

**Πανεπιστήμιο Κρήτης –Τμήμα Επιστήμης Υπολογιστών**

**ΗΥ252– Αντικειμενοστρεφής Προγραμματισμός**

**Διδάσκων: Ι. Τζίτζικας**

**Χειμερινό Εξάμηνο 2020-2021**

Stratego

Εισαγωγή

Εμμανουήλ Καστανίδης

4131

02/12/2022

Think and describe what you plan to do and why it will be useful.

Περιεχόμενα

[1. Εισαγωγή](#_Toc530045458) 2

[2. Η Σχεδίαση και οι Κλάσεις του Πακέτου Model](#_Toc530045459) 2

[3. Η Σχεδίαση και οι Κλάσεις του Πακέτου Controller](#_Toc530045460) 7

[4. Η Σχεδίαση και οι Κλάσεις του Πακέτου View](#_Toc530045461) 7

[5. Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML](#_Toc530045462) 9

[6. Λειτουργικότητα (Β Φάση)](#_Toc530045463) -

[7. Συμπεράσματα](#_Toc530045464) -

## Εισαγωγή

Στο συγκεκριμένο project έχει εφαρμοστεί η αρχιτεκτονική MVC.

Στη πρώτη ενότητα αναλύονται οι κλάσεις, η διασύνδεση και τα πακέτα που βρίσκονται εντός του πακέτου Model, καθώς και ο σχεδιασμός τους και η λειτουργικότητα τους.

Στη δεύτερη ενότητα αναλύεται η κλάση Controller που βρίσκεται εντός του πακέτου Controller, καθώς και η αλληλεπίδραση του με τις κλάσεις Player και View.

Στη τέταρτη ενότητα αναλύεται η κλάση View και η διασύνδεση ViewInterface που βρίσκονται εντός του πακέτου View.

Στη πέμπτη ενότητα δίδονται τα διαγράμματα UML των κλάσεων -πλην της View.

## Η Σχεδίαση και οι Κλάσεις του Πακέτου Model

Εντός του πακέτου Model περιέχεται πληθώρα πακέτων για λόγους διευκόλυνσης εύρεσης των κλάσεων.

Ένα εκ των πακέτων της Model είναι η game\_characters, η οποία περιέχει όλα τα τέρατα του παιχνιδιού και τις βασικές κλάσεις τους.

Η GameCharactersInterface είναι μία διεπαφή που περιέχει όλες τις βασικές λειτουργίες που οποιοδήποτε τέρας είναι ικανό να εκτελέσει (attack, move). Περιέχει και μεθόδους που ικανοποιούν τις αρχές της ενθυλάκωσης (getters, τα setters είναι τύπου protected και package).

Οι μεθόδοι move και attack είναι αφηρημένες, επειδή διαφορετικά τέρατα την υλοποιούν με διαφορετικό τρόπο.

Η αφηρημένη κλάση GameCharacters υλοποιεί τη παραπάνω διεπαφή και προσθέτει getters τύπου protected και μία μέθοδο που συγκρίνει τα rankings των δύο τεράτων. Περιέχει 3 διαφορετικές επιστρεφόμενες τιμές, όπου κάθε μία υποδεικνύει τις διαφορετικές πιθανές εκβάσεις της μάχης.

Οι μεταβλητές που περιέχει είναι η row, η οποία υποδεικνύει τη στήλη του τέρατος στο ταμπλό, τη line η οποία υποδεικνύει τη γραμμή του τέρατος στο ταμπλό, τη power η οποία υποδηλώνει το ranking του τέρατος, μία boolean alive η οποία υποδεικνύει τη ζωτική κατάσταση του τέρατος, την ID η οποία υποδεικνύει σε ποιόν παίκτη ανήκει το τέρας και τέλος τις εικόνες της κάρτας του εκάστοτε τέρατος -η επιλογή κάρτας εξαρτάται από το είδος τέρατος και τον παίκτη που την ελέγχει: η ιδιωτική μέθοδος update cards pictures επιτελεί τη λειτουργία εύρεσης των καρτών.

Ο κατασκευαστής της κλάσης δέχεται ορίσματα τη στήλη εκκινήσεως, τη γραμμή εκκινήσεως του τέρατος (η board αποφασίζει ποιά θα είναι αυτά), το ranking του τέρατος, τα τέρατα αποφασίζουν ποιό θα είναι αυτό και το αναγνωριστικό του παίκτη, το οποίο εξαρτάται από τον παίκτη που την αρχικοποιεί -στο board κι αυτό. Επίσης ελέγχει εάν η αρχική τοποθέτηση είναι ικανή ή εάν το αναγνωριστικό είναι έγκυρο.

Περιέχει και observers τύπου isEnemy ή isValidStartingPosition, isValidPosition τα οποία επιτελούν έργο που υπονοείται από τις ονομασίες τους (συνεισφέρουν σε ελέγχους τυπικών παραμέτρων).

Στο package ImmovableObjects περιέχονται τα τέρατα που δεν είναι ικανά να μετακινηθούν ή να επιτεθούν. Οι βασική κλάση αυτών είναι η αφηρημένη Immovable Object η οποία επεκτείνει την GameCharacters και υλοποιεί τις μεθόδους attack και move. Αφού τα στατικά τέρατα δε μπορούν ούτε να μετακινηθούν, αλλά ούτε να επιτεθούν, τα σώματα τους είναι ουσιαστικά κενά -εκτυπώνουν ένα μήνυμα λάθους.

Στο πακέτο ImmovableMonsters βρίσκονται τα τέρατα του παιχνιδιού και αυτά είναι τα Flag και Trap και επεκτείνουν τη κλάση ImmovableObject. Όπως ανέφερα παραπάνω, αυτές οι κλάσεις καθορίζουν το ranking τους -το Flag έχει 0, αφού αιχμαλωτίζεται από όλα τα μετακινήσιμα τέρατα, ενώ το Flag έχει ‘άπειρο’, αφού χάνει μόνο από τον κοντό.

Δε περιέχουν καμία μέθοδο, τα overrides της βασικής τους κλάσης καλύπτει τις ανάγκες τους.

Οι κατασκευαστές τους δέχονται ορίσματα την αρχική διεύθυνση τους στο ταμπλό και το αναγνωριστικό του παίκτη που τα ελέγχει.

Το πακέτο Forbidden\_Zone περιέχει τη κλάση Forbidden\_Zone, η οποία είναι αναπαριστά τις απαγορευμένες ζώνες. Για να μη συγχέεται με άλλα τέρατα έχει αναγνωριστικό -1 και οι τοποθεσίες εγκαταστάσεως τους είναι σταθερές. Βοηθάει στην επιβολή κανόνων.

Φεύγοντας από τα στατικά τέρατα, πηγαίνουμε στο πακέτο MoveableObjects, το οποίο περιέχει όλα τα μετακινήσιμα τέρατα. Η βασική κλάση αυτών είναι η αφηρημένη κλάση MoveableCharacters η οποία επεκτείνει την GameCharacters. Η κλάση αυτή υλοποιεί τις μεθόδους attack και move, αφού για τα μη-ειδικά τέρατα λειτουργούν ομοιόμορφα.

Η μέθοδος move επικαλείται την isValidPosition(), η οποία ελέγχει εάν οι τυπικές παράμετροι δεν αντιστοιχούν σε θέση εκτός του πίνακα ή σε απαγορευμένη ζώνη. Ύστερα επικαλείται τους setters των διευθύνσεων, οι οποίοι δεν διενεργούν ελέγχους, αφού είναι τύπου protected.

Η μέθοδος attack δέχεται ως παράμετρο το αμυνόμενο τέρας του αντιπάλου, οπότε κάνει έλεγχο ισοδυναμίας του τέρατος του αντιπάλου με το null -είναι αναγκαία η ύπαρξη τέρατος για τη μάχη. Μάχη μπορεί να υπάρξει μονάχα μεταξύ τεράτων των αντιπάλων παικτών, συνεπώς πραγματοποιείται και ένας σχετικός έλεγχος. Εάν οι παράμετροι είναι έγκυρες, καλείται η winsBattleAgainst η οποία προσωμοιάζει τη μάχη σύμφωνα με τα rankings και επιστρέφει αντίστοιχο κωδικό.

Εάν η επιστρεπτέα τιμή είναι το 1, τότε ο επιτιθέμενος κέρδισε, εάν η τιμή είναι 2 ο επιτιθέμενος έχασε και εάν είναι 3, τότε αιχμαλωτίστηκαν και τα 2 τέρατα.

Στο πακέτο Regular\_Monsters βρίσκονται τα μη-ειδικά τέρατα και επεκτείνουν τη κλάση MoveableCharacters. Δεν περιέχουν καμία μέθοδο, πλην του κατασκευαστή τους, ο οποίος καθορίζει τα rankings του εκάστοτε τέρατος.

Στο πακέτο SpecialMoveableObjects περιέχεται η αφηρημένη κλάση SpecialMoveableCharacters, η οποία επεκτείνει την MoveableCharacters. Αυτή κάνει override τη μέθοδο attack, επειδή τα ειδικά τέρατα διενεργούν διαφορετικά. Εάν το ειδικό τέρας πολεμήσει τέρας κανονικά, τότε η προσωμοίωσει της μάχης είναι όμοια με αυτήν των μη-ειδικών τεράτων. Οπότε και τα ειδικά τέρατα επικαλούνται την super.attack της SpecialMoveableCharacters εάν δεν επιτίθεται σε τέρας που έχει κάποια ειδική επίδραση.

Στο πακέτο Special\_Characters περιέχονται κλάσεις όλων των ειδικών τεράτων (Dwarf, Scout, Slayer). Οι κλάσεις αυτές επεκτείνουν τη SpecialMoveableCharacters και κάνουν override την attack της. Ο νάνος εάν επιτεθεί σε παγίδα, την αιχμαλωτίζει, διαφορετικά μάχονται κανονικά το τέρας και άρα επικαλείται την attack της SpecialMoveableCharacters. Ύστερα, ελέγχει εάν είναι ζωντανός και αν είναι, τον μετακινεί στη θέση του αντιπάλου εχθρού.

Η κλάση scout κάνει override την attack και τη φέρνει στη μορφή της attack της MoveableCharacters. Παρόλο που δε κινείται σαν τα υπόλοιπα τέρατα, οι κανονισμοί κινήσεως τους επιβάλλονται από τον board.

Φεύγοντας από το πακέτο game\_characters μεταβαίνουμε στο πακέτο Player\_and\_Board, το οποίο περιέχει τις κλάσεις /διασυνδέσεις Board, Player και PlayerInterface.

Το PlayerInterface είμαι εμφανώς μία διασύνδεση που υλοποιείται από τη κλάση player. Περιέχει όλες τις απαραίτητες ενέργειες που μπορεί να επιτελέσει ο παίκτης.

Μέσω της moveCharacter ο παίκτης μετακινεί το τέρας από τη θέση (initialLine, initialRow) στη θέση (nextLine, nextRow) -προφανώς υπάρχουν δημόσιοι observers που ελέγχουν αν οι θέσεις αντιστοιχούν σε έγκυρα τέρατα ή αν οι θέση μετακίνησης επιτρέπεται.

Αν η θέση (nextLine, nextRow) είναι νόμιμη για μετακίνηση από το τέρας, τότε υπάρχουν δύο ενδεχόμενα. Στη θέση αυτή υπάρχει ένα τέρας του αντίπαλου παίκτη, σ’αυτή τη περίπτωση η μέθοδος τα βάζει σε μάχη. Ή η θέση αυτή είναι κενή και επομένως το τέρας μετακινείται ακάθεκτο.

Η μέθοδος αυτή επιστρέφει τον νικητή της μάχης, εάν υπήρξε. Εάν δεν υπήρξε απλά επιστρέφει το τέρας του επιτιθέμενου.

Οι μέθοδοι observers του player είναι η matchContinues(), η οποία ελέγχει εάν ικανοποιείται κάποια συνθήκη τερματισμού του παιχνιδιού, και σε αυτή τη περίπτωση επιστρέφει true.

H canRevive η οποία ελέγχει εάν ο παίκτης μπορεί να αναγεννήσει κάποιο τέρας του, isValidPositionForRevival, η οποία ελέγχει εάν το σημείο που επέλεξε ο παίκτης να αναγεννήσει το τέρας του είναι νόμιμη

Ο κατασκευαστής του Player αποδέχεται μόνο ένα όρισμα, και αυτό είναι το mode. Η μεταβλητή mode, καθορίζει το είδος του παιχνιδιού, δηλαδή εάν τα τέρατα είναι ικανά να μετακινούνται μόνο προς τα μπρος ή εάν οι παίκτες θα παίζουν με 15 τέρατα έναντι των 30. Αν δημιουργηθούν πάνω από 2 αντικείμενα κλάσης, τότε ο κατασκευαστής δεν αρχικοποιεί καμία μεταβλητή και επιστρέφει αμέσως.

Εντός του κατασκευαστή, αν το playerCount == 0, τότε ο παίκτης λαμβάνει το αναγνωριστικό 1 και δημιουργείται ένα αντικείμενο τύπου Board. Εάν playerCount == 1, τότε ο παίκτης λαμβάνει το αναγνωριστικό 2.

Ύστερα, λαμβάνονται τα ζωντανά τέρατα του έκαστου παίκτη και αρχικοποιούνται και οι υπόλοιπες μεταβητές (εντός του κατασκευαστή του Board, τοποθετούνται τα τέρατα των παικτών πάνω στο ταμπλό).

Οι μεταβλητές που περιέχει κάθε παίκτης είναι ιδιωτικές και αυτές είναι η referenceToMonsters, η οποία είναι ένας πίνακας δύο διαστάσεων, όπου ο κορυφαίος πίνακας συκγρατεί ~~μέσω ‘κωδικοποίησης’~~ τα τέρατα που ελέγχει έκαστος παίκτης και ο δεύτερος πίνακας περιέχει το σύνολο των τεράτων που πρέπει να υπάρχουν στο έκαστο mode που επιλέγεται.

Περιέχει μία στατική μεταβλητή τύπου board, η οποία αναπαριστά το ταμπλό του παιχνιδιού και κατ’επέκταση καθορίζει τους κανονισμούς του παιχνιδιού. Περιέχει μία ακέραια μεταβλητή ID, που καθορίζει τον παίκτη. Οι παίκτες παίζουν το ίδιο παιχνίδι και άρα δε χρειάζεται να κρατάνε διαφορετικές υλοποιήσεις του board.

Έχει μία λίστα με περιεχόμενο τύπου GameCharacters (aliveMonsters) που περιλαμβάνει όλα τα τέρατα που ελέγχει και βρίσκονται πάνω στο ταμπλό -είναι ζωντανά.

Περιέχει μία ακόμα λίστα με περιεχόμενα τύπου GameCharacters (capturedMonsters), και περιλαμβάνει όλα τα τέρατα του αντιπάλου που κρατάει αιχμάλωτα.

Έχει έναν πίνακα με αντικείμενα GameCharacters monstersThatCanRevive -ατυχές όνομα, αλλά αν το αλλάξω τώρα θα χρειαστεί να κάνω άλλαγες σε αρκετές γραμμές κώδικα. Ο πίνακας αυτός περιλαμβάνει τα τέρατα του παίκτη που έχουν αναγεννήσει κάποιο αιχμάλωτο τέρας.

To στατικό playersCount είναι μετρητής που μετράει το σύνολο των παικτών και αν playerCount == 0, τότε δε επιτρέπεται να δημιουργηθούν νέα στιγμιότυπου της κλάσης.

Η κλάση Player περιέχει και μερικές βοηθητικές ιδιωτικές μεταβλητές. Ελπίζω να μην απαιτείται εξήγηση, αφού χρησιμοποιούνται εντός της κλάσης για να γίνει περισσότερο ευανάγνωστος ο κώδικας.

Η κλάση Board περιλαμβάνει τις στατικές μεταβλητές gameMode, η οποία συγκρατεί το mode του παιχνιδιού και το στατικό πίνακα piecesOnBoard (8x10), το οποίο είναι προσωμοίωση του ταμπλού και περιέχει όλα τα ζωντανά τέρατα και τις απαγορευμένες ζώνες.

Ο κατασκευαστής της board λαμβάνει τη παράμετρο mode, η οποία καθορίζει το mode του παιχνιδιού. Κατά την εκκίνηση, αρχικοποιείται ο πίνακας και τοποθετούνται τα τέρατα του κάθε παίκτη τυχαίως και προσθέτει και τις απαγορευμένες ζώνες του παιχνιδιού.

Η gameBoard περιέχει μερικές getters όπως η getPlayersAliveMonsters η οποία δέχεται τη λίστα aliveMonsters του player, το αναγνωριστικό του παίχτη και ανάλογα με το αναγνωριστικό προσθέτει τα κατάλληλα τέρατα που έχουν εγκατασταθεί. Πρέπει να χρησιμοποιείται μόνο κατά την εκκίνηση του αγώνα.

Περιέχει και τη getMonster, η οποία αναζητεί και επιστρέφει το αντικείμενο που βρίσκεται στο ταμπλό στη ζητούμενη διεύθυνση.

Η κλάση Board δεν είναι δημόσια και χρησιμοποιείται μόνο από τη κλάση Player. Ο Controller δεν έχει κανένα έλεγχο πάνω της, δεν έχει άμεση πρόσβαση. Στην ουσία καθορίζει και επιβάλλει τους κανονισμούς του παιχνιδιού.

Μερικοί από τους observers της κλάσεις είναι οι:

Η συνάρτηση canMove ελέγχει εάν το τέρας μπορεί να κινηθεί και επιστρέφει true εάν η απάντηση είναι καταφατική.

Η συνάρτηση isEnemy ελέγχει εάν η δοσμένη διεύθυνση αντιστοιχεί σε τέρας του αντιπάλου και επιστρέφει true, εάν αληθεύει.

Η συνάρτηση canMoveAtLeastAMonster δέχεται όρισμα τα ζωντανά τέρατα του επικαλεστή της και ελέγχει εάν τουλάχιστον ένα τέρας του, είναι ικανό να μετακινηθεί -και επιστρέφει true, εάν γίνεται. Χρησιμοποείται από τον Player για να ελέγχει εάν το παιχνίδι λήγει ή συνεχίζεται.

Η συνάρτηση hasSelectedMoveableMonster ελέγχει εάν η δοσμένη διεύθυνση αντιστοιχεί σε τέρας του επικαλεστή παίκτη και εάν το τέρας αυτό είναι ικανό να μετακινηθεί. (γενικά ή προς κάποια διεύθυνση, θα το ανανεώσω για τη φάση Β). Χρησιμοποιείται από τον Controller και τον Player για να ελέγχει εάν το πατημένο κουμπί αποτελεί έγκυρη επιλογή.

Η συνάρτηση isValidPositionForRevival, ελέχει εάν η δοσμένη διεύθυνση αντιστοιχεί σε θέση έγκυρη για αναγέννηση του τέρατος του παίκτη.

Φεύγοντας από τους observers, η κλάση Board περιέχει και τη μέθοδο moveMonster, η οποία μετακινεί το τέρας (δεδομένου ότι είναι έγκυρες οι διευθύνσεις) και εάν υπάρχει αντίπαλο τέρας στο πέρασμα του, του επιτίθεται -οποιοδήποτε τέρας σκοτωθεί, ανανεώνεται στις λίστες των Players και αφαιρείται από το ταμπλό και επιστρέφει το νικητήριο τέρας, αν υπάρχει. Ή αν δεν υπήρξε επίθεση, επιστρέφει το τέρας του επιτιθέμενου -για να ανανεωθεί στον View.

## Η Σχεδίαση και οι Κλάσεις του Πακέτου Controller

Στο πακέτο Controller έχω προσθέσει μονάχα μία κλάση την Controller. Η κλάση είναι δημόσια και εντός της, υπάρχει μία δεύτερη ιδιωτική κλάση, η ButtsonsPressed, η οποία υλοποιεί τη MouseListener και ανιχνεύει τα πατήματα του ποντικιού -θεωρώ πως τέρας επιλέγεται μονάχα όταν πατιέται το αριστερό κλικ.

Η κλάση Controller περιέχει τις μεταβλητές revivalPosition, η οποία είναι ένας πίνακας ακέραιων μεταβλητών και θα κρατάει τη τοποθεσία που ο παίκτης επιθυμεί να αναγεννήσει το νέο του τέρας.

Περιέχει προφανώς μία μεταβλητή τύπου View για να αλληλεπιδρά με το παράθυρο του παιχνιδιού και τέλος περιέχει δύο μεταβλητές τύπου Player, τις attacker και defender, που θα δείχνουν στον επιτιθέμενο και αμυνόμενο έκαστου γύρου.

Ο κατασκευαστής του δεν αναμένει κάποιο όρισμα και εντός αυτού αρχικοποιείται ένα στιγμιότυπο της κλάσης View και ύστερα επικαλείται τη μέθοδο getGamesMode της View και αποθηκεύει το mode του παιχνιδιού σε μία τοπική μεταβλητή, η οποία χρησιμοποιείται για την αρχικοποίηση των δύο αντίπαλων παικτών.

Ύστερα, ανανεώνει το παράθυρο του παιχνιδιού μέσω της μεθόδου nextRound της View.

Όταν η ButtonsPressed ανιχνεύει το πάτημα ενός κουμπιού του ποντικιού, τότε το Source του πατημένου κουμπιού αποθηκεύεται σε μία μεταβλητή τύπου Jbutton. Στην κλάση View κάθε κουμπί έχει ένα αναγνωριστικό του οποίου ο αριστερός αριθμός δηλώνει τη γραμμή στο ταμπλό του παιχνιδιού και το δεύτερο δηλώνει τη στήλη στο ταμπλό.

Αν το πλήκτρο που πατήθηκε είναι το αριστερό, τότε αποθηκεύουμε τη γραμμή και τη στήλη σε δύο ακέραιες τοπικές μεταβλητές.

Ο πρώτος έλεγχος γίνεται μέσω της View και ελέγχει εάν έχει επιλεχτεί ήδη κάποιο τέρας (για μετακίνηση). Εάν έχει επιλεχτεί, τότε παίρνουμε τη διεύθυνση του εν λόγω τέρατος στο ταμπλό και την αποθηκεύουμε σε έναν πίνακα ακεραίων.

Εάν η τοποθεσία που επιλέχτηκε για μετακίνηση δεν είναι νόμιμη, τότε ειδοποιώ τη View μέσω της invalidMove και περιμένει τον παίκτη να επιλέξει νέο προορισμό μετακίνησης.

Αν η τοποθεσία είναι νόμιμη, τότε μετακινούμε το τέρας μέσω της μεθόδου moveMonster της κλάσης Controller.

Εάν δεν έχει επιλεχτεί κάποιο τέρας, τότε ελέγχουμε εάν έχει επιλεχτεί κάποιο νόμιμό τέρας του επιτιθέμενου. Αν δεν έχει επιλεχτεί έγκυρο τέρας, τότε ειδοποιούμε τη View ότι είναι μη-έγκυρο το τέρας μέσω της IllegalMove της View.

Εάν η τοποθεσία μετακίνησης είναι νόμιμη, τότε μέσω του στιγμιοτύπου του επιτιθέμενου αναζητώ τα τετράγωνα του ταμπλού που μπορεί να μετακινηθεί το επιλεγόμενο τέρας, τα αποθηκεύει σε μία λίστα. Οι τοποθεσίες αυτές χρησιμοποιούνται σαν όρισμα στη highlightMoveablePositions της μεθόδου View, για να επισημανθεί στο παράθυρο τις πιθανές επιλογές μετακίνησης.

Η μέθοδος moveMonster του Controller μετακινεί στην ουσία το τέρας του επιτιθέμενου.

Αρχικά, μέσω του στιγμιοτύπου του επιτιθέμενου, μετακινούμε το τέρας μέσω της moveCharacter: η εν λόγω μέθοδος επιστρέφει το νικητήριο τέρας, σε περίπτωση μάχης ή το επιτιθέμενο τέρας, σε περίπτωση που μετακινείται κανονικά χωρίς τέρας.

Αν το επιτιθέμενο τέρας, αιχμαλώτισε κάποιο τέρας του αντιπάλου, τότε ανανέωσε το πάνελ του γύρου.

Έπειτα, πραγματοποιείται έλεγχος τερματισμού του παιχνιδιού. Εάν το παιχνίδι τελείωσε, τότε μέσω της endGame του View τερματίζω το παιχνίδι.

Διαφορετικά, πραγματοποιώ έλεγχο ικανότητας αναγέννησης. Ο έλεγχος αυτός γίνεται μέσω των στιγμιοτύπων επιτιθέμενου και του isThisPlayersMonster, μέθοδος που ελέγχει εάν το νικητήριο τέρας ανήκει στον επιτιθέμενο και μέσω της μεθόδου canRevive, η οποία ελέγχει εάν το τέρας μπορεί πράγματι να αναγεννήσει.

Αν μπορεί, τότε μέσω της selectMonsterToRevive της View, επιλέγω τέρας για αναγέννηση και τοποθεσία που επιθυμώ να το τοποθετήσω.

Ο έλεγχος εγκυρότητας τοποθέτησης πραγματοποιείται μέσω του στιγμιοτύπου του επιτιθέμενου και της μεθόδου isValidPositionForRevival.

Εάν η τοποθεσία είναι έγκυρη, τότε μέσω του στιγμιοτύπου του επιτιθέμενου και της reviveMonster, αναγεννώ το τέρας στον πυρήνα του παιχνιδιού.

Έπειτα, ανανεώνω το ταμπλό μέσω της updateBoard του View, αλλάζω τον γύρο του παιχνιδιού, και ύστερα αλλάζω και το πάνελ του γύρου του παραθύρου μέσω της nextRound της View.

Η μέθοδος isValidMovingPosition του Controller ελέγχει εάν η ζητούμενη τοποθεσία είναι έγκυρη για μετακίνηση. Αυτό γίνεται, μέσω του στιγμιοτύπου του επιτιθέμενου και της μεθόδου canMove.

Η μέθοδος isValidMonster του Controller ελέγχει εάν η επιλεγόμενη τοποθεσία αντιστοιχεί σε κάποιο τέρας του επιτιθέμενου. Αυτό γίνεται, μέσω του στιγμιοτύπου του επιτιθέμενου και της μεθόδου isMovingMonster.

Η μέθοδος attackerHasCapturedMonster ελέγχει εάν το τέρας του επιτιθέμενου έχει αιχμαλωτίσει κάποιο αντίπαλο τέρας. Ο έλεγχος αυτός γίνεται μέσω της GameCharacters και του στιγμιοτύπου του επιτιθέμενου συγκρίνοντας τα αναγνωριστικά των δύο (δεν είναι σωστός ο έλεγχος, επειδή αν δεν υπάρξει μάχη θα επιστραφεί δείκτης προς το τέρας του επιτιθέμενου). Εάν το τέρας υπάρχει και τα αναγνωριστικά του τέρατος και του επιτιθέμενου είναι ισοδύναμα, τότε εμφανώς το τέρας του επιτιθέμενου μετακινήθηκε επιτυχώς.

Η μέθοδος nextRound του Controller αλλάζει τον γύρω. Αναθέτει τον δείκτη του επιτιθέμενου στον αμυνόμενο και το αντίστροφο.

## Η Σχεδίαση και οι Κλάσεις του Πακέτου View

Το πακέτο View περιέχει τη διασύνδεση ViewInterface και τη κλάση View.

Η διασύνδεση ViewInterface περιέχει τις μεθόδου nextRound, η οποία δέχεται ως ορίσματα τα τέρατα του επιτιθέμενου, τα αιχμαλωτισμένα τέρατα του επιτιθέμενου και τα τέρατα του αμυνόμενου. Η μέθοδος αυτή κρύβει τις κάρτες του αμυνόμενου, προβάλλει τις κάρτες του αντιπάλου και ανανεώνει το πάνελ του γύρου και τα τέρατα που αιχμαλωτεί ο επιτιθέμενος.

Η μέθοδος getAttackersLocation επιστρέφει τη τοποθεσία του τέρατος του επιτιθέμενου που έχει επιλεχτεί για μετακίνηση. Η τοποθεσία αποθηκεύεται σε έναν πίνακα ακεραίων και τον επιστρέφει.

Η μέθοδος getGamesMode ερωτεί τον παίκτη τι είδος παιχνιδιού θέλει να παίξει και το επιστρέφει. Ο ακέραιος που επιστρέφεται αποτελεί ΄κώδικα’. Χρησιμοποιείται μόνο κατά την εκκίνηση του παιχνιδιού.

Η μέθοδος highlightMoveablePositions δέχεται μία λίστα ακέραιων τετραγώνων που επιτρέπεται να μετακινηθεί το επιλεγόμενο τέρας και προβάλλει το μονοπάτι στο παράθυρο.

Η μέθοδος monsterHasBeenSelected ελέγχει εάν ένα τέρας έχει επιλεγεί και επιστρέφει true αν ισχύει.

Η μέθοδος invalidMonster ειδοποιεί το παράθυρο ότι η επιλεγόμενη διεύθυνση δεν αντιστοιχεί σε έγκυρο τέρας του επιτιθέμενου και επαναφέρει τις τιμές των μεταβλητών στις προεπιλεγόμενες.

Η μέθοδος updatePanel δέχεται ως όρισμα μία λίστα με αντικείμενα GameCharacters που υποδεικνύουν τα τέρατα που κρατάει αιχμάλωτα ο παίκτης και ανανεώνει το πάνελ αιχμαλώτων τέρατος επιτιθέμενου.

Η μέθοδος selectMonsterToRevive δέχεται σαν όρισμα μία λίστα αντικειμένων τύπου GameCharacters που υποδεικνύουν τα τέρατα που κρατούνται αιχμάλωτα από τον αμυνόμενο, έναν πίνακα ακεραίων που κρατάνε τη τοποθεσία που επιθυμεί ο επιτιθέμενος να τοποθετήσει το αναγεννημένο τέρας.

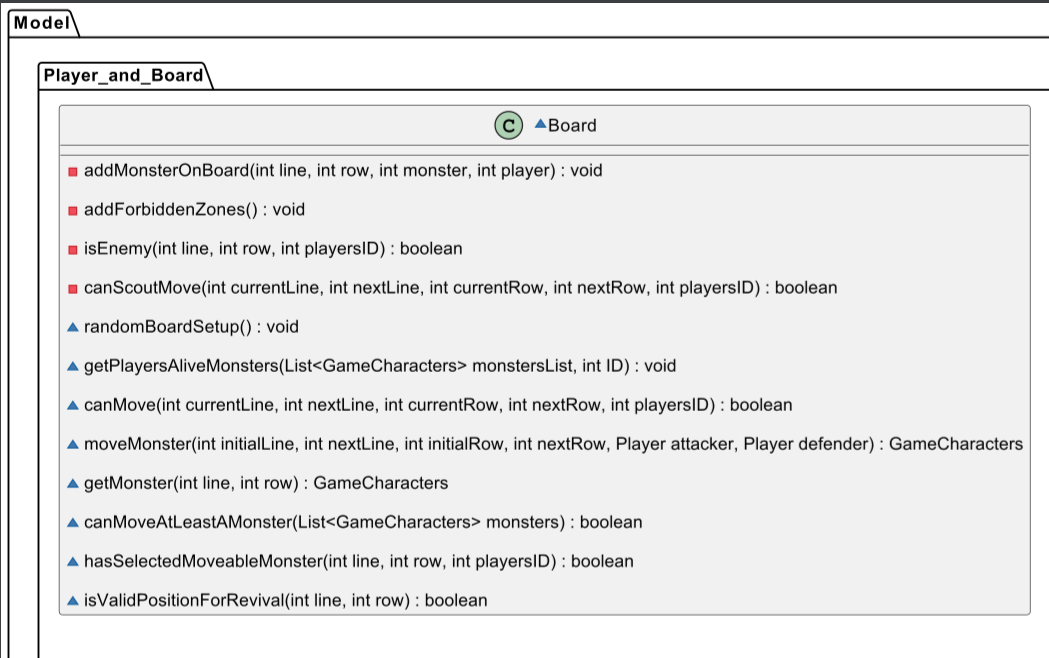
Η μέθοδος αυτή ερωτεί τον παίκτη το τέρας που θέλει να αναγεννήσει, τη τοποθεσία που θέλει να το τοποθετήσει και επιστρέφει σε μορφή int το δείκτη του τέρατος που θέλει να αναγεννήσει (θα το αλλάξω στη φάση Β, θα επιστρέφει το τέρας που θέλει να αφαιρέσει από τη λίστα και όχι ένα δείκτη τύπου int).

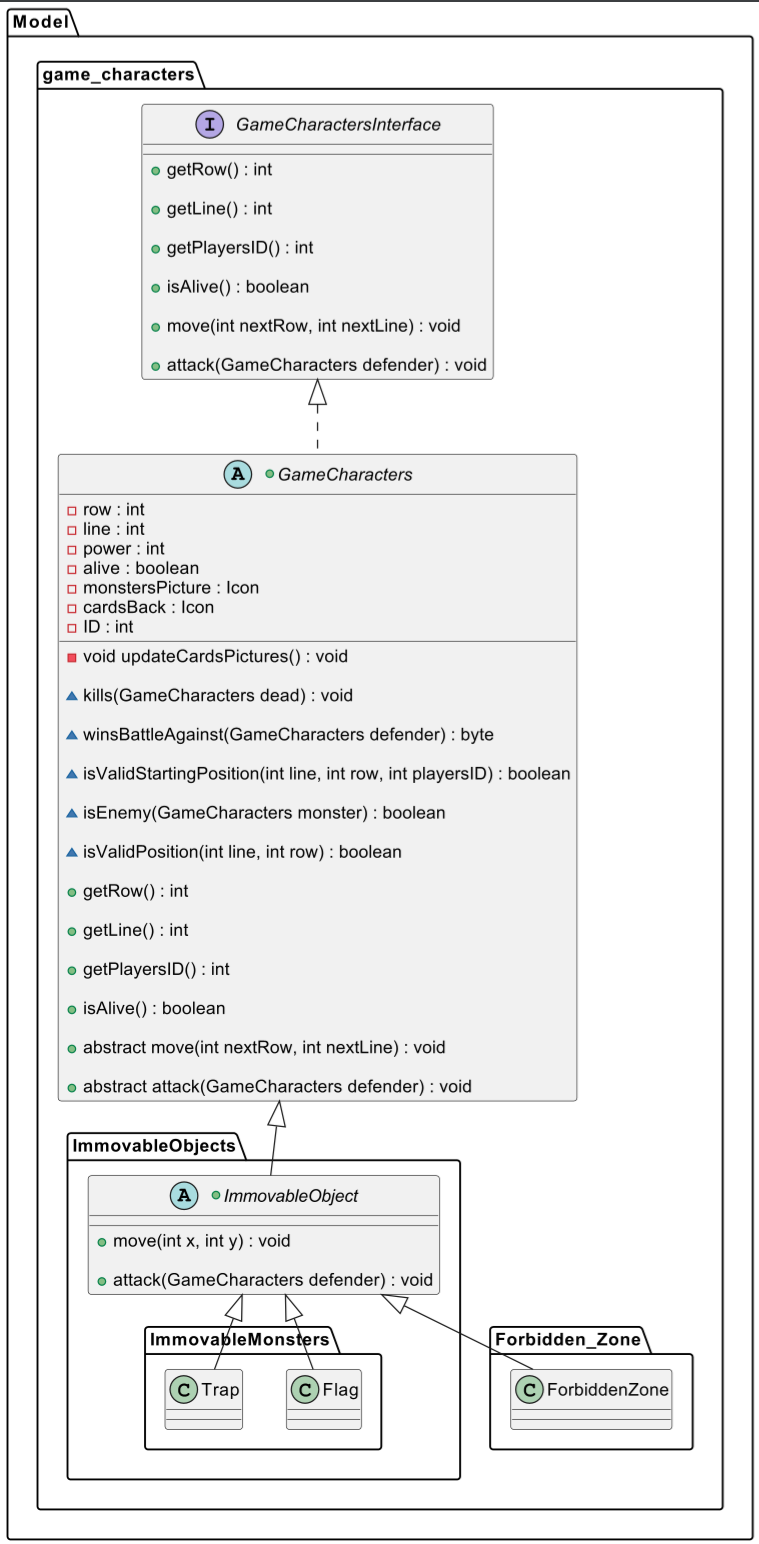
Η μέθοδος endGame απλά θα τερματίζει το παιχνίδι.

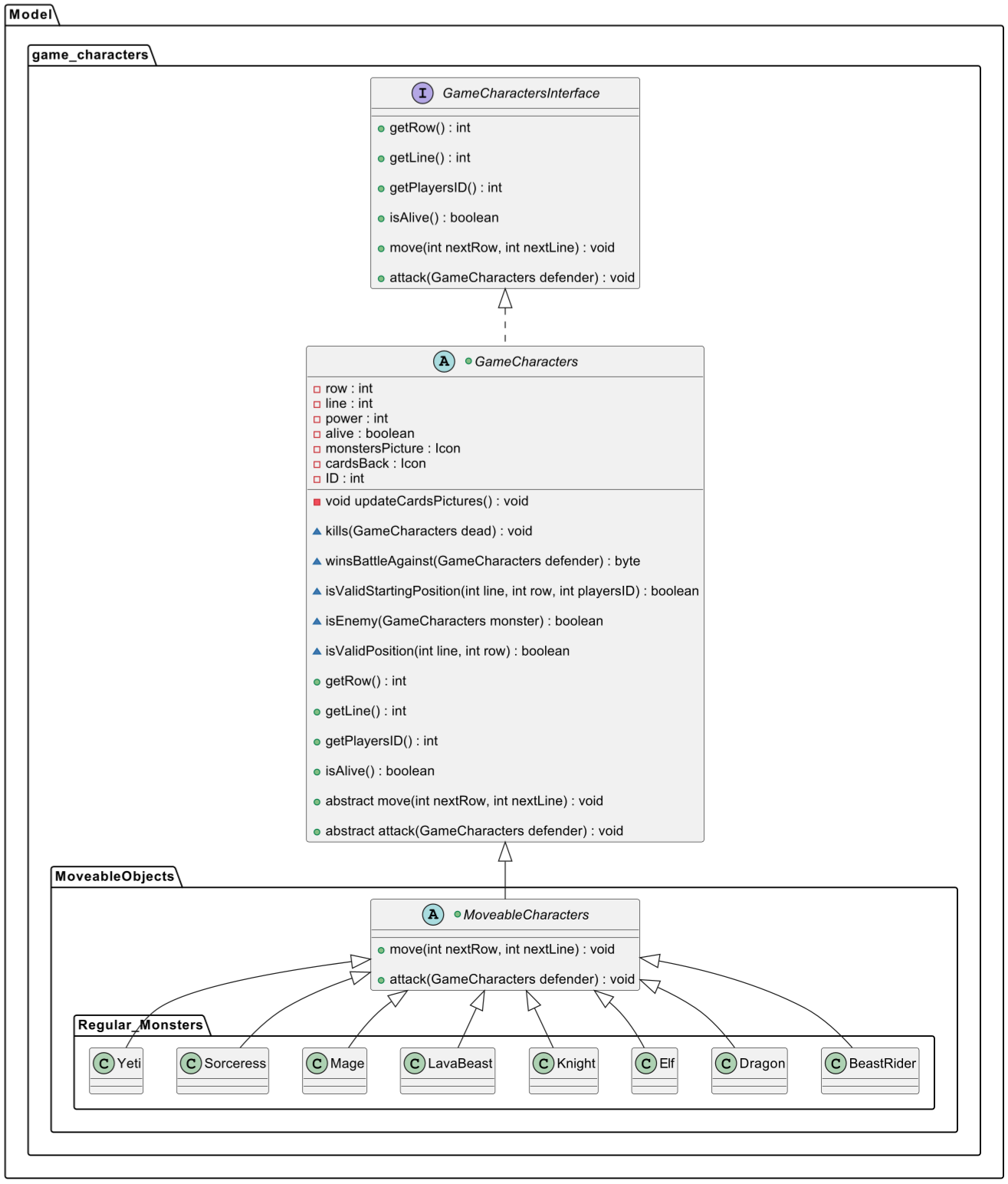
H View θα υλοποιεί αυτή τη διασύνδεση, ωστόσο δεν έχω επιχειρήσει να τη γράψω ακόμα.

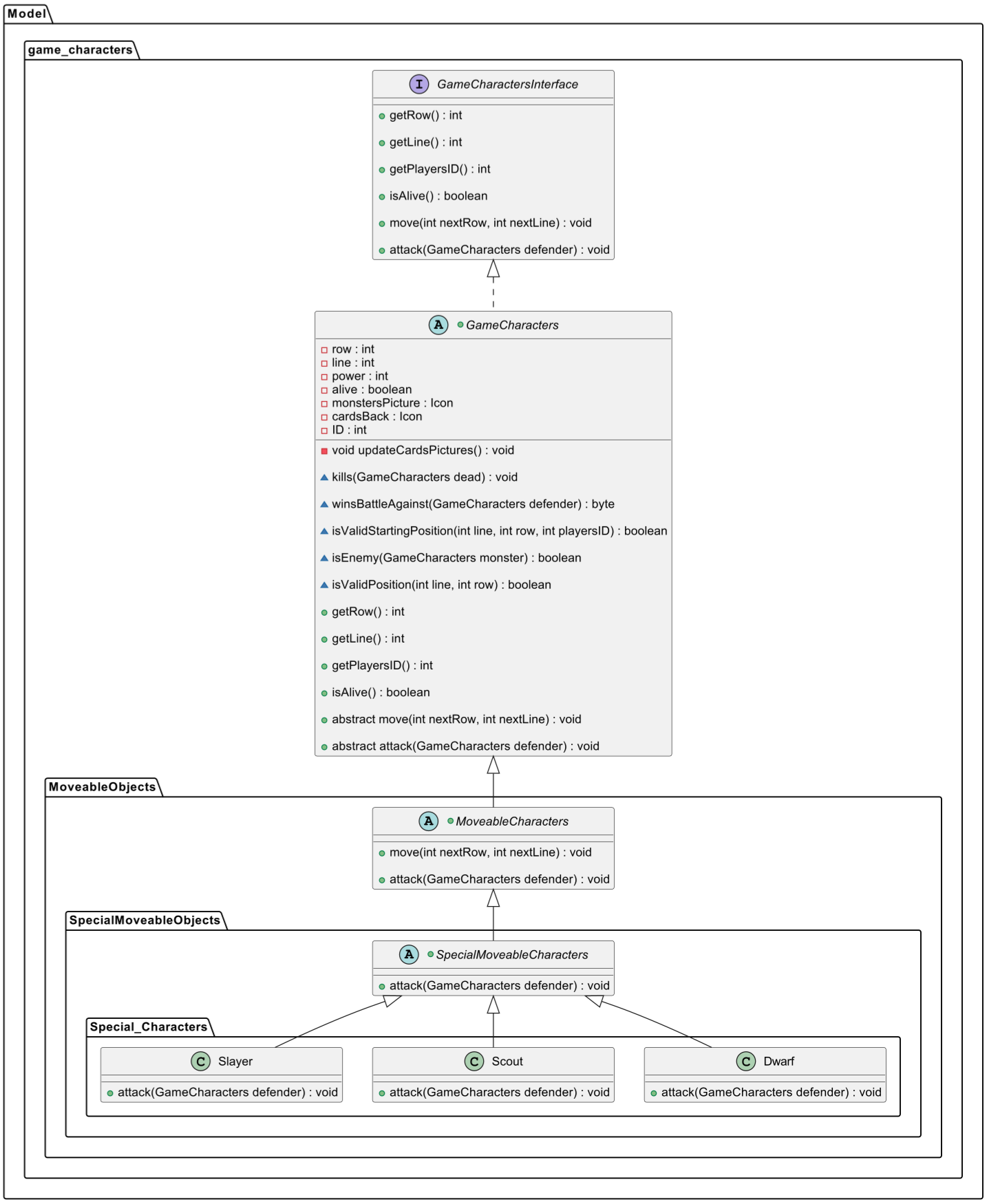
## Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML

## 











ΕΠΑΝΑΛΑΜΒΑΝΩ ΤΗ ΛΕΙΤΟΥΡΓΙΑ ΤΗΣ ΚΛΑΣΗΣ CONTROLLER, ΑΛΛΑ ΑΝΑΛΥΩ ΠΕΡΑΙΤΕΡΩ ΤΗ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕΤΑΞΥ ΤΩΝ ΚΛΑΣΕΩΝ.

Ο κατασκευαστής του δεν αναμένει κάποιο όρισμα και εντός αυτού αρχικοποιείται ένα στιγμιότυπο της κλάσης View και ύστερα επικαλείται τη μέθοδο getGamesMode της View και αποθηκεύει το mode του παιχνιδιού σε μία τοπική μεταβλητή, η οποία χρησιμοποιείται για την αρχικοποίηση των δύο αντίπαλων παικτών. Κατά την αρχικοποίηση των παικτών, εάν δεν υπάρχει παίκτης, δημιουργείτε ένας με αναγνωριστικό 1 και αρχικοποιεί τη στατική μεταβλητή Board (gameBoard) που περιέχει, χρησιμοποιώντας τη δική του τυπική μεταβλητή mode.

Ύστερα, μέσω της στατική μεταβλητής gameBoard επικαλείται τη μέθοδο getPlayersAliveMonsters, η οποία προσθέτει στη λίστα aliveMonsters τα ζωντανά τέρατα του παίκτη.

Ύστερα, ανανεώνει το παράθυρο του παιχνιδιού μέσω της μεθόδου nextRound της View.

Όταν η ButtonsPressed ανιχνεύει το πάτημα ενός κουμπιού του ποντικιού, τότε το Source του πατημένου κουμπιού αποθηκεύεται σε μία μεταβλητή τύπου Jbutton. Στην κλάση View κάθε κουμπί έχει ένα αναγνωριστικό του οποίου ο αριστερός αριθμός δηλώνει τη γραμμή στο ταμπλό του παιχνιδιού και το δεύτερο δηλώνει τη στήλη στο ταμπλό.

Αν το πλήκτρο που πατήθηκε είναι το αριστερό, τότε αποθηκεύουμε τη γραμμή και τη στήλη σε δύο ακέραιες τοπικές μεταβλητές.

Ο πρώτος έλεγχος γίνεται μέσω της View και ελέγχει εάν έχει επιλεχτεί ήδη κάποιο τέρας (για μετακίνηση). Εάν έχει επιλεχτεί, τότε παίρνουμε τη διεύθυνση του εν λόγω τέρατος στο ταμπλό και την αποθηκεύουμε σε έναν πίνακα ακεραίων.

Εάν η τοποθεσία που επιλέχτηκε για μετακίνηση δεν είναι νόμιμη, τότε ειδοποιώ τη View μέσω της μεθόδου της invalidMove και περιμένει τον παίκτη να επιλέξει νέο προορισμό μετακίνησης.

Αν η τοποθεσία είναι νόμιμη, τότε μετακινούμε το τέρας μέσω της μεθόδου moveMonster της κλάσης Controller.

Εάν δεν έχει επιλεχτεί κάποιο τέρας, τότε ελέγχουμε εάν έχει επιλεχτεί κάποιο νόμιμό τέρας του επιτιθέμενου. Αν δεν έχει επιλεχτεί έγκυρο τέρας, τότε ειδοποιούμε τη View ότι είναι μη-έγκυρο το τέρας μέσω της IllegalMove της View.

Εάν η τοποθεσία μετακίνησης είναι νόμιμη, τότε μέσω του στιγμιοτύπου του επιτιθέμενου αναζητώ τα τετράγωνα του ταμπλού που μπορεί να μετακινηθεί το επιλεγόμενο τέρας, και τα αποθηκεύει σε μία λίστα.

Η getMoveablePath επικαλείται τη getMonster της Board για να λάβει το τέρας (έπειτα από έλεγχο εγκυρότητας τέρατος, μέσω της Board ξανά). Έπειτα, επικαλείται τη canMove της Board για να εξετάσει εάν είναι νόμιμες συγκεκριμένες μετακινήσεις, τις αποθηκεύει σε μία λίστα τύπου Integer και την επιστρέφει στον Controller.

Οι τοποθεσίες αυτές χρησιμοποιούνται σαν όρισμα στη highlightMoveablePositions της μεθόδου View, για να επισημανθεί στο παράθυρο τις πιθανές επιλογές μετακίνησης.

Η μέθοδος moveMonster του Controller μετακινεί στην ουσία το τέρας του επιτιθέμενου.

Αρχικά, μέσω του στιγμιοτύπου του επιτιθέμενου, μετακινούμε το τέρας μέσω της moveCharacter: η εν λόγω μέθοδος επικαλείται τη μέθοδο hasSelectedMoveableMonster της Board, για να εξετάσει αν είναι έγκυρο το δοσμένο τέρας, ύστερα επικαλείται την moveMonster της Board. Εάν υπάρξει μάχη, τότε το Board εκκινεί τη μάχη μέσω της attack της GameCharacters. Αν πεθάνει κάποιο τέρας παίκτη, τότε η Board επικαλείται τη lostMonster της κλάσης Player, για να ανανεώσει τα στοιχεία του πυρήνα. Στο τέλος, επιστρέφει το νικητήριο τέρας, σε περίπτωση μάχης ή το επιτιθέμενο τέρας, σε περίπτωση που μετακινείται κανονικά χωρίς τέρας.

Έπειτα, πραγματοποιείται έλεγχος τερματισμού του παιχνιδιού. Εάν το παιχνίδι τελείωσε, τότε μέσω της endGame του View τερματίζω το παιχνίδι.

Διαφορετικά, πραγματοποιώ έλεγχο ικανότητας αναγέννησης. Ο έλεγχος αυτός γίνεται μέσω των στιγμιοτύπων επιτιθέμενου και του isThisPlayersMonster, μέθοδος που ελέγχει εάν το νικητήριο τέρας ανήκει στον επιτιθέμενο και μέσω της μεθόδου canRevive, η οποία ελέγχει εάν το τέρας μπορεί πράγματι να αναγεννήσει.

Αν μπορεί, τότε μέσω της selectMonsterToRevive της View, επιλέγω τέρας για αναγέννηση και τοποθεσία που επιθυμώ να το τοποθετήσω.

Ο έλεγχος εγκυρότητας τοποθέτησης πραγματοποιείται μέσω του στιγμιοτύπου του επιτιθέμενου και της μεθόδου isValidPositionForRevival, η οποία επικαλείται την isValidPositionForRevival της Board.

Εάν η τοποθεσία είναι έγκυρη, τότε μέσω του στιγμιοτύπου του επιτιθέμενου και της reviveMonster, αναγεννώ το τέρας στον πυρήνα του παιχνιδιού.

Έπειτα, ανανεώνω το ταμπλό μέσω της updateBoard του View, αλλάζω τον γύρο του παιχνιδιού, και ύστερα αλλάζω και το πάνελ του γύρου του παραθύρου μέσω της nextRound της View.

Η μέθοδος isValidMovingPosition του Controller ελέγχει εάν η ζητούμενη τοποθεσία είναι έγκυρη για μετακίνηση. Αυτό γίνεται, μέσω του στιγμιοτύπου του επιτιθέμενου και της μεθόδου canMove, η οποία με τη σειρά της επικαλείται την canMove της Board. Η canMove της Board, χρησιμοποιεί τη μέθοδο getPlayersID του GameCharacters για να πραγματοποιήσει έλεγχο.

Η μέθοδος isValidMonster του Controller ελέγχει εάν η επιλεγόμενη τοποθεσία αντιστοιχεί σε κάποιο τέρας του επιτιθέμενου. Αυτό γίνεται, μέσω του στιγμιοτύπου του επιτιθέμενου και της μεθόδου isMovingMonster.

Η μέθοδος attackerHasCapturedMonster ελέγχει εάν το τέρας του επιτιθέμενου έχει αιχμαλωτίσει κάποιο αντίπαλο τέρας. Ο έλεγχος αυτός γίνεται μέσω της GameCharacters και του στιγμιοτύπου του επιτιθέμενου συγκρίνοντας τα αναγνωριστικά των δύο (δεν είναι σωστός ο έλεγχος, επειδή αν δεν υπάρξει μάχη θα επιστραφεί δείκτης προς το τέρας του επιτιθέμενου). Εάν το τέρας υπάρχει και τα αναγνωριστικά του τέρατος και του επιτιθέμενου είναι ισοδύναμα, τότε εμφανώς το τέρας του επιτιθέμενου μετακινήθηκε επιτυχώς.