόχο έχει τον έλεγχο των πόντων του παίκτη, σε αντιστοιχία με το τύπο του παιχνιδιού.

Στέλλα:

Σύνταξη του report.

1. Αποτελέσματα - Επίλογος

Το παιχνίδι μας πλησιάζει την τελική του μορφή. Οι αλλαγές που έγιναν κρίθηκαν απαραίτητες από όλα τα μέλη της ομάδας μας, ώστε το παιχνίδι μας να είναι εύκολο στη συντήρηση και καθώς και να μπορεί να υποστηρίξει εύκολα νέες επεκτάσεις του. Τις ερχόμενες εβδομάδες θα τελειοποιήσουμε το πρόγραμμα, ώστε να εκτελείτε σε κάθε πιθανή περίπτωση χρήσης, καθώς και θα εκτελέσουμε ορισμένα στοχοποιημένα Τests έτσι ώστε να διορθώσουμε τυχόν λάθη που μπορεί προκύψουν.

5.Βιβλιογραφία

|  |  |
| --- | --- |
| [16] | Design Pattern - Factory Pattern - Tutorialspoint  <https://www.tutorialspoint.com/design_pattern/factory_pattern.htm> |
| [17] | Tutorial - Factory Design Pattern in Java  <https://www.digitalocean.com/community/tutorials/factory-design-pattern-in-java> |