



Πανεπιστήμιο Κρήτης –Τμήμα Επιστήμης Υπολογιστών

ΗΥ252– Αντικειμενοστρεφής Προγραμματισμός

Διδάσκων: Ι. Τζίζικας

Χειμερινό Εξάμηνο 2020-2021

*Εισαγωγή*

## Περιεχόμενα

<u>1.Εισαγωγή.....</u>	<u>2</u>
<u>2.Η Σχεδίαση και οι Κλάσεις του Πακέτου Model.....</u>	<u>3</u>
<u>3.Η Σχεδίαση και οι Κλάσεις του Πακέτου Controller.....</u>	<u>3</u>

<u>4.Η Σχεδίαση και οι Κλάσεις του Πακέτου View.....</u>	<u>3</u>
<u>5.Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML.....</u>	<u>3</u>
<u>6.Λειτουργικότητα (Β Φάση).....</u>	<u>3</u>
<u>7.Συμπεράσματα.....</u>	<u>4</u>

## 1. Εισαγωγή

Η εργασία υλοποιείται με βάση το μοντέλο MVC ώστε να καθιστήσει ευκολότερη την ανάπτυξη όσο και την συντήρηση του πρότζεκτ. Έτσι χωρίζουμε το πρότζεκτ σε Model,View,Controller καθώς και τον φάκελο resources για τα

στατικά αρχεία(π.χ εικόνες κτλ) οι υπόλοιπες ενότητες περιέχουν αναλυτικά την περιγραφή των κλάσεων και με ποιόν τρόπο θα υλοποιηθούν στην επόμενη φάση.

## 2. Η Σχεδίαση και οι Κλάσεις του Πακέτου Model

### Κλάσεις

#### class Board

##### Attributes:

- 1.private BufferedImage logo
- 2.private ArrayList<Tile> tiles
- 3.private ArrayList<Player> players

##### Methods:

- 1.public Board(Player p1,Player p2,MailCardDeck mailCardDeck,DealCardDeck dealCardDeck)
- 2.public Board(Player p1,Player p2)
- 3.getPlayerPosition(Player player)
- 4.getCurrentPlayerTile(Player player)
- 5.movePlayerToPosition(Player player,int position)
- 6.movePlayerToPositionRight(Player player,int position)
- 7.public Tile[] getTiles()
- 8.public int getTilePosition(Class tile,int startPosition)
- 9.public Player getPlayer1()
- 10.public Player getPlayer2()
- 11.public void adjustPlayersToTiles()
- 12.public void setDealCardDeck(DealCardDeck dealCardDeck)
- 13.public DealCardDeck getDealCardDeck()
- 14.public void setMailCardDeck(MailCardDeck mailCardDeck)
- 15.public MailCardDeck getMailCardDeck

#### class Tile

Η κλάση tile είναι η υπερκλάση των στοιχείων που απαρτίζουν τον πίνακα(board)

##### Attributes:

- 1.private int position
- 2.private String day
- 3.BufferedImage image

### **Methods:**

1. public Tile(String title, String description, int position, String day, BufferedImage image, Board board)

2. protected Tile(int position, String day, BufferedImage image, Board board)

3. protected Tile()

4. public String getTitle()

5. public String getDescription()

6. protected void setDescription(String description)

7. public String getDate()

8. public BufferedImage getImage()

9. public int getPosition()

10. abstract void performAction()

## **class Card**

Η κλάση card είναι η υπερκλάση όλων των καρτών του παιχνιδιού με τις βασικές ιδιότητες μιας κάρτας.

### **Attributes:**

- 1.private String title: Ο τίτλος της κάρτας
- 2.private String description: Η περιγραφή της κάρτας
- 3.private BufferedImage image: Η εικόνα της κάρτας\

### **Methods:**

- 1.public Card(String title,String description,BufferedImage image): Ο constructor μίας κάρτας
- 2.public String getTitle(): Μια μέθοδος accessor που επιστρέφει τον τίτλο της κάρτας
- 3.protected void setTitle(String title): Μια μέθοδος transformer που ορίζει τον τίτλο της κάρτας
- 4.public String getDescription(): Μια μέθοδος accessor που επιστρέφει την περιγραφή της κάρτας
- 5.protected void setDescription(String description): Μια μέθοδος transformer που θέτει την τιμή της περιγραφής της κάρτας
- 6.public BufferedImage getImage(): Μια μέθοδος που επιστρέφει την εικόνα της κάρτας.

## **class MailCard**

### **Attributes:**

1.String confirmText: Το μήνυμα που εμφανίζεται σαν κουμπί στη κάρτα

### **Methods:**

1.public MailCard(): O constructor μίας mail κάρτας

2.public void setConfirmText(String confirmText): Μία μέθοδος transformer η οποία ορίζει

3.public String getConfirmText()

4.public void setOwner(Player owner)

5.public Player getOwner()

6.public abstract void performAction()

## **class DealCard**

### **Attributes:**

1.private int buyPrice: Η αξία της κάρτας

2.private int sellPrice: Η τιμή πώλησης της κάρτας

3.String choice1: Η 1η επιλογή που έχει ο παίκτης

4.String choice2: Η 2η επιλογή που έχει ο παίκτης

### **Methods:**

1.public DealCard(String title,String description,BufferedImage image, String confirmText): O Constructor μίας deal κάρτας

2.public int getBuyPrice()

3.public int getSellPrice()

4.public String getChoice1()

5.public String getChoice2()

6.public String toString()

### **class Advertisement**

#### **Attributes:**

-

#### **Methods:**

1. public Advertisement(Board board)
2. public void performAction()

### **class Bill**

#### **Attributes:**

-

#### **Methods:**

1. public void Bill(Board board)
2. public void performAction()

### **class Charity**

#### **Attributes:**

-

#### **Methods:**

1.public Charity(Board board)

2.public void performAction()

### **class MoveToDealOrBuyer**

#### **Attributes:**

-

#### **Methods:**

1.public MoveToDealOrBuyer(Board board)

2.public void performAction()

### **class PayNeighbor**

#### **Attributes:**

-

#### **Methods:**

1.public PayNeighbor(Board board)

2.public void performAction()



## **class TakeFromNeighbor**

### **Attributes:**

-

### **Methods:**

1. public TakeFromNeighbor(Board board)
2. public void performAction()

## **class DealCardDeck**

### **Attributes:**

-

### **Methods:**

1. public DealCardDeck()
2. public void draw(Player player)

## **class MailCardDeck**

### **Attributes:**

-

### **Methods:**

1. public MailCardDeck()
2. public ArrayList<MailCard> getDeck()
3. public void shuffle()
4. public void draw(Player player)

## **class Event**

### **Attributes:**

- 1.private int id
- 2.private String eventName

### **Methods:**

- 1.protected Event(int id,String eventName,String alias,String description,BufferedImage image,int optionsNumber)
- 2.public int getId()
- 3.public String getEventName()
- 4.public String getAlias()
- 5.public String getDescription()
- 6.public int getOptionsNumber()
- 7.public String[] getOptionStrngs()
- 8.protected void setOptionStrings(String[] optionStrings)
- 9.public BufferedImage getImage()
- 10.public abstract void performAction()

## **class Jackpot**

### **Attributes:**

- 1.private float value

### **Methods:**

- 1.public Jackpot()
- 2.public float getValue()
- 3.public void setValue(float value)
- 4.public void addValue(float Value)
- 5.public void decreaseValue(float value)
- 6.public void performAction()

### **class SundayFootballDay**

#### **Attributes:**

1.

#### **Methods:**

1.public SundayFootballDay()

### **class ThursdayRiseValueCrypto**

#### **Attributes:**

-

#### **Methods:**

1.public ThursdayRiseValueCrypto()

### **class PayDay**

#### **Attributes:**

-

#### **Methods:**

1.public ThursdayRiseValueCrypto()

### **class ThursdayRiseValueCrypto**

#### **Attributes:**

-

#### **Methods:**

1.public ThursdayRiseValueCrypto()

## **class Player**

### **Attributes:**

- 1.private String name: Το όνομα του παίκτη
- 2.private float balance: Το χρηματικό υπόλοιπο του παίκτη
- 3.private float bills: Τα χρήματα που χρωστάει ο παίκτης
- 4.private float loans: Τα χρήματα που έχει πάρει ο παίκτης απο δάνεια

### **Methods:**

- 1.public Player(String name,float balance): Ο Constructor ενός παίκτη
- 2.public String getName(): Μία μέθοδος accessor που επιστρέφει το όνομα του παίκτη
- 3.public void setName(String name): Μία μέθοδος transformer που θέτει το όνομα του παίκτη
- 4.public float getBalance(): Μία μέθοδος accessor που επιστρέφει το χρηματικό υπόλοιπο του παίκτη
- 5.public void setBalance()
- 6.public void addDealCard(DealCard dealCard)
- 7.public void removeDealCard(DealCard dealCard)
- 8.public void payAllBills()
- 9.public void payBillAmount(float amount)
- 10.public void setBills(float bills)
- 11.public float getBills()
- 12.public void addBills(float bills)

- 13.public void removeBills(float bills)
- 14.public void setLoans(float loans)
- 15.public float getBills()
- 16.public void addBills(float bills)
- 17.public void removeBills(float bills)
- 18.public void setLoans(float loans)
- 19.public float getLoans()
- 20.public void takeLoan(float loanValue)
- 21.public void decreaseLoanValue(float loanValue)
- 22.public void addCash(float cash)
- 23.public void removeCash(float cash)

### 3. Η Σχεδίαση και οι Κλάσεις του Πακέτου Controller

Η κλάση controller είναι ο “εγκέφαλος” του μοντέλου MVC δηλαδή όλες οι αρχικοποιήσεις πραγματοποιούνται εδώ. Στην συγκεκριμένη υλοποίηση έχουν αρχικοποιηθεί οι παίκτες, οι τράπουλες, το ταμπλό καθώς και τα ζάρια του κάθε παίκτη.

Στή Β φάση του πρότζεκτ θα προσθέσουμε μεθόδους ώστε να επιτευχθεί ένα σύστημα σειρών(με τη κλαση Turn) στη κλάση Board επιπλέον θα υλοποιήσουμε τις μεθόδους που χρειάζεται ώστε να διαχειρίζονται και να συντονίζονται σωστά απο τον controller οι σειρές οι παίκτες κτλ.

#### **class Controller**

##### **Attributes:**

- 1.private Player p1,p2

2.private MailCardDeck mailCardDeck

3.private DealCardDeck dealCardDeck

4.private Board board

5.private int months

6.private Turn turn

7.public static Jackpot jackpot

#### **Methods:**

1.public Controller(int months)

2.public Turn getTurn()

3.public DealCardDeck getDealCardDeck()

4.public MailCardDeck getMailCardDeck()

5.public Board getBoard()

## 4. Η Σχεδίαση και οι Κλάσεις του Πακέτου View

Στο πακέτο view έχουμε μερικές κλάσεις όπως είναι η CardPorup η οποία χωρίζεται στις υποκλάσεις DealCardPorup και MailCardPorup στις οποίες δημιουργούνται τα παραθυράκια που θα εμφανίζονται ως κάρτες στον παίκτη. Στη συνέχεια είναι η BoardView η οποία αναπαριστά το ταμπλό με τις κάρτες και είναι κομμάτι της GraphicUI το οποίο είναι το ολοκληρωμένο UI.

### **class BoardView**

#### **Attributes:**

1.JLabel playerOnePawn

2.JLabel playerTwoPawn

3.ArrayList<JLabel> dates

4.ArrayList<JLabel> tileImages

5.ArrayList<JPanel> tiles

6.JPanel jackpot

7.GridBagConstraints c

8.Player p1

9.Player p2

10.Board board

11.JLabel jackpotImage

12.JPanel tilesPanel

13.InfoBoxView infoBoxView

#### **Methods:**

1.public BoardView(Board board, Player p1,Player p2)

2.public void updatePawns()

3.public void addInfoBoxView(InfoBoxView infoBoxView)

4. public InfoBoxView getInfoBoxView()

### **class DealCardPopup**

#### **Attributes:**

1. JLabel image
2. JLabel description
3. JButton choice1
4. JButton choice2

#### **Methods:**

1. public DealCardPopup(DealCard card, Player player)
2. public void setButtonsVisible(boolean visible)
3. public void setButtonsEnabled(boolean enabled)

### **class EventPopup**

#### **Attributes:**

1. JLabel image
2. JLabel description
3. JButton[] options

#### **Methods:**

1. public EventPoup(Event event)

### **class InfoBoxView**

#### **Attributes:**

1. JTextField title
2. JTextField monthsLeft
3. JTextField currentTurn
4. JTextField currentAction
5. Player currentPlayer

#### **Methods:**

1. public InfoBoxView(Player player)
2. public void setCurrentPlayer(Player player)
3. public void updateInfoBox()

### **class MailCardPoup**

#### **Attributes:**

1. JLabel description
2. JButton confirm
3. JLabel image

#### **Methods:**

1. public MailCardPopup(MailCard card)

## **class PlayerUI**

### **Attributes:**

- 1.JLabel name
- 2.JLabel balance
- 3.JLabel loan
- 4.JLabel bills
- 5.JPanel detailsPanel
- 6.JButton rollDice
- 7.JButton viewDealCards
- 8.JButton getLoan
- 9.JButton endTurn
- 10.JPanel buttonsPanel
- 11.JLabel diceImage
- 12.Player player

### **Methods:**

- 1.public PlayerUI(Player player,Turn turn,BoardView boardView)
- 2.public void updatePlayerInfo()
- 3.public JButton getEndTurn()
- 4.public void setEndTurnListener(ActionListener listener)
- 5.public void setEnabledButtons(boolean enabled)
- 6.private ImageIcon getDiceImage(Player player)

## **class TilePopup**

### **Attributes:**

- 1.JLabel image
- 2.JLabel description
- 3.JButton confirm

### **Methods:**

- 1.public TilePoup(Tile tile)

## **class GraphicUI**

### **Attributes:**

- 1.Controller controller
- 2.JLabel logo
- 3.BoardView boardPanel
- 4.InfoBoxView infoBox
- 5.PlayerUI playerOnePanel
- 6.PlayerUI playerTwoPanel
- 7.JPanel left
- 8.JPanel right
- 9.Board board

### **Methods:**

- 1.public GraphicUI() throws IOException



```
2. public static void main(String[] args)
```

## 5. Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML

Τα διαγράμματα UML υπάρχουν στον φάκελο του project.

## 6. Λειτουργικότητα (B Φάση)

Στή Β φάση καταφέραμε να υλοποιήσουμε τους παίκτες και τις λειτουργίες τους καθώς και τον πίνακα. Συγκεκριμένα η κλάση Board λειτουργεί σαν τον πίνακα του παιχνιδιού αρχικοποιώντας τα πλακάκια(τις θέσεις) του σε συγκεκριμένες είτε τυχαίες θέσεις ώστε να μπορεί το πiónι του παίκτη να προχωρήσει και να υλοποιηθεί το παιχνίδι. Οι βασικές κλάσεις που έχουν υλοποιηθεί πέρα του Board είναι οι εξής:

- Player: Η κλάση που αναπαράσχει τον παίκτη
- Dice: Η κλάση που αναπαράσχει το ζάρι του κάθε παίκτη
- Tile: Μία abstract κλάση που αναπαράσχει την γενική εικόνα ενός πλακακιού
- Turn: Η κλάση που μετράει την σειρά των παικτών(ποιοί παίζουν αυτή τη στιγμή κτλ.)
- MailCardDeck, DealCardDeck: Οι κλάσεις που αναπαριστούν τις τράπουλες του παιχνιδιού

Γενικώς, τα ερωτήματα που τέθηκαν υλοποιήθηκαν μερικώς η πλήρως. Ωστόσο υπήρχαν ορισμένα ερωτήματα τα οποία απαιτούσαν περισσότερη δουλειά.

## 7. Συμπεράσματα

Η εργασία αποτελείται από πολλαπλές κλάσεις που βασίστηκαν στο μοντέλο MVC. Το συγκεκριμένο μοντέλο βοηθάει ιδιαίτερα στη ανάπτυξη του προγράμματος και τη τροποποίηση του αργότερα. Οι δυσκολίες της εργασίας ήταν η εξοικίωση με το συγκεκριμένο μοντέλο και ο τρόπος με τον οποίο συνδεόνται οι διάφορες κλάσεις μεταξύ τους.