

| | |
|---|--|
| Συγγραφέας: < <i>Project Genesis</i> > | Κωδικός: < <i>I/Project Genesis /1.0</i> > |
| Έγκριση: < <i>Κουτσουσίμος Βαγγέλης</i> > | Ημερομηνία: < <i>23-05-2013</i> > |

<IT Creations>

< ***Jungle WarriorZ – Project Genesis*** >

Έγγραφο Περιγραφής Σχεδίου Λογισμικού

| | |
|---|--|
| Συγγραφέας: < <i>Project Genesis</i> > | Κωδικός: < <i>I/Project Genesis /1.0</i> > |
| Έγκριση: < <i>Κουτσουσίμος Βαγγέλης</i> > | Ημερομηνία: < <i>23-05-2013</i> > |

Ιστορικό

| Ημερομηνία | Έκδοση | Περιγραφή | Συγγραφέας |
|--------------|--------|--------------------------|----------------|
| <23-05-2013> | <1.0 > | << Ολοκλήρωση εγγράφου > | < Giannoutas > |
| | | | |
| | | | |
| | | | |

| | |
|---|---|
| Συγγραφέας: < <i>Project Genesis</i> > | Κωδικός: <1/ <i>Project Genesis</i> /1.0> |
| Έγκριση: < <i>Κουτσουσίμος Βαγγέλης</i> > | Ημερομηνία: < < 23-05-2013> |

Πίνακας Περιεχομένων

| | |
|---|-----------|
| 1. Εισαγωγή | 4 |
| 1.1 Σκοπός | 4 |
| 1.2 Ορισμοί, Ακρωνύμια και Συντομογραφίες | 4 |
| 1.3 Αναφορές | 4 |
| 1.4 Επισκόπηση | 4 |
| 2. Σχέδιο Δεδομένων | 4 |
| 2.1 Βάσεις Δεδομένων | 4 |
| 2.2 Αρχεία | 5 |
| 2.3 Δομές Δεδομένων | 5 |
| 3. Σχέδιο Μονάδων | 5 |
| 4. Περιγραφή Διασυνδέσεων | 13 |
| 5. Παραρτήματα | 21 |
| 5.1 Διαγράμματα Κλάσεων | 21 |
| 5.2 Πίνακας Ιχνηλάτισης | 23 |
| 5.3 Διαγράμματα Αναφορών και Οθονών | 23 |
| 5.4 Διαγράμματα Οντοτήτων - Συσχετίσεων | 23 |

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

1. Εισαγωγή

1.1 Σκοπός

Σκοπός του ΕΠΣΛ είναι ο καθορισμός της αρχιτεκτονικής σχεδίασης του λογισμικού Jungle Warriorz,, ενός 2D shooting game σε πλατφόρμα, δηλαδή των μονάδων από τις οποίες αποτελείται και των μεταξύ τους σχέσεων. Το συγκεκριμένο ΕΠΣΛ απευθύνεται στην ομάδα ανάπτυξης της εταιρείας ITCreations που θα πραγματοποιήσει την υλοποίηση, τον έλεγχο και τη συντήρηση του λογισμικού.

1.2 Ορισμοί, Ακρωνύμια και Συντομογραφίες

- KeyMap: Διαχείριση πλήκτρων χειρισμού του παιχνιδιού.
- Single Player : Είναι το παιχνίδι στο οποίο αναμένεται είσοδος μόνο από ένα παίκτη.
- Multi-Player : Είναι το παιχνίδι στο οποίο αναμένεται είσοδος από δύο παίκτες.
- Host Game : Η δημιουργία του server του παιχνιδιού
- H/Y : Ηλεκτρονικοί Υπολογιστές
- Server : Ο H/Y που είναι η πηγή όλων των γεγονότων του παιχνιδιού.
- Join Game : Η σύνδεση σε ένα server του παιχνιδιού.

1.3 Αναφορές

- Έγγραφο Περιγραφής Απαιτήσεων Λογισμικού (ΕΠΑΛ): I/Project Genesis/1.0
- ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΣΧΕΔΙΑΣΗ: UML, ΑΡΧΕΣ, ΠΡΟΤΥΠΑ ΚΑΙ ΕΥΡΕΤΙΚΟΙ ΚΑΝΟΝΕΣ "Σύγγραμμα, ΑΛΕΞΑΝΔΡΟΣ Ν. ΧΑΤΖΗΓΕΩΡΓΙΟΥ, 2005, ΕΚΔΟΣΕΙΣ ΚΛΕΙΔΑΡΙΘΜΟΣ ΕΠΕ
- <http://www.insomnia.gr/index>

1.4 Επισκόπηση

Στο σχέδιο δεδομένων περιγράφονται οι κύριες δομές δεδομένων που χρησιμοποιούνται καθώς και τα αρχεία που περιλαμβάνονται στο λογισμικό. Στο σχέδιο μονάδων γίνεται μια σύντομη περιγραφή των βασικών κλάσεων του λογισμικού καθώς και των μεθόδων τις οποίες περιλαμβάνουν. Στην περιγραφή διασυνδέσεων περιγράφονται όλες οι κλάσεις που σχετίζονται με την αλληλεπίδραση του λογισμικού με τον χρήστη. Στα παραρτήματα περιλαμβάνονται τα διαγράμματα που περιγράφουν την λειτουργικότητα του παιχνιδιού

2. Σχέδιο Δεδομένων

2.1 Βάσεις Δεδομένων

Δεν χρησιμοποιούμε καμία βάση δεδομένων

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < 1/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

2.2 Αρχεία

Στο λογισμικό υπάρχουν αρχεία ήχου (επέκταση .wav), αρχεία εικόνας (επέκταση .jpg, .png), και αρχεία κειμένου (επέκταση .txt), τα οποία αρχεία κειμένου βοηθάνε στην φόρτωση των επιπέδων (tiles, εχθροί, σημείο τερματισμού). Επίσης για την επιλογή χαρακτήρα, φόρτωση των δεδομένων χειρισμού του χαρακτήρα και κάποια άλλα δεδομένα για την λειτουργία του παιχνιδιού.

2.3 Δομές Δεδομένων

Χρησιμοποιούνται δομές δεδομένων τύπου ArrayList, Array και HashTable, για την αποθήκευση και διαχείριση μαζικών τύπων δεδομένων, όπως π.χ. bullets, tiles.

3. Σχέδιο Μονάδων

AnimatedSprite

Μέσα στην κλάση υπάρχουν όλες οι μέθοδοι που βοηθάνε στην δημιουργία animation για τους εχθρούς και τον ήρωα. Διατηρεί μέσα σε Arraylist όλες τις εικόνες που χρησιμοποιούνται για τα animations. Η κλάση περιέχει 3 ιδιωτικές μεθόδους. Στην ιδιωτική μέθοδο newdimensions, παίρνουμε τις μεγαλύτερες διαστάσεις μιας εικόνας. Οι άλλες 2 ιδιωτικές μέθοδοι, getimagewidth και getimageheight, παίρνουν το μήκος και το πλάτος της εικόνας.

- Μέθοδος **addFrame(Sprite image)**

Η μέθοδος αυτή δέχεται μια εικόνα και την προσθέτει στο τέλος ενός animation.

- Μέθοδος **play()**

Ξεκινάει το animation επί του αντικειμένου στο οποίο καλείτε.

- Μέθοδος **isPlaying(String animationName)**

Ελέγχει αν παίζει ένα animation του οποίου το όνομα παίρνει ως όρισμα.

- Μέθοδος **playFromFrame(int frame)**

Ξεκινάει ένα animation από ένα συγκεκριμένο frame το οποίο παίρνει ως όρισμα.

- Μέθοδος **goToFrame(int frame)**

Πηγαίνει σε συγκεκριμένο frame του animation.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

Background

Η κλάση αυτή κάνει εύκολη την τοποθέτηση των γραφικών του background του παιχνιδιού.

- **Μέθοδος Draw(Graphics graphics, Camera camera)**

Η μέθοδος αυτή δέχεται μια μεταβλητή τύπου Graphics μέσω της οποίας πραγματοποιούνται διάφορες ενέργειες όπως το rendering του παιχνιδιού, και μια μεταβλητή τύπου Camera που κρατάει τα όρια της οθόνης πάνω στην οποία ζωγραφίζονται τα γραφικά μέσω αυτής της μεθόδου.

Bullet

Η κλάση αυτή μοντελοποιεί τις σφαίρες που χρησιμοποιούνται στο παιχνίδι. Ένα αντικείμενο τύπου Bullet έχει δική του θέση(χ,ψ) στον κόσμο του παιχνιδιού στην οποία ζωγραφίζεται, μια εικόνα από την οποία αντιπροσωπεύεται και ταχύτητα με την οποία κινείται.

- **Μέθοδος getBounds()**

Δημιουργεί και επιστρέφει ένα νέο αντικείμενο τύπου Rectangle, που βοηθάει στον εντοπισμό συγκρούσεων μεταξύ των αντικειμένων που υπάρχουν στον κόσμο του παιχνιδιού, στην θέση (χ,ψ) του αντικειμένου επί του οποίου καλείτε με διαστάσεις αυτές του αντικειμένου.

- **Μέθοδος move()**

Με την κλήση αυτής της μεθόδου μετακινητέ ένα αντικείμενο τύπου Bullet στον κόσμο του παιχνιδιού.

Camera

Αυτή η κλάση έχει σχεδιαστεί με σκοπό την εύκολη διαχείριση των θέσεων όλων των αντικειμένων επί της οθόνης.

- **Μέθοδος follow(Character character)**

Η μέθοδος αυτή δέχεται ως όρισμα μια μεταβλητή τύπου Character και μας επιτρέπει να ακολουθούμε την κίνηση του αντικειμένου Character επί του οποίου καλείτε στον κόσμο του παιχνιδιού.

Character

Η μέθοδος αυτή χρησιμοποιείτε ως βάση για την δημιουργία των χαρακτήρων του παιχνιδιού και κληρονομείτε από τις ποιο ειδικευμένες κλάσεις Hero και Enemy που αναλύονται παρακάτω. Περιλαμβάνει γνωρίσματα όπως το όνομα, την θέση και την ταχύτητα ενός χαρακτήρα και μεθόδους για την ανανέωση αυτών γνωρισμάτων αλλά και για την επιβολή αλλαγών σε αυτά. Περιλαμβάνεται ένα στατικό ArrayList που περιέχει αντικείμενα τύπου Bullet που σχετίζονται με ένα αντικείμενο τύπου Character και κατ'επέκταση Hero.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

- **Μέθοδος update(float gameTime, Camera camera)**

Αυτή η μέθοδος δέχεται ως ορίσματα τον χρόνο του παιχνιδιού και μια μεταβλητή τύπου Camera και ανανεώνει τις τιμές των διάφορων γνωρισμάτων του χαρακτήρα επί του οποίου καλείτε.

- **Μέθοδος draw(Graphics graphics, Camera camera)**

Η μέθοδος αυτή δέχεται μια μεταβλητή τύπου Graphics μέσω της οποίας πραγματοποιούνται διάφορες ενέργειες όπως το rendering του παιχνιδιού, και μια μεταβλητή τύπου Camera που κρατάει τα όρια της οθόνης πάνω στην οποία ζωγραφίζονται τα γραφικά. Μέσω αυτής της μεθόδου ζωγραφίζονται τα γραφικά που έχουν σχέση με τον χαρακτήρα επί του οποίου καλείτε.

- **Μέθοδος move(int x, int y)**

Η μέθοδος αυτή δέχεται ως ορίσματα δύο ακέραιους χ,ψ και αλλάζει την θέση του αντικειμένου επί του οποίου καλείτε σύμφωνα με αυτούς.

- **Μέθοδος moveTo(int x, int y)**

Η μέθοδος αυτή δέχεται ως ορίσματα δύο ακέραιους χ,ψ και μετακινεί τον χαρακτήρα επί του οποίου καλείτε στην θέση με συντεταγμένες (χ,ψ).

- **Μέθοδος fire()**

Η μέθοδος αυτή καλείτε όταν ένας χαρακτήρα πυροβολάει. Όταν καλείτε δημιουργείται ένα νέο αντικείμενο τύπου Bullet και εισάγεται στο στατικό ArrayList που περιλαμβάνει όλα τα αντικείμενα Bullets που σχετίζονται με τον χαρακτήρα επί του οποίου καλούνται.

- **Μέθοδος die()**

Σκοτώνει τον χαρακτήρα επί του οποίου καλείτε με αποτέλεσμα να μην εμφανίζεται πλέον στη οθόνη.

- **Μέθοδος alive()**

Ελέγχει αν ένας χαρακτήρας είναι ζωντανός. Αν είναι επιστρέφει true αλλιώς false.

CollisionDetection

Στην κλάση αυτή ελέγχεται η ύπαρξη συγκρούσεων μεταξύ των χαρακτήρων και του κόσμου αλλά και μεταξύ δύο χαρακτήρων.

| | |
|---|--|
| Συγγραφέας: < <i>Project Genesis</i> > | Κωδικός: < <i>1/Project Genesis /1.0</i> > |
| Έγκριση: < <i>Κουτσουσίμος Βαγγέλης</i> > | Ημερομηνία: < <i>23-05-2013</i> > |

○ **Μέθοδος collisionWithFloor(Character character, Map map)**

Η μέθοδος αυτή δέχεται ως ορίσματα έναν χαρακτήρα και τον χάρτη του επιπέδου στο οποίο βρίσκεται ο παίκτης. Ελέγχεται αν ο χαρακτήρας βρίσκεται πάνω σε tiles τύπου solid αν όχι ο χαρακτήρας πέφτει.

○ **Μέθοδος collisionWithRightWall(Character character, Map map)**

Η μέθοδος αυτή δέχεται ως ορίσματα έναν χαρακτήρα και τον χάρτη του επιπέδου στο οποίο βρίσκεται ο παίκτης. Ελέγχεται αν ο χαρακτήρας μπορεί να κινηθεί προς τα δεξιά. Αν κάτι τον εμποδίζει τότε σταματάει.

○ **Μέθοδος collisionWithLeftWall(Character character, Map map)**

Η μέθοδος αυτή δέχεται ως ορίσματα έναν χαρακτήρα και τον χάρτη του επιπέδου στο οποίο βρίσκεται ο παίκτης. Ελέγχεται αν ο χαρακτήρας μπορεί να κινηθεί προς τα αριστερά. Αν κάτι τον εμποδίζει τότε σταματάει.

○ **Μέθοδος collisionWithEnemy(Hero hero, Enemy enemy)**

Αυτή η μέθοδος δέχεται ως ορίσματα δύο χαρακτήρες, έναν τύπου Hero και έναν τύπου Enemy και ελέγχει για μεταξύ τους συγκρούσεις και επιστρέφει την κατάλληλη τιμή.

○ **Μέθοδος collisionWithEndOfLevel(Hero hero, Goal endOfLevel)**

Αυτή η μέθοδος δέχεται ως ορίσματα έναν χαρακτήρα τύπου Hero και ένα αντικείμενο τύπου Goal που δηλώνει το τέλος του επιπέδου και ελέγχει αν ο χαρακτήρας έχει φτάσει στο τέλος του επιπέδου ή όχι και επιστρέφει την κατάλληλη Boolean τιμή.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

Enemy

Κληρονομεί την κλάση Character.

Με την κλάση αυτή μοντελοποιούνται οι εχθροί που υπάρχουν μέσα στο παιχνίδι. Περιλαμβάνει γνωρίσματα όπως την αρχική θέση ενός εχθρού, την απόσταση που έχει περπατήσει καθώς και αυτά που κληρονομεί από την κλάση Character και έχουν ήδη αναφερθεί. Περιέχει δύο στατικές μεθόδους τις createEnemy1 και createEnemy2 με τις οποίες δημιουργούνται δύο διαφορετικά είδη εχθρών.

- Μέθοδος update(float gameTime, Camera camera)

Αυτή η μέθοδος δέχεται ως ορίσματα τον χρόνο του παιχνιδιού και μια μεταβλητή τύπου Camera και ανανεώνει τις τιμές των διάφορων γνωρισμάτων του χαρακτήρα επί του οποίου καλείτε.

GameFrame

Η κλάση αυτή είναι υπεύθυνη για την δημιουργία του παραθύρου του παιχνιδιού.

- Μέθοδος run()

Με την κλήση αυτής της μεθόδου ξεκινάει το παιχνίδι.

Game

Αυτή είναι η βασική κλάση του παιχνιδιού μέσα στην οποία καλούνται όλες οι μέθοδοι για την φόρτωση και το update των γραφικών του παιχνιδιού. Κληρονομεί την κλάση GameFrame.

- Μέθοδος loadContent()

Η μέθοδος αυτή χρησιμοποιείται για την φόρτωση στην μνήμη όλων των στοιχείων του παιχνιδιού.

- Μέθοδος update()

Η μέθοδος αυτή χρησιμοποιείται για την ανανέωση όλων των στοιχείων του παιχνιδιού (π.χ. τι γίνεται αν υπάρξει κάποια σύγκρουση μεταξύ δύο χαρακτήρων).

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < 1/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013> |

- **Μέθοδος draw(Graphics2D graphics)**

Με την μέθοδο αυτή ζωγραφίζονται όλα τα γραφικά του παιχνιδιού(χαρακτήρες,background,κλπ.).

- **Μέθοδος checkEndOfLevel()**

Στην μέθοδο αυτή ελέγχεται αν έχουμε φτάσει στο τέλος του επιπέδου.Αν ναι φορτώνεται το επόμενο επίπεδο.Αντίστοιχα αν έχουμε φτάσει στο τέλος της τελευταίας πίστας εμφανίζονται τα credits.

- **Μέθοδος hud(Graphics2D graphics)**

Η μέθοδος αυτή τυπώνει στην οθόνη πληροφορίες όπως το score του παίκτη και τις διαθέσιμες σφαίρες και αν χάσει ο παίκτης εμφανίζει την κατάλληλη οθόνη.

- **Μέθοδος loadLevel(String fileName)**

Η μέθοδος αυτή δέχεται ως όρισμα ένα string που περιέχει την διαδρομή του αρχείου στο οποίο περιλαμβάνονται οι πληροφορίες ενός επιπέδου και φορτώνει αυτό το επίπεδο.

Goal

Με αυτή την κλάση μοντελοποιείτε μέσω ενός αντικειμένου το τέλος ενός επιπέδου.

TileType

Η κλάση αυτή περιέχει την πληροφορία αν ένα tile είναι blank ή solid, αν δηλαδή ο ήρωας μπορεί να πατήσει πάνω στο tile ή όχι. Αυτές οι 2 μεταβλητές είναι τύπου enum.

Υπάρχει 1 μέθοδος τύπου static:

- **TileType parse(String str)**

Επιστρέφει τον τύπο του tile ανάλογα με το string.

Tiles

Στην κλάση αυτή φορτώνουμε τα tiles σε arraylist.

Υπάρχουν 4 μέθοδοι τύπου static:

- **void addTile(Tile tile)**

Προσθήκη tile στην arraylist.

- **int getCount()**

Επιστρέφει το μέγεθος της λίστας.

- **void initialize(SpriteLoader spriteLoader)**

Προσθήκη του γραφικού του tile και τον τύπο του tile στην λίστα tiles.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < 1/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013> |

- **Tile getTile(int index)**

Επιστρέφει συγκεκριμένο tile από την λίστα

Tile

Στην κλάση αυτή βρίσκονται τα χαρακτηριστικά ενός tile.

- **Rectangle getBounds(int x,int y)**

Επιστρέφει ένα rectangle για το tile με dimensions 64x64.

- **void draw(Graphics graphics, int x, int y)**

Ζωγραφίζει το tile στις συντεταγμένες.

- **String getImagePath()**

Επιστρέφει το μονοπάτι που είναι αποθηκευμένο το tile.

- **TileType getTileType()**

Επιστρέφει τον τύπο του tile.

- **boolean isSolid()**

Επιστρέφει true ή false ανάλογα με το αν είναι solid ή όχι το tile.

SpriteLoader

Στην κλάση αυτή φορτώνονται τα sprites αφού έχουν προηγουμένως μπει σε ένα Hashtable.

- **Sprite load(String spriteName)**

Αν το hashtable δεν είναι άδειο και εμπεριέχει το όνομα του sprite που θέλουμε φορτώνει την εικόνα. Διαφορετικά πετάει μήνυμα λάθους.

Sprite

Η κλάση αυτή εμπεριέχει τα χαρακτηριστικά ενός sprite.

- **int getWidth()**

Επιστρέφει το πλάτος της εικόνας του sprite

- **int getHeight()**

Επιστρέφει το ύψος της εικόνας του sprite

- **String getImagePath()**

Επιστρέφει το μονοπάτι αποθήκευσης της εικόνας του sprite

- **void draw(Graphics graphics, int x, int y)**

Ζωγραφίζει το sprite στις συντεταγμένες χ,ψ.

| | |
|---|--|
| Συγγραφέας: < <i>Project Genesis</i> > | Κωδικός: < <i>1/Project Genesis /1.0</i> > |
| Έγκριση: < <i>Κουτσουσίμος Βαγγέλης</i> > | Ημερομηνία: < <i>23-05-2013</i> > |

- **void drawFlipped(Graphics graphics, int x, int y)**

Ζωγραφίζει το sprite στις συντεταγμένες x, y αντεστραμμένο.

Map

Η κλάση αυτή χρησιμοποιείται κυρίως για την φόρτωση των χαρτών στο παιχνίδι.

Περιέχει 1 στατική μέθοδο:

- **Map load(SpriteLoader spriteLoader, String fileName)**

Σε αυτή τη μέθοδο διαβάζουμε το txt αρχείο και δημιουργούμε τον χάρτη χρησιμοποιώντας το ύψος και το πλάτος καθώς και ότι στοιχεία υπάρχουν στο αρχείο χωρισμένα με έναν συγκεκριμένο χαρακτήρα. Στο τέλος θέτει σε συγκεκριμένες θέσεις τα tiles που χρειαζόμαστε.

Περιέχει 1 ιδιωτική μέθοδο:

- **initTiles(SpriteLoader spriteLoader)**

Όπου κάνει αρχικοποίηση της λίστας tiles με τα «κενά» tiles.

- **void setTile(int x, int y, Tile tile)**

Θέτει το tile στην λίστα tiles με συγκεκριμένες συντεταγμένες.

- **void draw(Graphics graphics, int offsetX, int offsetY)**

Ζωγραφίζει τα tiles στις συγκεκριμένες συντεταγμένες.

Keys

Στην κλάση αυτή θέτουμε ως στατικές τιμές τα αρχικά κουμπιά του παιχνιδιού.

ListOfKeys

Η κλάση αυτή δημιουργήθηκε για να διαβάζει δεδομένα από ένα αρχείο και να τα εκχωρεί σε 1 static arraylist.

Keyboard

Στην κλάση αυτή περιλαμβάνονται οι μέθοδοι που συσχετίζονται με το πάτημα των πλήκτρων. Επίσης γίνεται η δήλωση μεταβλητών τύπου enum για τις καταστάσεις των κουμπιών (αν είναι πατημένα ή όχι).

- **void poll()**

Διαβάζει σε κάθε frame την κατάσταση των κλειδιών.

- **void keyPressed(KeyEvent e)**

Αν ένα κουμπί είναι πατημένο

- **void keyReleased(KeyEvent e)**

Αν ένα κουμπί αφήνεται

- **boolean isKeyDown (int keyCode)**

Ελέγχει αν ένα κουμπί είναι πατημένο ή πιεσμένο.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

- **boolean isKeyUp(int keyCode)**

Ελέγχει αν δεν είναι πατημένο ένα κουμπί.

Hero

Αυτή είναι η κύρια κλάση του παίχτη.

- **void update(float gameTime, Camera camera, Keyboard input)**

Ελέγχει αν πέθανε ο παίκτης ώστε να ενημερώσει τις ζωές του, αν θα εμφανίσει το game over και την θέση στην οποία θα βγει ξανά.

- **private void keyPressed(Keyboard input)**

Ανάλογα με το κουμπί που πατήθηκε ο παίκτης κάνει την ανάλογη κίνηση.

Server

Η κλάση server είναι η κλάση στην οποία βρίσκονται όλα τα δεδομένα που οι χρήστες θέλουν να ανταλλάξουν μέσω ενός δικτύου στη συγκεκριμένη περίπτωση το score του παιχνιδιού. Η κλάση server περιμένει μία αίτηση απ' την κλάση client με τη μέθοδο accept() ώστε να πραγματοποιηθεί η ανταλλαγή των δεδομένων.

Client

Η κλάση client στέλνει μία αίτηση στην κλάση server, αν η αίτηση αυτή γίνει αποδεκτή απ' την κλάση server, ξεκινά η ανταλλαγή δεδομένων.

4. Περιγραφή Διασυνδέσεων

mainMenu

Στην κλάση αυτή δημιουργείται το βασικό παράθυρο διεπαφής με τον χρήστη και περιέχει πρόσβαση σε όλες τις βασικές λειτουργίες του παιχνιδιού. Αποτελεί την κύρια οθόνη του παιχνιδιού. Περιέχει 5 κουμπιά που δίνουν πρόσβαση σε άλλες οθόνες, καθώς και δυο ετικέτες (JLabel). Μία για τον κύριο τίτλο που εμφανίζεται στο πάνω μέρος της οθόνης και δηλώνει το όνομα του παιχνιδιού και μια ετικέτα που φορτώνει την φωτογραφία του background. Και τα 5 κουμπιά συνδέονται με buttonListeners για να καθορίζουν κάποια λειτουργικότητα. Επίσης υπάρχουν δύο δομές δεδομένων τύπου ArrayList, που αποθηκεύουν δεδομένα String και οι οποίες βοηθάνε για την εκτελεσιμότητα του διαχειριστή αλλαγής πλήκτρων του παιχνιδιού (Key Map).

- **public class GameButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

- **public void actionPerformed (ActionEvent e)**

Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση GameButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

- **private void writeDataToFile(String Path, ArrayList<String> list)**

Η συγκεκριμένη ιδιωτική μέθοδος δέχεται σαν όρισμα μια λίστα αποτελούμενη από String και τα αποθηκεύει σε ένα .txt αρχείο την διεύθυνση του οποίου την βρίσκει από το πρώτο όρισμα που περνιέται.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

- **private void readData(String Path, ArrayList<String> list)**

Η ιδιωτική μέθοδος αυτή, δέχεται σαν όρισμα μια λίστα (συνήθως άδεια στην αρχή του παιχνιδιού) με δεδομένα τύπου String. Στην συνέχεια μέσω της διεύθυνσης που περνιέται σαν πρώτο όρισμα στην μέθοδο ο χρήστης διαβάζει δεδομένα από ένα αρχείο .txt και τα εκχωρεί στην λίστα.

- **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περαστεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

SinglePlayerMenu

Στην συγκεκριμένη κλάση δημιουργείται το παράθυρο διεπαφής για το SinglePlayer κομμάτι του παιχνιδιού, όπου μπορεί ο χρήστης να ρυθμίσει επιλογές που αφορούν το παιχνίδι προτού αυτό ξεκινήσει. Η κλάση περιέχει 2 κουμπία (JButton). Ένα για να ξεκινάει το παιχνίδι και ένα για να επιστρέφει στο κυρίως μενού. Και τα δύο κουμπία συνδέονται με buttonListeners για να αποκρίνονται σε συμβάντα του χρήστη. Επίσης περιέχει 5 ετικέτες (JLabel). Μία για τον κύριο τίτλο του παραθύρου που δηλώνει σε ποία οθόνη βρίσκεται ο χρήστης, μια που φορτώνει το background του παραθύρου, δυο που δηλώνουν που αλλάζει τον χαρακτήρα ο χρήστης και που εισάγει το ψευδώνυμο του και μια που φορτώνει τους χαρακτήρες του παιχνιδιού. Υπάρχει μια λίστα που ο χρήστης μπορεί να διαλέξει τον χαρακτήρα που επιθυμεί να παίξει το παιχνίδι. Κάθε φορά που διαλέγει άλλον χαρακτήρα, αυτόματα αλλάζει και η εικόνα της ετικέτας που φορτώνει τους χαρακτήρες. Τέλος υπάρχει ένα πεδίο εισαγωγής κειμένου (JTextField) που εκεί ο χρήστης πληκτρολογεί ένα ψευδώνυμο.

- **public class MyRunner implements Runnable**

Εσωτερική κλάση που χρησιμοποιεί την διασύνδεση Runnable και απαιτείται για αντικείμενα Thread.

- **public void Run()**

Δημόσια μέθοδος της κλάσης MyRunner. Δημιουργεί ένα αντικείμενο τύπου Game και καλεί την μέθοδο run επί του αντικειμένου αυτού.

- **public class ListMouseListener implements MouseListener**

Εσωτερική κλάση που χρησιμοποιεί την διασύνδεση MouseListener, που είναι για απόκριση συμβάντων που δημιουργεί ο χρήστης μέσω του ποντικιού του. Συνδέεται με την λίστα που υπάρχει στο παράθυρο.

- **public void MouseClicked (MouseEvent e)**

Δημόσια μέθοδος που βρίσκεται στην κλάση ListMouseListener και που σκοπός της είναι να καθορίζει τι θα κάνει το πρόγραμμα σε συμβάντα που προκαλεί ο χρήστης. Στην συγκεκριμένη περίπτωση, να αλλάζει την εικόνα του χαρακτήρα ανάλογα σε ποίο κομμάτι της λίστας πάτησε ο χρήστης με το ποντίκι του.

- **public class SinglePlayerButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

- **public void actionPerformed (ActionEvent e)**

Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση SinglePlayerButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < 1/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

- ο **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περασθεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

- ο **private void writeDataToFile(String nickName, String character)**

Ιδιωτική μέθοδος που δέχεται σαν όρισμα το ψευδώνυμο που έχει γράψει στο JTextField και την διεύθυνση του χαρακτήρα που έχει επιλέξει και τα αποθηκεύει σε ένα αρχείο .txt

MultiPlayerMenu

Στην κλάση αυτή δημιουργούμε ένα παράθυρο με 3 κουμπία. Ένα για να πρόσβαση στην κλάση HostMenu, ένα για πρόσβαση στην κλάση JoinMenu και ένα κουμπί ακύρωσης που επιστρέφει στο κυρίως μενού. Τα κουμπία συνδέονται με buttonListeners για να αποκρίνονται σε ενέργειες του χρήστη. Η κλάση επίσης περιέχει δύο ετικέτες (JLabel). Μια για την φόρτωση της εικόνας του background του παραθύρου, και μια που δηλώνει τον τίτλο του παραθύρου.

- ο **public class MultiPlayerButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

- ο **public void actionPerformed (ActionEvent e)**

Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση MultiPlayerButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

- ο **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περασθεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

HostMenu

Στην συγκεκριμένη κλάση δημιουργείται το παράθυρο διεπαφής για το Host κομμάτι του MultiPlayer του παιχνιδιού, όπου μπορεί ο χρήστης να ρυθμίσει επιλογές που αφορούν το παιχνίδι προτού αυτό ξεκινήσει. Η κλάση περιέχει 2 κουμπία (JButton). Ένα για να ξεκινάει το παιχνίδι και ένα για να επιστρέφει στο κυρίως μενού. Και τα δύο κουμπία συνδέονται με buttonListeners για να αποκρίνονται σε συμβάντα του χρήστη. Επίσης περιέχει 5 ετικέτες (JLabel). Μία για τον κύριο τίτλο του παραθύρου που δηλώνει σε ποία οθόνη βρίσκεται ο χρήστης, μια που φορτώνει το background του παραθύρου, δυο που δηλώνουν που αλλάζει τον χαρακτήρα ο χρήστης και που εισάγει το ψευδώνυμο του και μια που φορτώνει τους χαρακτήρες του παιχνιδιού. Υπάρχει μια λίστα που ο χρήστης μπορεί να διαλέξει τον χαρακτήρα που επιθυμεί να παίξει το παιχνίδι. Κάθε φορά που διαλέγει άλλον χαρακτήρα, αυτόματα αλλάζει και η εικόνα της ετικέτας που φορτώνει τους χαρακτήρες. Τέλος υπάρχει ένα πεδίο εισαγωγής κειμένου (JTextField) που εκεί ο χρήστης πληκτρολογεί ένα ψευδώνυμο.

- ο **public class MyRunner implements Runnable**

Εσωτερική κλάση που χρησιμοποιεί την διασύνδεση Runnable και απαιτείται για αντικείμενα Thread.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

- **public void Run()**

Δημόσια μέθοδος της κλάσης MyRunner. Δημιουργεί ένα αντικείμενο τύπου Server.

- **public class ListMouseListener implements MouseListener**

Εσωτερική κλάση που χρησιμοποιεί την διασύνδεση MouseListener, που είναι για απόκριση συμβάντων που δημιουργεί ο χρήστης μέσω του ποντικιού του. Συνδέεται με την λίστα που υπάρχει στο παράθυρο.

- **public void MouseClicked (MouseEvent e)**

Δημόσια μέθοδος που βρίσκεται στην κλάση ListMouseListener και που σκοπός της είναι να καθορίζει τι θα κάνει το πρόγραμμα σε συμβάντα που προκαλεί ο χρήστης. Στην συγκεκριμένη περίπτωση, να αλλάζει την εικόνα του χαρακτήρα ανάλογα σε ποίο κομμάτι της λίστας πάτησε ο χρήστης με το ποντίκι του.

- **public class HostPlayerButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

- **public void actionPerformed (ActionEvent e)**

Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση HostPlayerButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

- **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περασθεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

- **private void writeDataToFile(String nickName, String character)**

Ιδιωτική μέθοδος που δέχεται σαν όρισμα το ψευδώνυμο που έχει γράψει στο JTextField και την διεύθυνση του χαρακτήρα που έχει επιλέξει και τα αποθηκεύει σε ένα αρχείο .txt.

JoinMenu

Στην συγκεκριμένη κλάση δημιουργείται το παράθυρο διεπαφής για το Join κομμάτι του MultiPlayer του παιχνιδιού, όπου μπορεί ο χρήστης να ρυθμίσει επιλογές που αφορούν το παιχνίδι προτού αυτό ξεκινήσει. Η κλάση περιέχει 2 κουμπιά (JButton). Ένα για να ξεκινάει το παιχνίδι και ένα για να επιστρέφει στο κυρίως μενού. Και τα δύο κουμπιά συνδέονται με buttonListeners για να αποκρίνονται σε συμβάντα του χρήστη. Επίσης περιέχει 6 ετικέτες (JLabel). Μία για τον κύριο τίτλο του παραθύρου που δηλώνει σε ποία οθόνη βρίσκεται ο χρήστης, μια που φορτώνει το background του παραθύρου, δυο που δηλώνουν που αλλάζει τον χαρακτήρα ο χρήστης και που εισάγει το ψευδώνυμο του, μια που φορτώνει τους χαρακτήρες του παιχνιδιού και μια που δηλώνει που εισάγει την ip address για να είναι επιτυχής η σύνδεση στον server. Υπάρχει μια λίστα που ο χρήστης μπορεί να διαλέξει τον χαρακτήρα που επιθυμεί να παίξει το παιχνίδι. Κάθε φορά που διαλέγει άλλον χαρακτήρα, αυτόματα αλλάζει και η εικόνα της ετικέτας που φορτώνει τους χαρακτήρες. Τέλος υπάρχει δυο πεδία εισαγωγής κειμένου (JTextField) που εκεί ο χρήστης πληκτρολογεί ένα ψευδώνυμο και την ip address.

- **public class MyRunner implements Runnable**

Εσωτερική κλάση που χρησιμοποιεί την διασύνδεση Runnable και απαιτείται για αντικείμενα Thread.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < 1/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

- **public void Run()**

Δημόσια μέθοδος της κλάσης MyRunner. Δημιουργεί ένα αντικείμενο τύπου Client.

- **public class ListMouseListener implements MouseListener**

Εσωτερική κλάση που χρησιμοποιεί την διασύνδεση MouseListener, που είναι για απόκριση συμβάντων που δημιουργεί ο χρήστης μέσω του ποντικιού του. Συνδέεται με την λίστα που υπάρχει στο παράθυρο.

- **public void MouseClicked (MouseEvent e)**

Δημόσια μέθοδος που βρίσκεται στην κλάση ListMouseListener και που σκοπός της είναι να καθορίζει τι θα κάνει το πρόγραμμα σε συμβάντα που προκαλεί ο χρήστης. Στην συγκεκριμένη περίπτωση, να αλλάζει την εικόνα του χαρακτήρα ανάλογα σε ποίο κομμάτι της λίστας πάτησε ο χρήστης με το ποντίκι του.

- **public class JoinPlayerButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

- **public void actionPerformed (ActionEvent e)**

Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση JoinPlayerButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

- **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περαστεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

- **private void writeDataToFile(String nickName, String character)**

Ιδιωτική μέθοδος που δέχεται σαν όρισμα το ψευδώνυμο που έχει γράψει στο JTextField, την διεύθυνση του χαρακτήρα που έχει επιλέξει και τα αποθηκεύει σε ένα αρχείο .txt.

CreditsMenu

Στην συγκεκριμένη κλάση δημιουργείται ένα παράθυρο όπου ο χρήστης μπορεί να διαβάσει τα credits του παιχνιδιού. Το παράθυρο αποτελείται από ένα κουμπί ακύρωσης (JButton) όπου γυρίζει στην οθόνη του κυρίως μενού, από 3 ετικέτες (JLabel) και από μια περιοχή κειμένου (JTextArea). Η μια ετικέτα είναι για να φορτώνει το background του παραθύρου, μια για να εμφανίζει τον τίτλο του παραθύρου και μια άλλη για να εμφανίζει το όνομα του παιχνιδιού. Στην περιοχή κειμένου εμφανίζονται σε μορφή κειμένου τα credits. Το κουμπί συνδέεται με buttonListener για να αποκρίνονται σε συμβάντα του χρήστη

- **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περαστεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

- **public class CreditsButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

- ο **public void actionPerformed (ActionEvent e)**

Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση CreditsButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

ExitFrame

Στην κλάση αυτή δημιουργείται ένα παράθυρο που ζητά επιβιβέωση η όχι για το αν ο χρήστης επιθυμεί να αποχωρήσει οριστικά από το παιχνίδι. Υπάρχουν δυο κουμπία (JButton) για την καταφατική η αρνητική απάντηση που θα δώσει ο χρήστης. Και τα δύο κουμπία συνδέονται με buttonListeners για να αποκρίνονται σε συμβάντα του χρήστη. Υπάρχουν επίσης δυο ετικέτες (JLabel). Μια για να φορτώνει την εικόνα του background στο παράθυρο και μια για να φαίνεται η ερώτηση που κάνει στον χρήστη.

- ο **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περασθεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

- ο **public class ExitButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

- ο **public void actionPerformed (ActionEvent e)**

Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση ExitButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

OptionsMenu

Στην κλάση αυτή δημιουργείται το παράθυρο διεπαφής όπου ο χρήστης μπορεί να κάνει αλλαγές και να ρυθμίσει λειτουργίες του παιχνιδιού όπως ο ήχος και τα πλήκτρα χειρισμού του χαρακτήρα. Πιο συγκεκριμένα, υπάρχει δυνατότητα σίγασης η όχι των ηχητικών εφέ, καθώς και της μουσικής του παιχνιδιού. Επίσης αν θέλει ο χρήστης μπορεί να αλλάξει τα κουμπιά που θα χειρίζεται τον χαρακτήρα στο παιχνίδι και να βάλει όποια ο ίδιος επιθυμεί.

Η κλάση αποτελείται από 3 κουμπία (JButton). Το ένα είναι κουμπί ακύρωσης και γυρνάει στην οθόνη του κυρίως μενού. Ένα κουμπί είναι για να κατοχυρωθούν οι αλλαγές που έγιναν και να σωθούν, και ένα άλλο είναι για να ανοίξει μια οθόνη πληκτρολογίου και να μπορεί ο χρήστης να διαλέξει το πλήκτρο που θέλει. Και τα 3 κουμπία συνδέονται με buttonListeners για να ενεργούν ανάλογα με τα συμβάντα που προκαλεί ο χρήστης. Υπάρχουν 4 check boxes (JCheckBox), 2 για τα ηχητικά εφέ, και 2 για την μουσική. Τα check boxes συνδέονται με 2 groups (JButtonGroup) για συγχρονισμό των επιλογών. Επίσης υπάρχουν 6 ετικέτες (JLabel) για να δηλώνουν τις διάφορες λειτουργίες που έχει την δυνατότητα να κάνει ο χρήστης καθώς και για να φορτώνουν το background του παραθύρου. Υπάρχει μία λίστα όπου περιέχει τις κινήσεις του χαρακτήρα στο παιχνίδι. Ακόμα ένα πεδίο κειμένου (JTextField) βρίσκεται στο παράθυρο για να εμφανίζει σε μορφή μικρού κειμένου το κουμπί που είναι προεπιλεγμένο για μια συγκεκριμένη κίνηση (την κίνηση στην οποία δείχνει εκείνη την στιγμή η λίστα). Τέλος δυο δομές δεδομένων τύπου ArrayList που περιέχουν δεδομένα String για την λειτουργία του KeyMap.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

- **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περασθεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

- **public class OptionButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

- **public void actionPerformed (ActionEvent e)**

Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση OptionButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

- **public class ListMouseListener implements MouseListener**

Εσωτερική κλάση που χρησιμοποιεί την διασύνδεση MouseListener, που είναι για απόκριση συμβάντων που δημιουργεί ο χρήστης μέσω του ποντικιού του. Συνδέεται με την λίστα που υπάρχει στο παράθυρο.

- **public voidMouseClicked (MouseEvent e)**

Δημόσια μέθοδος που βρίσκεται στην κλάση ListMouseListener και που σκοπός της είναι να καθορίζει τι θα κάνει το πρόγραμμα σε συμβάντα που προκαλεί ο χρήστης. Στην συγκεκριμένη περίπτωση, να αλλάζει το κείμενο στο πεδίο κειμένου που υπάρχει στο παράθυρο ανάλογα με την κίνηση που έχει επιλέξει ο χρήστης με το ποντίκι του στην λίστα.

- **private void writeDataToFile(String Path, ArrayList<String> list)**

Η συγκεκριμένη ιδιωτική μέθοδος δέχεται σαν όρισμα μια λίστα αποτελούμενη από String και τα αποθηκεύει σε ένα .txt αρχείο την διεύθυνση του οποίου την βρίσκει από το πρώτο όρισμα που περνιέται.

- **private void readData(String Path, ArrayList<String> list)**

Η ιδιωτική μέθοδος αυτή, δέχεται σαν όρισμα μια λίστα (συνήθως άδεια στην αρχή του παιχνιδιού) με δεδομένα τύπου String. Στην συνέχεια μέσω της διεύθυνσης που περνιέται σαν πρώτο όρισμα στην μέθοδο ο χρήστης διαβάζει δεδομένα από ένα αρχείο .txt και τα εκχωρεί στην λίστα.

KeyBoardMenu

Η συγκεκριμένη κλάση αποτελεί ένα παράθυρο που έχει την μορφή πληκτρολογίου και σκοπός της είναι να προσφέρει στον χρήστη μια άμεση και λειτουργική εικόνα του KeyMap. Το παράθυρο περιέχει πλήθος κουμπιών στην διάταξη που υπάρχουν στα συνηθισμένα πληκτρολόγια του εμπορίου και ο χρήστης πατώντας ένα κουμπί με το ποντίκι του, αποθηκεύει το κουμπί αυτό σαν πλήκτρο χειρισμού.

Συγκεκριμένα η κλάση αποτελείται από 41 κουμπία (JButton) όπου το κάθε κουμπί αντιστοιχεί στο αντίστοιχο ενός κανονικού πληκτρολογίου. Όλα τα κουμπία συνδέονται με buttonListeners για να αποκρίνονται κατάλληλα στα συμβάντα που προκαλεί ο χρήστης με το ποντίκι του. Μια ετικέτα (JLabel) για να φορτώνει το background του παραθύρου. Επίσης περιέχει μια μεταβλητή Boolean που έχει τον ρόλο σημαίας-φρουρού για τον έλεγχο επιλογής πλήκτρου που χρησιμοποιείται ήδη. Τέλος μια λίστα τύπου ArrayList που περιέχει String για την αποθήκευση αριθμών που αντιστοιχούν στα κουμπία πληκτρολογίου της Java.

| | |
|---|--|
| Συγγραφέας: < <i>Project Genesis</i> > | Κωδικός: < <i>1/Project Genesis /1.0</i> > |
| Έγκριση: < <i>Κουτσουσίμος Βαγγέλης</i> > | Ημερομηνία: < <i>23-05-2013</i> > |

○ **private void writeDataToFile(ArrayList<String> list)**

Η συγκεκριμένη ιδιωτική μέθοδος δέχεται σαν όρισμα μια λίστα αποτελούμενη από String και τα αποθηκεύει σε ένα .txt αρχείο διεύθυνση που είναι γνωστή και συγκεκριμένη μέσα στην μέθοδο.

○ **private void centerFrame (JFrame frame)**

Η μέθοδος αυτή μέσω υπολογισμών που πραγματοποιεί τοποθετεί το frame που έχει περαστεί σαν όρισμα στο κέντρο της οθόνης του υπολογιστή.

○ **public class KeyBoardButtonListener implements ActionListener**

Εσωτερική δημόσια κλάση που χρησιμοποιεί την διασύνδεση ActionListener για απόκριση συμβάντων σε ενέργειες του χρήστη.

○ **public void actionPerformed (ActionEvent e)**

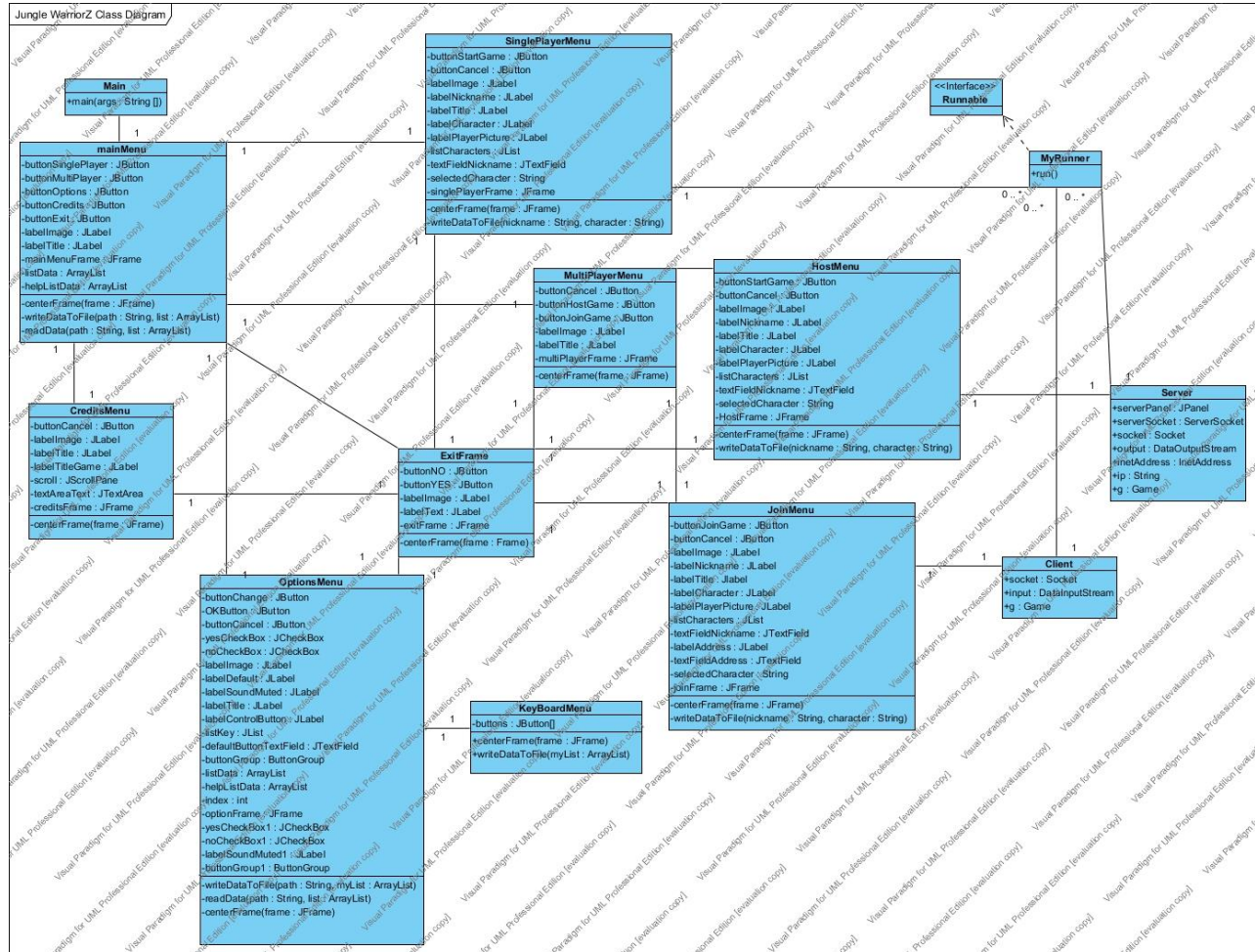
Δημόσια μέθοδος που βρίσκεται μέσα στην κλάση KeyBoardButtonListener και λειτουργία της είναι να καθορίζει τι θα συμβεί ανάλογα με το κουμπί που πατάει ο χρήστης.

| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /I.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013> |

5. Παραρτήματα

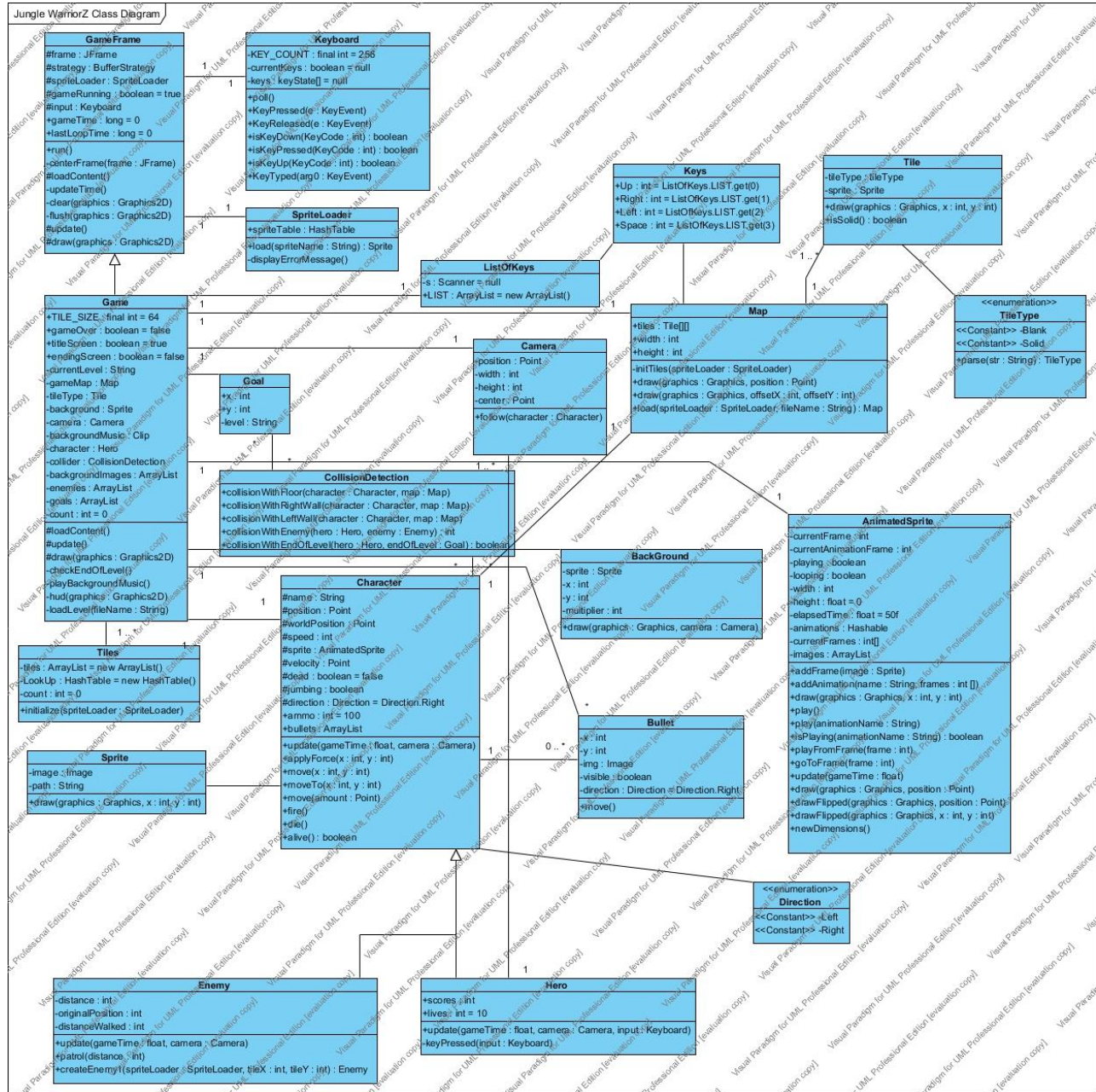
5.1 Διαγράμματα Κλάσεων

- Διάγραμμα κλάσεων για το πακέτο Menu του κώδικα.



| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < 1/Project Genesis /1.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

➤ Διάγραμμα κλάσεων για το πακέτο Game του κώδικα.



| | |
|------------------------------------|-------------------------------------|
| Συγγραφέας: < Project Genesis > | Κωδικός: < I/Project Genesis /I.0 > |
| Έγκριση: < Κουτσουσίμος Βαγγέλης > | Ημερομηνία: < 23-05-2013 > |

5.2 Πίνακας Ιχνηλάτισης

Ο πίνακας ιχνηλάτισης του παιχνιδιού με τις λειτουργικές απαιτήσεις και τις κύριες κλάσεις όπως είχαν καταγραφεί στο ΕΠΑΛ.

| Λειτουργικές απαιτήσεις | Έναξή SinglePlayer | Έναξή Host Game | Έναξή Join Game | Επιλογή Options | Επιλογή Credits | Επιλογή Exit |
|---|--------------------|-----------------|-----------------|-----------------|-----------------|--------------|
| Εισαγωγή nickname | x | x | x | | | |
| Επιλογή χαρακτήρα | x | x | x | | | |
| Ακύρωση παραθύρου | x | x | x | | | |
| Έναρξη παιχνιδιού | x | x | x | | | |
| Μήνυμα σφάλματος για nickname | x | x | x | | | |
| Μήνυμα σφάλματος για ip address | | | x | | | |
| Μήνυμα σφάλματος για μη σύνδεση του ιντερνετ | | | | | | |
| Μήνυμα σφάλματος για χρησιμοποιημένη ip address | | | x | | | |
| Εισαγωγή ip address | | | x | | | |
| Ίδιο nickname από Server και Client | | | x | | | |
| Επιλογή σίγασης ήχου | | | | x | | |
| Διαχείριση πλήκτρων κουμπιών | | | | x | | |
| Μήνυμα σφάλματος σε πλήκτρο που χρησιμοποιείται ήδη | | | | x | | |
| Ανάγνωση Credits | | | | | x | |
| Έξοδος από το παιχνίδι | | | | | | x |

5.3 Διαγράμματα Αναφορών και Οθονών

Δεν υπάρχει διάγραμμα Αναφορών και Οθονών.

5.4 Διαγράμματα Οντοτήτων – Συσχετίσεων

Δεν υπάρχει διάγραμμα Οντοτήτων – Συσχετίσεων.