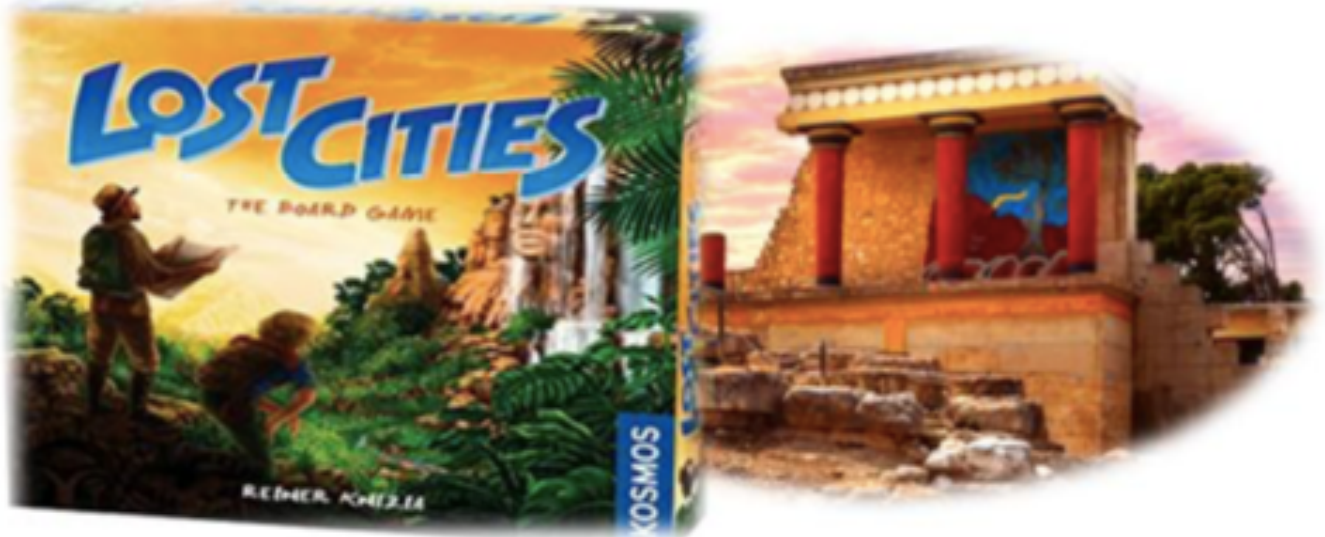

Αντικειμενοστραφής Προγραμματισμός ΗΥ-252

Project 2017

Ζερβός Γεώργιος

AM:3384

Ηράκλειο - 11 December 2017



Ενότητες:

- Εισαγωγή. Σελ. 2
- Package Model Σελ. 2-9
 - Card Interface & Card Classes. Σελ. 2-3
 - Class Position Σελ. 4
 - Class Pawn Σελ. 5
 - Finding Interface & Finding classes. Σελ. 6-7
 - Class Board Σελ. 8
 - Class PlaceType Σελ. 8
 - Class Player Σελ. 9
- Package Controller Σελ. 10-11
 - Class Controller. Σελ. 10-11
- Package View Σελ. 11

Εισαγωγή-Λίγα λόγια για την εργασία

Η εργασία θα σχεδιαστεί εξ ολοκλήρου πάνω στο μοντέλο MVC(Model View Controller). Το model θα είναι το κομμάτι της λειτουργίας του παιχνιδιού το view το κομμάτι του γραφικού περιβάλλοντος και το controller το μέρος που θα συνδέει το model με το view ώστε να αλληλεπιδρούν μαζί.

Package Model

Σε αυτό το πακέτο περιέχονται η διεπαφή Card, οι κλάσεις NumberCard και SpecialCard που υλοποιούν την Card, οι κλάσεις AriadneCard και MinotaurCard που κληρονομούν την Special Card, τη κλάση Position, τις κλάσεις SimplePosition και FindingPosition που κληρονομούν την Position, η κλάση Player, η κλάση Board, η διεπαφή Finding, οι κλάσεις που την υλοποιούν, Wallpictures, Statues και RareItems και οι 4 κλάσεις που κληρονομούν την RareItems, η κλάση Pawn και η κλάση Theseus που κληρονομεί την Pawn...

Card Interface & Card Classes:

Με τη διεπαφή Card υπάρχει το πλεονέκτημα ότι μπορούμε να φτιαξουμε ένα deck με κάρτες χωρίς να αναγκαστούμε να το χωρίσουμε σε Number Cards και σε special Cards.

Το interface παρέχει τις εξής μεθόδους:

1. `public PlaceType getCardtype(); Accessor(selector)`
Returns the type of the Card.
2. `public void setType(String type); Transformer (Mutative)`
Sets the type of the Card.
3. `public abstract String toString();`
Returns a String Representation of the Card.

Class NumberCard:

Attributes και επιπλέον μεθόδους εκτός της διεπαφής Card.

Τα attributes:

1. `private int number; //The number of the Number Card`
2. `private PlaceType cardtype; //The Type of the Number Card`

Οι υπόλοιπες μέθοδοι:

1. `public int getNumber(); Accessor(selector)`
Returns the number of the Card.
2. `public void setNumber(int number); Transformer (Mutative)`
Set Cards number.

Class SpecialCard:

Attributes και επιπλέον μεθόδους εκτός της διεπαφής Card.

Τα attributes:

1. private String name; //The name of the Number Card
2. private PlaceType cardtype; //The Type of the Number Card

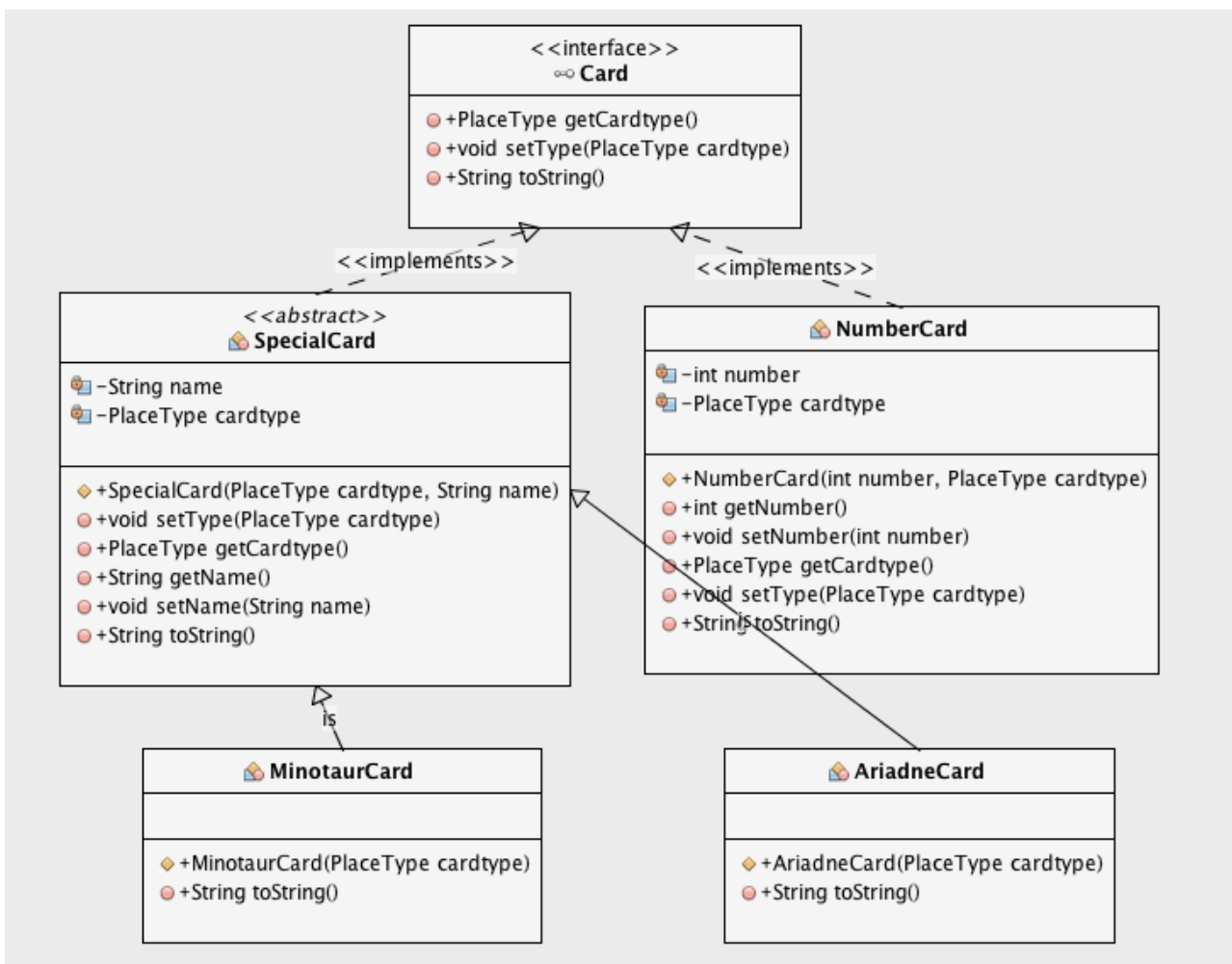
Οι υπόλοιπες μέθοδοι:

1. public String getName(); Accessor(selector)
Returns the name of the Card.
2. public void setName(String name); Transformer (Mutative)
Set Cards name.

Classes AriadneCard,MinotaurCard:

Αυτές οι κλάσεις επεκτείνουν την Special Card μέσω της εντολής super αποκτούν πρόσβαση στην κλάση Special Card και αρχικοποιούν τα πεδία name και cardtype.

Δίνεται η αναπαράσταση των κλάσεων που έχουν σχέση με τις κάρτες μέσω UML:



Class Position:

Η κλάση αυτή θα είναι το κάθε βήμα ενός πιονιού επάνω στο ταμπλό.

Τα attributes της Position:

1. private PlaceType placetype; //The Type of this Position
2. private int points; //The Points of this Position
3. private Pawn Pawn1; //Player 1's pawn on this Position
4. private Pawn Pawn2; //Player 2's pawn on this Position

Οι μέθοδοι της:

1. public PlaceType getPlaceType(); Accessor(selector)
Returns the type of this Position.
2. public void setPoints(int points); Transformer(Mutative)
Set Position's points.
3. public int getPoints(); Accessor(selector)
Returns the points of this Position.
4. public String toString(int i)
Returns the string representation of a specific position depending on the place type.

Class Simple Position:

Αυτή η κλάση επεκτείνει την Position και μέσω της εντολής super αποκτούν πρόσβαση στην κλάση Position και αρχικοποιούν τα πεδία placetype και points.

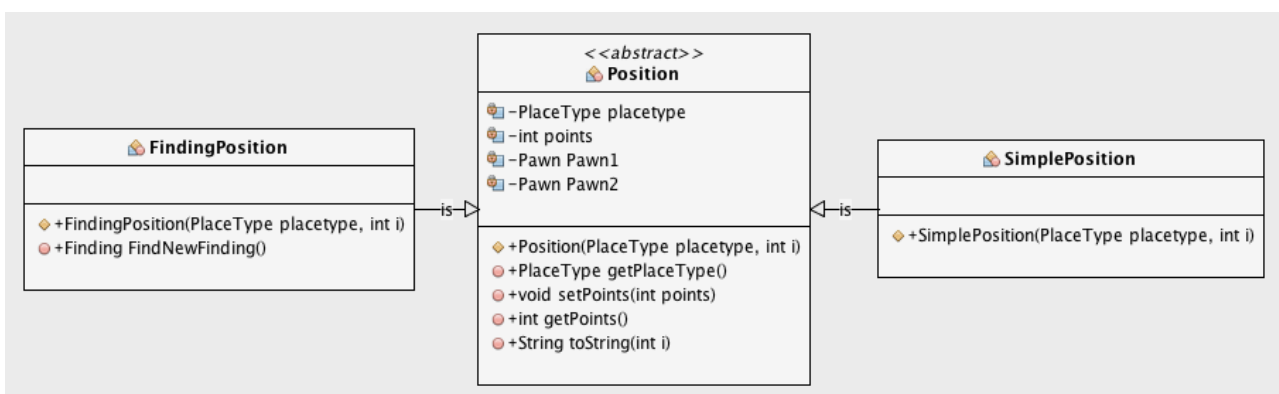
Class Finding Position:

Αυτή η κλάση επεκτείνει την Position και μέσω της εντολής super αποκτούν πρόσβαση στην κλάση Position και αρχικοποιούν τα πεδία placetype και points. Σε αυτή τη θέση επιπλέον ο παίχτης θα ανακαλύπτει ένα τυχαίο σπάνιο εύρημα.

Επιπλέον μέθοδοι:

1. public Finding FindNewFinding(); Accessor(selector)
Returns a new finding

Δίνεται η αναπαράσταση των κλάσεων που έχουν σχέση με τα Positions μέσω UML:



Class Pawn:

Αυτή η κλάση είναι το πiónι του παίχτη στο ταμπλό. Θα μετακινείται από τη μια θέση σε άλλη σε κάθε μονοπάτι. Θα συλλέγει τους πόντους ώστε να τους λάβει ο παίχτης σε κάθε βήμα του.

Τα attributes της Pawn:

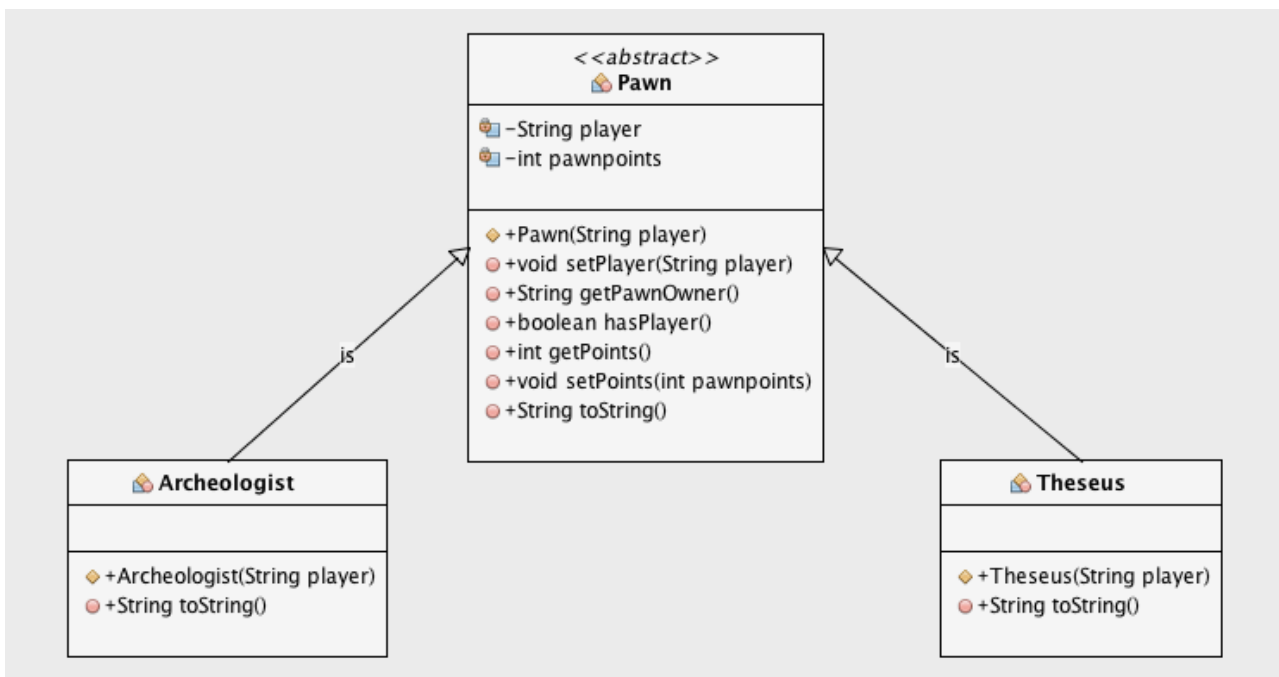
1. `private final String player; // The player of this pawn`
2. `private int pawnpoints; // The points that pawn collected`

Οι μέθοδοι της:

1. `public void setPlayer (String player); Transformer(Mutative)`
Set Player to pawn.
2. `public String getPawnOwner(); Accessor(selector)`
Returns the owner of this Pawn.
3. `public boolean hasPlayer(); Observer`
Return true if pawn has a player, false otherwise.
4. `public int getPoints(); Accessor(selector)`
Returns the points of this Pawn
5. `public void setPoints(int pawnpoints) Transformer(Mutative)`
Set points to pawn.
6. `public String toString();`
Returns the string representation of a Pawn

Classes Archeologist, Theseus:

Αυτές οι κλάσεις επεκτείνουν την Pawn μέσω της εντολής `super` αποκτούν πρόσβαση στην κλάση Pawn και αρχικοποιούν το πεδίο `player`.



Δίνεται η αναπαράσταση των κλάσεων που έχουν σχέση με τα Pawns μέσω UML:

Finding Interface & Finding classes:

Με τη διεπαφή Finding υπάρχει το πλεονέκτημα ότι μπορεί να έχει μια λίστα από τα αποκτημένα ευρήματα χωρίς να χρειαστεί να τα χωρίσουμε σε 3 λίστες ανάλογα το είδος το ευρήματος.

Το interface παρέχει τις εξής μεθόδους:

1. `public int getPoints();` Accessor(selector)
Returns the points of this finding.
2. `public void setPoints(int points);` Transformer(Mutative)
Set finding's points.
3. `public String toString();`
Returns the string representation of a finding.

Classes Statues & WallPictures:

Οι δύο αυτές κλάσεις δεν έχουν κάποια επιπλέον μέθοδο

Τα attributes τους:

1. `private int points;` //The points of the finding

Class RareFinding:

Attributes και επιπλέον μεθόδους εκτός της διεπαφής Finding.

Τα attributes:

1. `private int points;` //The points of the finding
2. `private boolean taken;` // Check if the item is taken

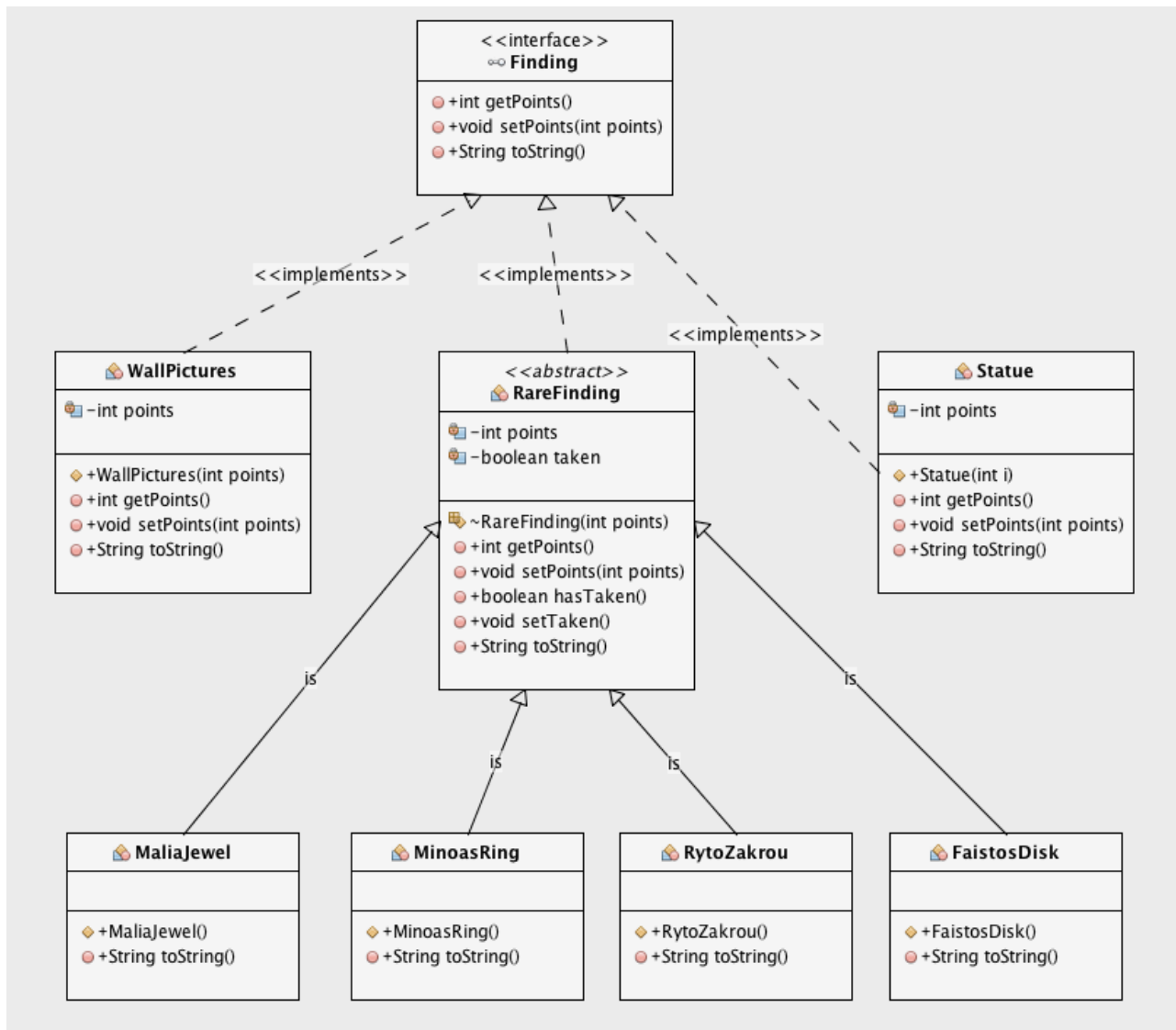
Οι υπόλοιπες μέθοδοι:

1. `public boolean hasTaken();` Observer
Return true if this item has taken from the other player, false otherwise
2. `public void setTaken();` Transformer(Mutative)
Set that this item is taken.

Classes FaistosDisc,KnososRing,MaliaJewel,RytoZakrou:

Αυτές οι κλάσεις επεκτείνουν την RareFinding μέσω της εντολής `super` αποκτούν πρόσβαση στην κλάση RareFinding και αρχικοποιούν το πεδίο `points`.

Δίνεται η αναπαράσταση των κλάσεων που έχουν σχέση με τα Finding μέσω UML:



Class Board:

Η κλάση αυτή απεικονίζει ουσιαστικά το ταμπλό του παιχνιδιού, με τη τράπουλα με τις κάρτες, τα 4 διαφορετικά μονοπάτια που πατάνε τα πιόνια του παίχτη και όλα τα ανεξερεύνητα ευρήματα που βρίσκονται σε συγκεκριμένες θέσεις σε κάθε μονοπάτι.

Τα attributes:

1. private ArrayList<Card> deck ; //The deck with the 100 cards
2. public Position[] PosK,PosF,PosM,PosZ; //The 4 different positions on the board
3. private ArrayList<WallPictures> wallpictures ; //The list of the remain WallPictures
4. private int Statues; //The number of the remaining undiscovered Statues
5. private RareFinding DF,MR,MJ,RZ; //The 4 different Rare Items

Οι μέθοδοι:

1. public void setAndShuffleDeck(); Transformer(Mutative)
Set a new deck of 100 new cards and shuffle them.
2. public Card pickCard(); Accessor(selector)
Remove and Return the top card of the deck to add it on the player's hand.
3. public boolean isDeckEmpty(ArrayList<Card> deck). Observer
Return true if the is empty, false otherwise.
4. public int getRemainStatues(); Accessor(selector)
Returns the remaining statues on the board.
5. public void setRemainStatues(); Transformer(Mutative)
Set the remain undiscovered statues.
7. public boolean isStatuesEmpty(); Observer
Returns true if the number of the remaining statues is 0, false otherwise.
8. public WallPictures pickaWallPicture(); Accessor(selector)
Returns a random WallPicture from the remaining list.
9. public boolean hasRareitemTaken(PlaceType placetype); Observer
Returns true if the item is taken, false otherwise.
10. public void setRareItemTaken(PlaceType placetype); Transformer(Mutative)
Set the RareItem taken.

Class PlaceType:

Είναι μια enum class που περιέχει τις 4 διαφορετικές τοποθεσίες του παιχνιδιού (KNOSOS,FAISTOS,MALIA,ZAKROS)

Class Player:

Η κλάση αντιπροσωπεύει έναν παίχτη ο οποίος έχει στην κατοχή του ένα χέρι με 8 κάρτες, μια λίστα με τα ευρήματα που έχει ανακαλύψει καθώς και τους πόντους που κερδίζει.

Τα attributes:

1. private ArrayList<Card> playersHand ; //The cards player has on his hand
2. private ArrayList<Finding> findings ; //The list of the player's findings
3. private Pawn Pn1,Pn2,Pn3,Pn4; //The 4 player's pawns
4. private int points=0; // Player's points
5. private int checkpoints=0; // Player's number of checkpoints
6. private final String name; // Player's name

Οι μέθοδοι:

1. public void setPoints(int points); Transformer(Mutative)
Set Players points.
2. public int getPoints(); Accessor(selector)
Returns the points of the player.
3. public String getName(); Accessor(selector)
Returns the name of the player.
4. public boolean HasCards(); Observer
Return true if the Player's hand is empty, false otherwise.
5. public void DrawCard(Card card); Transformer(Mutative)
Add a card in player's hand.
6. public ArrayList<Card> getCardinHand(int i); Accessor(selector)
Returns a selected card.
7. public boolean isHandFull(); Observer
Return true if the Player's holds 8 cards, false otherwise.
8. public void addCheckpoint(); Transformer(Mutative)
Add a checkpoint on this player.
9. public int getCheckpoints(); Accessor(selector)
Returns the checkpoints of the player.

Package Controller

Class Controller

Η κλάση αποτελεί τον πυρήνα όλων των λειτουργιών του παιχνιδιού, καθώς είναι υπεύθυνη για όλες τις ενέργειες των παιχτών και του ταμπλό, όπως τις κινήσεις που κάνουν οι παίκτες, τους ελέγχους που πρέπει να γίνουν σε κάθε γύρο, ποιος παίκτης έχει σειρά, πότε τελειώνει το παιχνίδι, καθώς και το ποιος είναι ο νικητής.

Τα attributes:

1. private Board board; // The board of the game
2. private Player P1,P2; // The 2 Players of the game
3. private int turn; // Player's turn (1 for Player1, 2 for Player 2)
4. private int checkpoints; // The total checkpoints of both players

Οι μέθοδοι:

1. public int seeTurn(); Accessor(selector)
Returns the turn number.
2. public void SetNextTurn(int turn); Transformer(Mutative)
Set the next turn.
3. public void PlacePawn(); Transformer(Mutative)
Set a pawn on a position.
4. public void movePawn(); Transformer(Mutative)
Move a pawn from a position to another.
5. public void setCardstoHand(); Transformer(Mutative)
Set 8 cards from the deck into players hands.
6. public void playCard(int i); Transformer(Mutative)
Play a card from players hand.
7. public int ChooseCardtoPlay(); Accessor(selector)
Returns the number of a specific card on the hand to play.
8. public void addStatuetoPlayer(); Transformer(Mutative)
Adds a statue in a player's finding list.
9. public void addWallPicturetoPlayer(); Transformer(Mutative)
Adds a Wall Picture in a player's finding list.
10. public void addRareItemtoPlayer(PlaceType placetype);
Add a Rare Finding in a player's finding list, depending on the place type and if the item is taken or not.

-
11. `public int collectPoints();` `Accessor(selector)`
Returns the total player's points
 12. `public boolean checkIfCheckpoint();` `Observer`
Return true if it is a checkpoint from a player, false otherwise.
 13. `public void setCheckPoints();` `Transformer(Mutative)`
Set the checkpoints of both players.
 14. `public boolean isGameFinished();` `Observer`
Returns true if game is finished, false otherwise.
 15. `public int seeWinner();` `Accessor(selector)`
Returns the winner of the game(1 for player1 , 2 for player2).
 16. `public URL cardurl(Card card,int i)` `Accessor(selector)`
Returns the url for the card images.
 17. `public URL numurl(PlaceType pl,int i)` `Accessor(selector)`
Returns the url for the number card images.
 18. `public URL PawnURL(Pawn pawn)` `Accessor(selector)`
Returns the url for the pawn images.
 19. `public boolean collectNewFinding(PlaceType placetype,int player)` `Observer`
Returns true if successfully collected a new finding, false otherwise.

Package View

Σε αυτό το πακέτο περιέχεται μια κλάση που δημιουργεί ένα frame και υλοποιεί την ActionListener για κάποιες ενέργειες που θα γίνονται μέσα σε αυτό.

Το frame αυτό περιέχει ένα βασικό panel όπου μέσα σε αυτό περιέχονται 3 layered Panes , 2 πάνω και κάτω για τους δύο παίκτες και ένα για το ταμπλό. Για τα layered Panes χρησιμοποίησα μια κλάση η οποία τα επεκτείνει για να μπορέσω να προθέσω την background εικόνα του ταμπλό.

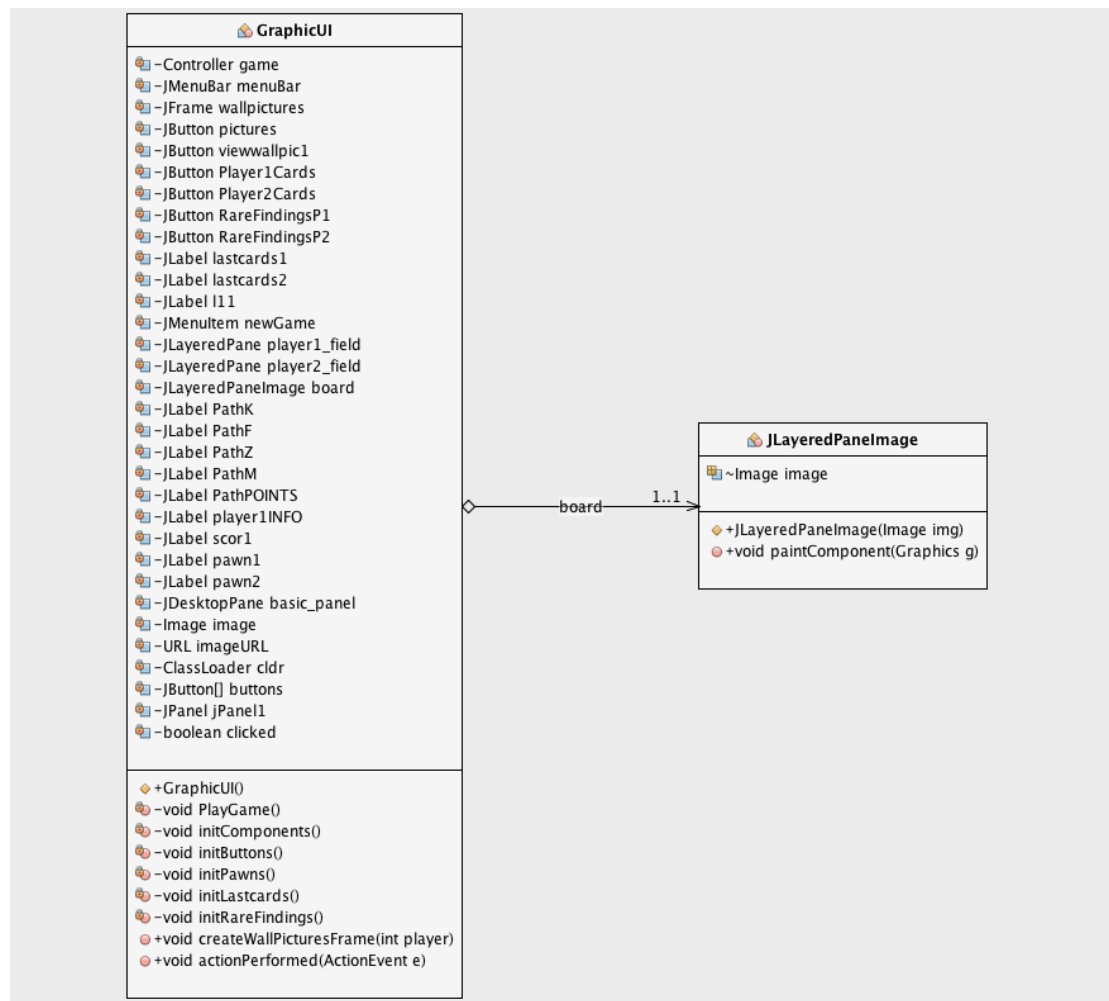
Στα 2 layered panes των παιχτών πρόσθεσα για κάθε παίκτη, 8 buttons για τις κάρτες, 1 button που θα ανόγει ένα νέο frame για να δείξει τις τοιχογραφίες του κάθε παίκτη, 3 labels για τις πληροφορίες όσο αναφορά τα πιόνια, τα αγαλματίδια και το σκορ του παίκτη, 4 labels που δείχνουν την τελευταία κάρτα που έπαιξε ο παίκτης για κάθε τοποθεσία καθώς και 4 buttons για τα 4 σπάνια ευρήματα τα οποία δεν κανουν καποια ενέργεια απλώς, ανάλογα αν τα έχει στη κατοχή του ο παίκτης γίνονται enable ή disable.

Στο layered pane του ταμπλό πρόσθεσα ένα button με το οποίο θα τραβάει μια νέα κάρτα ο παίκτης, ένα label το οποίο θα δείχνει πληροφορίες όσο αναφορά τις υπόλοιπες κάρτες της τράπουλας, τα checkpoints και το ποιος παίζει σε αυτόν τον γύρο. Πρόσθεσα 36 labels για τα μονοπάτια τα οποία με τη σειρά τους μπορούν να πάρουν το καθένα μέχρι 2 labels που αντιστοιχούν στα πιόνια των παιχτών. Αλλά 9 labels ήρθαν με τη σειρά τους για να δείχνουν τους πόντους για κάθε μονοπάτι.

Στην βασική κλάση του view (GraphicUI) εκτός από τον constructor και την μέθοδο που υλοποιεί ο Action Listener, περιέχονται οι εξής μέθοδοι:

- private void PlayGame() ; Η μέθοδος αυτή εξελίξει την πορεία του παιχνιδιού και ανανεώνει καποια πεδία που χρειάζονται αλλαγή .
- private void initComponents() ; Η μέθοδος αυτή δημιουργεί τα σχεδόν όλα τα μέρη του GUI(buttons, labels, layered panes κτλ.).
- private void initButtons(); κάθε φορά που καλείται αυτή η μέθοδος ανανεώνει τις εικόνες των buttons για τις κάρτες των παιχτών.
- private void initPawns() ; κάθε φορά που καλείται αυτή η μέθοδος ανανεώνει τις εικόνες των labels για τα πιόνια των παιχτών πάνω στις θέσεις του ταμπλό.
- private void initLastcards() ; κάθε φορά που καλείται αυτή η μέθοδος ανανεώνει τις εικόνες των labels για τις τελευταίες κάρτες που έπαιξαν οι παίκτες.
- private void initRareFindings(); κάθε φορά που καλείται αυτή η μέθοδος ανανεώνει τις εικόνες των buttons των ευρημάτων που κέρδισαν οι παίκτες.
- public void createWallPicturesFrame(int player); Η μέθοδος αυτή δημιουργεί ένα νέο frame το οποίο δείχνει τις τοιχογραφίες που κέρδισαν οι παίκτες.

Δίνεται η αναπαράσταση των κλάσεων που έχουν σχέση με το view μέσω UML:



ΤΕΛΟΣ ΑΝΑΦΟΡΑΣ