**UKKIL OPERATING SYSTEM SIMULATOR**

**(OPERATING SYSTEM LAB)**

****

**SUBMITTED BY:**

Iman Fatima (2021-SE-10)

Ume-Habiba (2021-SE-27)

Kausar Fatima (2021-SE-25)

Khadeeja Rasti (2021-SE-53)

Laiba Amber Ejaz (2021-SE-37)

**SUBMITTED TO:**

SIR WAQAS ALI

**DEPARTMENT OF COMPUTER SCIENCE**

**UNIVERSITY OF ENGINEERING AND TECHNOLOGY LAHORE (KSK CAMPUS)**

**TABLE OF CONTENTS:**

* INTRODUCTION
* VISION OF PRODUCT
* REQUIREMENTS

1. FUNCTIONAL REQUIREMENTS
2. NON- FUNCTIONAL REQUIREMENTS

* KEY TASKS

1. CORE FUNCTIONALITIES
2. OPTIONAL FUNCTIONALITIES

* TOOLS AND TECHNOLOGY
* TASKS
* KEY CONCEPTS
* WORK-BREAK-DOWN-STRUCTURE
* GANTT-CHART

**Introduction:**

An operating system (OS) is system software that manages computer hardware, software resources, and provides common services for computer programs. In this project we will design a simulation of operating system on our own named UKKIL.

**Vision of Product:**

**For** Linux-Management Community **who** wants to give an operating system platform to Linux users. **UKKIL** is an operating system simulator **that** will provide you File Management System, Contact Management System, Games and Novel Apps etc. **Unlike** typical operating system simulator projects, **our** focus will have to facilitate users with more security providing user validation as well as User-Friendly-GUI-Based apps.

**Requirements:**

**Functional Requirements:**

* **Registration:**

The system must be able to allow users to login through valid user\_id and password.

* **Hostname:**

The system must be able to display host-os name.

* **Contact Management:**

1. The system must be able to allow users to add contacts.
2. The system must be able to allow users to display contacts.
3. The system must be able to allow users to save contacts.

* **File Management:**

1. The system must be able to allow users to create files.
2. The system must be able to allow users to delete files.
3. The system must be able to allow users to move files.
4. The system must be able to allow users to copy files.
5. The system must be able to allow users to check file-info.

* **Novel Management App:**

The system must be able to allow users to search files and selected file should be displayed.

* **Image viewer management:**

The system must be able to allow users to search images and images should be displayed along with their details.

* **Calculator:**

1. The system must be able to allow users to add numbers.
2. The system must be able to allow users to subtract numbers.
3. The system must be able to allow users to multiply numbers.
4. The system must be able to allow users to divide numbers.
5. The system must be able to allow users to clear results.

* **Joke generator:**

The system must be able to allow users to generate a joke.

* **Notepad:**

1. The system must be able to allow users to create notepad file and write text in it.
2. The system must be able to do auto-save text written in notepad after every some of the seconds.

* **Text Editor**

1. The system must be able to allow users edit text in any opened file.

* **Tic-tac toe:**

1. The system must be able to allow users to play TIC-TAC-TOE Game and restart it after losing.

* **Date and time reliant:**

1. The system must be date and time reliant.

* **Timer:**

1. The system must be able to allow users to enter time in hrs, min, as well as seconds and pause, resume, and restart timer.

* **Log-out:**

1. The system must be able to allow users to log-out.

**Non-Functional Requirements:**

* **Performance:**

1. The system shall be able to provide instant services to 500 users at a time.
2. The system shall be lightweight and shall be able to respond instantly.

* **Security:**

1. The system shall be able provide password lock for opening the system.

* **Safety:**

1. The system shall be able to tolerate user’s mistake.

* **Reliability:**

1. The system shall be to preserve user’s data.
2. The system shall be to available 24/7.

* **Memory:**

1. The system shall be able to provide memory for storing 500 login’ data.
2. The system shall be able to make users able to use provide memory.

* **Data Integrity:**

1. The system shall be able to integrate data of different types such as numbers, alphabets etc.

* **Maintenance:**

1. The system shall be to be updated timely.

* **Usability:**

1. The system should be easy to use and intuitive for users, with clear navigation and a
2. user-friendly interface.

* **Portability:**

1. The system shall be working on Linux-Ubunto-Installed desktop.

* **Back & Restore:**

1. The system shall be able to store backup data of users even if user log-out of their account.

**KEY TASKS:**

* **User Validation:**

User credentials such as username and password would be validated and system would not allow unauthorized users to login to the operating system.

* **Core Functionalities:**

1. Notepad (writing into file with auto-save function)
2. Calculator
3. Time
4. Creating a file
5. Move file
6. Copy file
7. Delete file
8. Check file info
9. Image viewer
10. Text editor

* **Optional Functionalities:**

1. Timer
2. Joke Generator
3. Novel App
4. Game
5. Contact Manager

**Tools and Technology:**

Following Tools and Technologies would be used:

1. **Vmware:**

Vmware Player is a virtualization software package that allows running of

UBUNTO on Windows Operating System.

1. **Ubunto:**

UBUNTO is Linux-Based Operating System. It can be run on computers,

smartphones and network servers. All the development principles of UBUNTO OS are based

on Open-Source Software Development.

1. **C++:**

C++ is used as back-end language. In this we have used file handling.

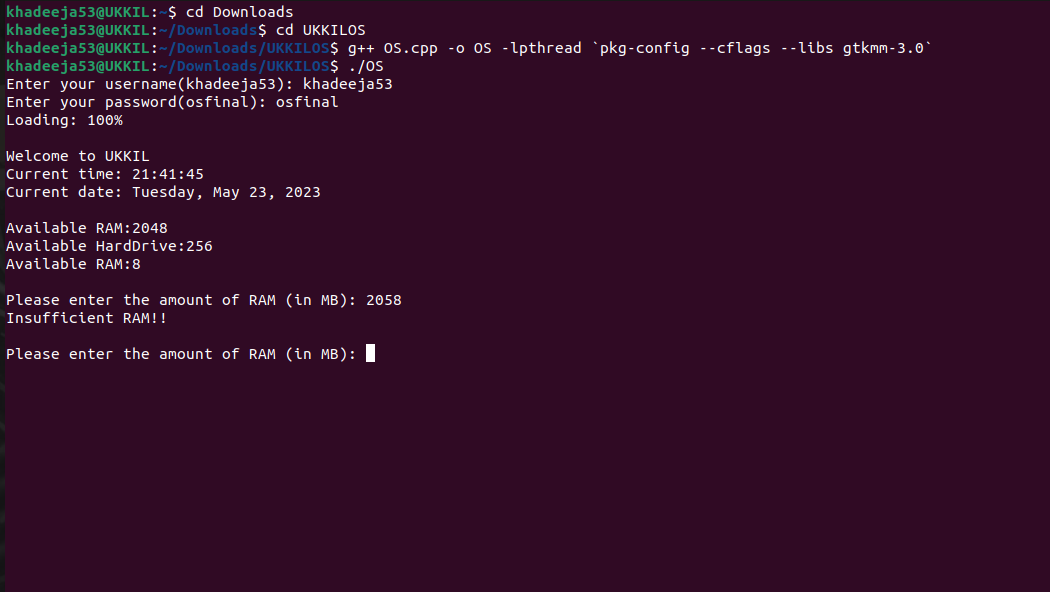
1. **Gtkmm:**

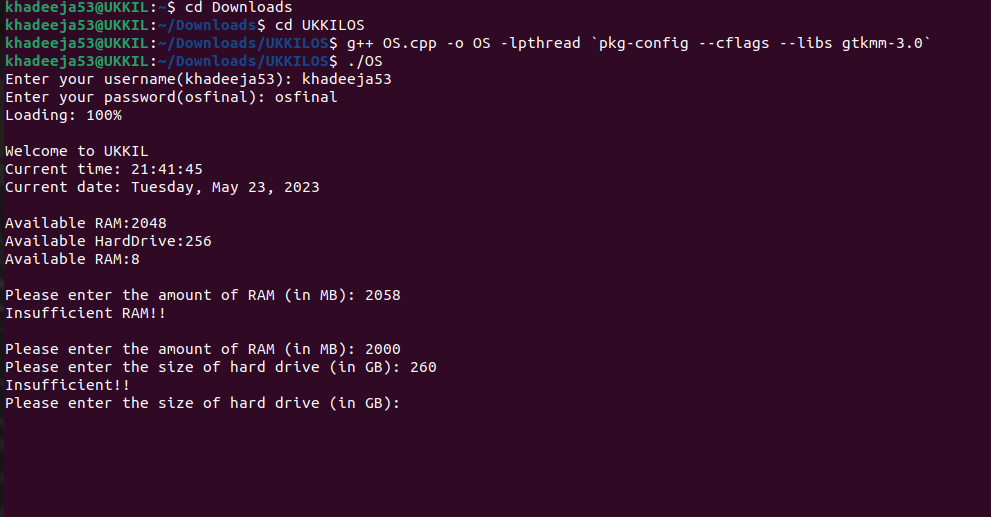
GTKMM is used for providing graphical interface.

**TASKS IMPLEMENTATION:**

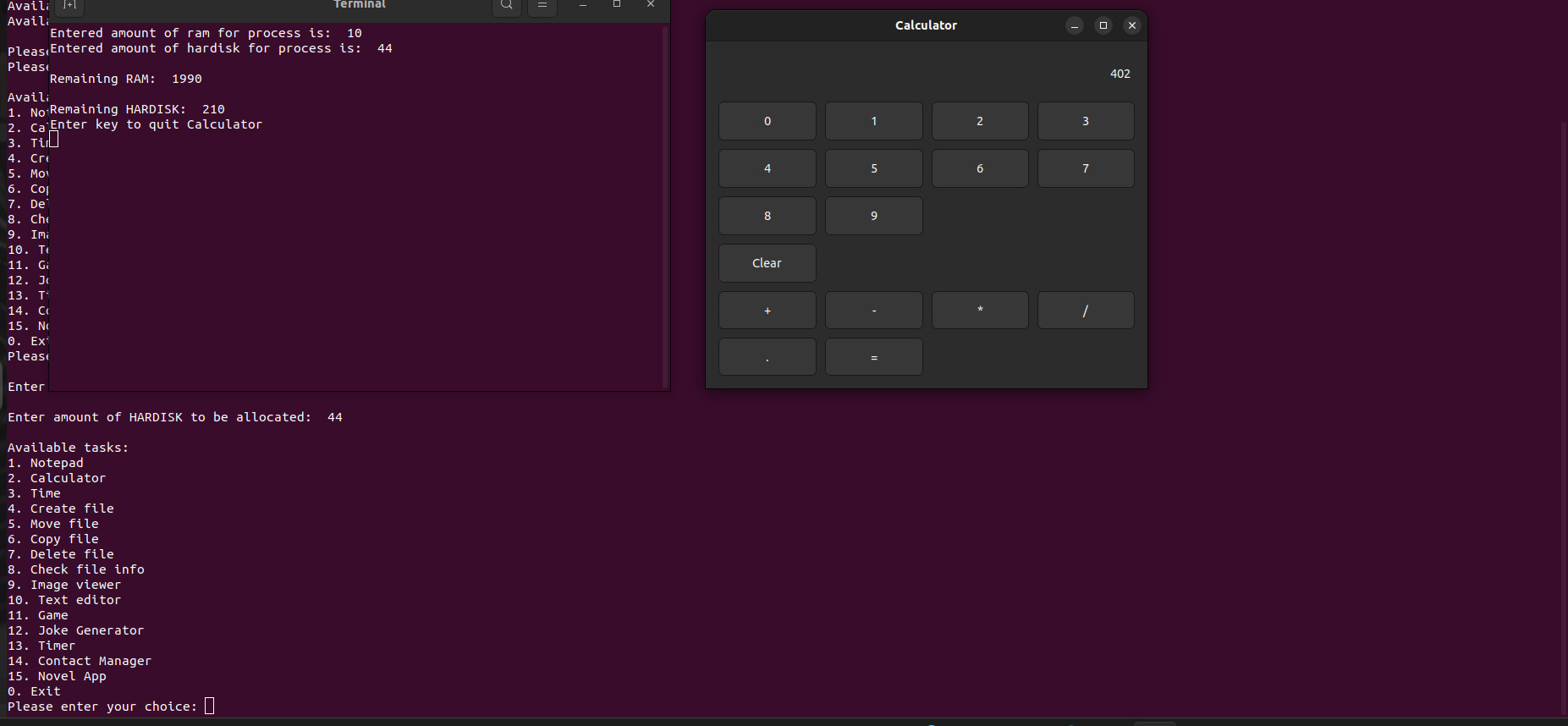
We allocated the memory of ram and hard drive, if user put beyond the limit, then message will show on the terminal If the user enters accurate information/memory, the user will perform the multiple tasks on operating system. If ,login information is not valid then (invalid username and password) s not valid message will be shown on screen.

**Main:**

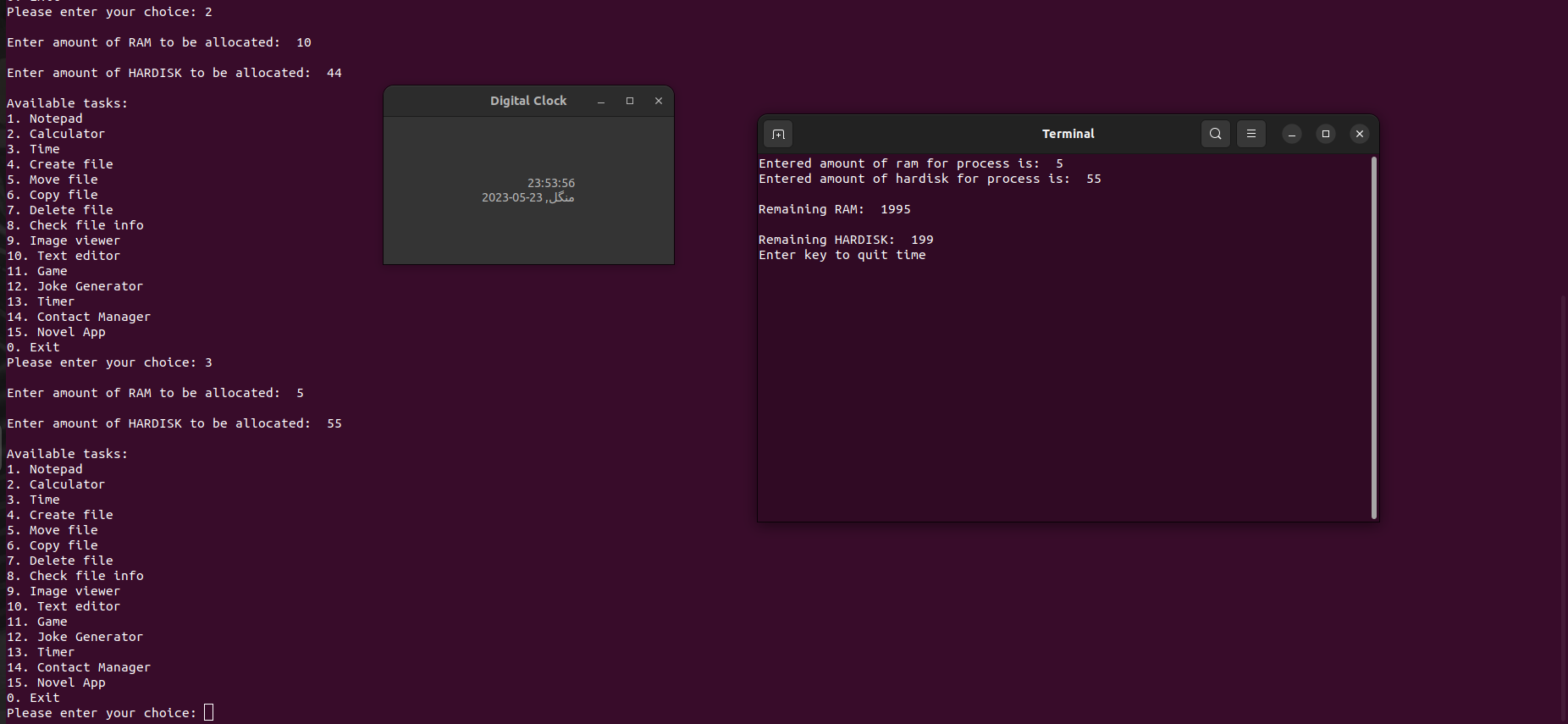
****

****

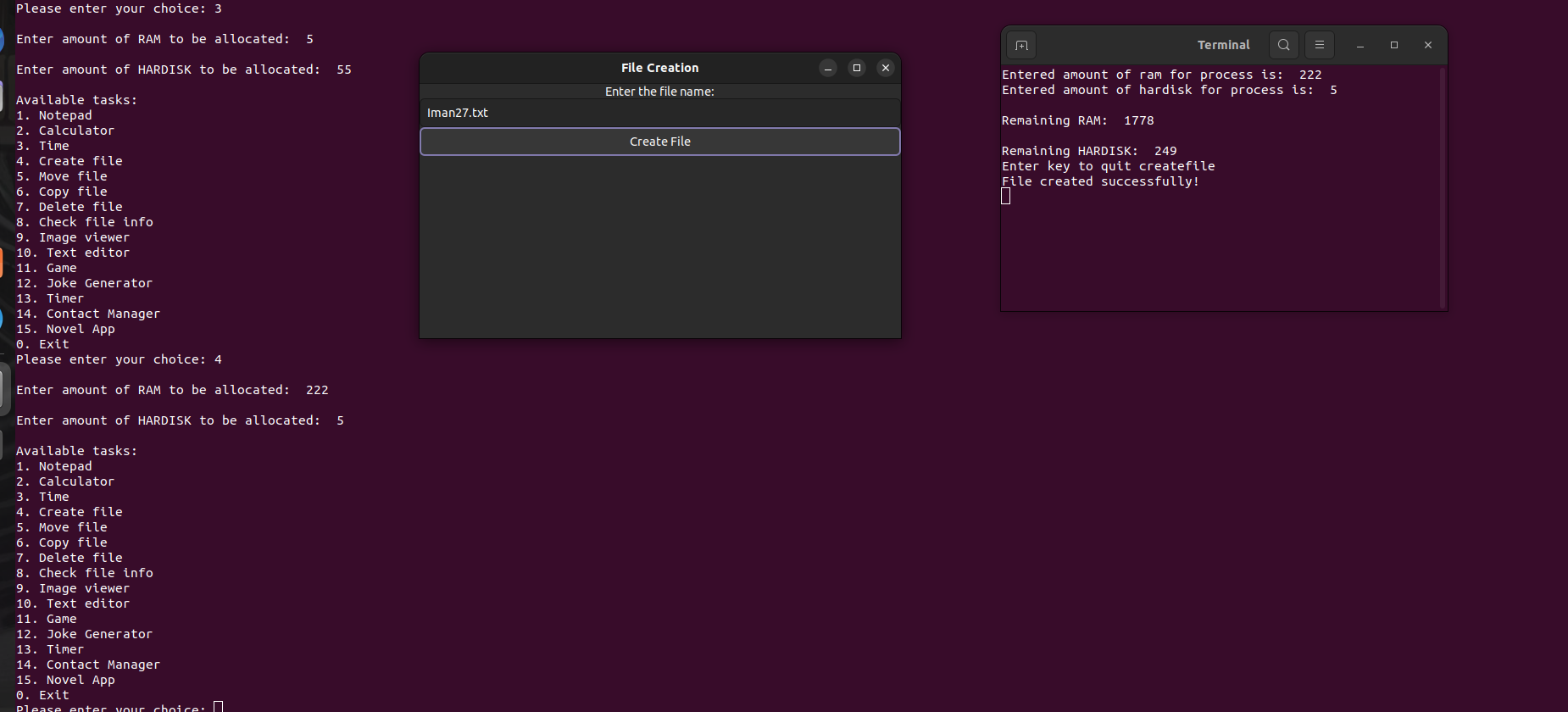
**Calculator:**

****

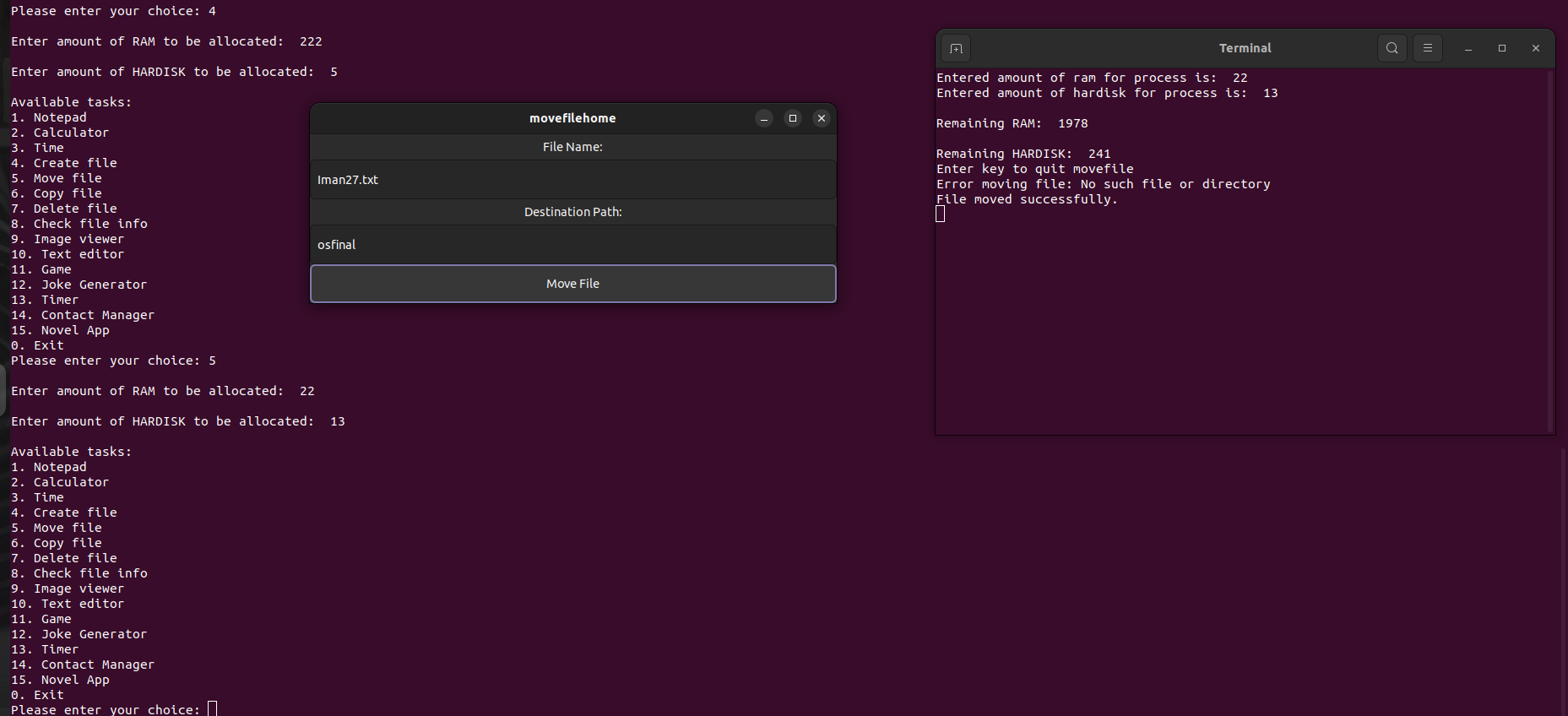
**Time:**

****

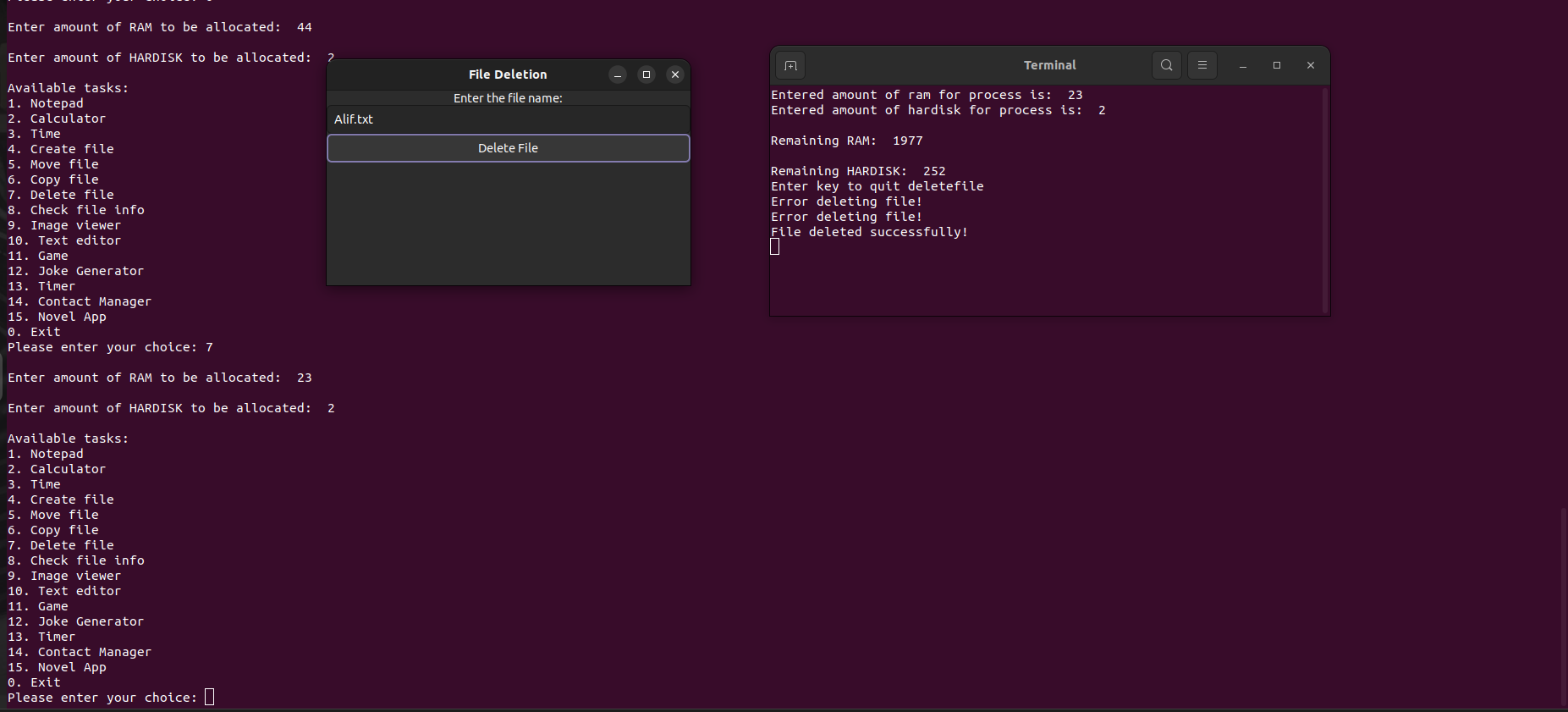
**Create File:**

****

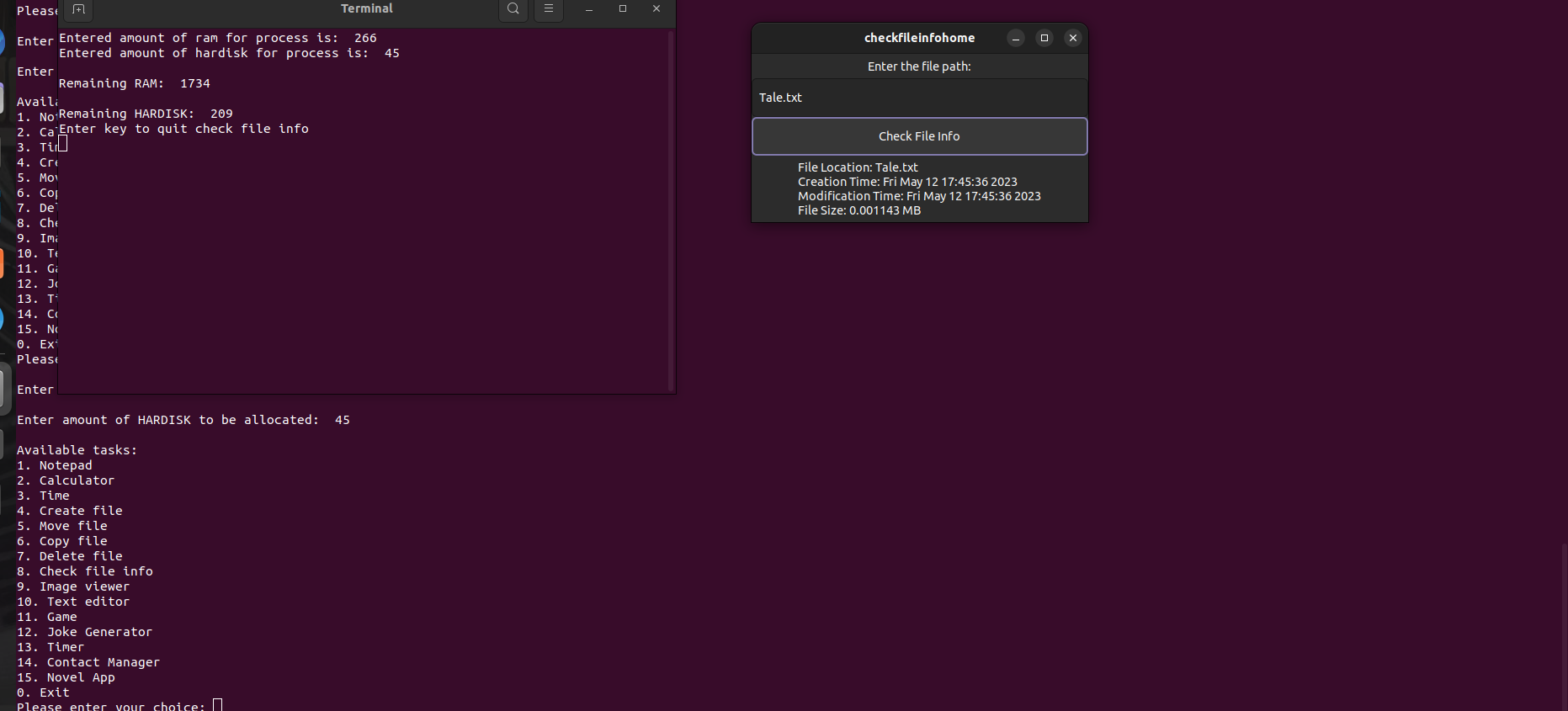
**Move File:**

****

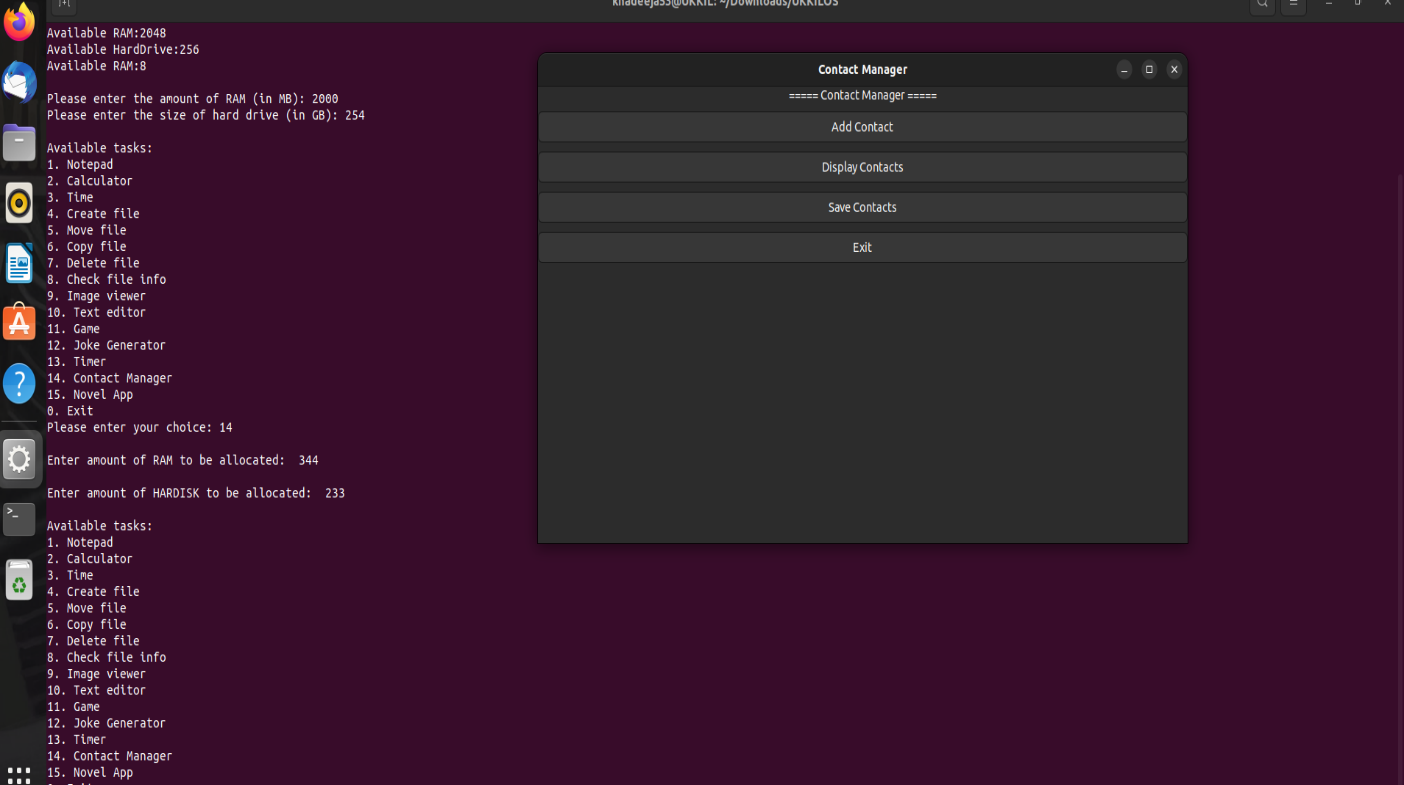
**Delete File:**

****

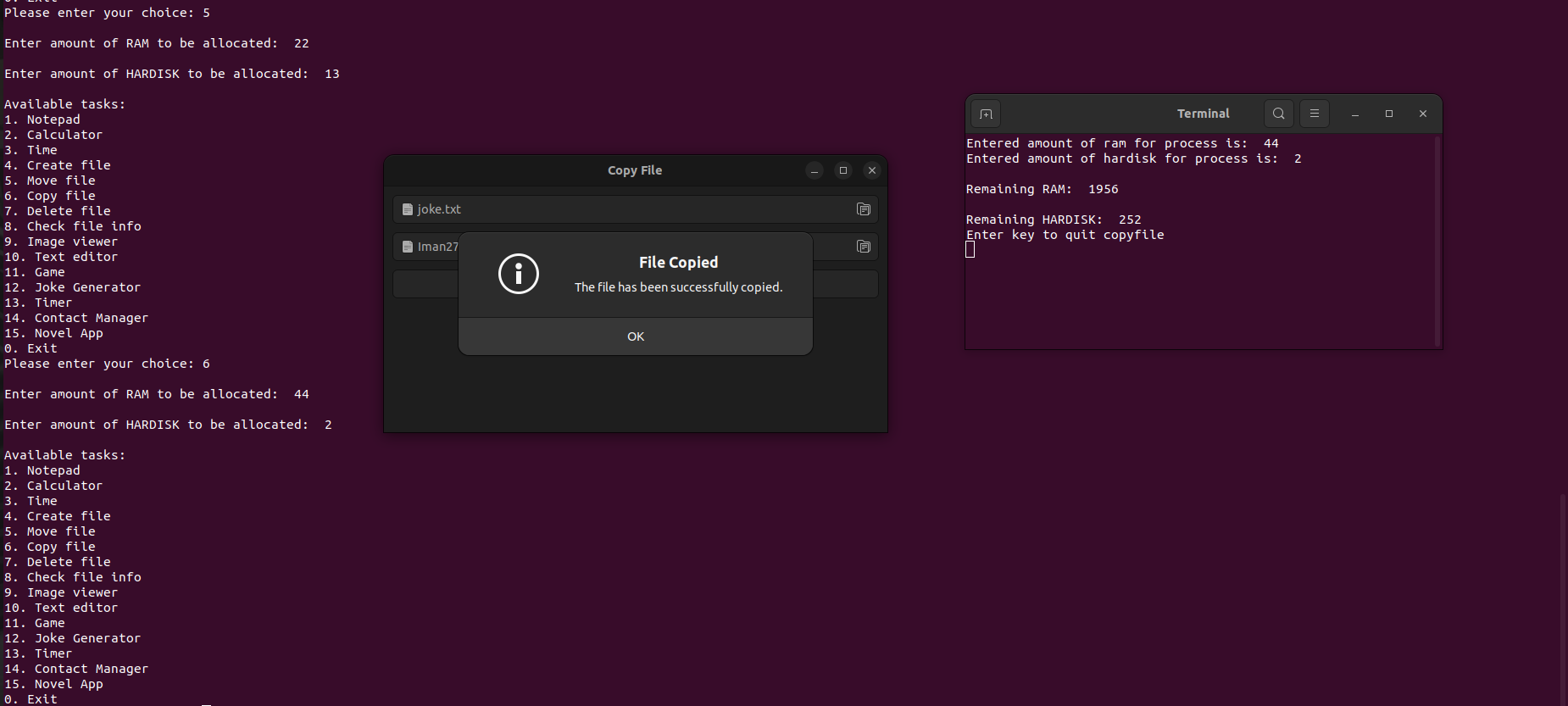
**Check File Info:**

****

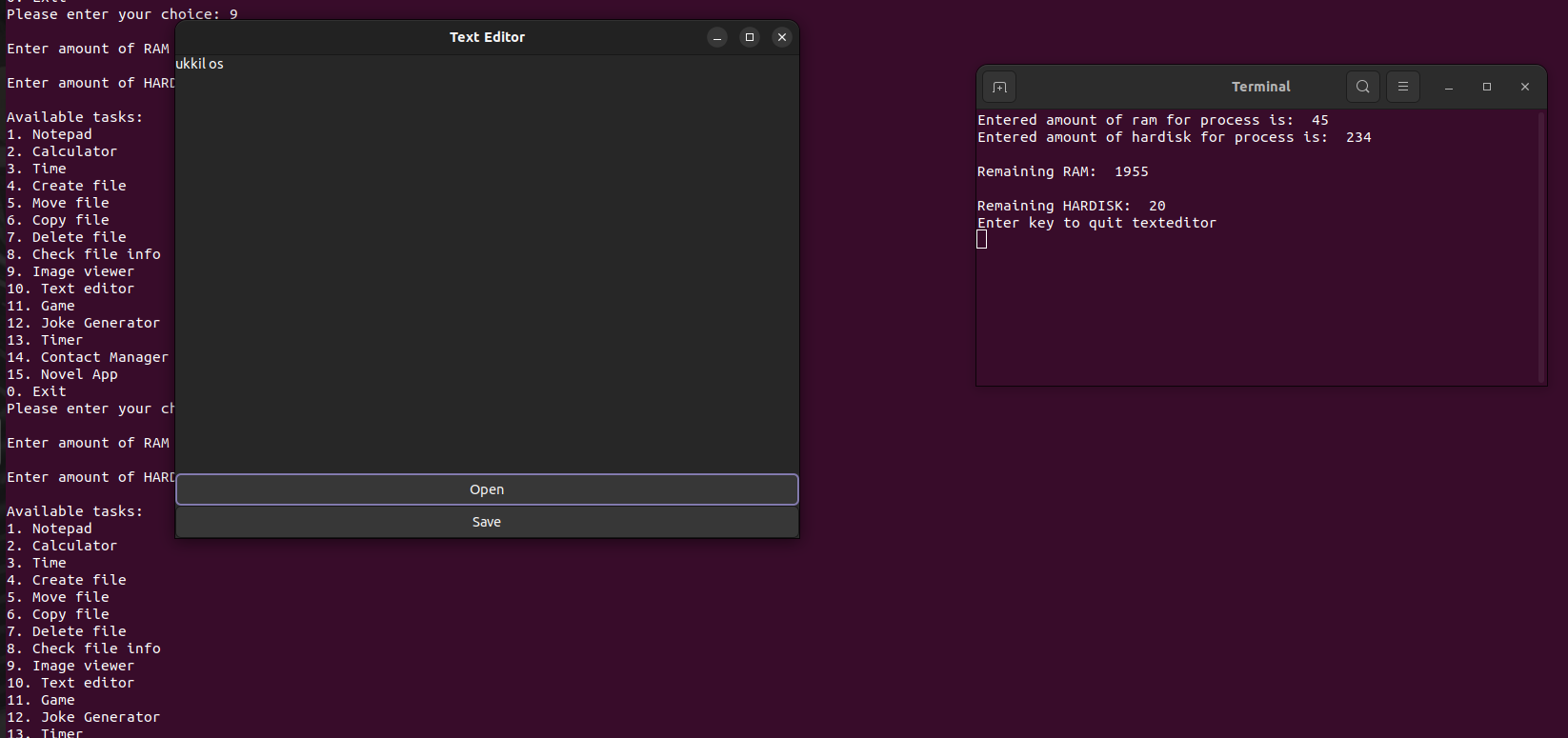
**Contact Manager:**

****

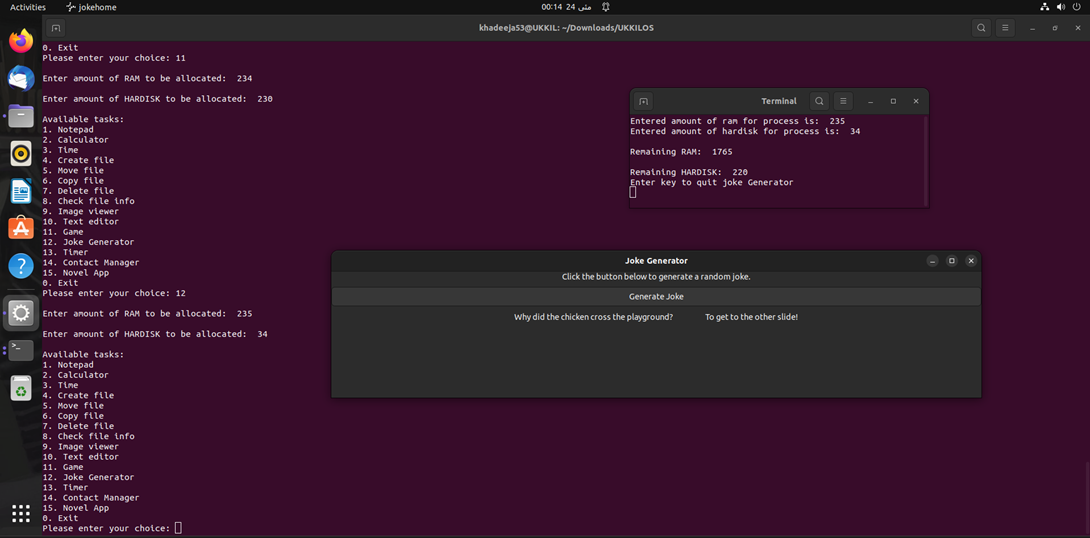
**File Copy:**

****

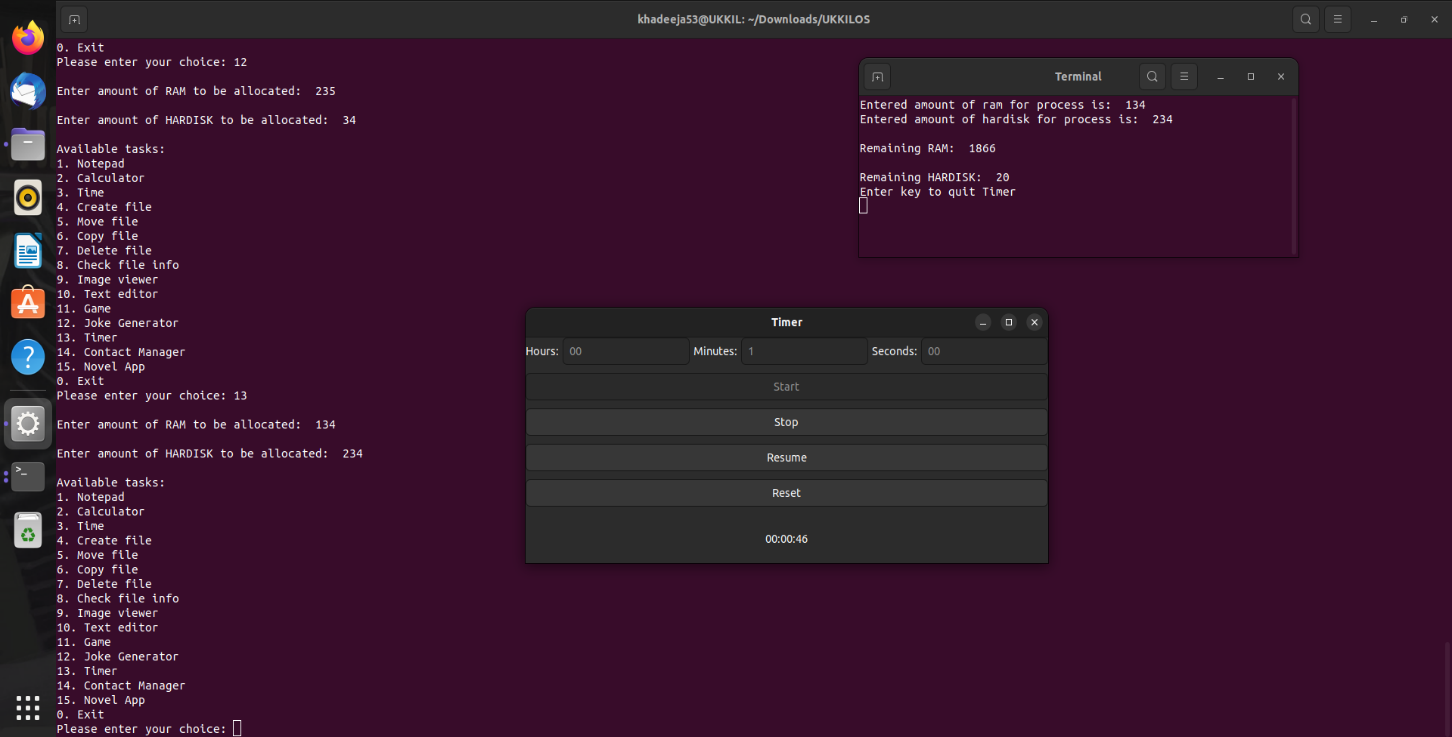
**Text Editor:**

****

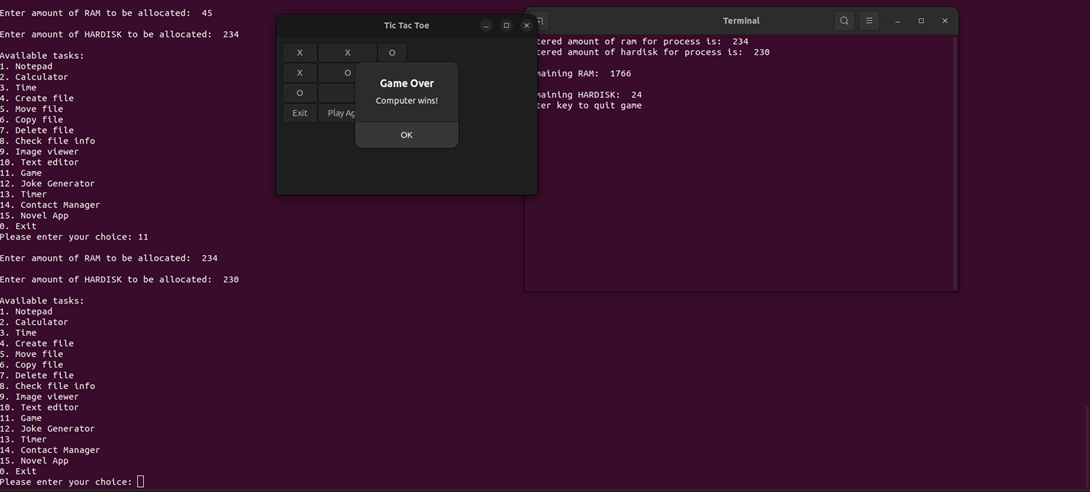
**Joke Generator:**

****

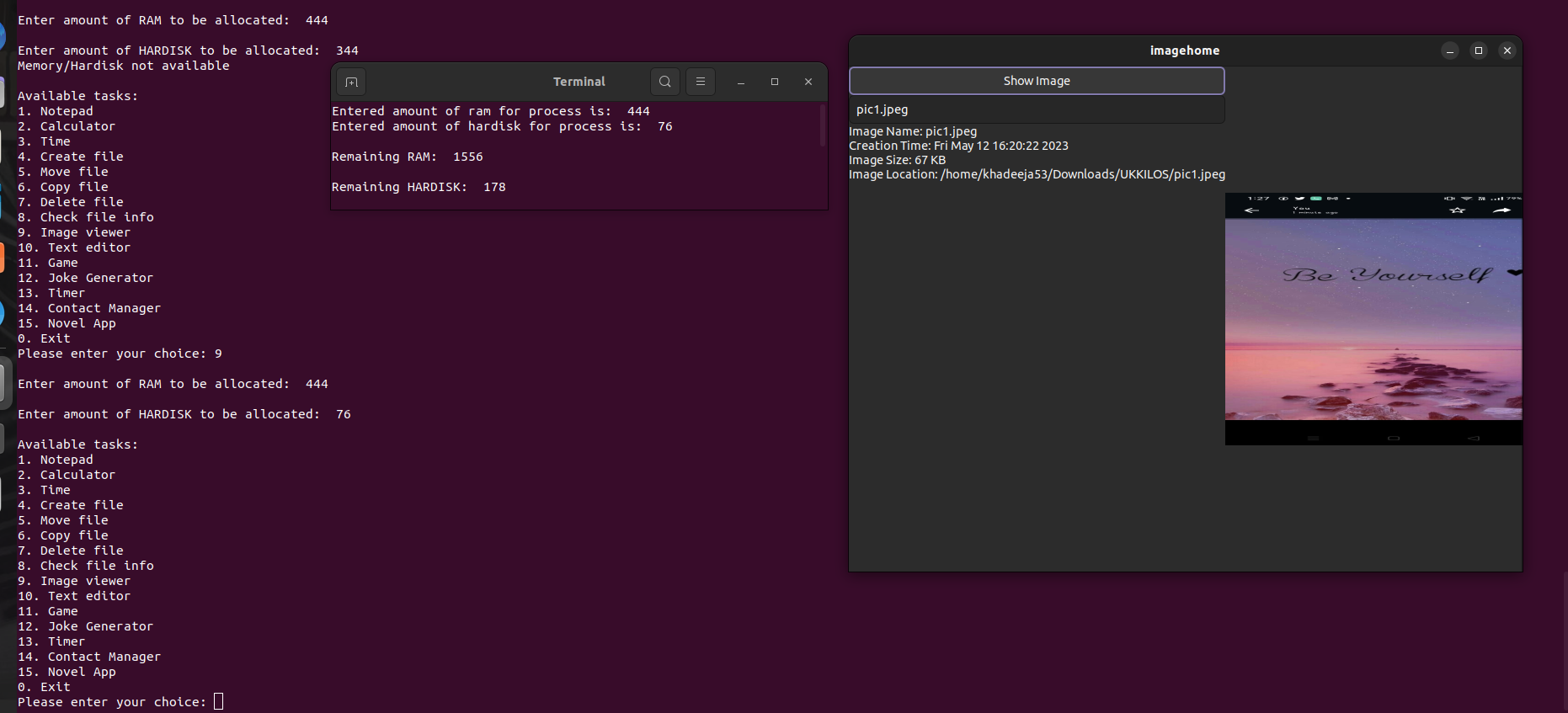
**Timer:**

****

**Tic-Tac-Toe Game:**

****

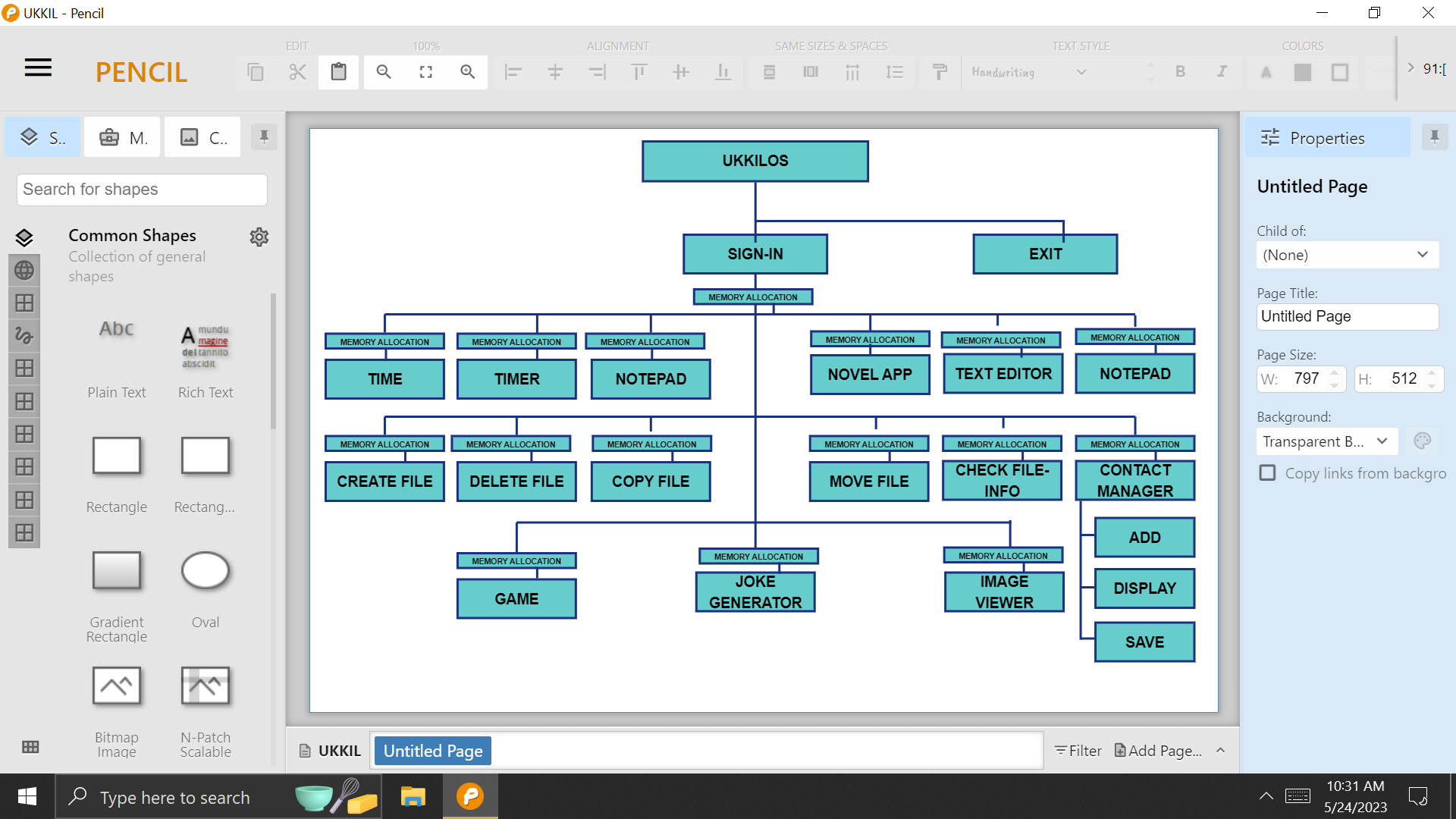
**Image Viewer:**

****

**KEY CONCEPTS:**

* Multitasking
* Context switching
* Resource allocation
* User mode and Kernel mode
* Process creation
* Threads
* EXEC commands
* Scheduling using mutual exclusion, semaphore and condition variable
* Scheduling techniques in ready queue. You should implement multilevel
* queue with different techniques on each level.

**WORK BREAK-DOWN STRUCTURE**



**GANTT-CHART**

**TIME ESTIMATION:**