ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

Master Chef 2018: 1η Εργασία

Κοσέογλου Σωκράτης

ΑΕΜ: 8837

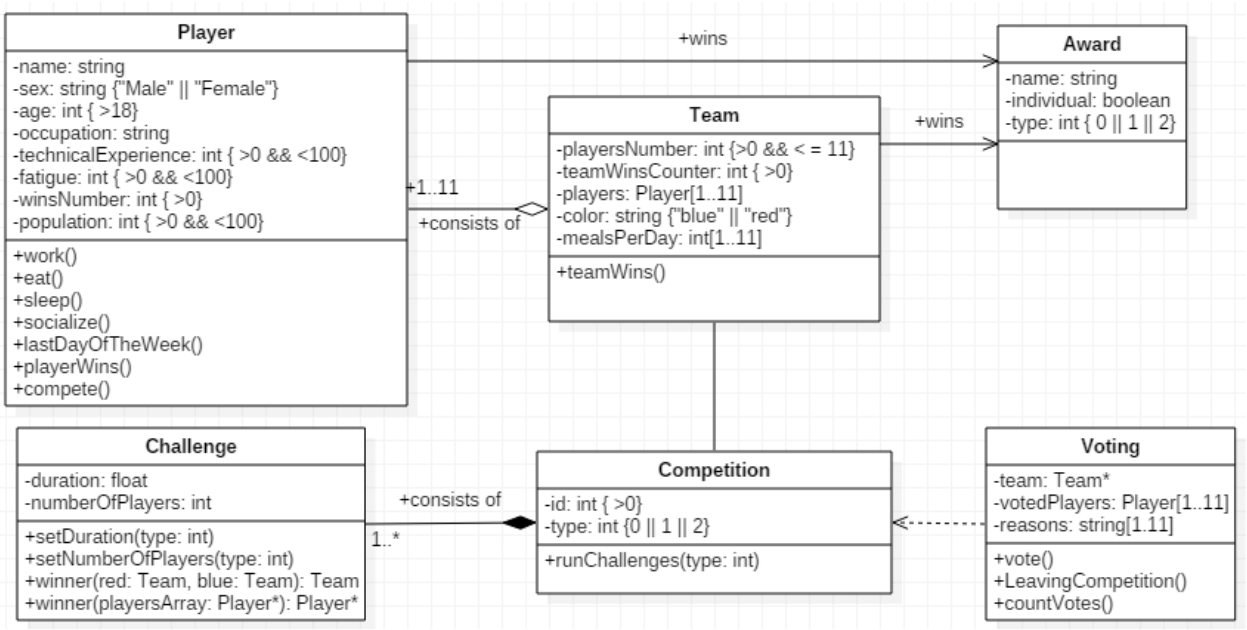
Sokrkose@ece.auth.gr

Καρακώστας Κωνσταντίνος

ΑΕΜ: 7892

karakosk@ece.auth.gr

UML Design



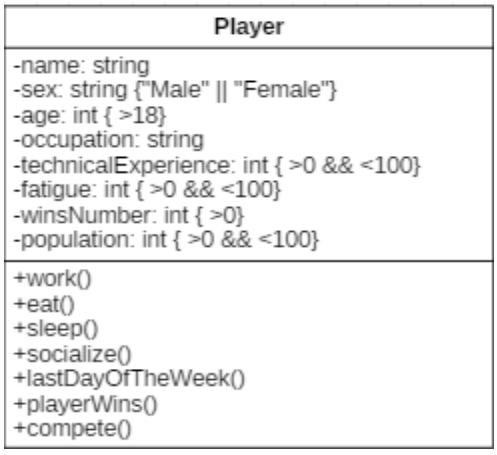
Περιγραφή Υλοποίησης

Αρχικά, το Master Chef είναι ένα παιχνίδι που αποτελείτε από 2 ομάδες, την μπλε και την κόκκινη. Ακόμη, η κάθε ομάδα αποτελείτε από το πολύ 11 παίκτες. Για τον λόγο αυτό στην UML υλοποίηση έχουμε δημιουργήσει δύο κλάσεις, την **Player** η οποία έχει τα καίρια χαρακτηριστικά ενός παίκτη αλλά και την κλάση **Team** η οποία μας δίνει την δυνατότητα να διαχειριζόμαστε τα δεδομένα της κάθε ομάδας (πχ χρώμα, μέλη/παίκτες, νίκες) οι οποίες συνδέονται με **Aggregation** και όχι **Composition** αφού η διαγραφή της κλάσης Team δεν συνεπάγει την διαγραφή της Player. Επιπλέον κατά την διάρκεια του παιχνιδιού οι παίκτες καλούνται να συμμετάσχουν σε διάφορους διαγωνισμούς, διαφορετικού τύπου. Για τον λόγο αυτό δημιουργήσαμε την κλάση **Competition.** Επιπλέον ο κάθε διαγωνισμός αποτελείτε από επιμέρους δοκιμασίες **Challenge** οι οποίες μπορεί να διαφέρουν τόσο στην διάρκεια όσο και στον αριθμό των παικτών που συμμετέχουν, γι΄ αυτό συνδέονται μεταξύ τους με **composition** αφού η κλάση Challenge είναι άρρηκτα συνδεδεμένη και δεν μπορεί να υπάρξει χωρίς την κλάση Competition. Τέλος με το πέρας του διαγωνισμού χρίζεται νικητής είτε ένας παίκτης είτε μια ομάδα, ανάλογα με την φύση του διαγωνισμού. Ο ατομικός νικητής κερδίζει ένα έπαθλο **Award** το οποίο ποικίλει, ενώ η ηττημένη ομάδα καλείτε να συμμετάσχει σε ψηφοφορία **Voting** ώστε να αποχωρίσει το μέλος που συνεισφέρει λιγότερο κατά την άποψη τους. Η κλάση Award συνδέεται με τις Team και Player με **Directed Association** καθώς τόσο η κάθε ομάδα όσο και ο κάθε παίκτης ατομικά μπορούν να **κερδίσουν** βραβείο. Τέλος η κλάση Voting συνδέεται με **Dependency** με την κλάση Competition καθώς η Voting δεν μπορεί να υπάρξει χωρίς την Competition ενώ η Competition συνδέεται με την Team με **Simple Association.**

Λίστα Κλάσεων

- **Player**
- **Team**
- **Competition**
- **Challenge**
- **Award**
- **Voting**

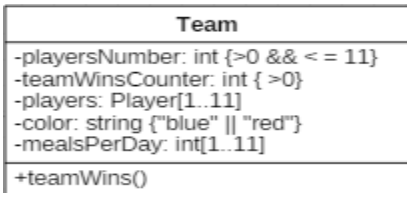
- # Player



Η κλάση αυτή περιέχει τα βασικά χαρακτηριστικά του κάθε παίκτη, όπως όνομα **name**, φύλο **sex**, ηλικία **age**, επάγγελμα εκτός παιχνιδιού **occupation**, τεχνική κατάρτιση **technical experience** η οποία αρχικοποιείται τυχαία, με τιμές από 0 έως 100, για κάθε παίκτη και θα γίνεται μέσα από τον constructor της κλάσης **Player**, κάτι το οποίο θεωρήσαμε ότι θα ήταν πλεονασμός να το συμπεριλάβουμε στη UML κλάσεων. Επιπλέον έχει τις μεταβλητές κούραση **Fatigue** και δημοφιλία **popularity** τα οποία αρχικοποιούνται επίσης στον constructor του Player με τιμές 0 και 50 αντίστοιχα. Τέλος ο κάθε παίκτης έχει ένα συγκεκριμένο αριθμό νικών

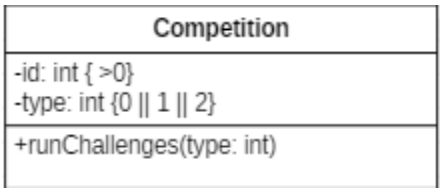
σε διαγωνισμούς που προσμετράτε από την μεταβλητή **winsNumber**.Όπως και σε κάθε κλάση οι μεταβλητές έχουν visibility: private συνεπώς θα χρειαστούν public συναρτήσεις της κλάσης Player όπως setName(), getName(), αλλά και άλλες συναρτήσεις οι οποίες θα είχαν αύξηση την πολυπλοκότητα του διαγράμματος εάν τις είχαμε συμπεριλάβει. Όσον αφορά τις συναρτήσεις έχουμε τις **work()**, **eat()**, **sleep()**, **socialize()**,**lastDayOfTheWeek()** οι οποίες όπως λένε και τα ονόματα τους μεταβάλουν τις μεταβλητές της κλάσης Player αναλόγως (όπως λέει στην εκφώνηση της άσκησης). Ακόμη, έχουμε την **playerWins()** η οποία αυξάνει τις νίκες ενός παίκτη όταν αυτός νικάει σε ένα διαγωνισμό και την **compete()** η οποία δείχνει ότι ένας παίκτης συμμετείχε σε κάποιο διαγωνισμό γεγονός που έχει ως αποτέλεσμα να μεταβάλει την κούραση του εκάστοτε παίκτη.

- # Team



Η κλάση αυτή έχει τα επιμέρους χαρακτηριστικά , αλλά και τις συναρτήσεις που τα μεταβάλουν, της κάθε ομάδας. Τέτοια χαρακτηριστικά είναι οι αριθμοί των παικτών της κάθε ομάδας **playersNumber** με περιορισμό που φαίνεται παραπάνω και αρχικοποίηση 11 στον constructor της Team, η **teamWinsCounter** που μετράει τις νίκες της κάθε ομάδας, **players** η οποία είναι τύπου Player και δείχνει ποίοι και πόσοι παίκτες αποτελούν την κάθε ομάδα, **color** το χαρακτηριστικό που ξεχωρίζει τις δύο ομάδες “blue” OR “red” αλλά και την μεταβλητή **mealsPerDay[]** η οποία είναι ένας παράλληλος πίνακας με τον πίνακα **players[]** η οποία στην αρχή της κάθε βδομάδας έχει τον αριθμό 14 σε όλα τα κελιά και δείχνει πόσες μερίδες φαγητό έχει φτιάξει ο κάθε παίκτης της ομάδας. Τέλος έχει την μέθοδο **teamWins()** η οποία αυξάνει τις νίκες τις κάθε ομάδας όταν κερδίζει ένα διαγωνισμό.

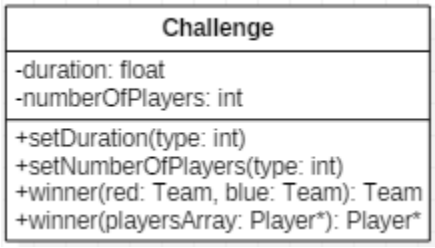
- # Competition



Ο κεντρικός πυλώνας του παιχνιδιού είναι οι διάφοροι διαγωνισμοί στους οποίους λαμβάνουν μέρος οι παίκτες του παιχνιδιού. Κάθε διαγωνισμός έχει ένα προσωπικό αριθμό ο οποίος αναπαριστάται με την μεταβλητή **id** και οι διαγωνισμοί είναι τριών ειδών:

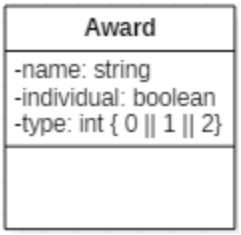
Ομαδικοί, Ασυλίας και Αποχώρησης οπότε η μεταβλητή **type** παίρνει αντίστοιχα τις τιμές 0,1 και 2. Τέλος έχει την μέθοδο **runChallenges()** η οποία δέχεται ως όρισμα τον τύπο του διαγωνισμού και από εκεί και πέρα ξεκινάει τις διάφορες δοκιμασίες οι οποίες ποικίλουν τόσο σε διάρκεια όσο και σε αριθμό παικτών, για τον λόγο αυτό ορίζουμε μια νέα κλάση όπως θα δούμε παρακάτω, την κλάση **Challenge**.

- **Challenge**



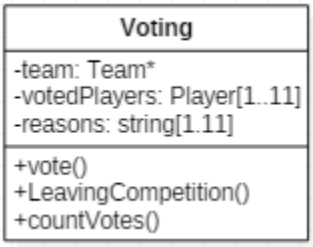
Αυτή η κλάση έχει ως μεταβλητές τον χρόνο της κάθε δοκιμασίας ενός διαγωνισμού **duration**, αλλά και τον αριθμό τον παικτών που συμμετέχουν **numberOfPlayers**. Τις μεταβλητές αυτές τις χειριζόμαστε μέσω των μεθόδων **setDuration()** και **setNumberOfPlayers()** οι οποίες παίρνουν σαν όρισμα την μεταβλητή **type** καθώς ο ομαδικός διαγωνισμός έχει μεγαλύτερη διάρκεια από τους άλλους δύο αλλά και ο αριθμός των παικτών που συμμετέχουν μεταβάλλεται ανάλογα με τον τύπο του διαγωνισμού. Ακόμη η συνάρτηση **winner** γίνεται overloading και χωρίζεται σε δυο κατηγορίες(Ατομικός και Ομαδικός Νικητής). Η συνάρτηση **winner(red: Team, blue: Team) : Team** παίρνει ως ορίσματα τις δύο ομάδες τύπου Team και γυρνάει την νικητήρια ομάδα του ομαδικού διαγωνισμού. Ενώ η **winner(playersArray: Player*): Player*** παίρνει σε κάθε δοκιμασία ατομικού διαγωνισμού ένα πίνακα playersArray[] τύπου Player* με τους **n** παίκτες που αγωνίζονται σε κάθε δοκιμασία και επιστρέφει κάθε φορά ένα πίνακα με **n-1** στοιχεία που είναι οι νικητές της κάθε δοκιμασίας μέχρις ότου στην τελική δοκιμασία ο πίνακας να έχει 1 μόνο στοιχείο το οποίο είναι και ο νικητής του ατομικού διαγωνισμού.

- **Award**



Στην κλάση αυτή ορίζουμε τα έπαθλα των νικητών και ορίζουμε τις μεταβλητές **name** που έχει το όνομα του νικητή, την μεταβλητή **individual** που δείχνει αν είναι ομαδικό ή όχι το έπαθλο και την την **type** που δηλώνει το είδος του επάθλου(ασυλία, φαγητό, εκδρομή). Δεν έχουμε δημιουργήσει μεθόδους καθώς από την εκφώνηση δεν μας δίνετε προς το παρόν κριτήριο επεξεργασίας αυτών των μεταβλητών.

- # Voting



Η κλάση αυτή αναπαριστά την διαδικασία ψηφοφορίας. Η συνάρτηση **vote()** διαχειρίζεται τους τρεις παράλληλους πίνακες/μεταβλητές της κλάσης Voting. Ο πρώτος πίνακας **team[]** έχει τους παίκτες της χαμένης ομάδας, ο δεύτερος πίνακας **votedPlayers[]** έχει τους παίκτες που ψηφίστηκαν ενώ ο τρίτος παράλληλος πίνακας **reasons[]** έχει τους λόγους για τους οποίους ψηφίστικαν. Στην συνέχεια η συνάρτηση **countVotes()** μετράει τις ψήφους του πίνακα **votedPlayers[]** και δίνει στην συνάρτηση **LeavingCompetion()** τους παίκτες με τις περισσότερες ψήφους, όπου οι κριτές επιλέγουν ποιος θα αποχωρίσει!