



Sistemas Operacionais I - 2024.1

# OPE-RAY

## FILESYSTEM



[https://github.com/cloudywitt/AngeloGuilherme\\_FinalProject\\_OS\\_RR\\_2024](https://github.com/cloudywitt/AngeloGuilherme_FinalProject_OS_RR_2024)



**Ângelo Garcia Fernandez & Guilherme Almeida da Luz**





Definição de Sistema de  
Arquivos

---

FUSE - FileSystem in Userspace

---

Libfuse

---

Ope-Ray FileSystem

---

Principais Operações

---

Vamos para o código!

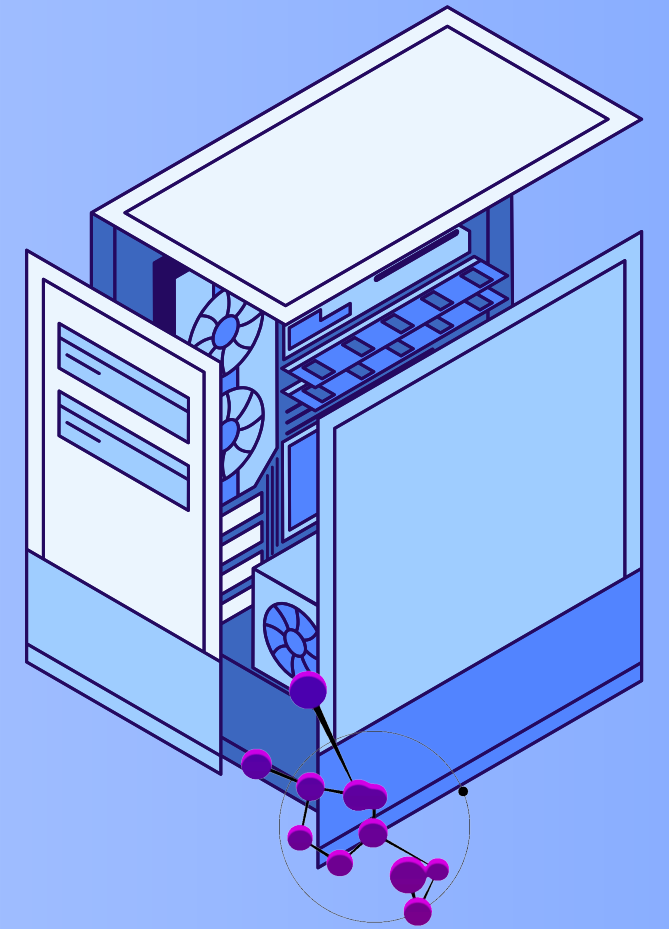
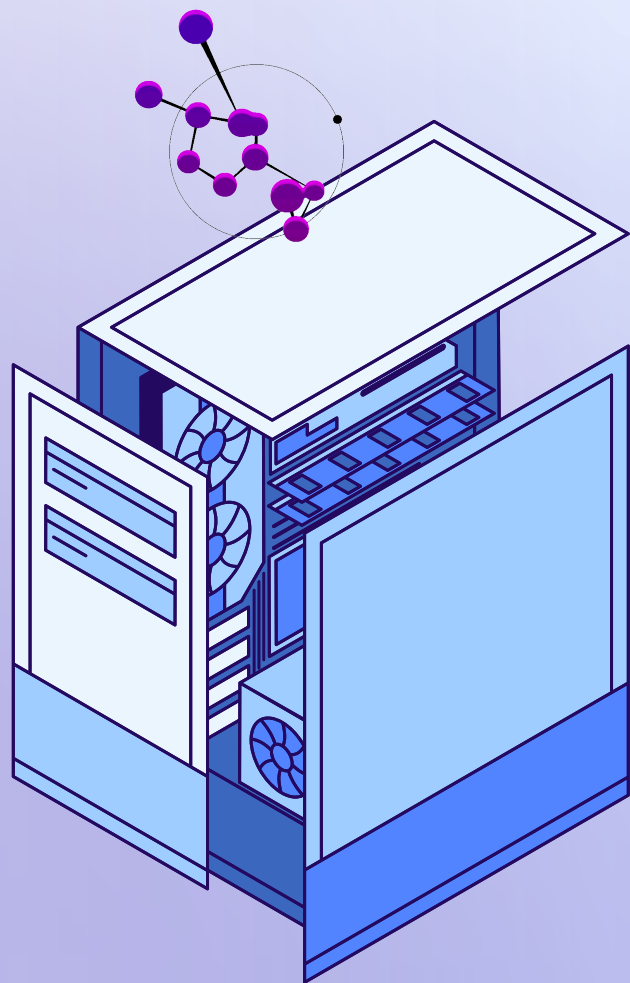


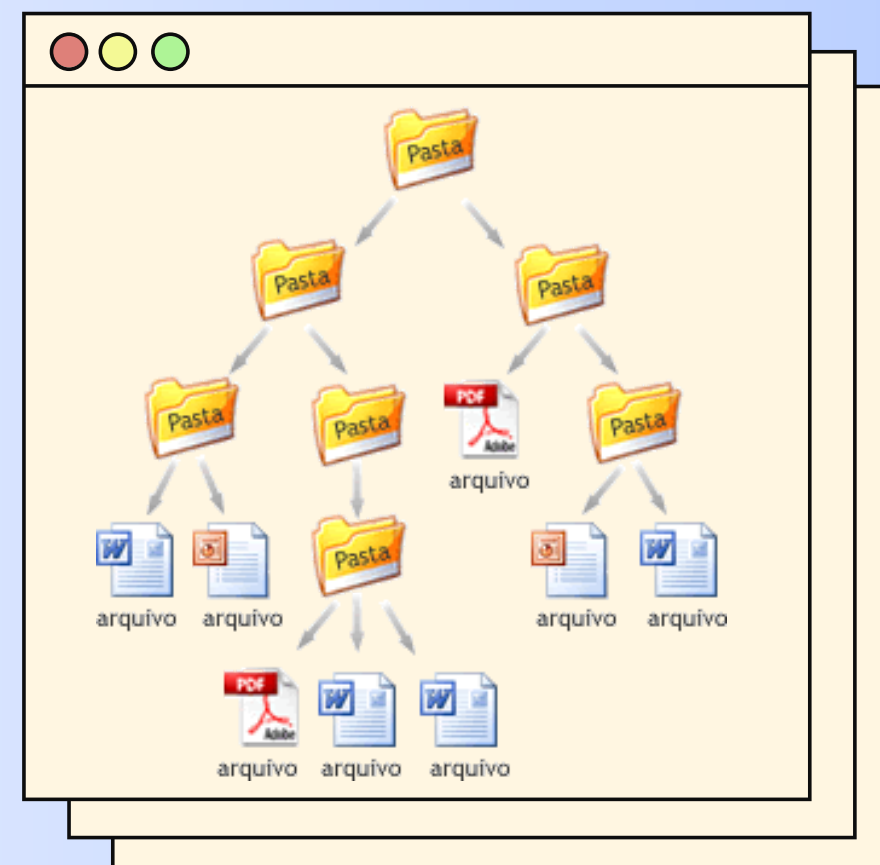
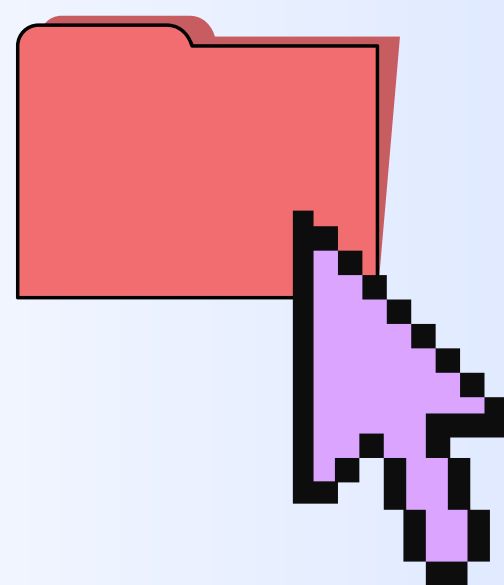
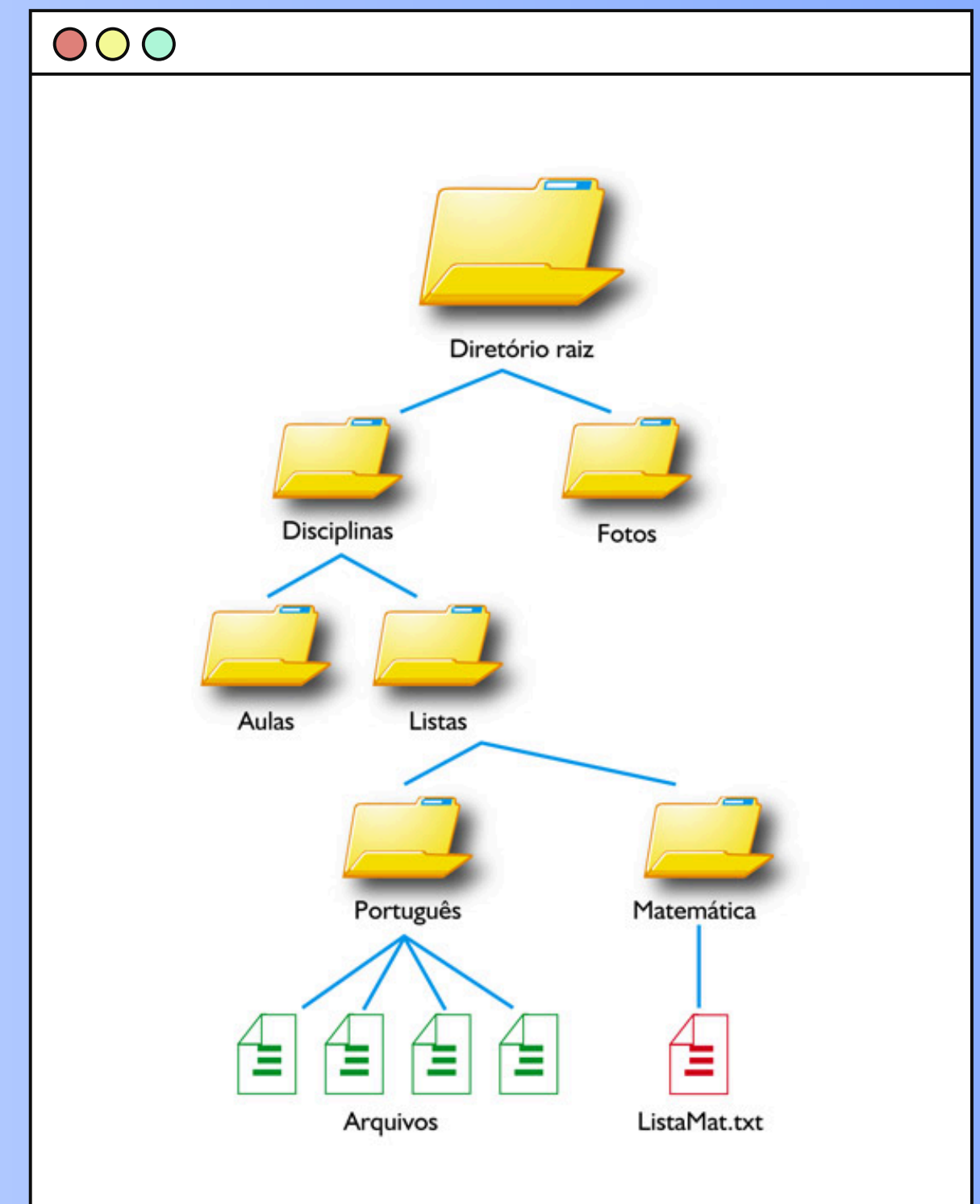
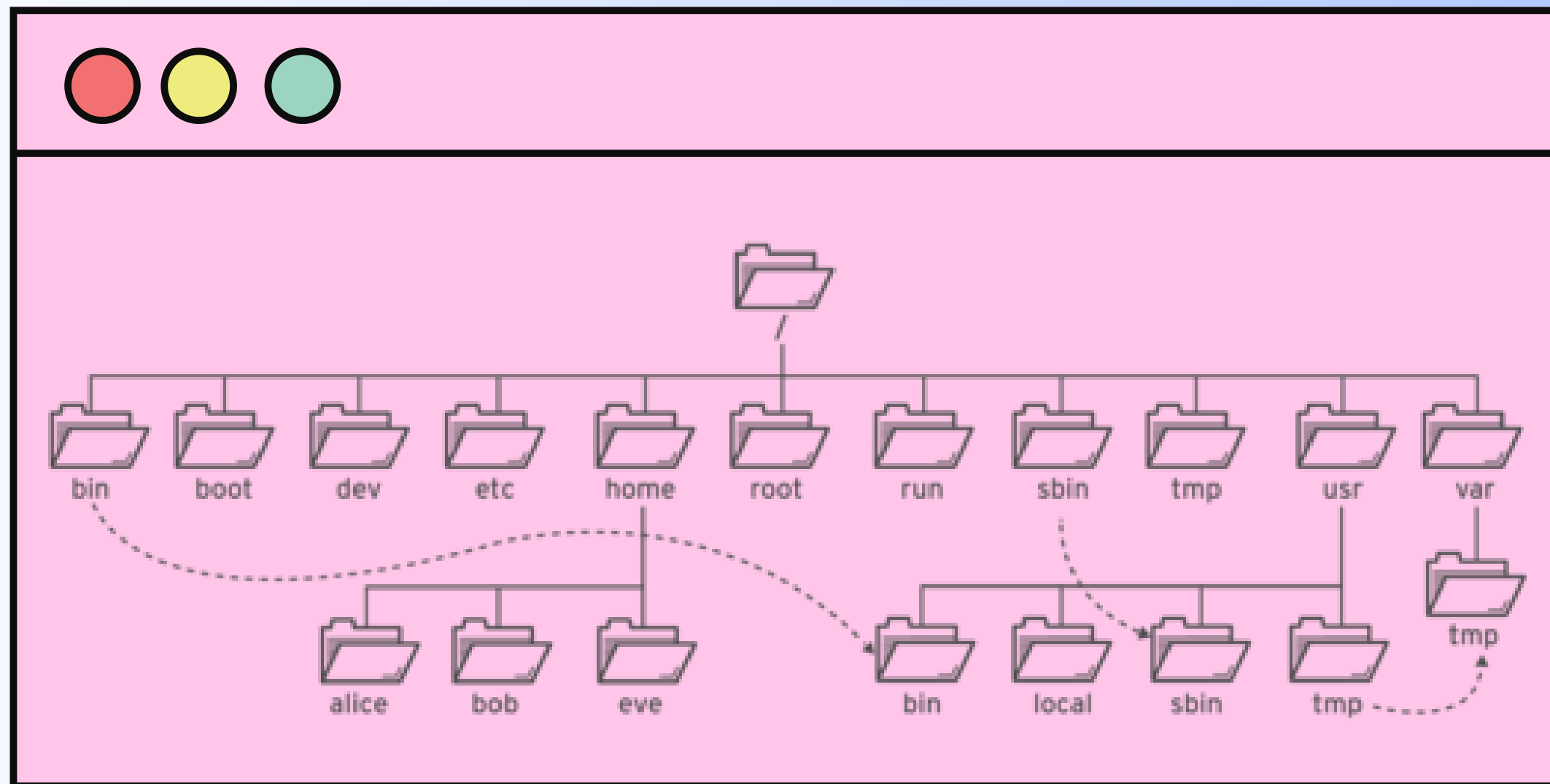
# DEFINIÇÃO DE SISTEMA DE ARQUIVOS:

É uma estrutura hierárquica (árvore de arquivo) de arquivos e diretórios, usada por um sistema operacional para organizar e gerenciar arquivos em um dispositivo de armazenamento, como um disco rígido, unidade de estado sólido (SSD) ou pendrive.

Ele define como os dados são armazenados, acessados e organizados no dispositivo de armazenamento.

Um sistema de arquivos reside em um único volume lógico. Todo arquivo e diretório pertence a um sistema de arquivos dentro de um volume lógico.







# FUSE

## FILESYSTEM IN USERSPACE



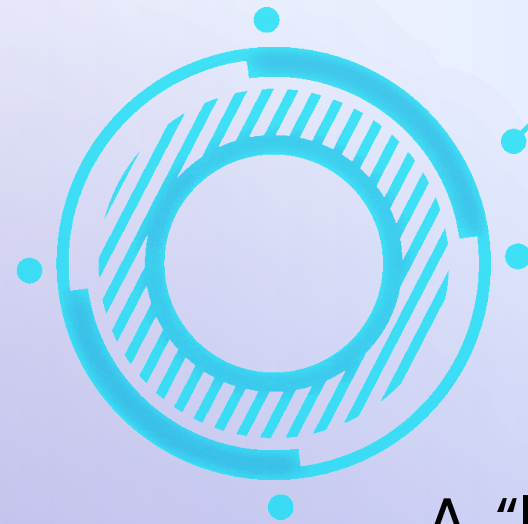
É uma interface para programas de usuário exportarem um sistema de arquivos para o kernel do Linux.

Consiste em um módulo do kernel (fuse.ko), uma biblioteca de espaço de usuário (libfuse.\*) e um utilitário de montagem (fusermount).

Um dos recursos mais importantes do FUSE é permitir montagens seguras e sem privilégios, simplificando o processo de criação e manipulação.

Um bom exemplo é sshfs: um sistema de arquivos de rede seguro usando o protocolo stfp.





# LIBFUSE

A “libfuse” fornece a referência de implementação para a comunicação com o módulo do kernel FUSE. Com funções para montar o sistema de arquivos, desmontá-lo, ler solicitações do kernel e enviar respostas de volta.

Além disso, oferece duas APIs: uma API síncrona de “alto nível” e uma API assíncrona de “baixo nível”. Em ambos os casos, as solicitações recebidas do kernel são passadas para o programa principal usando retornos de chamada.





# OPE-RAY FILESYSTEM

Sistema de arquivos simples (obtido aplicando os conceitos da interface FUSE) com funcionalidades básicas de criar, ler, escrever e apagar arquivos, bem como, árvore de diretórios.

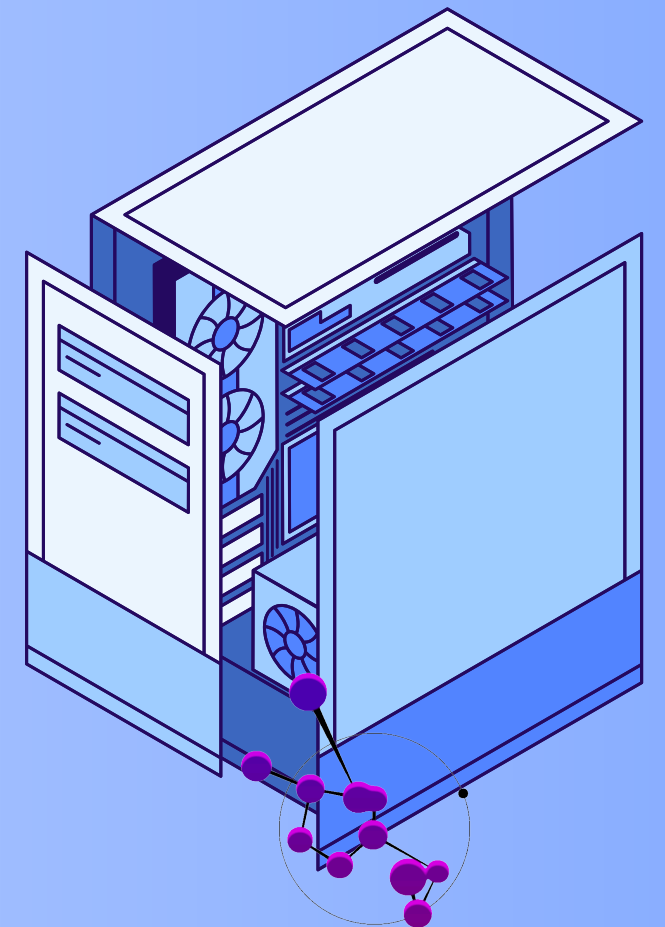
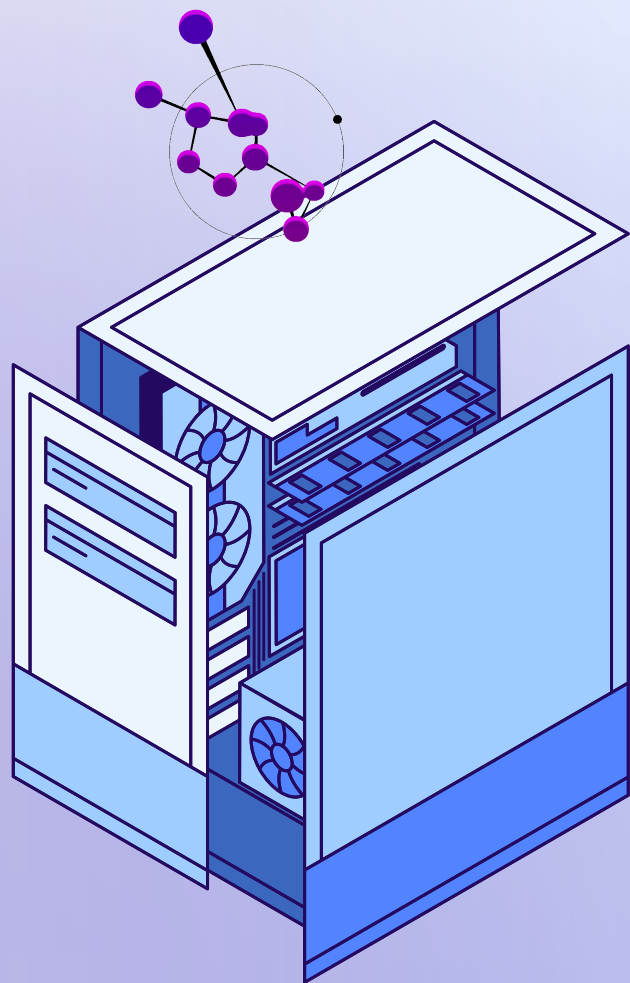




# PRINCIPAIS OPERAÇÕES:

- Read
- Write
- Make node
- Unlink
- Make dir
- Remove dir

```
.getattr = agfs_getattr,  
.readdir = agfs_readdir,  
.read = agfs_read,  
.mkdir = agfs_mkdir,  
.rmdir = agfs_rmdir,  
.mknod = agfs_mknod,  
.unlink = agfs_unlink,  
.write = agfs_write,  
.utimens = agfs_utimens,
```

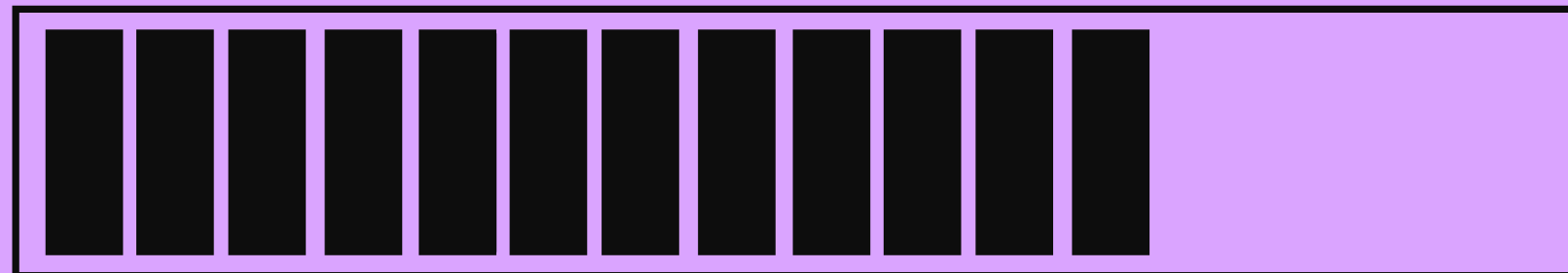






# VAMOS PARA O CÓDIGO!

● ● ● Ope-Ray FileSystem



Loading...



# LIMITAÇÕES

- Os arquivos, por serem armazenados na memória, não são persistentes; com isso, assim que o SA é desmontado ou o computador desliga, tudo o que tinha dentro dele é perdido.
- Falta de funcionalidades extras, como gerenciamento de permissões e o fornecimento de alguns dados a mais dos arquivos, como o tamanho correto e as datas de criação e acesso
- Estratégias de armazenamento dos arquivos e operações ineficientes e não tão confiáveis, por vias de simplificação

# REFERÊNCIAS:

<https://www.ibm.com/docs/pt-br/aix/7.3?topic=management-file-systems>

<https://www.kingston.com/br/blog/personal-storage/understanding-file-systems>

<https://github.com/libfuse/libfuse>

<https://github.com/torvalds/linux/blob/master/Documentation/filesystems/fuse.rst>

<https://www.controle.net/faq/o-que-e-sistema-de-arquivos-file-system>



Sistemas Operacionais I

# FIM!



**COMPONENTES:**

---

**Ângelo Garcia Fernandez  
Guilherme Almeida da Luz**