



OBJECT ORIENTED
PROGRAMMING HY-252
PROJECT 2022-2023



Phase 2

Δαμιανός Καρβούνης

CSD4897

6/12/2022

Το Stratego είναι ένα επιτραπέζιο παιχνίδι στρατηγικής για δύο παίκτες σε ένα ταμπλό τετράγωνων 8×10. Κάθε παίκτης ελέγχει 30 κομμάτια που αντιπροσωπεύουν μεμονωμένες τάξεις αξιωματικών και στρατιωτών σε έναν στρατό. Τα κομμάτια είναι εμπλευσμένα από τα βιβλία ice and fire. Ο στόχος του παιχνιδιού είναι να βρείτε και να συλλάβετε τη Σημαία του αντιπάλου ή να συλλάβετε τόσα πολλά κομμάτια του εχθρού που ο αντίπαλος δεν μπορεί να κάνει περαιτέρω κινήσεις.

Περιεχόμενα

1. Εισαγωγή
2. Package model
3. Package controller
4. Package viewer
5. Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML
6. Λειτουργικότητα
7. Συμπεράσματα

1.Εισαγωγή

Γιατί επιλέξαμε το μοντέλο MVC

Το πρόγραμμα χρησιμοποιεί το αρχιτεκτονικό μοτίβο λογισμικού **Model-view-controller (MVC)** που χρησιμοποιείται συνήθως για την ανάπτυξη διεπαφών χρήστη που διαιρούν τη σχετική λογική του προγράμματος σε τρία διασυνδεδεμένα στοιχεία. Αυτά είναι:

- 1) Το **Model**, που περιεχέει την λογική του παιχνιδιού, τους κανόνες και οτιδήποτε διαδικαστικό έχει να κάνει σχέση με το παιχνίδι
- 2) Το **Viewer** που είναι η γραφική αναπαράσταση του παιχνιδιού δηλαδή ότι βλέπει ο χρήστης
- 3) Το **Controller** που χρησιμοποιεί την λογική του model σαν είσοδο και αφού επεξεργαστεί τα δεδομένα, τα εμφανίζει σαν έξοδο μέσω του viewer, δηλαδή είναι ο συνδετικός κρίκος.

2.Package model

Σε αυτό το package περιεχετε η λογική και ο τρόπος λειτουργίας του προγράμματος. Οι βασικές κλάσεις είναι η **piece** και οι **συγγενικές της κλάσεις**.

Piece

Βασικά elements της abstract κλάσης είναι τα **piecerank**, **name**, **owner**. Η μεταβλητή piecerank τύπου int εμπεριέχει τον βαθμό του πιονιού, η μεταβλητή name τύπου String εμπεριέχει τον τύπο του πιονιού και η μεταβλητή owner τύπου player εμπεριέχει τον ιδιοκτήτη του πιονιού. Όλες οι μεταβλητές έχουν setters και getters εκτός της piecerank που δεν έχει setter.

Στην συνέχεια η piece χωρίζεται σε 2 υπό-κλασεις την **movable piece** που χρησιμοποιείται για τα κινούμενα πιόνια (η διάφορα είναι ότι έχει setter) και την **immovable piece** που είναι ίδια με την piece αλλά κάνει initialize τα ranks ως 0.

Απο την movable piece συγγενευουν οι υποκλασεις slayer, scout, dwarf, elf, beast/yeti, beastrider, sorceress, knight, mage, dragon.

Απο την immovable piece συγγενευουν οι υποκλασεις trap και flag. Το καθε ενα με το αναλογο constructor.

piece		
m	piece()	
f	name	String
f	piecerank	int
f	owner	player
m	setplayer(player)	void
m	getname()	String
m	getpiecerank()	int
m	getplayer()	player
m	setname(String)	void

3.Package controller

Αυτό το package είναι ο συνδετικός κρίκος μεταξύ του model και του viewer χρησιμοποιώντας τα inputs του χρήστη για να διαλέξει αναλόγως τι από το model και να το στείλει στο viewer για να το δείξει στον χρήστη μέσω της γραφικής διεπαφής. Οι βασικές κλάσεις είναι οι board, controllerjv και player

Board

Βασικά elements της κλάσης είναι το **board1**. Το board1 είναι array τύπου piece δηλαδή piece []. Έχει setters και getters για τα pieces που εγγράφονται στον πίνακα. Επίσης έχει getters σε δοσμένη θέση για piece name, piece owner και piece rank.

Επίσης υπάρχουν άλλες 2 συναρτήσεις η **nullpiece** που μηδενίζει τα περιεχόμενα του κουτιού με index i και η **boardinit** που κάνει initialize τον πίνακα αναλόγως με αν το gamemode είναι reduced η όχι. Τέλος ο constructor της board.

board	
m	board(player, player, boolean)
f	board1 piece []
m	nullpiece(int) void
m	getpiecerank(int) int
m	getname(int) String
m	boardinit(piece[], player, player, boolean) void
m	getpiece(int) piece
m	setpiece(int, piece) void
m	getplayer(int) player

Controllerjv

Βασικά elements της κλάσης είναι το **player1**, **player2**, **currentplayer**, **board2**, **turncounter**, **reduced army** και **gamestatus**. Το **gamestatus** έχει setters και getters, υπάρχει getter για την **currentplayer** που μπορεί και να επιστρέψει την αντίπαλη μεριά εκτός από την μεριά που παίζει (**getcurrentplayer**) και για το κάθε movable piece υπάρχουν getters που επιστρέφουν, επιστρέφει το στρινγκ τέλους (**getstatusstring**) και ένα getter που επιστρέφει την πιθανότητα νίκης (**winchance**). Στην συνέχεια υπάρχουν άλλες 2 συναρτήσεις, η **turnchange** που κάνει swap τον current player με τον άλλο και η **battle** που χρησιμοποιείται από την **viewerjv** για να δει ποιος κερδίζει τις μάχες επιστρέφοντας true για την νίκη και false για την ήττα. Τέλος ο constructor του **controllerjv**.

controllerjv		
controllerjv (boolean)		
player1	player	
currentplayer	player	
player2	player	
board2	board	
reducedarmy	boolean	
gamestatus	boolean	
turncounter	int	
getgamestatus ()	boolean	
getdeadyetis_beasts ()	int	
getdeadbeastriders ()	int	
getdeaddragons ()	int	
getcurrentplayer (boolean)	player	
battle (int, int)	boolean	
getdeaddwarves ()	int	
getdeadknights ()	int	
turnchange ()	void	
getdeadelves ()	int	
getdeadslayers ()	int	
getdeadsorceress ()	int	
setgamestatus (boolean)	void	
getstatusstring ()	String	
getdeadimages ()	int	
winchance ()	int	
getdeadscouts ()	int	

Player

Βασικά elements της κλάσης είναι το **status**, **totalattacks** και το **attackswon**. Το **status** έχει setters και getters και υπάρχει μια βοηθητική συνάρτηση που υπολογίζει το στατιστικό επίθεσης (**getattackstat**). Τέλος ο constructor του **player**.

player		
player(boolean)		
totalattacks	int	
status	boolean	
attackswon	int	
setstatus (boolean)	void	
getstatus ()	boolean	
getattackstat ()	int	

4.Package viewer

Σε αυτο το package περιεχετε η **γραφικη αναπαρασταση του παιχνιδιου** (με τη χρηση JFrame) δηλαδη το παραθυρο ,τα κουμπια, οι κινήσεις των στρατιωτων τα στατιστικά κλπ

Start menu

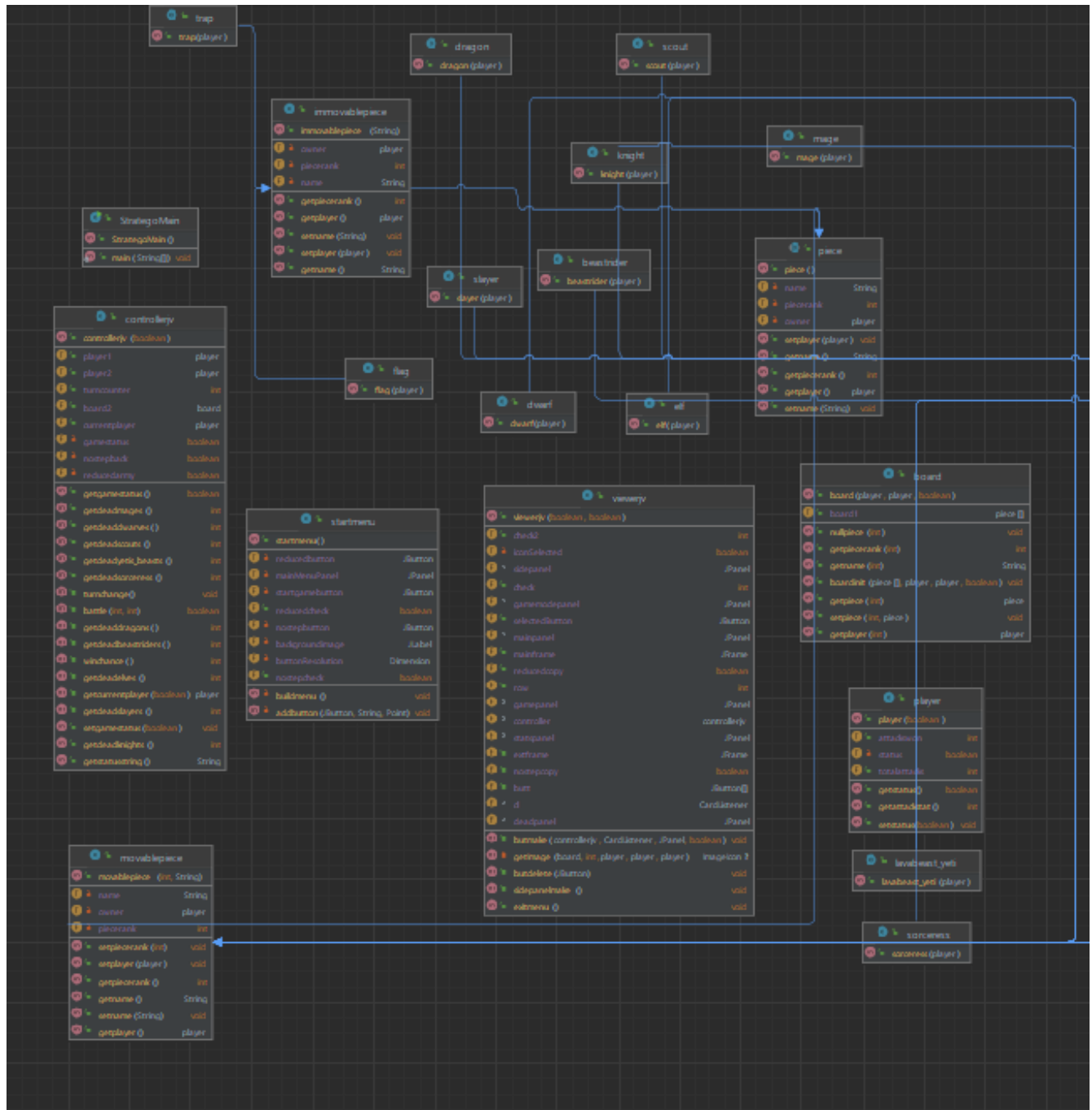
Βασικά elements της κλάσης που δημιουργεί το αρχικό μενού επιλόγων είναι τα **mainMenu**, **startgamebutton**, **nostepbutton**, **reducedbutton**, **backgroundimage**, **buttonresolution**, **nostepcheck** και **reducedcheck**. Υπάρχουν 2 κυρίες συναρτήσεις η **addbutton** που προσθέτει JButtons σε ένα JPanel και η **buildmenu** που φτιάχνει το βασικό frame και προσθέτει την εικόνα, τα κουμπιά κλπ πάνω της. Τέλος ο constructor της κλάσης.

Viewerjv

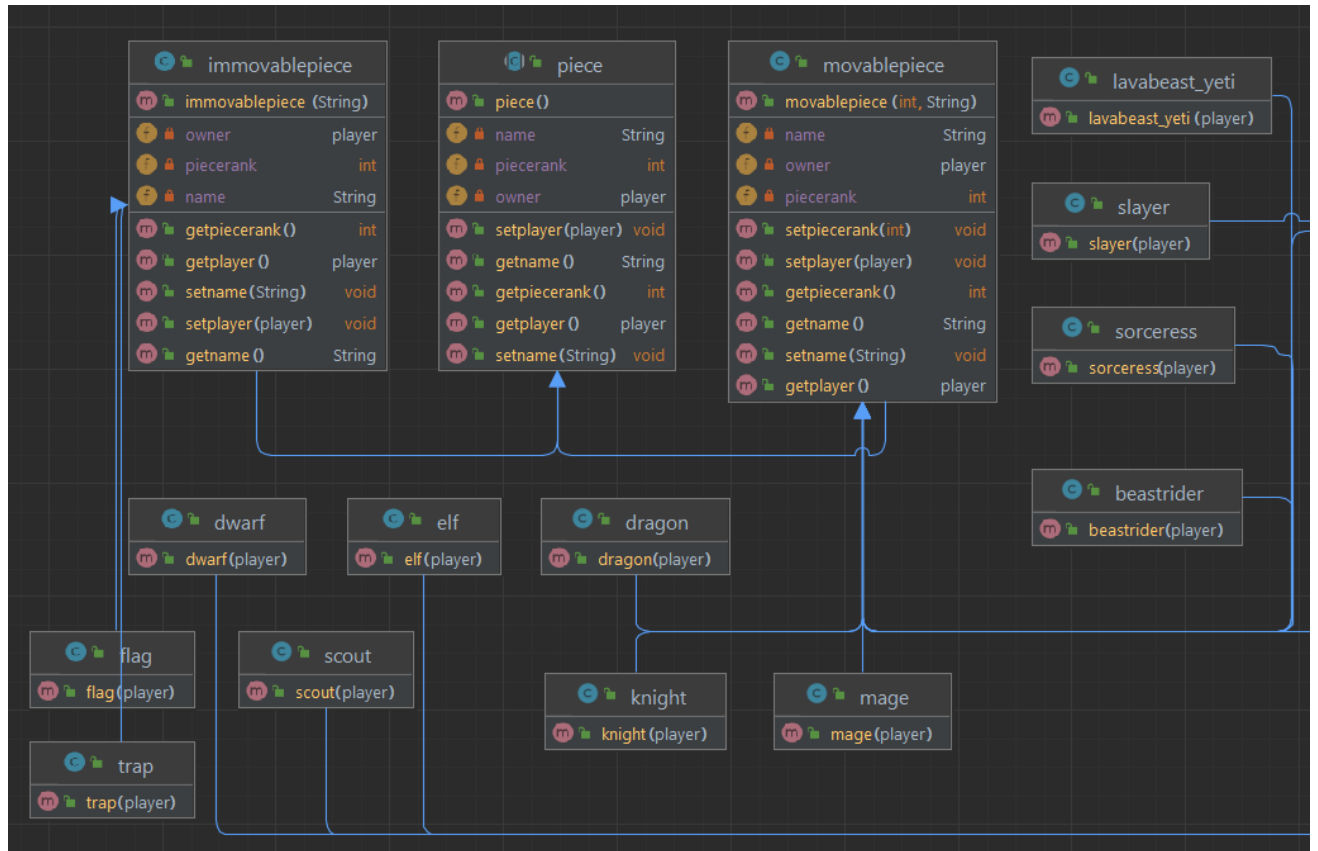
Βασικά elements της κλάσης είναι τα (boolean) **iconSelected**, **nostepcopy**, **reducedcopy**, (JButton) **selectedButton**, (JButton []) **butt**, (controllerjv) **controller**, (Cardlistener) **cl**, (JFrame) **mainframe**, **extframe**, (JPanel) **sidepanel**, **mainpanel**, **gamemodepanel**, **gamepanel**, **statspanel**, **deadpanel**, (int) **check**, **check2**, **row**. Στη συνέχεια η **getimage** επιστρέφει το ζητειντο Icon αναλόγως με το ποιος είναι ο παίκτης, το **sidepanelmake** φτιάχνει το side panel και ελέγχει τις τιμές πάνω σε αυτό, το **exitmenu** εμφανίζεται όταν το παιχνίδι τερματίζει και μπορείς είτε να το ξαναξεκινήσεις η να το τερματίσεις, το **butmake** και το **butdelete** είναι συναρτήσεις για χρήση επί των κουμπιών όπου αντιστοιχως φτιαχνουν/σβηνουν κουμπιά από τον πίνακα και από την μνήμη κουμπιών και το **mouseclicked** ελέγχει την κίνηση των κουμπιών με αρκετές if. Τέλος η κλάση έχει τον constructor της.

5. Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML

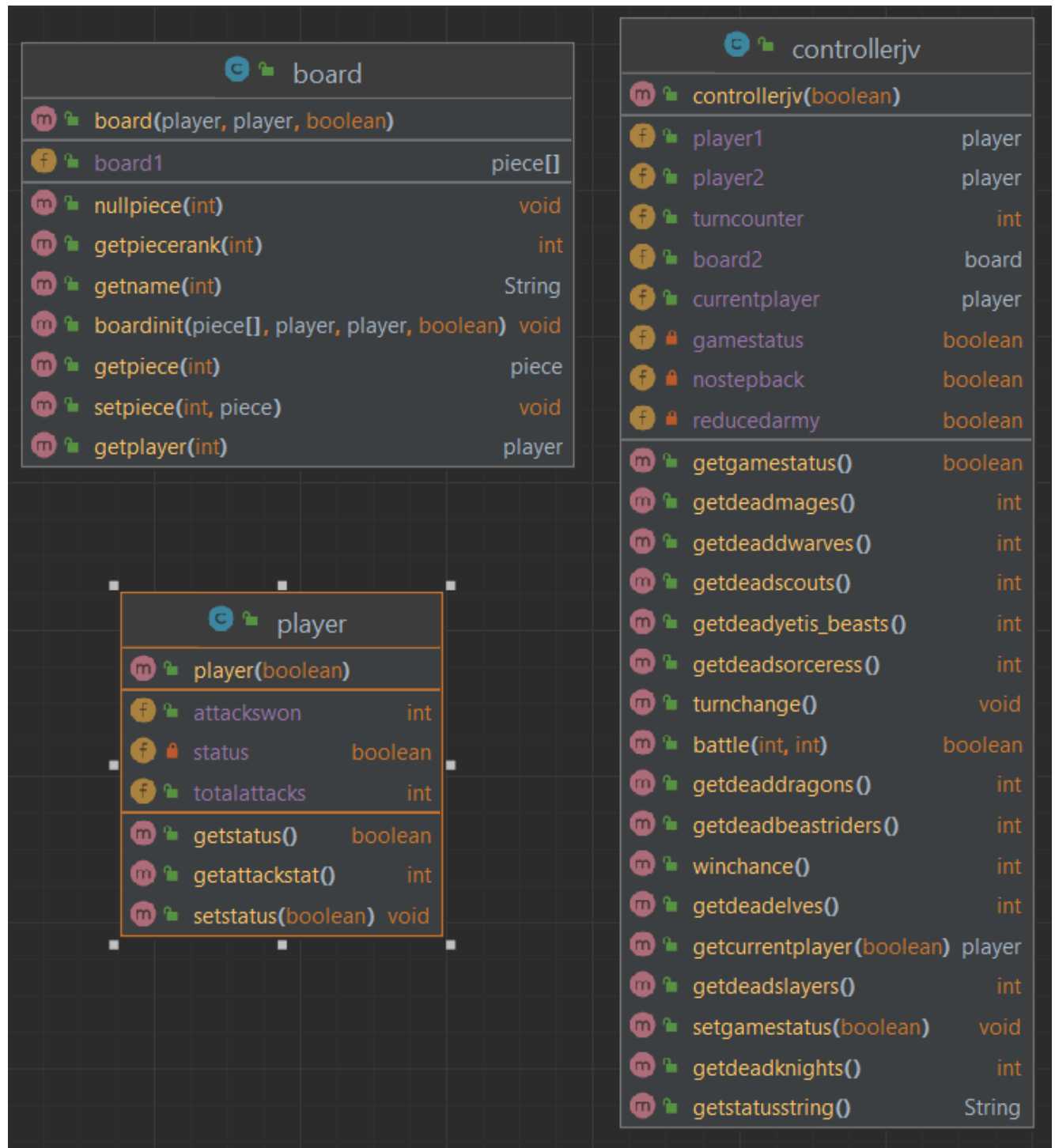
Γενικό UML διαγραμμα



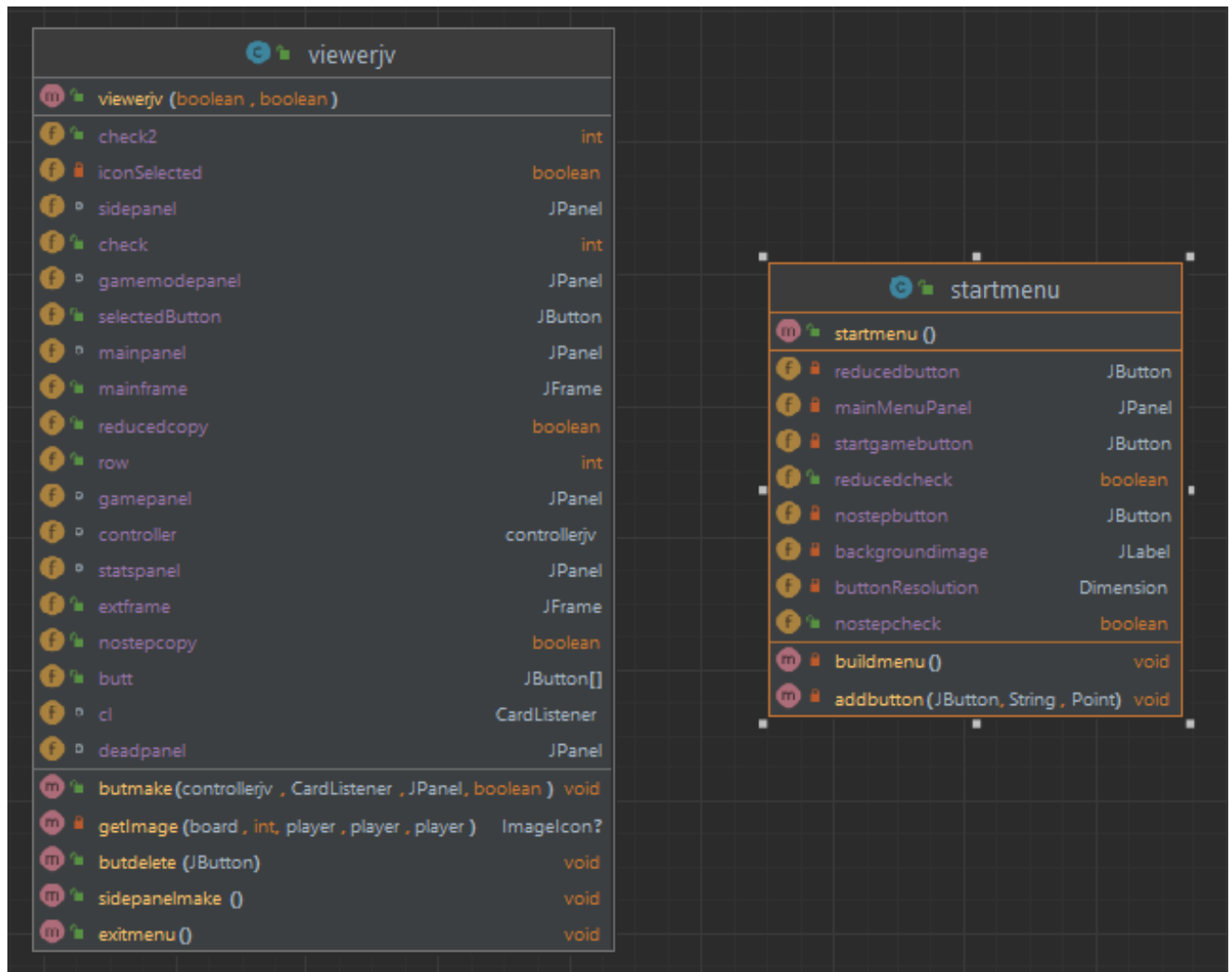
Package model UML



Package controller UML



Package viewer UML



6. Λειτουργικότητα (Β Φάση)

Το παιχνίδι δουλεύει κανονικά και έχουν εφαρμοστεί και οι επιπλέον κανόνες (reduced-army ,no-step-back). Βεβαία το Ressurrect feature δεν υλοποιήθηκε λόγω έλλειψης χρόνου αλλά έχουν προστεθεί τα νεκρά πioniα του καθε παικτη στο siderpanel. Επίσης το Win chance δεν είναι αρκετά ακριβές αλλά δεν το αφαιρέσα . Εκτός αυτών των δύο το πρόγραμμα ακολουθεί όλους τους υπολοίπους δοθειςας κανόνες.

7. Συμπεράσματα

Η άσκηση πέρασε πολλές αλλαγές από την πρώτη φάση καθώς επέτρεπε η πολύ ωμή λογική της να γίνει λειτουργική πράγμα που απαιτήσε αρκετό κοψίμο και ραψίμο (κάτι που ήταν αναμενόμενο).

ΤΕΛΟΣ