Operating Systems – CSE x61

Simple Shell

By students

Ali Mohamed Taima Hummus
AbdelRahman Ahmed Hassan AlSindiony
Mohamed Eid Mohamed Ali

Alexandria University

Faculty of Engineering

Spring 2022

Program Overview

The program is written In C programming Language for use on UNIX/Linux operating systems (OSs) as a simple shell. It's implemented by splitting the algorithm into multiple functions as a procedural programming method, following is the algorithm, code description, results for sample runs, and processes hierarchy.

Program Algorithm

1- Get input from the user and store it as string (buf).

2- Check (buf) for the special cases, empty input, exit, or help, and handle their occurrences.

3- Convert the user string (buf) to an array of strings (command) with the command at the first index and the parameters follow it.

4- Check (command) for cd command and handle its occurrence.

5- Fork a new child process to handle the user required process and report the log file.

6- Pass the (command) array to execvp function to proceed.

7- Wait the process until finishing its job and terminate it only if & doesn't exist at the end of the command, otherwise, skip waiting this process.

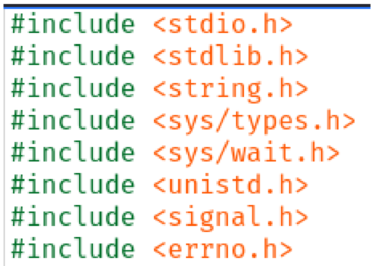
8- Handle (SIGCHLD) signal, recording the termination of a process to the log file.

9- In steps 7, 8, keep record of zombie processes, those who have been skipped without waiting and were terminated later.

10- Bury Zombie processes -if they existed- before prompting for a new input.

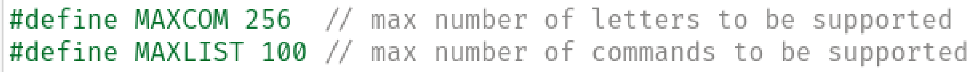
Code Description

- As usual, the C code starts with the libraries needed



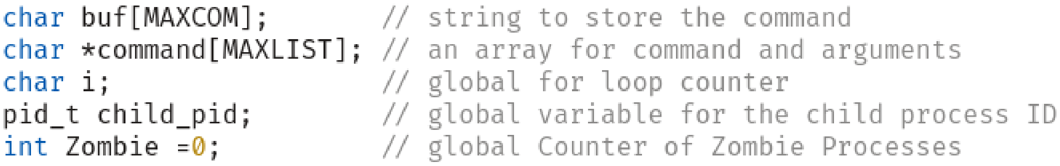
```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
#include <signal.h>
#include <errno.h>
```

- Next, we define the global constants used in the program



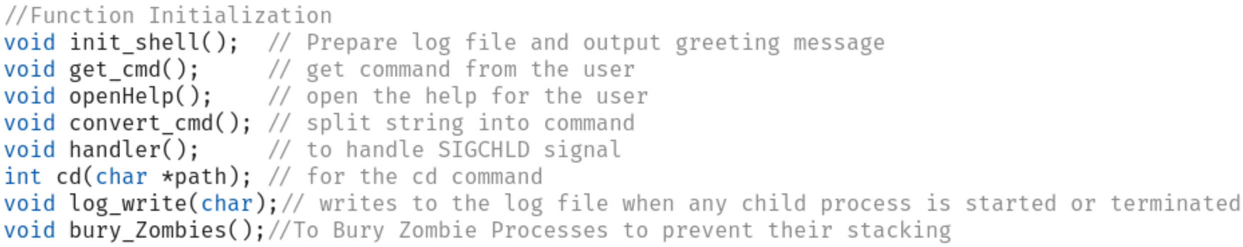
```c
#define MAXCOM 256  // max number of letters to be supported
#define MAXLIST 100 // max number of commands to be supported
```

- Also, we define the global variables



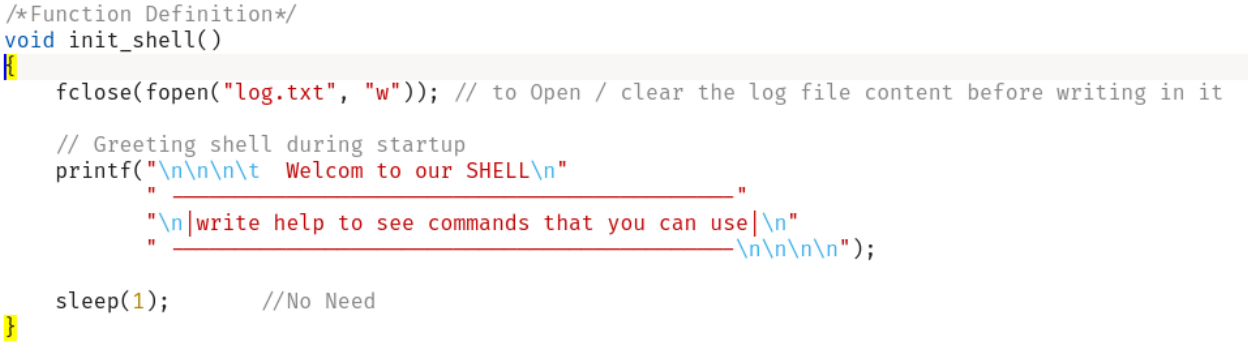
```c
char buf[MAXCOM];        // string to store the command
char *command[MAXLIST]; // an array for command and arguments
char i;                  // global for loop counter
pid_t child_pid;         // global variable for the child process ID
int Zombie =0;           // global Counter of Zombie Processes
```

- Then, we initialize the functions that will be used later



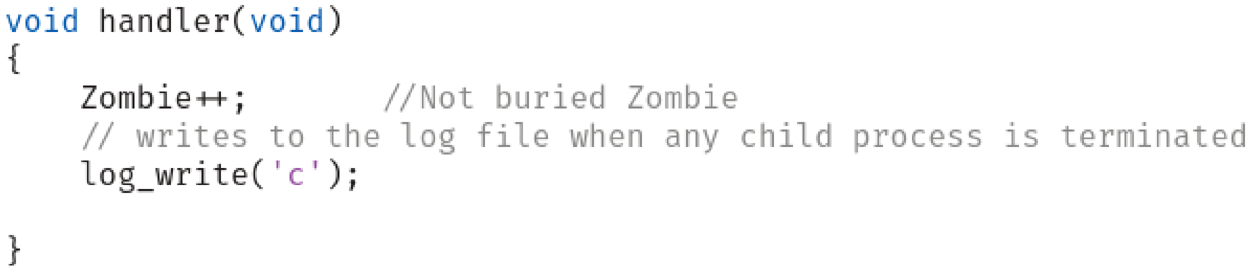
```c
//Function Initialization
void init_shell();  // Prepare log file and output greeting message
void get_cmd();     // get command from the user
void openHelp();    // open the help for the user
void convert_cmd(); // split string into command
void handler();     // to handle SIGCHLD signal
int cd(char *path); // for the cd command
void log_write(char);// writes to the log file when any child process is started or terminated
void bury_Zombies();//To Bury Zombie Processes to prevent their stacking
```

- The main function calls another function called init_shell which creates or clears the log file before writing on it to avoid appending on garbage data, it also prints a well-organized message to welcome the user.

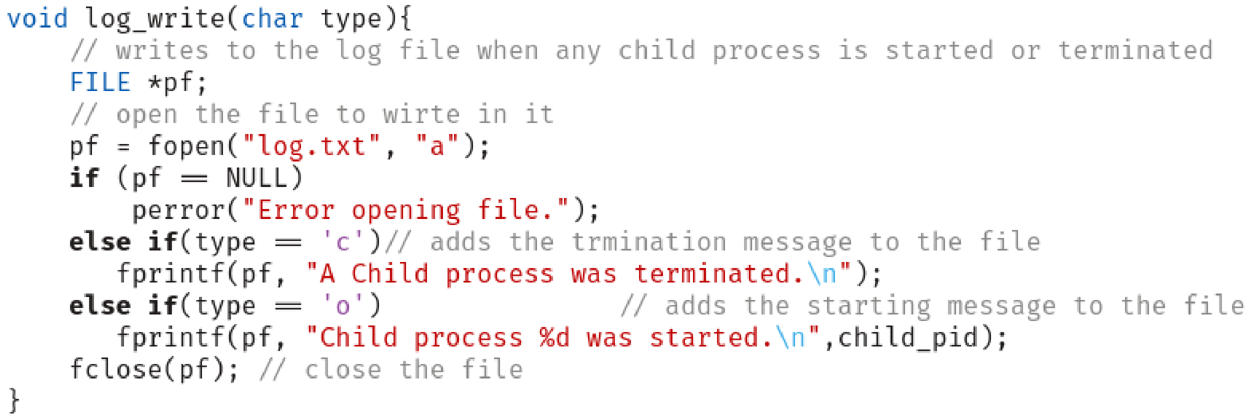


```c
/*Function Definition*/
void init_shell()
{
    fclose(fopen("log.txt", "w")); // to Open / clear the log file content before writing in it

    // Greeting shell during startup
    printf("\n\n\n\t  Welcom to our SHELL\n"
           "  ____________________________________________ "
           "\n|write help to see commands that you can use|\n"
           "  ____________________________________________\n\n\n\n");

    sleep(1);       //No Need
}
```

- Then the main function prepares for receiving SIGCHLD signal and calls a function called handler which in turn calls another function called log_write to write into the log file that a process was terminated.



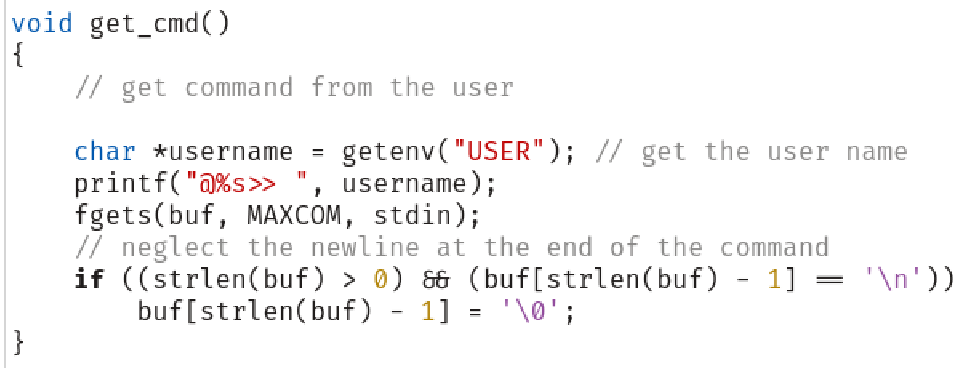
```
void handler(void)
{
     Zombie++;        //Not buried Zombie
     // writes to the log file when any child process is terminated
     log_write('c');

}
```



```
void log_write(char type){
    // writes to the log file when any child process is started or terminated
    FILE *pf;
    // open the file to wirte in it
    pf = fopen("log.txt", "a");
    if (pf == NULL)
        perror("Error opening file.");
    else if(type == 'c')// adds the trmination message to the file
       fprintf(pf, "A Child process was terminated.\n");
    else if(type == 'o')               // adds the starting message to the file
       fprintf(pf, "Child process %d was started.\n",child_pid);
    fclose(pf); // close the file
}
```

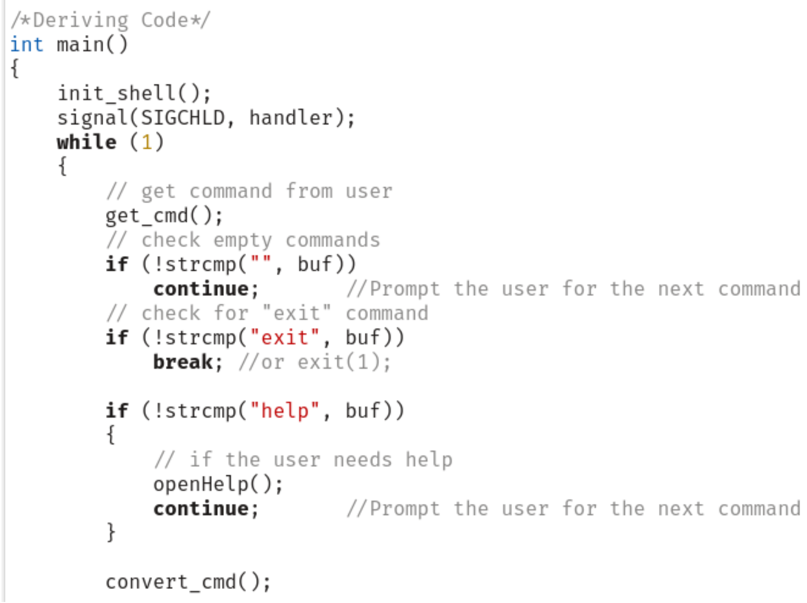
Note: each time a process terminates, we increment the zombie counter, in contrast, we decrement it after each *wait* call, so, the variable Zombie represents the number of available zombie processes

- The main function calls get_cmd function which takes string input from the user, by taking data of size MAXCOM from stdin file to buf array, also the function prints the user environment variable before each prompt.



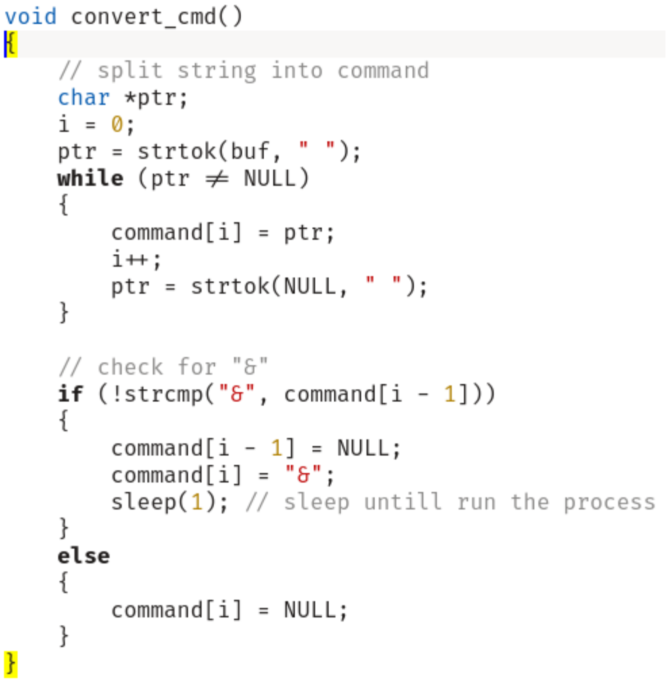
```
void get_cmd()
{
    // get command from the user

    char *username = getenv("USER"); // get the user name
    printf("@%s>> ", username);
    fgets(buf, MAXCOM, stdin);
    // neglect the newline at the end of the command
    if ((strlen(buf) > 0) && (buf[strlen(buf) - 1] == '\n'))
        buf[strlen(buf) - 1] = '\0';
}
```

- Three conditional statements now appear to implement step 2 in the algorithm for checking empty input, exit, or help. The latter calls a function to print the available commands to the user.



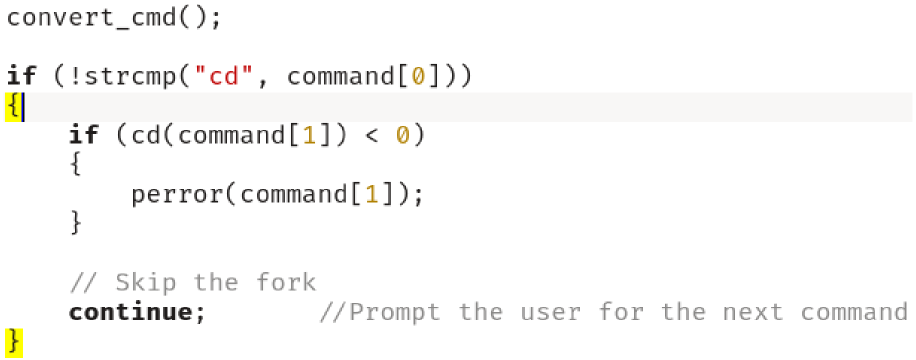
```
/*Deriving Code*/
int main()
{
    init_shell();
    signal(SIGCHLD, handler);
    while (1)
    {
        // get command from user
        get_cmd();
        // check empty commands
        if (!strcmp("", buf))
            continue;       //Prompt the user for the next command
        // check for "exit" command
        if (!strcmp("exit", buf))
            break; //or exit(1);

        if (!strcmp("help", buf))
        {
            // if the user needs help
            openHelp();
            continue;       //Prompt the user for the next command
        }

        convert_cmd();
```

- Now, another function *convert_cmd* call appears to convert the input stored at variable buf to the desired form of command to be passed to *execvp* function later, which is step 3 at the algorithm.



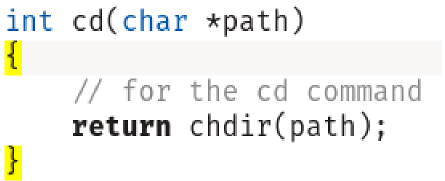
```
void convert_cmd()
{
    // split string into command
    char *ptr;
    i = 0;
    ptr = strtok(buf, " ");
    while (ptr ≠ NULL)
    {
        command[i] = ptr;
        i++;
        ptr = strtok(NULL, " ");
    }

    // check for "&"
    if (!strcmp("&", command[i - 1]))
    {
        command[i - 1] = NULL;
        command[i] = "&";
        sleep(1); // sleep untill run the process
    }
    else
    {
        command[i] = NULL;
    }
}
```

- Now, we implement step 5 at the algorithm, checking the special condition for *cd* command and handling it using a user defined function called *cd* with the path as the parameter at index 1 of the command array.



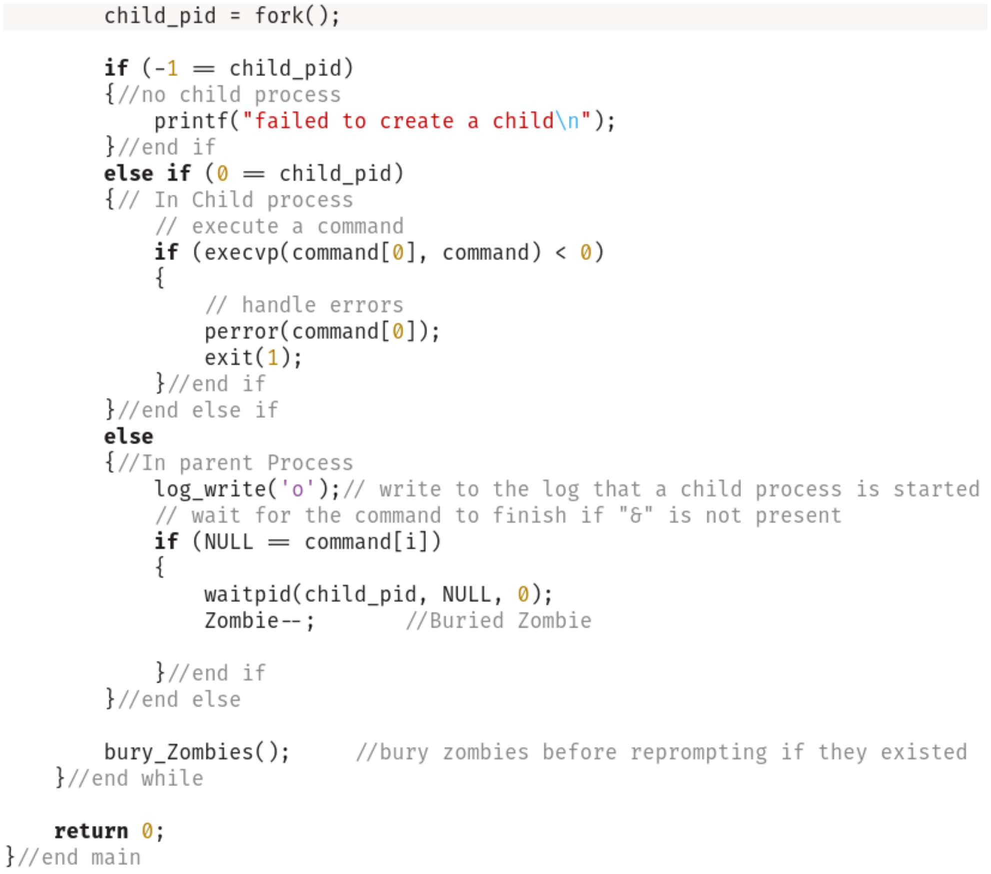
```
convert_cmd();

if (!strcmp("cd", command[0]))
{
    if (cd(command[1]) < 0)
    {
        perror(command[1]);
    }

    // Skip the fork
    continue;        //Prompt the user for the next command
}
```

cd is defined as:



```
int cd(char *path)
{
    // for the cd command
    return chdir(path);
}
```

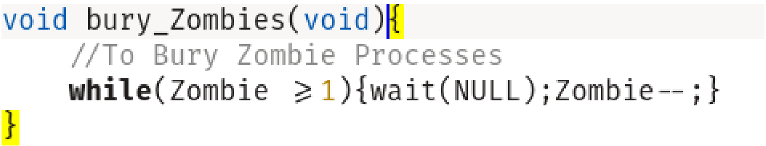
- Implementing step 5, we fork a new child process



```
        child_pid = fork();

        if (-1 == child_pid)
        {//no child process
            printf("failed to create a child\n");
        }//end if
        else if (0 == child_pid)
        {// In Child process
            // execute a command
            if (execvp(command[0], command) < 0)
            {
                // handle errors
                perror(command[0]);
                exit(1);
            }//end if
        }//end else if
        else
        {//In parent Process
            log_write('o');// write to the log that a child process is started
            // wait for the command to finish if "&" is not present
            if (NULL == command[i])
            {
                waitpid(child_pid, NULL, 0);
                Zombie--;       //Buried Zombie

            }//end if
        }//end else

        bury_Zombies();     //bury zombies before reprompting if they existed
    }//end while

    return 0;
}//end main
```

- Forking a new process we face three conditions, First, no process initialized, where the program prints an error to the user with that fact. Second, the child process is running, where the program passes the commands to be executed to the system call function *execvp*, finally the program issues *exit* system call to terminate the process after its execution. Third, the parent process is running, where it calls the function *log_write* to write that a process has started, then it waits for the child process to terminate if and only if the & character didn’t exist, otherwise it is skipped letting the process run at the background.

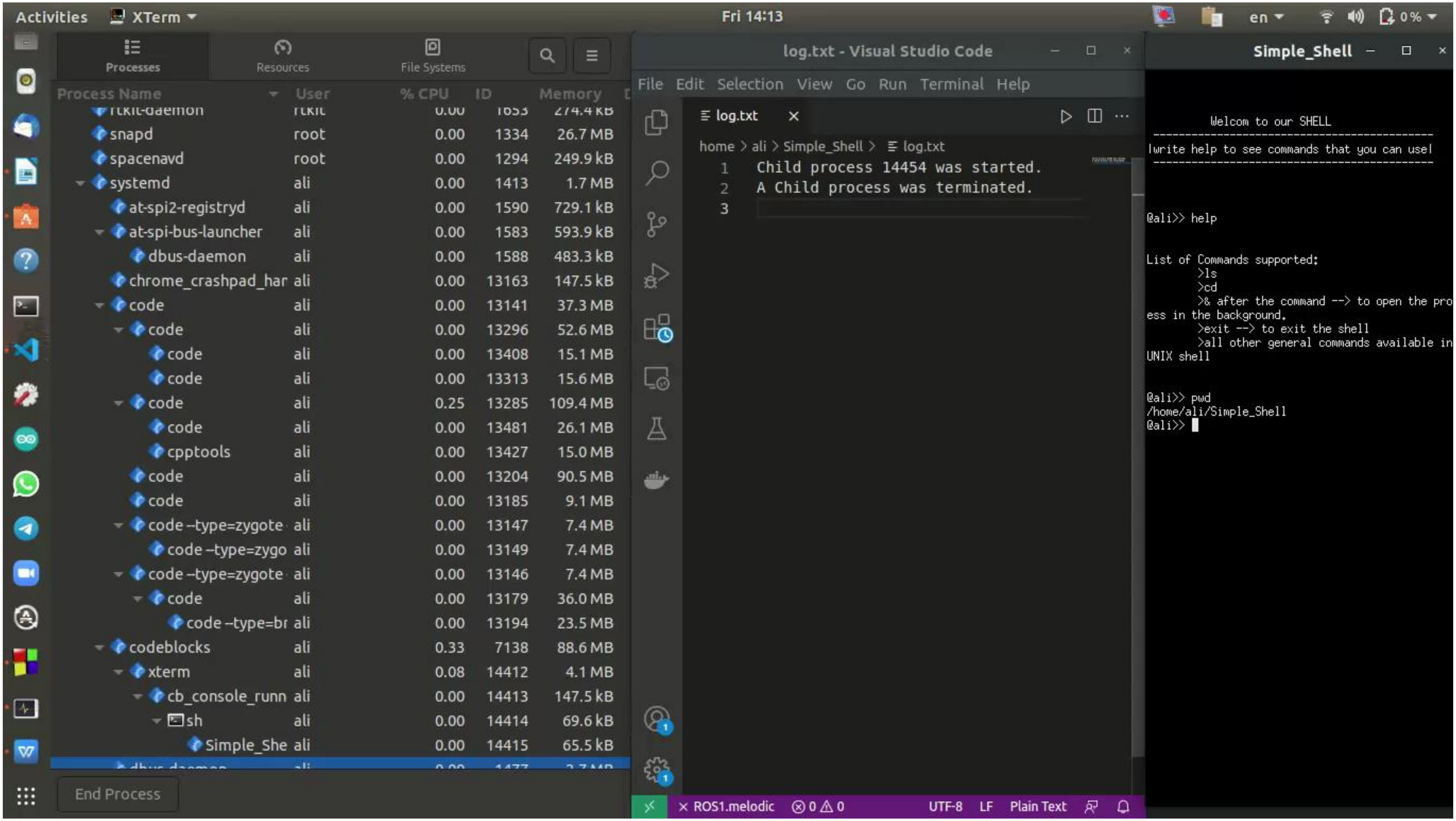
- Finally, the program buries any zombie process before prompting again using the function call *bury_Zombies* , following is its defining code

  ```
  void bury_Zombies(void){
      //To Bury Zombie Processes
      while(Zombie ≥1){wait(NULL);Zombie--;}
  }
  ```

  with the variable *Zombie* equals the number of backgrounds terminated processes.

Results and Sample Runs

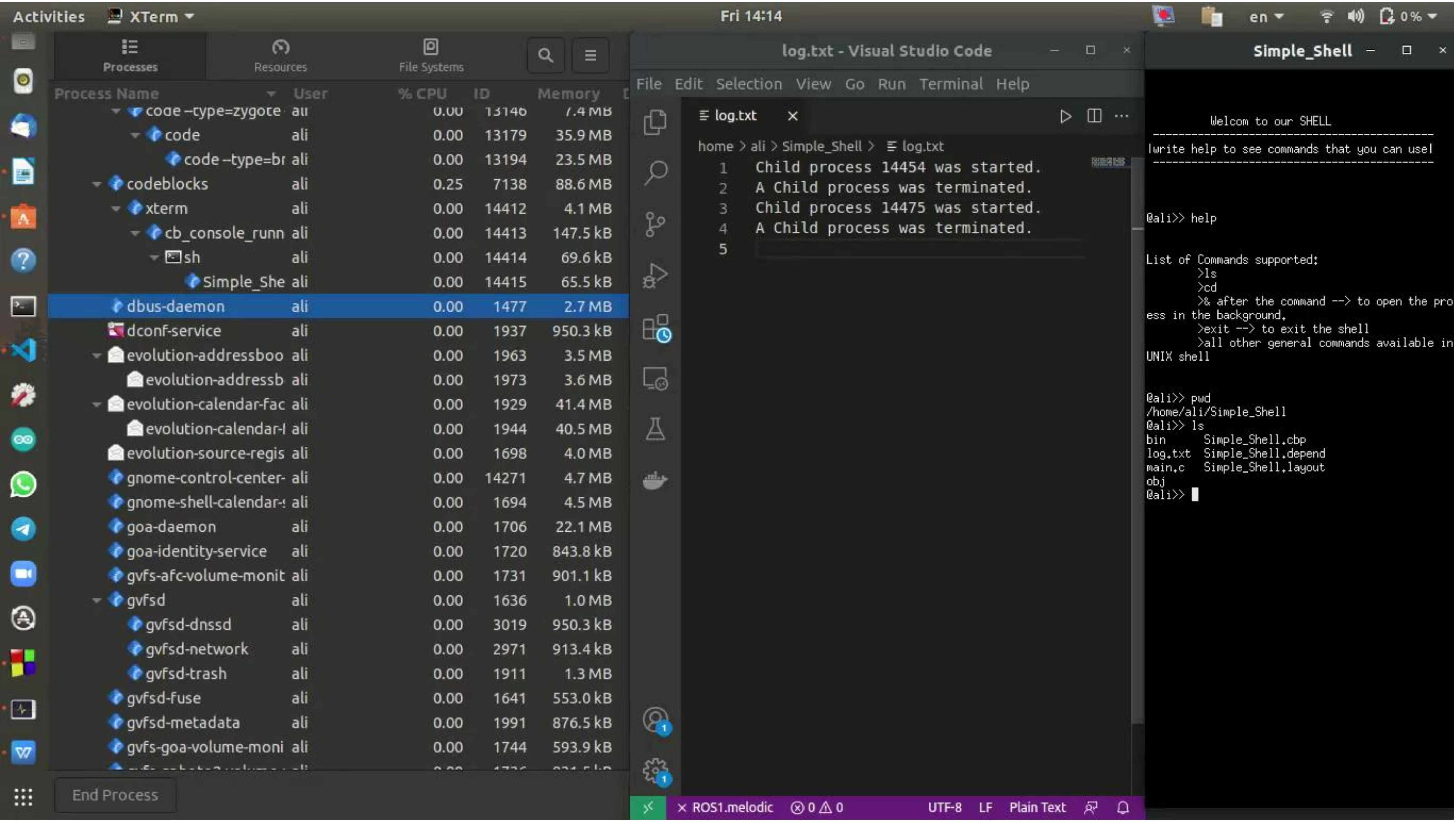


Process Name	User	% CPU	ID	Memory
rtkit-daemon	rtkit	0.00	1653	274.4 kB
snapd	root	0.00	1334	26.7 MB
spacenavd	root	0.00	1294	249.9 kB
systemd	ali	0.00	1413	1.7 MB
at-spi2-registryd	ali	0.00	1590	729.1 kB
at-spi-bus-launcher	ali	0.00	1583	593.9 kB
dbus-daemon	ali	0.00	1588	483.3 kB
chrome_crashpad_har	ali	0.00	13163	147.5 kB
code	ali	0.00	13141	37.3 MB
code	ali	0.00	13296	52.6 MB
code	ali	0.00	13408	15.1 MB
code	ali	0.00	13313	15.6 MB
code	ali	0.25	13285	109.4 MB
code	ali	0.00	13481	26.1 MB
cpptools	ali	0.00	13427	15.0 MB
code	ali	0.00	13204	90.5 MB
code	ali	0.00	13185	9.1 MB
code --type=zygote	ali	0.00	13147	7.4 MB
code --type=zygo	ali	0.00	13149	7.4 MB
code --type=zygote	ali	0.00	13146	7.4 MB
code	ali	0.00	13179	36.0 MB
code --type=br	ali	0.00	13194	23.5 MB
codeblocks	ali	0.33	7138	88.6 MB
xterm	ali	0.08	14412	4.1 MB
cb_console_runn	ali	0.00	14413	147.5 kB
sh	ali	0.00	14414	69.6 kB
Simple_She	ali	0.00	14415	65.5 kB
dbus-daemon	ali	0.00	1477	2.7 MB

```
Child process 14454 was started.
A Child process was terminated.

```

```
          Welcom to our SHELL
 ---------------------------------------------
|write help to see commands that you can use|
 ---------------------------------------------



@ali>> help


List of Commands supported:
        >ls
        >cd
        >& after the command --> to open the pro
ess in the background.
        >exit --> to exit the shell
        >all other general commands available in
UNIX shell


@ali>> pwd
/home/ali/Simple_Shell
@ali>> 
```

>>> help, then >>>pwd

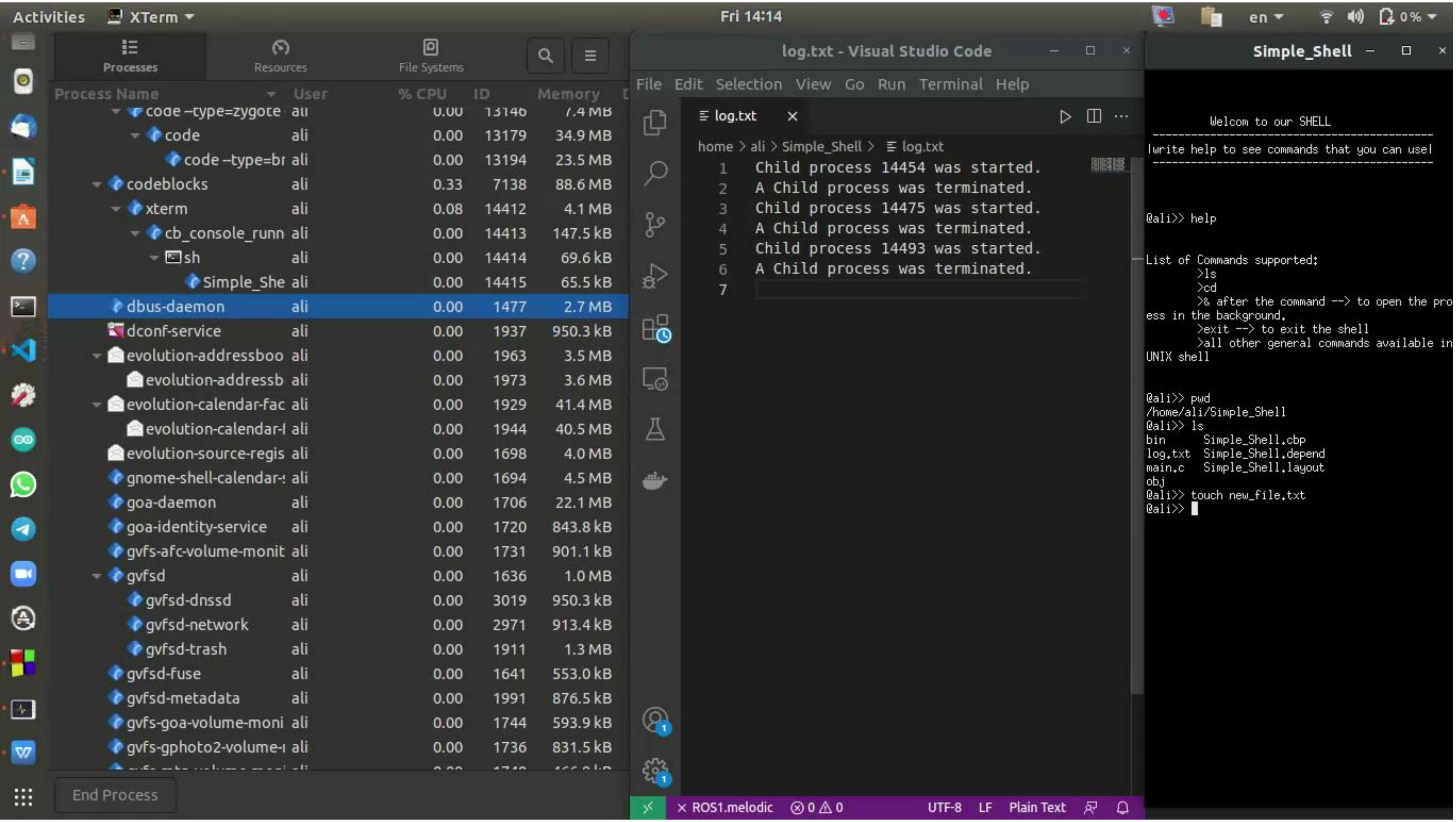


Process Name	User	% CPU	ID	Memory
code --type=zygote	ali	0.00	13146	7.4 MB
code	ali	0.00	13179	35.9 MB
code --type=br	ali	0.00	13194	23.5 MB
codeblocks	ali	0.25	7138	88.6 MB
xterm	ali	0.00	14412	4.1 MB
cb_console_runn	ali	0.00	14413	147.5 kB
sh	ali	0.00	14414	69.6 kB
Simple_She	ali	0.00	14415	65.5 kB
dbus-daemon	ali	0.00	1477	2.7 MB
dconf-service	ali	0.00	1937	950.3 kB
evolution-addressboo	ali	0.00	1963	3.5 MB
evolution-addressb	ali	0.00	1973	3.6 MB
evolution-calendar-fac	ali	0.00	1929	41.4 MB
evolution-calendar-l	ali	0.00	1944	40.5 MB
evolution-source-regis	ali	0.00	1698	4.0 MB
gnome-control-center-	ali	0.00	14271	4.7 MB
gnome-shell-calendar-	ali	0.00	1694	4.5 MB
goa-daemon	ali	0.00	1706	22.1 MB
goa-identity-service	ali	0.00	1720	843.8 kB
gvfs-afc-volume-monit	ali	0.00	1731	901.1 kB
gvfsd	ali	0.00	1636	1.0 MB
gvfsd-dnssd	ali	0.00	3019	950.3 kB
gvfsd-network	ali	0.00	2971	913.4 kB
gvfsd-trash	ali	0.00	1911	1.3 MB
gvfsd-fuse	ali	0.00	1641	553.0 kB
gvfsd-metadata	ali	0.00	1991	876.5 kB
gvfs-goa-volume-moni	ali	0.00	1744	593.9 kB

```
Child process 14454 was started.
A Child process was terminated.
Child process 14475 was started.
A Child process was terminated.
```

```
          Welcom to our SHELL
 ----------------------------------------------
|write help to see commands that you can use|
 ----------------------------------------------




@ali>> help


List of Commands supported:
        >ls
        >cd
        >& after the command --> to open the pro
ess in the background.
        >exit --> to exit the shell
        >all other general commands available in
UNIX shell


@ali>> pwd
/home/ali/Simple_Shell
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> 
```

>>> ls

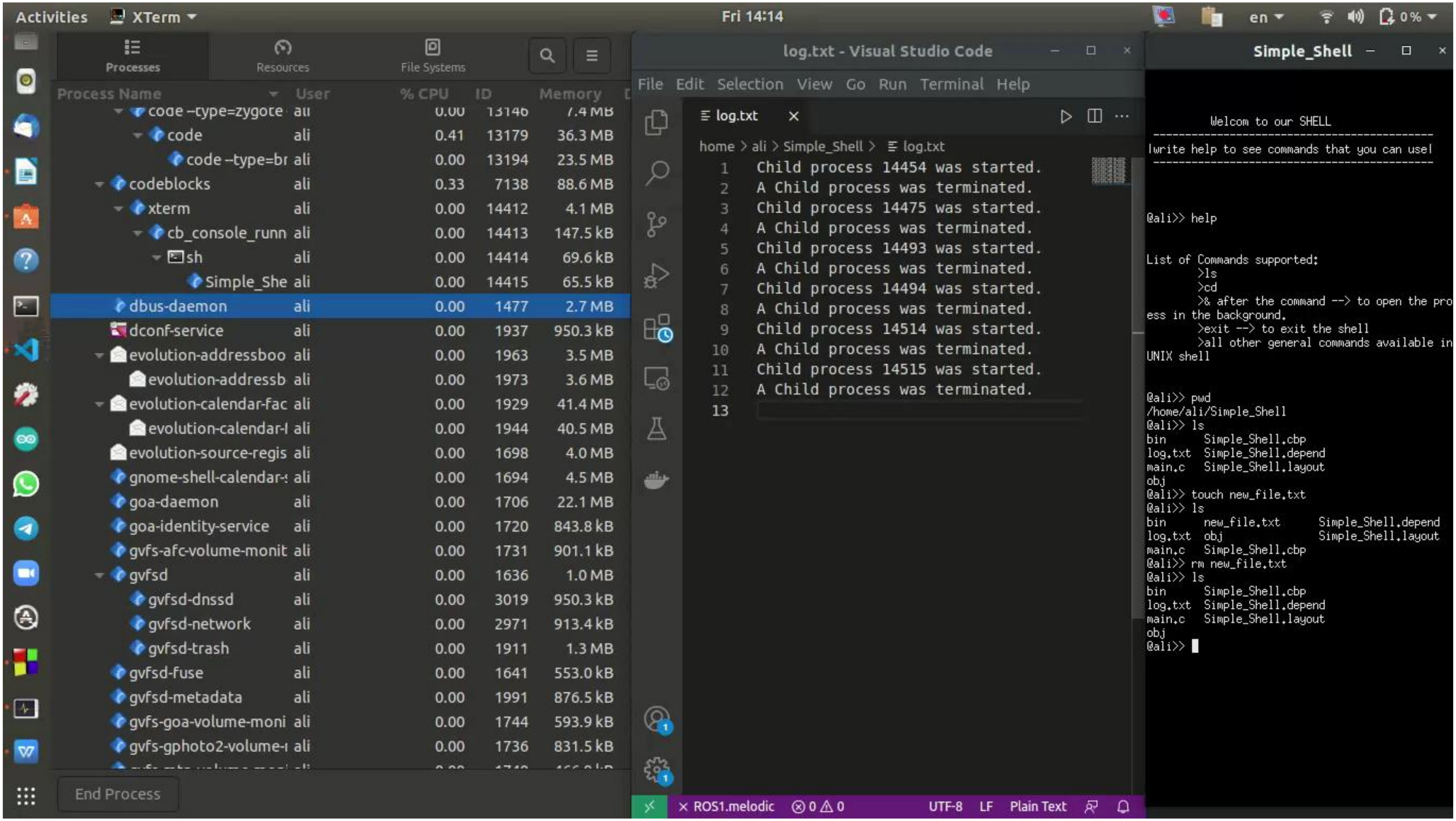


Process Name	User	% CPU	ID	Memory
code --type=zygote	ali	0.00	13146	7.4 MB
code	ali	0.00	13179	34.9 MB
code --type=br	ali	0.00	13194	23.5 MB
codeblocks	ali	0.33	7138	88.6 MB
xterm	ali	0.08	14412	4.1 MB
cb_console_runn	ali	0.00	14413	147.5 kB
sh	ali	0.00	14414	69.6 kB
Simple_She	ali	0.00	14415	65.5 kB
dbus-daemon	ali	0.00	1477	2.7 MB
dconf-service	ali	0.00	1937	950.3 kB
evolution-addressboo	ali	0.00	1963	3.5 MB
evolution-addressb	ali	0.00	1973	3.6 MB
evolution-calendar-fac	ali	0.00	1929	41.4 MB
evolution-calendar-l	ali	0.00	1944	40.5 MB
evolution-source-regis	ali	0.00	1698	4.0 MB
gnome-shell-calendar-:	ali	0.00	1694	4.5 MB
goa-daemon	ali	0.00	1706	22.1 MB
goa-identity-service	ali	0.00	1720	843.8 kB
gvfs-afc-volume-monit	ali	0.00	1731	901.1 kB
gvfsd	ali	0.00	1636	1.0 MB
gvfsd-dnssd	ali	0.00	3019	950.3 kB
gvfsd-network	ali	0.00	2971	913.4 kB
gvfsd-trash	ali	0.00	1911	1.3 MB
gvfsd-fuse	ali	0.00	1641	553.0 kB
gvfsd-metadata	ali	0.00	1991	876.5 kB
gvfs-goa-volume-moni	ali	0.00	1744	593.9 kB
gvfs-gphoto2-volume-i	ali	0.00	1736	831.5 kB

```
Child process 14454 was started.
A Child process was terminated.
Child process 14475 was started.
A Child process was terminated.
Child process 14493 was started.
A Child process was terminated.
```

```
          Welcom to our SHELL
 -----------------------------------------------
|write help to see commands that you can use|
 -----------------------------------------------



@ali>> help


List of Commands supported:
        >ls
        >cd
        >& after the command --> to open the pro
ess in the background.
        >exit --> to exit the shell
        >all other general commands available in
UNIX shell


@ali>> pwd
/home/ali/Simple_Shell
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> touch new_file.txt
@ali>> █
```

>>> touch new_file.txt

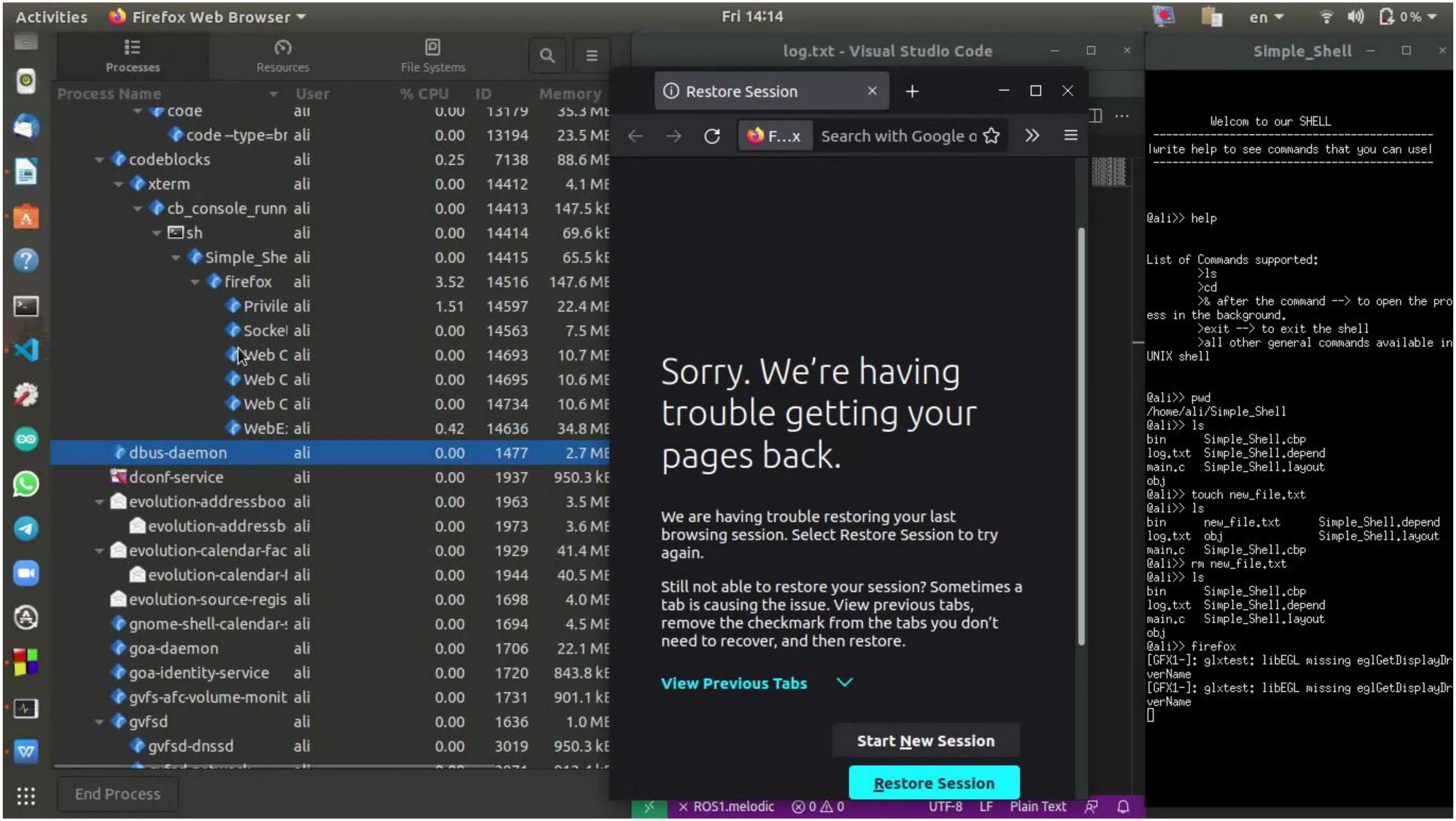


Process Name	User	% CPU	ID	Memory
code --type=zygote	ali	0.00	13146	7.4 MB
code	ali	0.41	13179	36.3 MB
code --type=br	ali	0.00	13194	23.5 MB
codeblocks	ali	0.33	7138	88.6 MB
xterm	ali	0.00	14412	4.1 MB
cb_console_runn	ali	0.00	14413	147.5 kB
sh	ali	0.00	14414	69.6 kB
Simple_She	ali	0.00	14415	65.5 kB
dbus-daemon	ali	0.00	1477	2.7 MB
dconf-service	ali	0.00	1937	950.3 kB
evolution-addressboo	ali	0.00	1963	3.5 MB
evolution-addressb	ali	0.00	1973	3.6 MB
evolution-calendar-fac	ali	0.00	1929	41.4 MB
evolution-calendar-f	ali	0.00	1944	40.5 MB
evolution-source-regis	ali	0.00	1698	4.0 MB
gnome-shell-calendar-s	ali	0.00	1694	4.5 MB
goa-daemon	ali	0.00	1706	22.1 MB
goa-identity-service	ali	0.00	1720	843.8 kB
gvfs-afc-volume-monit	ali	0.00	1731	901.1 kB
gvfsd	ali	0.00	1636	1.0 MB
gvfsd-dnssd	ali	0.00	3019	950.3 kB
gvfsd-network	ali	0.00	2971	913.4 kB
gvfsd-trash	ali	0.00	1911	1.3 MB
gvfsd-fuse	ali	0.00	1641	553.0 kB
gvfsd-metadata	ali	0.00	1991	876.5 kB
gvfs-goa-volume-moni	ali	0.00	1744	593.9 kB
gvfs-gphoto2-volume-r	ali	0.00	1736	831.5 kB

```
Child process 14454 was started.
A Child process was terminated.
Child process 14475 was started.
A Child process was terminated.
Child process 14493 was started.
A Child process was terminated.
Child process 14494 was started.
A Child process was terminated.
Child process 14514 was started.
A Child process was terminated.
Child process 14515 was started.
A Child process was terminated.
```

```
          Welcom to our SHELL
 -----------------------------------------
|write help to see commands that you can use|
 -----------------------------------------



@ali>> help


List of Commands supported:
        >ls
        >cd
        >& after the command --> to open the pro
ess in the background.
        >exit --> to exit the shell
        >all other general commands available in
UNIX shell


@ali>> pwd
/home/ali/Simple_Shell
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> touch new_file.txt
@ali>> ls
bin      new_file.txt      Simple_Shell.depend
log.txt  obj               Simple_Shell.layout
main.c   Simple_Shell.cbp
@ali>> rm new_file.txt
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> █
```

>>> rm new_file.txt



Process Name	User	% CPU	ID	Memory
code	ali	0.00	13179	35.3 MB
code --type=br	ali	0.00	13194	23.5 MB
codeblocks	ali	0.25	7138	88.6 MB
xterm	ali	0.00	14412	4.1 MB
cb_console_runn	ali	0.00	14413	147.5 kB
sh	ali	0.00	14414	69.6 kB
Simple_She	ali	0.00	14415	65.5 kB
firefox	ali	3.52	14516	147.6 MB
Privile	ali	1.51	14597	22.4 MB
Socke	ali	0.00	14563	7.5 MB
Web C	ali	0.00	14693	10.7 MB
Web C	ali	0.00	14695	10.6 MB
Web C	ali	0.00	14734	10.6 MB
WebE:	ali	0.42	14636	34.8 MB
dbus-daemon	ali	0.00	1477	2.7 MB
dconf-service	ali	0.00	1937	950.3 kB
evolution-addressboo	ali	0.00	1963	3.5 MB
evolution-addressb	ali	0.00	1973	3.6 MB
evolution-calendar-fac	ali	0.00	1929	41.4 MB
evolution-calendar-l	ali	0.00	1944	40.5 MB
evolution-source-regis	ali	0.00	1698	4.0 MB
gnome-shell-calendar-:	ali	0.00	1694	4.5 MB
goa-daemon	ali	0.00	1706	22.1 MB
goa-identity-service	ali	0.00	1720	843.8 kB
gvfs-afc-volume-monit	ali	0.00	1731	901.1 kB
gvfsd	ali	0.00	1636	1.0 MB
gvfsd-dnssd	ali	0.00	3019	950.3 kB

Sorry. We're having trouble getting your pages back.

We are having trouble restoring your last browsing session. Select Restore Session to try again.

Still not able to restore your session? Sometimes a tab is causing the issue. View previous tabs, remove the checkmark from the tabs you don't need to recover, and then restore.

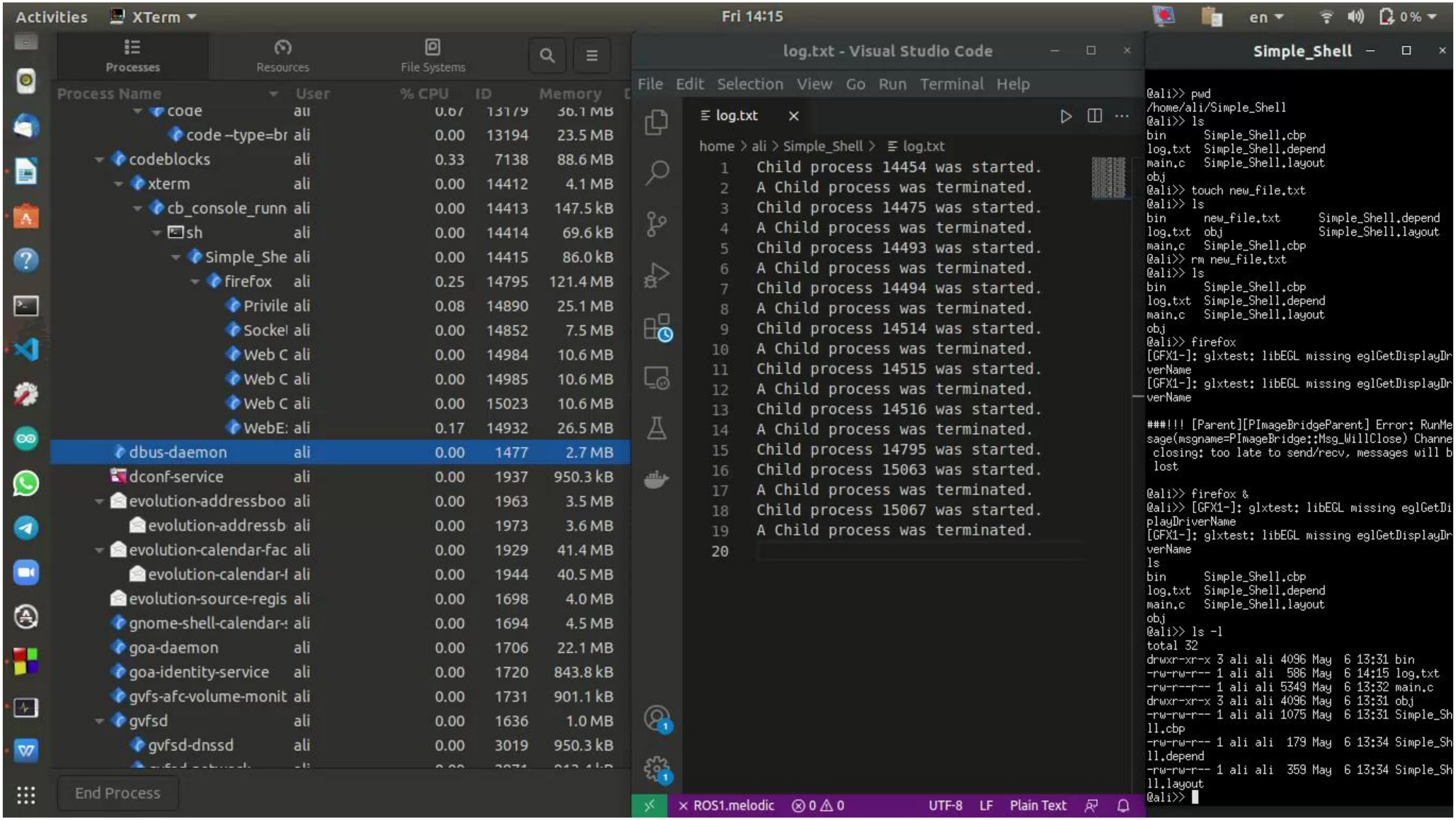
View Previous Tabs

Start New Session

Restore Session

```
         Welcom to our SHELL
 -----------------------------------------
|write help to see commands that you can use|
 -----------------------------------------



@ali>> help


List of Commands supported:
        >ls
        >cd
        >& after the command --> to open the pro
ess in the background.
        >exit --> to exit the shell
        >all other general commands available in
UNIX shell


@ali>> pwd
/home/ali/Simple_Shell
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> touch new_file.txt
@ali>> ls
bin      new_file.txt      Simple_Shell.depend
log.txt  obj               Simple_Shell.layout
main.c   Simple_Shell.cbp
@ali>> rm new_file.txt
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> firefox
[GFX1-]: glxtest: libEGL missing eglGetDisplayDr
verName
[GFX1-]: glxtest: libEGL missing eglGetDisplayDr
verName
▯
```

>>> firefox

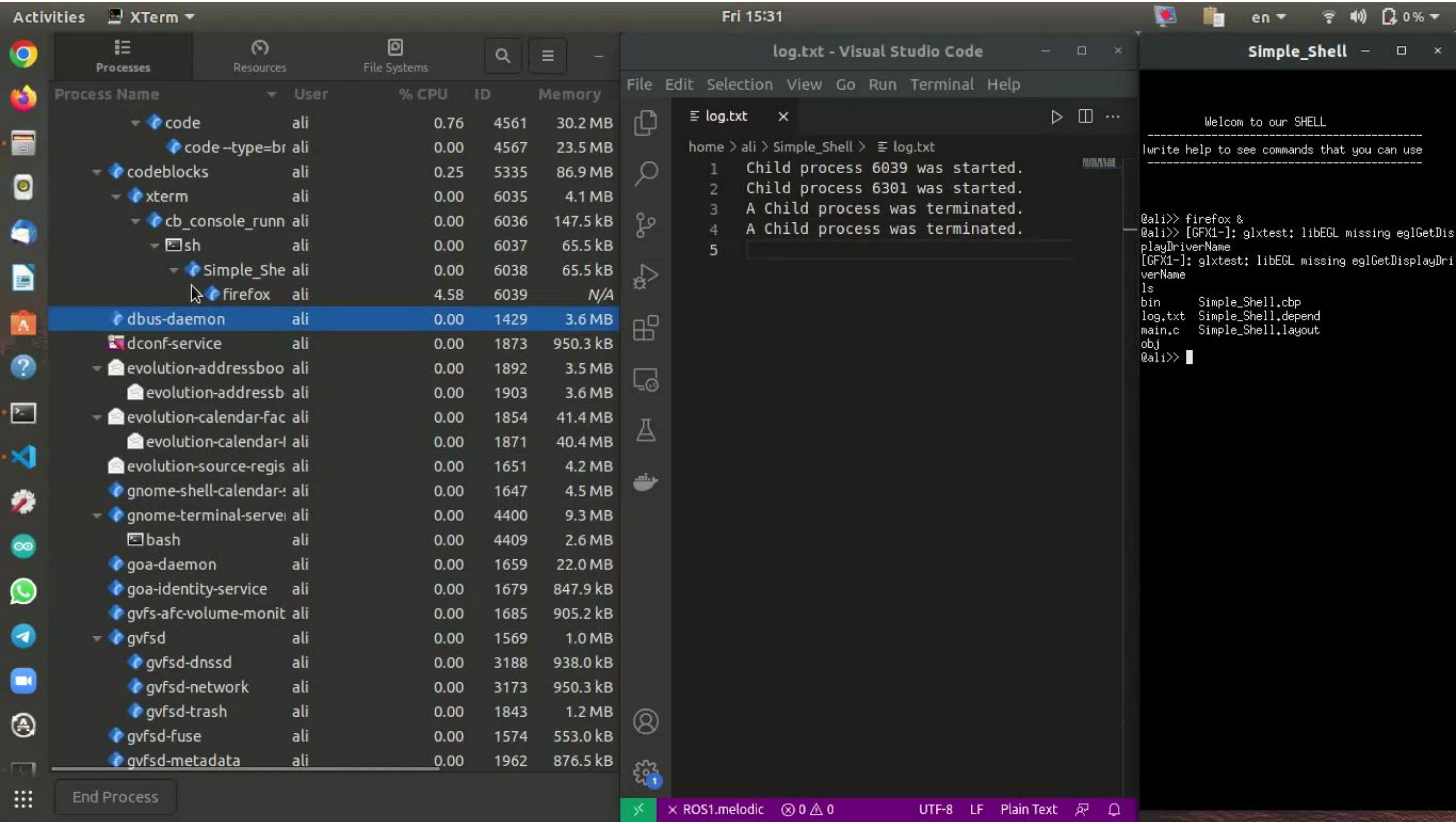


Process Name	User	% CPU	ID	Memory
code	ali	0.67	13179	36.1 MB
code --type=br	ali	0.00	13194	23.5 MB
codeblocks	ali	0.33	7138	88.6 MB
xterm	ali	0.00	14412	4.1 MB
cb_console_runn	ali	0.00	14413	147.5 kB
sh	ali	0.00	14414	69.6 kB
Simple_She	ali	0.00	14415	86.0 kB
firefox	ali	0.25	14795	121.4 MB
Privile	ali	0.08	14890	25.1 MB
Socket	ali	0.00	14852	7.5 MB
Web C	ali	0.00	14984	10.6 MB
Web C	ali	0.00	14985	10.6 MB
Web C	ali	0.00	15023	10.6 MB
WebE:	ali	0.17	14932	26.5 MB
dbus-daemon	ali	0.00	1477	2.7 MB
dconf-service	ali	0.00	1937	950.3 kB
evolution-addressboo	ali	0.00	1963	3.5 MB
evolution-addressb	ali	0.00	1973	3.6 MB
evolution-calendar-fac	ali	0.00	1929	41.4 MB
evolution-calendar-l	ali	0.00	1944	40.5 MB
evolution-source-regis	ali	0.00	1698	4.0 MB
gnome-shell-calendar-:	ali	0.00	1694	4.5 MB
goa-daemon	ali	0.00	1706	22.1 MB
goa-identity-service	ali	0.00	1720	843.8 kB
gvfs-afc-volume-monit	ali	0.00	1731	901.1 kB
gvfsd	ali	0.00	1636	1.0 MB
gvfsd-dnssd	ali	0.00	3019	950.3 kB

```
Child process 14454 was started.
A Child process was terminated.
Child process 14475 was started.
A Child process was terminated.
Child process 14493 was started.
A Child process was terminated.
Child process 14494 was started.
A Child process was terminated.
Child process 14514 was started.
A Child process was terminated.
Child process 14515 was started.
A Child process was terminated.
Child process 14516 was started.
A Child process was terminated.
Child process 14795 was started.
Child process 15063 was started.
A Child process was terminated.
Child process 15067 was started.
A Child process was terminated.
```

```
@ali>> pwd
/home/ali/Simple_Shell
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> touch new_file.txt
@ali>> ls
bin      new_file.txt       Simple_Shell.depend
log.txt  obj                Simple_Shell.layout
main.c   Simple_Shell.cbp
@ali>> rm new_file.txt
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> firefox
[GFX1-]: glxtest: libEGL missing eglGetDisplayDr
verName
[GFX1-]: glxtest: libEGL missing eglGetDisplayDr
verName

###!!! [Parent][PImageBridgeParent] Error: RunMe
sage(msgname=PImageBridge::Msg_WillClose) Channe
 closing: too late to send/recv, messages will b
 lost

@ali>> firefox &
@ali>> [GFX1-]: glxtest: libEGL missing eglGetDi
playDriverName
[GFX1-]: glxtest: libEGL missing eglGetDisplayDr
verName
ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> ls -l
total 32
drwxr-xr-x 3 ali ali 4096 May  6 13:31 bin
-rw-rw-r-- 1 ali ali  586 May  6 14:15 log.txt
-rw-r--r-- 1 ali ali 5349 May  6 13:32 main.c
drwxr-xr-x 3 ali ali 4096 May  6 13:31 obj
-rw-rw-r-- 1 ali ali 1075 May  6 13:31 Simple_Sh
ll.cbp
-rw-rw-r-- 1 ali ali  179 May  6 13:34 Simple_Sh
ll.depend
-rw-rw-r-- 1 ali ali  359 May  6 13:34 Simple_Sh
ll.layout
@ali>>
```

>>>ls -l

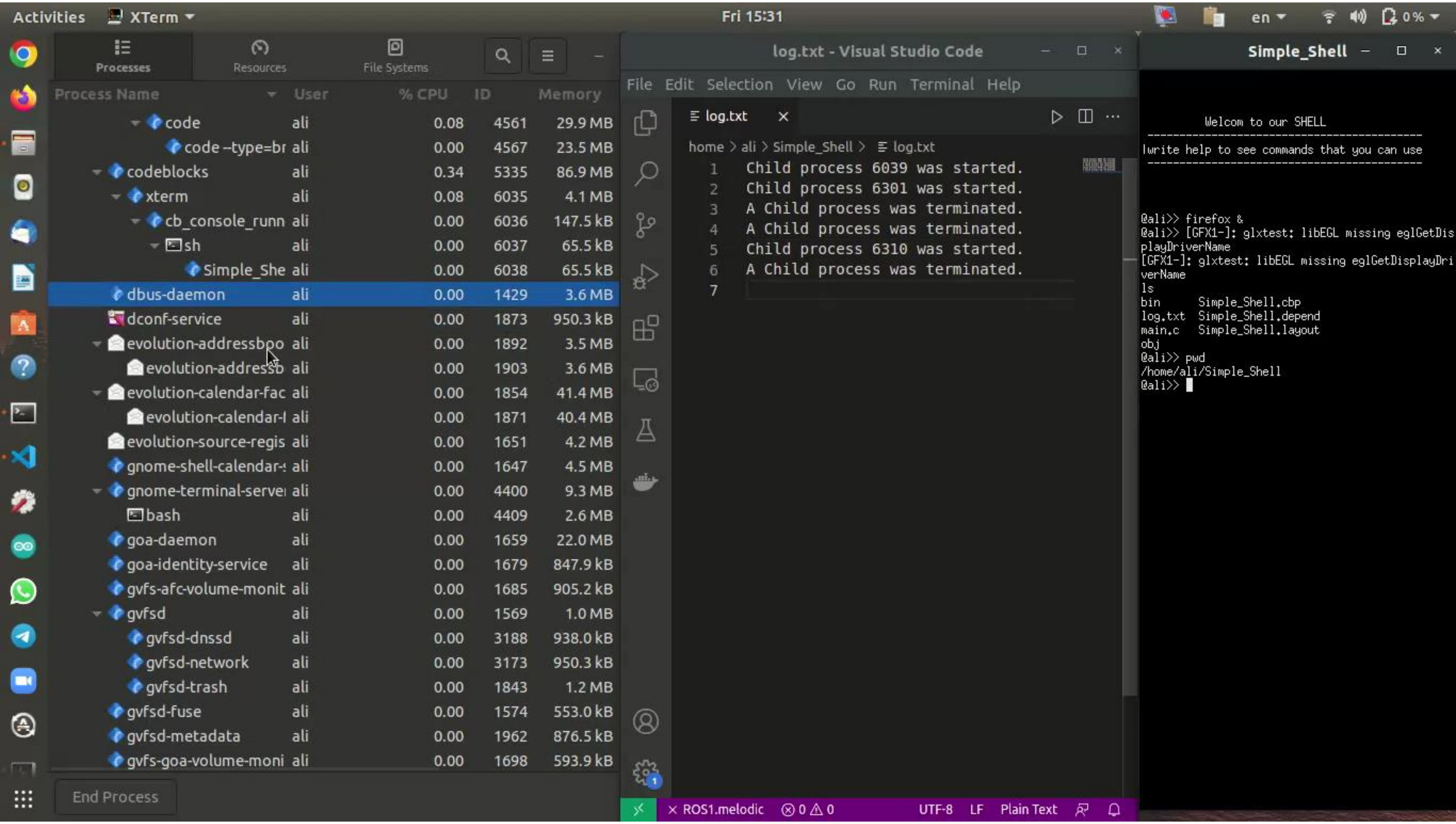


Process Name	User	% CPU	ID	Memory
code	ali	0.76	4561	30.2 MB
code --type=br	ali	0.00	4567	23.5 MB
codeblocks	ali	0.25	5335	86.9 MB
xterm	ali	0.00	6035	4.1 MB
cb_console_runn	ali	0.00	6036	147.5 kB
sh	ali	0.00	6037	65.5 kB
Simple_She	ali	0.00	6038	65.5 kB
firefox	ali	4.58	6039	N/A
dbus-daemon	ali	0.00	1429	3.6 MB
dconf-service	ali	0.00	1873	950.3 kB
evolution-addressboo	ali	0.00	1892	3.5 MB
evolution-addressb	ali	0.00	1903	3.6 MB
evolution-calendar-fac	ali	0.00	1854	41.4 MB
evolution-calendar-l	ali	0.00	1871	40.4 MB
evolution-source-regis	ali	0.00	1651	4.2 MB
gnome-shell-calendar-:	ali	0.00	1647	4.5 MB
gnome-terminal-server	ali	0.00	4400	9.3 MB
bash	ali	0.00	4409	2.6 MB
goa-daemon	ali	0.00	1659	22.0 MB
goa-identity-service	ali	0.00	1679	847.9 kB
gvfs-afc-volume-monit	ali	0.00	1685	905.2 kB
gvfsd	ali	0.00	1569	1.0 MB
gvfsd-dnssd	ali	0.00	3188	938.0 kB
gvfsd-network	ali	0.00	3173	950.3 kB
gvfsd-trash	ali	0.00	1843	1.2 MB
gvfsd-fuse	ali	0.00	1574	553.0 kB
gvfsd-metadata	ali	0.00	1962	876.5 kB

```
Child process 6039 was started.
Child process 6301 was started.
A Child process was terminated.
A Child process was terminated.
```

```
        Welcom to our SHELL
 ------------------------------------------
|write help to see commands that you can use
 ------------------------------------------


@ali>> firefox &
@ali>> [GFX1-]: glxtest: libEGL missing eglGetDis
playDriverName
[GFX1-]: glxtest: libEGL missing eglGetDisplayDri
verName
ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> 
```

Firefox was closed but still zombie processes



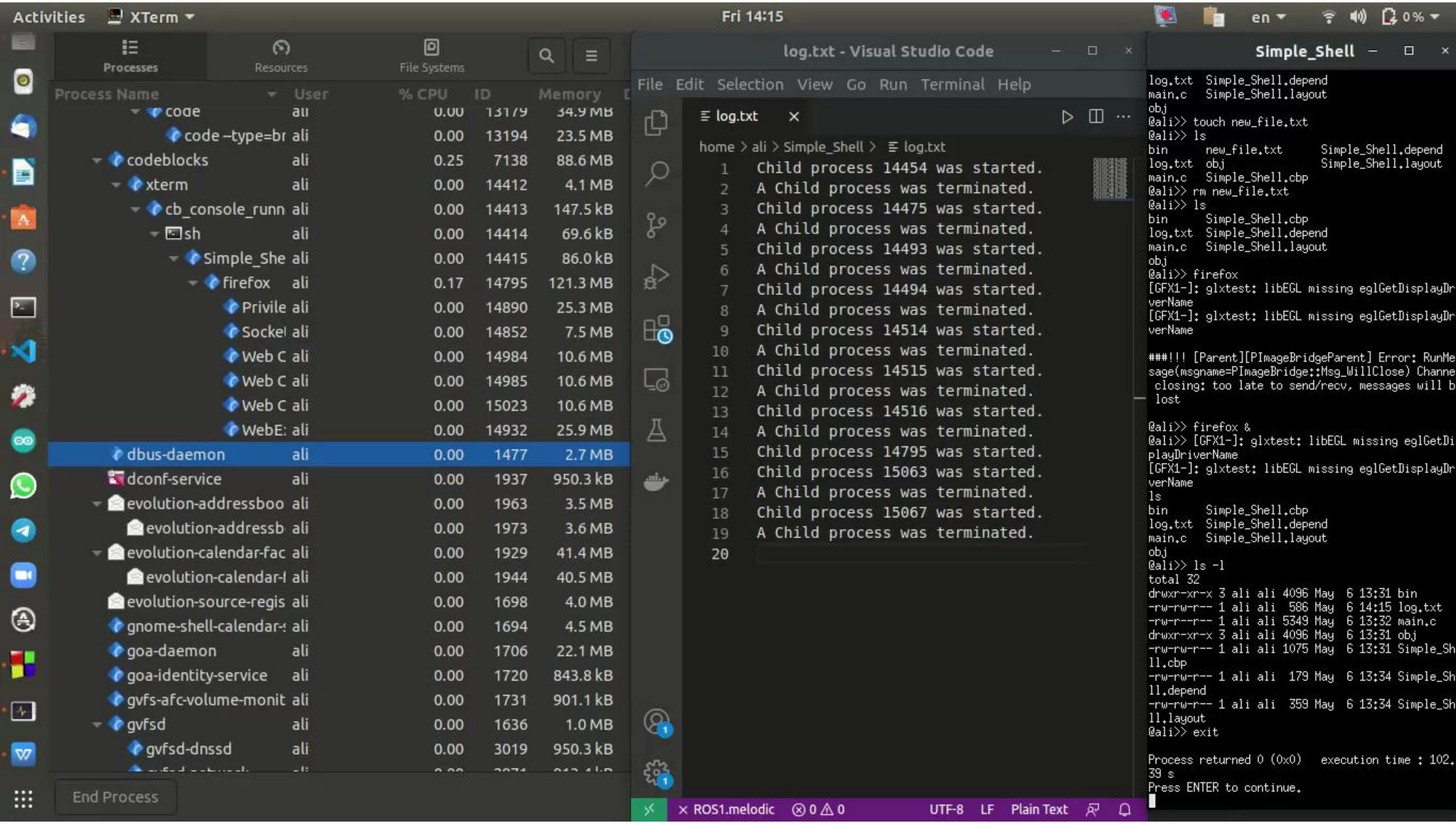
Process Name	User	% CPU	ID	Memory
code	ali	0.08	4561	29.9 MB
code --type=br	ali	0.00	4567	23.5 MB
codeblocks	ali	0.34	5335	86.9 MB
xterm	ali	0.08	6035	4.1 MB
cb_console_runn	ali	0.00	6036	147.5 kB
sh	ali	0.00	6037	65.5 kB
Simple_She	ali	0.00	6038	65.5 kB
dbus-daemon	ali	0.00	1429	3.6 MB
dconf-service	ali	0.00	1873	950.3 kB
evolution-addressboo	ali	0.00	1892	3.5 MB
evolution-addressb	ali	0.00	1903	3.6 MB
evolution-calendar-fac	ali	0.00	1854	41.4 MB
evolution-calendar-f	ali	0.00	1871	40.4 MB
evolution-source-regis	ali	0.00	1651	4.2 MB
gnome-shell-calendar-s	ali	0.00	1647	4.5 MB
gnome-terminal-serve	ali	0.00	4400	9.3 MB
bash	ali	0.00	4409	2.6 MB
goa-daemon	ali	0.00	1659	22.0 MB
goa-identity-service	ali	0.00	1679	847.9 kB
gvfs-afc-volume-monit	ali	0.00	1685	905.2 kB
gvfsd	ali	0.00	1569	1.0 MB
gvfsd-dnssd	ali	0.00	3188	938.0 kB
gvfsd-network	ali	0.00	3173	950.3 kB
gvfsd-trash	ali	0.00	1843	1.2 MB
gvfsd-fuse	ali	0.00	1574	553.0 kB
gvfsd-metadata	ali	0.00	1962	876.5 kB
gvfs-goa-volume-moni	ali	0.00	1698	593.9 kB

log.txt

```
Child process 6039 was started.
Child process 6301 was started.
A Child process was terminated.
A Child process was terminated.
Child process 6310 was started.
A Child process was terminated.
```

```
          Welcom to our SHELL
 ------------------------------------------
|write help to see commands that you can use
 ------------------------------------------



@ali>> firefox &
@ali>> [GFX1-]: glxtest: libEGL missing eglGetDis
playDriverName
[GFX1-]: glxtest: libEGL missing eglGetDisplayDri
verName
ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> pwd
/home/ali/Simple_Shell
@ali>> 
```

Zombie processes fully terminated after executing any command, i.e, >>>pwd



Process Name	User	% CPU	ID	Memory
code	ali	0.00	13179	34.9 MB
code --type=br	ali	0.00	13194	23.5 MB
codeblocks	ali	0.25	7138	88.6 MB
xterm	ali	0.00	14412	4.1 MB
cb_console_runn	ali	0.00	14413	147.5 kB
sh	ali	0.00	14414	69.6 kB
Simple_She	ali	0.00	14415	86.0 kB
firefox	ali	0.17	14795	121.3 MB
Privile	ali	0.00	14890	25.3 MB
Socke	ali	0.00	14852	7.5 MB
Web C	ali	0.00	14984	10.6 MB
Web C	ali	0.00	14985	10.6 MB
Web C	ali	0.00	15023	10.6 MB
WebE:	ali	0.00	14932	25.9 MB
dbus-daemon	ali	0.00	1477	2.7 MB
dconf-service	ali	0.00	1937	950.3 kB
evolution-addressboo	ali	0.00	1963	3.5 MB
evolution-addressb	ali	0.00	1973	3.6 MB
evolution-calendar-fac	ali	0.00	1929	41.4 MB
evolution-calendar-l	ali	0.00	1944	40.5 MB
evolution-source-regis	ali	0.00	1698	4.0 MB
gnome-shell-calendar-:	ali	0.00	1694	4.5 MB
goa-daemon	ali	0.00	1706	22.1 MB
goa-identity-service	ali	0.00	1720	843.8 kB
gvfs-afc-volume-monit	ali	0.00	1731	901.1 kB
gvfsd	ali	0.00	1636	1.0 MB
gvfsd-dnssd	ali	0.00	3019	950.3 kB

```
Child process 14454 was started.
A Child process was terminated.
Child process 14475 was started.
A Child process was terminated.
Child process 14493 was started.
A Child process was terminated.
Child process 14494 was started.
A Child process was terminated.
Child process 14514 was started.
A Child process was terminated.
Child process 14515 was started.
A Child process was terminated.
Child process 14516 was started.
A Child process was terminated.
Child process 14795 was started.
Child process 15063 was started.
A Child process was terminated.
Child process 15067 was started.
A Child process was terminated.
```

```
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> touch new_file.txt
@ali>> ls
bin      new_file.txt      Simple_Shell.depend
log.txt  obj               Simple_Shell.layout
main.c   Simple_Shell.cbp
@ali>> rm new_file.txt
@ali>> ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> firefox
[GFX1-]: glxtest: libEGL missing eglGetDisplayDr
verName
[GFX1-]: glxtest: libEGL missing eglGetDisplayDr
verName

###!!! [Parent][PImageBridgeParent] Error: RunMe
sage(msgname=PImageBridge::Msg_WillClose) Channe
 closing: too late to send/recv, messages will b
 lost

@ali>> firefox &
@ali>> [GFX1-]: glxtest: libEGL missing eglGetDi
playDriverName
[GFX1-]: glxtest: libEGL missing eglGetDisplayDr
verName
ls
bin      Simple_Shell.cbp
log.txt  Simple_Shell.depend
main.c   Simple_Shell.layout
obj
@ali>> ls -l
total 32
drwxr-xr-x 3 ali ali 4096 May  6 13:31 bin
-rw-rw-r-- 1 ali ali  586 May  6 14:15 log.txt
-rw-r--r-- 1 ali ali 5349 May  6 13:32 main.c
drwxr-xr-x 3 ali ali 4096 May  6 13:31 obj
-rw-rw-r-- 1 ali ali 1075 May  6 13:31 Simple_Sh
ll.cbp
-rw-rw-r-- 1 ali ali  179 May  6 13:34 Simple_Sh
ll.depend
-rw-rw-r-- 1 ali ali  359 May  6 13:34 Simple_Sh
ll.layout
@ali>> exit

Process returned 0 (0x0)   execution time : 102.
39 s
Press ENTER to continue.
```

>>> exit