

Cerinte proiect retele de calculatoare

Proiectul va consta intr-o aplicatie client - server pe socket-uri sau cu apel de metode la distanta, implementata intr-un limbaj la alegere (C#, Java, C++/C, etc.), putand fi realizat individual sau in echipa de maxim trei persoane, si va fi prezentat pe calculator in ultima activitate de seminar, pe o tema aleasa din lista de mai jos:

1. Partajarea fisierelor:

- Clientul se autentifica prin cont, triminand server-ului o lista cu fisierele pe care le publica, si primeste lista tuturor fisierelor publicate de catre ceilalti clienti autentificati;
- Cand un client se autentifica, ceilalti clienti autentificati primesc o notificare de adaugare a acestuia, impreuna cu lista de fisiere pe care o publica;
- Cand un client isi incheie sesiunea cu server-ul, acesta ii confirma incheierea sesiunii si notifica ceilalti clienti autentificati sa stearga din lista clientul respectiv;
- Un client poate solicita server-ului descarcarea unui fisier de la alti clienti;
- Server-ul solicita detinatorului fisierului respectiv citirea continutului acestuia;
- Ulterior, server-ul livreaza continutul fisierului clientului care l-a solicitat;
- Clientul salveaza fisierul in sistemul sau de fisiere;
- Fiecare client va avea un director gazda expus, care va fi monitorizat;
- La adaugarea unui nou fisier in acest director, clientul va notifica prin intermediul server-ului adaugarea fisierului;
- La stergerea unui fisier din acest director, clientul va notifica in mod similar ceilalti clienti prin intermediul server-ului.

2. Editarea partajata de fisiere text:

- Server-ul gestioneaza o lista de fisiere text dintr-un director gazda;
- Clientul se autentifica prin nume si primeste lista fisierelor text, precum si numele utilizatorului care are in editare fiecare fisier;
- Orice client autentificat poate solicita spre vizualizare continutul unui fisier, caz in care server-ul ii trimite ultima versiune de pe disc a acestuia;
- Un client poate solicita preluarea in editare a unui fisier disponibil, caz in care server-ul ii livreaza continutul acestuia si notifica ceilalti clienti ca fisierul respectiv este in editare de catre clientul care l-a solicitat;
- Clientul poate actualiza continutul fisierului, solicitand server-ului salvarea noii versiuni, caz in care server-ul va actualiza pe disc continutul fisierului cu ce a primit de la client si va notifica toti clientii care au in vizualizare acest fisier cu noul continut, pentru a-si actualiza datele afisate;
- Clientul poate renunta la editarea fisierului, caz in care server-ul va notifica tuturor clientilor autentificati ca resursa respectiva nu mai este in editare de catre clientul care o preluase, fiind disponibila pentru preluarea in editare;

- La adaugarea sau stergerea unui fisier de pe server, acesta va notifica toti clientii autentificati cu privire la numele fisierului afectat de operatie, pentru a fi adaugat sau sters din lista fisierelor disponibile in client.

3. Partajarea de obiecte in memorie:

- Clientul se conecteaza la server si primeste o lista de chei, fiecare cheie identificand un obiect publicat pe server de clientii conectati;
- Un client poate cauta un obiect pe server dupa cheie;
- Server-ul mentine un dictionar cu asocierile dintre chei si clientul pe care se gaseste obiectul corespunzator unei chei;
- La primirea unei solicitari de regasire a unui obiect dupa cheie, server-ul identifica pe ce client se afla obiectul si solicita transferarea continutului obiectului respectiv de pe clientul care-l stocheaza;
- In momentul primirii obiectului, server-ul il livreaza clientului care l-a solicitat;
- Un client poate publica un nou obiect pe server prin trimiterea unei chei care este asociata obiectului;
- Server-ul verifica unicitatea cheii in functie de care accepta inregistrarea obiectului in dictionar, notificand toti clientii conectati cu noua cheie publicata;
- Un client poate sterge o cheie de pe server publicata de el in prealabil, caz in care server-ul va notifica toti clientii conectati pentru stergerea cheii respective din lista.

4. Rezervarea de resurse:

- Server-ul gestioneaza o lista de resurse care pot fi alocate clientilor pentru un interval de timp;
- Clientul se autentifica prin nume si primeste lista resurselor impreuna cu lista rezervarilor pentru fiecare resursa;
- Un client poate solicita blocarea unei resurse pentru un interval de timp in vederea completarii unei rezervari;
- Server-ul notifica toti clientii autentificati in privinta blocarii resursei pentru rezervare, astfel incat un alt client sa nu mai poata solicita rezervarea aceleiasi resurse pentru acelasi interval de timp;
- Clientul care a initiat rezervarea poate anula solicitarea, caz in care server-ul notifica toti clientii ca resursa nu mai este blocata pe intervalul respectiv;
- Clientul care a blocat resursa poate finaliza rezervarea, caz in care server-ul notifica toti clientii autentificati in privinta noii rezervari create;
- Un client poate actualiza datele de inceput si sfarsit pentru o rezervare facuta de el in prealabil, caz in care server-ul va notifica toti clientii autentificati in privinta schimbarii respective;
- Un client poate sterge o rezervare facuta de el in prealabil, caz in care server-ul va notifica toti clientii autentificati in privinta stingerii acesteia.

5. Subscrierea si notificarea pentru canale de stiri:

- Server-ul gestioneaza o serie de canale identificate prin nume si descriere;
- Clientii se conecteaza la server si primesc lista canalelor;
- Un client poate publica un canal, caz in care server-ul notifica toti clientii conectati pentru a putea subscrie la el;
- Un client poate sterge un canal publicat de el, caz in care server-ul notifica toti clientii conectati;
- Un client poate subscrie la un canal pentru a fi notificat;
- Un client poate renunta la subscrierea la un anumit canal, caz in care nu va mai primi stirile publicate pe canalul respectiv;
- Clientul care a publicat un canal poate trimite stiri pe acel canal, caz in care toti clientii care au subscrib la el vor primi stirile respective;
- Server-ul filtreaza continutul stirilor pentru a le bloca pe acelea care contin anumite cuvinte predefinite care nu sunt permise, caz in care clientii care au subscrib la canalul pe care este livrata stirea nu vor mai fi notificati.

6. Vizualizarea sesiunii grafice la distanta:

- Server-ul gestioneaza lista sesiunilor grafice ale utilizatorilor conectati pe mai multe statii;
- Clientul se conecteaza, identificandu-se printr-un nume, care trebuie sa fie unic;
- Daca numele propus de utilizator este unic, atunci server-ul accepta conexiunea si notifica toti utilizatorii conectati pentru a-l adauga in lista;
- In caz ca mai exista un utilizator cu acelasi nume, server-ul refuza conectarea si notifica utilizatorul nou in acest sens;
- In momentul conectarii, un utilizator nou primeste lista cu toti utilizatorii care sunt deja conectati;
- Un utilizator conectat poate selecta prin intermediul server-ului un alt utilizator existent, urmand sa primeasca imaginea cu sesiunea grafica a lui la momentul respectiv;
- Clientul conectat primeste actualizari ale imaginii sesiunii grafice la distanta la un interval prestabilit, care sa permita urmarirea cursiva a activitatii la distanta;
- Cand un client inchide aplicatia, server-ul este notificat pentru a putea solicita clientilor conectati eliminarea acestuia din lista.

7. Culegerea de informatii sistem la distanta:

- Server-ul gestioneaza o serie de conexiuni client, permitand executia de interogari de tip Windows Management Instrumentation la distanta;
- Clientul se conecteaza la server, identificandu-se prin numele masinii si adresa IP;
- Server-ul trimite clientului conectat lista tuturor clientilor existenti;

- In acelasi timp, server-ul notifica toti clientii existenti pentru a adauga noul client;
- Un client compune o interogare WMI si selecteaza toti clientii sau un set de clienti, solicitand server-ului executia comenzii la distanta de catre clientii selectati;
- Server-ul notifica pe rand fiecare client selectat pentru a executa comanda primita si colecteaza rezultatul afisat de comanda;
- In final, server-ul raspunde clientului care a furnizat comanda cu rezultatele executiei acesteia la distanta, pe fiecare client selectat;
- Cand o aplicatie client se inchide, aceasta notifica server-ul pentru a solicita clientilor ramasi sa-l elimine din lista clientilor existenti.

8. Licitatie:

- Server-ul gestioneaza o lista de clienti, fiecare client putand pune in vanzare o mai multe produse;
- Clientul se conecteaza la server, indentificandu-se prin nume, care trebuie sa fie unic;
- In cazul in care mai exista un utilizator cu acelasi nume, server-ul refuza conectarea clientului;
- Server-ul accepta conexiunea client, raspunzand cu lista produselor in curs de licitare, pentru fiecare produs detaliindu-se numele celui care l-a oferit spre vanzare, pretul minim si pretul maxim la care s-a ajuns;
- Un client conectat poate pune in vanzare un produs identificat printr-un nume unic in lista de produse ale clientului si un pret minim de pornire;
- Durata unei licitatii este predefinita de server, fiind aceeasi pentru toate produsele puse in vanzare;
- Server-ul notifica toti clientii privind adaugarea unui nou produs disponibil pentru a fi licitat, cu detalii privind nume produsului, proprietarul si pretul de pornire;
- In momentul in care un client face o oferta pentru un produs, sunt notificati proprietarul si toti clientii care au ofertat pentru produsul respectiv;
- Clientii nu pot face oferte decat pentru produsele in curs de licitare;
- In momentul expirarii duratei unei licitatii, sunt notificati toti clientii pentru a marca produsul respectiv ca indisponibil pentru a mai fi licitat.

9. Retea de noduri:

- Fiecare nod este configurat sa se conecteze la o lista de noduri din proximitate;
- La pornirea unui nod, acesta incearca pe rand sa se conecteze la un nod din lista, in cazul conectarii cu succes, mentinand o singura conexiune deschisa;
- Fiecare nod expune o serie de servicii care pot fi pornite sau oprite prin rularea unei comenzi;

- Aplicatia client citeste la pornire lista serviciilor, precum si comanda care trebuie rulata pentru pornirea sau oprirea fiecaruia;
- Din aplicatia client se pot porni sau opri servicii;
- Aplicatia client permite interogarea starii serviciilor de pe fiecare nod, cat si pornirea sau oprirea unui serviciu de pe un anumit nod;
- In momentul in care o aplicatie client de pe un nod executa o comanda pe alt nod, nodul de la distanta confirma rezultatul executiei comenzii catre nodul care a lansat comanda.

10. Rutarea mesajelor:

- Server-ul gestioneaza o lista de destinatari, fiecare destinatar fiind identificat printr-un nume unic;
- Clientul se conecteaza la server si trimite un mesaj, in care precizeaza destinatarul, emitentul, subiectul si textul mesajului;
- In cazul in care destinatarul apare in lista gestionata de server, acesta va salva mesajul intr-un director local cu numele destinatarului, generand un nume de fisier bazat pe momentul salvarii mesajului;
- In cazul in care destinatarul nu apare in lista, server-ul are o lista de alte server-e pe care le interogheaza in momentul primirii unui mesaj, pentru a stabili catre ce alt server trebuie sa ruteze mesajul;
- Server-ul confirma clientului acceptarea mesajului;
- Server-ul poate primi pe aceeasi conexiune mesaje de la alte server-e pentru a verifica daca un anumit destinatar se gaseste in lista sa;
- In cazul in care destinatarul cautat nu se gaseste in lista sa, server-ul procedeaza similar livrarii unui mesaj interogand lista de server-e la care se poate conecta direct.

11. Gestiunea obiectelor dintr-o baza de date:

- Server-ul gestioneaza o lista de obiecte preluate dintr-o baza de date, fiecare obiect fiind identificat printr-o cheie unica;
- Client-ul poate interoga server-ul pentru a selecta o lista de obiecte cu anumite valori ale cheii;
- Server-ul tine pentru fiecare lista valorii cheilor pentru obiectele selectate;
- Clientii pot actualiza sau sterge obiecte dupa cheie;
- In momentul in care un obiect este modificat, server-ul va notifica acest lucru tuturor clientilor care selectasera in prealabil obiectul respectiv;
- Schimbarile asupra obiectelor sunt salvate in memoria server-ului si in baza de date.

12. Avionasele:

- Server-ul are o lista de fisiere de configurare, fiecare fisier corespunzand unei distributii a trei avioane pe o matrice de 10 x 10;

- O configuratie precizeaza distributia a trei avioane sub forma:

00A0000000

1111100020

0010002020

011100222B

0000002020

0000000020

0003330000

0000300000

0033333000

0000C00000

unde cifrele indica parti ale aceluiasi avion, iar literele corespund capului avionului, un avion avand urmatoarele segmente:

A - cap

11111 - aripi

1 - corp

111 - coada

- Server-ul alege in mod aleator o configuratie curenta;
- Clientii se conecteaza la server identificandu-se printr-un nume unic;
- Un client poate trage pentru a dobori avioanele, indicand linia si coloana in care trage;
- Un avion este doborat atunci cand se trage in capul lui;
- Server-ul ii va raspunde cu 0, daca niciun avion n-a fost atins, cu 1, daca o parte a unui avion a fost atinsa, sau cu X, daca avionul a fost atins in cap si doborat;
- Cand un client a reusit sa doboare toate avioanele, server-ul notifica toti clientii cu numele acestuia si alege alta configuratie curenta, jocul reluandu-se.

13. Centratea:

- Server-ul genereaza un numar de patru cifre diferite;
- Clientii se conecteaza la server identificandu-se printr-un nume unic;
- Un client propune server-ului solutia pentru numarul care trebuie ghicit, sub forma unui numar de patru cifre diferite;
- Server-ul ii raspunde clientului cu numarul ce centrate (cifre care se gasesc in numar pe aceeasi pozitie), respectiv de necentrate (cifre care se gasesc in numar, dar nu pe aceeasi pozitie);
- Cand un client a ghicit numarul server-ului, acesta notifica toti clientii cu numele celui care a ghicit si numarul de incercari din care a facut-o, generand alt numar, dupa care jocul se reia.

14. Motor pentru executia de script-uri la distanta:

- Clientii se conecteaza la server si publica o lista de script-uri identificate prin nume unic la nivelul server-ului;
- Server-ul tine lista clientilor impreuna cu asocierea ce script pe care client se gaseste;
- Pe durata unei sesiuni cu server-ul un client nu-si poate modifica lista de script-uri disponibile;
- Un client poate publica pe server o comanda compusa identificata printr-un nume unic la nivelul server-ului, constand intr-o secventa de script-uri apelabile in conducta cu un fisier de intrare primit ca argument, urmand ca iesirea generata de primul script din secventa sa fie trimisa ca intrare pentru cel de-al doilea si tot asa pana la ultimul script a carui iesire va constitui rezultatul executiei comenzii;
- Clientii pot suprascrie comenzile deja publicate prin publicarea pe server a uneia cu acelasi nume;
- Clientii pot solicita stergerea unei comenzi de pe server pe baza numelui acesteia;
- Server-ul primeste fisiere pentru executia unei comenzi compuse identificata dupa numele fisierului de intrare care trebuie sa coincida cu cel al comenzii de executat;
- Fisierul rezultat in urma executiei comenzii este trimis clientului care a initiat executia comenzii.

15. Coadă distribuită de mesaje:

- Fiecare client are o lista cu adresele unor server-e la care incearca sa se conecteze pe rand, pana reuseste sa se conecteze la primul;
- Fiecare client este la randul sau server, asteptand sa primeasca cereri de conectare sau de procesare de la alti clienti;
- Cand un client se conecteaza la un server, clientul ii comunica acestuia pe ce port poate fi contactat inapoi;
- Cand un server accepta o noua conexiune de la un client, acesta informeaza server-ul la care este conectat ca si client, cat si pe ceilalti clienti conectati la el in privinta punctului unde poate fi contactata inapoi conexiunea acceptata;
- Clientul poate subscrie sau poate renunta la o subscriere anterioara la server-ul la care este conectat pentru procesarea obiectelor identificate printr-o cheie;
- Cand un server primeste o cerere de subscriere sau de renuntare la subscriptie, acesta contacteaza la randul lui server-ul la care este conectat ca si client, cat si toti clientii conectati la el pentru a-i informa cu privire la ce client a efectuat operatia si pentru ce cheie;
- Clientii accepta mesaje in forma binara identificate printr-o cheie, produse de aplicatia clientului;

- Când un client accepta un mesaj, el identifică din datele stocate local ce clienți așteaptă să consume acest mesaj și unde să-i contacteze pentru a le livra mesajul, urmând să se conecteze la aceștia și să le trimită mesajul de procesat;
- La primirea unui mesaj pentru procesare, un server execută o comandă aferentă cheii mesajului de procesat, având ca argument datele mesajului primit spre procesare.

16. Depanarea programelor la distanță:

- Server-ul execută programe constând în instrucțiuni aritmetice cu atribuirea rezultatului într-o variabilă, instrucțiuni ce pot include referiri la valorile altor variabile, constante numerice, operatori aritmetici și paranteze, fiecare atribuire constituind o instrucțiune ce poate fi depanată de la distanță;
- Clienții se conectează la server și înregistrează o serie de puncte de oprire într-un program identificate prin numele programului și linia fiecărui punct de oprire a execuției;
- Server-ul acceptă un singur client care poate întrerupe execuția unui program la un moment dat;
- Clienții se pot atașa la un program, pot adăuga sau elimina puncte de oprire, sau se pot detașa de la depanarea execuției unui program;
- Aplicația server lansează în execuție mai multe programe în paralel;
- Pe durata execuției unui program, clientul care-l depănează execuția nu mai poate adăuga sau elimina puncte de oprire;
- În momentul în care server-ul ajunge în timpul execuției unui program la un punct interceptat de un client, acesta va aștepta din partea clientului comenzi pentru evaluarea unei variabile după nume sau de setare a valorii acesteia, respectiv va continua până la următorul punct de întrerupere a execuției după primirea din partea clientului a unei comenzi în acest sens;
- La încheierea execuției programului depanat, server-ul va notifica clientul care-l monitorizează în acest sens.

17. Sistem de fișiere distribuit:

- Server-ul expune un anumit director de pe mașina sa;
- Când un client se conectează, solicită informații despre directoarele și fișierele expuse de server, sincronizându-și conținutul unui director local cu cel de pe server, prin crearea, redenumirea, stergerea sau modificarea conținutului resurselor afectate, după caz.
- Când un client efectuează modificări asupra directorului sau local, aceste modificări sunt transmise server-ului, care efectuează aceleași modificări asupra directorului server, notificând toți clienții conectați pentru a-și sincroniza conținutul directorului lor local cu cel de pe server.

18. Sincronizarea proceselor la distanță:

- Server-ul asigura sincronizarea unor procese care se executa in aplicatiile client prin intermediul unor semafoare;
- Clientii se conecteaza la server si mentin conexiunea deschisa pe durata executiei procesului client;
- Un client poate solicita obtinerea accesului exclusiv pe un semafor identificat printr-un nume unic la nivelul server-ului;
- In cazul in care niciun alt client nu are acces exclusiv pe acel semafor, server-ul ii acorda clientului accesul, retinand ce client acceseaza semaforul respectiv;
- In cazul in care un client detine deja accesul la semaforul respectiv, server-ul va refuza clientului accesul exclusiv, adaugandu-l intr-o lista a clientilor care asteapta dupa eliberarea acelui semafor;
- Cand clientul care detine accesul exclusiv la un semafor il elibereaza, server-ul va da accesul exclusiv urmatorului client care asteapta, daca exista, notificandu-l in acest sens.

19. Executia paralela la distanta:

- Clientii incearca sa se conecteze pe rand la o lista de server-e, oprindu-se dupa ce reusesc sa se conecteze la primul;
- Clientul este la randul sau server, iar cand se conectaza la un server, ii trimite portul pe care asculta la randul sau sa primeasca procesari;
- Acesti clienti notifica la randul lor clientii lor cu privire la intrare unui nou client in cluster;
- Server-ul la care se conecteaza un client notifica restul clientilor conectati cu privire la adresa si portul unde poate fi contactat clientul acceptat;
- Un client poate solicita serverului cu cel mai mic grad de incarcare sa execute pe un numar de fire de executie in paralele o metoda a unei clase care intoarce un anumit rezultat;
- In cazul in care clasa respectiva nu se gaseste pe server-ul destinatie, clientul trimite continutul binar al clasei server-ului;
- Server-ul lanseaza in executie metoda pe numarul de fire de executie solicitat de client si cu argumentele primite de la acesta;
- Server-ul notifica toti clientii conectati la el pentru actualizarea gradului de incarcare cu procesarile primite sa le execute, precum si server-ul la care e el conectat, care atunci cand primeste notificarea in cauza, notifica la randul sau toti clientii care sunt conectati la el in acest sens;
- Cand server-ul a obtinut rezultatul in urma executiei unui fir, il trimite clientului si notifica toti clientii conectati la el pentru actualizarea gradului sau de incarcare, acestia notificand la randul lor clientii conectati la ei;
- Cand un client se deconecteaza, server-ul la care e conectat notifica toti clientii conectati la el, care la randul lor isi notifica si ei clientii conectati la ei.

20. Actualizarea aplicatiilor la distanta:

- Pe server exista o serie de aplicatii executabile, lista care nu se modifica pe durata rularii procesului server;
- Clientii se conecteaza la server si solicita lista acestora;
- Un client poate solicita descarcarea unei aplicatii;
- Server-ul mentine o lista cu toate aplicatiile descarcate de un client;
- Pe server se pot publica noi versiuni ale unei aplicatii prin suprascrierea celei existente;
- In acest caz, server-ul trimite tuturor clientilor care au descarcat aplicatia respectiva noua versiune;
- In cazul in care aplicatia ruleaza pe client, acesta salveaza versiunea primita si reincearca s-o suprascrie pe cea veche pana reuseste.

21. Distribuirea procesarii:

- Server-ul asculta pachetele venite pe adresa de loopback a subretelei pe un anumit port si tine o lista cu toti clientii activi;
- Clientii cand pornesc trimit un pachet pe adresa de loopback si portul pe care asculta server-ul prin care se inregistreaza in lista clientilor activi;
- Fiecare aplicatie client deschide un port pe care asteapta cereri de procesare, la inregistrarea cu server-ul comunicandu-i acest port;
- Inainte de inchidere, aplicatia client trimite un mesaj pe adresa de loopback si portul pe care asculta server-ul pentru stergerea clientului din lista clientilor activi;
- Server-ul asculta pe un port cereri de procesare de la clienti care constau in executia unui task custom furnizat la momentul cererii impreuna cu argumentele de apel, rezultatul executiei intors de server clientului constand in exit code-ul executiei task-ului;
- La primirea unui task de procesare, server-ul alege urmatorul client de procesare la rand, ii trimite pe portul de procesare codul binar al task-ului si argumentele de apel, clientul de procesare lansand in executie task-ul primit in local cu argumentele date, intorcand dupa terminarea executiei exit code-ul procesului care a rulat task-ul, care este apoi trimis de server clientului ce a solicitat executia task-ului respectiv.

22. Comunicatia la distanta prin intermediul tunelurilor:

- Pentru a traversa o regiune din Internet in care este deschis un singur port, solutia foloseste un server local care asculta pe o serie configurabila de porturi, si care trimite pachetele catre server-ul destinatie pe un singur port, adaugand insa si informatia despre portul destinatie dorit;
- La destinatie asculta un alt server pe singurul port accesibil la distanta, care extrage portul dorit din fiecare pachet si retrimite local pachetul pe portul dorit;

- Se vor implementa doi clienti care sa consume fiecare cate un tip de serviciu diferit, si cate un server corespunzator oferind serviciul respectiv, astfel incat sa se poata demonstra conectarea clientului la serverul de tunelare local, care mai departe se va conecta pe singurul port accesibil catre serverul de tunelare de la distanta, urmand ca acesta sa se conecteze la server-ul care ofera serviciul dorit la distanta.
- Spre exemplu: am putea implementa un server care trimite periodic timpul catre clienti si altul care face broadcast de mesaje text la nivel aplicatie (de tip talk).

23. Chat cu camere virtuale:

- Server-ul are o lista de camere virtuale, fiecare camera avand asociata o adresa de multicast, si raspunde la pachetele primite pe un anumit port prin care livreaza lista camerelor virtuale cu adresele de multicast aferente;
- In aplicatia server se pot adauga sau sterge camere virtuale;
- La pornire, clientul trimite un pachet pe adresa de broadcast a subretelei pe portul pe care asculta server-ul pentru a primi lista de camere virtuale si porturile lor aferente;
- Clientul se poate adauga in grupul unei camere virtuale sau poate iesi din camera virtuala curenta, fiind la un moment dat in cel mult un grup al unei camere virtuale;
- Clientul poate trimite mesaje text catre adresa de multicast aferenta camerei virtuale curente pe portul predefinit pentru comunicarea prin schimbul de mesaje;
- Fiecare client asculta adresa de multicast a camerei virtuale in care se afla pe portul predefinit pentru mesaje, afisand mesajele primite;
- La stergerea unei camere virtuale de catre server, precum si la adaugarea unei noi camere, server-ul va trimite un mesaj pe adresa de broadcast a subretelei si portul predefinit pentru comunicare.

24. Agent pentru controlul unor aplicatii de la distanta:

- Server-ul are configurata o lista de aplicatii, pentru fiecare aplicatie putand executa o serie de operatii, precum start, stop, restart, status, fiecare operatie avand asociata o comanda pe care server-ul o va rula cand este solicitata;
- Clientul se conecteaza la server si permite interogarea listei de aplicatii, precum si a comenzilor ce pot fi executate pentru fiecare aplicatie in parte;
- Clientul permite executia comenzilor pentru aplicatiile controlate, afisand rezultatul executiei acestora de catre server;
- De exemplu, controlul unui server de baze de date.

Punctajul aferent cerintelor este urmatorul:

- Server concurent care permite subscrierea si notificarea clientilor conform cerintelor functionale;
- Client capabil sa subscrie, sa invoce server-ul si sa trateze invocarile inapoi venite de la server.