Τ Ε Χ Ν Ο Λ Ο Γ Ι Κ Ο Ε Κ Π Α Ι Δ Ε Υ Τ Ι Κ Ο Ι Δ Ρ Υ Μ Α Σ Ε Ρ Ρ Ω Ν
Σ Χ Ο Λ Η Τ Ε Χ Ν Ο Λ Ο Γ Ι Κ Ω Ν Ε Φ Α Ρ Μ Ο Γ Ω Ν
Τ Μ Η Μ Α Π Λ Η Ρ Ο Φ Ο Ρ Ι Κ Η Σ & Ε Π Ι Κ Ο Ι Ν Ω Ν Ι Ω Ν

Ανάπτυξη MultiPlayer mode για τον εξομοιωτή PROMASI

Πτυχιακή εργασία του

Θεοδωρίδη Αλέξανδρου (1419)

Επιβλέπων: Δρ. Νικόλαος Πεταλίδης, Επιστημονικός Συνεργάτης

ΣΕΡΡΕΣ, ΜΑΙΟΣ 2011

Ευχαριστήρια

Θέλω να πω ένα μεγάλο ευχαριστώ σε όλους τους φίλους και συγγενείς που στάθηκαν δίπλα μου όλον αυτόν τον καιρό και βοήθησαν σημαντικά στην ανάπτυξη του έργου.

Ευχαριστώ τον κύριο Νίκο Πεταλίδη ο οποίος είναι υπεύθυνος του έργου και παρακολουθεί συνεχώς την εξέλιξη του βοηθώντας σημαντικά την ποιοτική ανάπτυξη του. Τον συνάδελφο Γρηγόρη Γρηγοριάδη ο οποίος ξεκίνησε το έργο και βοήθησε στην εξέλιξη του. Τον φοιτητή του τμήματος πληροφορικής και επικοινωνιών Αντώνη Χρονάκη με τον οποίων συνεργαζόμαστε συνεχώς. Επίσης θέλω να ευχαριστήσω τον φοιτητή του τμήματος πληροφορικής και επικοινωνιών Αναστάσιο Παπάζογλου για την ανάπτυξη της ιστοσελίδας του PROMASI http://www.promasi.gr.

Περίληψη

Η πτυχιακή αυτή ασχολείται με την εξέλιξη της πρώτης έκδοσης του εξομοιωτή PROMASI. Ο PROMASI είναι ένας PROject MAnagement Simulator (εξομοιωτής διαχείρισης έργων) ανοικτού λογισμικού που αναπτύσσεται με σκοπό κάποια στιγμή να χρησιμοποιηθεί για τη διδασκαλία της διαχείρισης έργων λογισμικού. Η παρούσα πτυχιακή επικεντρώνεται στην βελτίωση της πρώτης έκδοσης με την προσθήκη της δυνατότητας χρήσης του εξομοιωτή ταυτόχρονα από πολλούς χρήστες με σκοπό να υπάρξει ρεαλισμός και ανταγωνισμός μεταξύ τους όταν αυτοί θα συναγωνίζονται για να παραδώσουν το “καλύτερο δυνατό” έργο. Η δυνατότητα αυτή του εξομοιωτή ονομάστηκε MultiPlayer mode. Η πτυχιακή παρουσιάζει τις σχεδιαστικές αποφάσεις που πάρθηκαν προκειμένου αυτό να καταστεί δυνατό καθώς και τις νέες λειτουργίες του εξομοιωτή.

Πίνακας Περιεχομένων

1 Εισαγωγή....6
1.1 Δομή της εργασίας....8
2 PROMASI....10
2.1 System dynamics....10
2.2 Λειτουργίες της πρώτης έκδοσης....14
2.3 Άλλες λειτουργίες....18
2.4 Μειονεκτήματα της πρώτης έκδοσης....19
2.5 Παρόμοια Projects....22
3 PROMASI 2....23
3.1 Role-playing games (RPGs)....23
3.2 MultiPlayer....23
3.3 Νέες λειτουργίες....25
3.4 Άλλες λειτουργίες....25
3.5 Περιπτώσεις χρήσης....26
3.6 Βελτιώσεις....30
3.6.1 Αρχιτεκτονική του PROMASI....30
3.6.2 Γραφικό περιβάλλον....30
3.6.3 Αγορά εργασίας....31
3.6.4 Μενού επιλογής παιχνιδιού....31
3.6.5 Διαχείριση προσωπικού....31
3.6.6 Τοπολογία του PROMASI....32
3.7 Πρωτόκολλο επικοινωνίας....32
3.8 Εναλλακτικές λύσεις στην επικοινωνία....33
4 Σχεδίαση....34
4.1 Η σχεδίαση της πρώτης έκδοσης....34
4.1.1 Gui Package....35
4.1.2 Shell Package....35
4.1.3 PROMASI Core Package....35
4.1.4 Model Package....35
4.2 Αλλαγή αρχιτεκτονικής....36
4.3 Βιβλιοθήκες....37

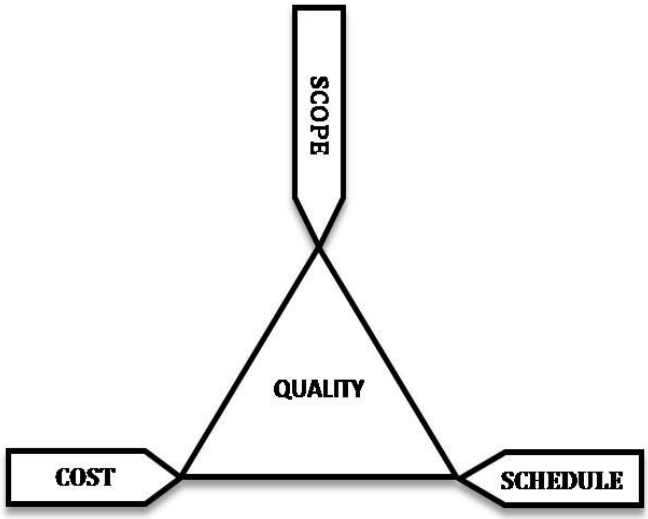
4.3.1 Σχέσεις ανάμεσα στις βιβλιοθήκες της εφαρμογής Server....38
4.3.3 Σχέσεις ανάμεσα στις βιβλιοθήκες της εφαρμογής Client....39
4.3.4 Utilities....40
4.3.5 SdSystem....41
4.3.6 Game....43
4.3.7 Network....44
4.3.8 Protocol....45
4.3.9 Επικοινωνία....51
4.3.10 Καταστάσεις του Client....53
4.3.11 Καταστάσεις του Server....58
4.3.12 Πλεονεκτήματα της νέας αρχιτεκτονικής....63
4.3.13 Σύζευξη....63
4.3.14 Πολλές υλοποιήσεις ταυτόχρονα – plugins....63
5 Τελική εφαρμογή....64
5.1 Διαδικασία εισαγωγής στο παιχνίδι....64
5.3 Αγορά εργασίας....68
5.4 Προγραμματισμός του έργου....69
6 Συμπεράσματα....71
6.1 Χρονοδιάγραμμα του έργου....71
6.2 Τελικοί στόχοι....71
6.3 Τα λάθη....72
7 Παράρτημα....73
7.1 Εργαλεία ανάπτυξης....73
7.1.1 Eclipse....73
7.1.2 Windows Builder....74
7.1.3 Visual Paradigm....75
7.1.4 Βιβλιοθήκη SWT....75
7.1.5 Source Control....76
8 Βιβλιογραφία....78

1 Εισαγωγή

Με τον όρο *διαχείριση έργου* εννοούμε ένα σύνολο από μεθόδους και τεχνικές που χρησιμοποιούνται με σκοπό την επιτυχή ολοκλήρωση ενός έργου, τη μείωση του κόστους παραγωγής και των κινδύνων που ενδέχεται να εμφανιστούν κατά την διάρκεια εκτέλεσής του (Lock, 2007).

Ο διαχειριστής ενός έργου (Project Manager) είναι υπεύθυνος για το έργο, είναι αυτός που θα χωρίσει το συνολικό έργο σε επιμέρους μικρές εργασίες και θα αναθέσει την καθεμία από αυτές στους υπαλλήλους της επιχείρησης. Ο βασικός σκοπός του κάθε Project Manager είναι να πετύχει το καλύτερο δυνατό αποτέλεσμα μέσα σε λιγότερο δυνατό χρόνο με το μικρότερο δυνατό κόστος χωρίς να εκτεθεί σε κινδύνους. Το παρακάτω σχήμα περιγράφει την ποιότητα του έργου σε σχέση με τους στόχους, προϋπολογισμό και τον χρόνο.

Εικόνα 1: Οι βασικές παράμετροι ενός έργου



Σε ιδανική περίπτωση η ποιότητα του έργου παραμένει ίδια σε σχέση με τις απαιτήσεις των πελατών δηλαδή μπορεί να θεωρηθεί σταθερή. Αν οι ποιότητα του έργου είναι σταθερή τότε οι μόνες μεταβλητές που μπορούν να μεταβάλλονται από σχήμα είναι το:

- *Εύρος* που είναι και ο κύριος στόχος του έργου και περιγράφεται από τις απαιτήσεις του πελάτη.

- *Κόστος* που είναι ο προϋπολογισμός του έργου.
- *Χρονοδιάγραμμα* που αφορά τα χρονικά περιθώρια μέσα στα οποία πρέπει να ολοκληρωθεί το έργο.

Με αυτόν τον τρόπο αν για παράδειγμα μειώσουμε τον χρόνο για την υλοποίηση του έργου χωρίς να αυξήσουμε τις δαπάνες για την πρόσληψη νέου προσωπικού, μπορεί το τελικό αποτέλεσμα να μην ικανοποιεί όλες τις απαιτήσεις του πελάτη. Αντιθέτως μπορούμε να αυξήσουμε τον χρόνο αυξάνοντας και τα έξοδα για την υλοποίηση του έργου ταυτόχρονα καλύπτοντας και τις περισσότερες απαιτήσεις.

Η διαχείριση ενός έργου δεν είναι εύκολη υπόθεση. Δε σημαίνει για παράδειγμα ότι απλά δαπανώντας περισσότερα χρήματα και προσλαμβάνοντας περισσότερο προσωπικό το χρονοδιάγραμμα ενός έργου θα μπορέσει να συρρικνωθεί. Ένας λόγος για να συμβεί κάτι τέτοιο είναι ότι το έργο μπορεί να περιέχει κάποια κομμάτια που απαιτούν περιορισμένο αριθμό εργαζομένων ή εργαζομένους με εξειδικευμένες γνώσεις που ο αριθμός τους είναι περιορισμένος, και το συγκεκριμένο κομμάτι του έργου δεν μπορεί να υλοποιηθεί παράλληλα με τα υπόλοιπα κομμάτια. Τέτοιου είδους περιπτώσεις ονομάζονται κρίσιμα μονοπάτια και παίζουν καθοριστικό ρόλο για τον χρόνο υλοποίησης του έργου.

Για παράδειγμα ας φανταστούμε ότι έχουμε να υλοποιήσουμε ένα έργο λογισμικού στο οποίο μπορούμε να απασχολήσουμε ταυτόχρονα έως και 3 υπαλλήλους ενώ υπάρχουν και κάποια κρίσιμα μονοπάτια στα οποία μπορεί να δουλέψει μόνο ένα άτομο. Για να ολοκληρωθεί με επιτυχία το έργο πρέπει να γίνει σωστός προγραμματισμός με τον οποίο θα αναθέσει το σωστό κομμάτι του συνολικού έργου στο σωστό υπάλληλο. Ο Project Manager πρέπει να λάβει υπόψη του την εμπειρία που έχει ο κάθε υπάλληλος ώστε να του αναθέσει το σωστό κομμάτι του έργου. Επίσης πρέπει να υπολογίσει τη διάρκεια της κάθε εργασίας ή πολλές φορές και με ποια σειρά θα εκτελεστούν οι αυτές οι εργασίες. Επιπλέον πρέπει να επαναλαμβάνει τον προγραμματισμό του έργου πολλές. Ο Project Manager αναγκάζεται να αλλάζει τον προγραμματισμό του κατά την διάρκεια του έργου γιατί μπορεί να προκύψουν πάρα πολλές απρόβλεπτες καταστάσεις μέσα στο χρονικό διάστημα που διαρκεί το έργο.

Είναι φανερό επομένως ότι η διαχείριση ενός έργου μπορεί να επηρεαστεί από πολλούς παράγοντες και προφανώς όσο πιο μεγάλο είναι το έργο και όσο πιο πολλοί μετέχουν σε αυτό τόσο πιο δύσκολη γίνεται η διαχείριση του.

Σε πολύ μικρά έργα όμως η διαχείριση είναι πιο εύκολη ειδικά αν για παράδειγμα παράγοντες όπως το κόστος ή το χρονοδιάγραμμα δεν επηρεάζουν το έργο όπως γίνεται συχνά σε φοιτητικές εργασίες. Κάτι τέτοιο όμως δημιουργεί διάφορα προβλήματα στη διδασκαλία θεμάτων που αφορούν τη διαχείριση έργου γιατί ακριβώς είναι δύσκολο να δημιουργηθούν οι συνθήκες ώστε οι σπουδαστές να μπορούν να βιώσουν τις δυσκολίες που εμφανίζονται σε ένα ρεαλιστικό έργο. Ένας τρόπος για να ξεπεραστεί το πιο πάνω πρόβλημα είναι η λύση της προσομοίωσης όπου οι σπουδαστές παρακολουθούν την εξέλιξη ενός “εικονικού” έργου στο οποίο καλούνται να λάβουν αποφάσεις που αφορούν στην υλοποίηση του.

Η πτυχιακή αυτή παρουσιάζει μια τέτοια προσέγγιση, επεκτείνοντας τον προσομοιωτή PROMASI ο οποίος πρωτοπαρουσιάστηκε στην πτυχιακή εργασία (Γρηγοριάδης, 2009). Σκοπός της πτυχιακής αυτής πιο συγκεκριμένα είναι η επέκταση του περιβάλλοντος του PROMASI ώστε αυτό να γίνει πιο ρεαλιστικό δίνοντας τη δυνατότητα σε πολλούς σπουδαστές ταυτόχρονα να δοκιμάζουν τις δυνατότητές διαχείρισης ενός έργου σε συνθήκες που προσεγγίζουν αυτές της λειτουργίας μιας πραγματικής επιχείρησης.

1.1 Δομή της εργασίας

Η εργασία αναφέρεται στην εφαρμογή προσομοίωσης PROMASI. Περιγράφει την δομή της εφαρμογής και τις βασικότερες λειτουργίες της και τις αρχές προσομοίωσης στις οποίες στηρίζεται. Γίνεται επίσης μια κριτική για τα κυριότερα προβλήματα που αντιμετωπίζει.

Το δεύτερο κεφάλαιο αναφέρεται στην δεύτερη έκδοση της εφαρμογής όπως διαμορφώνεται σε αυτήν την εργασία. Περιγράφει τις επιπρόσθετες λειτουργίες πολλών χρηστών (multi-player mode), τις περιπτώσεις χρήσης της εφαρμογής καθώς και την τοπολογία και την λειτουργία της.

Η συνέχεια είναι αφιερωμένη στην σχεδίαση και την υλοποίηση της εφαρμογής.

Περιγράφει τις μεθόδους και τεχνικές που χρησιμοποιήθηκαν για την υλοποίηση της. Επίσης σε αυτό το κεφάλαιο περιγράφεται το πρωτόκολλο επικοινωνίας που υλοποιήθηκε για την υποστήριξη της λειτουργίας πολλών χρηστών.

Το επόμενο κεφάλαιο μιλάει για τις λειτουργίες τις δεύτερης έκδοσης του PROMASI έτσι όπως πρέπει να την δει ο τελικός χρήστης και περιέχει τις βασικές οδηγίες χρήσης της εφαρμογής.

Η εργασία ολοκληρώνεται με κάποιες βασικές παρατηρήσεις και συμπεράσματα από το συγκεκριμένο έργο. Τέλος το παράρτημα αναφέρεται στα εργαλεία και βιβλιοθήκες που χρησιμοποιήθηκαν για την υλοποίηση του PROMASI.

2 PROMASI

Το PROMASI είναι ένας προσομοιωτής που έχει ως στόχο την παροχή μιας πλατφόρμας εκπαίδευσης των μελλοντικών Project Managers μέσα στα πλαίσια του μαθήματος Τεχνολογία Λογισμικού 2 της σχολής πληροφορικής και επικοινωνιών του ΤΕΙ Σερρών. Ο εκπαιδευτικός στόχος της εφαρμογής είναι να μάθουν οι σπουδαστές, την διαχείριση ενός έργου λογισμικού μέσα στις συνθήκες που είναι κοντά στις συνθήκες της πραγματικής αγοράς. Ο σπουδαστής σ' αυτήν την περίπτωση εκτελεί τα καθήκοντα ενός project manager, διαχειρίζεται δηλαδή μια ομάδα προγραμματιστών που έχουν στόχο την υλοποίηση ενός έργου λογισμικού. Το έργο εξελίσσεται σε βάθος χρόνου και ο στόχος του project manager είναι να προγραμματίσει την εξέλιξη αυτή με τέτοιο τρόπο ώστε να αποφευχθεί ο κίνδυνος αποτυχίας και να μειώσει το κόστος της υλοποίησης του έργου.

Το PROMASI παρέχει όλα τα βασικά εργαλεία που χρειάζεται ένας project manager. Για παράδειγμα ένα από αυτά τα εργαλεία είναι ο Gantt diagram sheduler. Με αυτό το εργαλείο ο σπουδαστής μπορεί να προγραμματίσει ένα έργο λογισμικού κατασκευάζοντας ένα διάγραμμα Gantt. Τα διαγράμματα Gantt παρέχουν όλες τις πληροφορίες για την εξέλιξη του έργου σε βάθος χρόνου. Σε ιδανική περίπτωση ένα διάγραμμα που κατασκευάστηκε στην αρχή του έργου δεν αλλάζει ποτέ, και αυτό θα σήμαινε ότι έγινε ο σωστός προγραμματισμός του έργου. Στην πραγματικότητα όμως συμβαίνει το αντίθετο. Υπάρχουν πάρα πολλές λεπτομέρειες που μπορούν να επηρεάσουν ένα έργο λογισμικού. Για αυτό είναι αδύνατον να γίνει ο σωστός προγραμματισμός του έργου εξαρχής. Η δουλειά του σπουδαστή επομένως στον προσομοιωτή θα είναι να προσαρμόζει συνεχώς τα διαγράμματα Gantt, δηλαδή τον προγραμματισμό όταν υπάρχει κάποια απόκλιση από τον αρχικό σχεδιασμό.

2.1 System dynamics

Το PROMASI είναι ουσιαστικά ένας προσομοιωτής. Προσομοίωση είναι η αναπαράσταση της συμπεριφοράς κάποιου συστήματος του πραγματικού κόσμου με την βοήθεια ενός μοντέλου που το περιγράφει. Υπάρχουν διαφόρων ειδών μοντέλα προσομοίωσης όπως για παράδειγμα τα στατικά μοντέλα προσομοίωσης όπου ο χρόνος δεν επηρεάζει τα

αποτελέσματα της προσομοίωσης, δυναμικά μοντέλα όπου ο χρόνος επηρεάζει άμεσα τα αποτελέσματα, αιτιοκρατικά μοντέλα τα αποτελέσματα των οποίων είναι προβλέψιμα πριν την εκτέλεση τους λόγο απουσίας τυχαίων τμημάτων, στοχαστικά μοντέλα στα οποία υπάρχει τουλάχιστον ένα τμήμα με τυχαία είσοδο κ.α. (Madachy, 2008)

Η επιστήμη των υπολογιστών σήμερα είναι σε θέση να προσφέρει αλγορίθμους και τεχνικές για την κατασκευή και χρήση ενός μοντέλου προσομοίωσης. Το βασικό πλεονέκτημα της προσομοίωσης είναι το χαμηλός κόστος και το μηδενικό ρίσκο με αξιόπιστα αποτελέσματα κατά την εκτέλεση των πειραμάτων η δοκιμών. Για παράδειγμα η αυτοκινητοβιομηχανία χρησιμοποιεί τα μοντέλα προσομοίωσης φτιαγμένα με την βοήθεια του υπολογιστή για την εξέταση της αεροδυναμικής η της ασφάλειας του νέου μοντέλου αυτοκινήτου. Η μετεωρολογική υπηρεσία χρησιμοποιεί τα μοντέλα προσομοίωσης για την πρόγνωση του καιρού. Εμείς μπορούμε να χρησιμοποιήσουμε την προσομοίωση στην εκπαίδευση των μελλοντικών project managers κατασκευάζοντας ένα μοντέλο που να προσομοιώνει την λειτουργία μίας επιχείρησης παραγωγής λογισμικού.

Η προσομοίωση στον εξομοιωτή PROMASI στηρίζεται σε μοντέλα που είναι γραμμένα σε System Dynamics (Sterman, 2000) με κάποιες αλλαγές οι οποίες είναι απαραίτητες για να μπορέσει το μοντέλο να γίνει διαδραστικό.

Η γλώσσα των System dynamics περιλαμβάνει κάποια βασικά αντικείμενα από τα οποία και κατασκευάζονται τα μοντέλα προσομοίωσης. Τα βασικά αντικείμενα της γλώσσας αυτής είναι τα εξής.

- *Flow* περιγράφει ρυθμό με τον οποίο μεταφέρεται η πληροφορία από ένα αντικείμενο σε ένα άλλο.
- *Stock* περιγράφει τα βασικά μεγέθη του μοντέλου προσομοίωσης.
- *Variable* περιγράφει μια σταθερή ποσότητα μέσα στο μοντέλο προσομοίωσης.

Για παράδειγμα αν θέλουμε να προσομοιώσουμε ένα σύστημα με αλεπούδες και λαγούς και έχουμε την πληροφορία ότι οι αλεπούδες καταναλώνουν τους λαγούς με έναν ρυθμό X και οι λαγοί πολλαπλασιάζονται με έναν άλλον ρυθμό Y τότε οι ποσότητες αυτές μπορούν να περιγραφούν με δυο flows “fox consumption of rabbits” και “rabbit births” που δηλώνουν

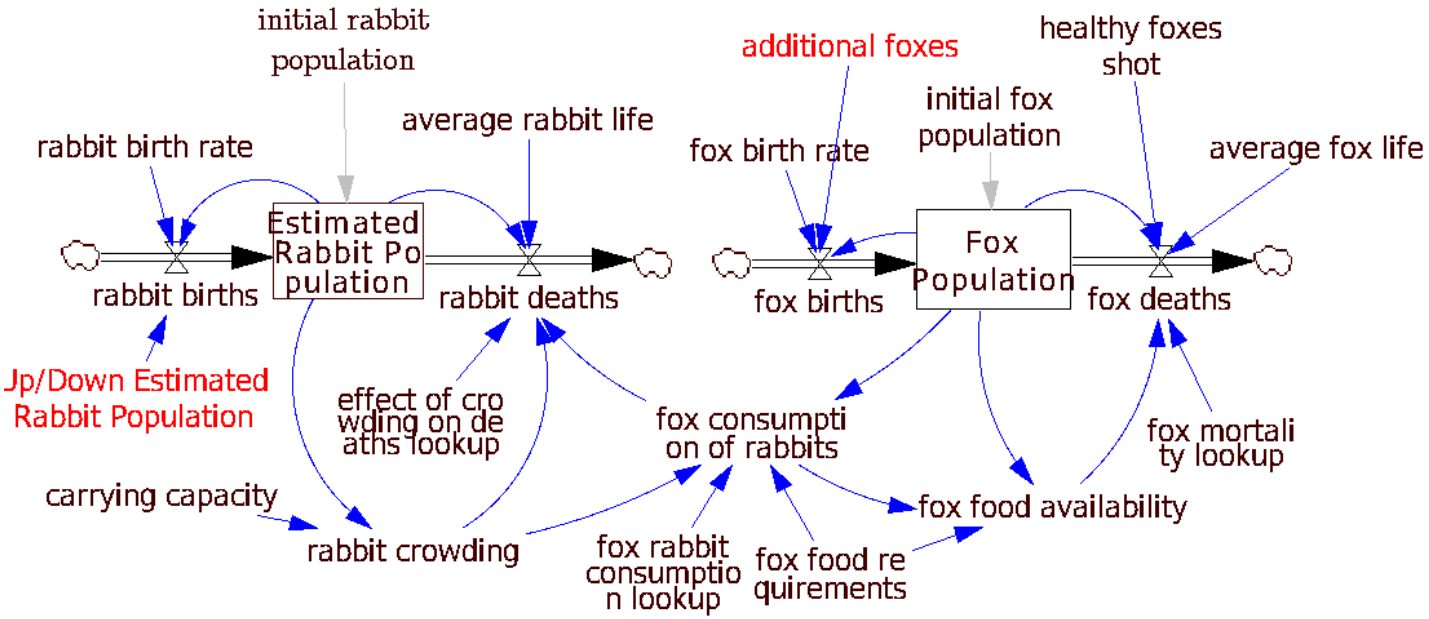
το ρυθμό γεννήσεων των κουνελιών και το ρυθμό με τον οποίο καταναλώνονται από τις αλεπούδες.

Από την άλλη ο αριθμός των αλεπούδων και λαγών που υπάρχουν στο σύστημα μπορεί να περιγραφεί με ένα stock “Estimated Rabbit Population” και “Fox Population” αντίστοιχα. Οι μεταβλητές αυτές αναπαριστούν πρακτικά την κατάσταση του συστήματος που μας ενδιαφέρει ανά πάσα στιγμή. Διάφορες μεταβλητές (variables) συμπληρώνουν το μοντέλο με βοηθητικά στοιχεία (για παράδειγμα ο αρχικός αριθμός των αλεπούδων).

Τα stocks συνδέονται με τα flows με συναρτήσεις, όπως οι ακόλουθες:

```
rabbit births = Estimated Rabbit Population * rabbit birth rate
```

Ο ρυθμός πολλαπλασιασμού των λαγών εξαρτάται από τον συνολικό πληθυσμό τους στην περιοχή και από τις διαθέσιμες τροφές που υπάρχουν στην γύρω περιοχή. Ο αριθμός των αλεπούδων θα εξαρτηθεί από τις διαθέσιμες ποσότητες τροφίμων στην περιοχή δηλαδή από τον αριθμό των λαγών. Η εξομοίωση του συστήματος θα γίνει για διάφορες χρονικές στιγμές από t=0 έως N. Το μοντέλο (Pruyt, 2011) στην επόμενη σελίδα αναπαριστά γραφικά το προηγούμενο παράδειγμα.



Εικόνα 2:

Παράδειγμα μοντέλου system dynamics

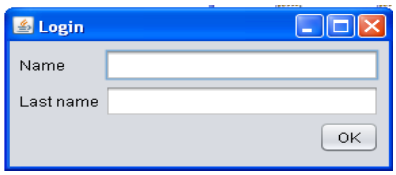
2.2 Λειτουργίες της πρώτης έκδοσης

Στην πρώτη του έκδοση το PROMASI υλοποιεί τις εξής βασικές λειτουργίες.

- Εισαγωγή στο σύστημα
- Επιλογή κατάστασης λειτουργίας (PlayMode)
- Επιλογή παιχνιδιού
- Πρόσληψη προσωπικού
- Προγραμματισμός του έργου με το διάγραμμα Gantt.

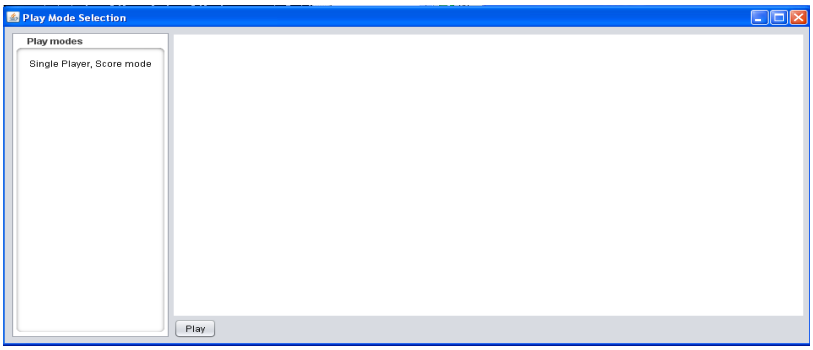
Εισαγωγή στο σύστημα.

Ο χρήστης εισάγει το όνομα και τον κωδικό πρόσβασης για να αποκτήσει την πρόσβαση στο profile του. Όταν ο χρήστης πατήσει το κουμπί OK εμφανίζεται το μενού επιλογής παιχνιδιού.



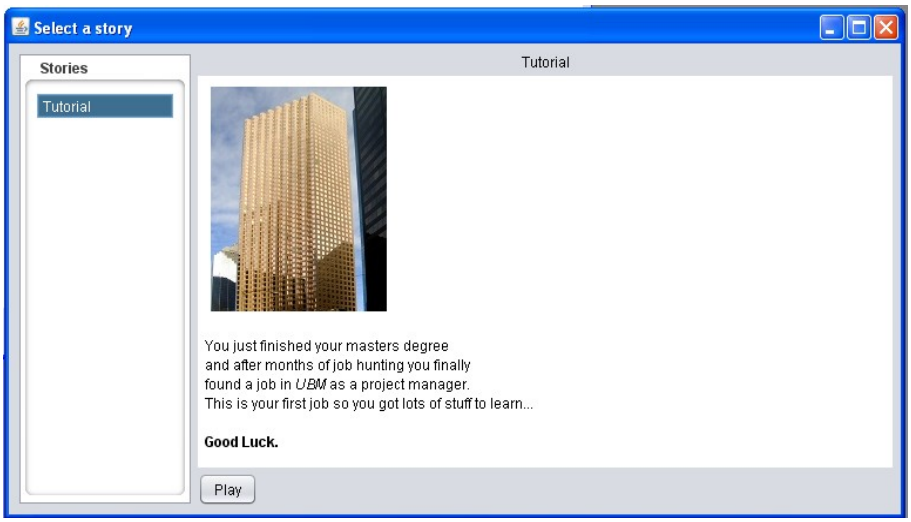
Επιλογή PlayMode.

Σήμερα στο PROMASI υπάρχει μόνο το SinglePlayer mode και ο σκοπός της πτυχιακής αυτής είναι η προσθήκη ενός ακόμα mode (MultiPlayer) που θα δίνει την δυνατότητα στον παίκτη να παίζει online και να ανταγωνιστεί με τους άλλους παίκτες που θα μπουν στο ίδιο παιχνίδι με αυτόν.



Επιλογή παιχνιδιού – Ιστορίας

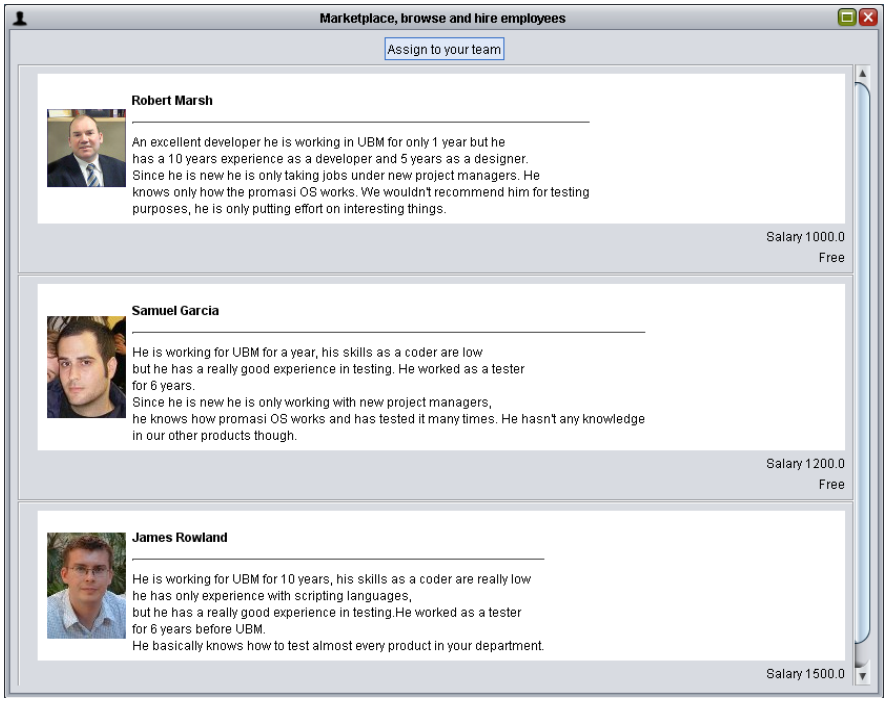
Όταν ολοκληρωθεί η διαδικασία εισόδου στο σύστημα και η επιλογή του mode το PROMASI εμφανίζει το μενού επιλογής των ιστοριών που είναι διαθέσιμες στο σύστημα. Μια ιστορία περιγράφει μια σειρά από έργα καθώς και μία επιχείρηση που καλείται να τα ολοκληρώσει. Το μενού επιλογής εμφανίζει το όνομα της ιστορίας και την περιγραφή της στο διπλανό πλαίσιο στην περίπτωση που ο χρήστης αποφασίσει να επιλέξει την συγκεκριμένη ιστορία.



Πρόσληψη προσωπικού.

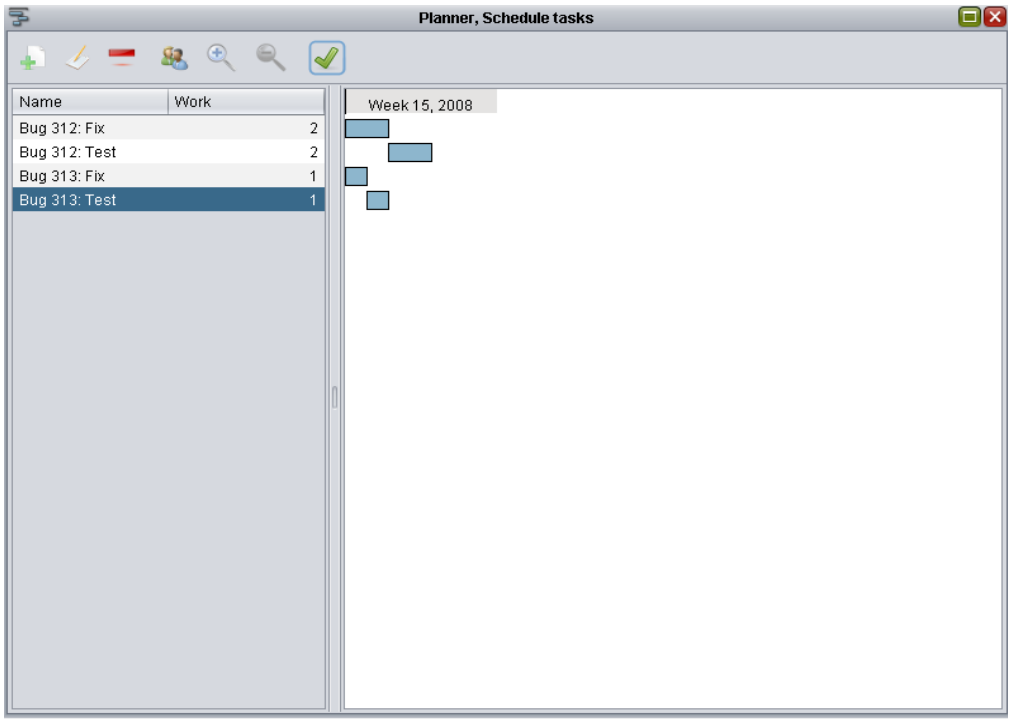
Μετά την επιλογή της ιστορίας το παιχνίδι ξεκινά και ο χρήστης έχει την δυνατότητα να προσλάβει και να απολύσει το προσωπικό της εταιρίας.

Στο παράθυρο διαχείρισης προσωπικού φαίνονται τα βασικά χαρακτηριστικά των εργαζομένων σε μορφή βιογραφικού σημειώματος. Τα χαρακτηριστικά που αναγράφονται στο βιογραφικό σημείωμα του κάθε εργαζομένου επηρεάζουν σημαντικά την εξέλιξη του έργου κατά την διάρκεια του παιχνιδιού.



Προγραμματισμός του έργου.

Μετά την επιλογή της ιστορίας το παιχνίδι ξεκινά και ο χρήστης μπορεί να προγραμματίσει την διαδικασία ανάπτυξης λογισμικού για αυτήν την ιστορία χρησιμοποιώντας το διάγραμμα Gantt.



Όταν ένας χρήστης ξεκινάει το παιχνίδι του PROMASI, εμφανίζεται το κυρίως παράθυρο του παιχνιδιού που θεωρείται ο πίνακας ελέγχου της επιχείρησης. Στον πίνακα ελέγχου υπάρχουν διάφορες επιλογές: Ο χρήστης μπορεί να προσλάβει προσωπικό και να προγραμματίσει την διαδικασία της ανάπτυξης του λογισμικού, χτίζοντας τα διαγράμματα Gantt. Η κάθε ενέργεια του χρήστη επηρεάζει άμεσα τα αποτελέσματα που θα παράγει το μοντέλο προσομοίωσης. Ο στόχος του PROMASI είναι να μάθει στους εκπαιδευόμενους να προγραμματίζουν την διαδικασία ανάπτυξης λογισμικού με τέτοιον τρόπο ώστε

- Να μειώνουν τον ρίσκο αποτυχίας του έργου.
- Να μειώνουν το κόστος παραγωγής λογισμικού.

2.3 Άλλες λειτουργίες

Εκτός από την εφαρμογή προσομοίωσης το PROMASI περιλαμβάνει μία εφαρμογή σχεδιασμού των μοντέλων προσομοίωσης απαραίτητων για την λειτουργία του παιχνιδιού. Η εφαρμογή αυτή, με το όνομα Core Designer, χρησιμοποιεί συγκεκριμένης δομής γλώσσα με την οποία κατασκευάζονται τα μοντέλα. Η γλώσσα αυτή είναι όμοια με τη γλώσσα σχεδίασης του εργαλείου εξομοίωσης system dynamics, Vensim, με κάποιες μικρές παραλλαγές που εξυπηρετούν τους εσωτερικούς μηχανισμούς του PROMASI.

Με την βοήθεια του Core Designer μπορούν να σχεδιαστούν διάφορα μοντέλα προσομοίωσης που προσομοιώνουν το σύστημα παραγωγής προϊόντων λογισμικού. Τα μοντέλα αυτά όμως θα πρέπει να χαρακτηρίζονται από κάποιες σταθερές εισόδους και εξόδους ώστε να μπορούν να χρησιμοποιηθούν στο PROMASI.

Για παράδειγμα μια από τις εισόδους αυτές είναι οι υπάλληλοι με τα χαρακτηριστικά τους. Κάθε υπάλληλος της εταιρίας έχει ένα σύνολο από δεξιότητες ή “χαρακτηριστικά” και ο στόχος του παίκτη είναι να ορίσει στους υπαλλήλους τις κατάλληλες εργασίες ώστε να πετύχει την καλύτερη αξιοποίηση των δεξιοτήτων τους. Μια δεύτερη είσοδος είναι το χρονοδιάγραμμα του έργου χωρισμένο σε εργασίες σε μορφή διαγράμματος Gantt. Το διάγραμμα Gantt περιγράφει με ποια σειρά και από ποιους θα υλοποιηθούν οι εργασίες του έργου. Ο παίκτης πρέπει να ορίσει την σωστή σειρά με την οποία θα υλοποιηθούν οι εργασίες λαμβάνοντας υπόψη την σημασία της κάθε εργασίας για το συνολικό έργο.

Η πρώτη έξοδος που λαμβάνει κανείς παίζοντας το παιχνίδι είναι το έργο που απομένει να υλοποιηθεί. Μια άλλη έξοδος που δείχνει την επιτυχία του διαχειριστή έργου είναι το τελικό αποτέλεσμα – σκορ του παιχνιδιού απεικονισμένο στο οικονομικό κομμάτι της εταιρίας, δηλαδή τον προϋπολογισμό του έργου και το μερίδιο της αγοράς που καταλαμβάνει, που στη γλώσσα του PROMASI λέγονται “prestige points”.

2.4 Μειονεκτήματα της πρώτης έκδοσης

- Δεν υπάρχουν κατάλληλα μοντέλα προσομοίωσης.

  Ο βασικός λόγος απουσίας ενός καλού μοντέλου προσομοίωσης, δηλαδή μοντέλου που θα μπορούσε να περιγράψει όλη την διαδικασία ανάπτυξης του έργου με την απαραίτητη λεπτομέρεια, είναι η απουσία μιας εύχρηστης εφαρμογής σχεδίασης μοντέλων προσομοίωσης, εφαρμογής που θα μπορούσε να καλύψει όλες τις ανάγκες στην κατασκευή ενός πολύπλοκου μοντέλου με πολλαπλές διεργασίες.

- Δεν υπάρχει η δυνατότητα απολύσεως των εργαζομένων.

  Σήμερα το PROMASI υλοποιεί την περίπτωση που ονομάζεται “πρόσληψη προσωπικού” όμως αν κατά διάρκεια του παιχνιδιού ο παίκτης αντιληφθεί ότι υπάρχει ανάγκη να απολύσει κάποιον από τους υπαλλήλους του λόγω μικρού προϋπολογισμού για παράδειγμα το PROMASI δεν παρέχει αυτήν την δυνατότητα.

- Το μοντέλο προσομοίωσης περιγράφει συνολικά το έργο και για αυτό είναι πολύπλοκο και δύσκολο στην υλοποίηση.

  Σήμερα το PROMASI χειρίζεται μονολιθικά μοντέλα προσομοίωσης που σημαίνει ότι όλες οι εργασίες που περιγράφουν το project ανήκουν σε ένα ενιαίο μοντέλο. Αυτό όμως δυσκολεύει την ανάπτυξη των μοντέλων προσομοίωσης κυρίως λόγω πολυπλοκότητας τους.

- Η αγορά εργασίας είναι στατική.

  Κατά την εξέλιξη του παιχνιδιού μόνο ο παίκτης έχει πρόσβαση στην αγορά εργασίας και οποιαδήποτε στιγμή μπορεί να προσλάβει οποιονδήποτε ελεύθερο υπάλληλο στην εταιρία του. Στην πραγματικότητα όμως ο υπάλληλος αποφασίζει αν θέλει να δουλέψει στην εταιρία που του κάνει πρόταση για μια θέση.

- Τα χαρακτηριστικά του προσωπικού παραμένουν σταθερά - δεν αλλάζουν με την πάροδο του χρόνου

  Έτσι τα χαρακτηριστικά του προσωπικού δεν εξελίσσονται κατά την διάρκεια υλοποίησης ενός έργου κάτι που δεν συμβαίνει στην πραγματική ζωή. Για παράδειγμα αν ένας εργαζόμενος δουλεύει για πολύ καιρό σε μια συγκεκριμένη θέση αποκτά την εμπειρία άρα βελτιώνει κάποια από τα χαρακτηριστικά του, επίσης αν ο εργαζόμενος δουλεύει πολλές ώρες η κούραση πρέπει να απεικονίζεται στα χαρακτηριστικά του. Η κούραση είναι μια από τις σημαντικότερες παραμέτρους που πρέπει να λαμβάνει υπόψη του ο Project Manager.

- Δεν υπάρχει μέτρο σύγκρισης του τελικού αποτελέσματος αν στο παιχνίδι πήρε μέρος μόνο ένα άτομο.

  Το Single Player mode δεν μπορεί να δώσει την ρεαλιστική εικόνα για το πόσο τελικά καλός είναι ο παίκτης που εκτελεί τα καθήκοντα του Project Manager λόγω του ότι ο παίκτης δεν μπορεί να συγκρίνει τα αποτελέσματα του με αντίστοιχα ενός άλλου παίκτη.

- Δεν υπάρχει ο ανταγωνισμός που χαρακτηρίζει τις σύγχρονες επιχειρήσεις.

  Στην πραγματικότητα οι επιχειρήσεις ανταγωνίζονται ώστε να αποκτήσουν το καλύτερο δυνατό προσωπικό στο PROMASI όμως δεν συμβαίνει κάτι τέτοιο αντιθέτως ο χρήστης έχει την δυνατότητα να προσλάβει οποιονδήποτε διαθέσιμο ελεύθερο επαγγελματία από την αγορά εργασίας.

Τέλος σε επίπεδο αρχιτεκτονικής το PROMASI ακολούθησε την αρχιτεκτονική των enterprise applications (Wikipedia, “Enterprise Software”). Enterprise applications είναι οι εφαρμογές που προορίζονται κυρίως στις επιχειρήσεις που ασχολούνται με την διαχείριση μιας βάσης δεδομένων όπου αποθηκεύονται τα στοιχεία των πελατών ή και τα προϊόντα τους. Ο στόχος των enterprise applications είναι να διευκολύνει την πρόσβαση και την αποθήκευση των δεδομένων απαραίτητων για την λειτουργία μιας τέτοιας επιχείρησης. Η βασική δομή μιας τέτοιας εφαρμογής περιγράφεται από τα τρία βασικά συστατικά

Database Access module, Model και Graphical User Interface. Το PROMASI όμως δεν ανήκει στην κατηγορία των enterprise applications αλλά είναι ένας προσομοιωτής. Η χρήση ακατάλληλης αρχιτεκτονικής στο PROMASI περιορίζει την επέκτασή της εφαρμογής σε πολλούς ταυτόχρονους χρήστες.

2.5 Παρόμοια Projects

Εκτός από το PROMASI υπάρχουν και άλλα προγράμματα που προσπαθούν και επιτυγχάνουν την προσομοίωση ανάπτυξης έργων λογισμικού. Μερικά από αυτά είναι:

1. **SESAM** (Ludewig et al, 1996) Είναι ένας από τους πρώτους προσομοιωτές που αναπτύχθηκαν για τους εκπαιδευτικούς σκοπούς. Η λειτουργία του βασίζεται στα μοντέλα προσομοίωσης που είναι γραμμένα σε μια ιδική γλώσσα που χρησιμοποιείται στο SESAM. Η βασική ιδέα του SESAM είναι ίδια με αυτήν του PROMASI. Ο εκπαιδευτής κατασκευάζει ένα μοντέλο για την προσομοίωση λειτουργίας μιας εταιρίας και ο εκπαιδευόμενος προσπαθεί να υλοποιήσει έργο βασισμένο σ' αυτό το μοντέλο.

2. **SIMSE** (Navaro, 2006) Προσομοιωτής που και αυτός με την σειρά υλοποιήθηκε για τους εκπαιδευτικούς σκοπούς. Το SIMSE παρέχει μια γλώσσα με την οποία ο χρήστης έχει την δυνατότητα να περιγράφει την λειτουργία της επιχείρησής χτίζοντας αντικείμενα και τις σχέσεις μεταξύ τους. Αυτά τα αντικείμενα μπορεί να περιγράφουν τους πόρους που διαχειρίζεται η εταιρία. Οι πόροι αυτοί μπορούν να είναι το προσωπικό, τα μηχανήματα, τα έγγραφα της εταιρίας κ. α. Εκτός από τους πόρους ο χρήστης έχει την δυνατότητα να σχεδιάσει τις συσχετίσεις που υπάρχουν ανάμεσα τους. Για παράδειγμα ο υπάλληλος μπορεί να χρησιμοποιεί το έγγραφο για να υλοποιεί το έργο που του έχει ανατεθεί.

3. **TOPSIM**. Προσομοιωτής του Project Manager έχει την υλοποίηση του MultiPlayer mode. Το κύριο μειονέκτημά του είναι ότι ανήκει στην κατηγορία εμπορικών προϊόντων.

4. **Sharkworld**. Ακόμα ένα εμπορικό προϊόν με όνομα Sharkworld. Είναι ένα online παιχνίδι προσομοίωσης βασίζεται στην τεχνολογία Flash και απαιτεί εγγραφή για την δοκιμαστική έκδοση.

3 PROMASI 2

Στην πρώτη του έκδοση το PROMASI υποστηρίζει μόνο το Singleplayer mode και για αυτό δεν μπορεί να θεωρηθεί ότι προσομοιώνει ρεαλιστικά τις συνθήκες της αγοράς. Για να γίνει πιο ρεαλιστική η προσομοίωση, πάρθηκε η απόφαση να υλοποιηθεί το Multiplayer mode. Στο Multiplayer mode το PROMASI θα πρέπει να υποστηρίζει πολλούς παίκτες που θα μπορούν να παίζουν ταυτόχρονα υλοποιώντας ίδια έργα, και να μοιράζονται την ίδια αγορά εργασίας.

3.1 Role-playing games (RPGs)

RPG είναι μια από τις κατηγορίες παιχνιδιών που το κυρίως χαρακτηριστικό τους είναι η ανάθεση κάποιου ρόλου στον παίκτη. Ο παίκτης εκτελεί καθήκοντα του χαρακτήρα του, μαζεύοντας εμπειρία από τις αποστολές που εκτελεί. Υπάρχουν πολλά γνωστά και επιτυχημένα παιχνίδια της κατηγορίας RPG. Όλα όμως αυτά τα παιχνίδια εστιάζονται σε εικονικούς κόσμους και κανένα από αυτά δεν είναι φτιαγμένο για εκπαιδευτικούς σκοπούς. Το PROMASI μπορεί να πάρει κάποια στοιχεία των RPG παιχνιδιού ώστε να γίνει ποιο διασκεδαστικό σαν παιχνίδι. Η βασική ιδέα του RPG στο PROMASI είναι η ταυτόχρονη συμμετοχή πολλών παικτών που θα παίζουν διαφορετικούς ρόλους αλλά θα επικοινωνούν μεταξύ τους μέσω της αγοράς εργασίας. Στο βάθος χρόνου ο κάθε παίκτης εξελίσσει την επιχείρησή του καταλαμβάνοντας μεγαλύτερο μερίδιο αγοράς και ανεβάζοντας το rating της επιχείρησης.

3.2 MultiPlayer

Ο σκοπός του MultiPlayer mode είναι η εξέλιξη του PROMASI ως διαδικτυακού παιχνιδιού στρατηγικής με χαρακτηριστικά RPG. Ο κάθε παίκτης του παιχνιδιού που έχει εγγραφεί στους χρήστες του PROMASI θα έχει μία επιχείρηση εγγεγραμμένη στον λογαριασμό του. Η επιχείρηση αυτή θα χαρακτηρίζεται από τον προϋπολογισμό που διαθέτει και από τα prestige points (βαθμούς) πού έχει αποκτήσει κατά την διάρκεια λειτουργίας της. Ο προϋπολογισμός στην συγκεκριμένη περίπτωση είναι πολύ σημαντικός παράγοντας για την λειτουργία της επιχείρησης. Από αυτόν εξαρτάται το πόσους υπαλλήλους και με ποια προσόντα μπορεί να κρατήσει η επιχείρησή και επίσης το πόσο μεγάλα έργα μπορεί να αναλάβει. Αν κατά την διάρκεια του παιχνιδιού ο προϋπολογισμός της επιχείρησης δεν

αρκεί για να καλύψει τις ανάγκες της, όπως οι μισθοδοσίες, τότε ο παίκτης αποχωρεί από το τρέχον παιχνίδι και ταυτόχρονα μειώνονται τα prestige points της επιχείρησής του ενώ επιστρέφονται όλα τα χρήματα που έχει σπαταλήσει κατά την διάρκεια του συγκεκριμένου παιχνιδιού για να μπορεί να συνεχίσει στο μέλλον.

Η κοινή αγορά εργασίας είναι αυτή που θα δίνει δυναμισμό στο παιχνίδι δημιουργώντας στους χρήστες την αίσθηση λειτουργίας της επιχείρησης στο πραγματικό περιβάλλον. Οι παίκτες θα ανταγωνίζονται ανεβάζοντας τη βαθμολογία της επιχείρησης τους ώστε να γίνει πιο ανταγωνιστική στην αγορά εργασίας.

Αν μια επιχείρηση έχει χαμηλό rating τότε έχει και την μικρότερη πιθανότητα να προσλάβει έμπειρο προσωπικό, όσο αυξάνεται το rating της επιχείρησης τόσο αυξάνονται και οι πιθανότητες πρόσληψης. Αν ένας χρήστης πιστεύει ότι την συγκεκριμένη περίοδο χρειάζεται να προσλάβει υπαλλήλους τότε καταθέτει μία αγγελία στην αγορά εργασίας για την πρόσληψη προσωπικού. Αν η αγορά εργασίας διαθέτει άτομα που έχουν τις κατάλληλες απαιτήσεις ώστε να προσληφθούν στην εταιρία τότε ο χρήστης μπορεί να κάνει πρόσληψη το άτομο που πιστεύει ότι θα βοηθήσει στην υλοποίηση του έργου.

3.3 Νέες λειτουργίες

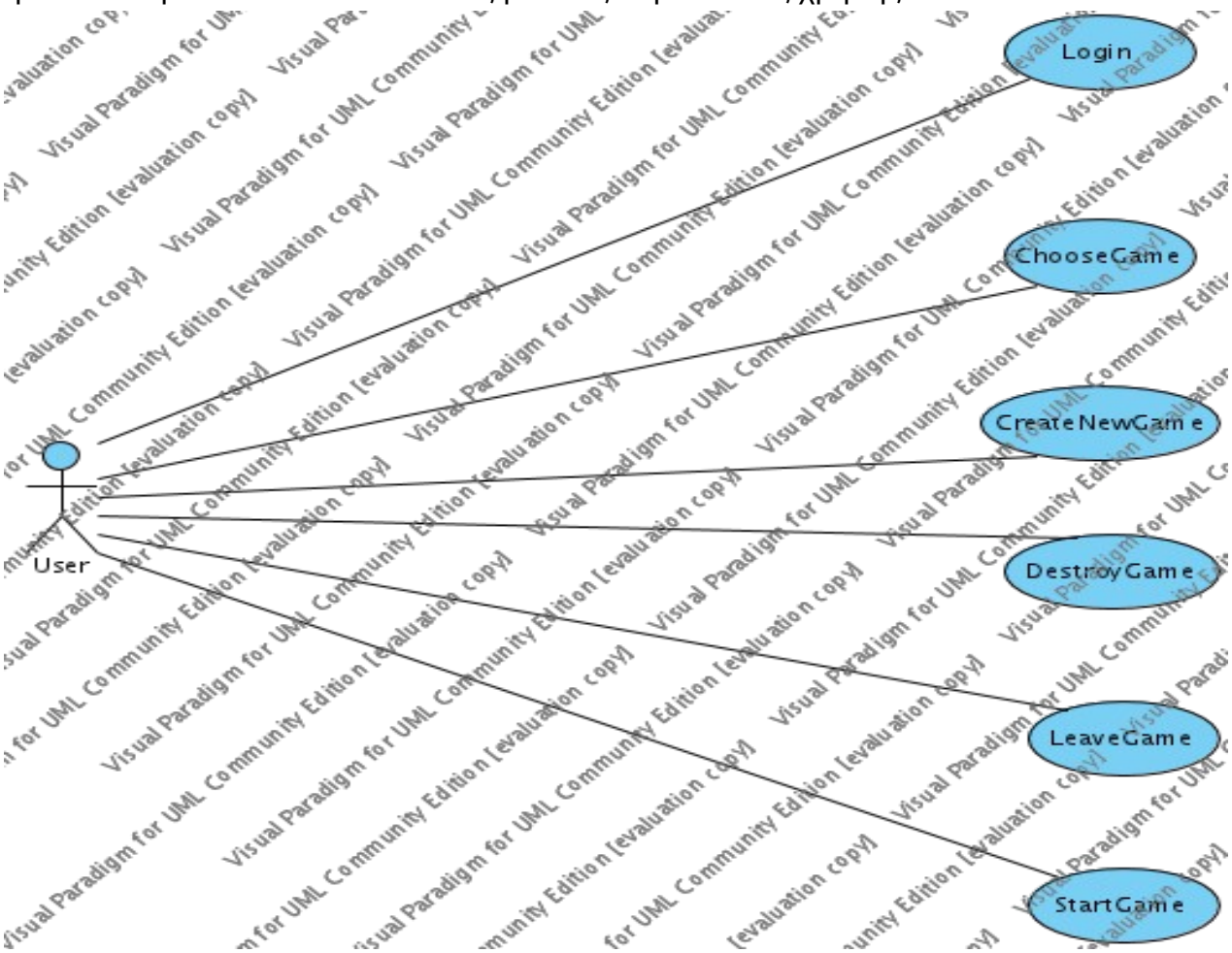
Για την υλοποίηση του Multiplayer mode στο PROMASI χρειάζεται να προστεθούν νέες λειτουργίες και να βελτιωθούν οι ήδη υπάρχουσες. Για παράδειγμα σήμερα η λειτουργία διαχείρισης προσωπικού δεν μπορεί να καλύψει το Multiplayer mode . Επίσης χρειάζεται να προστεθούν νέα μενού εισαγωγής που να δίνουν στον χρήστη την δυνατότητα επιλογής του παιχνιδιού που θέλει παίξει. Ο χρήστης θα μπορεί να έχει μία η περισσότερες εταιρίες στον λογαριασμό του. Η κατάσταση της κάθε εταιρίας θα αποθηκεύεται στην βάση δεδομένον. Ο χρήστης θα πρέπει να διαλέξει την επιχείρηση με την οποία θα παίξει το παιχνίδι πριν την επιλογή του παιχνιδιού. Όταν τελειώνει ένα παιχνίδι η επιχείρηση του κάθε παίκτη βαθμολογείται με βάση τα αποτελέσματα που έφερε σε σύγκριση με τους υπόλοιπους παίκτες. Με βάση αυτήν την βαθμολογία και ανάλογος του rating των αντιπάλων ανανεώνεται και το rating της επιχείρησης του χρήστη. Κάθε υπάλληλος θα περιγράφεται από ένα σύστημα προσομοίωσης ώστε να διαμορφώνονται τα χαρακτηριστικά του κατά την εξέλιξη του παιχνιδιού.

3.4 Άλλες λειτουργίες

Ελεύθεροι επαγγελματίες στην αγορά εργασίας να παίρνουν απόφαση αν θέλουν να εργαστούν στην εταιρία που προσφέρει την θέση η όχι ανάλογα με κάποια χαρακτηριστικά της εταιρίας. Εταιρίες μπορούν να χωριστούν σε κατηγορίες, πολυεθνική, μεσαία, μικρή, αναλόγως του μεγέθους της εταιρίας η εταιρία αυτή έχει πρόσβαση σε διαφορετική αγορά εργασίας. Οι κατηγορίες των εταιριών μπορούν να περιγραφούν από το rating. Εξέλιξη του παιχνιδιού σαν ένα RPG με βάση τα παραπάνω.

3.5 Περιπτώσεις χρήσης

Από την παραπάνω ανάλυση προκύπτει ότι για την υλοποίηση του Multiplayer mode πρέπει να προστεθούν οι ακόλουθες βασικές περιπτώσεις χρήσης.



Εισαγωγή στο σύστημα	
Χρήστης	Σπουδαστής
Προαπαιτούμενα	Ο χρήστης έχει επιλέξει την επιλογή MultiPlayer mode στο παράθυρο επιλογής του τύπου του παιχνιδιού και η εφαρμογή client έχει συνδεθεί στον server.
Κυρίως σενάριο	Ο χρήστης πληκτρολογεί το όνομα και τον κωδικό πρόσβασης στο σύστημα και πατάει το κουμπί εισαγωγής και εισέρχεται στο σύστημα. Εμφανίζεται το παράθυρο επιλογής παιχνιδιού.
Εναλλακτικά σενάρια	**1.** Ο χρήστης κλείνει το παράθυρο εισαγωγής. Εμφανίζεται το παράθυρο επιλογής τύπου παιχνιδιού. **2.** Αποτυχία στην εισαγωγή στο σύστημα ο κωδικός πρόσβαση η το όνομα του χρήστη είναι λάθος. Εμφανίζεται το κατάλληλο μήνυμα αποτυχίας.

Επιλογή παιχνιδιού	
Χρήστης	Σπουδαστής
Προαπαιτούμενα	Ο χρήστης έχει εισέλθει στο σύστημα του PROMASI και στο σύστημα υπάρχουν διαθέσιμα παιχνίδια που βρίσκονται στην κατάσταση αναμονής.
Κυρίως σενάριο	Ο χρήστης επιλέγει ένα παιχνίδι από την λίστα με τα διαθέσιμα παιχνίδια και πατάει το κουμπί εισαγωγή.
Εναλλακτικά σενάρια	**1.** Ο χρήστης κλείνει το παράθυρο επιλογής παιχνιδιού. Η εφαρμογή τερματίζει.

Δημιουργία νέου παιχνιδιού	
Χρήστης	Σπουδαστής
Προαπαιτούμενα	Ο χρήστης έχει εισέλθει στο σύστημα του PROMASI. Ο χρήστης πατάει το κουμπί δημιουργίας παιχνιδιού στο παράθυρο επιλογής παιχνιδιού και το σύστημα εμφανίζει το παράθυρο δημιουργίας νέου παιχνιδιού.
Κυρίως σενάριο	**1.** Ο χρήστης πληκτρολογεί το όνομα του παιχνιδιού και πατάει το κουμπί δημιουργίας παιχνιδιού. Η εφαρμογή εμφανίζει το παράθυρο αναμονής παικτών το παιχνιδιού.
Εναλλακτικά σενάρια	**1.** Η δημιουργία παιχνιδιού απέτυχε. Ο χρήστης παραμένει στο παράθυρο δημιουργίας παιχνιδιού. **2.** Ο χρήστης κλείνει το παράθυρο δημιουργίας παιχνιδιού και η εφαρμογή εμφανίζει το παράθυρο επιλογής παιχνιδιού.

Κατάργηση του παιχνιδιού	
Χρήστης	Σπουδαστής
Προαπαιτούμενα	Ο χρήστης έχει δημιουργήσει νέο παιχνίδι και βρίσκεται στην κατάσταση αναμονής παικτών.
Κυρίως σενάριο	**1.** Ο χρήστης κλείνει το παράθυρο αναμονής παικτών. Το σύστημα εμφανίζει το παράθυρο επιλογής παιχνιδιού.

Εκκίνηση του παιχνιδιού	
Χρήστης	Σπουδαστής
Προαπαιτούμενα	Ο χρήστης βρίσκεται στο παράθυρο αναμονής παικτών του παιχνιδιού.
Κυρίως σενάριο	**1.** Ο χρήστης πατάει το κουμπί εκκίνησης του παιχνιδιού. Το σύστημα εμφανίζει το παράθυρο εξέλιξης του παιχνιδιού.

Έξοδος από το παιχνίδι	
Χρήστης	Σπουδαστής
Προαπαιτούμενα	Ο χρήστης βρίσκεται στο παράθυρο εξέλιξης του παιχνιδιού.
Κυρίως σενάριο	**2.** Ο χρήστης κλείνει το παράθυρο εξέλιξης του παιχνιδιού. Το σύστημα εμφανίζει το παράθυρο επιλογής παιχνιδιού.

3.6 Βελτιώσεις

3.6.1 Αρχιτεκτονική του PROMASI.

Σήμερα η οργάνωση του έργου του PROMASI περιγράφεται ως ένα μονολιθικό πρόγραμμα που συνδυάζει μέσα του το κυρίως παιχνίδι και το πρόγραμμα σχεδίασης μοντέλων προσομοίωσης. Όμως αυτό το μοντέλο ανάπτυξης έχει πολλά μειονέκτημά και το μεγαλύτερο από όλα αυτά είναι η δύσκολη παρακολούθηση και συντήρηση του κώδικα.

Εκτός από αυτό το μοντέλο ανάπτυξης ενός έργου υπάρχουν και διαφορές εναλλακτικές λύσεις. Μία από αυτές τις λύσεις είναι ο χωρισμός του έργου σε μικρότερα κομμάτια ή “βιβλιοθήκες”. Κάθε βιβλιοθήκη πρέπει να υλοποιεί μια συγκεκριμένη λειτουργία και να εμφωλιάζει μέσα της την λογική υλοποίησης για να κρύβει την πολυπλοκότητα της υλοποίησης μέσα της. Η καλή σχεδίαση μιας βιβλιοθήκης οδηγεί στην εύκολη χρήση και την επαναχρησιμοποίηση της. Είναι πολύ σημαντικό να διατηρούμε το API όσο πιο απλό γίνεται ώστε να μην φορτώνουμε τον προγραμματιστή, που θα χρησιμοποιήσει την βιβλιοθήκη μας, με πληροφορίες τις οποίες δεν είναι απαραίτητο να τις γνωρίζει. Στον αντικειμενοστραφή προγραμματισμό λοιπόν υπάρχει η έννοια της ενθυλάκωσης (encapsulation) η οποία περιγράφει την ιδιότητα της εξαφάνισης της άχρηστης για τον προγραμματιστή πληροφορίας κάτω από μια καλά σχεδιασμένη διεπαφή.

3.6.2 Γραφικό περιβάλλον.

Το γραφικό περιβάλλον του PROMASI σήμερα βασίζεται στην βιβλιοθήκη Swing (Walrath, 2004) η οποία παρέχει πολλές έτοιμες λύσεις, για τους προγραμματιστές στην υλοποίηση του γραφικού περιβάλλοντος. Το βασικό μειονέκτημα της στην περίπτωση του PROMASI είναι η απουσία της έτοιμης υλοποίησης Component που να παρέχει την δυνατότητα της σχεδίασης του διαγράμματος Gantt. Επειδή η υλοποίηση ενός τέτοιου Component με την χρήση της Swing library απαιτεί πολύ χρόνο και θα μπορούσε να γίνει θέμα μιας άλλης πτυχιακής εργασίας πήραμε την απόφαση να χρησιμοποιήσουμε την βιβλιοθήκη που θα μας έδινε την έτοιμη λύση για την κατασκευή του διαγράμματος Gantt. Μια τέτοια βιβλιοθήκη είναι η SWT library (Guojie, 2004). Για περισσότερες πληροφορίες σχετικά με την SWT δείτε και το παράρτημα που αφορά τα εργαλεία ανάπτυξης του PROMASI.

3.6.3 Αγορά εργασίας.

Στο Multiplayer mode η αγορά εργασίας είναι κοινή για όλους τους παίκτες του παιχνιδιού για αυτό η επιλογή ενός υπαλλήλου από κάποιον παίκτη επηρεάζει την συνολική αγορά εργασίας με αποτέλεσμα οι άλλοι παίκτες να μην μπορούν να προσλάβουν τον ίδιο υπάλληλο στην εταιρία τους. Η αγορά εργασίας θα είναι πλήρως συγχρονισμένη για όλους τους παίκτες του ίδιου παιχνιδιού. Η ενέργεια του ενός παίκτη θα απεικονίζεται την ίδια χρονική στιγμή στις οθόνες των υπολοίπων.

3.6.4 Μενού επιλογής παιχνιδιού.

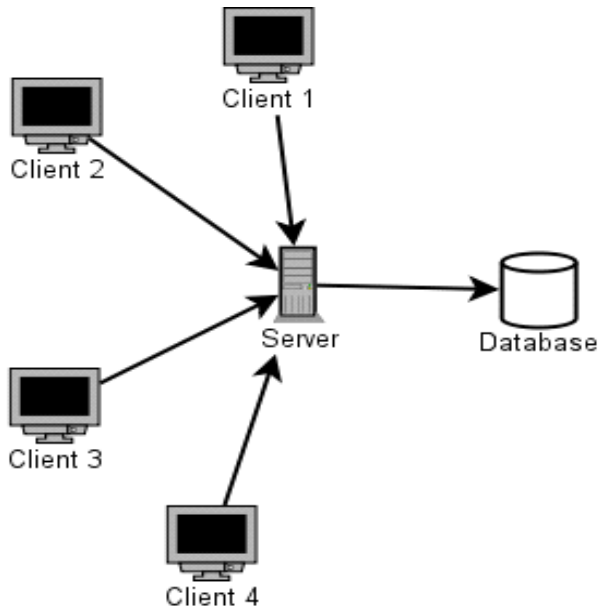
Σήμερα το PROMASI υλοποιεί την διαδικασία εισαγωγής στο σύστημα στον οδηγό επιλογής παιχνιδιού. Όμως η διαδικασία αυτή δεν είναι απαραίτητη για το παιχνίδι στο οποίο παίρνει μέρος μόνο ένας χρήστης. Άρα ο οδηγός επιλογής παιχνιδιού μπορεί να αλλάξει ώστε να παραβλέπει αυτήν την άσκοπη διαδικασία. Εκτός από αυτό θα προστεθεί ένα νέο κλάδος στον οδηγό που θα αφορά το MultiPlayer mode.

3.6.5 Διαχείριση προσωπικού.

Στη νέα έκδοση του PROMASI αλλάζει το μενού διαχείρισης προσωπικού όπου θα εμφανίζονται πλέον δυο tabs τα οποία θα περιγράφουν την τρέχουσα κατάσταση του προσωπικού της επιχείρησης και την τρέχουσα κατάσταση της αγοράς εργασίας αντίστοιχα. Κάθε επαγγελματίας στο PROMASI αναπαριστάται με ένα μοναδικό κωδικό για όλους του παίκτες του παιχνιδιού.

3.6.6 Τοπολογία του PROMASI

Το PROMASI πρέπει να χρησιμοποιεί την αρχιτεκτονική του πελάτη-εξυπηρετητή για να υλοποιεί το Multilayer mode, και για αυτό θα αποτελείται από δύο κομμάτια: τον εξυπηρετητή (server) και τον πελάτη (client). Κάθε χρήστης θα έχει στην διάθεση του ένα πρόγραμμα client στο οποίο θα χειρίζεται το παιχνίδι. Το πρόγραμμα server πρέπει να τρέχει σε μια γνωστή τοποθεσία ώστε να την γνωρίζει ο κάθε client. Το παρακάτω διάγραμμα περιγράφει την δομή του συστήματος



3.7 Πρωτόκολλο επικοινωνίας

Για την διασύνδεση τον clients με τον server, επιλέχθηκε το πρωτόκολλο TCP/IP Transmission Control Program/Internet Protocol. Για την ορθή επικοινωνία μεταξύ client-server αναπτύχθηκε ένα πρωτόκολλο που βασίζεται στην περιγραφική γλώσσα XML.

3.8 Εναλλακτικές λύσεις στην επικοινωνία

Για την υλοποίηση του MultiPlayer mode στο PROMASI επιλέχθηκε το πρωτόκολλο Tcp με σταθερή σύνδεση.

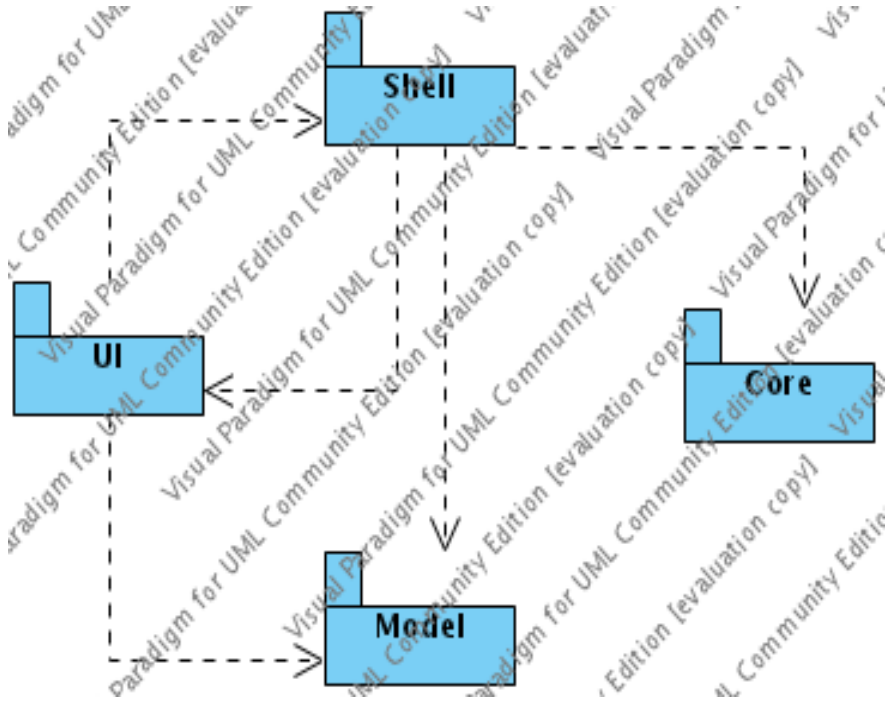
'Ομως υπάρχουν και άλλοι τρόποι που θα μπορούσαν να χρησιμοποιηθούν στην επικοινωνία. Ένας από αυτούς είναι η χρήση των WebServices (Papazoglou, 2007) με διατήρηση της Session του χρήστη. Τα WebServices μπορούν να χρησιμοποιήσουν διάφορα πρωτόκολλα για την μεταφορά δεδομένων “SOAP, JSON κ.α.”. Είναι ευρέως διαδεδομένα και χρησιμοποιούνται από διάφορες ομάδες και εταιρίες. Πολλές εταιρίες παρέχουν online APIs για τις web εφαρμογές που στηρίζονται στα WebServices. Η υλοποίηση των WebServices υπάρχει σε διάφορες πλατφόρμες και γλώσσες προγραμματισμού.

Το κύριο μειονέκτημα τους είναι η ταχύτητα, που είναι μικρή λόγω της χρήσης του πρωτοκόλλου HTTP μέσω της διασύνδεσης TCP η οποία είναι σχεδιασμένη για επικοινωνία άλλου τύπου. Εκτός από την ταχύτητα επικοινωνίας στο PROMASI ο server πρέπει να γνωρίζει για την τρέχουσα κατάσταση του κάθε client συνδεδεμένου σε αυτόν και να τον ενημερώνει για τις εσωτερικές αλλαγές του. Για να το πετύχουν αυτό οι web εφαρμογές χρησιμοποιούν την τεχνική βασισμένη στην δημιουργία και διατήρηση της Session για κάθε client. Η διατήρηση της Session για κάθε client που επικοινωνεί με τον server απαιτεί την εγκατάσταση και ρύθμιση μιας εφαρμογής WebServer. Η Microsoft παρέχει μια λύση στην επικοινωνία χρησιμοποιώντας τα WCF (Cibraro et al 2010) μέσω της πλατφόρμας .NET χωρίς την απαραίτητη παρουσία του WebServer αλλά η λύση αυτή προϋποθέτει την κατασκευή ενός μηχανισμού που θα προσομοίωνε το Session του χρήστη. Τελικά τα WebServices είναι κατασκευασμένα για άλλου είδους εφαρμογές και για αυτόν τον λόγο δεν χρησιμοποιήθηκαν στην υλοποίηση του PROMASI.

4 Σχεδίαση

4.1 Η σχεδίαση της πρώτης έκδοσης

Το κυρίως πρόγραμμα αποτελείται από τέσσερα επιμέρους στρώματα “GUI, Shell, Model, Core. Οι σχέσεις ανάμεσα σ' αυτά τα τέσσερα επιμέρους στρώματα περιγράφεται από το παρακάτω σχεδιάγραμμα.



Το κάθε επιμέρους στρώμα του PROMASI είναι υπεύθυνο για ένα σύνολο από λειτουργίες που έχουν ένα κοινό σκοπό. Για παράδειγμα το στρώμα Core είναι υπεύθυνο για την υλοποίηση ενός συστήματος δυναμικής “System Dynamics” που δίνει την δυνατότητα προσομοίωσης διαφόρων κλειστών συστημάτων που συναντάμε στον πραγματικό κόσμο.

4.1.1 Gui Package

Graphical User Interface περιέχει όλες τις απαραίτητες κλάσεις για να παρέχει στον χρήστη ένα γραφικό περιβάλλον ελέγχου του παιχνιδιού. Το GUI χρησιμοποιεί το Shell και το Model για τον χειρισμό του Core. Όλο του γραφικό περιβάλλον της παλιάς έκδοσης του PROMASI είναι γραμμένη με την χρήση της βιβλιοθήκης Swing. Στην νέα έκδοση του PROMASI θα χρησιμοποιηθεί η βιβλιοθήκη SWT.

4.1.2 Shell Package

Παρέχει την διασύνδεση ανάμεσα στα GUI, Model και Core, περιέχει όλη την λογική του PROMASI. Το Shell σ' αυτήν την περίπτωση αναπαριστά την υλοποίηση του Controller στο μοτίβο σχεδίασης MVC “Model-View-Controller”.

4.1.3 PROMASI Core Package

Υλοποιεί μια μηχανή προσομοίωσης βασισμένη στα συστήματα δυναμικής “System Dynamics”, παρέχει όλες τις απαραίτητες κλάσεις για την υλοποίηση ενός κλειστού συστήματος προσομοίωσης. Το Core επικοινωνεί με το Shell μέσω ενός μηχανισμού που ονομάζεται Communicator.

4.1.4 Model Package

Περιέχει κλάσεις οντοτήτων που περιγράφουν την επιχείρηση. Κάποιες από αυτές είναι Employee, Company, Project, Accountant. Οι κλάσεις αυτές σχετίζονται μεταξύ τους με διάφορους τρόπους. Για παράδειγμα η κλάση Company περιέχει μια λίστα από τα αντικείμενα τύπου Employee και ένα τουλάχιστον αντικείμενο τύπου Accountant.

4.2 Αλλαγή αρχιτεκτονικής.

Η αρχιτεκτονική στην οποία βασιζόταν η παλιά έκδοση του PROMASI είναι κατάλληλη για την υλοποίηση των enterprise εφαρμογών. Τέτοιου είδους εφαρμογές προορίζονται κυρίως για επιχειρήσεις που ασχολούνται με την διαχείριση μιας βάσης δεδομένων όπου αποθηκεύονται τα στοιχεία των πελατών ή και τα προϊόντα τους. Ο στόχος αυτών των εφαρμογών είναι να διευκολύνει την πρόσβαση και την αποθήκευση των δεδομένων απαραίτητων για την λειτουργία μιας τέτοιας επιχείρησης. Η βασική δομή τέτοιου είδους εφαρμογής περιγράφεται από τα τρία βασικά επίπεδα.

1. **Επίπεδο πρόσβασης στο μέσο αποθήκευσης.** Αυτό το επίπεδο είναι υπεύθυνο για την επικοινωνία με το μέσο αποθήκευσης των δεδομένων. Το μέσο αποθήκευσης μπορεί να είναι είτε ένα αρχείο κειμένου είτε μια βάση δεδομένων η κάτι άλλο. Συνήθως σε εφαρμογές που χρησιμοποιούν αυτού του είδους αρχιτεκτονική το στο επίπεδο πρόσβασης στο μέσω αποθήκευσης μπορούμε να διαπιστώσουμε την υλοποίηση του μοτίβου σχεδίασης DAO “Data Access Object”.

2. **Επίπεδο του μοντέλου.** Σ' αυτό το επίπεδο υπάρχει η περιγραφή των εγγραφών της βάσης υπό μορφή κάποιων οντοτήτων για παράδειγμα κλάσεων. Κάθε κλάση συνήθως αναπαριστά μια εγγραφή σε κάποιων πίνακα της βάσης δεδομένων.

3. **Επίπεδο παρουσίασης**. Αυτό το επίπεδο είναι υπεύθυνο για την παρουσίαση των δεδομένων στον χρήστη. Σ' αυτό το επίπεδο συνήθως γίνεται χρήση κάποιας βιβλιοθήκης προορισμένης στην ανάπτυξη του γραφικού περιβάλλοντος.

Το PROMASI όμως δεν ανήκει στην κατηγορία των enterprise εφαρμογών αλλά είναι ένας προσομοιωτής. Για αυτό το λόγο αποφασίσαμε ότι πρέπει να αλλάξουμε την αρχιτεκτονική του έργου με σκοπό να μειώσουμε την πολυπλοκότητα του κώδικα, της λογικής υλοποίησης, και να αφαιρέσουμε τις πολύπλοκες εξαρτήσεις ανάμεσα στα επιμέρους κομμάτια του έργου. Η απλοποίηση αυτή στοχεύει στην ευκολότερη συντήρηση και ανάπτυξη της εφαρμογής. Εκτός από την ευκολότερη συντήρηση και παρακολούθηση του κώδικα η αρχιτεκτονική αυτή δίνει την δυνατότητα στην επαναχρησιμοποίηση των

βιβλιοθηκών του έργου. Για παράδειγμα αν σε κάποιο άλλο έργο που θα βασίζεται στην τεχνολογία Java χρειαστεί να υλοποιήσουμε ένα σύστημα βασισμένο στα συστήματα δυναμικής δεν θα χρειαστεί να αναπτύξουμε την βιβλιοθήκη που υλοποιεί την συγκεκριμένη λειτουργία από την αρχή αλλά μπορούμε να πάρουμε την ήδη έτοιμη βιβλιοθήκη από το έργο του PROMASI.

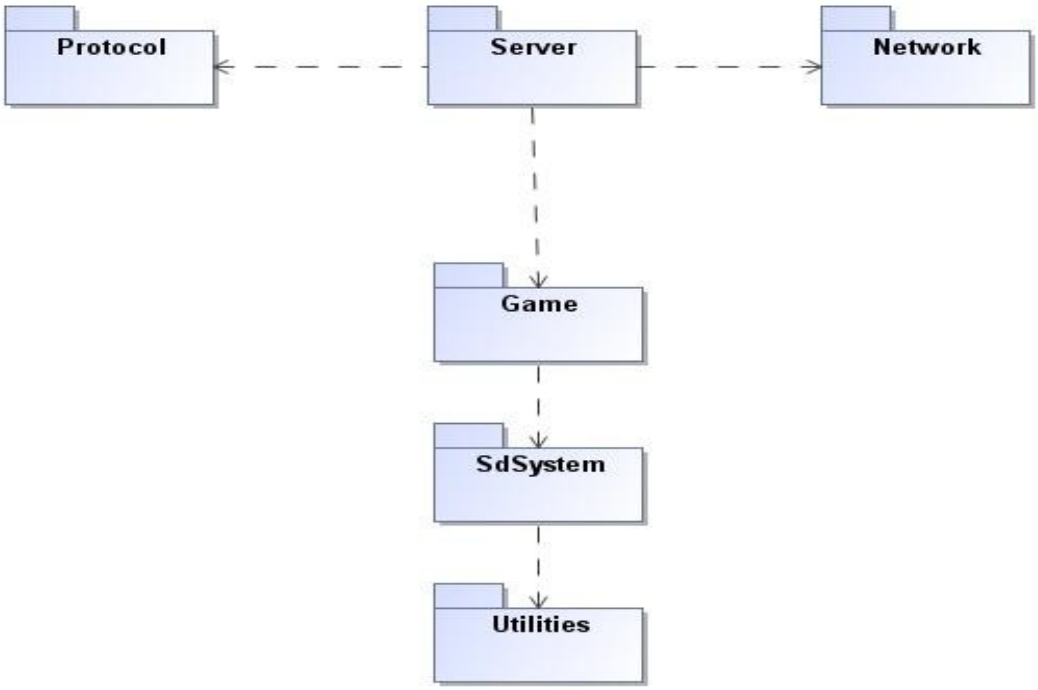
Το σχέδιο υλοποίησης του PROMASI διαιρεί το Project σε πέντε βιβλιοθήκες και δύο εφαρμογές. Κάθε βιβλιοθήκη παρέχει την συγκεκριμένη λειτουργία ταυτόχρονα διατηρώντας την υψηλού βαθμού γενίκευση (abstraction) με σκοπό να μειωθούν όσο το δυνατόν περισσότερο οι εξαρτήσεις ανάμεσα στις βιβλιοθήκες. Και οι δύο εφαρμογές χρησιμοποιούν κάποιες από τις βιβλιοθήκες για την υλοποίηση της εσωτερικής λειτουργίας τους. Πρέπει να σημειωθεί ότι και οι δύο εφαρμογές πρέπει να είναι της ίδιας έκδοσης μεταξύ τους και με τις βιβλιοθήκες που χρησιμοποιούν. Ο βασικότερος λόγος που επιβάλλεται η ίδια έκδοση σε όλο το έργο είναι ότι το PROMASI είναι πλέον μια εφαρμογή που στηρίζεται σε ένα πρωτόκολλο επικοινωνίας το οποίο πρέπει να διατηρείται ίδιο για τον client και τον server.

4.3 Βιβλιοθήκες

Στη δεύτερη έκδοση του PROMASI υπάρχουν πέντε βιβλιοθήκες. Κάθε βιβλιοθήκη του PROMASI είναι ένα ξεχωριστό έργο και υλοποιεί την προσχεδιασμένη λογική. Τα επιμέρους κομμάτια του συνολικού έργου του PROMASI έχουν όσο το δυνατόν χαμηλότερη εξάρτηση το ένα από το άλλο. Για παράδειγμα η εφαρμογή του Client καλό είναι να μην γνωρίζει απολύτως τίποτα για το σύστημα προσομοίωσης “SdSystem” στην περίπτωσή μας γιατί στο μέλλον το σημερινό σύστημα προσομοίωσης μπορεί να αντικατασταθεί με κάποιο άλλο που ίσος να πετυχαίνει καλύτερα αποτελέσματα σε σχέση με το σημερινό. Οι σχέσεις ανάμεσα στις βιβλιοθήκες του PROMASI σήμερα μπορούν να απεικονιστούν με τα παρακάτω διαγράμματα.

4.3.1 Σχέσεις ανάμεσα στις βιβλιοθήκες της εφαρμογής Server.

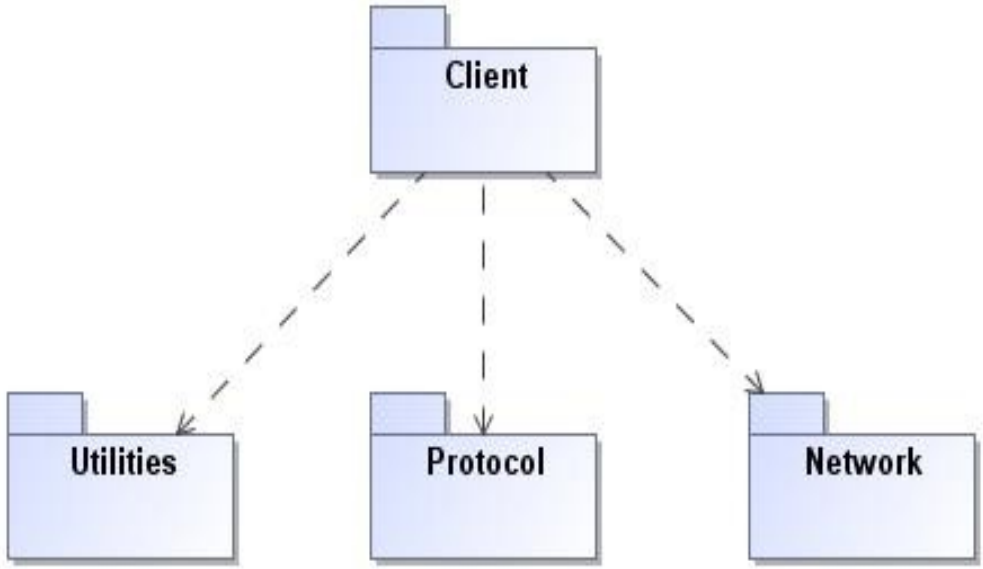
Ο server ως εφαρμογή χρησιμοποιεί και τις πέντε βιβλιοθήκες του PROMASI. Οι βιβλιοθήκες Protocol και Network είναι απαραίτητες για την επικοινωνία με τον client. Επειδή το παιχνίδι στο MultiPlayer mode πρέπει να είναι συγχρονισμένο ανάμεσα σε όλους τους clients του παιχνιδιού η υλοποίηση της προσομοίωσης πρέπει να ενσωματωθεί στον server άρα ο server πρέπει να χρησιμοποιεί την βιβλιοθήκη Game. Η βιβλιοθήκη Game με την σειρά της χρησιμοποιεί την SdSystem και Utilities.



4.3.2

4.3.3 Σχέσεις ανάμεσα στις βιβλιοθήκες της εφαρμογής Client.

Ο client από την μεριά του στο MultiPlayer mode χρησιμοποιεί μόνο τις τρεις από τις πέντε διαθέσιμες βιβλιοθήκες. Οι βιβλιοθήκες Protocol και Network χρησιμοποιούνται για την επικοινωνία με τον server και η βιβλιοθήκη Utilities ως μια βοηθητική βιβλιοθήκη. Ο client δεν χρειάζεται να γνωρίζει τίποτα για τον μηχανισμό προσομοίωσης και είναι υπεύθυνος μόνο για την παρουσίαση του τελικού αποτελέσματος της λειτουργίας του server.



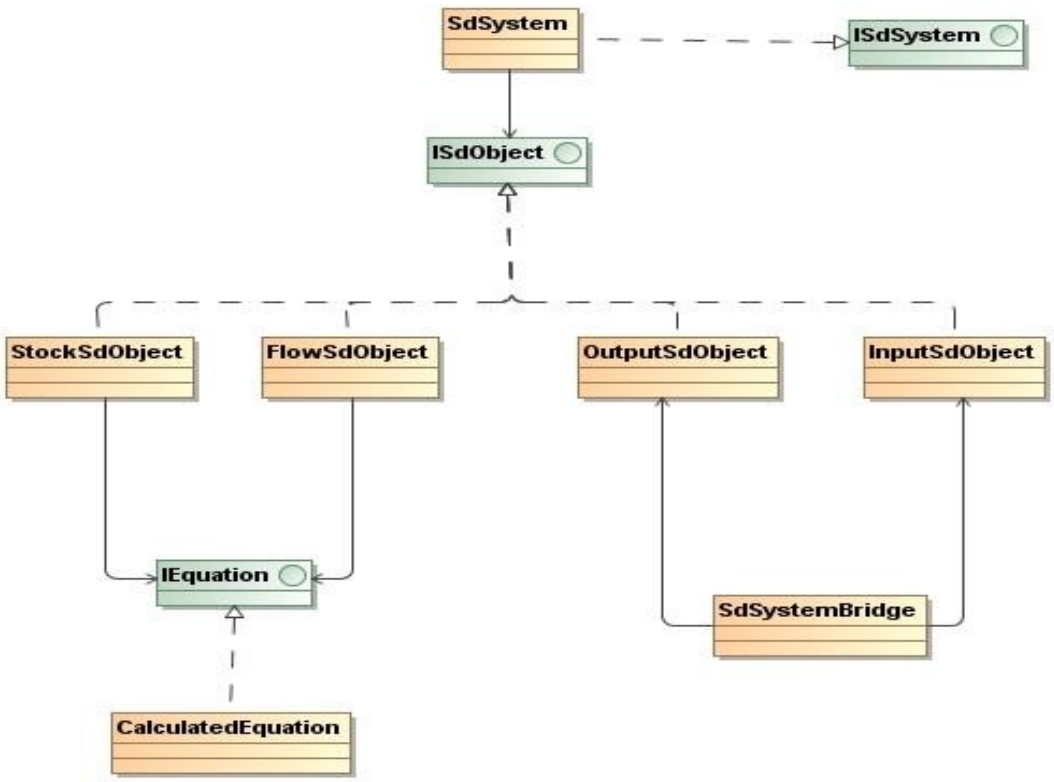
4.3.4 Utilities

Η βιβλιοθήκη αυτή δεν έχει εξάρτηση από καμία άλλη και παρέχει τις βοηθητικές κλάσεις που χρειάζονται στην υλοποίηση του έργου. Κάποιες από αυτές είναι:

- **Clock** περιγράφει το ρολόι του συστήματος. Το ρολόι είναι μία από τις βασικότερες οντότητες του έργου εφόσον ο χρόνος είναι ένα από τις εισόδους του συστήματος.
- **RootDirectory** περιγράφει τον αρχικό φάκελο με δεδομένα απαραίτητα για την λειτουργία του PROMASI. Ο φάκελος αυτός περιέχει τις ρυθμίσεις και τις ιστορίες του PROMASI.
- **SerializableObject** περιγράφει ένα Object το οποίο έχει την μέθοδο serialize που μετατρέπει το συγκεκριμένο object σε ένα XML string. Η κλάση αυτή χρησιμοποιείται σαν γωνική για τις κλάσεις που μετατρέπονται σε XML string.

4.3.5 SdSystem

Υλοποιεί την βιβλιοθήκη που μας επιτρέπει να δημιουργούμε κλειστά συστήματα προσομοίωσης βασισμένα στα συστήματα δυναμικής “System Dynamics”. Η βιβλιοθήκη αυτή περιέχει κλάσεις που απεικονίζουν ένα συστήματα δυναμικής “System Dynamics”. Με τον όρο απεικόνιση στον αντικειμενοστραφή προγραμματισμό εννοούμε την περιγραφή ενός συστήματος με βάση τα δομικά συστατικά του. Για παράδειγμα ένα σύστημα δυναμικής έτσι όπως μας το περιγράφουν τα βιβλία παρουσιάζεται σαν ένα σύνολο από κάποια αντικείμενα, τα οποία στην συγκεκριμένη περίπτωση ονομάζονται SdObjects, που το καθένα από αυτά εκτελεί ένα ξεχωριστό ρόλο και εκτελεί συγκεκριμένη διεργασία. Όλα τα βασικά αντικείμενα σε ένα σύστημα δυναμικής διασυνδέονται και επικοινωνούν μεταξύ τους. Στο σύνολό τους τα βασικά συστατικά αυτά απεικονίζουν ένα μοντέλο προσομοίωσης. Στο πιο κάτω διάγραμμα παρουσιάζεται η δομή της βιβλιοθήκης σε μορφή UML. Από το διάγραμμα φαίνεται ότι υπάρχουν συνολικά τέσσερα βασικά συστατικά ενός μοντέλου προσομοίωσης.

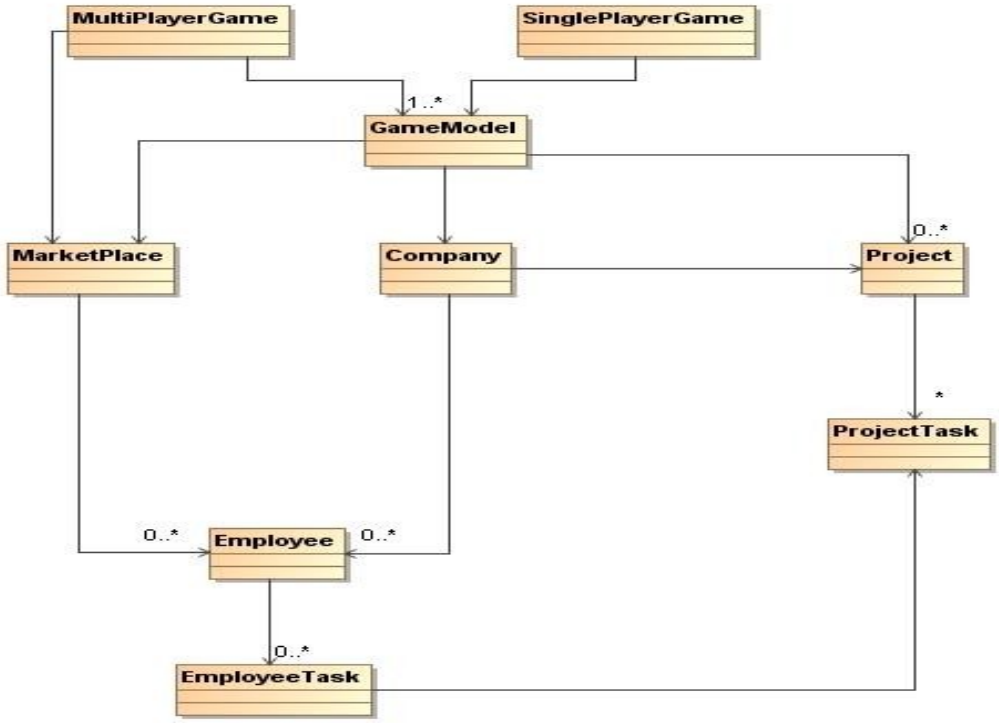


1. **StockSdObject**
2. **FlowSdObject**
3. **InputSdObject**
4. **OutputSdObject**

Δυο από αυτά τα συστατικά συγκεκριμένα το StockSdObject και FlowSdObject πρέπει να περιέχουν μια συνάρτηση υπολογισμού τις τελικής εξόδου. Τα άλλα δυο συστατικά InputSdObject και OutputSdObject χρησιμοποιούνται για την διασύνδεση των μοντέλων προσομοίωσης μεταξύ τους. Για την διασύνδεση των μοντέλων χρησιμοποιείται η κλάση SdSystemBridge η οποία λειτουργεί σαν γέφυρα ανάμεσα σε δυο μοντέλα.

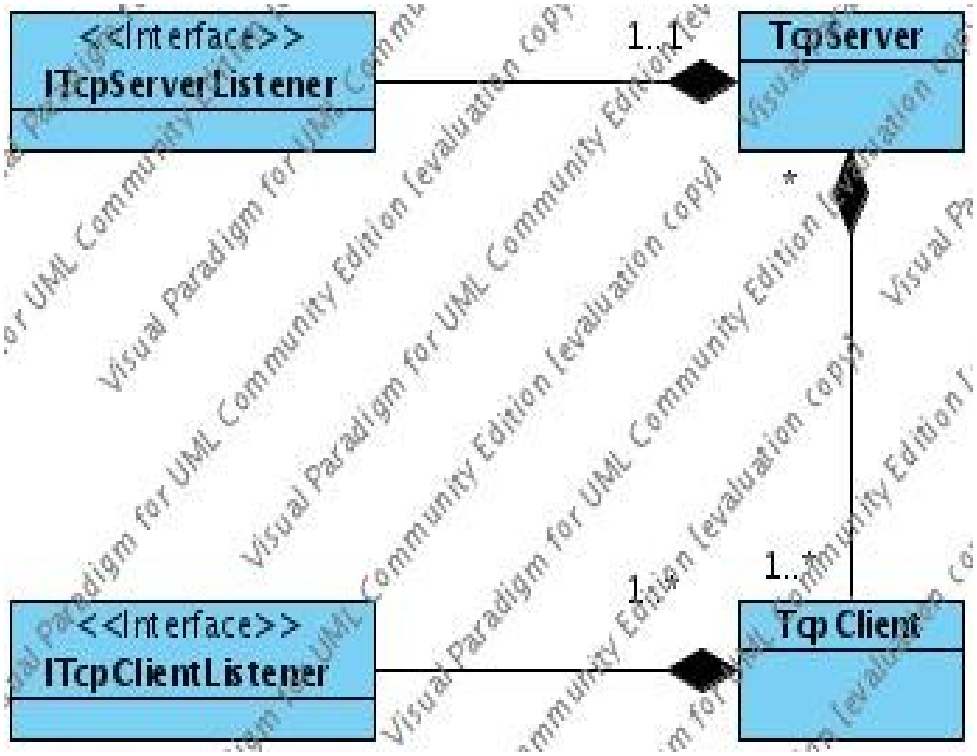
4.3.6 Game

Παρέχει την υλοποίηση της προσομοίωσης μίας επιχείρησης, το project αυτό έχει την εξάρτηση από τα SdSystem και Utilites.. Γενικός υπάρχουν δύο κατηγορίες παιχνιδιών MultiPlayer και SinglePlayer game. Το SinglePlayer game περιέχει ένα μοντέλο προσομοίωσης “GameModel” το οποίο με την σειρά του περιλαμβάνει μια σειρά από Projects, την κλάση που περιγράφει την επιχείρηση “Company” και την αγορά εργασίας “MarketPlace”. Κάθε επιχείρηση έχει τους υπαλλήλους της άρα υπάρχει η σχέση περιεκτικότητας 1..* ανάμεσα στην κλάση “Company” και την κλάση Employee. Η αγορά εργασίας επίσης περιλαμβάνει τους ελεύθερους “υπαλλήλους” άρα έχουμε και εδώ την ίδια σχέση περιεκτικότητας 1...*.Κάθε επιχείρηση μπορεί να υλοποιεί το πολύ ένα Project την κάθε χρονική στιγμή. Το Project περιγράφεται από κάποιες διεργασίες “Tasks”. Ένας υπάλληλος μπορεί να εκτελεί μια διεργασία του project την κάθε χρονική στιγμή. Ο υπάλληλος ώμος βλέπει τις υποχρεώσεις του σαν διεργασίες που απευθύνονται μόνο σ αυτόν και σε κάποια μεγάλα έργα ο υπάλληλος μπορεί να μην γνωρίζει την συνολική κατάσταση του έργου εκτελώντας μόνο τα “Tasks” που του αναθέτουν.



4.3.7 Network

Παρέχει της κλάσεις υπεύθυνες για την επικοινωνία μέσου δικτύου. Οι βασικές κλάσεις είναι TcpServer, TcpClient. Οι σχέσεις ανάμεσα σ αυτές τις κλάσεις περιγράφονται στο ακόλουθο UML διάγραμμα. Από το διάγραμμα φαίνεται ότι ο TcpServer μπορεί να περιέχει πολλούς TcpClients επίσης όσο ο TcpServer τόσο και ο TcpClient μπορούν να παράγουν events μέσω της υλοποίησης του ItcpServerListener και ItcpClientListener αντίστοιχα. Το σχεδιάγραμμα αυτό στην ουσία δείχνει την κλασσική υλοποίηση του μοτίβου σχεδίασης που ονομάζεται Strategy η Policy.



4.3.8 Protocol

Περιέχει όλες τις κλάσεις που αντιπροσωπεύουν τα μηνύματα στην επικοινωνία ανάμεσα στους Client-Server. Τα μηνύματα που μπορεί να ανταλλάξει το πρωτόκολλο περιγράφονται παρακάτω:

Μήνυμα	Περιγραφή
AssignEmployeeTasksRequest	Στέλνεται από τον client με σκοπό της ανάθεσης εργασιών στους εργαζομένους της εταιρίας. Το μήνυμα αυτό περιέχει τις διεργασίες σε μορφή XML.
CreateGameFailedResponse	Στέλνεται από τον server με σκοπό την ενημέρωση του client για την αποτυχία δημιουργίας νέου παιχνιδιού.
CancelGameRequest	Στέλνεται από τον client ο οποίος βρίσκεται στην κατάσταση αναμονής παικτών του παιχνιδιού **WaitingPlayersClientState**. Το μήνυμα αυτό ενημερώνει τον server για την ακύρωση του παιχνιδιού. Ο server με την σειρά του ενημερώνει όλους τους παίκτες του παιχνιδιού οτι το παιχνίδι ακυρώνεται.
CancelGameResponse	Το μήνυμα αυτό περιγράφει την απάντηση του server στο μήνυμα του client για την ακύρωση του παιχνιδιού.
CreateGameRequest	Αυτό το μήνυμα μπορεί να σταλεί από τον client όταν βρείσκεται στην κατάσταση δημιουργίας παιχνιδιού. Με την παραλαβή τέτοιου είδους μηνύματος ο server προσπαθεί να δημιουργήσει ένα νέο παιχνίδι. Στην περίπτωση επιτυχημένης προσπάθειας ο server απαντάει με το μήνυμα **CreateGameResponse.** Στην περίπτωση αποτυχίας ο server στέλνει

Μήνυμα	Περιγραφή
	CreateGameFailedResponse.
CreateGameResponse	Στέλνεται από τον server στην περίπτωση επιτυχημένης προσπάθειας δημιουργίας νέου παιχνιδιού.
DischargeEmployeeRequest	Στέλνεται από τον client στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**. Το μήνυμα αυτό απεικονίζει το τέλος της συνεργασίας με κάποιων επαγγελματία.
EmployeeDischargedRequest	Στέλνεται από τον server με σκοπό την ενημέρωση των clients για τις αλλαγές στην αγορά εργασίας κατά την διάρκεια του παιχνιδιού.
EmployeeHiredRequest	Στέλνεται από τον server με σκοπό την ενημέρωση των clients για τις αλλαγές στην αγορά εργασίας κατά την διάρκεια του παιχνιδιού. Οι clients βρίσκονται στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**.
GameCanceledRequest	Στέλνεται από τον server με σκοπό την ενημέρωση των clients για την ακύρωση του παιχνιδιού που έχουν επιλέξει. Οι clients βρίσκονται στην κατάσταση αναμονής παιχνιδιού **WaitingGameClientState**.
GameFinishedRequest	Μήνυμα που ενημερώνει όλους τους clients για την λήξη του παιχνιδιού όταν βρίσκονται στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**.
GameStartedRequest	Με αυτό το μήνυμα ο server ενημερώνει όλους του παίκτες του ίδιου παιχνιδιού για την εκκίνηση του παιχνιδιού. Όλοι οι παίκτες πρέπει

Μήνυμα	Περιγραφή
	να βρίσκονται στην κατάσταση αναμονής του παιχνιδιού **WaitingGameClientState**.
GameStartedResponse	Αυτό το μήνυμα προέρχεται από τον server και ενημερώνει τον δημιουργό του παιχνιδιού για την εκκίνησή του. Ο δημιουργός του παιχνιδιού βρίσκεται στην κατάσταση αναμονής των παικτών **WaitingPlayersClientState**.
HireEmployeeRequest	Ο client μπορεί να στείλει τέτοιου είδους μήνυμα όταν θέλει να προσλάβει κάποιον ελεύθερο επαγγελματία από την αγορά εργασίας. Ο client βρίσκεται στην κατάσταση αναμονής των παικτών **WaitingPlayersClientState**.
InternalErrorResponse	Στέλνεται στην περίπτωση κάποιου απρόβλεπτου λάθους είτε στον server είτε στον client.
JoinGameFailedResponse	Στην περίπτωση που ο server δεν μπόρεσε να εισάγει τον client που έστειλε το μήνυμα εισαγωγής στο παιχνίδι δημιουργείται ένα τέτοιο μήνυμα και ενημερώνει τον client για την αποτυχία εισαγωγής του στο παιχνίδι. Ο client βρίσκεται στην κατάσταση επιλογής παιχνιδιού **ChooseGameClientState**.
JoinGameRequest	Το μήνυμα αυτό ενημερώνει τον server για την επιλογή παιχνιδιού που έκανε ο client. Ο client βρίσκεται στην κατάσταση επιλογής παιχνιδιού **ChooseGameClientState**.
JoinGameResponse	Στέλνεται από τον server ο οποίος ενημερώνει τον client για την ολοκλήρωση εισαγωγής του στο παιχνίδι. Ο client βρίσκεται στην κατάσταση επιλογής παιχνιδιού **ChooseGameClientState**.

Μήνυμα	Περιγραφή
LoginFailedResponse	Στέλνεται από τον server για να ενημερώσει τον client για την αποτυχία εισαγωγής του στο σύστημα. Ο client βρίσκεται στην κατάσταση εισαγωγής στο σύστημα **LoginClientState**.
LoginRequest	Στέλνεται από τον client στην προσπάθειά του για την εισαγωγή στο σύστημα. Ο client βρίσκεται στην κατάσταση εισαγωγής στο σύστημα **LoginClientState**.
LoginResponse	Είναι μήνυμα ενημέρωσης του client για την επιτυχημένη εισαγωγή στο σύστημα. Ο client βρίσκεται στην κατάσταση εισαγωγής στο σύστημα **LoginClientState**.
MessageRequest	Χρησιμοποιείται για ανταλλαγή μηνυμάτων μεταξύ τον clients στην κατάσταση αναμονής παιχνιδιού **WaitingGameClientState** και στην κατάσταση αναμονής παικτών **WaitingPlayersClientState**.
OnExecuteStepRequest	Στέλνεται από τον server με σκοπό την ενημέρωση του client για την εκτέλεση ενός βήματος του έργου στον server. Ο client βρίσκεται στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**.
OnTickRequest	Στέλνεται από τον server με σκοπό την ενημέρωση του client για την εσωτερική κατάσταση του ρολογιού του server. Το μήνυμα αυτό περιέχει την τρέχον εμηρομηνια και την ώρα του συστήματος. Τα δεδομένα αυτά είναι ίδια για όλους τους clients του ίδιου παιχνιδιού. Ο client βρίσκεται στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**.

Μήνυμα	Περιγραφή
ProjectAssignedRequest	Στέλνεται από τον server με σκοπό την ενημέρωση του client για την παραλαβή ενός νέου έργου. Το μήνυμα αυτό περιέχει πληροφορίες για τον νεο έργο καθώς και την περιγραφή της κατάστασης της εταιρίας. Ο client βρίσκεται στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**.
ProjectFinishedRequest	Στέλνεται από τον server με σκοπό την ενημέρωση του client για την ολοκλήρωση του τρέχον έργου. Ο client βρίσκεται στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**.
StartGameRequest	Στέλνεται από τον client για να ενημερώσει τον server για την εκκίνηση του παιχνιδιού. Ο client βρίσκεται στην κατάσταση δημιουργίας παιχνιδιού **MakeGameClientState**. Ο server είναι στην κατάσταση επιλογής παιχνιδιού **ChoseGameClientState**.
StartGameResponse	Είναι η απάντηση του server στο μήνυμα του client που αφορά την εκκίνηση του παιχνιδιού. Ο client βρίσκεται στην κατάσταση δημιουργίας παιχνιδιού **MakeGameClientState**. Ο server είναι στην κατάσταση επιλογής παιχνιδιού **ChoseGameClientState**.
UpdateAvailableGameListRequest	Στέλνεται από τον server και ενυμερώνει τους clients για τα παιχνίδια που είναι διαθέσιμα στον server. Ο client βρίσκεται στην κατάσταση επιλογής παιχνιδιού **ChooseGameClientState**.
UpdateGameListRequest	Στέλνεται από τον client για να πληροφορίσει τον server ότι είναι απαραίτητη η ενημέρωση της λίστας με τα παιχνίδια του.

Μήνυμα	Περιγραφή
UpdateGamePlayersListRequest	Στέλνεται από τον server για την ενημέρωση της λίστας παικτών στο επιλεγμένο παιχνίδι του client. Ο client βρίσκεται στην κατάσταση αναμονής παιχνιδιού **WaitingGameClientState** η αναμονής παικτών **WaitingPlayersClientState**.
WrongProtocolResponse	Μήνυμα το οποίο ενυμερώνει τον client η τον server για κάποιο πιθανό λάθος στο πρωτόκολλο επικοινωνίας.
LeaveGameRequest	Δημιουργείται από τον client για να ενημερώσει τον server για την έξοδο του από το παιχνίδι. Ο client βρίσκεται στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**.
LeaveGameResponse	Η απάντηση του server στο μήνυμα του client για την έξοδο από το παιχνίδι. Ο client βρείσκεται στην κατάσταση εξέλιξης του παιχνιδιού **PlayingGameClientState**.

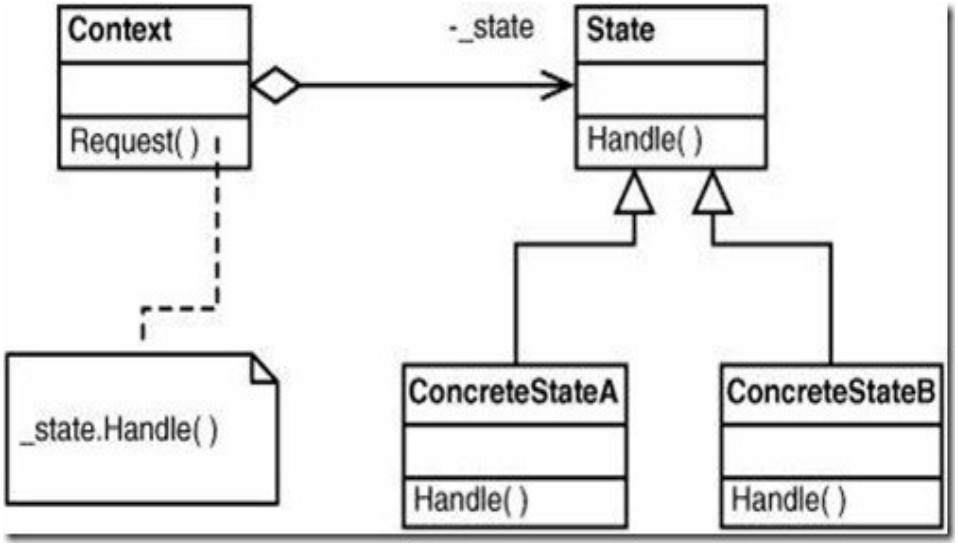
Κάθε κλάση του πρωτοκόλλου κληρονομεί από την SerializableObject που παρέχει την serialize() μέθοδο για την μετατροπή της σε XML κείμενο, όταν το μήνυμα μετατραπεί σε XML, κωδικοποιείται με τον αλγόριθμο **BASE64** για να σταλεί στο δίκτυο. Κάθε μήνυμα του PROMASI στην τελική του μορφή έτσι όπως ταξιδεύει στο δίκτυο είναι μια γραμμή του BASE64 string. Το κάθε μήνυμα δηλαδή χωρίζεται με το σύμβολο της αλλαγής γραμμής '\n'. Από την άλλη μεριά ο παραλήπτης διαβάζει γραμμή γραμμή κείμενο που λαμβάνει από το δίκτυο, αποκωδικοποιεί το BASE64 string σε string XML και στην συνέχεια μετατρέπει το XML string που έλαβε σε αντικείμενο του πρωτοκόλλου πρέπει να σημειωθεί ότι και ο client και ο server πρέπει να χρησιμοποιούν την ίδια έκδοση του πρωτοκόλλου με σκοπό να εξελίσσεται ομαλά η επικοινωνία ανάμεσά τους.

4.3.9 Επικοινωνία

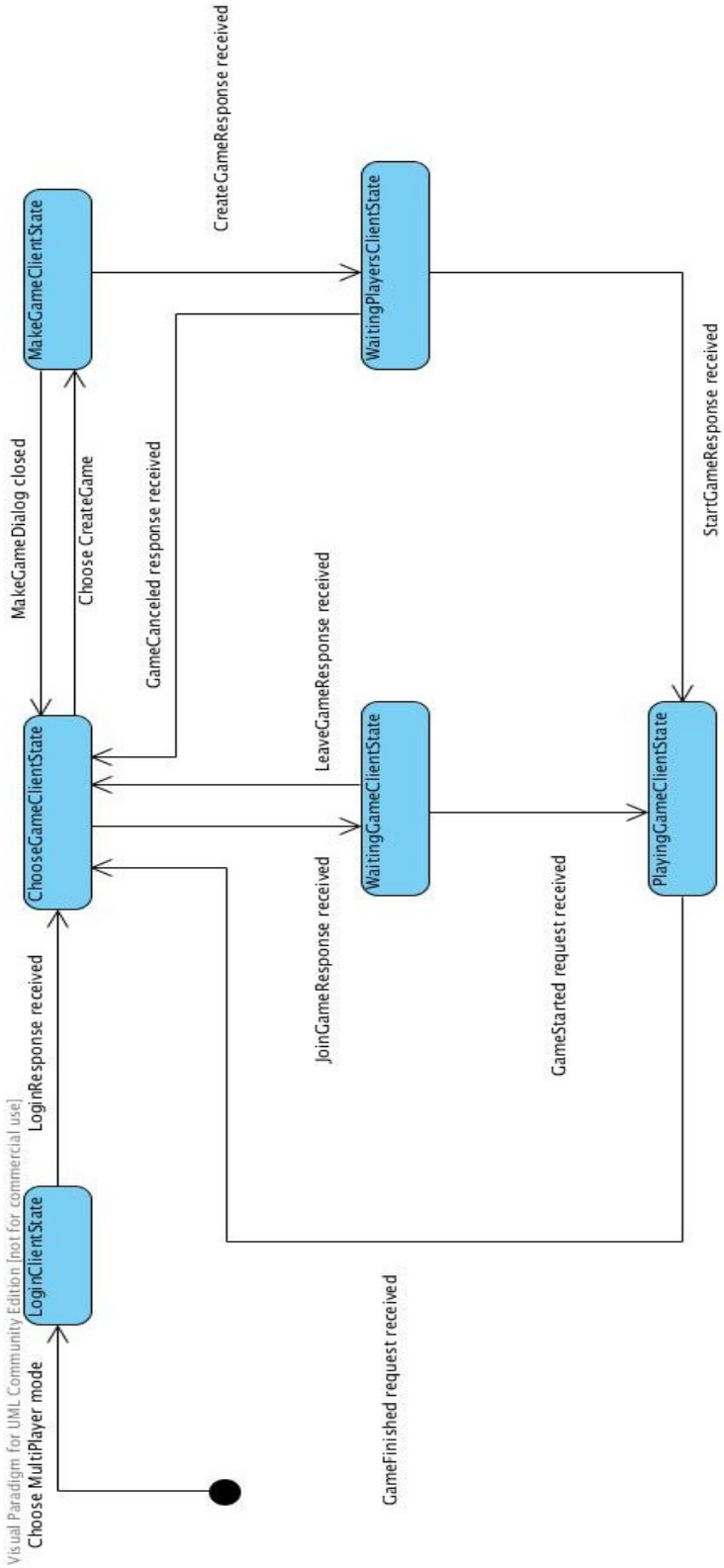
Κάθε Request στο PROMASI έχει ένα η περισσότερα Responses. Στην περίπτωση που το σύστημα δεν μπορεί να αναγνωρίσει το μήνυμα που έλαβε λόγο κάποιου λάθους στο πρωτόκολλο επικοινωνίας στέλνεται ένα μήνυμα τύπου “WrongProtocolResponse” και η επικοινωνία ανάμεσα στους Client-Server τερματίζεται.

Επειδή στην επικοινωνία χρησιμοποιείται σταθερή σύνδεση TCP το πρωτόκολλο του PROMASI είναι ασύγχρονο. Για τον συγχρονισμό των μηνυμάτων του PROMASI υλοποιήθηκαν δύο μηχανές καταστάσεων μία από την μεριά του Client και μία ακόμα από την μεριά του Server. Και οι δύο μηχανές έχουν κάποιες διακριτές καταστάσεις στις οποίες επιτρέπεται η αποστολή και αποδοχή μόνο προκαθορισμένου είδους μηνυμάτων. Η αλλαγή των καταστάσεων εξαρτάται από τα μηνύματα που έχουν λάβει ο Client και ο Server κατά την επικοινωνία τους καθώς και από τις ενέργειες του χρήστη στο γραφικό περιβάλλον.

Όλες οι μηχανές καταστάσεων του ProMaSi υλοποιούνται με την εφαρμογή του State design pattern (Gamma et al, 1994) το οποίο περιγράφεται στο παρακάτω διάγραμμα. Το Context στο ProMaSi είναι η κλάση **ProMaSiClient** το State είναι η κλάση **AbstractClientState**. Τα States του ProMaSi περιγράφονται παρακάτω.


Context
Request( )
-_state
State
Handle( )
_state.Handle( )
ConcreteStateA
Handle( )
ConcreteStateB
Handle( )

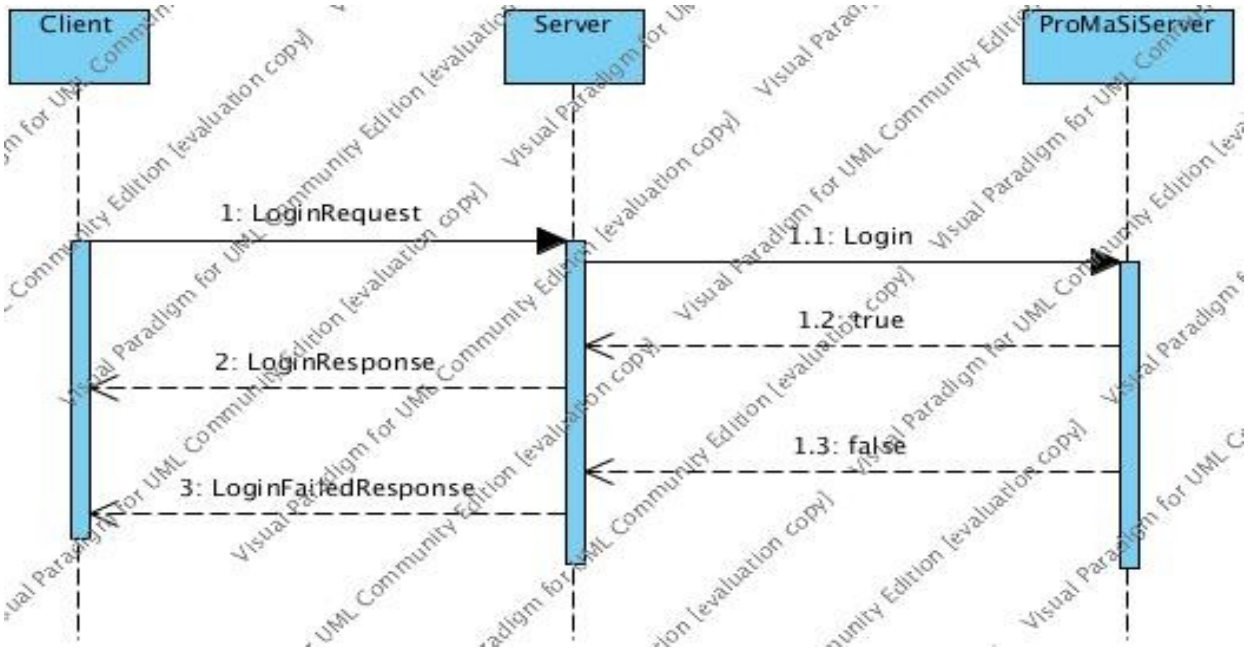
4.3.10 Καταστάσεις του Client.



LoginClientState

Κατάσταση στην οποία το server περιμένει τον client για την αποστολή συνθηματικών για την εισαγωγή στο σύστημα. Ο client πρέπει να στείλει το μήνυμα που αντιπροσωπεύει το LoginRequest με σκοπό να περάσει στην επόμενη κατάσταση.

Αποστολή	**Παραλαβή**
LoginRequest	LoginResponse LoginFailedResponse



ChooseGameClientState

Στην κατάσταση αυτή ο client βρίσκεται στην κατάσταση επιλογής, υπάρχουν συνολικά δυο επιλογές που μπορεί να κάνει ο client. Οι καταστάσεις αυτές είναι Δημιουργία νέου παιχνιδιού και εισαγωγή στο ήδη υπάρχων παιχνίδι. Κατά συνέπεια ο client μπορεί να στείλει δυο ειδών μηνύματα JoinGameRequest και CreateGameRequest ανάλογα με τις επιλογές του χρήστη.

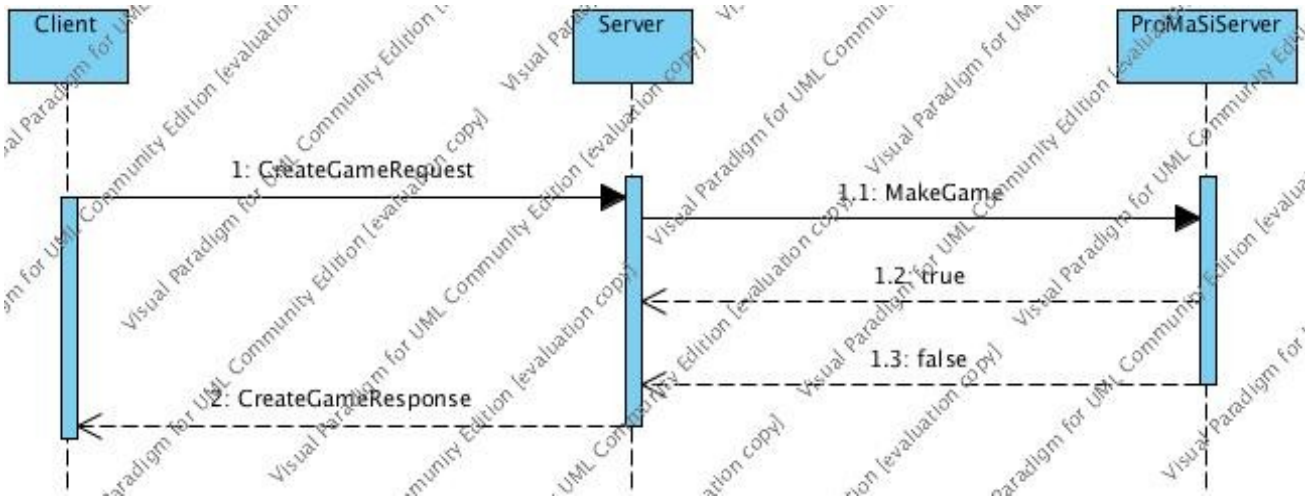
Αποστολή	**Παραλαβή**
JoinGameRequest	JoinGameResponse JoinGameFailedResponse

MakeGameClientState

Στην κατάσταση δημιουργίας παιχνιδιού ο client μπορεί να δημιουργήσει καινούριο παιχνίδι στέλνοντας κατάλληλο μήνυμα “MakeGameRequest” στον server.

Αποστολή	**Παραλαβή**
CreateGameRequest	CreateGameResponse CreateGameFailedReponse

<u>Sequence diagram.</u>



WaitingPlayersClientState

Στην περίπτωση που ο client έχει επιλέξει την δημιουργία του καινούριου παιχνιδιού στην προηγούμενη κατάσταση τότε θα βρίσκεται στην κατάσταση αναμονής των παικτών του

παιχνιδιού. Στην κατάσταση αυτή ο client έχει την δυνατότητα της συνομιλίας με τους υπόλοιπους παίκτες που θέλουν να πάρουν μέρος στο παιχνίδι που δημιούργησε. Επίσης στην κατάσταση αυτή ο δημιουργός του παιχνιδιού μπορεί να στείλει μήνυμα εκκίνησης του παιχνιδιού StartGameRequest για να ξεκινήσει το παιχνίδι.

Αποστολή	**Παραλαβή**
StartGameRequest MessageRequest	GameStartedRequest UpdateGamePlayersListRequest MessageRequest

WaitingGameClientState

Η κατάσταση αυτή περιγράφει την αναμονή του χρήστη για την εκκίνηση του παιχνιδιού στο οποίο έχει εισέλθει. Σ' αυτήν την κατάσταση ο χρήστης έχει την δυνατότητα επικοινωνίας με τους υπόλοιπους παίκτες του παιχνιδιού.

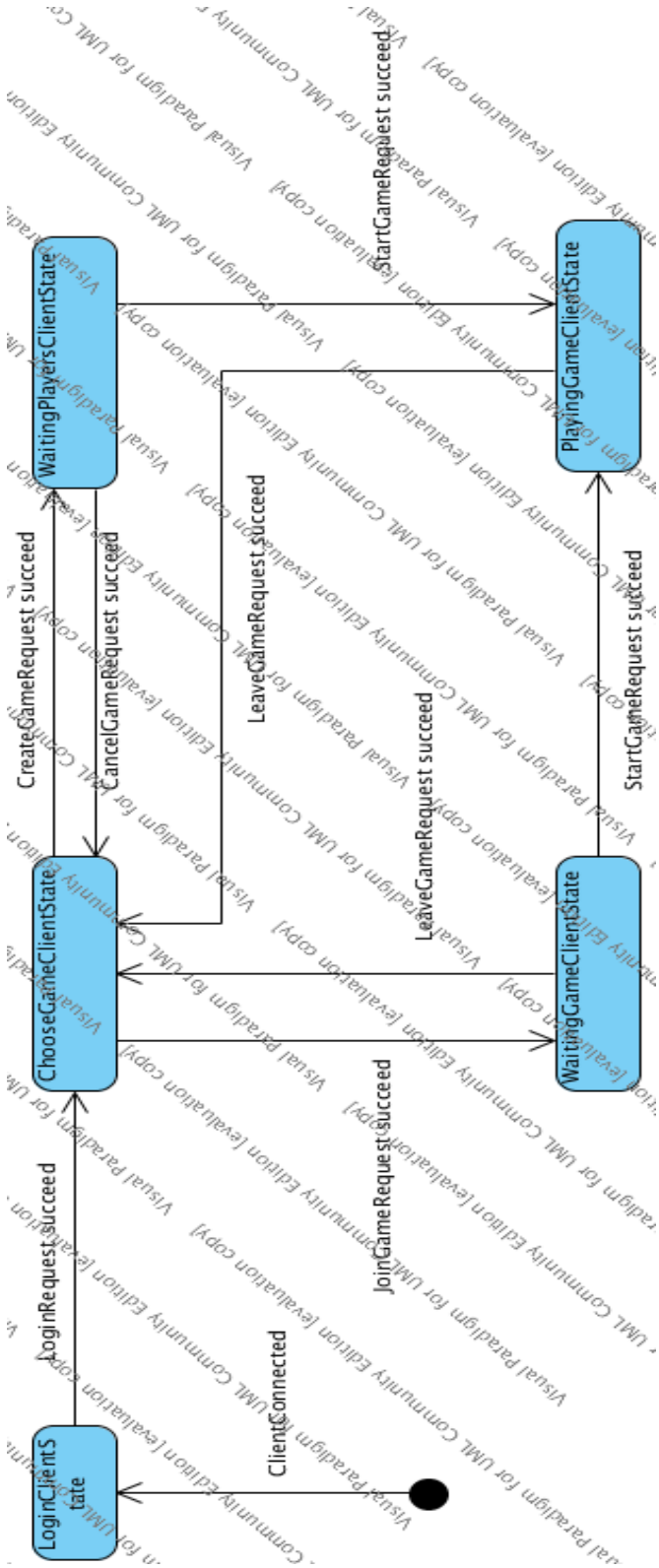
Αποστολή	**Παραλαβή**
StartGameRequest MessageRequest	GameStartedRequest UpdateGamePlayersListRequest MessageRequest GameCanceledRequest

PlayingGameClientState

Αυτή η κατάσταση περιγράφει την εξέλιξη του παιχνιδιού. Όταν ο client βρίσκεται στην κατάσταση αυτή έχει δικαίωμα να στείλει τα μηνύματα ελέγχου του παιχνιδιού.

Αποστολή	**Παραλαβή**
HireEmployeeRequest DischargeEmployeeRequest AssignEmployeeTasksRequest	EmployeeHiredRequest ProjectAssignedRequest ProjectFinishedRequest GameFinishedRequest

4.3.11 Καταστάσεις του Server



Το παραπάνω διάγραμμα περιγράφει τις καταστάσεις του client συνδεδεμένο στον server του ProMaSi. Στην πρώτη επαφή του με τον server “TCP Connection” ο client μπαίνει στην κατάσταση LoginClientState. Το server περιμένει να λάβει το LogiRequest από τον client. Μόλις ο server λάβει το LoginRequest και στην περίπτωση που η διαδικασία Login πέρασε ομαλά ο client περνάει στην επόμενη κατάσταση ChooseGameClientState. Στην κατάσταση αυτή ο client έχει συνολικά δυο επιλογές.

1. Δημιουργία νέου παιχνιδιού
2. Επιλογή ήδη υπάρχον παιχνιδιού.

Το server περιμένει δυο ειδών μηνύματα για την αλλαγή της κατάστασης του client.

Στην περίπτωση που το server δεχτεί μήνυμα τύπου CreateGameRequest και η διαδικασία της δημιουργίας παιχνιδιού ολοκληρωθεί πλήρως τότε ο client περνάει στην κατάσταση αναμονής των παικτών WaitingPlayersClientState. Ο client που δημιούργησε το παιχνίδι γίνεται αυτόματα GameMaster και μπορεί να ακυρώσει το παιχνίδι οποιαδήποτε στιγμή. Στην περίπτωση που το server λάβει ένα μήνυμα τύπου JoinGameRequest τότε ο client αυτός παίρνει μέρος στο παιχνίδι και ο περνάει στην κατάσταση αναμονής παιχνιδιού

LoginClientState

Όταν ο server βρίσκεται σ' αυτήν την κατάσταση περιμένει από τον client ένα μήνυμα εισαγωγής στο σύστημα, αν για κάποιο λόγο λάβει κάποιο άλλο μήνυμα διαφορετικό από το LoginRequest τότε στέλνει στον client το μήνυμα λάθους και οι επικοινωνία μεταξύ του client και server τερματίζεται.

Αποστολή	**Παραλαβή**
LoginResponse LoginFailedResponse	LoginRequest

ChooseGameClientState

Σ' αυτήν την κατάσταση ο server περιμένει από τον client ένα μήνυμα επιλογής παιχνιδιού JoinGameRequest. Αν για κάποιο λόγο λάβει κάποιο άλλο μήνυμα διαφορετικό από το JoinGameRequest τότε στέλνει στον client το μήνυμα λάθους και οι επικοινωνία μεταξύ του client και server τερματίζεται.

Αποστολή	**Παραλαβή**
JoinGameResponse JoinGameFailedResponse	JoinGameRequest

WaitingPlayersClientState

Η κατάσταση αυτή περιγράφει την αναμονή του server για την εκκίνηση του παιχνιδιού από τον δημιουργό του. Ο client που που παίρνει αυτήν την κατάσταση είναι ο δημιουργός του παιχνιδιού και έχει το δικαίωμα να στείλει είτε το μήνυμα εκκίνησης του παιχνιδιού GameStartedRequest είτε το μήνυμα επικοινωνίας με τους άλλους χρήστες MessageRequest.

Αποστολή	**Παραλαβή**
GameStartedRequest UpdateGamePlayersListRequest MessageRequest CancelGameRequest	StartGameRequest MessageRequest CancelGameResponse

WaitingGameClientState

Η κατάσταση αυτή είναι αντίστοιχη της κατάστασης αναμονής παικτών του παιχνιδιού με διαφορά ότι σ αυτήν την κατάσταση μπαίνει ο client που έχει επιλέξει το παιχνίδι ενός άλλου δημιουργού.

Αποστολή	**Παραλαβή**
GameStartedRequest UpdateGamePlayersListRequest MessageRequest GameCanceledRequest	StartGameRequest MessageRequest

PlayingGameClientState

Αυτή η κατάσταση του server περιγράφει την εξέλιξη του παιχνιδιού. Ο server μπορεί να ενημερώνει τον κάθε παίκτη του παιχνιδιού για την εσωτερική κατάσταση του.

Αποστολή	**Παραλαβή**
EmployeeHiredRequest	HireEmployeeRequest
ProjectAssignedRequest	DischargeEmployeeRequest
ProjectFinishedRequest	AssignEmployeeTasksRequest

4.3.12 Πλεονεκτήματα της νέας αρχιτεκτονικής

Η νέα αρχιτεκτονική του Project δίνει νέες δυνατότητες στον τρόπο ανάπτυξης του και επίσης ποιο εύκολη παρακολούθηση και συντήρησης του κώδικα. Πλέον στο PROMASI μπορούν να δουλεύουν πολλοί προγραμματιστές ταυτόχρονα χωρίς να επηρεάζουν ο ένας των άλλων. Επίσης μπορούν να αναπτυχθούν νέες λειτουργίες χωρίς την γνώση για τις λεπτομέρειες της υλοποίησης των παλιών λειτουργιών.

4.3.13 Σύζευξη

Το PROMASI μπορεί να αναπτύσσεται σε ομάδα. Επειδή το κάθε project του PROMASI υλοποιεί μία συγκεκριμένη λειτουργία μπορεί να το αναλάβει ένας προγραμματιστής χωρίς να γνωρίζει απαραίτητα για τα υπόλοιπα projects. Μικρότερη σύζευξη είναι ένας από τους βασικότερους στόχους από την άποψη του σχεδίασης της εφαρμογής. Όσο μικρότερη είναι η σύζευξη μεταξύ των των κλάσεων τόσο ποιο απλός και κατανοητός είναι ο κώδικας. Και αν ένας κώδικας είναι απλός τότε έχουμε την μικρότερη πιθανότητα να δημιουργηθούν τα σφάλματα και μπορούμε να αφιερώσουμε περισσότερο χρόνο στην ανάπτυξη των υπολειτουργιών της εφαρμογής.

4.3.14 Πολλές υλοποιήσεις ταυτόχρονα – plugins.

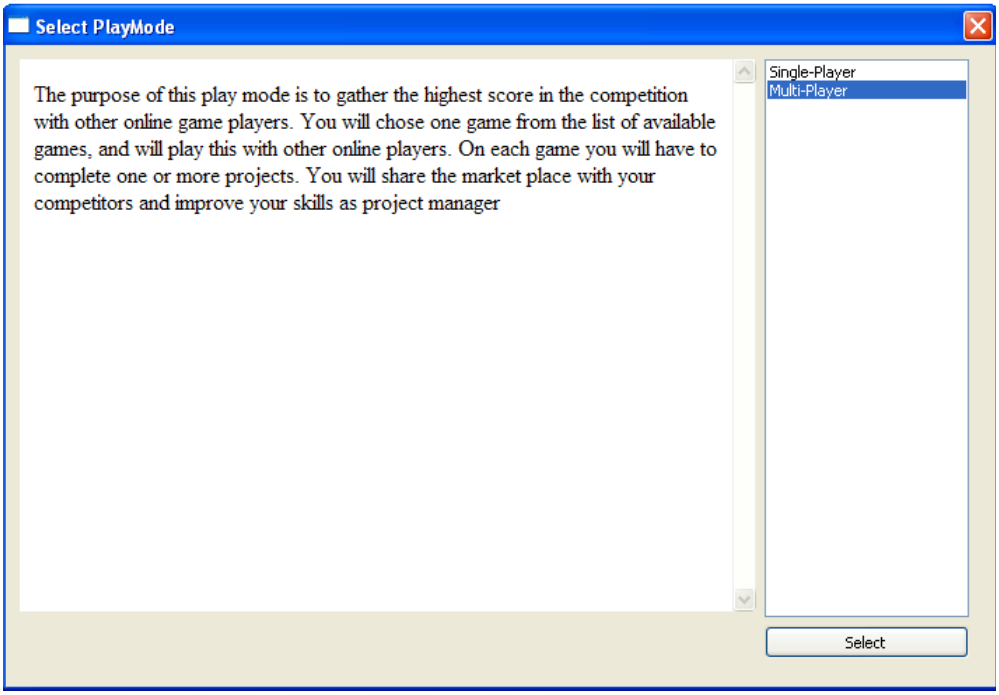
Η αρχιτεκτονική αυτή δίνει την δυνατότητα να δουλεύουν πολλοί προγραμματιστές ταυτόχρονα μειώνοντας το συνολικό χρόνο για την υλοποίηση του έργου και την πολυπλοκότητα του κώδικα. Αυτό επιτυγχάνεται με την σχεδίαση ενός interface για κάθε βιβλιοθήκη του PROMASI. Το interface αυτό δίνει μόνο την απαραίτητη πληροφορία στον προγραμματιστή που θα χρησιμοποιήσει την βιβλιοθήκη. Ο προγραμματιστής δεν πρέπει να γνωρίζει για την εσωτερική δομή της βιβλιοθήκης. Με αυτόν τον τρόπο ένας προγραμματιστής μπορεί να χρησιμοποιεί την βιβλιοθήκη μέσω ενός interface ενώ ένας άλλος προγραμματιστής μπορεί να συντηρεί και να εξελίσσει αυτήν την βιβλιοθήκη παράλληλα. Εκτός από αυτό το πλεονέκτημα η επικοινωνία μέσω ενός interface δίνει την δυνατότητα εξέλιξης κάποιου μέρους της εφαρμογής ανεξαρτήτως αν έχουν υλοποιηθεί τα υπόλοιπα κομμάτια. Ο προγραμματιστής μπορεί να χρησιμοποιήσει τα Mock objects για να προσομοιώνει την λειτουργία ενός κομματιού της εφαρμογής του ενώ γράφει ένα άλλο κομμάτι που αναγκαστικά χρειάζεται την ύπαρξη του πρώτου για να λειτουργήσει.

5 Τελική εφαρμογή

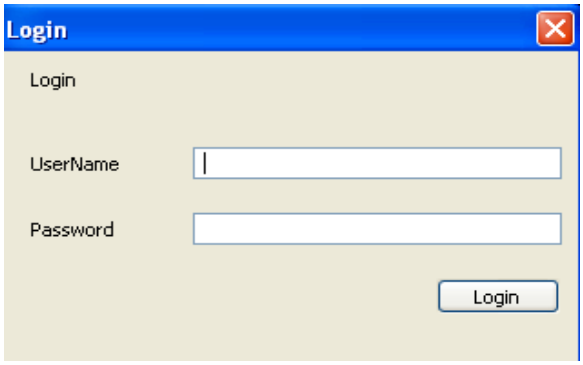
5.1 Διαδικασία εισαγωγής στο παιχνίδι.

Η νέα εφαρμογή του PROMASI ξεκινάει με το μενού επιλογής του PlayMode. Ο χρήστης έχει συνολικά δυο επιλογές σ' αυτό το στάδιο

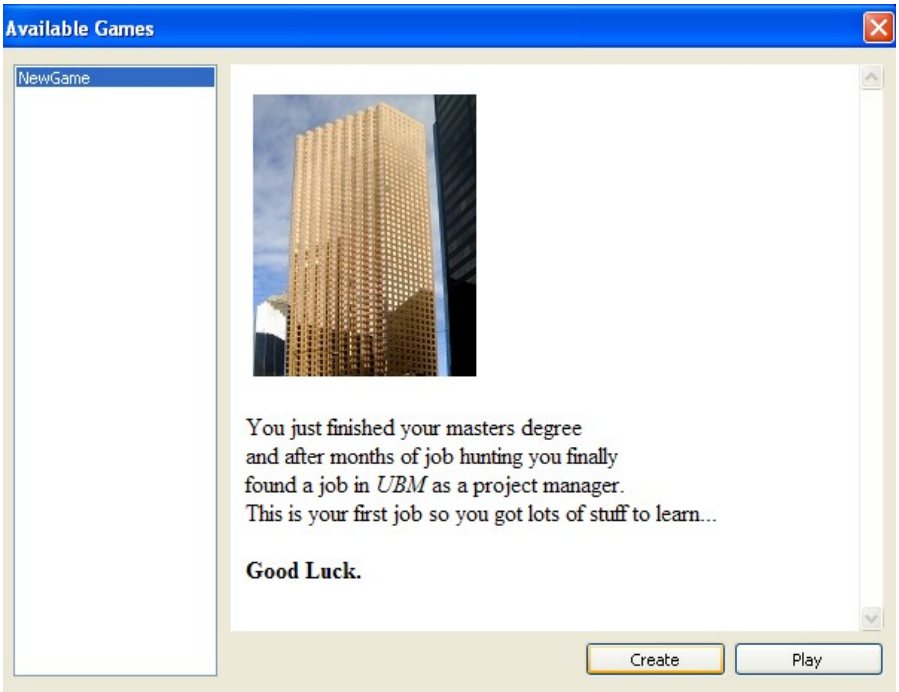
1. **SinglePlayer**
2. **MultiPlayer**



Στην περίπτωση που ο χρήστης επιλέξει το MultiPlayer mode εμφανίζεται το παράθυρο διαλόγου εισαγωγής στο σύστημα. Ο χρήστης πρέπει να εισάγει ένα ψευδώνυμο και ένα κωδικό για να εισέλθει στο σύστημα. Το ψευδώνυμο είναι μοναδικό για κάθε χρήστη. Αν η διαδικασία εισαγωγής αποτυγχάνει η εφαρμογή εμφανίζει το κατάλληλο μήνυμα σφάλματος. Ο χρήστης πρέπει να προσπαθήσει ξανά με τα νέα στοιχεία.

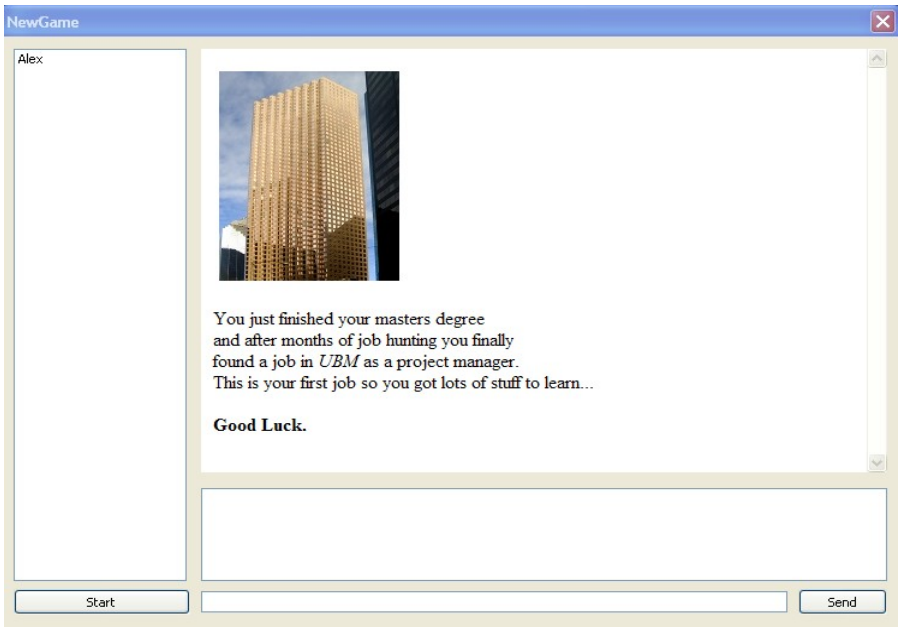


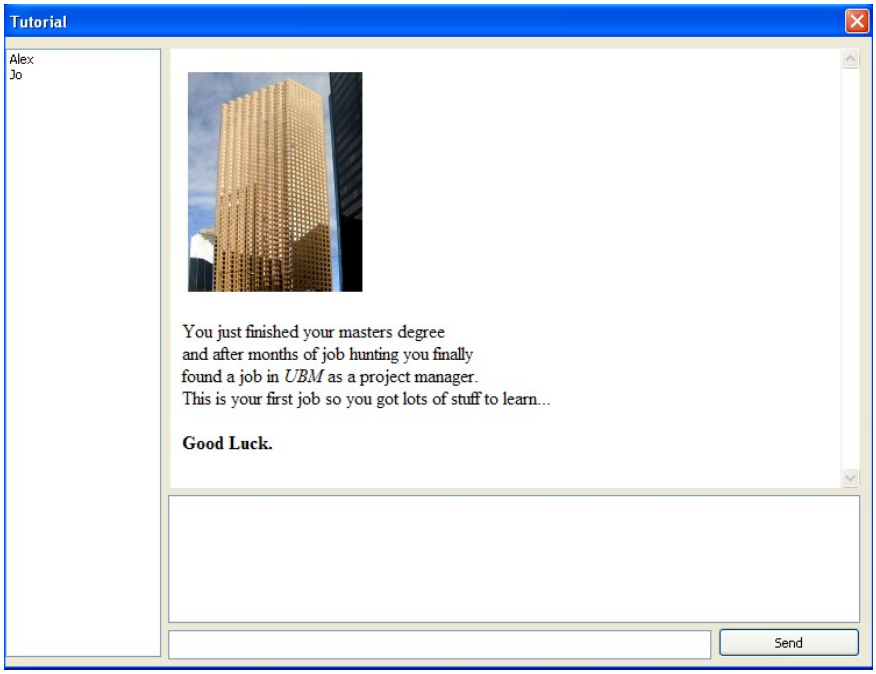
Με την επιτυχημένη είσοδο στο σύστημα ο χρήστης πρέπει να επιλέξει το παιχνίδι που θα παίξει. Το παράθυρο επιλογής παιχνιδιού παρέχει την λίστα με τα διαθέσιμα παιχνίδια στον server. Ο χρήστης μπορεί να επιλέξει ένα από αυτά και να διαβάσει τις πληροφορίες για αυτό το παιχνίδι. Για την εισαγωγή στο παιχνίδι ο χρήστης πρέπει να πατήσει το κουμπί Play. Εκτός από τα διαθέσιμα παιχνίδια ο χρήστης μπορεί να επιλέξει την επιλογή δημιουργίας νέου παιχνιδιού.



Αν ο χρήστης επιλέξει την δημιουργία νέου παιχνιδιού εμφανίζεται το ιδικό παράθυρο δημιουργίας παιχνιδιού στο οποίο εμφανίζονται όλα τα παιχνίδια που είναι διαθέσιμα τοπικά στο μηχάνημα του χρήστη. Ο χρήστης μπορεί να επιλέξει οποιοδήποτε από αυτά

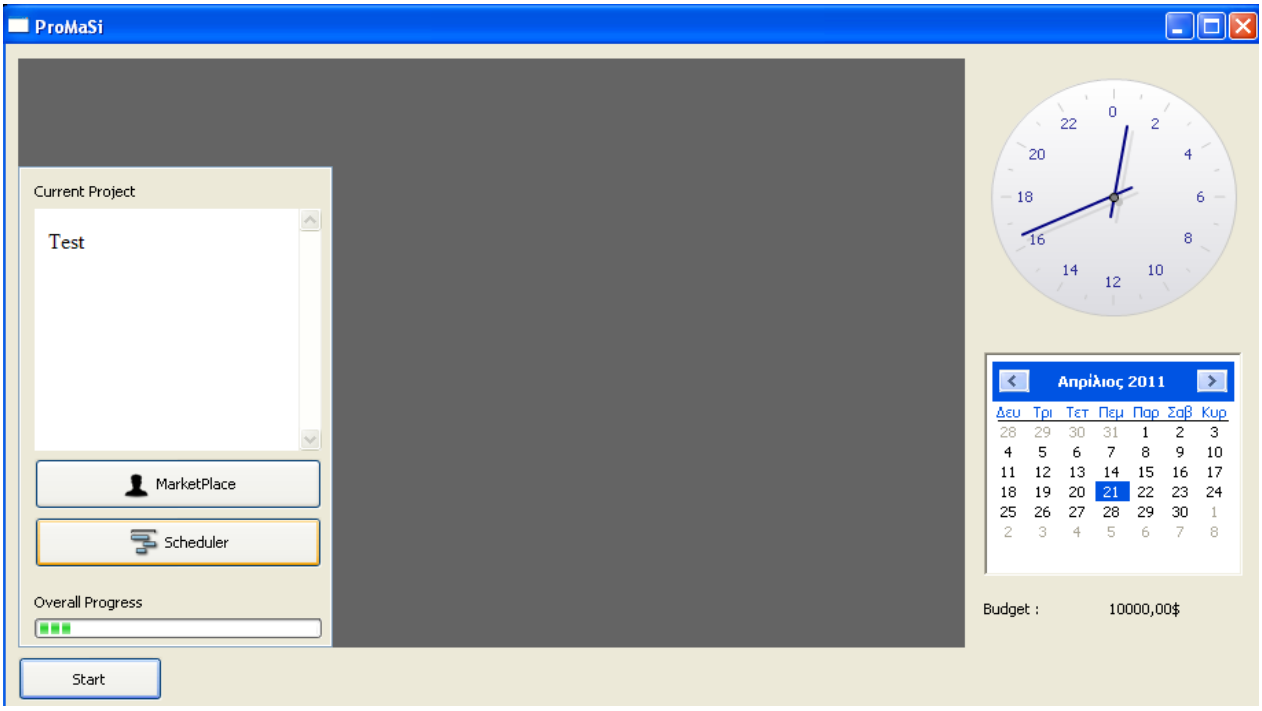
και να προχωρήσει στην δημιουργία τέτοιου παιχνιδιού στον server. Όταν ο χρήστης πατήσει το κουμπί Create το παιχνίδι της επιλογής του ανεβαίνει στον server και οι υπόλοιποι χρήστες που είναι συνδεδεμένοι στον ίδιο server μπορούν να δουν αυτό το παιχνίδι.





Εκτός της επιλογής δημιουργίας νέου παιχνιδιού ο χρήστης μπορεί να επιλέξει την εισαγωγή στο ήδη υπάρχον παιχνίδι. Σ' αυτήν την περίπτωση εμφανίζεται το παράθυρο αναμονής παιχνιδιού όπου ο χρήστης έχει την δυνατότητα συνομιλίας με τους υπόλοιπους

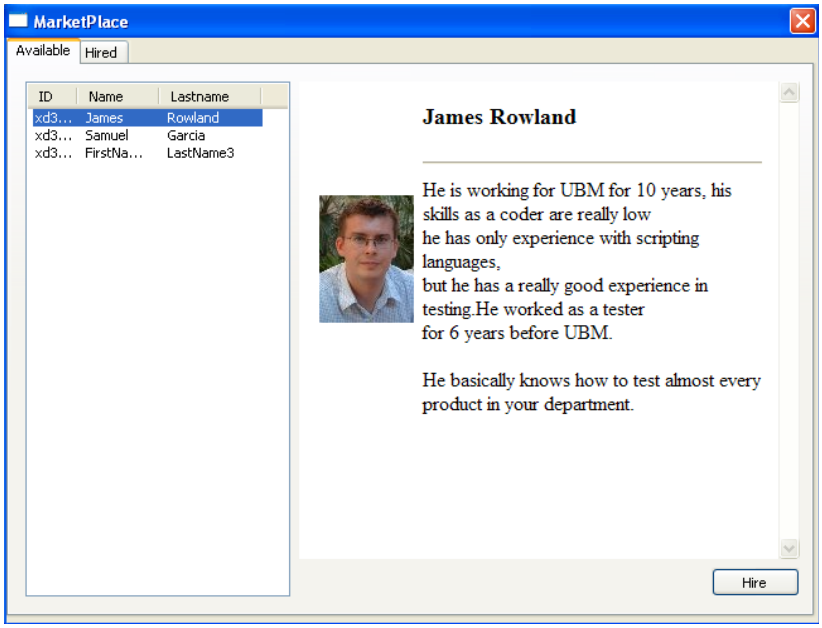
παίκτες του ίδιου παιχνιδιού. Σ' αυτήν την κατάσταση ο χρήστης περιμένει να ξεκινήσει το παιχνίδι που έχει επιλέξει. Όταν ο δημιουργός του παιχνιδιού πατήσει το κουμπί Start σε όλους τους παίκτες του παιχνιδιού θα εμφανιστεί το κυρίως παράθυρο του PROMASI. Το παράθυρο αυτό προσομοιώνει το περιβάλλον ενός ηλεκτρονικού υπολογιστή για να δίνει την αίσθηση ρεαλιστικότητα στον χρήστη του PROMASI.



5.2 Αγορά εργασίας.

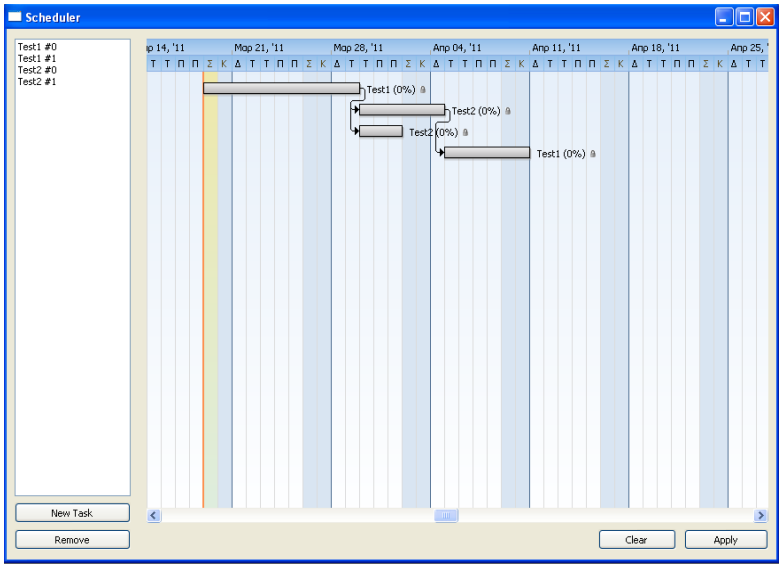
Η αγορά εργασίας στο MultiPlayer mode είναι μοιρασμένη ανάμεσα σε όλους του παίκτες του ίδιου παιχνιδιού και ο χρήστης έχει την πρόσβαση σ' αυτήν μέσω του μενού διαχείρισης προσωπικού.

Ο χρήστης έχει την δυνατότητα να προσλάβει ή να απολύσει το προσωπικό της εταιρίας. Τα κριτήρια για την πρόσληψη του κάθε υποψηφίου αξιολογεί ο ίδιος ο παίκτης του παιχνιδιού διαβάζοντας το βιογραφικό σημείωμα του υποψηφίου.

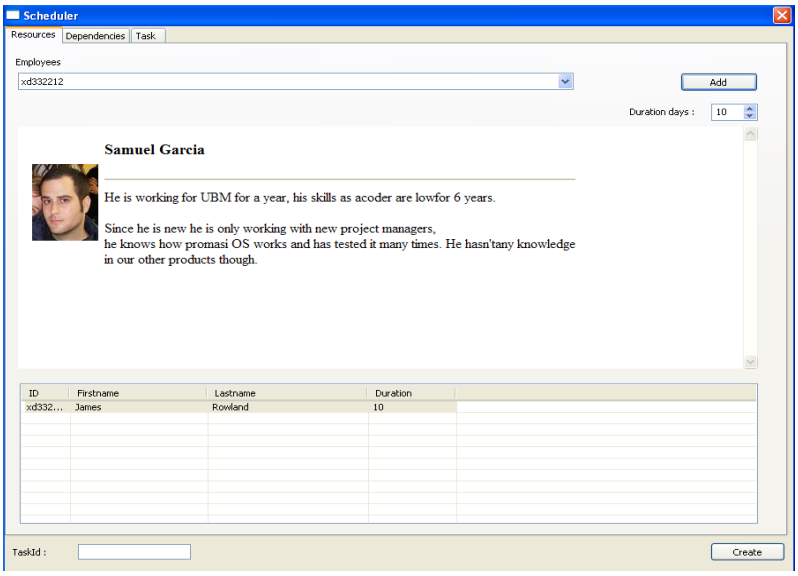


5.3 Προγραμματισμός του έργου.

Στην νέα έκδοση του PROMASI άλλαξε το παράθυρο προγραμματισμού της εργασίας. Η σχεδίαση του έργου γίνεται με την βοήθεια της βιβλιοθήκης **Nebula**. Ο χρήστης μπορεί να δημιουργεί διεργασίες τοποθετώντας τις στο συνολικό διάγραμμα Gantt. Κάθε διεργασία πρέπει να έχει τουλάχιστον έναν εργαζόμενο που την εκτελεί και μπορεί να έχει η να μην έχει κάποιες προαπαιτούμενες διεργασίες. Επίσης κάθε διεργασία σχεδιασμένη στο διάγραμμα Gantt αναφέρεται στην συγκεκριμένη διεργασία του έργου.



Για την προσθήκη νέας διεργασίας ο χρήστης πατάει το κουμπί δημιουργίας διεργασίας “New Task”. Η εφαρμογή εμφανίζει την φόρμα επεξεργασίας της διεργασίας. Σ' αυτήν την φόρμα ο χρήστης πρέπει να συσχετίσει την νέα διεργασία του με την διεργασία του έργου, να ορίσει το προσωπικό που θα εκτελεί την διεργασία και τα τυχόν προαπαιτούμενα για αυτήν την διεργασία.


Scheduler
Resources
Dependencies
Task
Employees
xd332212
Add
Duration days :
10
Samuel Garcia
He is working for UBM for a year, his skills as acoder are lowfor 6 years.
Since he is new he is only working with new project managers,
he knows how promasi OS works and has tested it many times. He hasn'tany knowledge
in our other products though.
ID
Firstname
Lastname
Duration
xd332...
James
Rowland
10
TaskId :
Create

6 Συμπεράσματα

6.1 Χρονοδιάγραμμα του έργου

Η υλοποίηση του MultiPlayer mode για το PROMASI διάρκεσε συνολικά δέκα επτά μήνες, επτά από τους οποίους ξοδεύτηκαν στην προσπάθεια υλοποίησης του MultiPlayer mode με δεδομένη την παλιά αρχιτεκτονική του PROMASI. Δυο μήνες για την σχεδίαση του πρωτοκόλλου επικοινωνίας και της νέας αρχιτεκτονικής του έργου, τέσσερις μήνες για την υλοποίηση της νέας έκδοσης του PROMASI και τέσσερις μήνες για την συγγραφή αυτού του εγγράφου. Κατά μέσο όρο δαπανούνταν περίπου 70 ανθρωποώρες/μήνα.

Η υλοποίηση του έργου δεν έχει ολοκληρωθεί πλήρως και συνεχίζεται και αυτήν την στιγμή. Τα βασικά στοιχεία που πρέπει να υλοποιηθούν στο μέλλον είναι.

1. Σχεδιαστής μοντέλων προσομοίωσης.
2. Ένα ρεαλιστικό μοντέλο προσομοίωσης.
3. Βάση δεδομένων για την διαχείριση λογαριασμών του χρήστη.
4. Επικοινωνία με την βάση και λειτουργία αποθήκευσης της τρέχουσας κατάστασης της εταιρίας που χειρίζεται ο χρήστης.

Για την υλοποίηση των παραπάνω λειτουργιών απαιτείται επιπλέον χρόνος. Ο προγραμματισμός του έργου δείχνει ότι ένα ρεαλιστικό μοντέλο προσομοίωσης πρέπει να υλοποιηθεί μέχρι τον Σεπτέμβρη του 2011. Εκτός από το μοντέλο προσομοίωσης πρέπει να διορθωθούν κάποιες δυσλειτουργίες που υπάρχουν στο σύστημα του PROMASI.

6.2 Τελικοί στόχοι

Ο κύριος στόχος του έργου ήταν η υλοποίηση ενός MultiPlayer mode για την εφαρμογή προσομοίωσης PROMASI. Κατά την διάρκεια του έργου πετύχαμε τον βασικό στόχο και υλοποιήσαμε το MultiPlayer mode στον οποίο ο χρήστης μοιράζεται την κοινόχρηστη αγορά εργασίας με τους υπόλοιπους παίκτες του παιχνιδιού. Επειδή κατά την διάρκεια του έργου άλλαξε η αρχιτεκτονική του PROMASI και προστέθηκαν οι βασικοί στόχοι που υπήρχαν στην προηγούμενη έκδοση.

6.3 Λάθη

Το μεγαλύτερο λάθος που κόστισε και τον περισσότερο χρόνο στην υλοποίηση του έργου ήταν τελικά η προσπάθεια να στηριχτούμε στην ήδη υπάρχουσα υλοποίηση του PROMASI. Η πολυπλοκότητα του κώδικα έκανε αδύνατη την υλοποίηση του MultiPlayer mode. Αυτό οδήγησε στην σχεδίαση και υλοποίηση μιας νέας αρχιτεκτονικής και την υλοποίησης του έργου με βάση αυτήν. Η χρήση της νέας αρχιτεκτονική προϋποθέτει την υλοποίηση των παλιών λειτουργιών του έργου από την αρχή συμπεριλαμβανομένου και το γραφικό περιβάλλον μαζί με τα κυρίως συστατικά του προσομοιωτή.

7 Παράρτημα

7.1 Εργαλεία ανάπτυξης

Για την υλοποίηση του PROMASI χρησιμοποιήθηκε η αντικειμενοστραφής γλώσσα προγραμματισμού Java και περιβάλλον ανάπτυξης λογισμικού “IDE” **Eclipse**.

Για την ανάπτυξη γραφικού περιβάλλοντος χρησιμοποιήθηκε το WindowsBuilder plugin. Το WindowsBuilder plugin είναι ελεύθερο λογισμικό που μας επιτρέπει να κατασκευάζουμε τα παράθυρα χρησιμοποιώντας τις ποιο γνωστές και διαδεδομένες βιβλιοθήκες της Java για την ανάπτυξη του γραφικού περιβάλλοντος “**SWT, SWING, GWT, AWT**”, μέσα από ένα φιλικό στον χρήστη γραφικό περιβάλλον. Η ανάπτυξη του γραφικού περιβάλλοντος στο PROMASI γίνεται με την βοήθεια της βιβλιοθήκης SWT. Κάθε project “βιβλιοθήκη” του PROMASI είναι ένα plugin. Για την δημιουργία και ανάπτυξη των plugins χρησιμοποιήθηκε μια ειδική έκδοση του Eclipse “Eclipse for RCP and RAP developers”. Τέλος για την διαχείριση του κώδικα χρησιμοποιείται το σύστημα διαχείρισης “Source Control” SVN.

7.1.1 Eclipse

Το Eclipse είναι ένα, cross platform, ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών IDE γραμμένο σε Java. Το Eclipse υποστηρίζει πολλών κατηγοριών plugins. Επίσης παρέχει ένα online σύστημα ανανέωσης και εγκατάστασης τον plugins “MarketPlace”. Εκτός από την Java το Eclipse υποστηρίζει και άλλες γλώσσες προγραμματισμού “C++, PHP, Python, COBOL, Ruby κ.α”. Για την γλώσσα προγραμματισμού Java το Eclipse υποστηρίζει το code autocompletion, code formatting, refactoring κ.α.

7.1.2 Windows Builder

Windows Builder είναι ένα εύχρηστο εργαλείο για ανάπτυξη του γραφικού περιβάλλοντος στα έργα λογισμικού που χρησιμοποιούν την γλώσσα προγραμματισμού java και περιβάλλον ανάπτυξης Eclipse. Η εγκατάσταση του Windows Builder είναι πολύ απλή μέσα από το μενού εγκατάστασης του MarketPlace του Eclipse.

Μετά την εγκατάσταση του WindowsBuilder είμαστε έτοιμοι για την σχεδίαση του Gui μέσα από το περιβάλλον του. Για την δημιουργία νέου παραθύρου ακολουθούμε τα έξις βήματα. Στο menu του Eclipse επιλέγουμε File -> New -> Other, εμφανίζεται παράθυρο επιλογής όπου μπορούμε να επιλέξουμε τι θέλουμε να δημιουργήσουμε. Πηγαίνουμε στο WindowsBuilder->SWT Designer->Swt και επιλέγουμε το Dialog. Πατάμε next και πληκτρολογούμε το όνομα της κλάσης που θέλουμε να δημιουργήσουμε. Όταν η κλάση είναι έτοιμη μπορούμε να την ανοίξουμε χρησιμοποιώντας το Windows Builder “δεξί click πάνω στο αρχείο της κλάσης Opent With->Windows Builder Editor.

7.1.3 Visual Paradigm

Visual Paradigm είναι εργαλείο κατασκευής διαγραμμάτων UML. Είναι γραμμένο σε Java και παρέχει το γραφικό περιβάλλον για την κατασκευή διαφόρων τύπων UML διαγραμμάτων. Με Visual Paradigm μπορούμε να κατασκευάσουμε:

- **Class diagam**
- **Sequence diagram**
- **State machine diagram**
- **UseCase diagram**
- **Communication diagram**
- **Package diagram**
- **Deployment diagram**
- **Component diagram**
- **Composite structure diagram**

Το Visual Paradigm είναι ένα εμπορικό ποιόν αλλά παρέχει και την Community έκδοση που είναι δωρεάν για μη εμπορική χρήση. Στο PROMASI χρησιμοποιείται η Community έκδοση του Visual Paradigm.

7.1.4 Βιβλιοθήκη SWT

The Standard Widget Toolkit είναι μια Open source και cross platform βιβλιοθήκη

ανάπτυξης του γραφικού περιβάλλοντος για τους προγραμματιστές Java. Η swt παρέχει μια πληθώρα από έτοιμα component που επιτρέπουν την εύκολη και γρήγορη ανάπτυξη του γραφικού περιβάλλοντος.

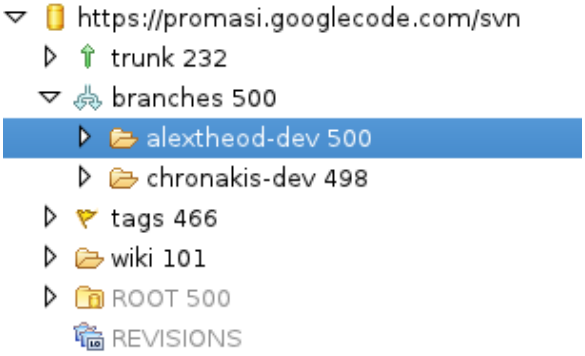
Εκτός από τα βασικά components της swt στο PROMASI χρησιμοποιήθηκαν και κάποια άλλα components που βασίζονται στην βιβλιοθήκη αυτή. Για παράδειγμα το διάγραμμα Gantt και το CDateTime είναι έτοιμα components του Nebula project.

7.1.5 Source Control

Για την ευκολότερη διαχείριση και την ασφάλεια του κώδικα οι σύγχρονοι μέθοδοι ανάπτυξης λογισμικού προϋποθέτουν την ύπαρξη κάποιας εφαρμογής διαχείρισης εκδόσεων. Το σύστημα διαχείρισης εκδόσεων επιτρέπει την εύκολη αποθήκευση και την ανάκτηση του κώδικα μέσω του απομακρυσμένου server. Για την διαχείριση των εκδόσεων του PROMASI χρησιμοποιήθηκε το σύστημα διαχείρισης εκδόσεων SVN. Το σύστημα αυτό παρέχει την δυνατότητα δημιουργίας του βασικού κορμού του project και επιμέρους κλαδιών όπου αποθηκεύονται είτε δοκιμαστικές εκδόσεις είτε εκδόσεις στις που υλοποιούν κάποιες επιπρόσθετες λειτουργίες.

Σήμερα το PROMASI χρησιμοποιεί το σύστημα διαχείρισης εκδόσεων με παροχέα την Google “GoogleCode”. Για να εισέλθει κανείς στο σύστημα αυτό πρέπει να εγκαταστήσει ένα πρόγραμμα client για το σύστημα SVN, να εισάγει την URL “https://promasi.googlecode.com/svn/” που περιγράφει την τοποθεσία του project και τα συνθηματικά του για την πρόσβαση στο σύστημα. Στο SVN υπάρχει ένας γενικός κορμός trunk όπου είναι αποθηκευμένη η προηγούμενη έκδοση του PROMASI. Εκτός από τον κυρίως κορμό στο SVN του PROMASI υπάρχουν τέσσερα κλαδιά branches

Το κλαδί με όνομα **alextheod-dev** είναι αυτό που αφορά την τρέχουσα υλοποίηση του multiplayer.


https://promasi.googlecode.com/svn
trunk 232
branches 500
alextheod-dev 500
chronakis-dev 498
tags 466
wiki 101
ROOT 500
REVISIONS

8 Βιβλιογραφία

- Γρηγοριάδης, Γρηγόριος: *PRO.MA.SI*, Πτυχιακή Εργασία, Τμήμα Πληροφορικής & Επικοινωνιών, ΤΕΙ Σερρών, 2009
- Cibraro, Pablo and Claeys, Kurt and Cozzolino, Fabio and Johann Grabner : Professional WCF 4: Windows Communication Foundation with .NET 4, John Wiley & Sons, 2010
- Gamma, Erich Helm, Richard Johnson, Ralph and John Vlissides: *Design patterns : elements of reusable object-oriented software*. Addison Wesley,1994
- Guojie, Jackwind Li: *Professional Java Native Interfaces with SWT/JFace*, John Wiley & Sons, 2004
- Ludewig, Jochen and Deininger, Marcus. TEACHING SOFTWARE PROJECT MANAGEMENT BY SIMULATION: THE SESAM PROJECT, Irish Quality Association (ed.): 5th European Conference on Software Quality, Dublin 1996, pp.417 – 426
- Lock, Dennis: *Project Management*, Gower Publishing Ltd, 2007
- Madachy, Raymond, *Software process dynamics*, IEEE Press, 2008
- Navarro, Emilly: SimSE: *A Software Engineering Simulation Environment for Software Process Education*, PhD Thesis, University of California, Irvine, 2006
- Papazoglou, Michael: *Web Services: Principles and Technology*, Prentice Hall, 2007
- Pruyt, Erik: *Rabbits and Foxes Model*, http://forio.com/simulate/e.pruyt/rabbits-and-foxes/simulation/# 19/03/2011
- Sterman, John D. *Business Dynamics: Systems thinking and modeling for a complex world*. McGraw Hill, 2000
- Walrath, Kathy, Campione, Mary, Huml, Alison and Sharon Zakhour: *The JFC Swing Tutorial: A Guide to Constructing GUIs*, Prentice Hall; 2 edition. 2004
- Wikipedia, “Enterprise Software”, http://en.wikipedia.org/wiki/Enterprise_software