

**Πανεπιστήμιο Κρήτης –Τμήμα Επιστήμης Υπολογιστών**

**ΗΥ252– Αντικειμενοστραφής Προγραμματισμός**

**Διδάσκων: Ι. Τζίτζικας**

**Χειμερινό Εξάμηνο 2020-2021**

**

*Εισαγωγή*



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

**Περιεχόμενα**

[**1.**](#_heading=h.gjdgxs) **Εισαγωγή**

[**2.**](#_heading=h.30j0zll) **Η Σχεδίαση και οι Κλάσεις του Πακέτου Model**

[**3.**](#_heading=h.1fob9te) **Η Σχεδίαση και οι Κλάσεις του Πακέτου Controller**

[**4.**](#_heading=h.3znysh7) **Η Σχεδίαση και οι Κλάσεις του Πακέτου View**

[**5.**](#_heading=h.2et92p0) **Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML**

[**6.**](#_heading=h.tyjcwt) **Λειτουργικότητα (Β Φάση)**

[**7.**](#_heading=h.3dy6vkm) **Συμπεράσματα**

## Εισαγωγή

Η εργασία χωρίζεται με βάση το μοντέλο MVC (Model View Control).Ουσιαστικά, η υλοποίηση της εργασίας έχει χωριστεί σε τρία διαφορετικά πακέτα , το καθένα με τις δικές του κλάσεις και μεθόδους.Συνοπτικά τα πακέτα είναι τα ακόλουθα:View, όπου και υλοποιείται η γραφική αναπαράσταση του επιτραπέζιου, Model όπου υλοποιούνται ολες οι βασικές οντότητες του επιτραπέζιου(π χ ο παίκτης) και Controller οπου και ορίζονται όλες οι κλάσεις και μέθοδοι που ενώνουν τα View κι Model και περατώνουν την εκτέλεση του παιχνιδιού. Ακολουθεί αναλυτικότερη αναφορά στο κάθε πακέτο παρακάτω σε αυτή την αναφορά.

## Η Σχεδίαση και οι Κλάσεις του Πακέτου Model

Αρχικά, θα αναφέρουμε την δομή του πακέτου Model .

**Το ταμπλό(board)**:

Το ταμπλό περιέχει δύο κλάσεις. Την Board η οποία λειτουργεί ως η υλοποίηση του πραγματικού ταμπλό του παιχνιδιού, και την month, της οποίας ο σκοπός είναι ο ορισμός του “χρόνου” που απομένει στο παιχνίδι, κοινώς οι γενικοί γύροι του παιχνιδιού Και οι δύο κλάσεις περιέχουν τις κατάλληλες set και get μεθόδους καθώς και τους αντιστοιχους constructors. Πιο συγκεκριμενα η board και η month έχουν τις ακόλουθες επιπρόσθετες μεθόδους (συνοδευόμενες ) απο την εξήγηση της λειτουργικότητας τους στην αγγλική γλώσσα:

Adds a mail card c to the array list mail cards

public void addMailCard(mail\_card c){

mailCards.add(c);

}

Adds a deal card c to the array list deal cards

public void addDealCard(deal\_card c){

dealCards.add(c);

}

Plays the card then removes it from stack or adds it to personal deck of the player. Population of said card is decreased by one.

public void play\_card(Card c){};

This method is specifically designed to be applied to deal cards as they are going to be the ones that the player is going to choose to buy and keep in his deck or deny

public deal\_card TakeDealCard(){

deal\_card C=dealCards.get(0);

dealCards.remove(0);

return C;

}

This method is used to determine whether the player p is actually going to win the jackpot while being his turn

public void Jackpot(Player p,int current\_dice\_roll){};

This method is meant to enable the player located on a Sunday tile to bet on a said football game.Depending on his dice roll and his personal bet choice,the player is either rewarded with double his bet money or met with a total loss of said money.

public void football(Player p,String current\_day,int current\_dice\_roll){};

This method allows the player to bet on the rise or fall of cryptocurrency.Depending on the dice roll and the player's own personal bet choice he can be met with either 0 income,loss of his already placed in bet money or doubling that and gaining income.

public void crypto(Player p,String current\_day,int current\_dice\_roll){};

This method is meant to be used to determine how many months each player has ahead of him to finish the game.Months in this case represent the times a player has to run through the board.

public int calculate\_remaining\_months(Player p1, Player p2){

return 0;

}

**Οι κάρτες(Card):**

Η κάρτα περιέχει εφτά κλάσεις(Advertisement , Bill,Card, Charity, deal\_card, GetMoneyFromNeighbor, mail\_card , MoveToSpecifiedSpot, PayTheNeighbor) εκ των οποίων οι δύο έχουν οριστεί abstract καθώς “κουβαλάνε” μεθόδους τις οποίες πρέπει οι κλάσεις-παιδιά τους να κάνουν override(card και mail\_card) .Αυτές οι κλάσεις αναπαριστούν όλες τις κάρτες που χρησιμοποιούνται σε αυτό το επιτραπέζιο, οι οποίες αποτελούν και τις θέσεις (tiles).Πέρα από τις βασικές μεθόδους setters και getters καθώς και τους constructors της κάθε κλάσης περιέχεται και η ακόλουθη μέθοδος :

This method is meant to be used as the means of executing the effect of each card by the player.

public abstract void player\_action(Player p);

Η οποία είναι και η μέθοδος που γίνεται είναι inherited και overwritten απο κάθε subclass που την υλοποιεί , ανάγοντας την στα δικά του μέτρα και στην δική του θεμιτη λειτουργικότητα.

**Ο παίκτης(player)**:

H κλάση παίκτης περιέχει όλες τις μεθόδους που υλοποιούν κάθε κίνηση του παίκτη κατά την διάρκεια του επιτραπέζιου, και ουσιαστικά δίνει σάρκα και οστά αλλά και νόημα σε όλο το παιχνίδι. Χωρίς αυτές τις μεθόδους και χωρίς την ύπαρξη του παίκτη δεν υπάρχει λόγος ύπαρξης του παιχνιδιού. Πέρα από τις βασικές μεθόδους setters και getters καθώς και τους constructors της κλάσης περιέχονται και οι ακόλουθες μέθοδοι :

This method is meant to be used as the means of rolling the player's dice.It generates a random number between 1 and 6.

public int Roll\_the\_dice(){return 0;};

This method is used to increase the bank account of the player

public void money\_addition(int money\_increase){

this.money+=money\_increase;

}

\*This method is used to decrease the bank account of the player

public void money\_subtraction(int money\_decrease){

this.money-=money\_decrease;

}

This method is meant to end a players turn and allow the next player to play

public void end\_turn(){};

This method is meant to be used to allow the player to get a loan.It should be mentioned that this is a loan by choice and not a forced loan(the case where the player doesn't have enough money to pay a bill).

public void get\_loan(int increase){};

This method allows the player to get a forced loan when he lacks the money to pay a bill.

public void forced\_loan(int insufficient\_funds ){};

This method allows the player to move his pawn through the board. This method combined with the dice roll are the building blocks of movement in this board game

public void movePosition(int x) {}

This method allows the player to add a deal card to his personal deck.

public void addCards(Card card) {

this.player\_cards.add(card);

}

This method allows the player to update his month(not on command) at the end of his round.

public void update\_month(int position){};

}

Είναι σημαντικό να αναφερθεί ότι ο παίκτης σαν οντότητα δεν είναι μόνο το πιόνι πάνω στο ταμπλό, αλλά περιέχει και όλα εκείνα τα γνωρίσματα του παίκτη που συντελούν στην ολοκλήρωση του παιχνιδιού οπως π χ οι κάρτες που έχει στην διαθεση του ή η δυνατότητά του να ρίχνει το ζάρι και να εκτελεί εντολές που του δίνονται.

**Η θέση(Position):**

Η θέση σε αυτή την περίπτωση δεν λειτουργεί ακριβώς κυριολεκτικά σαν θέση πάνω στο ταμπλό, αλλά περισσότερο ορίζει τις ενέργειες που εκτελούνται σε κάθε μία θέση του, πράγμα που την κάνει το πιο περίπλοκο set κλασεων και υποκλασεων στο παιχνίδι. Το set αυτο περιέχει δεκαπέντε κλάσεις (BothPlayerDicePosition, buyer, CardPosition, Casino, dela\_positon, DicePosition, lottery\_ticket, mail\_potition, OnePlayerDicePosition, PayDay, Position, Radio Contest, raffle, start, Yard Sale) εκ των οποίων οι πέντε έχουν οριστεί abstract καθώς “κουβαλάνε” μεθόδους τις οποίες πρέπει οι κλάσεις-παιδιά τους να κάνουν override(BothPlayerDicePosition,CardPosition, DicePosition, OnePlayerDicePosition και Position ). Πέρα από τις βασικές μεθόδους setters και getters καθώς και τους constructors των κλασεων και υποκλασεων περιέχονται και οι ακόλουθες μέθοδοι :

\*This is the method that executed the action on payday

void performAction(Player p){};

}

This is the method that allows the action to be executed when the player is on a dice position

abstract void performAction(Player p, int dice\_number);

}

This is the method that implements the action connected to this one player dice action tile

abstract void performAction(Player p ,int dice\_number);

This is the method that implements the action on a dice position that involves both players

abstract void performAction(Player p ,int dice\_number);

}

This is the method that allows the player to get the cards if he is on a card position

public abstract void getCards(Card c, Player p);

}

This method is meant to be used to enable the player to get the mail cards. He picks up the amount given by pick number

public void getCards(Card c, Player p, int pick\_number){};

## Η Σχεδίαση και οι Κλάσεις του Πακέτου Controller

Στην συνέχεια θα επικεντρωθούμε στον Controller. Αυτό το πακέτο κλάσεων αποτελεί ουσιαστικα τον συνδετικό κρίκο μεταξύ του Model και του View και θα είναι αυτό που θα δώσει ζωή στο παιχνίδι μας. Ουσιαστικά σε αυτο το σημείο του κώδικα συναντάται το front end του παιχνιδιού με το back end. Εδώ γίνεται η απόδοση των λειτουργιών που θα εκτελούνται απο το καθε κουμπί αντιστοιχα της γραφικής διεπαφής(GUI). Eδώ ορίζονται και αρχικοποιούνται τα View κι Model και αυτο το κομμάτι του κωδικα θα κληθεί να χειριστεί ο χρήστης για να παίξει το παιχνίδι, εξού και το όνομα controller(μερικώς απο αυτο και απο το γεγονός ότι ο κώδικας που γράφεται εδω ελέγχει και τα υπόλοιπα μέρη του παιχνιδιού). Αυτός αποτελείται, πέρα από τις βασικες μεθόδους setters και getters και τον αντίστοιχο constructor της κλάσεις από τις ακόλουθες μεθόδους:

This method is given to us in order to be able to read from the CSV files.

public String[][] readFile(String path, String type)

This method is meant to be used to generate a random turn and give it to a player

public void generate\_turn()

This method is meant to be used in order to check if the game has finished or not

public boolean checkIfGameFinished()

This method is meant to be used to execute the action for the player in the certain position he is in

public void actionOnPosition(int position, Player p)

This method is meant to be used to implement the effect of a mail card on the player

public void MailCardEffect(Player p)

Επισης στο πακέτο κώδικα αυτό, ορίζονται και κάποιες κλασεις που κληρονομούν την κλάση ActionListener. Αυτές οι κλάσεις θα δίνουν ζωή στα κουμπιά στην δεύτερη φάση του πρότζεκτ. Για τωρα έχουν οριστεί σαν abstract για να μπορούν να μείνουν με άδειο σώμα.

## Η Σχεδίαση και οι Κλάσεις του Πακέτου View

Τέλος, το πακέτο View περιέχει τον βασικό κώδικα για την δημιουργία του GUI του επιτραπέζιου μας. Αυτό θα μας επιτρέψει να αλληλεπιδράμε με το παιχνιδι (οχι σε εμας αλλα στους παικτες :) ). Περα απο τις βασικές μεθόδους setters και getters και τους απαραιτητους constructors , στο πακέτο αυτό βρίσκονται και οι ακόλουθες μέθοδοι:

This method is meant to be used to show the deal cards of the game

public void showDealCard(int i, String dealCards[][])

This method is meant to be used in order to show the mail cards of the game

public void showMailCard(int i,String mailCards[][])

This method is meant to be used to show the pop-up for the cryptocurrency event of the game

public void showCrypto()

This method is meant to be used to show the pop-up for the football event of the game

public void showFootballGame()

This method is meant to be used in order to be triggered after the cryptocurrency event occurs and show the pop-up

public void chain\_reaction\_crypto()

This method is meant to be used in order to be triggered after the football event occurs and show the pop-up

public void chain\_reaction\_football()

This method is meant to be used as a way of updating the number of money in the jackpot visually

public void update\_jackpot()

This method is meant to be used as a visual pop-up for the start of the game in order for players to be able to choose the months aka the total rounds that they are going to play

public void choose\_month()

This method is meant to be used to update the visual representation of the dice on the board

public void dice\_update(int dice\_roll, Player p)

This method is meant to be used in order to initialize each player's box of information

public void set\_player\_box(){};

This method is meant to be used to initialize the cards of the game

public void initCards(ArrayList<Card> cards) {}

This method is meant to be used to update the pawns of each player after a certain movement

public void updatePawn(int position) {};

This method is meant to be used to update the information inside the player's box

public void updateInfobox(String message,Player p) {}

This method is meant to be used top show the winning player after the game has finished

public void showWinningMessage(String message) {

}

This method is meant to be used to create a visual menu for the get loan button when the player willingly gets a loan .This menu is going to present him some choices and he is going to be able to choose one

public void get\_loan\_menu(){};

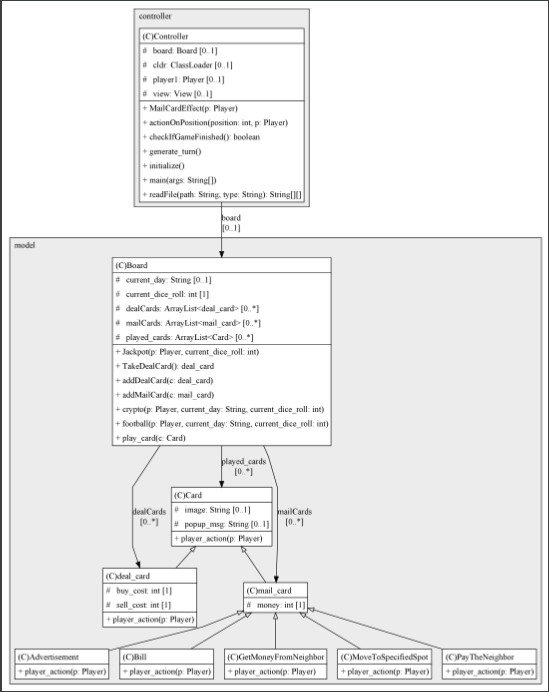
This method is meant to be used as a visual pop-up for the choice of the number of the raffle

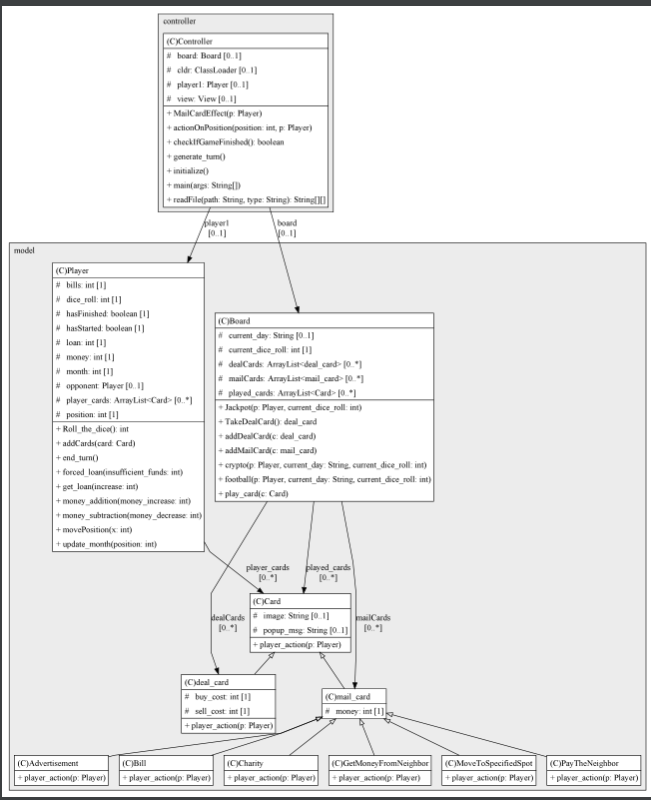
public int raffle\_choice\_generator(Player p){return 0;};

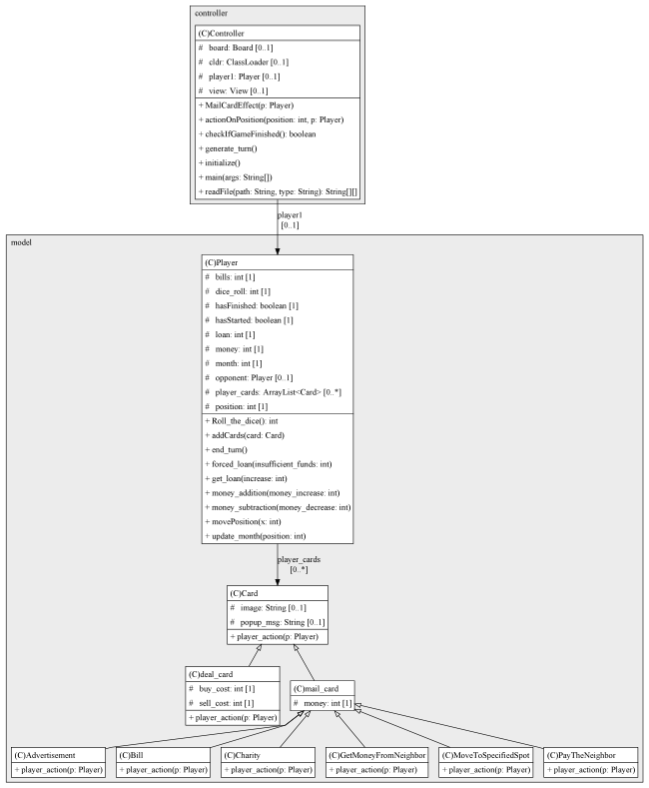
}

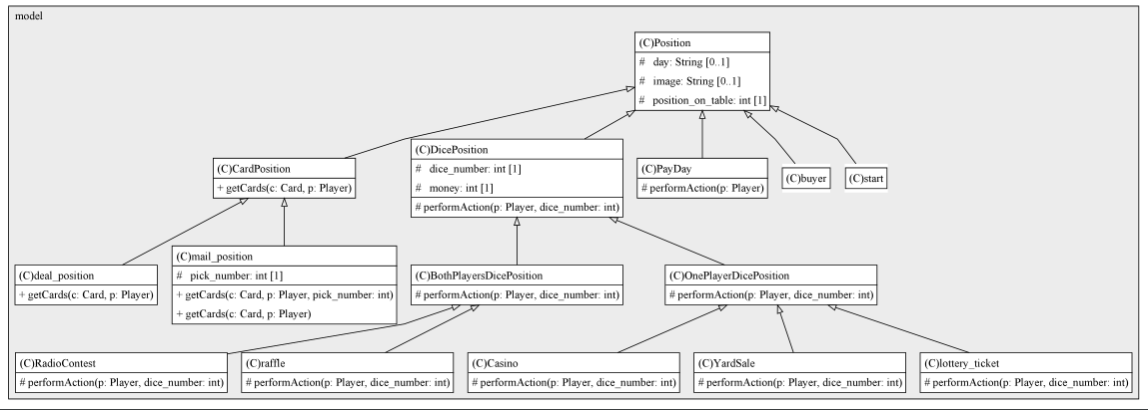
## Η Αλληλεπίδραση μεταξύ των κλάσεων – Διαγράμματα UML

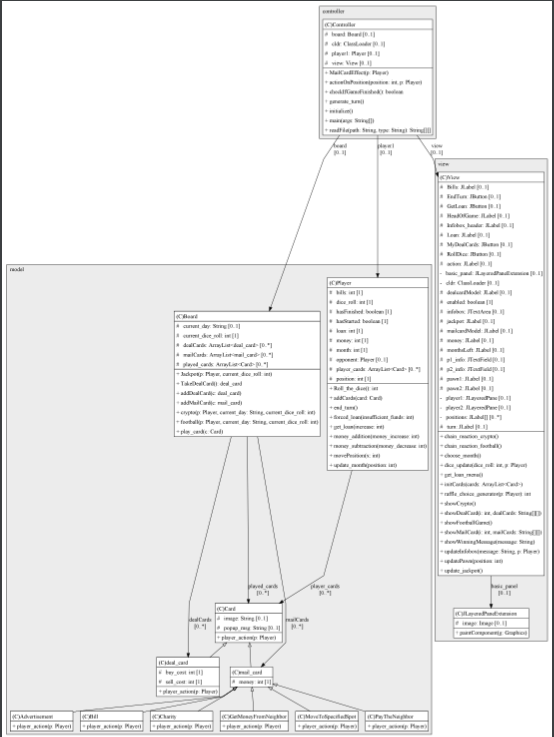
Σε αυτο το σημείο παρέχονται κάποια διαγράμματα UML τα οποία απεικονίζουν όλα τα παραπάνω και φανερώνουν την δομή του κάθε πακέτου

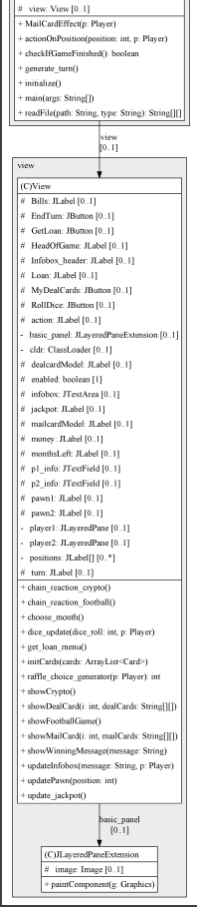
**

**

**

**

**

**

(Τα παραπάνω περιέχονται και στον πηγαίο κώδικα).

## Λειτουργικότητα (Β Φάση)

Στην υλοποίηση της δεύτερης φάσης ακολουθήθηκε πιστά , όσο αυτό γινόταν, τα πλάνα της πρώτης φάσης του παιχνιδιού. Δεν υπάρχει ιδιαίτερη διαφοροποίηση του κώδικα από τις ήδη δηλωμένες μεθόδους και λειτουργίες που είχαν οριστεί στην πρώτη φάση. Το μόνο που άλλαξε ειναι οτι απο τον κώδικα αφαιρέθηκαν πολλές από τις αρχικές μέθοδοι , οι οποίες κατα την διάρκεια της ανάπτυξης του παιχνιδιού κρίθηκαν είτε περιττές(στην πλειοψηφία) είτε λάθος. Μαζί με το πηγαίο κωδικα, υπαρχει ενα readme file στο οποίο αναγράφονται οδηγίες για το τρέξιμο του προγράμματος,καθως και οτιδήποτε παρεκκλίνει από τα ζητούμενα της εκφώνησης σε συνάρτηση με την υλοποίηση του προγράμματος. Ακολουθούν τα τελικά σχεδιαγράμματα UML τα οποία και μαρτυρούν την δομή του κώδικα στο τέλος της γραφής του(Τα οποία δεν γίνονται επισύναψη καθώς συνεχώς crashαρει ο loader τους για αυτο και τα εμπεριέχω όλα μαζί με τον πηγαίο κώδικα σε ενα package .)

## 7.Συμπεράσματα

Θεωρώ ότι η Α Φάση του project, παρόλο που αποτελεί βασική αρχή για την υλοποίηση του προγράμματος και σίγουρα δυνατό θεμέλιο για τον ορθό προγραμματισμό και καταμερισμό τον έργων, σίγουρα θα υποστεί πολλές αλλαγές στην πορεία και κατα την υλοποίηση της Β Φάσης του project, οπότε σίγουρα πολλα από αυτά που παρουσιάζονται σε αυτή την αναφορά θα αλλάξουν είτε θα προστεθούν είτε και ακόμα θα εξαφανιστούν.

Στην φάση Β του πρότζεκτ προσωπικά αντιμετώπισα αρκετά προβλήματα κυρίως με την γραφική αναπαράσταση του παιχνιδιού για αυτό και μερικές λειτουργίες πιθανώς να λείπουν από το παιχνίδι. Ιδιαίτερης δυσκολίας αποδείχθηκε ο χειρισμός του πιονιού, κυρίως κατα το γραφικό του κομμάτι. Τα κουμπιά αφήνονται σκόπιμα ελεύθερα στον χρήστη (εκτός από αυτά των επιλογών καρτών) για να υπάρχει μια αίσθηση grief στο παιχνίδι. Οπως και στην πραγματική ζωή , έτσι και εδώ οι δύο παίκτες καλούνται να παίξουν υπευθυνα χωρίς να καταστρέφουν την ολη εμπειρία για τον αντίπαλό τους. Μερικά features όπως τα παράθυρα διαλόγων μερικών από τα special events του παιχνιδιού φαίνεται μερικές φορές να χαλάνε χωρίς προφανή λόγο.