

Πανεπιστήμιο Κρήτης - Τμήμα Επιστήμης Υπολογιστών

Έτος 2021-22 , Χειμερινή περίοδος

HY252-Αντικειμενοστραφής Προγραμματισμός

Διδάσκων : I. Τζίτζικας

Ηλίας Καψής - csd4652

ΠΡΟΤΖΕΚΤ

Εισαγωγή :

Σκοπός αυτής της άσκησης , είναι ο σχεδιασμός και η υλοποίηση του διάσημου παιχνιδιού Payday στα Ελληνικά . Σε αυτή τη παρουσίαση θα αναλύσω την σκέψη πίσω από τον σχεδιασμό του παιχνιδιού και θα επισημάνω τις διάφορες κλάσεις και μεθόδους που χρησιμοποιώ.

Περιεχόμενα

- 1. Εισαγωγή 2**
- 2. Η Σχεδίαση και οι Κλάσεις του Πακέτου Model –UML MODEL 2**
- 3. Η Σχεδίαση και οι Κλάσεις του Πακέτου Controller–UML MODEL 10**
- 4. Η Σχεδίαση και οι Κλάσεις του Πακέτου View –UML MODEL 17**
- 5. Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML 21**
- 6. Λειτουργικότητα + Σημειώση 23**

Εισαγωγή (vol 2):

Αρχικά , έπρεπε να χωρίσω το πρόγραμμα μου στο μοντέλο MVC. Δηλαδή, έχω χωρίσει τα γραφικά του παιχνιδιού από τον βασικό πυρήνα του. Οπότε ο χρήστης βλέπει το view, με το controller αλλάζει τις ιδότητες του model , τα οποία στην συνέχεια αλλάζουν και το view. Ετσι , τα packages model και view για να αλληλεπιδρασουν χρειαζονται την μεσολαβηση του Controller η οποια εξαιτιας αυτης της ικανοτητας που έχει , μπορουμε να την καλεσουμε σαν τον εγκεφαλο του παιχνιδιου.

Package Model:

Το package model θα περιέχει τις κλάσεις :

- Class Position
- Abstract class player
- Class controller extends player
- Abstract Class Tile
- MessageTile extends Tile
- AgreementTile extends Tile
- Class Board
- Class Dice

Abstract Class Player

Αρχικά φτιάχνουμε μια abstract class Player η οποια θα έχει

Τα attributes:

Private Int money;

Private Int loanedcash;

Private int bills;

Private String name;

Private int month;

Dice roll = new Dice();

```
Position cords = new Position();
```

```
Stack<AgreementCards> stack = new Stack<AgreementCards>();
```

Οι μέθοδοι:

Player(String x); //constructor that calls set

Set(); //sets money to 3500 and loanedcash to 0, month=0, and
cords.board[] = 0

Class Controller extends Player

Θα εχει τα attributes της inherited κλασης Player και τα methods

CharacterString x)

Void move(int x) αλλαζει τη θεση του παικτη

Int RollDice() επιστεφει ακαιρεο από 1 εως 6

Void changeMoney(int x) αλλαζει χρηματα κατα x

Void changeLOAN(int x) παιενει x\$ δανειο

Void changeBills(int x) αλλαζει το συνολικο ποσο απληρωτων λογαριασμων

Int GetMoney(){retutn super.money}

Int getloaned(){return super.loanedcash}

Int get bills(){return super.bills}

Void giveMoney(Character b, int x) δινει στο b x ποσο χρηματων

Void paybills() αποπληρωνει το χρεος του

Boolean hasFinished(Board v) ελεγχει αν ο παικτης εχει τελειωσει το
παιχνιδι

Void buy(int x) αγοραζει και μειωνει τα χρηματα του παικτη κατά x

Void adjust(int x) Αυτοματα αν τα χρηματα του παικτη δεν αρκουν για να αποπληρωσει το ποσο x δανειζονται αρκετα χρηματα στον παικτη για να ολοκληρωθει η αγορα.

Class Position

Αυτή η κλάση θα δείχνει σε ποια θεση του board βρίσκεται ο player και θα η οποια μπορει να φτασει μεχρι 35

Τα attributes είναι :

Int x ακεραιος που θυμαται την θεση καποιου object

Οι μεθοδοι είναι :

Void method setPosition(int x, Character c){} αυτή η μεθοδος εχει σχεδιαστει ετσι ώστε όταν το cords παει >30 να πηγαινει στην θεση 30 η οποια είναι το PayDayTile του board μας.Επισης αν ειμαστε ηδη στην θεση 30, προσθετουμε το αριθμο ζαριου με το-1 επειδη ξερουνε ότι οι πινакες στην java αρχιζουν από 0 αρα -1+1(min number of dice) = 0.

Class Dice

Αυτή η κλαση θα εχει όριο από 1 μεχρι 6.

Attributes:

Int numb=0;

METHODS:

Int roll(){} επιστρεφει αριθμο 1 εως 6

Int getDice() επιστρεφει τον αριθμο που είναι αποθηκευμενος στο x

CLASS BOARD

Αυτή η κλαση δημιουργει το board του παιχνιδιού οπου αρχικοποιουνται οι καρτες και τα πλακιδια του παιχνιδιου.

Τα attributes :

public Tile[] MyBoard = new Tile[31]; Πινακας των πλακιδιων που είναι instance Tile

public int time; Οι μηνες που θα διαρκεσει το παιχνιδι

AgreementCards[] AT = new AgreementCards[20]; Πινακας που αποθηκευονται ολες οι καρτες συμφωνιας του παιχνδιου με τα instances τους

MailCards[] MT = new MailCards[48];

Πινακας που αποθηκευονται ολες οι καρτες μυνηματος του παιχνδιου

String[][] mailCards = new String[48][4]; Πινακας που αποθηκευει τα στοιχεια των καρτων μυνηματος

Int jackpotamount Είναι το συνολο των χρηματων που εχουν μαζευτει στο jackpot Tile

Int currentcard Αυτος ο αριθμος δειχνει σε ποια καρτα συμφωνιας βρισκομαστε. Η χρηση της είναι πολύ σημαντικη καθως αντι να αποβαλω κάθε καρτα που αγοραζει ο χρηστης μολις την αγορασει καποιος την προσπαιρνω.Οταν τελειωσουν ολες αρχιζω από το 0.

int currentmail Αυτος ο αριθμος δειχνει σε ποια καρτα μυνηματος βρισκομαστε. Η χρηση της είναι πολύ σημαντικη καθως αντι να αποβαλω κάθε καρτα που αγοραζει ο χρηστης μολις την αγορασει καποιος την προσπαιρνω.Οταν τελειωσουν ολες αρχιζω από το 0.

`String[][] dealCards = new String[20][8];` Πίνακας που αποθηκευει τα στοιχεια των καρτων συμφωνιας

Οι μεθοδοι :

`Board(int x)` Δημιουργει το board και αρχικοποιει τους μηνες που θα διαρκεσει το παιχνιδι

`void initTiles()` Αρχικοποιει τις καρτες του παιχνιδιου

`void initBoard()` Αρχικοποιει τα πλακιδια του παιχνιδιου

`CheckEmpty(int x, Tile c)` Ελεχγει κενες θεσεις και βαζει σε τυχαια σειρα τα πλακιδια του παιχνιδιου

`void readFile(String path, String type)` Διαβαζει από το csv αρχειο τα attributes κάθε καρτας

`Void randomiseCards()` Αυτή η μεθοδος ανακατευει τις καρτες του παιχνιδιου

Interface CARDS

Αυτό το interface αντιπροσοπευει τις δυνατοτητες καποιας καρτας και δινει καποιες ειδικες μεθοδους:

`void get(Character r);` είναι ακομα unimplimented

`void move(Character r);` κουναει το παικτη σε καποιο πλακιδιο του board(Mail card ability)

`void pay(Character r, Character b);` ο παικτης r πληρωνει ένα ποσο τον παικτη b (Mail card ability)

`void getpayed(Character r, Character b);` ο παικτης b πληρωνει ένα ποσο τον παικτη r (Mail card ability)

`void payBank(Character r);` ο παικτης πληρωνει ένα ποσο στην τραπεζα

(Mail card ability)

void billCard(Character r); ο λογαριασμός του παικτη χρεωνεται καποιο ποσο
(Mail card ability)

void buy(Character r); ο παικτης αγοραζει καποια καρτα (Deal card ability)

void sell(Character r); ο παικτης πουλαει καποια καρτα (Deall card ability)

Void getcard(int temp Character c); Δινει στον χρηστη την διαθεσιμη καρτα συμφωνιας και την προσθετει στο inventory του

CLASS MailCards implementsCards

Αυτή η κλαση εξελιζει την κλαση cards και αντιποροπευει καρτες μυνηματος του παιχνιδιου .

Τα attributes:

int value; το ποσο που εκδηλωνει η καρτα

String image; η φωτοφραφια της καρτας

String message; το μυνημα της καρτας

String type; ο τυπος της καρτας

Τα methods περιγραφονται αναλυτικα στην Cards

CLASS AgreementCards implements Cards

Αυτή η κλαση εξελιζει την κλαση cards και αντιποροπευει καρτες συμφωνιας του παιχνιδιου .

Τα attributes:

int value; το ποσο που αξιζει η καρτα για αγορα

String image; η φωτοφραφια της καρτας

String message; το μυνημα της καρτας

Int sell value το πόσο που αξιζει η πωληση της καρτας

Τα methods περιγραφονται αναλυτικα στην Cards

Abstract CLASS Tile

Αυτή η κλαση αντιπροπευει ένα γενικευμένο πλακιδιο του παιχνιδιου. Αυτη χωριζεται σε πολλες υποκλασεις οι οποιες δεν εχουν καποιο function αλλα χρησιμοποιουνται για την αναγνωριση ποιας ενεργειας πρεπει να ακολουθησει ο παικτης αφου βρισκεται πανω σε ένα συγκεκριμενου ειδους πλακιδιου.

ΟΛΕΣ ΟΙ ΥΠΟΚΛΑΣΕΙΣ:

Class BuyerTile

Class CasinoTile

Class DealTILE

Class JackPotTile

Class Lottery Tile

ClassMail Tile

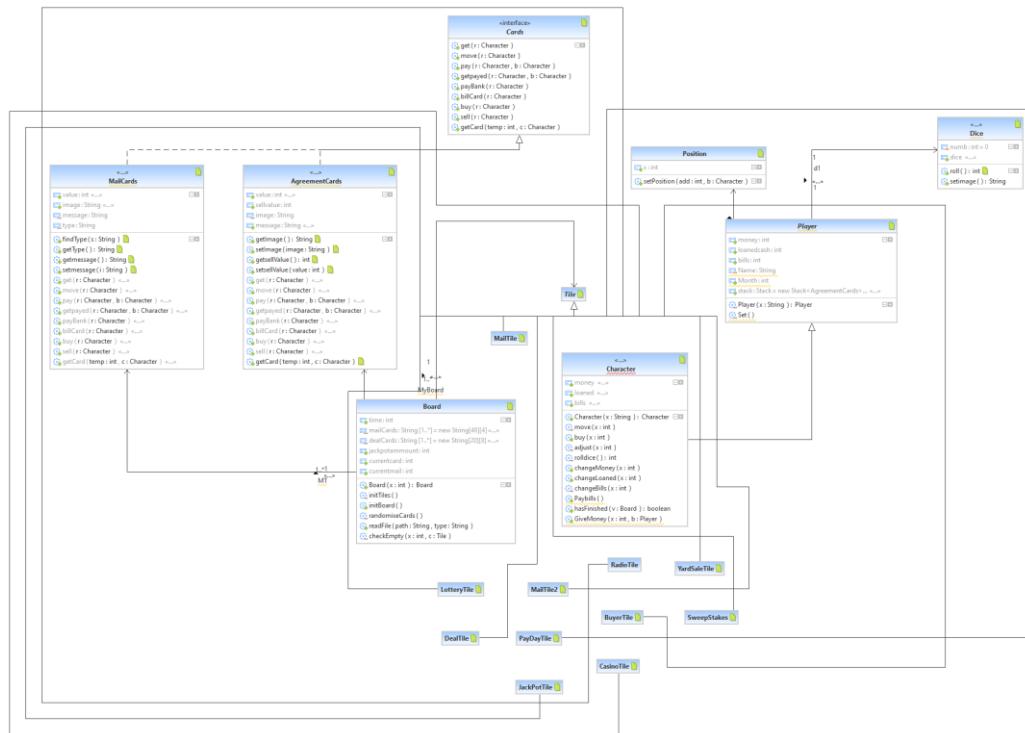
ClassMail Tile2

Class PayDay Tile

Class Radio Tile

Class SweepStakes Tile

Class YardSale Tile



Αυτό είναι το UML Class Diagramm του Model. Η κλαση character κανει extend την κλαση Player αρα παιρνει όλα τα attributes και τις μεθοδους της. Ειδικοτερα, η player εχει επισης σαν attribute το dice ώστε να μπορει να ριχνει ζαρι και Position για να βρισκουμε σε ποια θεση του board είναι. Οι κλασεις που συμβολιζουν τις καρτες συμφωνιας και μυνηματος του παιχνιδιου μας (οι MailCards και οι Agreement cards) κανουν extend το interface Cards το οποιο εχει πολλες μεθοδους. Τελος , η κλαση Board που αναπαριστα το ταμπλο του παιχνιδιου. Η οποια συμπεριλαμβανει μεσα τις καρτες μυνηματος , συμφωνιας και κάθε ειδος ειδικου πλακιδιου του παιχνιδιου.

Package Controller:

Το package controller εχει ως σκοπο να είναι ο εγκεφαλος του παιχνιδιου.

Δηλαδη είναι εκεινο το οποιο θα επιτρεπει το view και το model να επικοινωνουν(αλληλεπιδρουν) και να προχωρανε το παιχνιδι. Το controller καταφερνει να αρχικιοποιει το παιχνιδι , αφηνει τους παικτες να παιξουν την σειρα τους και στο τελος τελειωνει το παιχνιδι και ανακοινωνει τον νικητη. Αρα θα χρειαστει καποιες μεθοδους και attribute που επιτεπουν να το κανει αυτό.

CLASS Controller

Κλαση η οποια ελεγχει την ροη του παιχνιδιου

Τα attributes :

Static int x; Αυτή η μεταβλητη είναι χρησιμη για τον υπολογισμο της σειρας του παιχνιδιου

Static view Myview; Είναι attribute της κλασης γραφικων που εφτιαξα η οποια φροντιζει να απεικονισει σε παραθυρο το board μας .

Static int turn : Ελεγχει την σειρα του παιχνιδιου.

Static int fr1; : Χρησιμοποιειται σε περιπτωσεις οπου ενας από τους χρηστες βρισκεται σε radio Tile και πρεπει να ριξουν ξανα ζαρι

Static int fr2; : Το ίδιο με fr1 απλως για τον άλλο χρηστη

Static int decision; χρησιμοποιηται για όταν ο χρηστης βρισκεται σε μερα Κυριακη και Πεμπτη και πρεπει να κανει μια αποφαση .

Static int AmmountMail ; Είναι χρησιμο για όταν ο χρηστης πρεπει να παρει καρτες Μυνηματος αναλογα σε ποιο tile πεσει γινεται 1 ή 2, αναλογα αν είναι Maletile ή Mailetile2.

Static int yesfroll ; Πολύ σημαντικο γιατι αυτό ξεχωριζει την διαφορα μεταξυ κανονικης ριξης ζαριους και ψευτικη . Όταν είναι true ο χρηστης μπορει να ριξει ζαρι χωρις να κουνησει από την θεση του .

Static boolean p1hasplayed; Επισης πολύ σημαντικο όταν ο χρηστης πατησει end αυτό γινεται true και μπορει να πατησει τα κουμπια του καθως είναι σειρα του αλλου παικτη . Κανονικα το turn θα αρκουσε αλλα επειδη σε καποια tiles ο παικτης πρεπει να ξαναριξει ζαρι μια τετοια μεταβλητη είναι αναγκαια.

static boolean p2hasplayed ; το ίδιο με πριν απλα για τον άλλο παικτη

static Character player1; ο πρωτος παικτης

static Character player2; ο δευτερος παικτης

Static string in χρησιμοποιεται σπανια για να κανει update το Text μας

public static Board b1; To board του παιχνιδιου μας που δημιουργησαμε στο Model. Αναγκαιο για την δημιουργεια της πλακετας του παιχνιδιου.

public static boolean mgo; Χρησιμοποιηται για να απογορευει τον χρηστη να σηκωνει καρτες μυνηματος χωρις να βρισκεται πανω στο ειδικο tole τους. Γινεται true όταν παιφτει πανω σε mail tile 1 ή 2.

public static boolean dgo; χρησιμοποιηται για να απογορευει τον χρηστη να σηκωνει καρτες μυνηματος χωρις να βρισκεται πανω στο ειδικο tole τους. Γινεται true όταν παιφτει πανω σε DealTile

static boolean Sunday; χρησιμοποιηται για όταν ο Χρηστης πεσει σε μερα Κυριακη και όταν γινεται αληθης επιτρεπει στον χρηστη να κανει την καταλληλη πραξει

static boolean Thursday; ρησιμοποιηται για όταν ο Χρηστης πεσει σε μερα Πεμπτη και όταν γινεται αληθης επιτρεπει στον χρηστη να κανει την καταλληλη πραξει

static boolean p1; όταν γινεται αληθης επιτρεπει στον παικτη να πατησει το κουμπι inventory και να πουλησει καρτα

static boolean p2; ίδιο με πριν

Οι μεθόδοι :

private static void controller() Δημιουργεί τον 1ο εγκεφαλο του παιχνδιου και αρχικοποιει την σειρα του παιχνιδιου το board μας , τα γραγικα μας και τα ενημερωτικα σχολια.

static int selectTime() Αρχικοποιει την διαρκεια του παιχνιδιου

static int calculate(Character p) Υπολογιζει το σκορ του παικτη

static boolean hasGameFineshed(Character a, Character b,Board c)

Ελεγχει αν το παιχνιδι εχει τελειωσει

Static int calulateTurn() αποφασιζει την σειρα των παικτων

Void updateText() and updateText(String S)και οι 2 εχουν σκοπο το να ανανεωνουν τα ενημερωτικα μυνηματα του παιχνιδιου.

Void play(character c, int x) Ο δευτερος εγκεφαλος του παιχνιδιου και παρα πολύ σημαντικος.Πηγαινει στην θεση που βρισκεται ο παικτης και βρισκει αναλογα σε ποιο Tile ΚΑΙ!! ποια μερα που είναι την καταλληλη πραξη και αν τελειωσε ο παικτης που αν τελειωσε κανει σιγουρο ότι δεν αλλαζει η σειρα.

Void roll1() το ζαρι του πρωτου χρηστη και μετακινει τον παικτη στη θεση που εριξε το ζαρι του. Επισης ελεγχει αν εριξε εξι αν τελειωσε το παινιδι και αν επαισε σε μερα Κυριακη η Πεμπτη.

Void roll2() ίδιο με το προηγουμενο απλως είναι του αλλου χρηστη

Static void fakeroll() ; Ζος εγκεφαλος παιχνιδιου αναλμβανει τις πραξεις που πρεπει ο χρηστης να ξαναριξει ζαρι όπως πχ sweepstake , Sunday , Thursday κτλπ, και τα στελνει πισω στο play(character x,int x) η στην καταλληλη συναρτηση τους.Ειναι αναγκαια επειδη αν δεν υπηρχε και ηταν μονο το roll το προγραμμα θα εκανε τις πραξεις χωρις να προλαβει ο χρηστης να ξαναριξει το ζαρι. Αυτή η μεθοδος κανει σιγουρο ότι πρωτα θα ριξει ζαρι ο χρηστης και μετα θα παει στη καταλληλη συναρτηση .

Static void end1(); Αυτή η μεθοδος τελειωνει την σειρα του παικτη **1** αφου εχει ολοκληρωσει τις πραξεις του.Επισης τσεκαρει αν εχουν τελειωσει και οι **2** παικτες την σειρα τους οπου μετα το ξαναβαζει στην αρχικη σειρα τους

.

Static void end2() Ιδιο με end1()

Static void inventory1() ; ανοιγει το inventory του παικτη και βλεπει ποιες καρτες εχει ααγορασει .Επισης εχει την δυνατοτητα να πουλησει καρτες.

Static void inventory2() ; ιδιο με το πριν

static void sellcard(AgreementCards s , Character b; Παιρνει τον χρηστη και την καρτα που εχει στο inventory του και την πουλα.

static void dealcard(); Ανοιγει ένα καινουργιο παραθυρο και παιρνει μια καρτα συμφωνιας και την προβαλει . Ο παικτης εχει την δυνατοτητα να την αγορασει. Στην πραγματικοτητα ειναι η συναρτηση που καλειται από τον listener του κουμπιου deal

static void mailcard(); Ανοιγει ένα καινουργιο παραθυρο και προβαλλει μια καρτα μυνηματος . Εχει την δυνατοτητα να καταλαβαινει τι ειδος καρτας ειναι με αποτελεσμα να καταλβαινει τι ειδος πραξη ο παικτης πρεπει να κανει.Ειναι στην πραγματικοτητα η συναρτηση που καλειτε από τον listener του κουμπιου mail.

static void Sweep(); Είναι η συναρτηση που καλειται όταν ο χρηστης πεσει σε Sweepstakes Tile . Κανει τις καταλληλες πραξεις για αυτό το ειδικο tile.

static void lottery(); Είναι η συναρτηση που καλειται αν ο χρηστης πεσει σε Lottery Tile . Ανοιγει ένα νέο παραθυρο και βαζει στους χρηστες να επιλεξουν διαφορετικους αριθμους.

Static void radio(); Είναι η συναρτηση που καλειται αν ο χρηστης πεσει σε Radio Tile .Ανοιγει νέο παραυθυρο και δινει στους χρηστες ζαρια να ριξουν.

Static void buyer(); Όταν ο χρηστης πεσει σε Buyer Tile καλειται αυτή η συναρτηση και ο χρηστης μπορει να πουλησει μια καρτα του.

static void casino(Character c); Συναρτηση που καλειται αν πεσει σε casino tile και κανει τις καταλληλες πραξεις.

static void yard() Συναρτηση που καλειται και αν ο παικτης πεσει σε YardSale Tile

Static void PayD() Είναι η συναρτηση που καλειται όταν ο χρηστης φτασει στο τελος του μηνα. Κανει τις πραξεις που λενε οι οδηγιες.

static void sun(); Είναι η συναρτηση που καλειται όταν ο χρηστης πεσει σε μερα Κυριακη .Κανει τις πραξεις που πρεπει (Συνηθως ο χρηστης πρεπει να το κανει αφου πρωτα τελειωσει τις άλλες πραξεις του)

static void thur() το ίδιο με πριν απλα για τις μερες τις Πεμπτης.

Static void endGame(); Τελειωνει το παιχνδι και βρισκει τον νικητη



Αυτό είναι το UML Class Diagramm της κλασης Controller .Μπορουμε να παρατηρουμε ολες τις ιδιαιτερες μεθοδους για κάθε Tile του παιχνιδιου αλλα και τα Attributes . Δυο πολύ σημαντικα είναι το board b1 = new Bord() και το view MyView = new view().

Package VIEW:

Το package view αναλαμβάνει τα γραφικά του παιχνιδιού και το interface με το οποίο οι Παικτες επικοινωνούν για να κανουν πράξεις στο παιχνίδι

Class VIEW:

Κλαση η οποια προβαλλει στην οθονη τα γραφικα του παιχνιδιου.Συγκεκριμενα αναπαριστα τα 31 Tiles του παιχνιδιου αλλα και το JackPot Tile με buttons που δεν εχουν actionPerformed.Επισης εχει 4 button ένα για κάθε χρηστη με τα οποια , οι παικτες μπορουν να αλληλεπιδρασουν και να παιξουν το παιχνιδι.Τελος , εχει 3 JLabels που δειχνουν ενημερωτικα σχολια για το παιχνδι.

Τα attributes:

JFrame Mp = new JFrame("PAYDAY"); Το Frame του παιχνιδιου

JPanel panel1 = new JPanel(); Το panel για την αναπαρασταση του παιχνιδιου

JPanel panel2 = new JPanel(); Το panel για την τοποθετηση κουμπιων για τους παικτες(στο μελλον θα χωριστει σε 2 μικροτερα panels 1 για κάθε παικτη)

Jpanel panel3 = new JPanel(); Θα ηταν το 3o panel αλλα δεν ακολουθησα αυτην αρχιτεκτονικη και λαθος μου καθως αναγκαστηκα να κανω το frame non resizable.

JTextArea info = new JTextArea(); Περιοχη που φαινονται ενημερωτικα σχολια για το παιχνιδι.

**public JLabel label1 = new JLabel("<html>Player 1
Money: 0
Loan: 0
bills: 0");** Label που δειχνεί τα χρηματά του Player1

**public JLabel label2 = new JLabel("<html>Player 2
Money: 0
Loan: 0
bills: 0");** Label που δειχνεί τα χρηματά του Player2

public JLabel label3 = new JLabel("<html>JackPot Money:"); Label που δειχνεί ποσα χρηματά εχουν μαζευτεί στο jackpot Tile.

public JButton roll1 = new JButton("Roll dice") Το ζαρί του πρωτου παικτη

public JButton roll2 = new JButton("Roll dice");Το ζαρί του 2ου παικτη

public JButton card1 = new JButton("My Deal Cards");Το κουμπι που δινει στον παικτη την δυνατοτητα να ανοιξει το inventory του

public JButton card2 = new JButton("My Deal Cards");Ιδιο με το πριν απλα για άλλο παικτη

public JButton loan1 = new JButton("Get a Loan"); ο κουμπι που δινει στον παικτη την δυνατοτητα να παρει δανειο

public JButton loan2 = new JButton("Get a Loan"); Το ίδιο αλλα για τον άλλο παικτη

public JButton turn1 = new JButton("End Turn"); Το παταει ο παικτης για να τελειωσει την σειρα του και να αρχισει η σειρα του αλλου παικτη

public JButton turn2 = new JButton("End Turn"); Το ίδιο με πριν απλα για τον άλλο παικτη

public JButton dice1 = new JButton();Αναπαραστα γραφικα το ζαρι του πρωτου παικτη

public JButton dice2 = new JButton(); Αναπαραστα γραφικα το ζαρι του δευτερου παικτη

public JButton mail = new JButton();Το κουμπι που επιτρεπει στον χρηστη να παρει καρτα μυνηματος

public JButton deal= new JButton(); Το κουμπι που επιτρεπει στον χρηστη να παρει καρτα συμφωνιας

Οι μεθοδοι:

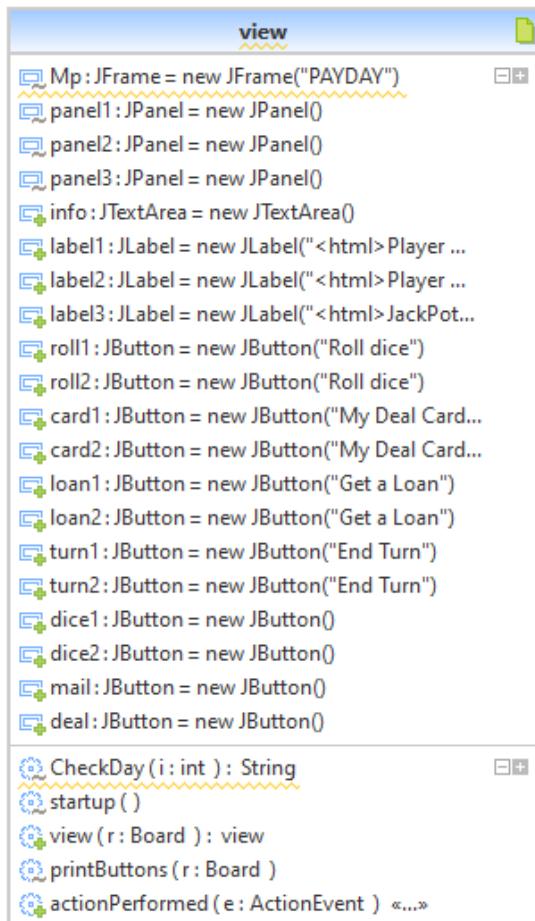
public view(Board r) Δημιουργει την ένα instance της κλασης view που

Αναλαμβανεθ την δημιουργια παραθυρου για προβολη γραφικων

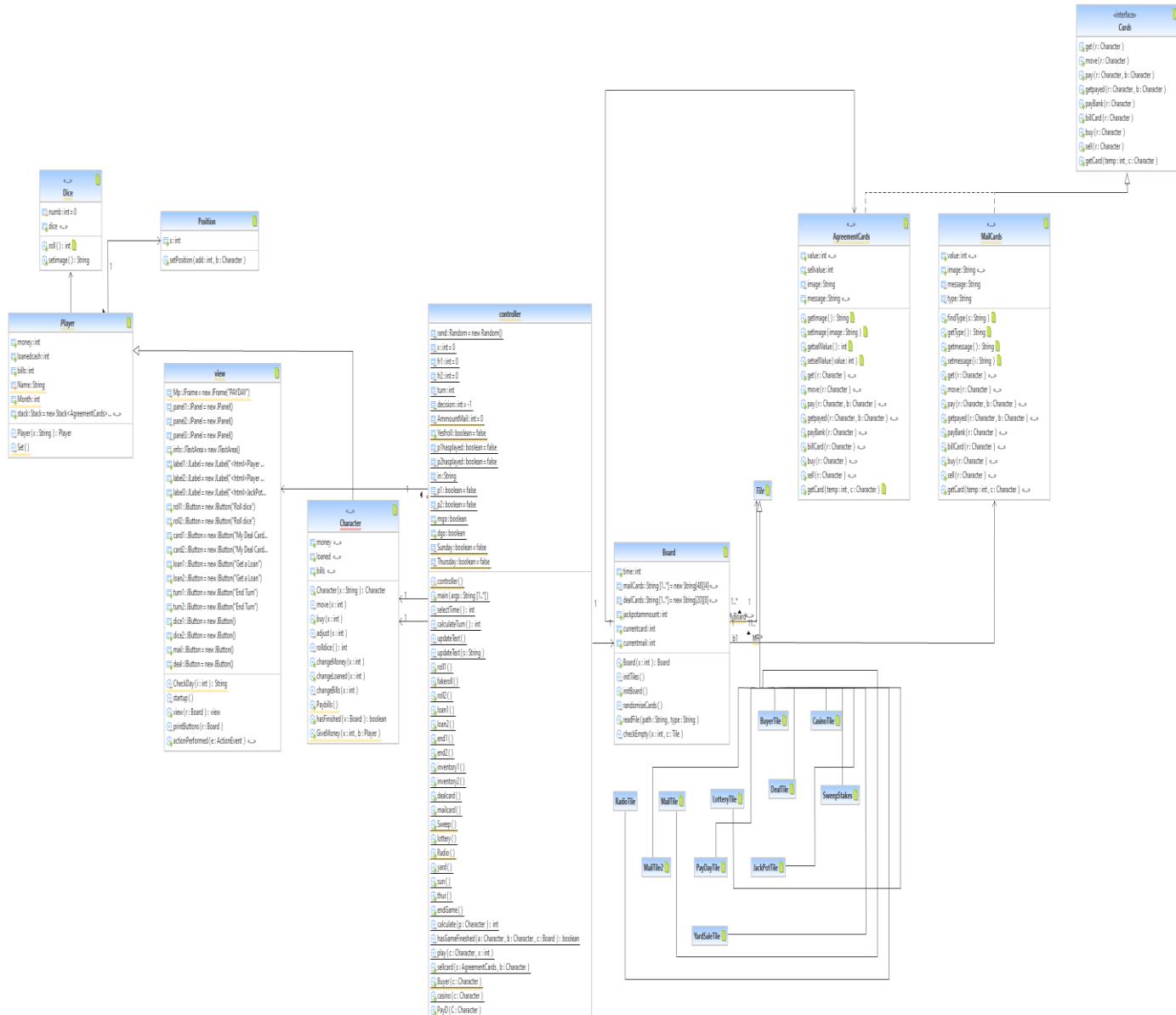
void printButtons (Board r) Τυπωνει όλα τα πλακιδια του board

String CheckDay(int i) Τυπωνει τις μερες του κάθε πλακιδιου που ανηκει στο board

void startup(); Πολύ σημαντική δινει , λειτουργικοτητα στα κουμπια.



UML PROJECT DIAGRAMM



Από το UML CLASS DIAGRAMM μπορουμε να μελετησουμε τις διασυνδεσεις μεταξυ των κλασσεων. Αρχικα , η κλαση Player συνδεεται με τις κλασεις Dice (η οποια δινει δινατοτητα στον παικτη να ριξει ζαρι) και Position (η οποια αποθηκευει την θεση του παικτη στο board).Μετα , η κλαση Character συνδεται με την player αφου την κανει extend και παιρνει τα attributes και προσφερει νεες μεθοδους.Στην συνεχεια , οι κλασεις Agreement Cards , DealCards οι οποιες κανουν extend Cards και αντιπροσωπευουν τις καρτες του παιχνιδιο.Παραλληλα, το Board συνδεεται με τις καρτες και την κλαση Tiles η οποια αποτελειται από πολλα instances της.Η κλαση controller από το package controller τωρα συνδεεται με τη Board και την Character επειδη ετσι το controller εχει συνδεση με το package model .Επισης συνδεεται με την κλαση View από το package view που εχει αναλαβει τα γραφικα του παιχνιδιου.Ετσι βλεπουμε ότι κάθε package εχει καποια σχετικη ανεξαρτησια καθως ο μονος τροπος για να επικοινωνισουν view και model είναι μεσω του εγκεγαλου του παιχνιδιου δηλαδη του controller.

Λειτουργικότητα

Το παιχνίδι αρχικά αποφασίζει πια θα είναι η σειρά των παικτών (μια διαδικασία η οποία γίνεται τυχαιά κάθε φορά) και αρχικοποιεί τα tiles του board και τις καρτες του παιχνδιου(τα οποια επισής κάθε φορά επιλεγονται σε τυχαια σειρα.) Στην συνεχεια δινεται η δινατοτητα στους παικτες να παιζουν με ενναλαξ σειρα . Κάθε φορα που καποιος παικτης πεφτει σε ειδικο Tile βγαινει ενημερωτικο σχολιο που του λεει τι να κανει καθως και σε ποια θεση βρισκεται .Σε περιπτωση που ενας παικτης εχει τελειωσει ο άλλος παικτης συνεχιζει το παιχνιδι μεχρι να τελειωσει . Οι καρτες μολις τις χρησιμοποιεισει καποιος απποριππονται εκτος αν τελειωσουν οι καρτες που στην συγκεκριμενη περιπτωση ο ξαναμπαινουν πισω. Αν καποιος χρηστης θελει να αγορασει κατι και δεν του φτανουν τα λεφτα ο κωδικας αυτοματα του βαζει δανεικα λεφτα.Παραλληλα οι παικτες μπορουν να πουλησουν καρτες συμφωνιας μονο αν πεσουν σε buyer Tile ενώ σε περιπτωση που η καρτα μυνηματος είναι move to buyer αν ο παικτης βρισκεται σε θεση που δεν υπαρχει άλλος buyer μεχρι το τελος του μηνα μενει στην θεση του.Το tile Payday δινει 3500\$ στον παικτη , του επιβαλλει φορο , πληρωνει όλα τα bills ανεξαρτητα αν δεν εχει αρκετα καθαρα χρηματα και του δινει δινατοτητα να ξεπληρωσει ένα ποσο του δανειου .Αν είναι τελευταιος μηνας ο παικτης ξαναριχνει το ζαρι για τελευταια φορα ώστε να <<ξεπερασει >> την μερα PayDay.

Σημειωση

Το μονο functionality του παιχνιδιου που είναι λιγο buggy είναι οι μερες Κυριακη και παρασκευη .Ενω στον κωδικα το εχω βαλει να γινονται μετα από τις κανονικες πραξεις του κάθε Tile το παραθυρο βγαινει ταυτοχρονα .Δεν περιμενω να παρω ολους τους βαθμους από τις μερες Κυριακη Πεμπτη αλλα όχι και κανεναν. Όταν ο χρηστης πεσει σε μερα οπου πρεπει να σηκωσει mail η deal card προτιμοτερο είναι να παιξει πρωτα thurdsay η sunday .Σε buyer tile και lottery δεν εχει σημασια ουτε στο casino. Προβλημα παρουσιαζεται σε radio Tile η yardsale- sweepstake οπου

μπαιδευονται οι σειρες. Σας ευχαριστω για την υπομονη εκ των προτετων
☺.

Τελος