**Remote monitoring & management**

**pentru distribuții Linux**

* PROIECT PRACTICĂ 2023 -



Std. cap.: Bălăcescu Anamaria

Grupa: C112A

**CUPRINS**

[**Scopul proiectului** 3](#_Toc139019466)

[**Diagrama stagiilor de interacțiune** 3](#_Toc139019467)

[**Set-up client** 4](#_Toc139019468)

[**Prezentarea GUI** 6](#_Toc139019469)

# **Scopul proiectului**

**O rețea la nievlul unei companii implică atât gestionarea calculatoarelor din incinta sediilor companiei, cât și un suport sustenabil pentru gestionarea angajaților care lucrează în sistem remote.**

**Implicațiile acestui sistem de management presupun utilizarea unui server cu acces de nivel administrator la nivelul calculatoarelor puse la dispoziție.**

**Luând în considerare cazul unei companii multi-naționale, aceasta va distribui angajaților calculatoare cu setări predefine care asigură gestionarea aplicațiilor, proceselor și serviciilor la nivel de administrator, precum și conexiunea cu serverul. Vom presupune o conexiune de tip vpn care rulează în background permițând accesul fără limitarea sau interferanța cu acțiunile desfășurate de utilizator.**

**Administratorul de rețea (denumit aici server) va recepționa în prima etapă datele de înregistrare (username, adresa IP) ale clientului și va face schimb de chei ssh cu acesta pentru a stabili o conexiune sustenabilă care nu necesita introducerea parolei.**

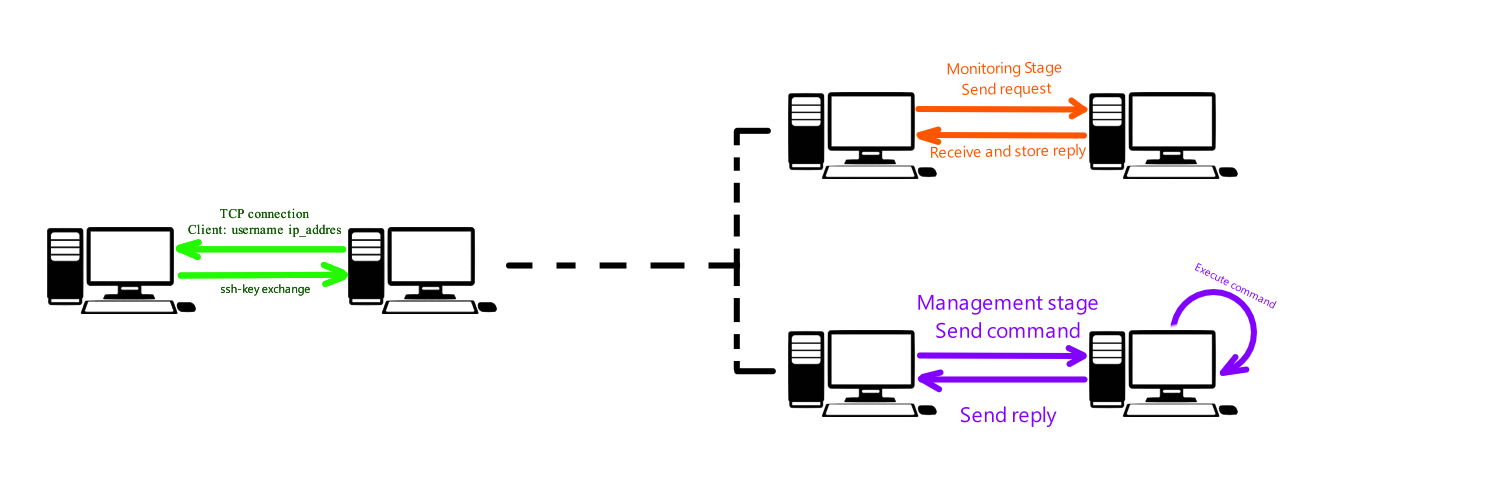
**În cadrul aplicației se vor realiza doua tipuri de acțiuni: monitoring și remote management. În secțiunea de monitoring vom rula automat comenzi care să preia datele despre capacitatea memoriei, dispozitivele hardware, dispozitivele I/O, procese active, temperatura, istoricul conexiunilor ssh, utilizatori și timpul lor de autentificare, precum și informații despre sistem(CPU, GPU, OS version, Host, Uptime, Pachete, Shell, rezoluție, terminal, memorie, etc)**

**Pentru secțiunea de remote management vor putea realiza la alegere acțiuni de oprie/pornire servicii/procese, copiere de fișiere, instalare de aplicații/servicii în funcție de distribuția de Linux pe care lucram (am tratat distribuțiile Debian, RedHat, openSSUE, Arch). Managementul în sistem remote a devenit o componentă vitală în funcționalitatea unei companii considerând respectarea regulilor de etică profesională, a legislației, precum și distribuirea de pachete necesare în finalizarea task-urilor pentru fiecare angajat în parte.**

**Pentru comoditatea și accesibilitatea în vederea prelucrării, am adăugat în structura GUI a aplicației noastre opțunea de a vizualiza toti utilizatorii rețelei care au fost conectați până la momentul curent și implicit, după selectarea utilizatorului dorit, afișarea fișierelor cu informațiile obținute în etapa de monitorizare.**

**În ceea ce privește interacțiunea propriu-zisa cu aplicația, am realizat o interfață GUI-firendly cu grafică prietenoasă pentru a facilita acomodarea cu utilizarea ei și a întrerupe presiunea resimțită de angajați în mediul de lucru.**

# **Diagrama stagiilor de interacțiune**



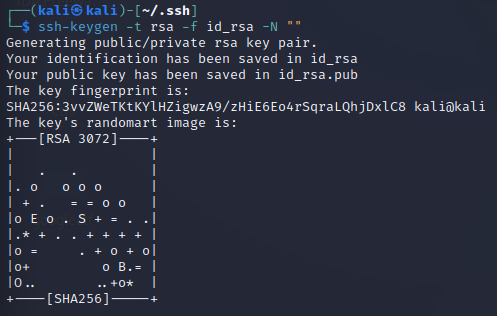
# **Set-up client**

La nivelul companiei, se setează permisiunile și configurările necesare penrtu gestionarea clientului înainte de predarea dispozitivului spre folosire.

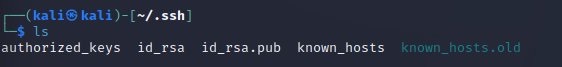
Această setare prezintă mai multe etape printre care numărăm și configurarea setărilor de accesibilitate ssh (aici vom utiliza o configurare ssh fără parolă pentru comoditate și eficiență considerând că folosim un VPN pentru a satisface și cerințele de securitate), percum și setarea drepturilor de administrator pentru utilizatorul în cauză.

Etapa 1 – starea configurațiilor ssh.

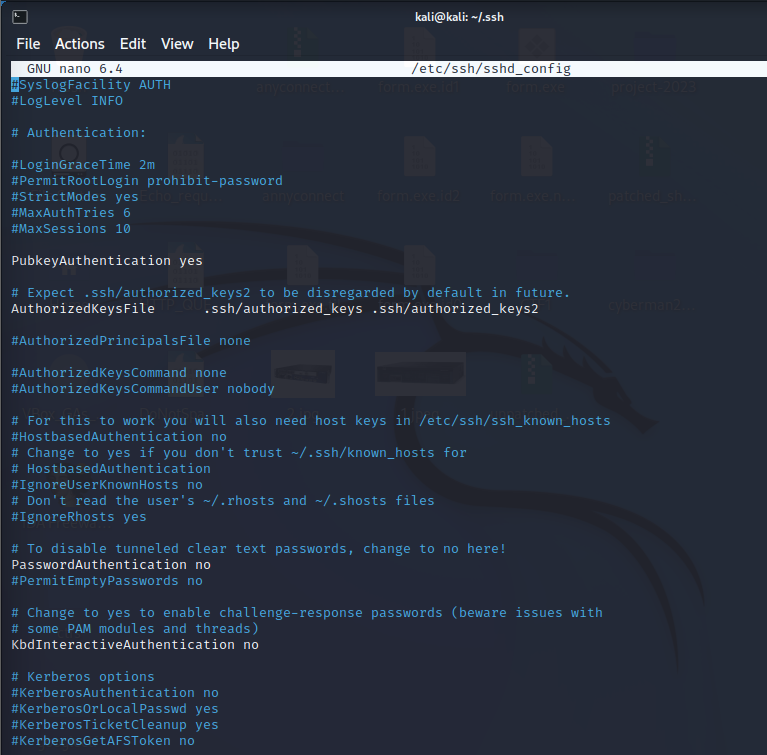
La nivelul utilizatorului, în directorul .ssh se generează perechea de chei publică/privată prin apelarea din CLI a comenzii: ssh-keygen -t rsa -f id\_rsa -N ""



La nivelul directorului .ssh vom avea create cheile necesare în fișierele id\_rsa, respectiv id\_rsa.pub, precum și fișierul de chei autorizate authorized\_keys.



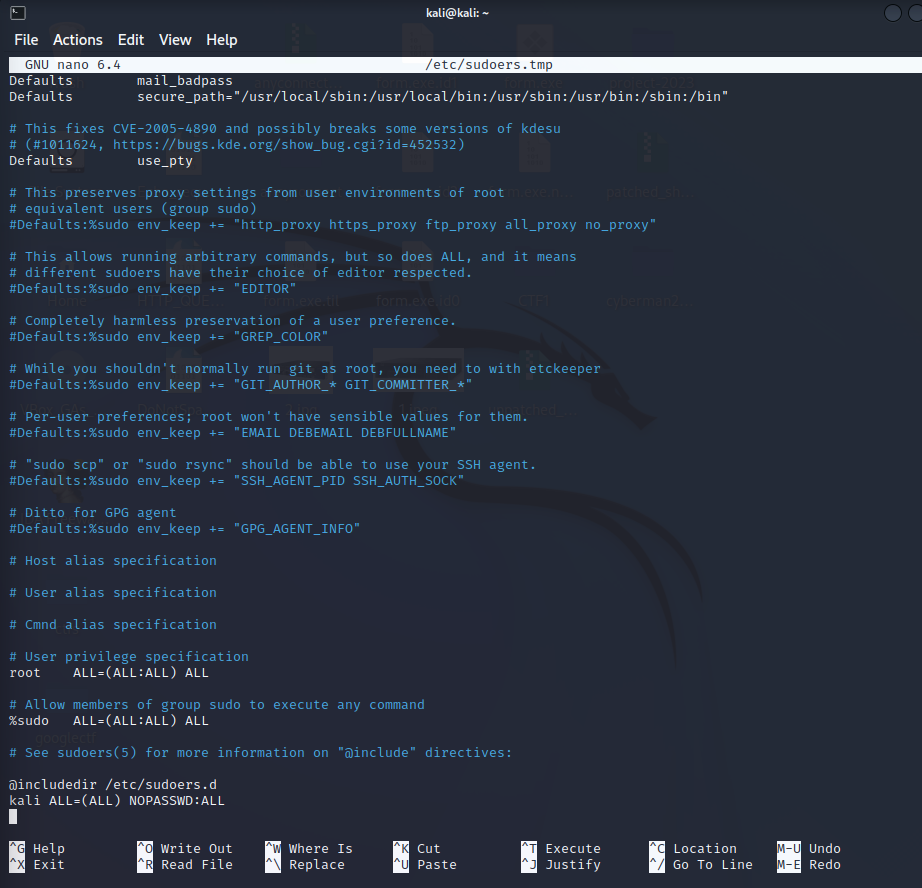
Pentru a realiza conexiunea ssh, avem nevoie să configurăm setările ssh gasite in fișierul /etc/ssh/sshd\_config. Aici vom apela comanda **sudo nano /etc/ssh/sshd\_config** și vom verifica să avem asigurate specificațiile: PubkeyAuthentication yes, AuthorizedKeysFile .ssh/authorized\_keys .ssh/authorized\_keys2, PasswordAuthentication no.



Facem setările și pentru server intrucât vom avea nevoie de cheia sa privată pentru a accesa clientul.

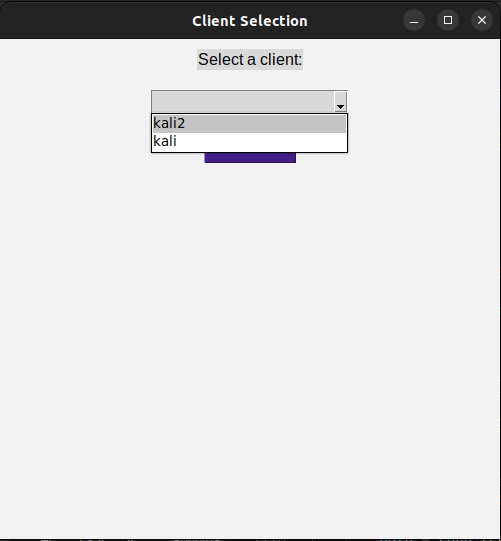
Etapa 2 presupune setarea drepturilor sudo pentru utilizatorul nostru.

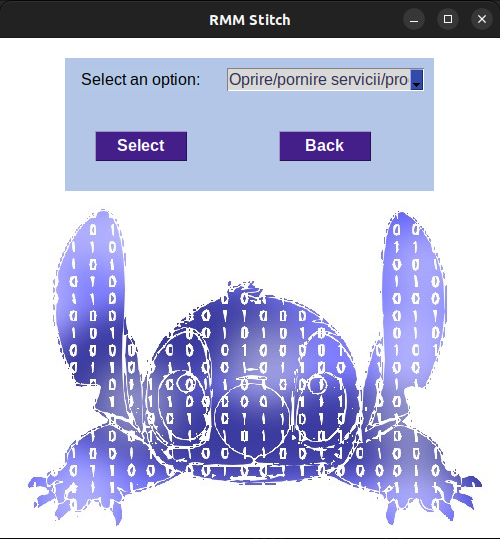
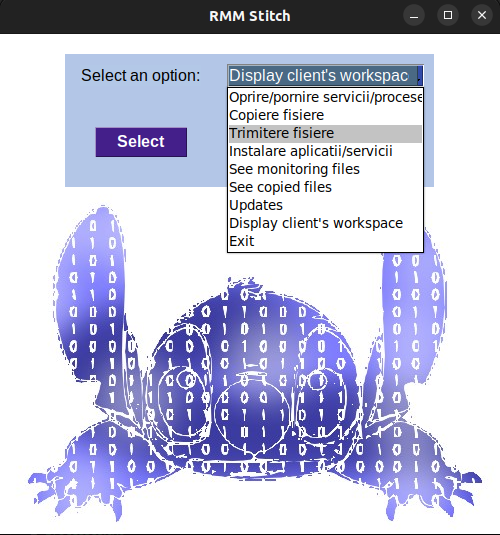
Prima comanda este **sudo su** care ne permite înregistrarea ca root (administrator absolut) și ulterior modificarea fișierului visudo prin apelarea comenzii **sudo visudo**. Se va deschide o nouă fereastră cu conținutul aferent. Mergem la finalul acestuia și adaugăm numele utilizatorului nostru cu permisiunile dorite.

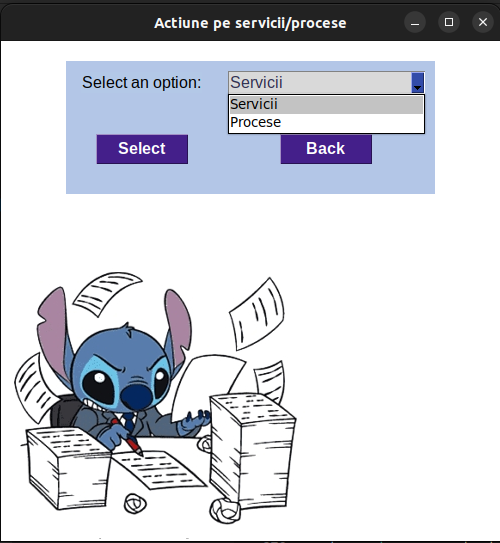
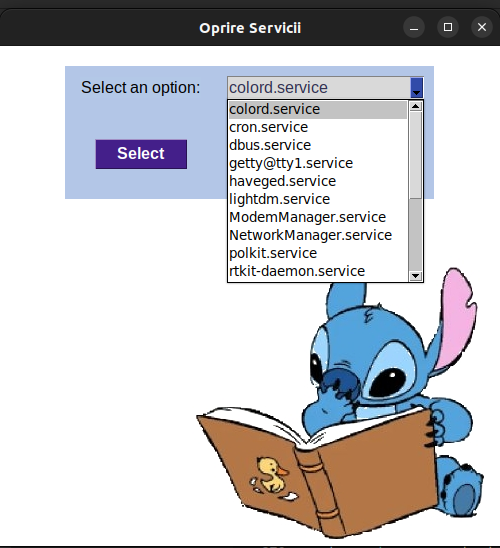
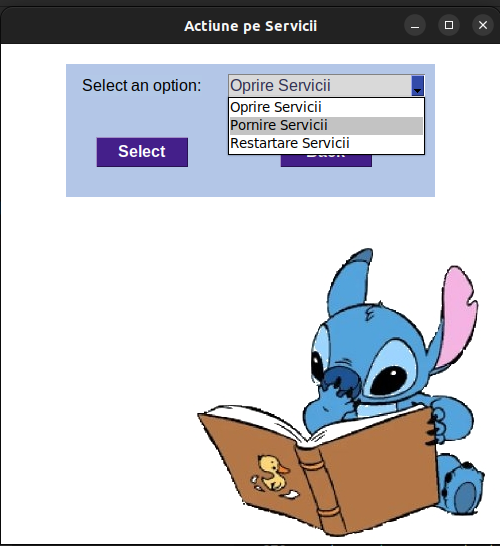
****

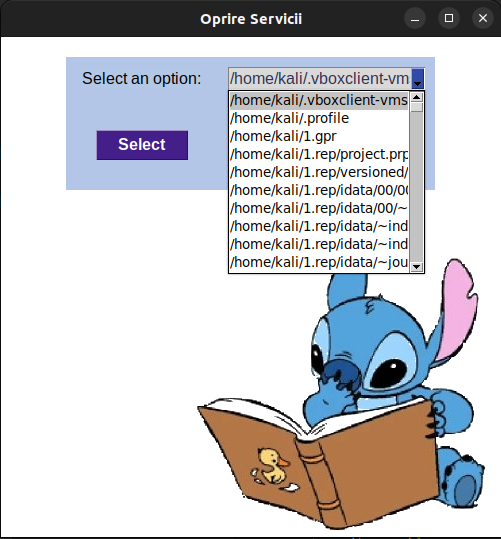
Set-up-ul nostru este complet.

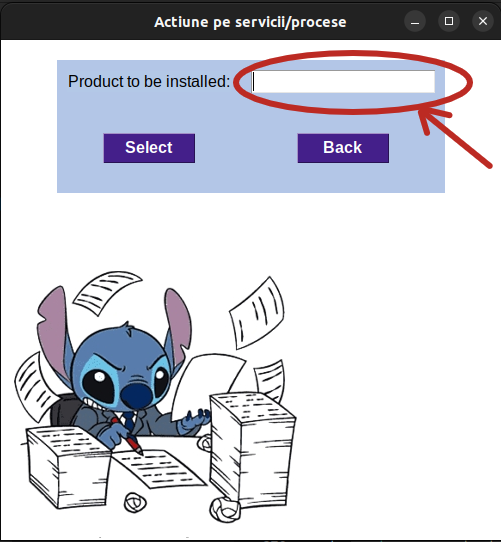
# **Prezentarea GUI**

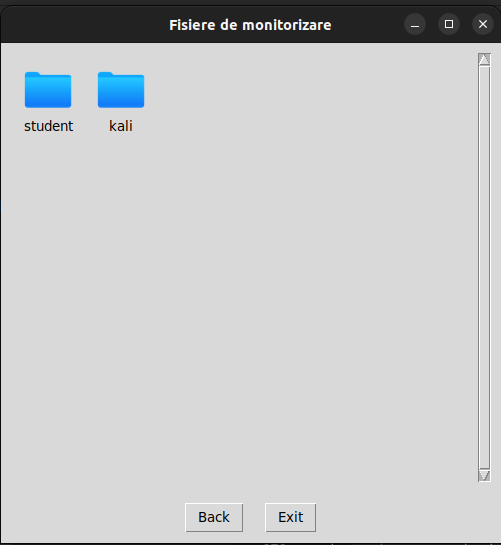
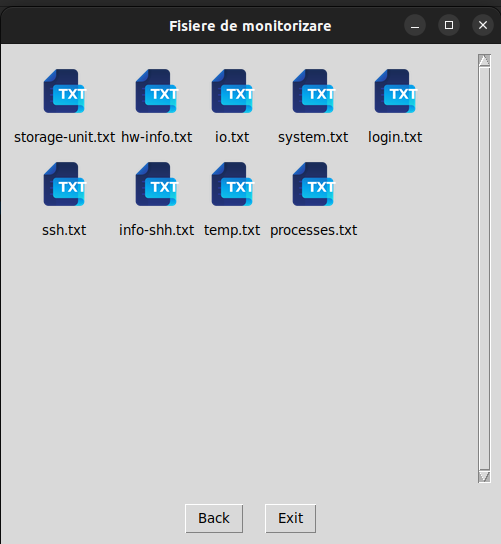
 Pagina principală conține informații despre clienții conectați la rețeaua noastră. După selectionarea clientului dorit din slide bar vom fi direcționați către pagina de pornire a aplicației de monitorizare.

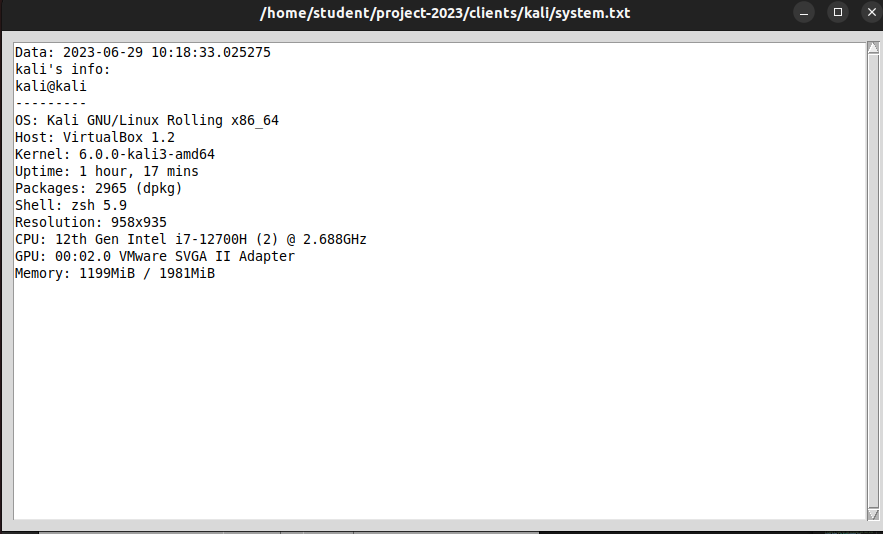
**** Pagina noastră de pornire vine cu un slide bar pentru selectarea acțiunii pe care vrem sa o desfășurăm:

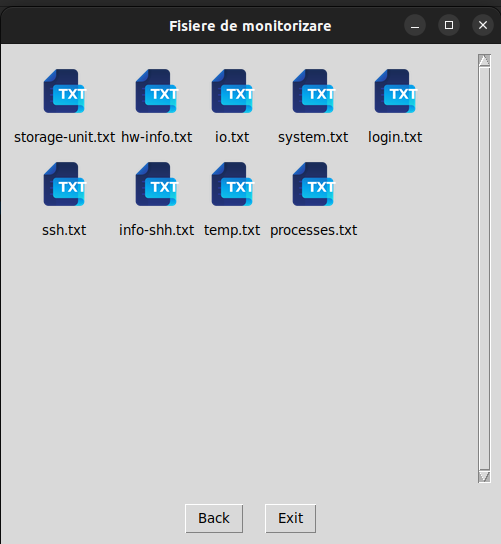
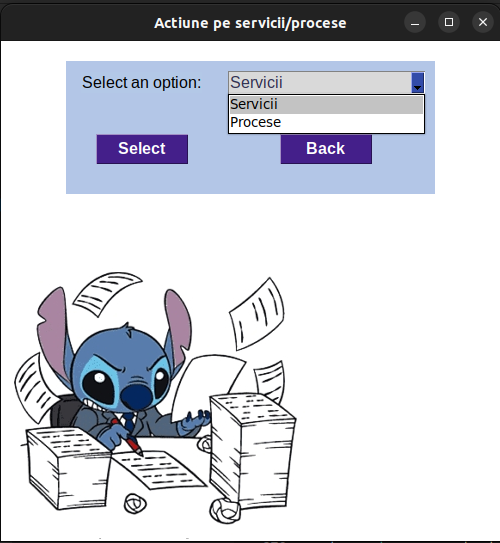
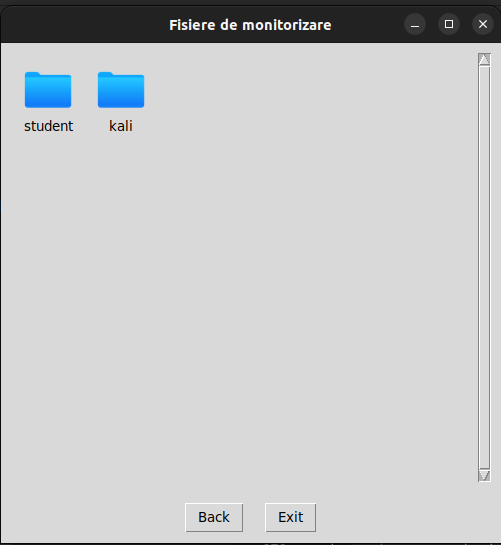
Pentru fiecare opțiune vor fi redirecționați către ecranul corespunzător. În cazul opțiunilor cu mai multe tipuri de acțiuni furnizate vom fi redirecționați către un ecran de selecție a celei corespunătoare cerințelor noastre (ex.: opțiunea 1)

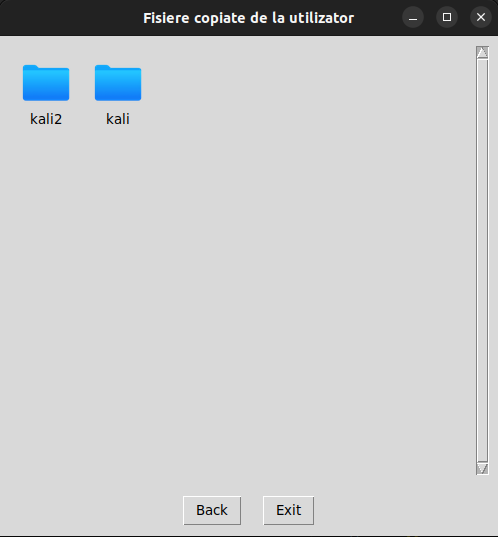
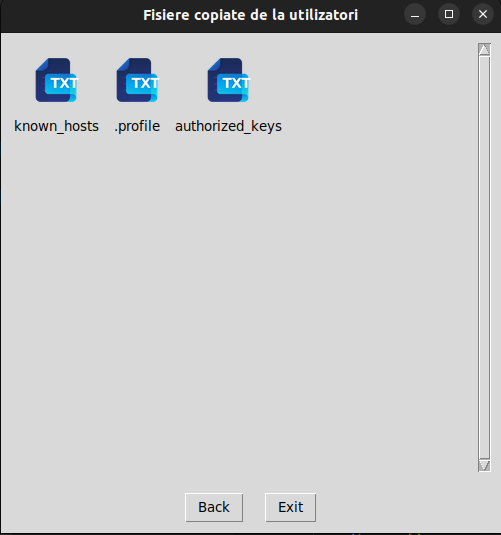
 Pentru opțiunea 2 – Copiere fișiere, ni se vor pune la dispoziție toate fișierele clientului nostru din care putem alege ce fișier se va copia în folderul corespunzător utilizatorului din directorul **„copied\_files”** de la nivelul server-ului nostru.

În cazul opținulor care necesită input (ex.: opțiunea 3) vom deschise un ecran care ne va cere să introducem de la tastatură un stri.

****În ceea ce privește informațiile obținute în etapa de monitorizare, acestea pot fi observate prin selectarea opțiunii 4. Vom fi redirecționați într-o fereastră în care vom vedea directoarele specific fiecărui client în parte. Prin selectarea celui dorit vom deschide o nouă fereastră care ne va prezenta toate fișierele obținute. De aici putem deschide orice fișier dorit care va apărea printr-o fereastră de tip pop-up din care putem prelua datele dorite. În scopuri de securitate aceste fișiere vor fi updatate periodic, iar data și ora ultimului update vor fi afișate în partea de sus a fișierului.

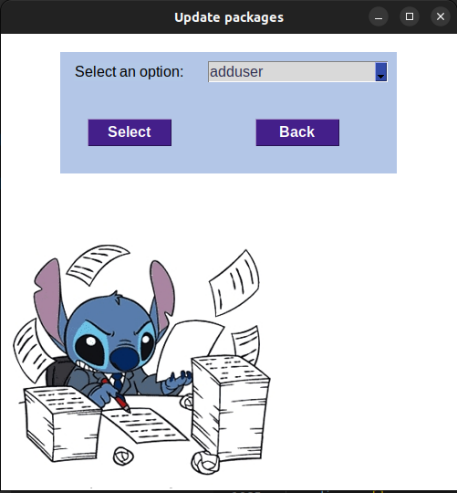
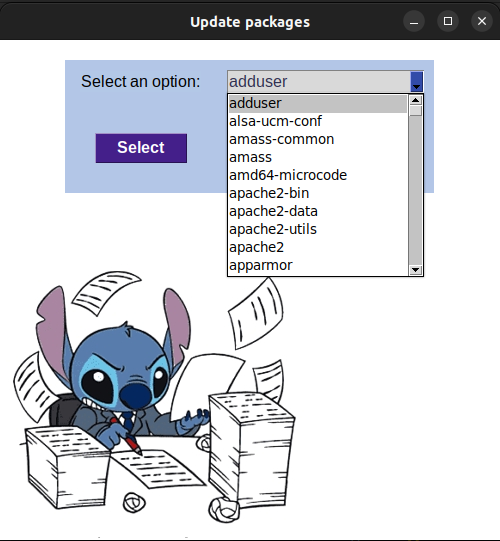


**** Pentru comoditate și utilitate am adăugat fiecărui ecran butoane de închidere a programului și de direcționare către fereastra precedent și unde am crezut ca este necesar un scrollbar pentru a putea ajunge mai rapid la informația dorită.

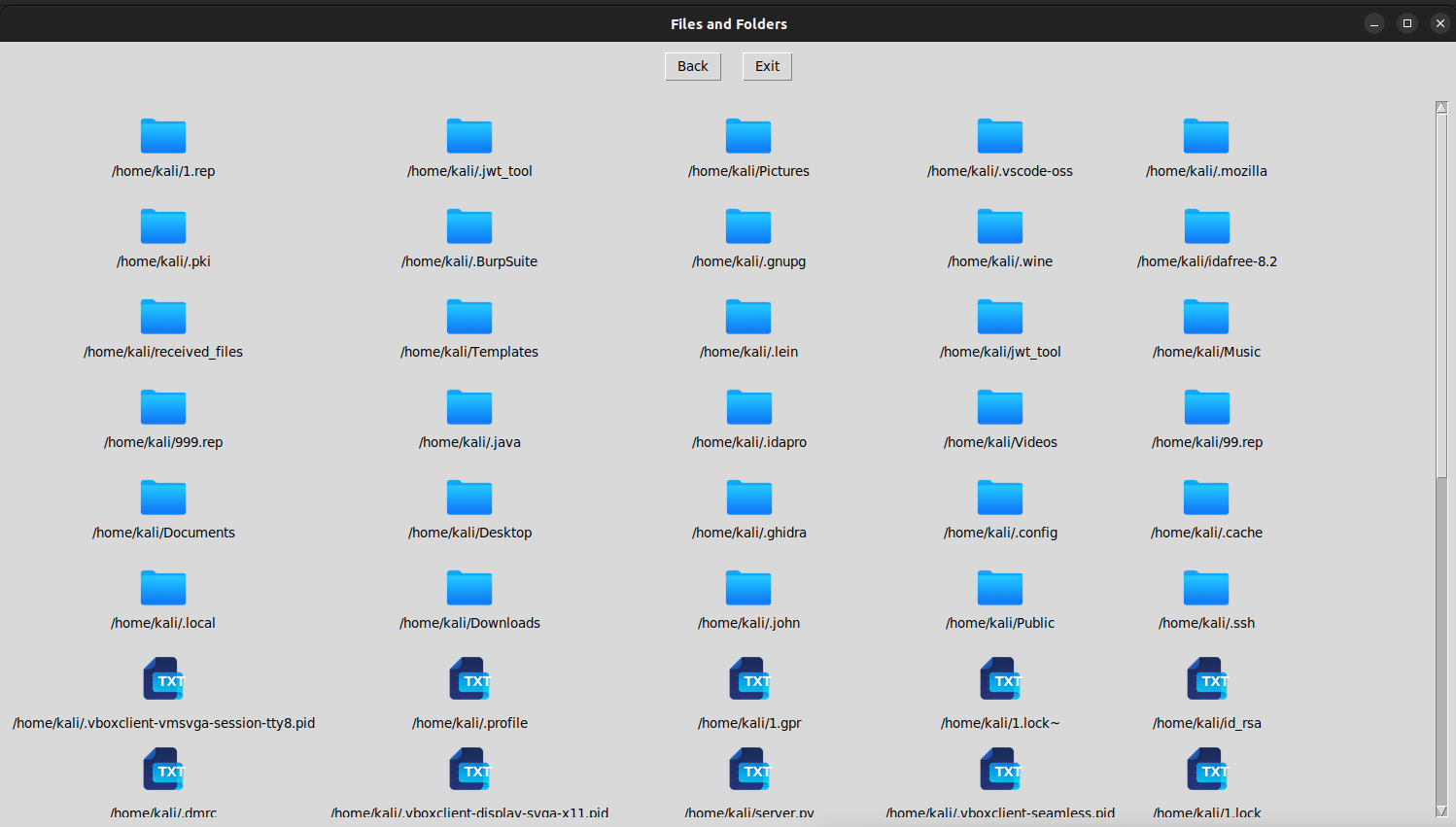
 Asemănător am gândit și opțiunea de vizualizare a fișierelor copiate de la clienții noștri



În scopul menținerii unui angajat la ordinea de zi cu programele necesare, am creat o opțiune de update a serviciilor și aplicațiilor disponibile. Prin selectarea opțiunii **„Update”** din ecranul meniului vom fi direcționați către ecranul de afișare a pachetelor care pot primii update.



O funcționalitate mai interesantă a aplicației este posibilitatea de a vedea spațiul de lucru al utilizatorului nostru în timp real.



Putem de asemenea să navigăm și să accesăm directoare și fișiere pentru a verifica și conținutul lor. Vom deschide în continuare directorul **„/home/kali/received\_files”** care este de asemnea directorul folosit în trimiterea fișierelor de la server către client (ca o acțiune care să faciliteze partajarea de informație între companie și angajați, spre exemplu) și fișierul **„server.py”** pentru verificarea fișierelor.

