

FILE-SHARING

Echipa:

PaleProgrammers

Membrii:

Ciobanu Alin

Ciuca Cristian

Răutoiu Alin-Ștefan

CUPRINS

- [1. Descrierea generala a aplicatiei](#)
- [2. Cerinte functionale](#)
 - [2.1. Actori](#)
 - [2.1.1. Administratorul de sistem](#)
 - [2.1.2. Utilizatorul](#)
 - [2.2. Descrierea folosirii aplicatiei](#)
 - [2.2.1. Autentificare](#)
 - [2.2.2 Log out](#)
 - [2.2.3 Cautare prin fişiere](#)
 - [2.2.4 Descarcarea fişierelor](#)
 - [2.3. Descrierea cerintelor functionale](#)
- [3. Arhitectura sistemului](#)
- [4. Cerinţele modelării datelor](#)
- [5. Cerinţe non-funcţionale](#)
 - [5.1 Cerinţele interfeţei cu utilizatorul](#)
 - [5.2 Cerinţele de performanţă](#)
 - [5.3 Cerinţele de disponibilitate şi fiabilitate](#)
 - [5.4 Cerinţele de securitate](#)
 - [5.5 Cerinţele de administrării sistemului şi de mentenanţă](#)
 - [5.6 Cerinţele pregătirii utilizatorului](#)
- [6. Consideraţii de testare](#)

1. Descrierea generala a aplicatiei

Scopul acestui proiect este elaborarea unui sistem ce va permite utilizatorilor să partajeze fișiere în cadrul unei rețele.

Utilizatorii vor avea posibilitatea de a căuta fișiere pe server și de a naviga prin ierarhia de fișiere partajate de un utilizator, iar apoi să descarce fișierele selectate.

2. Cerinte functionale

2.1. Actori

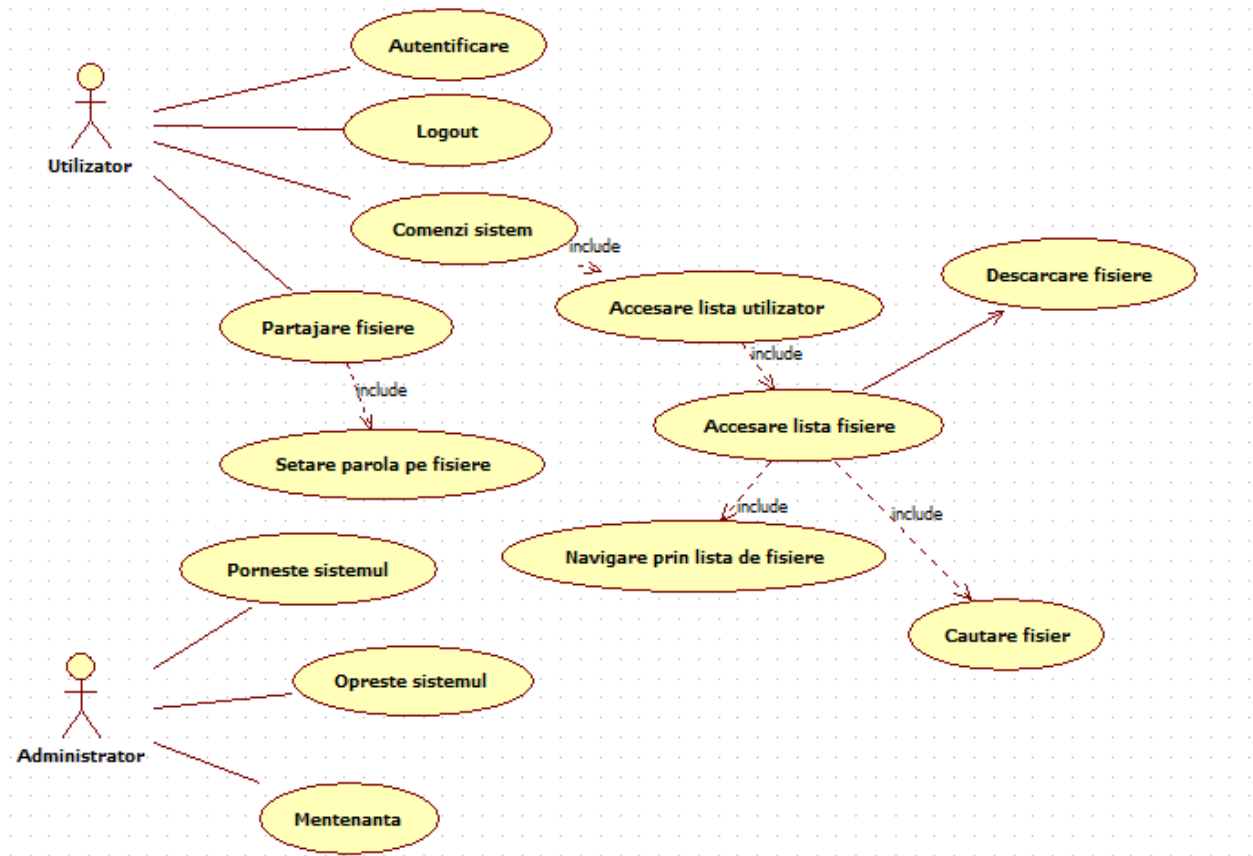
2.1.1. Administratorul de sistem

- pornește/oprește sistemul
- asigură buna funcționare a sistemului

2.1.2. Utilizatorul

- navigare prin ierarhia de fișiere a altui utilizator
- căutare fișiere
- inițiere/reluare transfer de fișiere
- protejare fișier prin parola
- alte actiuni(configurari, partajare fișiere, etc.)

2.2. Descrierea folosirii aplicatiei



2.2.1. Autentificare

Utilizatorii nu se vor autentifica propriu-zis in sistem, ci vor trebui sa introduca o parola pentru a descarca un fișier. De asemenea ei vor trebui sa seteze o parola pe fișierele proprii, parola ce va fi folosita de ceilalti utilizatori care vor sa ii acceseze fișierele

2.2.1.1. Preconditie

Utilizatorul trebuie sa se conecteze la server.

2.2.1.2. Descriere narativa

Scenariul principal

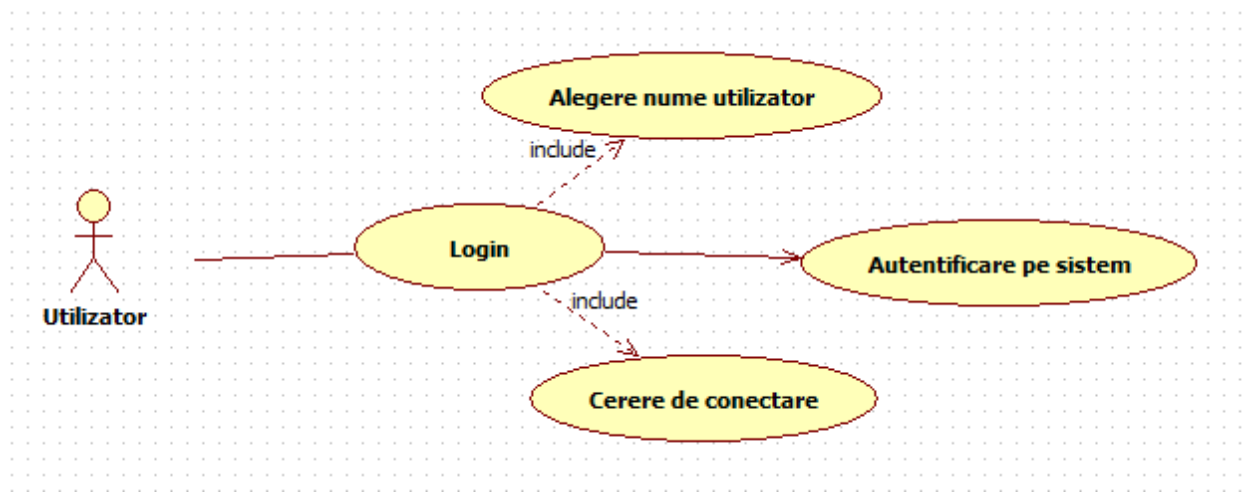
- ❖ Utilizatorul va introduce adresa serverului, isi va alege un nume de utilizator si se va conecta la server
- ❖ Daca serverul va accepta utilizatorul, va permite accesul la lista de utilizatori
- ❖ Utilizatorul se va conecta la un utilizator din lista de utilizatori
- ❖ Se parcurge lista de fișiere a utilizatorului ales
- ❖ Se alege un fișier și se introduce o parola pentru accesul la acesta
- ❖ Daca parola e recunoscută, utilizatorii vor putea sa descarce sau să deschidă fișierul

Scenariu alternativ

- ❖ Adresa serverului este gresita
 - Sistemul va afisa un mesaj de eroare
- ❖ Utilizatorul nu este acceptat
 - Numele este luat de alt utilizator din sistem
 - Va fi afisat un mesaj de eroare si se va inchide sesiunea de autentificare
- ❖ Parola nu e recunoscuta
 - Se interzice accesul la fișier
 - Se afiseaza un mesaj de eroare

2.2.1.3. Postconditie

Utilizatorul poate initia transferul de fișiere.



2.2.2 Log out

Pentru a parasii sistemul, utilizator va apela o comanda de iesire. Serverul va inchide conexiunea si va restrictiona accesul catre structura de fișiere a utilizatorului care a parasit sistemul. Daca un utilizator paraseste aplicatia fara sa execute o comanda de iesiere, acesta va fi automat scos din sistem de catre server dupa ce acesta descopera faptul ca utilizatorul nu mai foloseste aplicatia.

2.2.2.1 Preconditie

Utilizatorul trebuie sa foloseasca sistemul.

Serverul trebuie sa fie pornit.

2.2.2.3 Descriere narativa

- ❖ Utilizatorul apeleaza comanda de iesire din sistem.
- ❖ Serverul proceseaza comanda si il scoate din sistem.

Alternativ, utilizatorul nu apeleaza comanda de iesire din sistem, dar inchide aplicatia fara sa anunte serverul. Serverul inchide conexiunea cu utilizatorul.

2.2.2.2 Postconditie

Utilizatorul nu mai are acces la sistem si nu mai poate descarca sau partaja fişiere.



2.2.3 Cautare prin fişiere

Utilizatorul va avea posibilitatea de a cauta prin ierarhia de fişiere a unui alt utilizator sau a tuturor celorlalti utilizatori. Va trimite la server comanda de cautare impreuna cu cuvintele cheie cautate, iar serverul ii va intoarce rezultatul.

2.2.3.1 Preconditie

Utilizatorul trebuie sa fie conectat la server.

Serverul trebuie sa fie pornit pentru a raspunde utilizatorului.

2.2.3.2 Descriere narativa

- ❖ Utilizatorul introduce comanda de cautare impreuna cu cuvintele cheie cautate (eventual si cu numele utilizatorului la care vrea sa caute).
- ❖ Serverul proceseaza cautarea si intoarce un rezultat utilizatorului spunandu-i unde a gasit fişiere cautate.
- ❖ Utilizatorul poate naviga pentru a gasi si descarca fişierul.

2.2.3.3 Postconditie

Utilizatorul primeste o lista cu utilizatorii care au fisierul cautat.

2.2.4 Descarcarea fişierelor

Utilizatorul aplicatiei poate naviga prin structura de fişiere a unui utilizator ales. El va avea la dispozitie comenzi de listare a continutului folderelor si descarcare a fişierelor.

Serverul va trebui sa poata salva starea unui fisier aflat in proces de descarcare si sa continue descarcarea din punctul in care a fost lasat in caz ca utilizatorul opreste conexiunea sau intervine o eroare in comunicatie.

2.2.4.1 Preconditii

Utilizatorul trebuie sa fie conectat cu serverul

Serverul trebuie sa fie pornit.

2.2.4.2 Descriere narativa

- ❖ Utilizatorul apeleaza comanda de lista a continutului unui director al unui alt utilizator.
- ❖ Serverul proceseaza cererea si intoarce raspunsul clientului.
- ❖ Utilizatorul va vedea raspunsul si va putea cere un fisier pentru a fi descarcat.
- ❖ Serverul va initia o conexiune intre cei doi si va facilita descarcarea fisierului.
- ❖ Serverul nu va stoca fisierele utilizatorilor, ci doar ii va ajuta sa comunice, afisandu-le informatii despre ceilalti utilizatori din sistem.
- ❖ Serverul va stoca starea fiecarui fisier in parte pentru a putea relua descarcarea din acelasi punct in care a fost lasata.

2.3. Descrierea cerintelor functionale

Cerinta 1:

Caz de utilizare: Solicitare lista utilizatori, selectare utilizator, solicitare lista fisiere utilizator.

Descriere: Sistemul trebuie sa poata pastra listele de fisiere ale utilizatorilor conectati si fiecare utilizator trebuie sa aiba acces la acestea.

Caz de testare: Utilizatorul trebuie sa primeasca lista de fisiere solicitata

Cerinta 2:

Caz de utilizare: Autentificare pentru descarcarea fisierului, descarcare fisier daca autentificarea a reusit

Descriere: Utilizatorul trebuie sa poata descarca fisiere daca are aceasta permisiune

Caz de testare: Utilizatorul trebuie sa obtina fisierul solicitat sau un mesaj de eroare in care autentificarea a esuat

Cerinta 3:

Caz de utilizare: Solicitare lista utilizatori, selectare utilizator, cautare fisier in lista utilizatorului selectat

Descriere: Posibilitatea de a cauta fisiere intr-o lista mare de fisiere

Caz de testare: Utilizatorul primeste o lista cu fisierele gasite sau un mesaj in cazul in care niciun fisier nu a fost gasit

Cerinta 4:

Caz de utilizare: Solicitare lista utilizatori, cautare utilizator

Descriere: Utilizatorul trebuie sa poata cauta usor alt utilizator

Caz de testare: Returnare lista utilizatori gasiti sau mesaj de eroare in caz ca nu a fost niciun utilizator gasit

3. Arhitectura sistemului

Aplicația va funcționa pe sistemul de operare Linux.

Aplicația se va baza pe modelul client-server, datele utilizatorilor fiind centralizate pe un server. Aplicația se bazează pe prezența unui singur server în sistem. Pot exista mai mulți clienți (utilizatori) care să se conecteze la server.

Pornirea aplicației constă în operația de deschidere a unui executabil care va facilita comenzile implementate. Pentru a avea o comunicație funcțională, serverul va trebui să fie pornit înainte ca utilizatorii sistemului să se conecteze la aplicație.

Conexiunea utilizator-server și utilizator-utilizator va fi făcută peste Internet, nu este nevoie ca utilizatorii și serverul să fie în aceeași rețea locală.

Transferul de fișiere se inițiază printr-o cerere, a unui fișier, adresată server-ului, urmând ca acesta din urmă să faciliteze conexiunea dintre cei doi utilizatori, iar transferul propriu-zis se realizează peer to peer.

4. Cerințele modelării datelor

Nu se va vorbi foarte multe despre modelarea datelor fiindcă nu se va folosi o bază de date.

Datele utilizatorilor vor fi formate din:

- identificator utilizator
- ierarhie de fișiere partajate
- date necesare pentru conectarea la un utilizator pentru inițierea unui transfer de fișiere (peer to peer)

5. Cerințe non-funcționale

5.1 Cerințele interfeței cu utilizatorul

Utilizatorul va interacționa cu aplicația prin linia de comandă. Va putea selecta directoare pentru partajare, le va putea proteja și va putea explora ierarhia de fișiere a altor utilizatori.

5.2 Cerințele de performanță

5.3 Cerințele de disponibilitate și fiabilitate

Întrucât aplicația nu stochează efectiv fișierele serverul care o găzduiește nu va fi deosebit de solicitat (rămâne în urma testării să descoperim pragurile critice), astfel problemele care ar putea apărea ies din sfera responsabilității noastre.

5.4 Cerințele de securitate

Fișierele trebuie să poată fi protejate printr-o parolă cunoscută doar de utilizatorul care le deține.

5.5 Cerințele de administrării sistemului și de mentenanță

Se va putea impune un număr maxim de utilizatori și alte limitări în privirea partajării datelor. Se vor putea face interogări pe listă de utilizatori obținând diferite statistici asupra felului cum

utilizează aplicația. Administratorul poate decide stergerea utilizatorilor din sistem si va avea la dispozitie comenzi pentru a face acest lucru.

Aplicația va loga apariția erorilor, înregistrând tipul și timpul apariției lor.

5.6 Cerințele pregătirii utilizatorului

Aplicația va veni cu un manual care descrie detaliat comenzile acceptate și lucrul cu interfața. Din interfață se pot cere informații limitate despre comenzi(de exemplu parametrii primiți).

6. Considerații de testare

Testarea ar trebui să se facă luând în considerare diferite cazuri de utilizare ale aplicației. Atât corecte, cât și incorecte.

Testarea ar trebui sa implice cazurile in care conexiunea nu poate fi stabilita intre client si server sau intre clienti. Pot aparea erori pe socketii folositi. Vor trebui luate in considerare si simulate aceste cazuri.

In testare, va fi luat in considerare un numar maxim de 512 utilizatori conectati simultan in sistem.

Exemplu teste:

	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7
C1	X	X	X					
C2				X				
C3	X	X	X		X	X		
C 4	X	X	X				X	
.....								

Test 1: Niciun utilizator conectat

Test 2: Exista utilizatori conectati

Test 3: Utilizatorul nu are fisiere partajate

Test 4: Autentificare esuata pentru fisiere

Test 5: Nu au fost gasite fisiere

Test 6: Mai multe fisiere au fost gasite

Test 7: Mai multi utilizatori au fost gasiti

Alte teste

Exemplu search match pentru "cuvant":

cuvant, **cuvant123**, **altcuvant**, **altcuvantgasit**