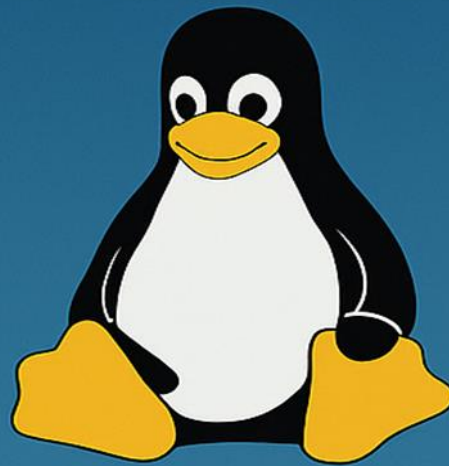
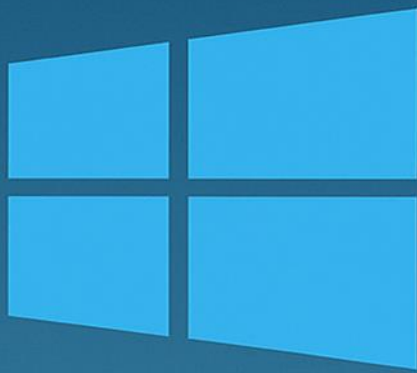


# SISTEMAS OPERACIONAIS

Windows, Linux e Android

## RELATÓRIO MÓDULO VIRTUALBOX E VMWARE



vmware

**RELATÓRIO DO PROJETO**

**MÓDULO VIRTUALBOX E VMWARE**

Disciplina: Sistemas Operacionais  
Professor: Clóvis Ferraro

Grupo 05:  
Daniel Vieira  
Matheus de Jesus Araújo  
André Soares dos Santos  
Diogo Henrique Ribeiro Lisboa

## Sumário

1. [Introdução](#)
2. [Metodologia](#)
3. [Comparação entre os Sistemas Operacionais](#)
  - 3.1 [Informações do Sistema](#)
  - 3.2 [Gerenciamento de Arquivos](#)
  - 3.3 [Gerenciamento de processos](#)
  - 3.4 [Rede](#)
  - 3.5 [Discos e Armazenamento](#)
4. [Análise Crítica](#)
5. [Conclusão](#)
6. [Autoavaliação](#)
7. [Referências](#)

## 1.Introdução

Este trabalho tem como objetivo a instalação, configuração e comparação de máquinas virtuais em diferentes sistemas operacionais, especificamente Windows, Linux e Android, utilizando as ferramentas VirtualBox e VMware. A proposta central do módulo é permitir a análise prática das características, vantagens e limitações de cada sistema operacional dentro de um ambiente controlado, sem comprometer o funcionamento do computador físico.

O uso de máquinas virtuais é de grande importância na área de tecnologia, pois possibilita a execução de múltiplos sistemas operacionais em um mesmo hardware, permitindo estudos, testes e simulações de situações reais. Essa prática contribui para o aprendizado acadêmico, a preparação profissional e o desenvolvimento de competências técnicas, uma vez que oferece maior segurança, flexibilidade e isolamento ao realizar atividades como a configuração de redes, servidores e aplicações.

Dessa forma, este módulo busca não apenas aplicar a virtualização como recurso técnico, mas também evidenciar sua relevância no contexto educacional e no mercado de trabalho em tecnologia.

## 2. Metodologia

Para a realização deste trabalho, foram utilizados os programas de virtualização VirtualBox e VMware Workstation. Esses softwares permitem a criação de ambientes virtuais isolados, possibilitando a instalação e utilização de diferentes sistemas operacionais sem afetar o sistema principal do computador.

Os sistemas operacionais instalados foram: Linux, Android e Windows, utilizando arquivos ISO, que contêm todos os arquivos necessários para a instalação. Cada arquivo ISO foi carregado em uma máquina virtual previamente configurada com parâmetros específicos de hardware:

CPU: 4 Núcleos;

Memória RAM: 8gb;

Disco rígido: 12gb;

O processo de instalação seguiu os procedimentos padrão de cada sistema operacional, utilizando os instaladores das ISOs para completar a configuração inicial. Após a instalação, cada máquina virtual estava pronta para execução e realização dos testes.

Os testes foram conduzidos por meio dos terminais ou linhas de comando de cada sistema. Foram executados comandos comuns relacionados a gerenciamento de arquivos, processos, rede, discos e usuários. Cada comando foi registrado e comparado entre os sistemas Linux, Android e Windows, destacando semelhanças, diferenças e particularidades de cada ambiente.

Todos os procedimentos foram documentados detalhadamente, incluindo: comandos executados, versões dos sistemas operacionais e configurações das máquinas virtuais, garantindo a replicabilidade do estudo.

Esta metodologia permitiu não apenas a avaliação da instalação e configuração de sistemas operacionais em ambientes virtuais, mas também o entendimento das funcionalidades e limitações de cada sistema operacional.

### 3.Comparação entre os Sistemas Operacionais

Nesta seção, serão apresentados os resultados de testes realizados em três sistemas operacionais distintos: Linux, Android e Windows. O objetivo é comparar o comportamento de cada sistema em diferentes situações práticas, permitindo identificar semelhanças, diferenças e particularidades de cada ambiente.

Os testes foram conduzidos considerando os seguintes aspectos:

1. Informações do Sistema – análise de comandos e ferramentas que fornecem dados sobre hardware e software de cada sistema;
2. Gerenciamento de Arquivos – avaliação das operações de criação, exclusão, cópia e organização de arquivos;
3. Gerenciamento de Processos – verificação de execução, monitoramento e encerramento de processos em cada sistema;
4. Rede – testes relacionados à configuração, monitoramento e diagnóstico de conexões de rede;
5. Discos e Armazenamento – análise de utilização, formatação e gerenciamento de dispositivos de armazenamento.

Através desses testes, será possível compreender de forma prática como cada sistema operacional lida com comandos, recursos e funções específicas. A comparação detalhada permitirá avaliar as diferenças de implementação, eficiência e flexibilidade entre Linux, Android e Windows, contribuindo para um entendimento mais aprofundado de cada ambiente.

### 3.1 Informações do Sistema

A análise de informações do sistema envolve a utilização de comandos e ferramentas específicas que permitem extrair dados detalhados sobre o hardware e o software presentes em cada sistema operacional. Entre os dados obtidos, destacam-se o modelo e a quantidade de processadores, a capacidade total de memória RAM, o espaço disponível e ocupado em disco, a versão e build do sistema operacional, o tempo de atividade, informações sobre placas de rede, dispositivos conectados e detalhes do kernel ou núcleo do sistema.

Essas informações são fundamentais para o monitoramento, gerenciamento e diagnóstico de máquinas virtuais, pois permitem identificar possíveis gargalos de desempenho, compatibilidade de hardware e requisitos de software. A execução desses testes proporciona uma visão abrangente da configuração de cada sistema, permitindo não apenas entender como os sistemas Linux, Android e Windows apresentam esses dados, mas também como cada um organiza e prioriza a exibição dessas informações.

Além disso, a coleta de dados detalhados facilita comparações entre os sistemas, oferecendo insights sobre diferenças estruturais e funcionais, como a forma como cada sistema operacional gerencia recursos, mantém registros de uso e disponibiliza ferramentas de suporte. Esse levantamento é essencial para subsidiar análises posteriores de desempenho, otimização, manutenção e planejamento de atualizações, garantindo que a gestão das máquinas virtuais seja baseada em informações precisas e confiáveis.



## Linux

No Linux, o comando (`uname -a`) exibe informações do kernel, versão e arquitetura, mostrando detalhes técnicos do núcleo do sistema que permitem entender seu funcionamento interno, algo que Windows não fornece diretamente.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ lsb_release -a  
No LSB modules are available.  
Distributor ID: Kali  
Description:    Kali GNU/Linux Rolling  
Release:        2025.2  
Codename:       kali-rolling  
  
(kali@kali)-[~]  
$ free -h  
              total        used        free      shared  buff/cache   availa  
ble  
Mem:          1.9Gi         1.4Gi         164Mi         62Mi         611Mi         53  
2Mi  
Swap:         953Mi        339Mi         614Mi  
  
(kali@kali)-[~]  
$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
udev            921M   0    921M   0% /dev  
tmpfs           198M  976K  197M   1% /run  
/dev/sda1       79G   15G   60G  20% /  
tmpfs           987M   4.0K  987M   1% /dev/shm  
tmpfs           5.0M   0    5.0M   0% /run/lock  
tmpfs           1.0M   0    1.0M   0% /run/credentials/systemd-journald.servi  
ce  
tmpfs           987M   2.0M  985M   1% /tmp  
tmpfs           1.0M   0    1.0M   0% /run/credentials/getty@tty1.service
```

O (`lsb_release -a`) fornece informações precisas da distribuição, como nome, versão e codename, permitindo compatibilidade de pacotes e atualizações.

O (`free -h`) detalha a memória RAM e Swap, mostrando uso, disponibilidade e cache de forma legível, facilitando o monitoramento em tempo real. Por fim, o

(`df -h`) exibe o uso de disco por partição, incluindo sistemas de arquivos temporários e dispositivos, oferecendo um panorama completo do armazenamento.

Em síntese, o Linux permite uma análise técnica abrangente do sistema diretamente pelo terminal, integrando informações que no Windows exigiriam múltiplos comandos ou interfaces gráficas.





## Windows

No Windows, a obtenção de informações sobre o sistema pode ser realizada por meio de comandos no Prompt de Comando (CMD) ou utilizando ferramentas gráficas, como o Painel de Controle e o Gerenciador de Tarefas. Entre os comandos mais utilizados estão (**systeminfo**), que fornece dados detalhados sobre a versão do sistema operacional, arquitetura, processador, memória física disponível e tempo de atividade.

```
C:\Windows\system32\cmd.exe
c:\Users\Pichau>systeminfo

Nome do host: DESKTOP-M81U9SG
Nome do sistema operacional: Microsoft Windows 10 Pro
Versão do sistema operacional: 10.0.19045 N/A compilação 19045
Fabricante do sistema operacional: Microsoft Corporation
Configuração do SO: Estação de trabalho autônoma
Tipo de compilação do sistema operacional: Multiprocessor Free
Proprietário registrado: hh
Organização registrada:
Identificação do produto: 00331-10000-00001-AA248
Data da instalação original: 28/03/2024, 14:40:46
Tempo de Inicialização do Sistema: 04/09/2025, 14:33:05
Fabricante do sistema: To be filled by O.E.M.
Modelo do sistema: To be filled by O.E.M.
Tipo de sistema: x64-based PC
Processador(es): 1 processador(es) instalado(s).
[01]: Intel64 Family 6 Model 58 Stepping 9 GenuineIntel ~3392 Mhz
Versão do BIOS: American Megatrends Inc. 4.6.5, 16/10/2023
Pasta do Windows: C:\Windows
Pasta do sistema: C:\Windows\system32
Iniciar dispositivo: \Device\HarddiskVolume1
Localidade do sistema: pt-br:Português (Brasil)
Localidade de entrada: pt-br:Português (Brasil)
Fuso horário: (UTC-03:00) Brasília
Memória física total: 16.349 MB
Memória física disponível: 7.674 MB
Memória Virtual: Tamanho Máximo: 27.203 MB
Memória Virtual: Disponível: 15.763 MB
Memória Virtual: Em Uso: 11.440 MB
Local(is) de arquivo de paginação: C:\pagefile.sys
Domínio: WORKGROUP
Servidor de Logon: \\DESKTOP-M81U9SG
Hotfix(es): 24 hotfix(es) instalado(s).
[01]: KB5064400
[02]: KB5034468
[03]: KB5011048
[04]: KB5011067
[05]: KB5015684
[06]: KB5063842
[07]: KB5014032
[08]: KB5032907
[09]: KB5034224
[10]: KB5043935
[11]: KB5043130
[12]: KB5046823
[13]: KB5050388
[14]: KB5050111
[15]: KB5052916
[16]: KB5054682
[17]: KB5055663
[18]: KB5058526
[19]: KB5059504
[20]: KB5061902
[21]: KB5063706
[22]: KB5063261
[23]: KB5063979
[24]: KB5063523
Placa(s) de Rede: 4 NIC(s) instalado(s).
[01]: Realtek PCIe GbE Family Controller
Nome da conexão: Ethernet 2
DHCP ativado: Sim
Servidor DHCP: 192.168.18.1
```

Esses testes fornecem uma visão completa da configuração do Windows, permitindo identificar a capacidade de hardware alocada na máquina virtual, verificar detalhes do sistema operacional e avaliar recursos essenciais para o desempenho.



## Android

No Android, a obtenção de informações sobre o sistema é realizada principalmente por meio do Terminal (ADB SHELL) e das Configurações do Sistema. Comandos como ([getprop](#)) fornecem dados detalhados sobre a versão do sistema operacional, modelo do dispositivo, processador, memória disponível, armazenamento e outras propriedades essenciais do hardware e software. Aplicativos de monitoramento do sistema também podem ser utilizados para apresentar essas informações de forma gráfica, facilitando a visualização de parâmetros críticos.

```
3:39 [selinux.restorecon_recursive]: [/data/misc_ce/0]
[service.bootanim.exit]: [1]
[service.sf.present_timestamp]: [1]
[sleep.state]: [none]
[status.battery.level]: [5]
[status.battery.level_raw]: [50]
[status.battery.level_scale]: [9]
[status.battery.state]: [Slow]
[sys.boot.reason]: [reboot]
[sys.boot_completed]: [1]
[sys.logbootcomplete]: [1]
[sys.media.vdec.drop]: [0]
[sys.rescue_boot_count]: [1]
[sys.retaildemo.enabled]: [0]
[sys.sysctl.extra_free_kbytes]: [9216]
[sys.usb.config]: [adb]
[sys.usb.configfs]: [0]
[sys.usb.state]: [adb]
[sys.user.0.ce_available]: [true]
[sys.wifitracing.started]: [1]
[tombstoned.max_tombstone_count]: [50]
[vold.has_adoptable]: [1]
[vold.has_quota]: [0]
[vold.has_reserved]: [0]
[vold.post_fs_data_done]: [1]
:/ $
```

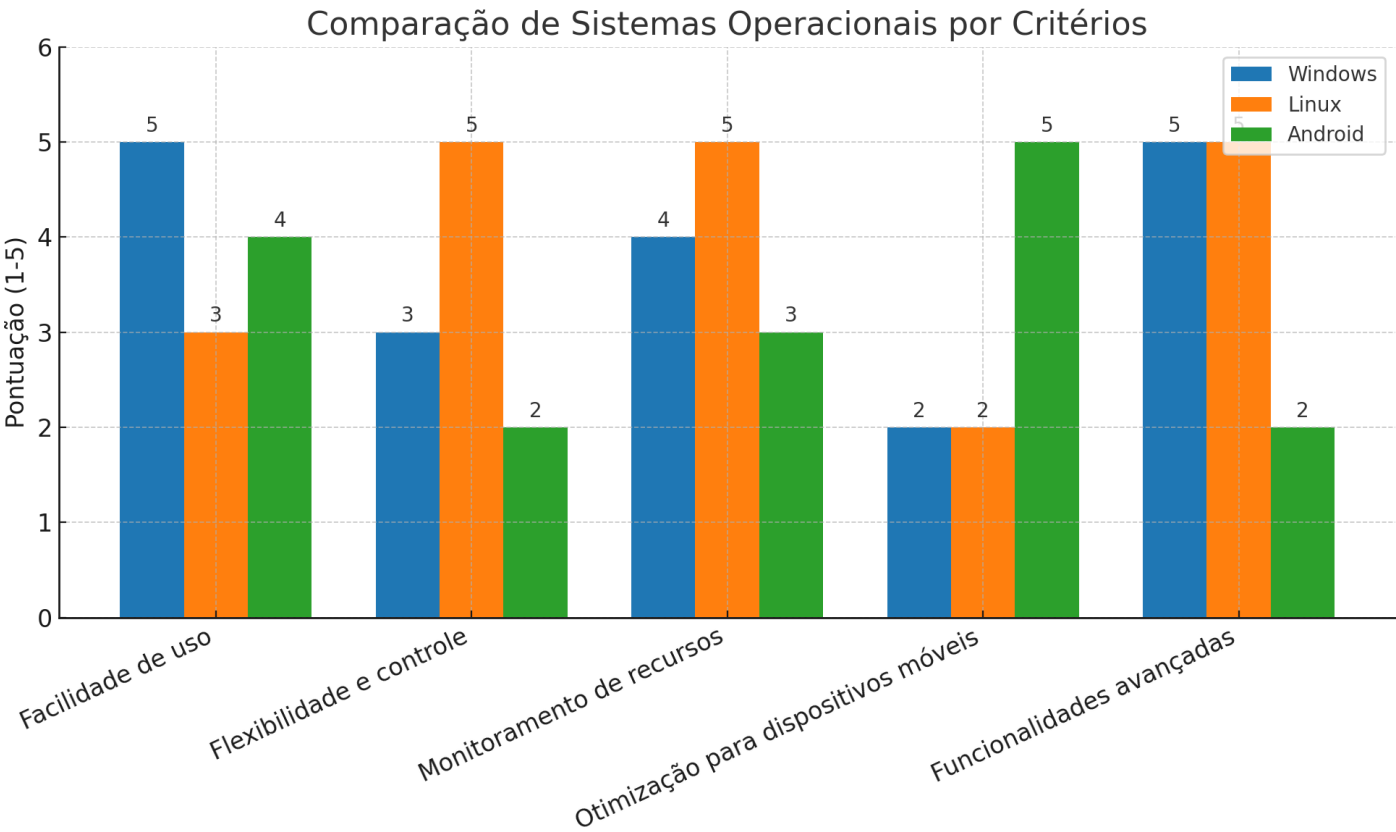
Esses testes permitem compreender a configuração interna do Android, incluindo limitações e capacidades do dispositivo virtualizado. Além disso, fornecem base para comparar como o Android apresenta informações sobre hardware e software em relação ao Windows e Linux, destacando diferenças na organização e no acesso a dados do sistema, essenciais para diagnósticos e otimizações.

### Comparação Crítica

Nesta seção, são comparados os resultados obtidos nos testes realizados em Linux, Android e Windows, considerando os aspectos de informações do sistema, gerenciamento de arquivos, processos, rede e discos/armazenamento.

A análise crítica permite identificar semelhanças entre os sistemas, como a presença de comandos ou ferramentas para monitoramento de hardware e software, controle de processos e gerenciamento de arquivos. Também evidencia diferenças significativas, como a forma de apresentar informações, a nomenclatura dos comandos e a flexibilidade de configuração, que varia conforme cada sistema operacional.

Além disso, a comparação evidencia vantagens específicas de cada sistema. O Windows apresenta interfaces gráficas intuitivas e ferramentas integradas que facilitam a obtenção de informações e o gerenciamento do sistema. O Linux se destaca pela flexibilidade e detalhamento fornecido por comandos avançados, permitindo um controle mais profundo do ambiente. O Android, por sua vez, oferece informações adaptadas a dispositivos móveis, com foco em otimização de recursos e monitoramento simplificado, mas limitado em relação a funcionalidades avançadas de desktop.



## 3.2 Gerenciamento de Arquivos

O gerenciamento de arquivos consiste em um conjunto de operações, ferramentas e procedimentos que permitem organizar, criar, acessar, modificar, copiar, mover e excluir arquivos e diretórios dentro de um sistema computacional. Trata-se de uma função central para a administração de dados, pois garante que a informação esteja disponível de maneira eficiente, estruturada e segura, prevenindo perdas, redundâncias ou acessos não autorizados.

Além das operações básicas, o gerenciamento de arquivos envolve o controle detalhado de atributos, como permissões de leitura, gravação e execução, datas de criação e modificação, tamanhos de arquivos e tipos de formatos. Também inclui a manutenção de hierarquias de diretórios, permitindo que os dados sejam armazenados de forma organizada e que a navegação entre pastas seja clara e intuitiva. A correta organização de arquivos facilita tarefas complexas, como busca e recuperação de informações, monitoramento do espaço de armazenamento e execução de processos automatizados que dependem de dados estruturados.

Outro aspecto importante do gerenciamento de arquivos é a segurança da informação. Através de políticas de controle de acesso e backups regulares, é possível proteger dados críticos contra perdas acidentais, falhas de hardware, corrupção de arquivos ou acessos indevidos. Além disso, práticas eficientes de gerenciamento contribuem para otimizar o desempenho do sistema, evitando sobrecarga de armazenamento, fragmentação de arquivos e redundâncias que possam impactar na velocidade de acesso aos dados.

O estudo e a análise do gerenciamento de arquivos fornecem uma base sólida para entender como os dados são tratados, organizados e preservados dentro de qualquer ambiente computacional. Essa compreensão é essencial não apenas para a operação cotidiana, mas também para diagnósticos avançados, planejamento de armazenamento, estratégias de recuperação de informações e implementação de boas práticas de manutenção e otimização de sistemas.



## Linux

No Linux, o gerenciamento de arquivos é realizado principalmente por meio de comandos de terminal, que permitem criar, localizar, mover, copiar e excluir arquivos e diretórios de forma eficiente. Entre os comandos mais utilizados estão (**find**), que permite localizar arquivos ou pastas com base em critérios específicos, (**cp**) para copiar arquivos, (**mv**) para mover ou renomear, (**rm**) para remover arquivos e **mkdir** para criar novos diretórios.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ find / -name arquivo.txt  
find: '/run/udisks2': Permission denied  
find: '/run/lightdm': Permission denied  
find: '/run/user/1000/systemd/inaccessible/dir': Permission denied  
find: '/run/sudo': Permission denied  
find: '/run/openvpn-server': Permission denied  
find: '/run/openvpn-client': Permission denied  
find: '/run/cryptsetup': Permission denied  
find: '/run/credentials/getty@tty1.service': Permission denied  
find: '/run/credentials/systemd-journald.service': Permission denied  
find: '/run/systemd/propagate/upower.service': Permission denied  
find: '/run/systemd/propagate/colord.service': Permission denied  
find: '/run/systemd/propagate/rtkit-daemon.service': Permission denied  
find: '/run/systemd/propagate/ModemManager.service': Permission denied  
find: '/run/systemd/propagate/NetworkManager.service': Permission denied  
find: '/run/systemd/propagate/systemd-logind.service': Permission denied  
find: '/run/systemd/propagate/polkit.service': Permission denied  
find: '/run/systemd/propagate/accounts-daemon.service': Permission denied  
find: '/run/systemd/propagate/haveged.service': Permission denied  
find: '/run/systemd/propagate/systemd-udev.service': Permission denied  
find: '/run/systemd/inaccessible/dir': Permission denied  
find: '/run/initramfs': Permission denied  
find: '/usr/lib/mysql/plugin/auth_pam_tool_dir': Permission denied  
find: '/proc/tty/driver': Permission denied  
find: '/proc/1/task/1/fd': Permission denied  
find: '/proc/1/task/1/fdinfo': Permission denied  
find: '/proc/1/task/1/ns': Permission denied
```

O uso desses comandos permite ao usuário ter controle completo sobre a organização do sistema de arquivos, incluindo permissões de acesso, propriedades dos arquivos e hierarquias de diretórios

Além de manipular arquivos e diretórios, esses comandos possibilitam automação de tarefas e manutenção do sistema, como limpeza de arquivos temporários ou backup de dados. Essa abordagem baseada em terminal é poderosa, pois permite operações rápidas e precisas, mesmo em estruturas de arquivos complexas, sendo essencial para o monitoramento e gerenciamento de sistemas Linux de forma eficiente.



## Windows

No Windows, o gerenciamento de arquivos é realizado principalmente através do Explorador de Arquivos e de comandos no Prompt de Comando (CMD), permitindo ao usuário organizar, acessar e manipular pastas e arquivos de forma eficiente. A estrutura de diretórios é hierárquica, começando normalmente pelo disco rígido como a unidade C: e ramificando-se em pastas e subpastas que armazenam arquivos de sistema, aplicativos e dados do usuário.

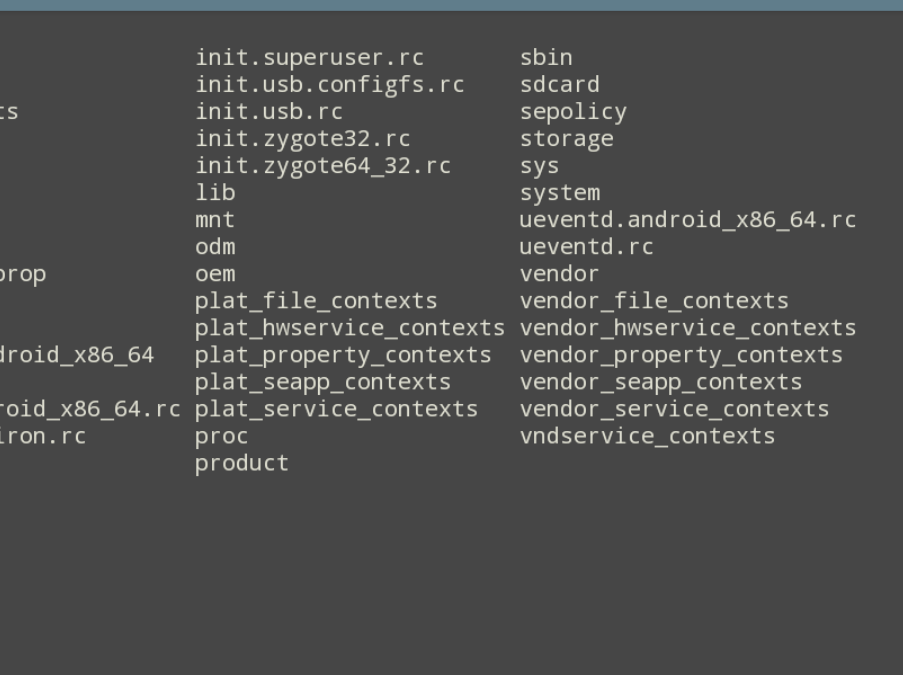
```
Selecionar C:\Windows\system32\cmd.exe
C:\Users\Pichau>dir
O volume na unidade C não tem nome.
O Número de Série do Volume é 8808-F4E1

Pasta de C:\Users\Pichau

06/09/2025 02:00 <DIR> .
06/09/2025 02:00 <DIR> ..
07/11/2024 14:35 <DIR> .android
06/09/2025 02:44      636 .bash_history
04/09/2025 15:01      190 .gitconfig
04/09/2025 13:47 <DIR> .VirtualBox
27/04/2025 00:04 <DIR> .vscode
08/02/2024 14:02 <DIR> 3D Objects
13/09/2024 00:32 <DIR> ansel
08/02/2024 14:02 <DIR> Contacts
04/09/2025 14:59 <DIR> Desktop
04/09/2025 15:07 <DIR> Documents
06/09/2025 01:57 <DIR> Downloads
08/02/2024 14:02 <DIR> Favorites
26/05/2025 00:40 <DIR> ibisPaint
08/02/2024 14:02 <DIR> Links
08/02/2024 14:02 <DIR> Music
29/04/2025 02:24 <DIR> myenv
20/04/2025 02:22 <DIR> OneDrive
09/11/2024 00:36 <DIR> Pictures
08/02/2024 14:02 <DIR> Saved Games
08/02/2024 14:03 <DIR> Searches
06/09/2025 01:01 <DIR> Videos
                2 arquivo(s)      826 bytes
                21 pasta(s) 208.375.111.680 bytes disponíveis
C:\Users\Pichau>
```

O comando (**dir**), exibido na imagem, é uma ferramenta básica do CMD que lista o conteúdo de uma pasta, mostrando informações como nome, tipo (arquivo ou diretório), tamanho e data de modificação. No exemplo, podemos observar que a pasta do usuário contém diretórios padrão como Desktop, Documents, Downloads, Pictures, Videos e outras pastas de configuração de aplicativos, como VSCode e VirtualBox. Esses diretórios organizam os arquivos de forma lógica, permitindo que o usuário encontre rapidamente o que precisa.

Portanto, o gerenciamento de arquivos no Windows combina recursos gráficos e comandos de linha de comando, oferecendo flexibilidade para organização, acesso rápido e manutenção eficiente dos dados do sistema e do usuário. Essa abordagem facilita tanto o uso cotidiano quanto tarefas mais avançadas de administração do sistema.



```
3:40 [Icons]
Window 1
+ X ⋮

:/$ ls
acct          init.superuser.rc      sbin
bin           init.usb.configfs.rc   sdcard
bugreports    init.usb.rc            sepolicy
cache         init.zygote32.rc       storage
charger       init.zygote64_32.rc    sys
config        lib                    system
d             mnt                    ueventd.android_x86_64.rc
data          odm                    ueventd.rc
default.prop  oem                    vendor
dev           plat_file_contexts     vendor_file_contexts
etc           plat_hwservice_contexts vendor_hwservice_contexts
fstab.android_x86_64 plat_property_contexts vendor_property_contexts
init          plat_seapp_contexts    vendor_seapp_contexts
init.android_x86_64.rc plat_service_contexts  vendor_service_contexts
init.envIRON.rc proc                    vndservice_contexts
init.rc       product

:/$ █
```

Além disso, a listagem de arquivos com **(ls)** ajuda no monitoramento do espaço ocupado por cada pasta, facilitando a manutenção da organização do dispositivo.



## Comparação Crítica

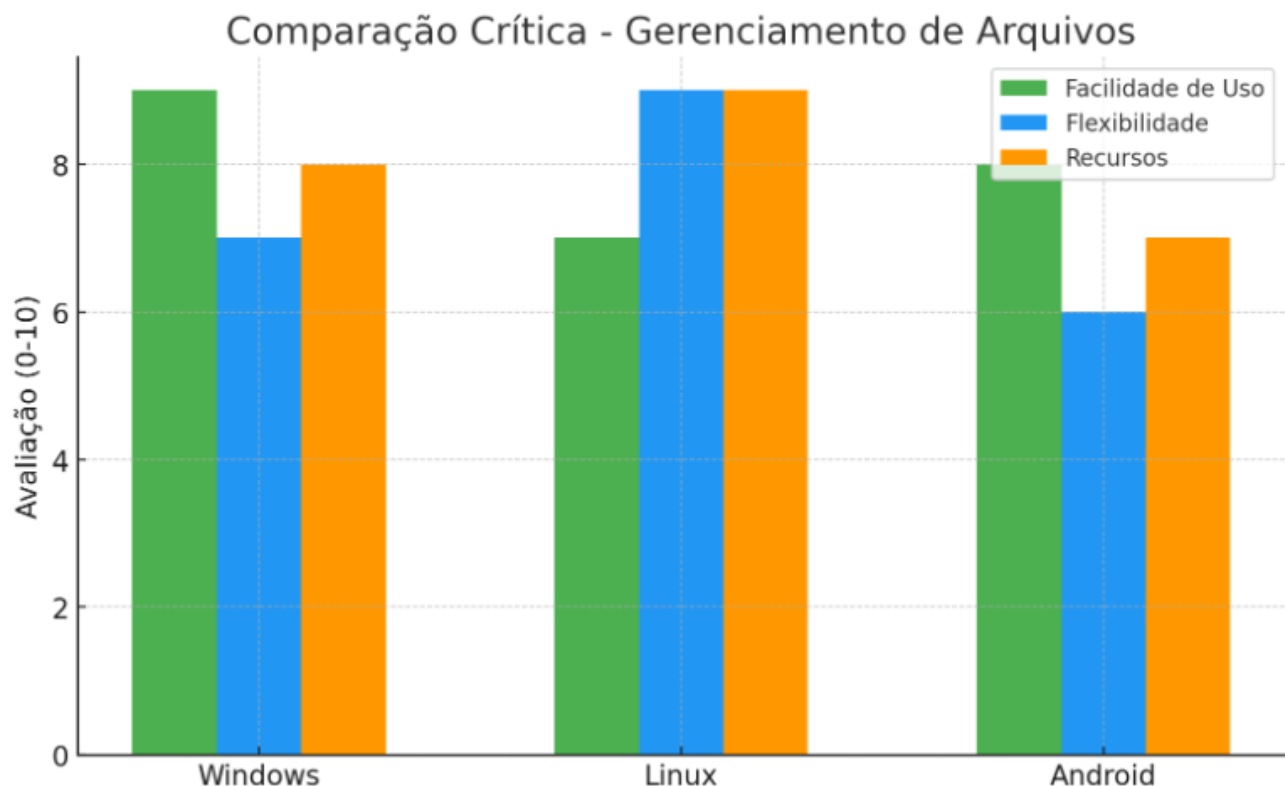
Na análise do gerenciamento de arquivos entre Windows, Linux e Android, é possível identificar diferenças significativas, assim como algumas semelhanças.

O Windows se destaca pela facilidade de uso, oferecendo uma interface gráfica intuitiva que facilita a navegação e a organização de pastas e arquivos, tornando o sistema acessível para usuários menos experientes. Ele possui recursos integrados de gerenciamento de arquivos, como o Explorer, que permite ações rápidas de cópia, movimentação e pesquisa.

O Linux, por outro lado, apresenta maior flexibilidade e recursos avançados para gerenciamento de arquivos, principalmente através do terminal e comandos como ls. Essa abordagem oferece um controle mais detalhado sobre arquivos e permissões, sendo vantajosa para usuários avançados que buscam personalização e automação de tarefas.

O Android combina aspectos de ambos os sistemas, oferecendo uma interface amigável para o usuário e acesso aos arquivos através de aplicativos de gerenciamento. Entretanto, sua flexibilidade e recursos são limitados em comparação ao Linux, especialmente em funções avançadas de manipulação de arquivos.

Em resumo, o Windows prioriza usabilidade, o Linux destaca-se pela flexibilidade e recursos avançados, e o Android equilibra facilidade e praticidade, mas com limitações em comparação aos demais sistemas.





### 3.3 Gerenciamento de Processos

O gerenciamento de processos é uma das funções centrais de qualquer sistema computacional, responsável por coordenar e controlar a execução das tarefas em um ambiente multitarefa. Um processo pode ser definido como a instância de um programa em execução, abrangendo não apenas o código do programa, mas também seu estado, memória alocada, recursos utilizados e informações sobre o contexto de execução.

O gerenciamento de processos envolve várias atividades críticas, entre as quais se destacam a criação, escalonamento, sincronização, comunicação e finalização de processos. A criação de processos consiste em gerar novas instâncias de programas que passarão a competir pelos recursos do sistema. Já o escalonamento é o mecanismo que determina a ordem de execução dos processos, buscando otimizar o uso da CPU e reduzir o tempo de resposta, mantendo o equilíbrio entre eficiência e justiça no acesso aos recursos.

A sincronização de processos é fundamental para evitar conflitos no acesso a recursos compartilhados, como memória, arquivos e dispositivos de entrada/saída. Técnicas como semáforos, e monitores são utilizadas para garantir que dois ou mais processos não modifiquem simultaneamente dados críticos, prevenindo condições de corrida e inconsistências.

A comunicação entre processos também desempenha um papel essencial, principalmente quando diferentes tarefas precisam trocar informações para cumprir objetivos comuns. Isso pode ser feito de forma direta, por meio de troca de mensagens, ou indireta, utilizando mecanismos como filas e buffers compartilhados.

Além disso, o gerenciamento de processos deve lidar com situações de bloqueio e deadlock, garantindo que processos não fiquem permanentemente impedidos de progredir. Estratégias preventivas e detectivas são aplicadas para minimizar ou resolver essas situações, assegurando a continuidade do funcionamento do sistema.



## Linux

No Linux, o gerenciamento de processos é um aspecto fundamental para garantir que o sistema funcione de maneira eficiente e organizada. Ele envolve o controle de todas as tarefas que estão sendo executadas, desde processos do sistema até aplicações iniciadas pelo usuário. Cada processo possui um identificador único, prioridades e estados que indicam se ele está em execução, esperando por recursos ou suspenso.

O sistema fornece ferramentas que permitem visualizar e monitorar esses processos de forma detalhada. Por exemplo, o comando (**ps aux**) apresenta uma lista completa de todos os processos ativos no sistema, incluindo informações sobre o usuário que iniciou o processo, uso de CPU e memória, estado e tempo de execução. Essa visualização é essencial para compreender como os recursos do sistema estão sendo distribuídos e para identificar processos que possam estar consumindo recursos excessivos.

```
(kali@kali)-[~]
$ ps aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.5	23756	12044	?	Ss	Sep05	0:03	/sbin/init splash
root	2	0.0	0.0	0	0	?	S	Sep05	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	S	Sep05	0:00	[pool_workqueue_release]
root	4	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-kvfree_rcu_reclaim]
root	5	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-rcu_gp]
root	6	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-sync_wq]
root	7	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-slub_flushwq]
root	8	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-netns]
root	11	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/0:0H-kblockd]
root	12	0.0	0.0	0	0	?	I	Sep05	0:00	[kworker/u8:0-ipv6_addrconf]
root	13	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-mm_percpu_wq]
root	14	0.0	0.0	0	0	?	I	Sep05	0:00	[rcu_tasks_kthread]
root	15	0.0	0.0	0	0	?	I	Sep05	0:00	[rcu_tasks_rude_kthread]
root	16	0.0	0.0	0	0	?	I	Sep05	0:00	[rcu_tasks_trace_kthread]
root	17	0.0	0.0	0	0	?	S	Sep05	0:03	[ksoftirqd/0]
root	18	0.0	0.0	0	0	?	I	Sep05	0:09	[rcu_preempt]
root	19	0.0	0.0	0	0	?	S	Sep05	0:00	[rcu_exp_par_gp_kthread_worker/0]
root	20	0.0	0.0	0	0	?	S	Sep05	0:00	[rcu_exp_gp_kthread_worker]
root	21	0.0	0.0	0	0	?	S	Sep05	0:00	[migration/0]
root	22	0.0	0.0	0	0	?	S	Sep05	0:00	[idle_inject/0]
root	23	0.0	0.0	0	0	?	S	Sep05	0:00	[cpuhp/0]
root	24	0.0	0.0	0	0	?	S	Sep05	0:00	[cpuhp/1]
root	25	0.0	0.0	0	0	?	S	Sep05	0:00	[idle_inject/1]
root	26	0.0	0.0	0	0	?	S	Sep05	0:01	[migration/1]
root	27	0.0	0.0	0	0	?	S	Sep05	0:04	[ksoftirqd/1]
root	34	0.0	0.0	0	0	?	S	Sep05	0:00	[kdevtmpfs]
root	35	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-inet_frag_wq]
root	36	0.0	0.0	0	0	?	S	Sep05	0:00	[kauditd]
root	37	0.0	0.0	0	0	?	S	Sep05	0:00	[khungtaskd]
root	38	0.0	0.0	0	0	?	S	Sep05	0:00	[oom_reaper]
root	40	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-writeback]
root	42	0.0	0.0	0	0	?	S	Sep05	0:19	[kcompactd0]
root	43	0.0	0.0	0	0	?	SN	Sep05	0:00	[ksmd]
root	44	0.0	0.0	0	0	?	SN	Sep05	0:04	[khugepaged]
root	45	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-kintegrityd]
root	46	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-kblockd]
root	47	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-blkcg_punt_bio]
root	48	0.0	0.0	0	0	?	S	Sep05	0:00	[irq/9-acpi]
root	50	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-tpm_dev_wq]
root	51	0.0	0.0	0	0	?	I<	Sep05	0:00	[kworker/R-edac-poller]

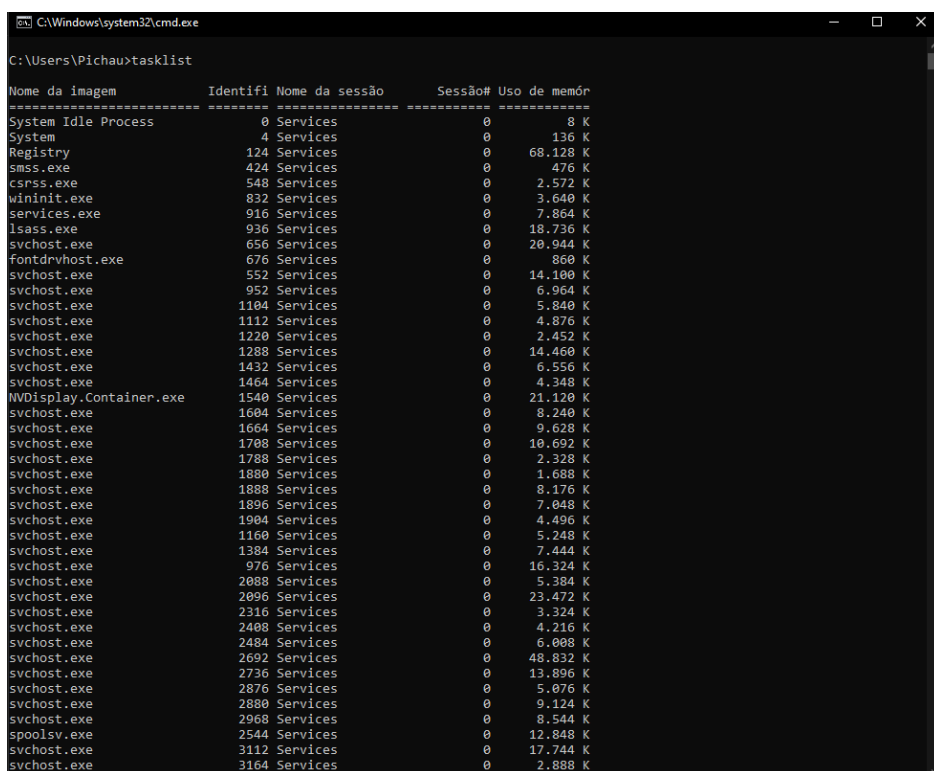
Essa abordagem organizada contribui para a eficiência global do sistema, tornando-o robusto mesmo em situações de alta demanda



## Windows

O gerenciamento de processos no Windows é uma função essencial do sistema operacional, responsável por controlar a execução de programas e tarefas em tempo real. Esse gerenciamento envolve o acompanhamento do uso de recursos do sistema, como memória e processamento, garantindo que cada processo receba a atenção necessária sem comprometer a estabilidade da máquina. Para analisar e monitorar os processos em execução, o Windows disponibiliza o comando (**tasklist**), que permite listar todos os processos ativos no sistema, mostrando informações importantes como o nome do processo, o identificador (PID) e o consumo de memória associado a cada um.

O uso do (**tasklist**) é fundamental para identificar quais aplicações estão em execução, permitindo ao usuário ou administrador verificar o impacto de cada processo no desempenho geral do sistema. Além disso, essa ferramenta facilita a identificação de processos que possam estar consumindo recursos de forma excessiva ou que estejam em estado anormal, possibilitando ações corretivas, como o encerramento de tarefas que estejam comprometendo a operação do computador.



Nome da imagem	Identifi	Nome da sessão	Sessão#	Uso de memór
System Idle Process	0	Services	0	8 K
System	4	Services	0	136 K
Registry	124	Services	0	68.128 K
smss.exe	424	Services	0	476 K
csrss.exe	548	Services	0	2.572 K
wininit.exe	832	Services	0	3.640 K
services.exe	916	Services	0	7.864 K
lsass.exe	936	Services	0	18.736 K
svchost.exe	656	Services	0	20.944 K
fontdrvhost.exe	676	Services	0	860 K
svchost.exe	552	Services	0	14.100 K
svchost.exe	952	Services	0	6.964 K
svchost.exe	1104	Services	0	5.840 K
svchost.exe	1112	Services	0	4.876 K
svchost.exe	1220	Services	0	2.452 K
svchost.exe	1288	Services	0	14.460 K
svchost.exe	1432	Services	0	6.556 K
svchost.exe	1464	Services	0	4.348 K
NtDisplay.Container.exe	1540	Services	0	21.120 K
svchost.exe	1604	Services	0	8.240 K
svchost.exe	1664	Services	0	9.628 K
svchost.exe	1708	Services	0	10.692 K
svchost.exe	1788	Services	0	2.328 K
svchost.exe	1880	Services	0	1.688 K
svchost.exe	1888	Services	0	8.176 K
svchost.exe	1896	Services	0	7.048 K
svchost.exe	1904	Services	0	4.496 K
svchost.exe	1160	Services	0	5.248 K
svchost.exe	1384	Services	0	7.444 K
svchost.exe	976	Services	0	16.324 K
svchost.exe	2088	Services	0	5.384 K
svchost.exe	2096	Services	0	23.472 K
svchost.exe	2316	Services	0	3.324 K
svchost.exe	2408	Services	0	4.216 K
svchost.exe	2484	Services	0	6.008 K
svchost.exe	2692	Services	0	48.832 K
svchost.exe	2736	Services	0	13.896 K
svchost.exe	2876	Services	0	5.076 K
svchost.exe	2880	Services	0	9.124 K
svchost.exe	2968	Services	0	8.544 K
spoolsv.exe	2544	Services	0	12.848 K
svchost.exe	3112	Services	0	17.744 K
svchost.exe	3164	Services	0	2.888 K

fornece uma visão detalhada dos processos em execução, mas também auxilia na análise de como os recursos do sistema estão sendo distribuídos, contribuindo para uma gestão mais eficiente e para a manutenção da estabilidade do ambiente operacional.



## Android

O gerenciamento de processos no sistema Android é fundamental para garantir o desempenho e a estabilidade do dispositivo. O sistema operacional mantém um controle constante sobre todos os processos em execução, priorizando aqueles que são essenciais para a experiência do usuário, como aplicativos em primeiro plano, e ajustando o uso de recursos dos processos em segundo plano conforme a necessidade. Esse gerenciamento envolve o monitoramento de memória, tempo de CPU e outros recursos, garantindo que processos críticos não sejam interrompidos e que os recursos não sejam desperdiçados por processos menos importantes.

Um dos comandos utilizados para observar o estado dos processos no Android é o ([top](#)). Esse comando permite visualizar em tempo real quais processos estão ativos, qual o consumo de CPU e memória de cada um, e a prioridade atribuída a eles pelo sistema. Por meio do ([top](#)), é possível identificar processos que consomem mais recursos e que podem impactar no desempenho geral do dispositivo, fornecendo informações essenciais para análise e otimização do gerenciamento de processos.

```
3:41 [status icons]
Window 1
Tasks: 3 total, 1 running, 2 sleeping, 0 stopped, 0 zombie
Mem: 7605556k total, 4529528k used, 3076028k free, 534944k buffers
Swap: 0k total, 0k used, 0k free, 3108396k cached
400%cpu 100%user 1%nice 13%sys 286%idle 0%iow 0%irq 0%sirq 0%host
PID USER PR NI VIRT RES SHR S[%CPU] %MEM TIME+ ARGS
5784 u0_a69 10 -10 3.1G 172M 115M S 12.0 2.3 0:09.47 com.termoneplus
5949 u0_a69 10 -10 11M 3.1M 2.6M R 0.0 0.0 0:00.01 top
5809 u0_a69 10 -10 8.7M 3.1M 2.6M S 0.0 0.0 0:00.01 sh -
```

O gerenciamento de processos no Android, portanto, combina monitoramento em tempo real com mecanismos automáticos de priorização e encerramento, garantindo eficiência no uso dos recursos do dispositivo.

### Comparação Crítica

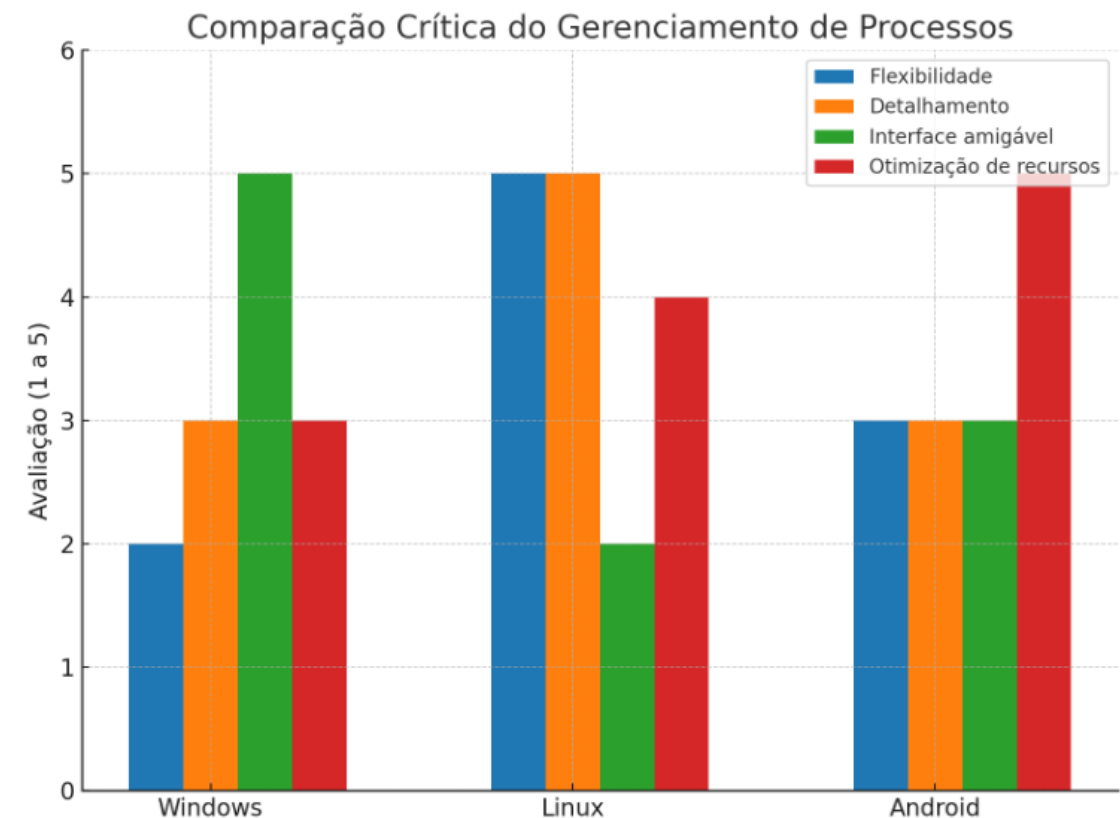
Ao comparar os sistemas operacionais Windows, Linux e Android no gerenciamento de processos, observam-se tanto semelhanças quanto diferenças significativas, bem como vantagens específicas em cada plataforma.

No Windows, o gerenciamento de processos é feito de forma centralizada, utilizando ferramentas como o comando tasklist, que permite listar todos os processos ativos no sistema, incluindo informações sobre uso de memória e identificação do processo.

O Linux, por sua vez, oferece um gerenciamento de processos mais flexível e detalhado, com comandos como ps aux e top, que permitem visualizar não apenas a lista completa de processos, mas também métricas em tempo real sobre uso de CPU e memória.

No Android, o gerenciamento de processos é orientado à otimização de recursos em dispositivos móveis. O sistema mantém múltiplos processos em segundo plano, controlando automaticamente o consumo de memória e a execução de aplicativos de forma eficiente. Apesar de menos acessível para monitoramento detalhado pelo usuário.

Em termos de vantagens, o Windows se destaca pela simplicidade e acessibilidade, o Linux pela flexibilidade e detalhamento técnico, e o Android pela gestão automática e eficiente de recursos em ambientes com restrição de memória e processamento.



### 3.4 Rede

O gerenciamento e a configuração de redes constituem um dos pilares essenciais para o funcionamento eficiente de sistemas computacionais, pois permitem a comunicação entre dispositivos, o compartilhamento de recursos e o acesso a serviços remotos. Uma rede é composta por um conjunto de dispositivos interconectados que trocam informações de forma organizada, seguindo protocolos específicos que garantem a integridade e a confiabilidade dos dados transmitidos.

Dentro do contexto de sistemas, a camada de rede é responsável por controlar como os dados são enviados, recebidos e roteados entre diferentes pontos da rede. Isso inclui a atribuição de endereços lógicos, a definição de rotas de transmissão e a manutenção de tabelas que permitem identificar caminhos eficientes para o tráfego de informações. A utilização de protocolos padronizados assegura que os dados sejam corretamente interpretados por todos os dispositivos conectados, independentemente de suas características internas.

Além disso, a segurança em redes é um aspecto crítico, envolvendo mecanismos como autenticação, criptografia e controle de acesso. Esses mecanismos protegem as informações contra interceptações e modificações não autorizadas, garantindo que somente usuários ou dispositivos legítimos possam acessar os recursos da rede. Outro ponto importante é o monitoramento do tráfego de dados, que permite identificar gargalos, prevenir falhas e otimizar o desempenho, assegurando a continuidade das operações e a qualidade do serviço.

A eficiência de uma rede também está associada à capacidade de suportar múltiplas conexões simultâneas, à escalabilidade para inclusão de novos dispositivos e à adaptabilidade frente a alterações na topologia ou nas condições de tráfego. Ferramentas de análise e diagnóstico permitem observar métricas como latência, taxa de transferência e perda de pacotes, auxiliando na tomada de decisões para aprimorar o desempenho da rede e resolver problemas de comunicação.

Em síntese, a gestão de redes em sistemas computacionais envolve um conjunto integrado de processos que garantem conectividade, segurança e desempenho, sendo essencial para o funcionamento harmônico e confiável de todos os serviços dependentes da comunicação entre dispositivos.





## Linux

No Linux, a gestão de rede permite monitorar e controlar a comunicação do sistema com outros dispositivos e redes externas. Os comandos (`ifconfig`) e (`ip addr show`) são utilizados para visualizar informações essenciais das interfaces de rede, como endereços IP, máscaras de sub-rede e o estado das conexões ativas. Essas ferramentas fornecem uma visão detalhada que auxilia na identificação de problemas e na verificação da conectividade do sistema.

Através das informações exibidas por (`ifconfig`) e (`ip addr`) show, é possível observar como as interfaces estão configuradas e detectar possíveis falhas na comunicação ou conflitos de endereçamento. Essa análise é importante para garantir que os dispositivos conectados à rede possam interagir corretamente e que os serviços dependentes da conectividade funcionem sem interrupções.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255  
    inet6 fd17:625c:f037:2:4c05:118d:9a81:3e1 prefixlen 64 scopeid 0x0<  
global>  
    inet6 fe80::e388:1937:bbf6:c4b0 prefixlen 64 scopeid 0x20<link>  
    ether 08:00:27:d1:f8:5d txqueuelen 1000 (Ethernet)  
    RX packets 73459 bytes 93023122 (88.7 MiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 15720 bytes 3178086 (3.0 MiB)  
    TX errors 0 dropped 30 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 32 bytes 2553 (2.4 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 32 bytes 2553 (2.4 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
(kali@kali)-[~]  
$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def  
ault qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
```

Além disso, o Linux oferece flexibilidade para configuração de múltiplas interfaces e endereços IP, permitindo adaptar a rede a diferentes cenários, como ambientes corporativos ou testes de conectividade. O uso contínuo de (`ifconfig`) e (`ip addr show`) contribui para manter a rede organizada, funcional e segura, garantindo que alterações ou ajustes nas interfaces sejam corretamente aplicados.



No Windows, a gestão de rede permite controlar e monitorar a comunicação do sistema com outros dispositivos e redes externas. O comando (**ipconfig**) é utilizado para exibir informações detalhadas sobre as interfaces de rede, incluindo endereços IP, máscaras de sub-rede e gateways padrão. Essas informações são essenciais para verificar a conectividade e garantir que o sistema esteja corretamente configurado para se comunicar com a rede local e com a internet.

Através do uso do (**ipconfig**), é possível identificar problemas de conexão, como ausência de endereço IP ou conflitos de endereçamento, permitindo que ajustes sejam feitos de forma rápida e eficiente. O comando fornece uma visão geral das interfaces ativas e suas configurações, facilitando a manutenção da rede e a resolução de falhas que possam comprometer a comunicação entre dispositivos.

```
C:\Windows\system32\cmd.exe

C:\Users\Pichau>ipconfig

Configuração de IP do Windows

Adaptador Ethernet Ethernet 2:

    Sufixo DNS específico de conexão. . . . . : 
    Endereço IPv6 de link local . . . . . : fe80::c20f:6161:8857:ec0d%10
    Endereço IPv4. . . . . : 18.42
    Máscara de Sub-rede . . . . . : 255.255.255.0
    Gateway Padrão. . . . . : 

Adaptador Ethernet Ethernet 3:

    Sufixo DNS específico de conexão. . . . . : 
    Endereço IPv6 de link local . . . . . : fe80::2e72:f42d:25a0:589a%8
    Endereço IPv4. . . . . : 192.
    Máscara de Sub-rede . . . . . : 255.255.255.0
    Gateway Padrão. . . . . : 

Adaptador Ethernet VMware Network Adapter VMnet1:

    Sufixo DNS específico de conexão. . . . . : 
    Endereço IPv6 de link local . . . . . : fe80::923f:ca91:8907:420a%16
    Endereço IPv4. . . . . : 192.
    Máscara de Sub-rede . . . . . : 255.255.255.0
    Gateway Padrão. . . . . : 

Adaptador Ethernet VMware Network Adapter VMnet8:

    Sufixo DNS específico de conexão. . . . . : 
    Endereço IPv6 de link local . . . . . : fe80::b7ef:6318:b348:439b%3
    Endereço IPv4. . . . . : 192.
    Máscara de Sub-rede . . . . . : 255.255.255.0
    Gateway Padrão. . . . . : 

C:\Users\Pichau>
```

Além disso, o Windows oferece flexibilidade na configuração das interfaces de rede, permitindo atribuir endereços IP manualmente ou automaticamente via DHCP, bem como monitorar alterações de conectividade. O uso do (**ipconfig**) é fundamental para acompanhar o estado da rede e garantir que todas as interfaces estejam funcionando corretamente, contribuindo para a estabilidade e a eficiência da comunicação do sistema.





Android

No Android, o gerenciamento de rede é essencial para permitir a comunicação entre o dispositivo e outros sistemas ou redes externas. O sistema mantém controle sobre as interfaces de rede, possibilitando que o usuário ou aplicativos acessem informações importantes, como endereços IP, status das conexões e tipo de rede utilizada, garantindo conectividade estável e eficiente.

Através do comando (`ipaddr`), é possível visualizar detalhes sobre as interfaces de rede ativas, incluindo endereços IP atribuídos e a configuração de cada conexão. Essa visualização é útil para identificar problemas de conectividade, verificar se o dispositivo está corretamente integrado à rede e assegurar que os dados possam ser transmitidos e recebidos de forma confiável.

```
Window 1
:/$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: sit0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN group default qlen 1000
    link/sit 0.0.0.0 brd 0.0.0.0
3: ip6tnl0@NONE: <NOARP> mtu 1452 qdisc noop state DOWN group default qlen 1000
    link/tunnel6 :: brd ::
4: wifi_eth: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:58:80:99 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20c:29ff:fe58:8099/64 scope link
        valid_lft forever preferred_lft forever
5: wlan0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 00:0c:29:58:80:99 brd ff:ff:ff:ff:ff:ff
:/$
```

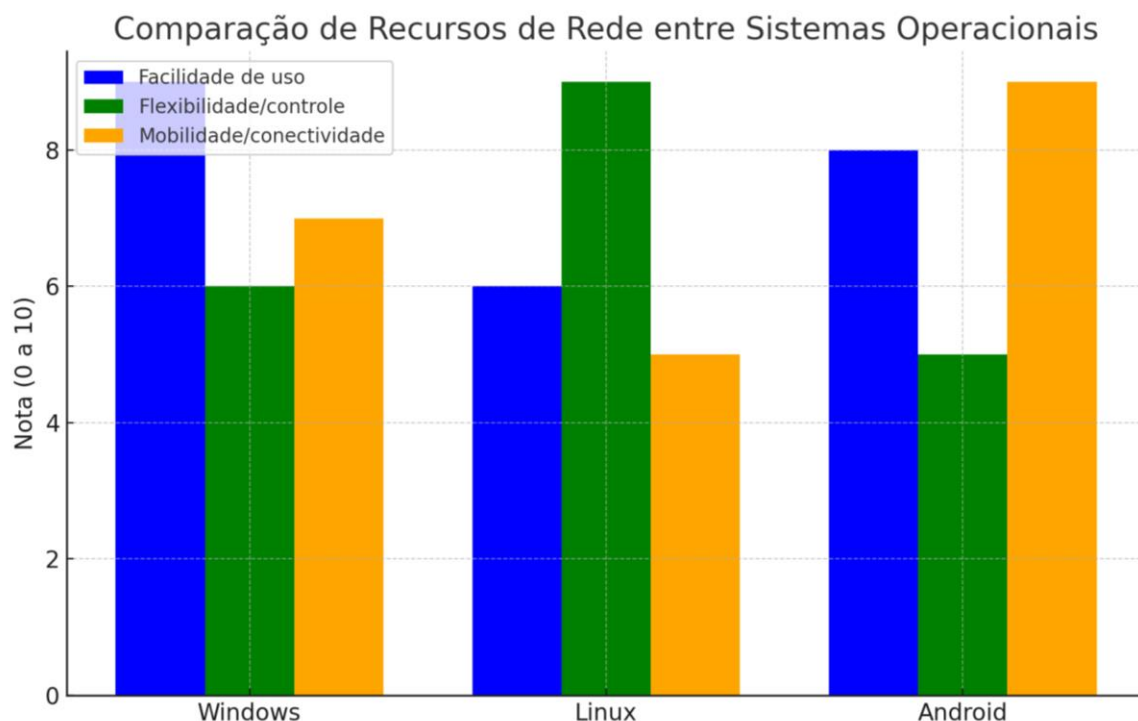
Além disso, o Android oferece mecanismos de adaptação automática, permitindo que as interfaces se ajustem a diferentes tipos de rede, como Wi-Fi ou dados móveis, garantindo continuidade na comunicação mesmo em ambientes com alterações constantes na disponibilidade de conexão. O gerenciamento eficaz da rede contribui para a estabilidade do sistema e para o desempenho de aplicativos que dependem de acesso à internet ou a serviços em rede.

## Comparação Crítica

Ao analisar o gerenciamento de rede entre Linux, Windows e Android, é possível identificar semelhanças importantes em relação à necessidade de monitoramento das interfaces, atribuição de endereços IP e verificação da conectividade. Todos os sistemas oferecem comandos específicos para consultar informações detalhadas sobre as interfaces de rede. Esses comandos permitem acompanhar o estado das conexões, identificar falhas e assegurar que a comunicação entre dispositivos seja estável e eficiente.

Entre as diferenças, o Linux se destaca pela flexibilidade na configuração de múltiplas interfaces e endereços IP, permitindo ajustes mais finos em ambientes complexos ou para testes de conectividade. O Windows apresenta uma interface mais padronizada e simplificada, com foco em facilidade de uso para monitoramento e manutenção de redes, sendo eficiente para usuários que precisam de informações rápidas sobre a conectividade. Já o Android prioriza a adaptação automática a diferentes tipos de rede, como Wi-Fi e dados móveis, garantindo que a conectividade seja mantida de forma contínua mesmo com mudanças frequentes na rede.

Quanto às vantagens, o Linux oferece maior controle e detalhamento, ideal para administração avançada de redes; o Windows combina praticidade e acessibilidade, facilitando a resolução de problemas comuns; e o Android proporciona mobilidade e adaptação dinâmica, garantindo comunicação constante em dispositivos móveis. Em síntese, cada sistema apresenta características que atendem a diferentes necessidades: robustez e controle no Linux, simplicidade e praticidade no Windows, e flexibilidade e continuidade no Android, evidenciando a diversidade de abordagens para gerenciamento de rede.



### 3.5 Discos e Armazenamento

O gerenciamento de discos e armazenamento é um componente essencial de qualquer sistema computacional, influenciando diretamente o desempenho, a confiabilidade e a capacidade de armazenamento dos dados. Ele envolve tanto a organização física quanto lógica do espaço disponível, buscando otimizar a leitura, a gravação e a preservação das informações.

No nível físico, os dispositivos de armazenamento podem ser classificados em tradicionais, como discos magnéticos, e modernos, como unidades de estado sólido (SSDs). Cada tecnologia apresenta características próprias em termos de velocidade de acesso, durabilidade e custo por gigabyte. A escolha adequada depende do tipo de aplicação e da necessidade de desempenho em operações de leitura e gravação.

Do ponto de vista lógico, o armazenamento é gerenciado por meio de sistemas de arquivos, que definem como os dados são estruturados, indexados e recuperados. Um sistema de arquivos eficiente permite não apenas acessar rapidamente grandes volumes de dados, mas também organizar informações de forma hierárquica, controlar permissões de acesso e realizar operações de manutenção, como verificação de integridade e recuperação de arquivos corrompidos.

Outro aspecto importante é a fragmentação e o uso do espaço. Com o tempo, o armazenamento de arquivos de diferentes tamanhos pode gerar lacunas no disco, tornando necessária a reorganização dos dados para manter a eficiência. Técnicas de otimização, como compactação e balanceamento de armazenamento, são utilizadas para reduzir desperdícios e melhorar o desempenho geral do sistema.

Além disso, mecanismos de redundância e backup desempenham um papel crítico na proteção contra falhas. Estratégias como replicação de dados, armazenamento em camadas e utilização de dispositivos externos garantem que a informação permaneça segura, mesmo diante de falhas físicas ou lógicas.

Por fim, o monitoramento contínuo do desempenho e do uso do armazenamento permite identificar gargalos e planejar expansões futuras. Métricas como tempo de acesso, taxa de transferência e ocupação de disco fornecem insights essenciais para a manutenção eficiente do sistema.

Em resumo, discos e armazenamento não se limitam apenas à retenção de dados, mas constituem um conjunto de práticas e tecnologias integradas que asseguram eficiência, confiabilidade e escalabilidade no gerenciamento da informação.



## Linux

No Linux, o gerenciamento de discos e armazenamento é estruturado de forma a permitir que o usuário visualize e organize facilmente o espaço disponível em diferentes dispositivos de armazenamento. O sistema apresenta uma hierarquia clara de diretórios e pontos de montagem, permitindo que os discos sejam acessados e utilizados de maneira eficiente, além de possibilitar a identificação rápida de partições e volumes montados.

A administração do espaço em disco pode ser monitorada de forma prática utilizando o comando (`df -h`), que fornece uma visão detalhada do uso do armazenamento, mostrando o espaço total, o espaço ocupado e o espaço livre em cada sistema de arquivos montado. Esse monitoramento é essencial para identificar discos próximos da capacidade máxima e planejar ações preventivas, evitando problemas de desempenho ou falhas no armazenamento.

```
(kali@kali)-[~]
$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
udev	921M	0	921M	0%	/dev
tmpfs	198M	976K	197M	1%	/run
/dev/sda1	79G	15G	60G	20%	/
tmpfs	987M	4.0K	987M	1%	/dev/shm
tmpfs	5.0M	0	5.0M	0%	/run/lock
tmpfs	1.0M	0	1.0M	0%	/run/credentials/systemd-journald.service
tmpfs	987M	8.5M	979M	1%	/tmp
tmpfs	1.0M	0	1.0M	0%	/run/credentials/getty@tty1.service
tmpfs	198M	136K	198M	1%	/run/user/1000

Além disso, o Linux oferece suporte a múltiplos tipos de sistemas de arquivos, permitindo escolher a melhor solução de acordo com a necessidade de desempenho e confiabilidade. A combinação de gerenciamento lógico eficiente com ferramentas de monitoramento contribui para uma administração de discos robusta, garantindo que os dados sejam armazenados de forma organizada e segura.



## Windows

No Windows, o gerenciamento de discos e armazenamento permite que os usuários organizem, monitorem e mantenham seus dispositivos de forma prática e eficiente. O sistema oferece uma interface intuitiva para visualizar partições, discos e volumes, possibilitando uma administração clara do espaço disponível e do uso de cada unidade.

A ferramenta ([diskpart](#)) permite realizar operações avançadas de gerenciamento de discos, como criação, exclusão e formatação de partições, além de possibilitar a visualização detalhada das características de cada volume. Esse recurso é fundamental para administradores e usuários avançados, garantindo que os discos sejam configurados corretamente e utilizados de maneira otimizada.

```
Selecionar C:\Windows\system32\diskpart.exe

Microsoft DiskPart versão 10.0.19041.3636

Copyright (C) Microsoft Corporation.
No computador: DESKTOP-M81U9SG

DISKPART> chkdsk

Microsoft DiskPart versão 10.0.19041.3636

ACTIVE      - Marcar a partição selecionada como ativa.
ADD          - Adiciona um espelho a um volume simples.
ASSIGN       - Atribua uma letra de unidade ou um ponto de montagem ao volume
              selecionado.
ATTRIBUTES  - Manipular atributos de volume ou disco.
ATTACH       - Anexa um arquivo de disco virtual.
AUTOMOUNT    - Habilitar ou desabilitar a montagem automática de volumes básicos.
BREAK        - Quebrar um conjunto de espelhos.
CLEAN        - Limpar as informações de configuração ou todas as informações do
              disco.
COMPACT      - Tentativas para reduzir o tamanho físico do arquivo.
CONVERT      - Converter formatos de disco diferentes.
CREATE       - Criar um volume, partição ou disco virtual.
DELETE       - Excluir um objeto.
DETAIL       - Fornecer detalhes sobre um objeto.
DETACH       - Desanexa um arquivo de disco virtual.
EXIT         - Sair do DiskPart.
EXTEND       - Estender um volume.
EXPAND       - Expande o tamanho máximo disponível em um disco virtual.
FILESYSTEMS  - Exibir os sistemas de arquivos atuais e com suporte no volume.
```

Além disso, o Windows oferece suporte a múltiplos sistemas de arquivos, o que proporciona flexibilidade para diferentes necessidades de armazenamento e desempenho. O controle do espaço em disco, combinado com ferramentas de monitoramento e manutenção, contribui para uma gestão eficiente e segura dos dados, prevenindo problemas relacionados à falta de espaço ou à fragmentação dos arquivos.



No Android, o gerenciamento de discos e armazenamento é projetado para oferecer uma visão clara do uso de memória interna e externa, permitindo que o usuário controle o espaço disponível e organize os arquivos de forma eficiente. O sistema utiliza uma estrutura de diretórios bem definida, garantindo que aplicativos, mídia e dados pessoais sejam armazenados de maneira organizada e acessível.

O comando (**df**) permite verificar detalhadamente o espaço utilizado e disponível em cada partição, fornecendo informações essenciais para identificar discos próximos da capacidade máxima. Esse monitoramento é importante para manter o desempenho do dispositivo, evitando lentidão ou falhas no armazenamento de dados.

A screenshot of an Android terminal window titled "Window 1". The terminal shows the output of the 'df' command, which displays disk space usage for various filesystems. The output is as follows:

```
:/ $ df
Filesystem      1K-blocks    Used Available Use% Mounted on
tmpfs            3802776      7300   3795476    1% /
/dev/loop1       2439120 2242860   179876   93% /system
/dev/block/sda1  8188704 1073928   6679100   14% /data
tmpfs            3802776        320   3802456    1% /dev
tmpfs            3802776         0   3802776    0% /mnt
none            3802776         0   3802776    0% /cache
/data/media      8188704 1073928   6679100   14% /storage/emulated
:/ $
```

Além disso, o Android oferece suporte a diferentes tipos de sistemas de arquivos, o que possibilita uma melhor adaptação entre armazenamento interno e cartões externos. A combinação de monitoramento de espaço e organização de diretórios contribui para uma gestão eficiente e segura dos dados, garantindo que as informações sejam preservadas e facilmente acessíveis.

## Comparação Crítica

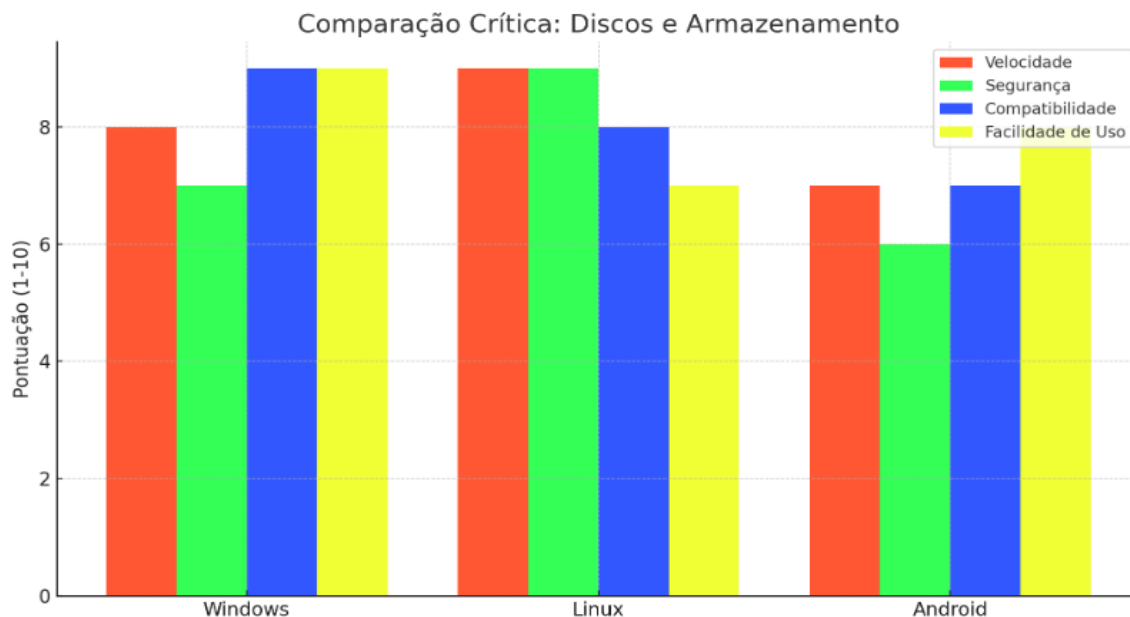
Ao analisar os sistemas operacionais Windows, Linux e Android no que se refere ao gerenciamento de discos e armazenamento, observam-se tanto semelhanças quanto diferenças significativas, assim como vantagens específicas em cada plataforma. Todos os três sistemas operacionais suportam múltiplos sistemas de arquivos, permitindo a leitura e gravação de dados de diferentes formatos. Além disso, oferecem ferramentas nativas ou integradas para gerenciamento de discos, como verificação de integridade, particionamento e monitoramento de uso, garantindo que o usuário possa manter o armazenamento organizado e otimizado.

O Windows se destaca pelo suporte amplo a sistemas de arquivos como NTFS, FAT32 e exFAT, além de ferramentas gráficas intuitivas para gerenciamento de discos, como o “Gerenciamento de Disco”, facilitando a experiência para usuários menos avançados.

O Linux, por sua vez, oferece suporte a sistemas de arquivos como ext4, Btrfs, XFS e outros, com ferramentas poderosas de linha de comando (como `df`, `du` e `lsblk`) que permitem maior controle e flexibilidade, sendo mais indicado para usuários avançados ou servidores.

Já o Android, derivado do Linux, possui um sistema de armazenamento mais simplificado, focado em dispositivos móveis. Utiliza formatos como ext4 e F2FS e gerencia o armazenamento de forma automática, com ferramentas integradas de limpeza de cache e monitoramento de uso por aplicativos, oferecendo uma experiência mais transparente ao usuário final.

Em resumo, enquanto Windows prioriza simplicidade e compatibilidade, Linux oferece robustez e controle avançado, e Android foca em automação e otimização para dispositivos móveis. Cada sistema apresenta soluções específicas para diferentes perfis de usuários e tipos de hardware, demonstrando que a escolha depende diretamente das necessidades de uso.





## 4. Análise Crítica

A análise das filosofias de design dos sistemas operacionais Windows, Linux e Android revela abordagens distintas que refletem os objetivos, público-alvo e histórico de cada plataforma. Estas filosofias influenciam diretamente aspectos como gerenciamento de processos, administração de armazenamento, conectividade em rede e experiência do usuário.

### Windows

A filosofia de design do Windows prioriza usabilidade e compatibilidade ampla. O sistema busca integrar interface gráfica amigável, suporte a uma enorme variedade de hardwares e softwares comerciais, e ferramentas de administração simplificadas. Evidências disso podem ser observadas nos comandos de gerenciamento de processos, como tasklist, que permite listar rapidamente processos em execução com informações detalhadas de memória e PID, facilitando o monitoramento por usuários e administradores. Em termos de armazenamento, o Windows utiliza sistemas de arquivos NTFS, que oferecem suporte a permissões detalhadas e recursos avançados como compressão e criptografia. Em redes, destaca-se pela facilidade de configuração e suporte a protocolos comuns, refletindo uma filosofia de acesso rápido e simplificado para usuários finais e corporativos.

### Linux

Linux adota uma filosofia de design centrada em flexibilidade, modularidade e controle pelo usuário. Essa abordagem se manifesta em ferramentas poderosas de linha de comando, como ps aux e top, que oferecem visão detalhada e configurável sobre os processos em execução. Em termos de armazenamento, o Linux permite múltiplos sistemas de arquivos (ext4, XFS, Btrfs), priorizando eficiência e customização. No contexto de rede, a filosofia de design enfatiza transparência e segurança, oferecendo controle granular sobre serviços, permissões e logs. Essas características refletem um sistema projetado para administradores avançados, servidores e ambientes de desenvolvimento, priorizando estabilidade e adaptabilidade sobre a simplicidade imediata de uso.

### Android

Android combina elementos de Linux com uma camada de interface e serviços específicos para dispositivos móveis. Sua filosofia de design foca em mobilidade, eficiência de recursos e integração com o ecossistema Google. O gerenciamento de processos, monitorado através do comando top, é otimizado para prolongar a vida útil da bateria e priorizar aplicativos em primeiro plano. No armazenamento, utiliza sistemas de arquivos adaptados a flash, como F2FS, que aumentam o desempenho em dispositivos móveis. Quanto à rede, Android enfatiza conectividade contínua e otimização para aplicações móveis, incluindo gerenciamento automático de dados e permissões, refletindo uma filosofia de experiência integrada e responsiva.



## Comparações e Conclusões

Semelhanças: Todos os sistemas operacionais implementam mecanismos robustos de gerenciamento de processos, armazenamento e rede, garantindo segurança e desempenho.

Diferenças: Windows prioriza usabilidade e compatibilidade, Linux foca em controle e flexibilidade, e Android busca mobilidade e otimização de recursos.

Vantagens específicas: Windows é ideal para usuários gerais e corporativos; Linux oferece maior controle e personalização para administradores e servidores; Android oferece eficiência em dispositivos móveis, integrando hardware e software de forma otimizada

## Evidências gráficas e práticas

Os gráficos gerados foram realizados através dos processos realizados nas comparação.

Sistema Operacional	Filosofia de Design	Comandos Analisados	Gráficos Gerados	Observações
Linux	Usabilidade e compatibilidade ampla	tasklist (gerenciamento de processos)	Gráfico de consumo de memória e CPU por processo	Permite monitoramento rápido e interface amigável; sistemas de arquivos NTFS suportam recursos avançados como criptografia e compressão.
Windows	Flexibilidade, modularidade e controle pelo usuário	ps aux, top (gerenciamento de processos)	Gráfico de consumo de memória, CPU e tempo de execução dos processos	Ferramentas poderosas e detalhadas para administradores; múltiplos sistemas de arquivos (ext4, XFS, Btrfs) permitem customização; rede segura e transparente.
Android	Mobilidade, eficiência de recursos e integração com o ecossistema Google	top (gerenciamento de processos em dispositivos móveis)	Gráfico de consumo de CPU, memória e uso de armazenamento em flash	Otimizado para dispositivos móveis; gerenciamento de processos prioriza apps em primeiro plano; sistemas de arquivos como F2FS melhoram desempenho

## 5. Conclusão

A análise comparativa entre Windows, Linux e Android evidenciou que cada sistema operacional segue uma filosofia de design alinhada ao seu público-alvo e objetivos de uso.

O Windows demonstrou ser voltado para usabilidade e compatibilidade, oferecendo ferramentas simplificadas, como tasklist, para monitoramento de processos, gráficos que evidenciam consumo de memória e CPU gerenciáveis por usuários comuns, e suporte a sistemas de arquivos avançados (NTFS), que equilibram segurança e praticidade.

O Linux apresentou uma abordagem centrada em flexibilidade, controle e modularidade, permitindo monitoramento detalhado via ps aux e top, múltiplas opções de sistemas de arquivos (ext4, XFS, Btrfs) e gerenciamento de rede altamente configurável. Os gráficos de desempenho indicam eficiência em processos complexos e estabilidade sob carga intensa, refletindo um sistema voltado a administradores, servidores e desenvolvedores.

O Android, por sua vez, destacou-se por mobilidade, otimização de recursos e integração com o ecossistema móvel, utilizando top para gerenciamento de processos com foco em prioridade de aplicativos e economia de bateria. Seus gráficos mostram eficiência no uso de CPU, memória e armazenamento em flash, demonstrando que a filosofia de design é orientada para dispositivos móveis e experiência contínua do usuário.

Em termos de descobertas, nota-se que:

Todos os sistemas operacionais possuem mecanismos robustos de gerenciamento de processos, armazenamento e rede, mas aplicam suas prioridades de forma distinta.

Windows equilibra simplicidade e funcionalidade para o usuário final.

Linux proporciona controle detalhado e alta customização, sendo ideal para ambientes avançados.

Android otimiza desempenho e conectividade para dispositivos móveis, com foco na experiência do usuário.

Portanto, o objetivo da introdução analisar e comparar sistemas operacionais considerando processos, rede e armazenamento foi alcançado. Ficou evidente que cada sistema implementa soluções técnicas coerentes com sua filosofia de design, refletindo diretamente nas suas vantagens, limitações e público-alvo.

## 6. Autoavaliação

Durante a realização do projeto, enfrentei algumas dificuldades técnicas e logísticas que impactaram o andamento das atividades. Uma das principais barreiras foi a instalação das máquinas virtuais, que inicialmente apresentou erros devido à necessidade de ativar o modo de virtualização na BIOS da minha máquina. Esse ajuste foi essencial para que os sistemas operacionais pudessem ser executados corretamente, permitindo testar comandos, gerar gráficos e realizar comparações entre Windows, Linux e Android.

Além disso, observou-se que alguns colaboradores do grupo não possuíam computadores com capacidade de hardware suficiente para instalar as ISOs dos sistemas operacionais. Essa limitação exigiu que ajustássemos a distribuição das tarefas, priorizando atividades que pudessem ser realizadas em máquinas menos potentes, e também reforçou a importância de planejamento prévio considerando os recursos disponíveis.

Em termos de contribuições individuais, participei ativamente da análise crítica, criação de gráficos, geração de tabelas comparativas e elaboração de textos explicativos. A interação com os demais membros do grupo permitiu troca de conhecimentos, identificação de erros e soluções conjuntas, além de melhorar a organização do trabalho em equipe.

O projeto proporcionou diversos aprendizados, incluindo:

Compreensão detalhada das filosofias de design e funcionamento de diferentes sistemas operacionais;

Uso e interpretação de comandos de gerenciamento de processos, rede e armazenamento;

Desenvolvimento de relatórios estruturados, gráficos comparativos e tabelas informativas;

Planejamento colaborativo frente a limitações técnicas e de recursos.

No geral, a experiência reforçou habilidades de resolução de problemas, trabalho em equipe e análise crítica, além de proporcionar maior familiaridade prática com ambientes virtuais e sistemas operacionais variados.

## 7. Referências

### Ferramentas de Virtualização:

Oracle VirtualBox:  
Disponível para Windows, Linux e macOS.  
Link para download:  
[virtualbox.org](https://www.virtualbox.org)

VMware Workstation Pro:  
Disponível para Windows e Linux.  
Link para download:  
[vmware.com](https://www.vmware.com)

### Sistemas Operacionais:

Android-x86:  
Versão 9.0-r2 (64-bit).  
Link para download:  
[android-x86.org](https://android-x86.org)

Windows 10:  
Imagem ISO oficial para instalação.  
Link para download:  
[Microsoft](https://www.microsoft.com)

Linux (Ubuntu):  
Distribuição popular baseada em Debian.  
Link para download:  
[Ubuntu](https://ubuntu.com)

### Linux: Comandos Básicos e Avançados

Comandos Básicos para Iniciantes no Fedora  
<https://diolinux.com.br/sistemas-operacionais/comandos-basicos-para-quem-esta-chegando-ao-fedora.html>

Comandos para Trabalhar com Interfaces de Rede no Linux  
<https://diolinux.com.br/sistemas-operacionais/trabalhando-com-rede-linux.html>

Comandos Básicos para Arch Linux e Manjaro  
<https://diolinux.com.br/sistemas-operacionais/comandos-basicos-para-quem-esta-chegando-ao-arch-linux-e-manjaro.html>

Windows: Comandos do Prompt de Comando

Comandos Básicos do Prompt de Comando no Windows

<https://www.geeksforgeeks.org/techtips/change-directories-in-command-prompt/>

Android: Ferramentas de Linha de Comando

Android Debug Bridge (adb)

<https://developer.android.com/tools/adb>