****

**ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ**

Μάθημα: Κατανεμημένα Συστήματα

Εργαστηριακός Διδάσκων: Αναγνωστόπουλος Μ. – Δαμόπουλος Δ.

**2η ομαδική Εργασία**

**Μέλοι ομάδας εργασίας:**

Μαρία Θεοδωράκη Α.Μ.:321/2008041

Βερροιοπούλου Αθηνά Α.Μ.:321/2008011



**Ημερομηνία Παράδοσης : 29/04/2013**

**ΖΗΤΟΥΜΕΝΑ**

**Α)Υλοποίηση**

**Να υλοποιηθεί το παραπάνω σύστημα σε Java χρησιμοποιώντας RMI για την διαδιεργασιακή επικοινωνία. Στην απάντησή σας στο αρχείο κειμένου θα πρέπει να περιγράψετε σύντομα:**

**α) τις βασικές σχεδιαστικές αποφάσεις που έχετε λάβει για την υλοποίηση της εφαρμογής,**

**β) τυχόν παραδοχές που έχετε κάνει, πέραν των προδιαγραφών που σας έχουν δοθεί, και**

**γ) οδηγίες για την εκτέλεση τόσο του πελάτη όσο και των εξυπηρετητών.**

**Απάντηση: (Α,Β,Γ)**

Η εφαρμογή μας στο παιχνίδι bingo σε Java είναι ένας bingo\_client για τον πελάτη και 2 εξυπηρετητές ο game\_server, και ο win\_server, άρα δύο Java προγράμματα που τρέχουν σε ξεχωριστές Java Virtual Machines. Η εφαρμογή παιχνίδι είναι ο server, και η εφαρμογή αναπαραγωγής είναι ο πελάτης. Στα προγράμματα γίνετε χρήση RMI για αυτό φτιάχνουμε μια διεπαφή. Ότι υπάρχει μέσα στη διεπαφή πρέπει να υπάρχει και στο κύριως πρόγραμμα.

Ο **εξυπηρετητής game\_server** καταγράφει τους παίκτες (public boolean authentication(String username, String pin) ) και δημιουργεί τις κάρτες που παίζουν ( public Card[] deltia(long seed) ), ανακοινώνει τις μπάλες που έχουν επιλεγεί από την κλήρωση(public BingoBall announceNextBalls()) , επαληθεύει τη νίκη των παιχτών(public boolean verify(Card c)) και συνδέετε με τον άλλον εξυπηρετητή τον win\_server(game\_server ()). Υπάρχουν οι κλάσεις Card και BingoBall για να φτιαχτεί το κυρίως παιχνίδι , δηλαδή οι κάρτες και η διεπαφή\_2 που ενώνει τον ένα server με τον άλλον.

Ο **εξυπηρετητής win\_server** καταγράφει τους μέχρι τώρα νικητές σε μία λίστα (Current\_Nikitis(String NIKITIS) ) και δέχετε αιτήματα καταχώρησης (Katachorisi\_Nikiti(String user\_name)) από τον game\_server με τον οποίο συνδέετε.

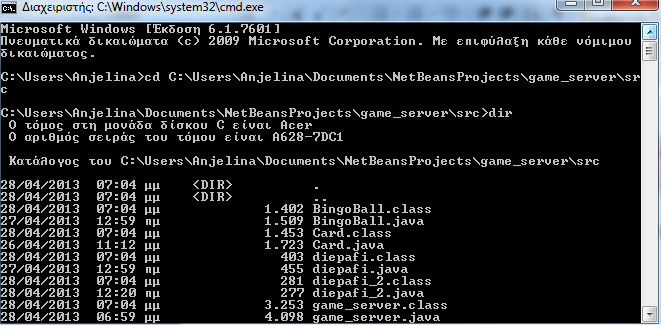
Ο **πελάτης bingo\_client** βάζει έναν χρήστη να δώσει στοιχεία για συμμετοχή στο παιχνίδι (authentication(String username, String pin)). Αν ο παίκτης έχει τη δυνατότητα να παίξει,

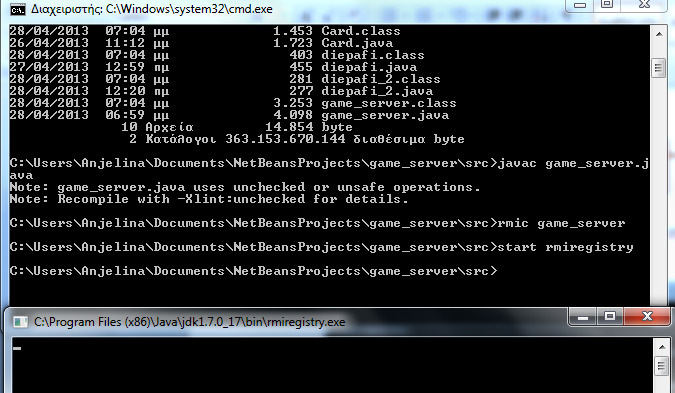
( Card[] deltia(long seed)) το παιχνίδι δίνει κάποιες κάρτες BINGO στον παίκτη. Καθώς το παιχνίδι προχωρά οι μπάλες ανακοινώνονται(void announcedBalls(BingoBall b)). Όταν η κάρτα έχει BINGO (5 σε μια σειρά, στήλη ή διαγώνιο) ο χρήστης κάνει κλικ σε ένα κουμπί το οποίο ειδοποιεί το παιχνίδι του τη νίκη(boolean verify(Card c)). Μπορεί να ρωτήσει ποιοί είναι οι μέχρι τώρα νικητές και να αποχωρήσει(close()). ‘Εχω σαν κλάση το client\_choice με το οποίο εμφανίζει ένα μενού επιλογών. Στον πελάτη βρίσκονται και τα γραφικά.

Παρακάτω βλέπουμε το διάγραμμα για την υλοποιήση της εφαρμογής

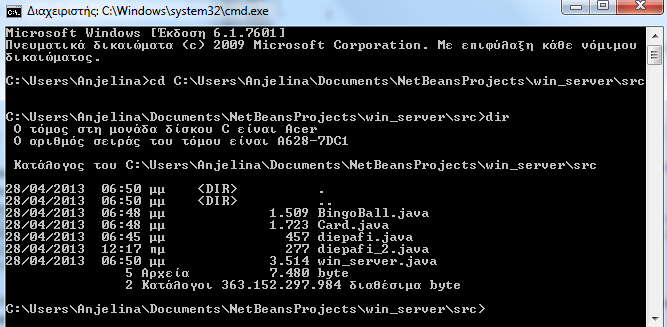


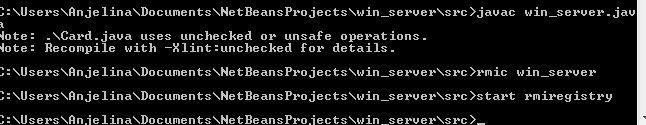
Για να εκτελέσουμε το παιχνίδι κάνουμε χρήση του cmd με το οποίο βλέπουμε τα παρακάτω. Πρώτα τρέχω τον game\_server:



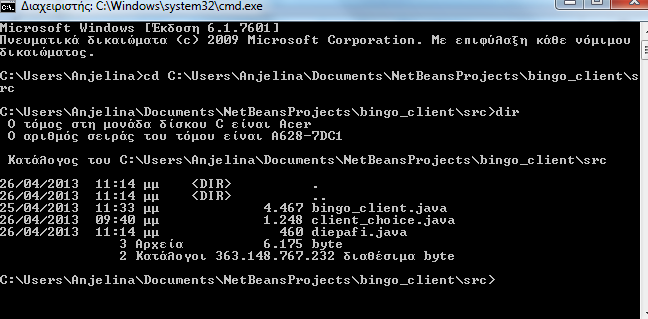


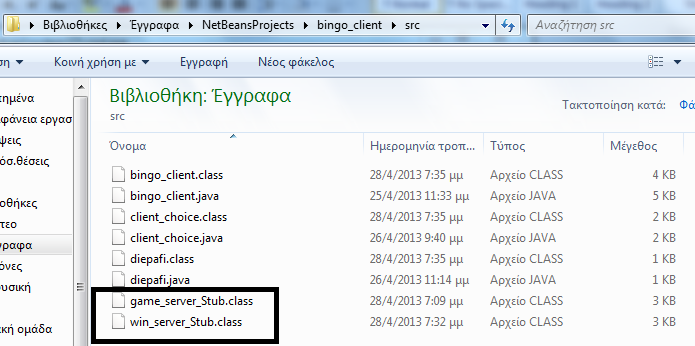
Για να εκτελέσουμε το παιχνίδι κάνουμε χρήση του cmd με το οποίο βλέπουμε τα παρακάτω. Πρώτα τρέχω τον win\_server:

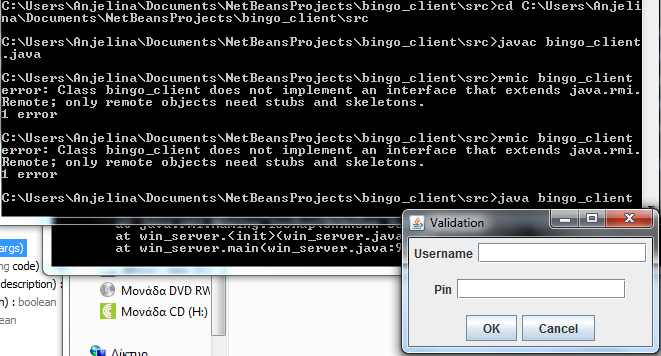




Για να εκτελέσουμε το παιχνίδι κάνουμε χρήση του cmd με το οποίο βλέπουμε τα παρακάτω. Πρώτα τρέχω τον bingo\_client:







**Β) Συνθήκες ανταγωνισμού (race conditions)**

Τι θα γίνει εάν δύο ή περισσότεροι χρήστες ταυτόχρονα ισχυριστούν ορθά ότι έχουν

Μπινγκο?

public synchronized Card[] deltia(long seed) throws RemoteException{// methodos gia a8fentikopoihsi

if(lock==true){// an einai true

try {

wait();// perimenoun oi ipolipi kai gia anagnosi kai gia eggrafi

} catch (InterruptedException ex) {

Logger.getLogger(game\_server.class.getName()).log(Level.SEVERE, null, ex);

}

}

lock=true;// an perasei apo ton pano elexo dld den grafei kapios allos tote tin kano true

Card[] cards = new Card[3];

Random generator = new Random(seed);

int randomInt = generator.nextInt(999999);

for (int i = 0; i < 3; i ++){

cards[i] = new Card(generator);

}

lock=false;// kano false tin metavliti oste na mporoun na diavasoun kai na gra4oun oi ipolipoi xristes

notifyAll();

return cards;

}

Ποιος παίχτης είναι πρώτος, δεύτερο,τρίτος κ.ο.κ. Με ποια σειρά θα καταγραφούν

Στο Winner server?

public synchronized String Katachorisi\_Nikiti(String user\_name){// methodos gia nea eggrafi

if(lock==true){// an einai true

try {

wait();// perimenoun oi ipolipi kai gia anagnosi kai gia eggrafi

} catch (InterruptedException ex) {

Logger.getLogger(win\_server.class.getName()).log(Level.SEVERE, null, ex);

}

}

lock=true;// an perasei apo ton pano elexo dld den grafei kapios allos tote tin kano true

Random randomGenerator = new Random();

int randomInt = randomGenerator.nextInt(999999);//dimiourgo ena tixaio arithmo

String num=Integer.toString(randomInt);

username.add(user\_name);// perno to username

Winner.add(num);// kai ton tixaio aritho san afm

x="O nikitis katachorithike me arithmo: "+num;// minima gia epistrofi

lock=false;// kano false tin metavliti oste na mporoun na diavasoun kai na gra4oun oi ipolipoi xristes

notifyAll();// tous idopio

return x;

}