



LivreCamp

Índice

Aula 1

- > Exercício 1
- > Linguagem de Programação
- > Softwares e Licenças
- > Exercício 2

Aula 2

- > Sistema Operacionais
- > Diretório
- > Linux
 - > Quem Usa o Linux?
- > Shell
 - > pwd
 - > ls
 - > cd
 - > mkdir
 - > cp
 - > mv
 - > rm

Aula 3

- > wget
- > clear
- > cat
- > less
- > grep
- > man e help
- > ps
- > kill
- > Distribuições
 - > Debian
 - > Ubuntu
 - > Mint
 - > Fedora
 - > Arch
 - > Manjaro
 - > Kali
 - > Tails

> Ambientes Gráficos

- > Gnome 3
- > Unity
- > Cinnamon
- > MATE
- > KDE

Minicurso GNU/**Linux**

>> aula_1 - Licenças<<

Exercício

Exercício - Lidando com código fechado

Abra o terminal e navegue para onde você deseja salvar o programa e digite o código abaixo:

```
$ git clone https://github.com/LivreCamp/super-sum-calculator.git
```

Navegue até a pasta “proprietário”:

```
$ cd super-sum-calculator/prorietario
```

Execute o programa:

```
$ ./ssc-64
```

Verifique a licença:

```
$ cat LICENSE
```

Softwares

Softwares - Linguagens de Programação

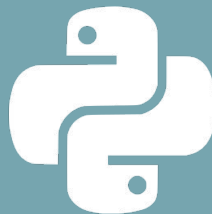
É uma linguagem designada para comunicar instruções para um computador; Um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador.



Linguagem
C++



Rust



Python



Java

Softwares - Que licença usar ?

	Uso comercial	Distribuição	Modificar	Patente	Uso Privado	Disponibilizar Código	Licença e Copyright	Uso na web e distribuição	Mesma Licença	Apontar Mudanças	Compromisso	Uso de marca	Garantia
GNU AGPLv3													
GNU GPLv3													
GNU LGPLv3													
MIT License													
The Unlicense													

Limitação de Uso

Condição de Uso

Permissão de Uso

<https://choosealicense.com>

Exercício

Exercício - Lidando com código aberto

Navegue até a pasta “gnu-gplv3”

```
$ cd ../gnu-gplv3
```

Abra o código fonte:

```
$ gedit ssc.c
```

Recompile ele com suas modificações:

```
$ gcc ssc.c -o ssc
```

Rode ele:

```
$ ./ssc
```

Comandos

Navegar pelos diretórios:

```
> cd nome-do-diretorio
```

Criar um novo Diretório

```
> mkdir nome-do-diretorio
```

Executar um programa

```
> ./nome-do-programa
```

Compilar um programa em C

```
> gcc programa.c -o programa
```

Mostra o texto de um arquivo

```
> cat arquivo
```

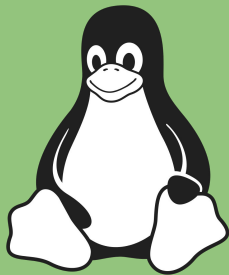
Minicurso GNU/**Linux**

>> aula_2 - Introdução ao Linux <<

Sistemas Operacionais

Sistemas Operacionais - O que é um SO?

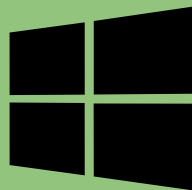
É um programa ou um conjunto de programas cuja função é gerenciar os recursos do sistema (definir qual programa recebe atenção do processador, gerenciar memória, criar um sistema de arquivos, etc.), fornecendo uma interface entre o computador e o usuário.



Linux



Mac OS



Windows



Android

GNU/Linux

Gnu/Linux - Quem usa o Linux?#

> 482 dos 500 computadores mais rápidos do mundo (top500.org, novembro de 2013).

Android, Peugeot, Amazon, Novell, Google, IBM, Wikipedia, DreamWorks Animation, Union Bank da Califórnia, Banco do Brasil, Bolsa de Valores de Londres, Bolsa de Valores de Nova York, NASA, CERN, Internet Archive, Casa Branca, Departamento de Defesa dos EUA, vários setores de governos de vários países (por exemplo, Brasil, Alemanha, Espanha, França, China, Rússia, África do Sul, Turquia, Venezuela, Cuba, Suíça, Índia e Paquistão), etc.

...e qualquer bom profissional de computação.

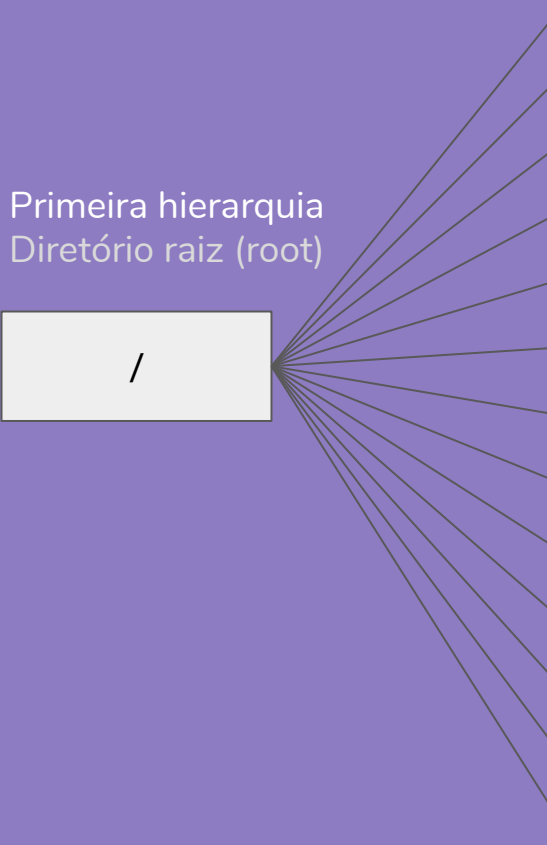
Gnu/Linux - Diretórios e Arquivos

- > Diretório é o local usado para armazenar conjuntos de arquivos, para melhor organização e localização
- > Filesystem Hierarchy Standard (padrão para sistema de arquivos hierárquico), ou FHS:
 - > Define os principais diretórios, e o seu conteúdo, em um sistema operacional do tipo *nix (Unix-like)
- > Especificado por “/” (Unix-like) ao invés de “\” (Windows)
- > Todo arquivo/diretório possui regras de permissões de escrita, leitura e execução (Read, Write, eXecute).

Gnu/Linux - Diretórios e Arquivos

Primeira hierarquia
Diretório raiz (root)

/



/bin/	binários do usuário - cat, ls, cd,...
/boot/	arquivos de sistema do boot
/dev/	arquivos de dispositivos
/etc/	arquivos de configuração do sistema
/home/	diretórios dos usuários
/lib/	bibliotecas do sistema
/mnt/	ponto genérico de montagem
/root/	diretório do usuário root
/sbin/	executáveis administrativos - mkfs, reboot
/srv/	diretório para dados do sistema
/tmp/	arquivos temporários
/usr/	arquivos compartilhados pelo sistema
/var/	escrita de dados - logs e arquivos de configuração

Gnu/Linux - Shell

- > O shell é um programa que pega os seus comandos do teclado e manda para o sistema operacional executar. Ele roda dentro do terminal.
- `/bin/sh` - Bourne Shell, criado por Steve Bourne
- `/bin/bash` - Bourne Again Shell, padrão da maioria das distribuições
- `/bin/zsh` - Versão aprimorada e mais poderosa de shells anteriores
- `/bin/fish` - Um shell amigável e fácil de usar

[Comparação entre os command shells]

Gnu/Linux - Shell - Copiando e Colando

- > No terminal os atalhos de copiar e colar utilizam atalhos diferentes do usual.
- > Cada interface pode ter um atalho diferente. O comum é `Ctrl+Shift+C` e `Ctrl+Shift+V`.
- > Utilize botão direito do mouse > Copy / Paste na maioria dos ambientes.
- > `Ctrl + C` no terminal é utilizado para encerrar o programa que está sendo executado. Execute o comando abaixo:

```
$ gedit
```

Utilize `Ctrl+C` no terminal para fechar o gedit.

Gnu/Linux - Shell - pwd

- > O diretório de trabalho corrente é o diretório em que um processo (shell no nosso caso) está no momento.
- > O comando pwd (print working directory) simplesmente exibe o nome do diretório onde estamos.

```
$ pwd
```

```
/home/seu-usuario
```

Gnu/Linux - Shell - ls

- > O comando `ls` (list) lista todo o conteúdo de um diretório. Por padrão é listado o diretório corrente.

```
$ ls -l
```

exibe uma listagem detalhada do diretório: permissão, número de hardlinks, dono, grupo, tamanho em bytes, modificação e nome

```
$ ls /bin/
```

exibe o conteúdo do diretório `/bin/`

|

Gnu/Linux - Shell - ls

- > Nos sistemas *nix os arquivos ou diretórios são considerados ocultos adicionando o caractere . (ponto) no início dele.

```
$ ls -a
```

exibe todos os arquivos do diretório, incluindo ocultos

Gnu/Linux - Shell - Usando o *

- > Para podermos lidar com vários arquivos de uma vez, ou para escrever padrões de filtros, podemos utilizar o * ao escrever o caminho dos nossos arquivos.
- > Ele é interpretado como um coringa. Todos os arquivos que puderem ser enquadrados serão usados.

```
$ ls *.txt
```

Lista todos os arquivos que terminam com .txt

Exercício

Liste todos os arquivos de **/bin** que começam com **get**

Exercício

Liste todos os arquivos de **/bin** que começam com **get**

```
$ ls /bin/get*
```

Gnu/Linux - Shell - Caminhos Importantes

- > Existem alguns diretórios e “atalhos” para caminhos que são bastante utilizados no ambiente Unix

`.` (Diretório atual)

`-` (Último diretório)

`..` (Diretório superior)

`/` (Root do Sistema)

`~/` (Home do usuário)

`./` (execução de script/binário)

Gnu/Linux - Shell - cd

- > O comando cd (change directory) muda o diretório de trabalho corrente do shell

```
$ cd Downloads
```

```
$ cd /home/$USER
```

```
$ cd ../bin/
```

```
$ cd ~/Documents/
```

```
$ cd
```

Exercício

1. Entre na pasta **/var/log**
2. Liste os arquivos que terminam com **.log**
3. Navegue até sua Home
4. Liste todos os arquivos ocultos

Exercício

1. Entre na pasta **/var/log**

```
$ cd /var/log
```

2. Liste os arquivos que terminam com **.log**

```
$ ls *.log
```

3. Navegue até sua Home

```
$ cd ~
```

4. Liste todos os arquivos ocultos

```
$ ls -a
```


Gnu/Linux - Shell - touch

- > O comando `touch` cria um arquivo. Se o arquivo já existir o comando apenas altera a data de modificação dele.

```
$ touch arquivo1
```

```
$ touch ~/Desktop/arquivo2
```

Gnu/Linux - Shell - mkdir

- > O comando mkdir (make directory) cria uma nova pasta, com o nome especificado.
- > Se você colocar um caminho completo, cria uma pasta no lugar especificado

```
$ mkdir pasta1
```

```
$ mkdir ~/Desktop/nova-pasta
```

Gnu/Linux - Shell - cp

- > O cp (copy) copia um arquivo (ou vários) para o destino.

```
$ cp arquivo1 arquivo1-copia
```

- > Se o destino for uma pasta, copiará os arquivos usando o mesmo nome

```
$ cp arquivo1 ./pasta1/
```

- > Se você especificar vários arquivos, ou um padrão, todos os arquivos serão copiados

```
$ cp arquivo1* ./pasta1/
```

- > Para copiar pastas, usar -r (recursivo)

```
$ cp -r pasta1 copia-da-pasta1
```

Gnu/Linux - Shell - mv

- > O comando mv (move) move um arquivo ou pasta de um lugar para outro. Pode ser usado para renomear arquivo, ou colocar em outra pasta, ou até os 2 ao mesmo tempo

```
$ mv arquivo1 arquivo-renomeado
```

```
$ mv arquivo1-copia ./pasta1/arquivo3
```

```
$ mv arquivo* ./pasta1/
```

```
$ mv copia-da-pasta1 ./pasta2
```

Gnu/Linux - Shell - rm

- > O comando `rm` é para apagar (remove) um arquivo. Após remover com `rm` ele não poderá ser recuperado. **NÃO** vai para lixeira.

```
$ rm ~/Desktop/arquivo2
```

- > O comando `rm` também serve para apagar um diretório com o parâmetro `-r` (recursive)

```
$ rm -r pasta1 pasta2
```

```
$ rm -r ~/Desktop/nova-pasta
```

Dúvidas?

Minicurso GNU/**Linux**

>> aula_3 - Introdução ao Linux Pt2 <<

Gnu/Linux - Shell - wget

- > **wget** (web get) faz download da URL especificada.
- > Parâmetro **-O** (precisa ser O maiúsculo) especifica o nome do arquivo de saída (nome do arquivo onde o download vai ser salvo).

```
$ wget https://duckduckgo.com
```

```
$ wget http://bit.ly/2fzhioU -O gplv3.txt
```


Gnu/Linux - Shell - clear

- > Comando para limpar o texto do seu terminal. Apaga todo o texto já impresso nele.
- > `Ctrl + L` é um atalho de teclado para o `clear`.

```
$ clear
```

Gnu/Linux - Shell - cat

- > O comando `cat` (concatenate) exhibe o conteúdo de um ou mais arquivos. Chama-se `cat` por que ele concatena ao exibir mais de um arquivo.

```
$ cat arquivo1 arquivo2 ...
```

```
$ cat gplv3.txt
```

Gnu/Linux - Shell - less

- > O comando `less` (show less) exibe o conteúdo de um arquivo e nos permite navegar nele.

```
$ less gplv3.txt
```

- > Navegue com as setas do teclado, e para sair utilize a tecla Q.

Gnu/Linux - Shell - grep

- > Procura por um padrão em um arquivo (ou arquivos) especificado, e imprime na saída as linhas em que esse padrão é encontrado.
- > Útil para filtrar uma saída, ou para procurar uma certa palavra em um arquivo.

```
$ grep palavra arquivo
```

```
$ grep GPL gplv3.txt
```

Gnu/Linux - Shell - grep

- > Parâmetro `-i` faz com que ignore letras maiúsculas e minúsculas e `-v` mostra apenas as linhas que não possuem o padrão.

```
$ grep -i -v paLaVra /var/log/*.log
```

Gnu/Linux - Shell - man & help

- > Existem dois jeitos de você descobrir como usar um comando: `man` e `--help/-h`
- > Vários comando podem ser executados com o `--help` e/ou `-h` logo após seu nome para pedir uma breve explicação do comando.

```
$ less --help
```

```
$ wget --help
```

```
$ wget -h
```

Gnu/Linux - Shell - man & help

- > O comando `man` (manual) mostra o manual daquele comando. Navegue no manual usando as setas do teclado, e aperte Q para sair.

```
$ man cp
```

Gnu/Linux - Shell - &

- > Se você abrir um programa com interface gráfica no terminal, o terminal ficará bloqueado.
- > & permite rodar programas em segundo plano e evitar isso.

```
$ gedit &
```

```
$ gimp &
```


Gnu/Linux - Shell - Redirecionamento

> Toda saída gerada por um comando é enviada para a saída padrão (tela do terminal).

> Podemos redirecionar a saída para um arquivo:

> - Cria ou sobrescreve um arquivo

>> - Cria ou anexa no fim de um arquivo

```
$ grep GPL gp1v3.txt >> arquivo1.txt
```

> Podemos também obter a entrada de um programa a partir de um arquivo usando o símbolo <

```
$ ./seuprograma < entrada.txt
```

Gnu/Linux - Shell - ps

> Comando para exibir os processos concorrentes em execução no seu terminal.

```
$ ps
```

PID	TTY	TIME	CMD
-----	-----	------	-----

8103	pts/0	00:00:01	gimp
------	-------	----------	------

7912	pts/0	00:00:00	gedit
------	-------	----------	-------

7942	pts/0	00:00:00	ps
------	-------	----------	----

Gnu/Linux - Shell - ps

> Utilize a flag -C para especificar um nome, e assim encontrar um processo fora do terminal.

```
$ ps -C chromium
```

```
$ ps -C firefox
```

> Utilize a flag -e para listar todos os processos.

```
$ ps -e
```

Gnu/Linux - Shell - kill

- > Comando para terminar (matar, fechar) processos usando o PID. Obtenha o PID usando o ps .

```
$ kill 1234
```

- > Use o `killall` para matar todos os processos que tenham o nome especificado.

```
$ killall gimp
```

Gnu/Linux - Shell - Pipe |

> Ao escrever um comando no terminal, podemos encadear mais de um comando na mesma linha usando o |

```
$ touch arq1 | cp arq1 arq1-copia
```

> Quando isso ocorre, a saída padrão do primeiro comando é usada como entrada padrão do segundo comando!

```
$ ps -e | grep chrom
```

Gnu/Linux - Permissões

> Todos os arquivos e pastas nos sistemas têm permissões. Já vimos estas permissões no comando `ls -l`

```
$ ls -l
```

```
drwxr-xr-x
```

```
-rw-r--r--
```

Gnu/Linux - Permissões

1	2	3	4	5	6
d	rwx	r-x	r-x	2	ra12345
					cc2000
-	rw-	r--	r--	2	root
					root

> Estas letras significam:

d: Directory r: Read (leitura) w: Write (escrita) x: eXecute (execução)

> 1 indica se é um diretório ou arquivo.

> 2 indica permissões do dono do arquivo.

> 3 indica permissões do grupo dono do arquivo.

> 4 indica permissões para todos os outros usuários.

> 5 indica o dono do arquivo

> 6 indica o grupo dono do arquivo

Gnu/Linux - Permissões - chmod

> Para alterar estas permissões utilizamos o chmod (change mode). No chmod devemos especificar quais permissões queremos alterar, e como queremos alterar, e no fim o arquivo (ou arquivos) que queremos alterar:

```
$ chmod u+rw arquivo1
```

```
$ chmod o-w arquivo1
```


Gnu/Linux - Permissões - chmod

- > Devemos especificar o que vamos alterar, e quais são as alterações.
- > Utilize (a) para all (todas as configurações);
 - (u) para user (usuário dono);
 - (g) para group (grupo dono);
 - (o) para other (outros usuários);
 - (+) para dar permissão;
 - (-) para remover permissão.

```
$ chmod a+rwx arquivo1
```

```
$ chmod o-w arquivo1
```

Gnu/Linux - Permissões - chown

> Altera o owner (dono) do arquivo (change ownership).

> chown <usuario>:<grupo>

```
$ chmod a+rwX arquivo1
```

```
$ chown ra12345:cc2017 arquivo1
```

```
$ chown root:root arquivo1
```

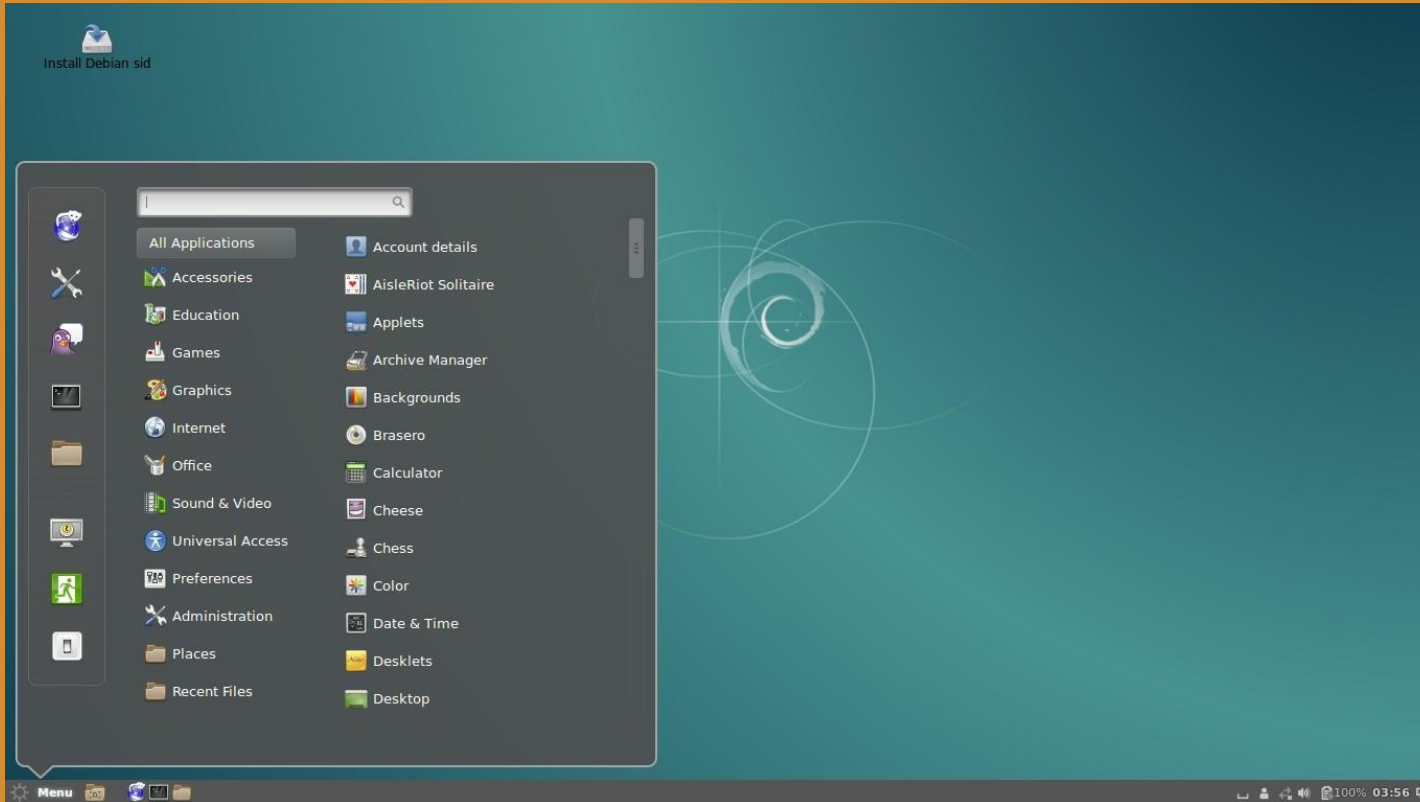
Distribuições

Gnu/Linux - Distros - Debian

- > Lançada em agosto de 1993
- > Base para mais de 100 outras distribuições
- > Completamente não-comercial
- > Mantida pela comunidade (The Debian Project)
- > É uma das distribuições mais estáveis e seguras
- > Suporta mais arquiteturas de processadores do que qualquer outra distribuição
- > Distribuição conservadora: novas arquiteturas não são incluídas imediatamente



Gnu/Linux - Distro - Debian

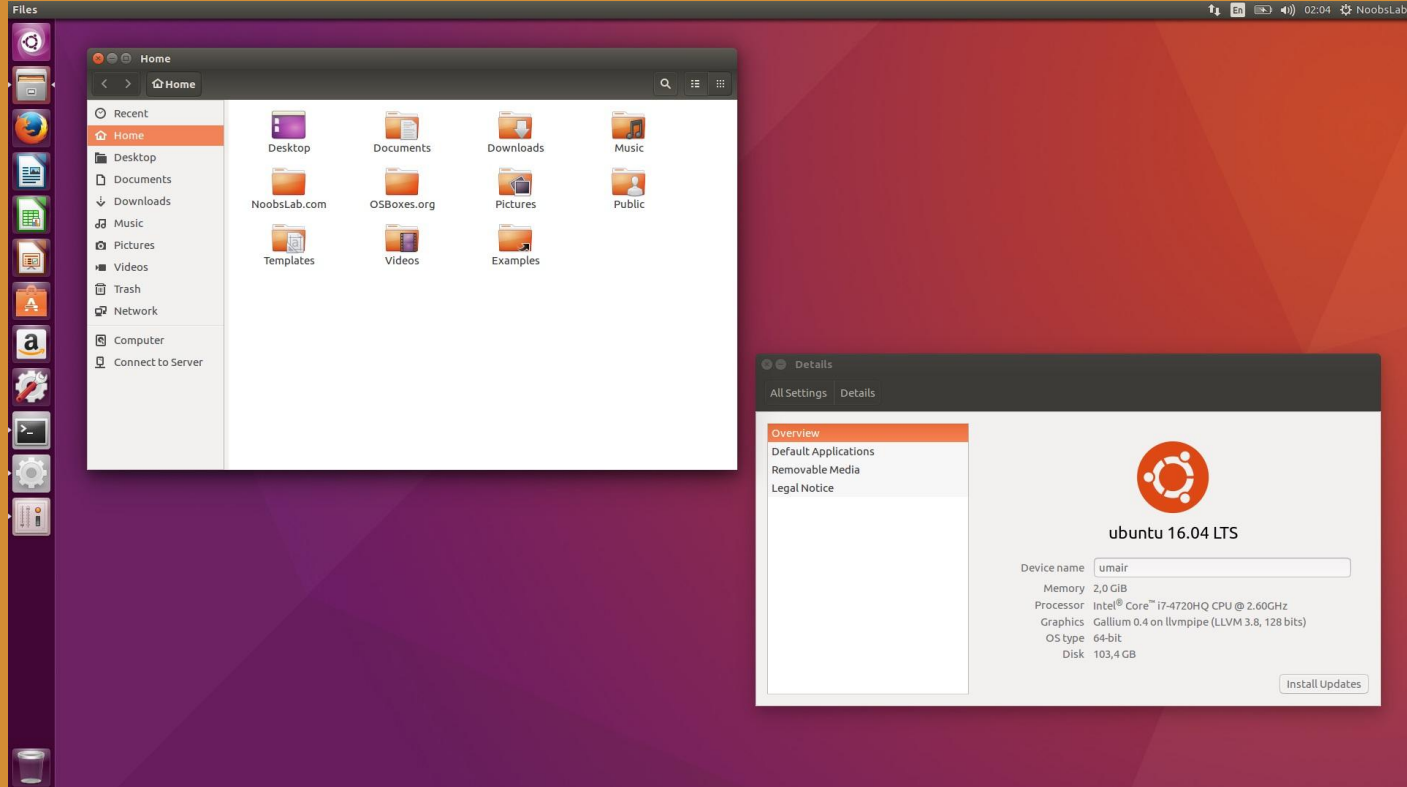


Gnu/Linux - Distros - Ubuntu

- > Lançada em outubro de 2004 por Mark Shuttleworth
- > "Linux for human beings"
- > Segue uma filosofia de que se pessoas comuns não conseguem usar, elas não têm a liberdade de usar.
- > Baseada no Debian (Shuttleworth trabalhava no projeto do Debian)
- > Patrocinada pela empresa Canonical



Gnu/Linux - DistroS - Ubuntu

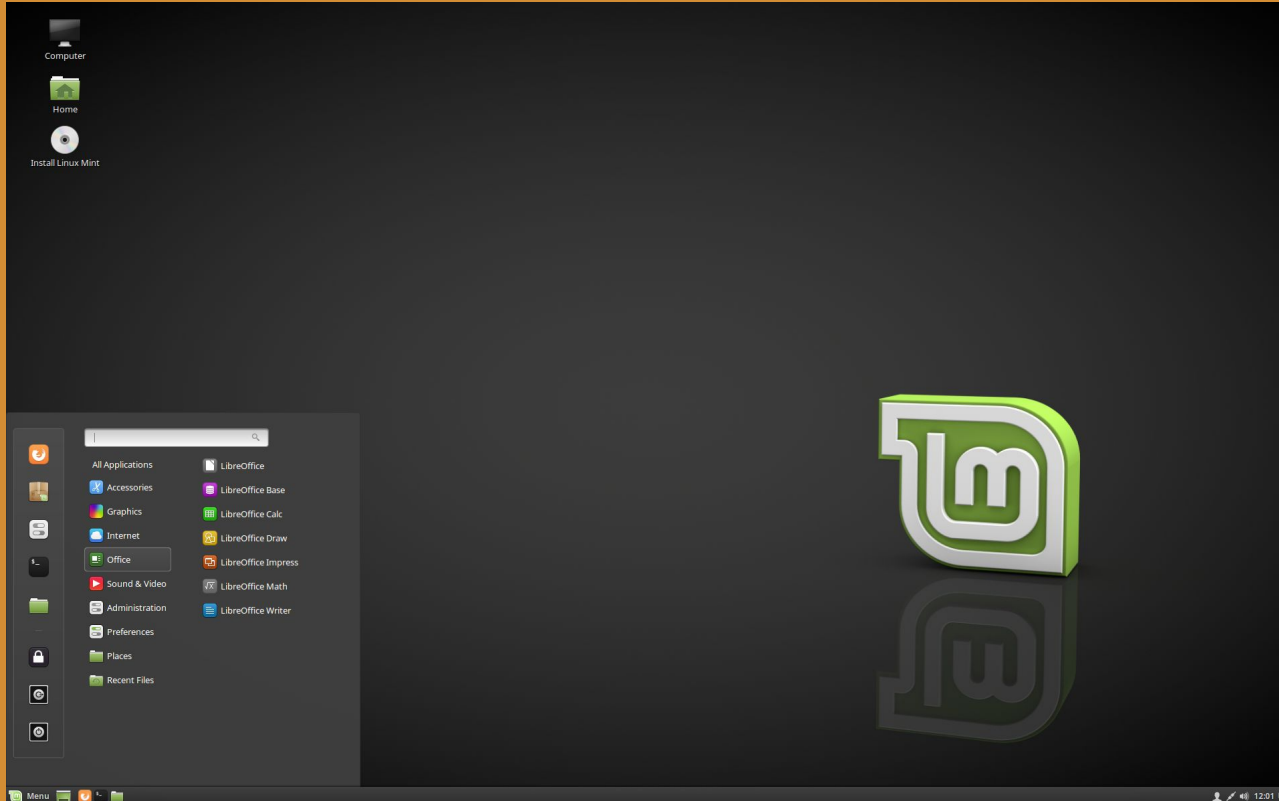


Gnu/Linux - Distros - Mint

- > Lançada em setembro de 2006 por Clement Lefebvre
- > Baseado no Debian e no Ubuntu
- > Mantido pela comunidade
- > Distribuição mais popular no dis
- > “Improved Ubuntu” ou “Ubuntu done right”
- > Ganhou força quando o Ubuntu decidiu trocar o Gnome pelo Unity, e o Gnome foi atualizado para o Gnome 3. Muitos usuários preferiam o Gnome 2, e a partir daí o mint se tornou o maior fork do Ubuntu, pois utilizava o Mate (continuação do Gnome2). Hoje em dia a interface padrão do Mint é a Cinnamon



Gnu/Linux - Distro - Mint



Gnu/Linux - Distros - Fedora

- > Lançada em novembro de 2003 pela empresa Red Hat
- > Distribuição de propósito geral, feita exclusivamente de software de código livre
- > Mantida pela comunidade (Projeto Fedora)
- > Novas versões normalmente a cada seis meses
- > Distribuição utilizada por Linus Torvalds
- > É a distribuição padrão nos laboratórios do Instituto de Computação



Gnu/Linux - Distros - Fedora



Gnu/Linux - Distros - Arch Linux

- > Lançado em 2002
- > Mantido pela comunidade
- > Foco em simplicidade e escolhas dos usuários. Instalação mínima, necessitando de edição de arquivos de configuração pelo próprio usuário, e a partir daí usuário decide o que precisa / quer instalar
- > Rolling release, ou seja, não existem versões do arch, os programas são sempre atualizados
- > Muito difícil para um iniciante, mas certamente após passar por esse desafio, o usuário terá muito mais domínio sobre o sistema.

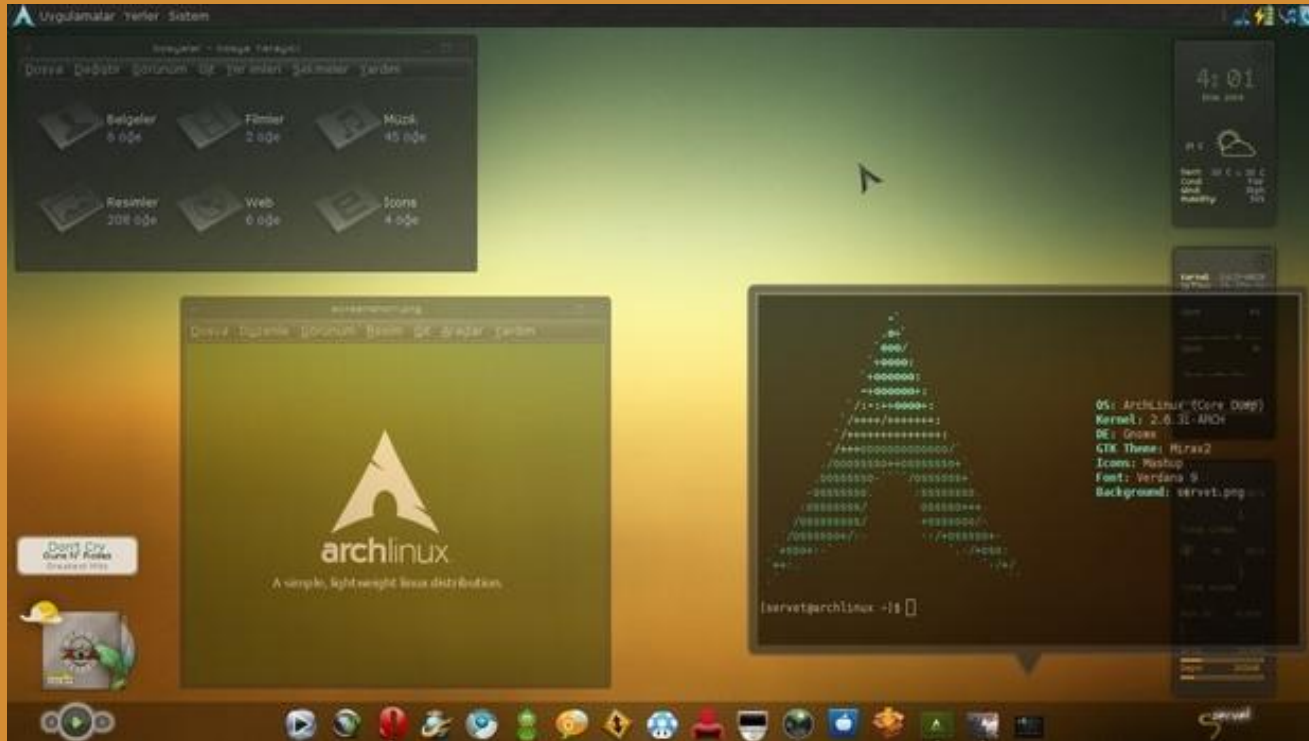


Gnu/Linux - Distro - Arch Linux

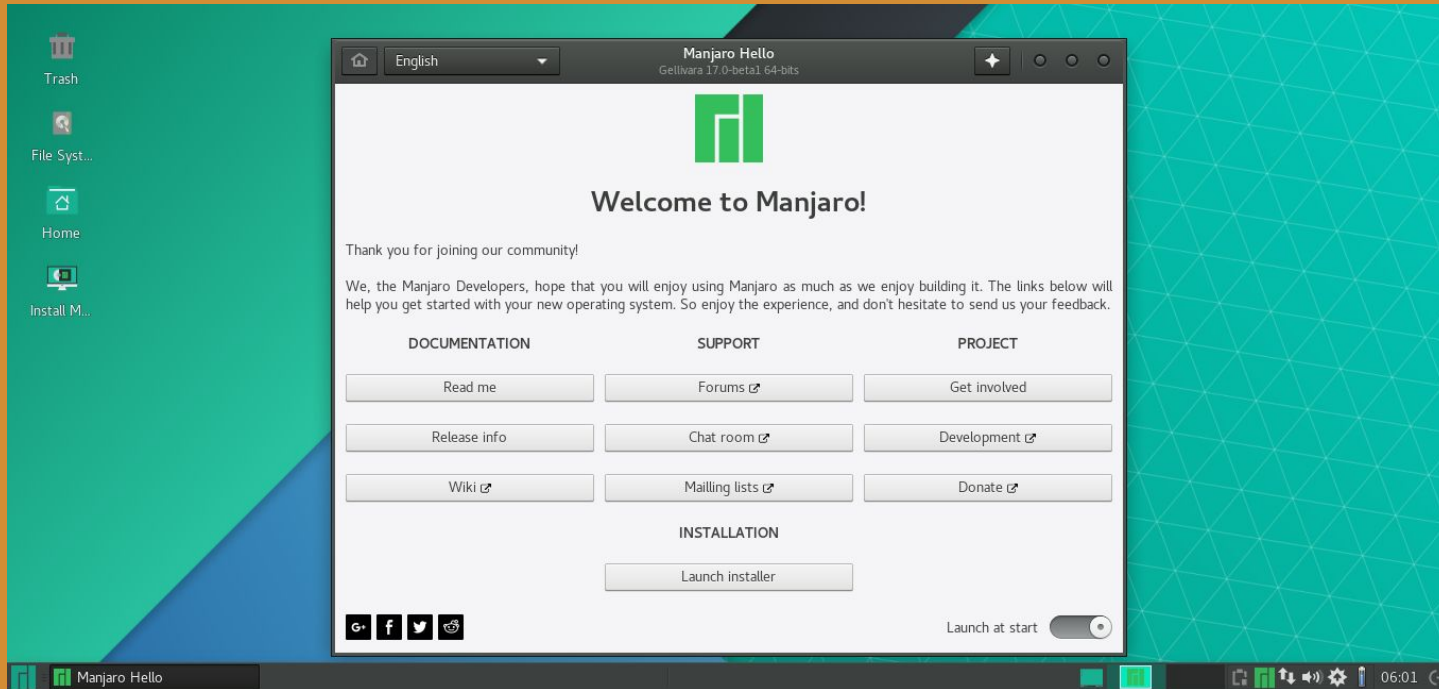
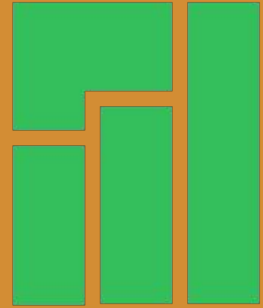
```
Arch Linux 4.9.11-1-ARCH (tty1)
archiso login: root (automatic login)
root@archiso ~ # _
```



Gnu/Linux - Distros - Arch Linux



Gnu/Linux - Distros - Manjaro

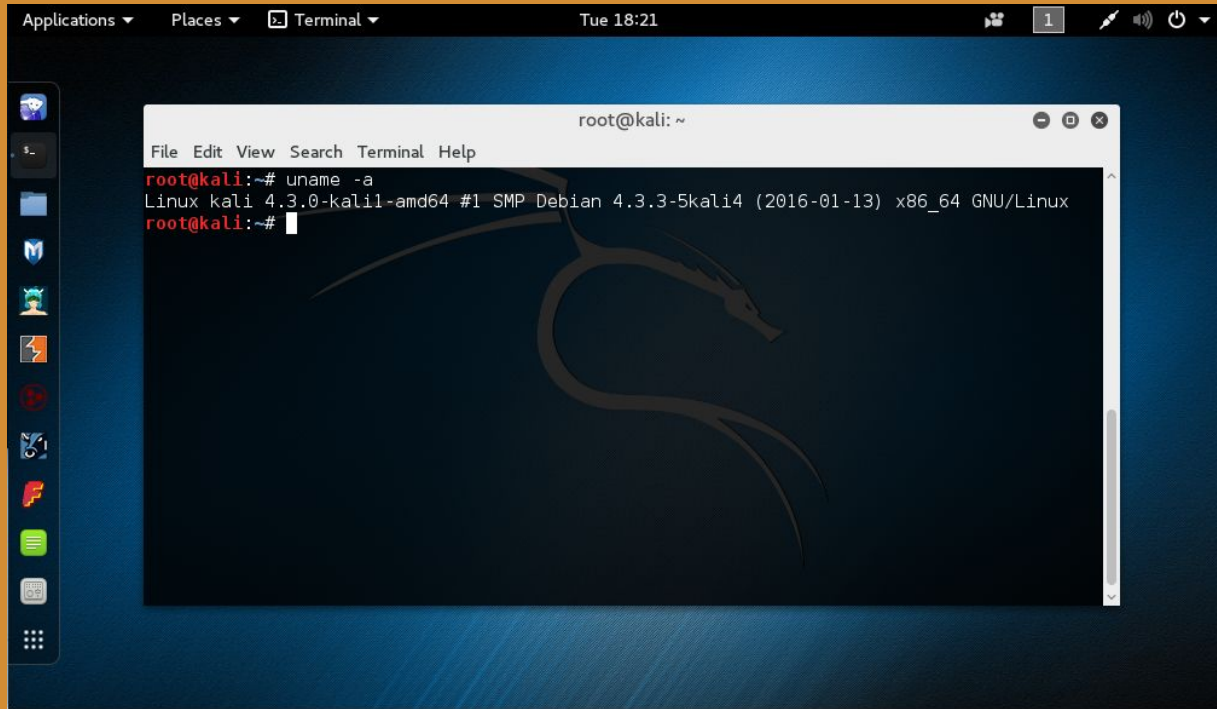


Gnu/Linux - Distros - Kali Linux

- > Lançado em 2013
- > Baseada em Debian, foco em segurança e pentest.
- > Possui diversas ferramentas de segurança e de forense digital já instaladas, e inúmeros scripts e ferramentas no repositório
- > Ideal para se carregar num pen drive e usar como ferramenta de teste de segurança.



Gnu/Linux - Distros - Kali Linux

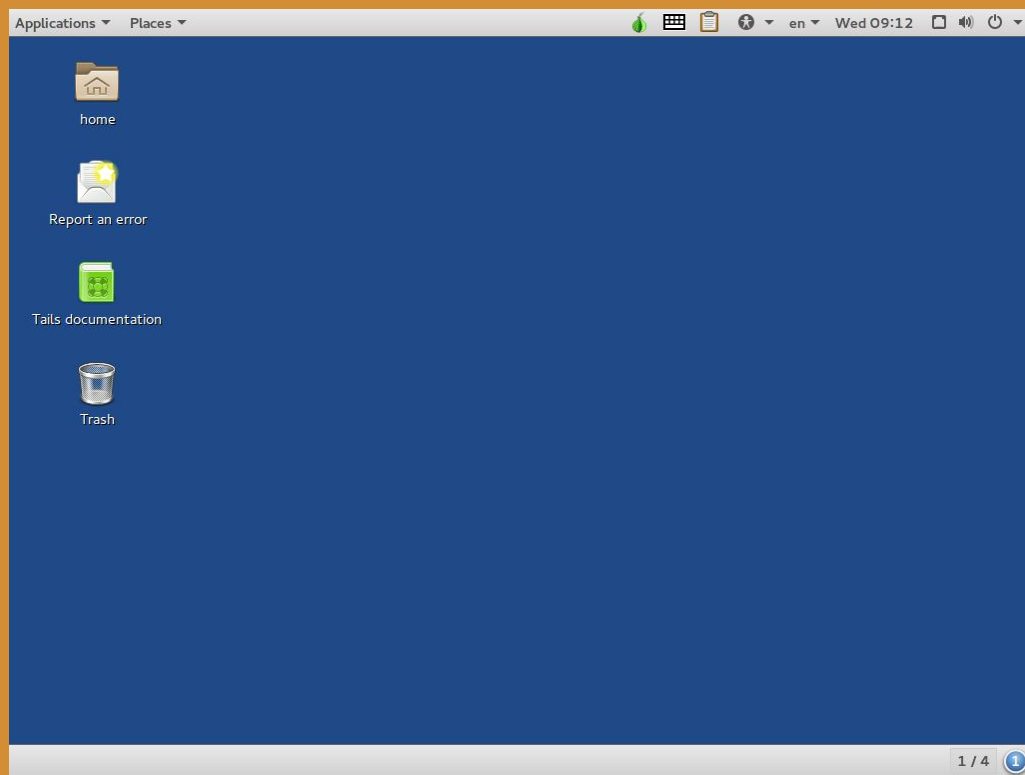


Gnu/Linux - Distros - Tails

- > Lançado em 2011
- > Baseada em Debian, foco em privacidade.
- > Não é possível instalar ela no Disco Rígido. Ela funciona somente na RAM do computador, para minimizar vestígios de uso.
- > Toda a conexão passa pela rede anônima Tor.



Gnu/Linux - Distro - Tails



Ambientes gráficos

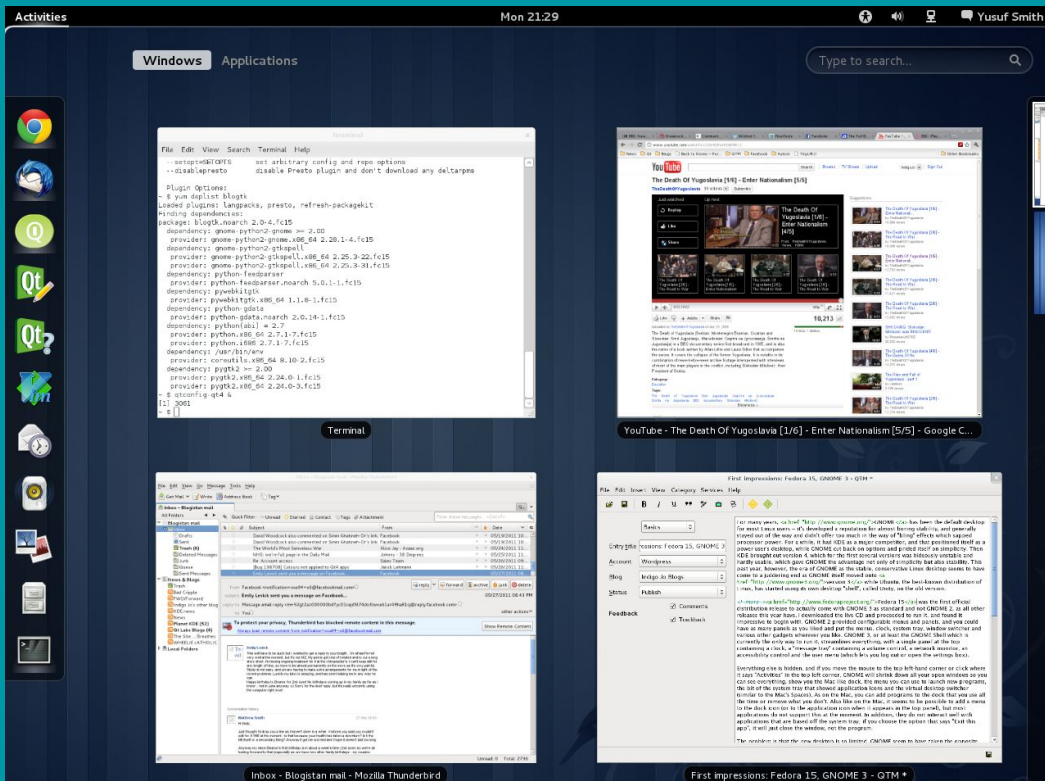
Gnu/Linux - Ambientes gráficos

- > Embora o terminal seja muito útil, muitas tarefas não podem ser executadas somente pelo terminal.
- > Os ambientes gráficos são softwares separados do núcleo do Linux e podem ser trocadas.
- > Muitas distros possuem uma como “padrão”, mas podem ser trocadas por outras.
- > Nos laboratórios do IC existem diversas opções de interface, mas o padrão é o GNOME3.

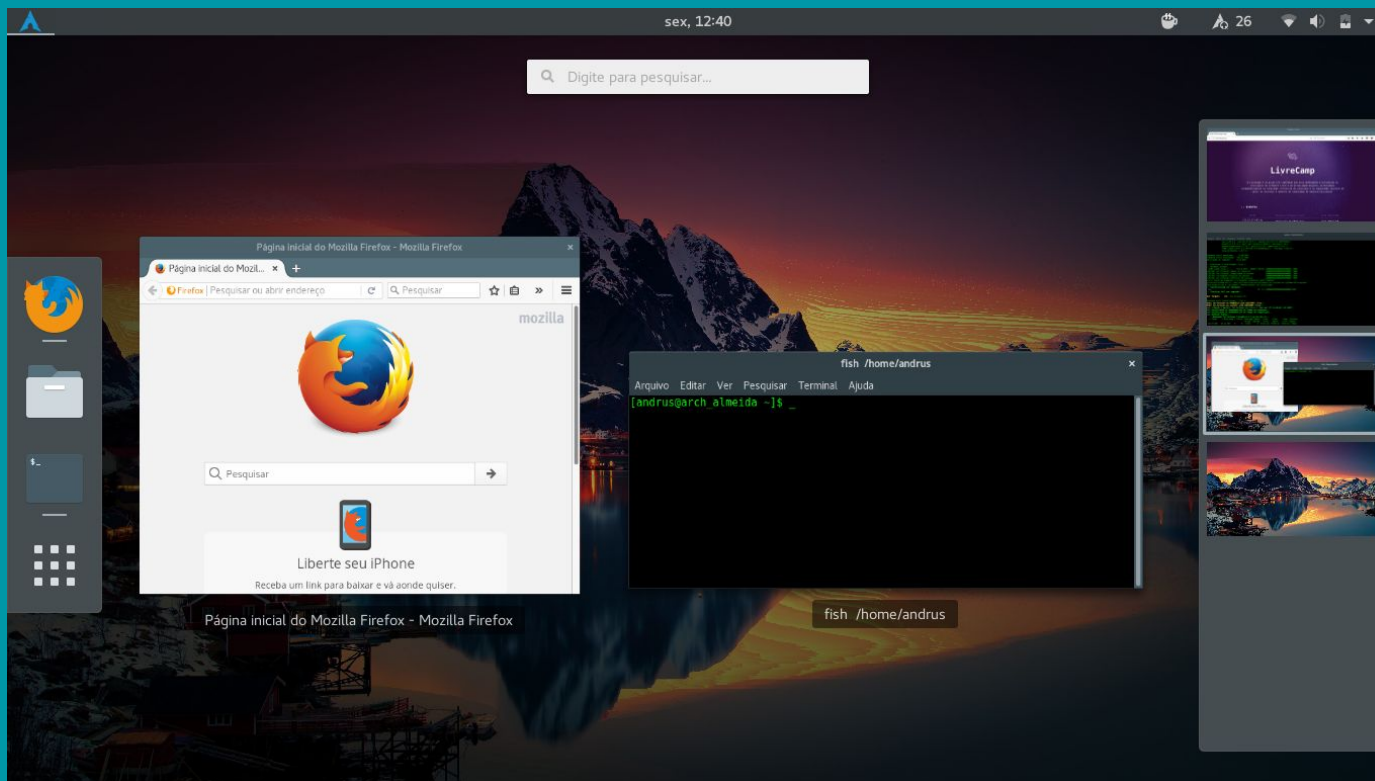
Gnu/Linux - GNOME 3

- > Interface moderna
- > Visa a facilidade de uso
- > Compatível com tablets
- > Pouco customizável

Gnu/Linux - GNOME 3



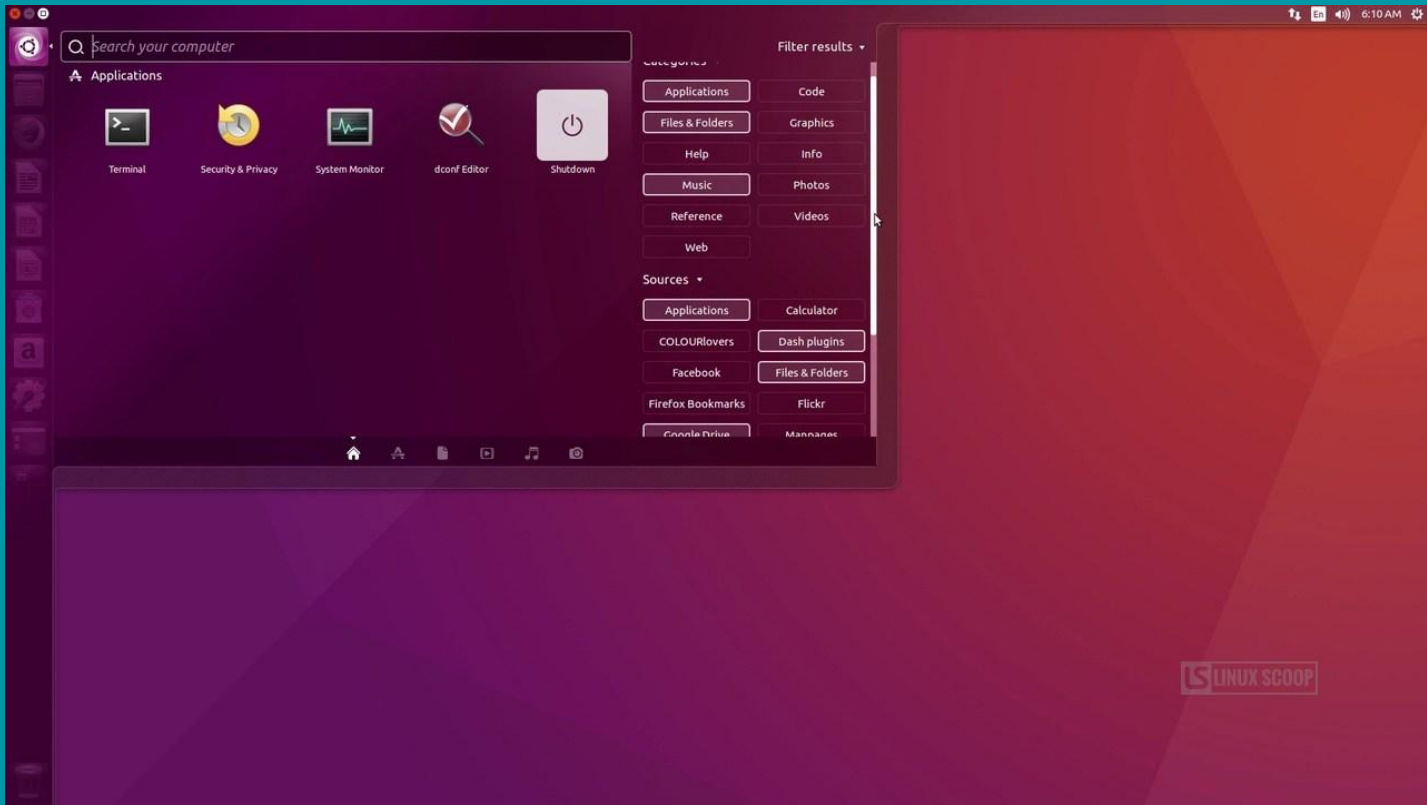
Gnu/Linux - GNOME 3



Gnu/Linux - Unity

- > Interface criada para o Ubuntu
- > Restringe as customizações, em troca de ser facilmente reconhecido e impedir que o usuário faça más escolhas.
- > Conecta o usuário além dos dados locais a diversos outros dados remotos, tais como: Amazon, Wikipedia e sites de compra

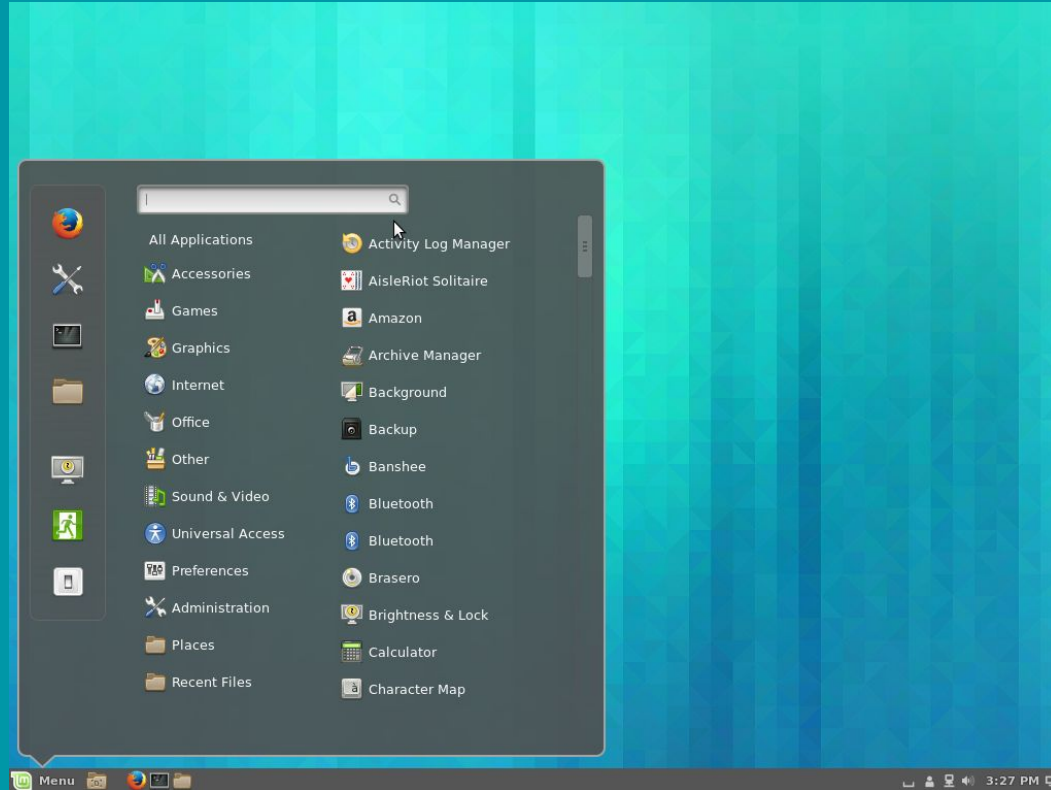
Gnu/Linux - Unity



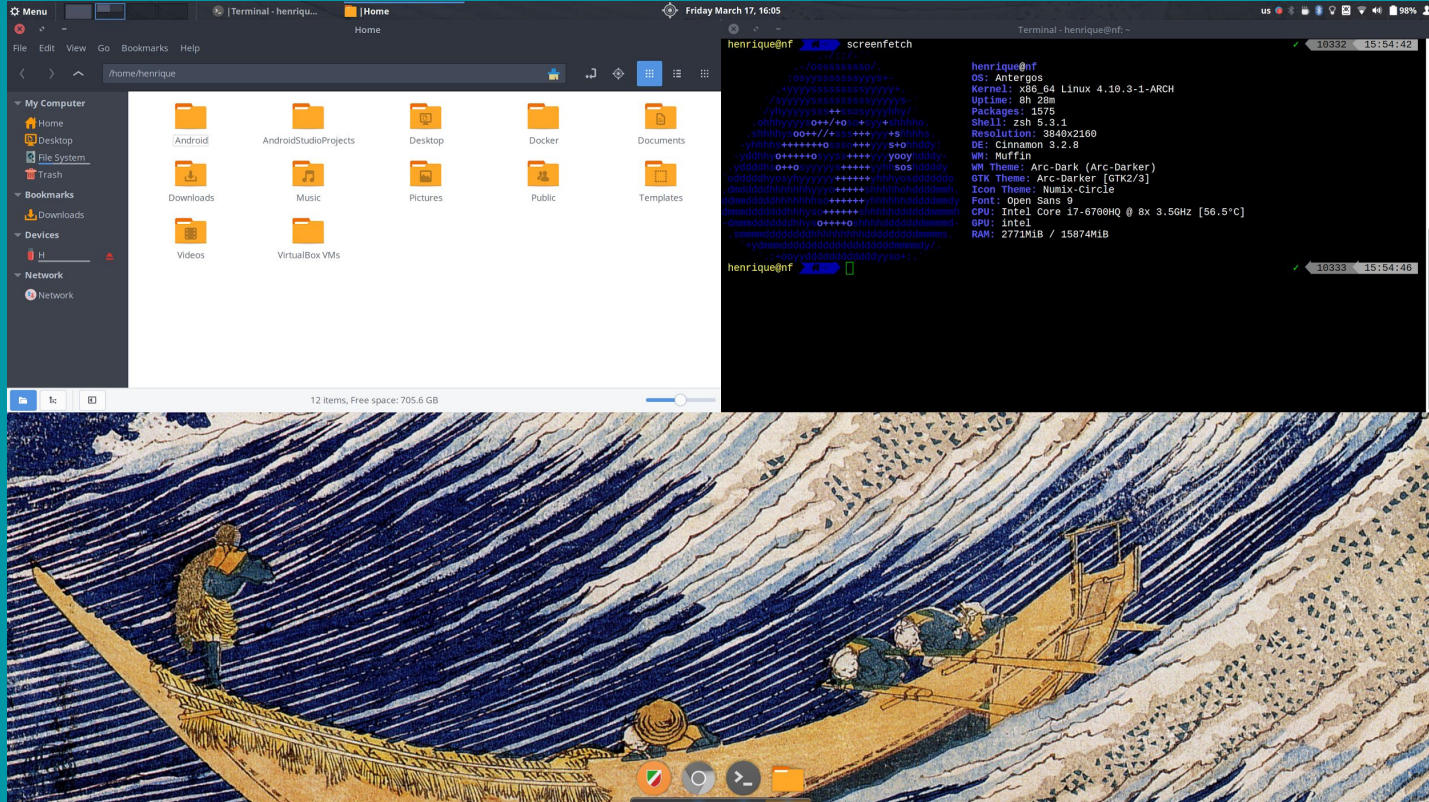
Gnu/Linux - Cinnamon

- > Desenvolvido e mantido pela comunidade do Linux Mint
- > Fork do gnome shell (Gnome2)
- > Altamente customizável
- > Contém traços do Gnome3

Gnu/Linux - Cinnamon



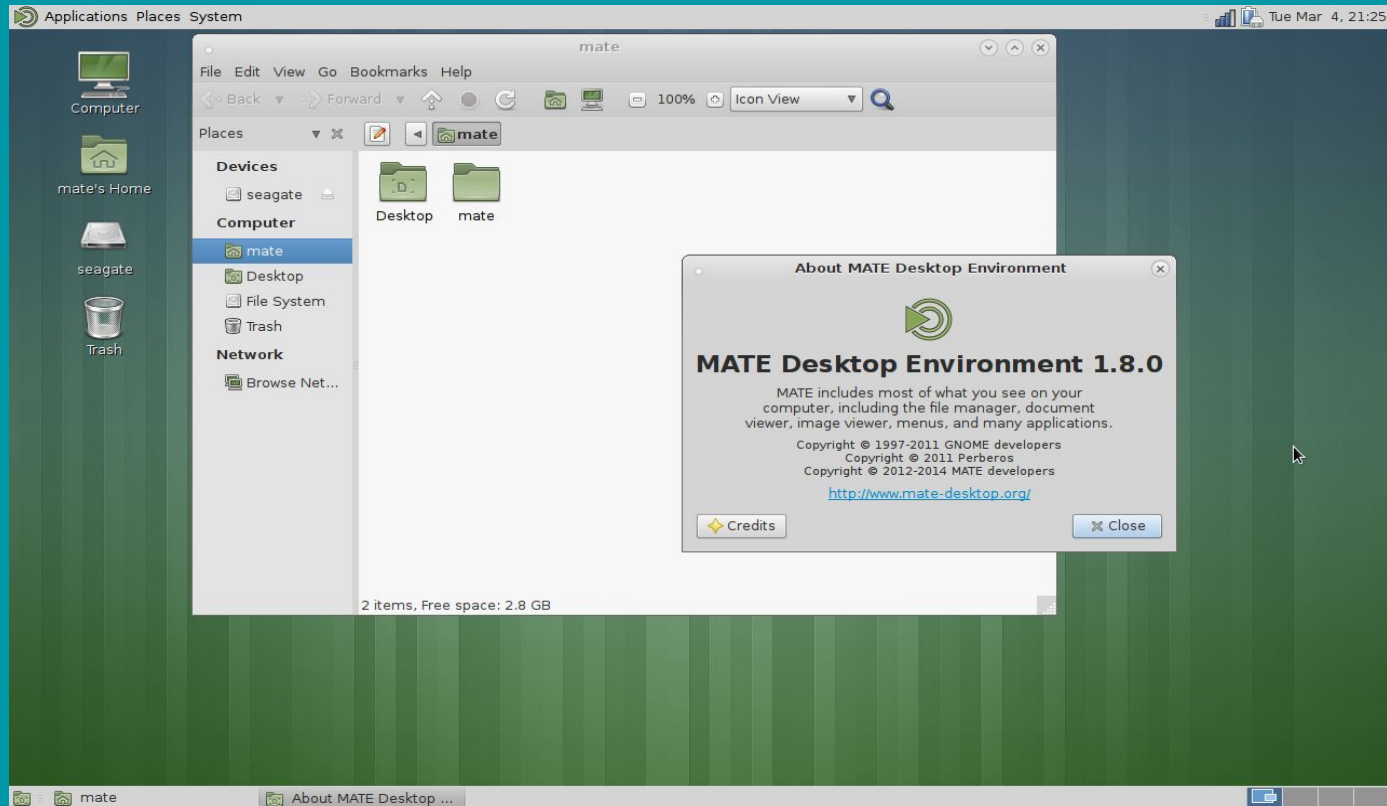
Gnu/Linux - Cinnamon



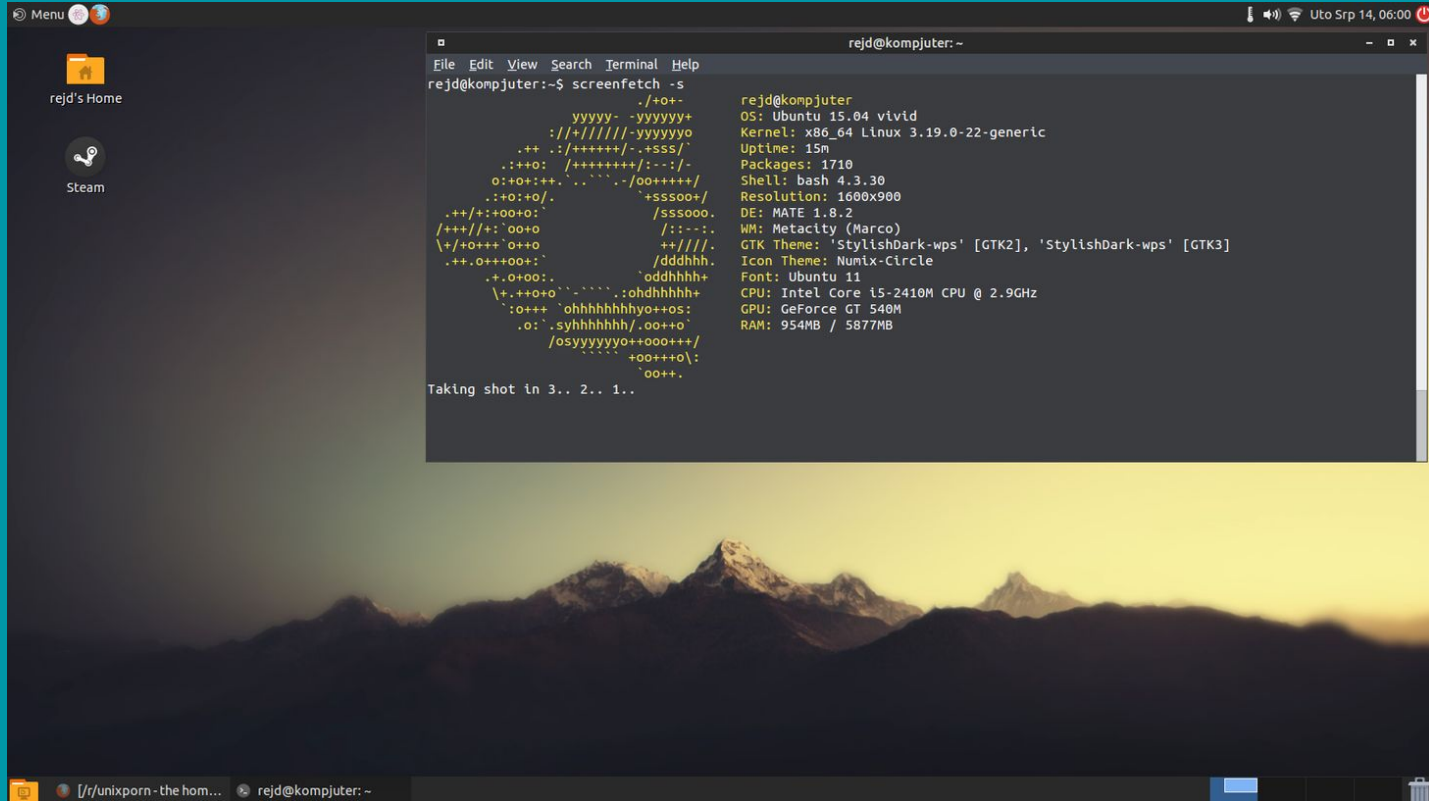
Gnu/Linux - MATE

- > Fork do gnome shell (Gnome2)
- > Altamente customizável
- > Mantém o conceito
e versatilidade dos panels

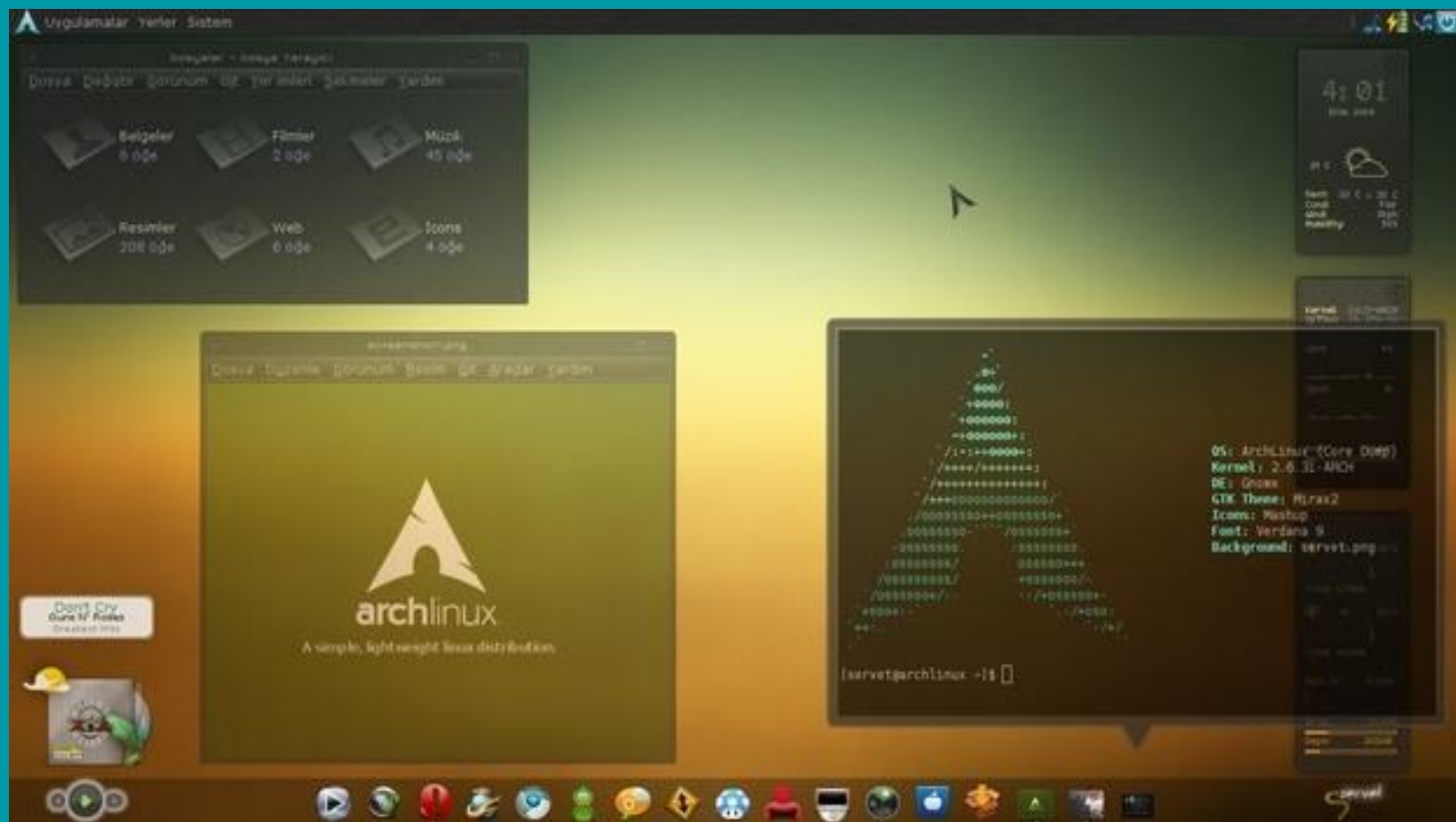
Gnu/Linux - MATE



Gnu/Linux - MATE



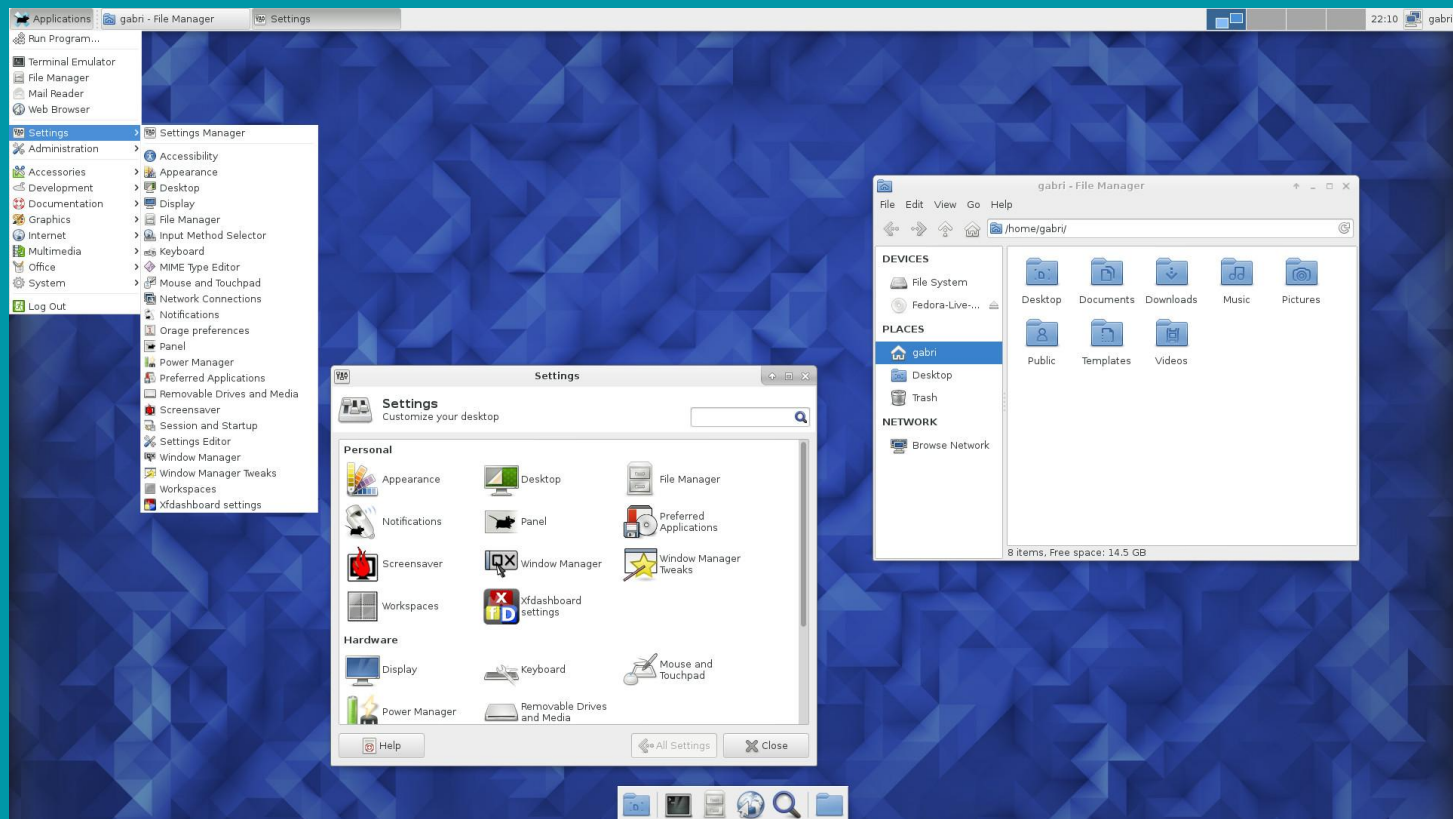
Gnu/Linux - MATE



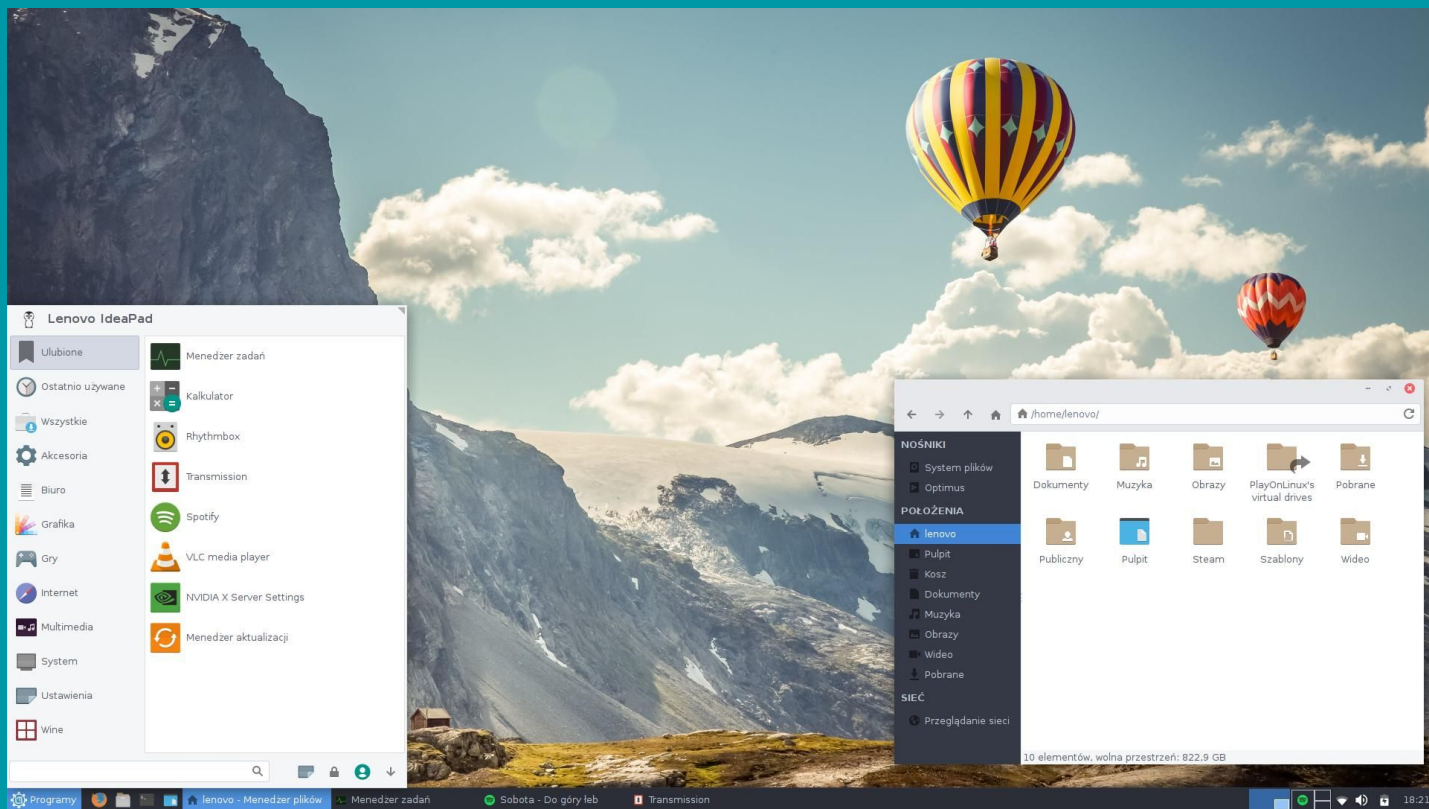
Gnu/Linux - XFCE

- > Tem como objetivo ser o mais leve possível, e ainda assim manter uma aparência legal
- > Bastante customizável

Gnu/Linux - XFCE



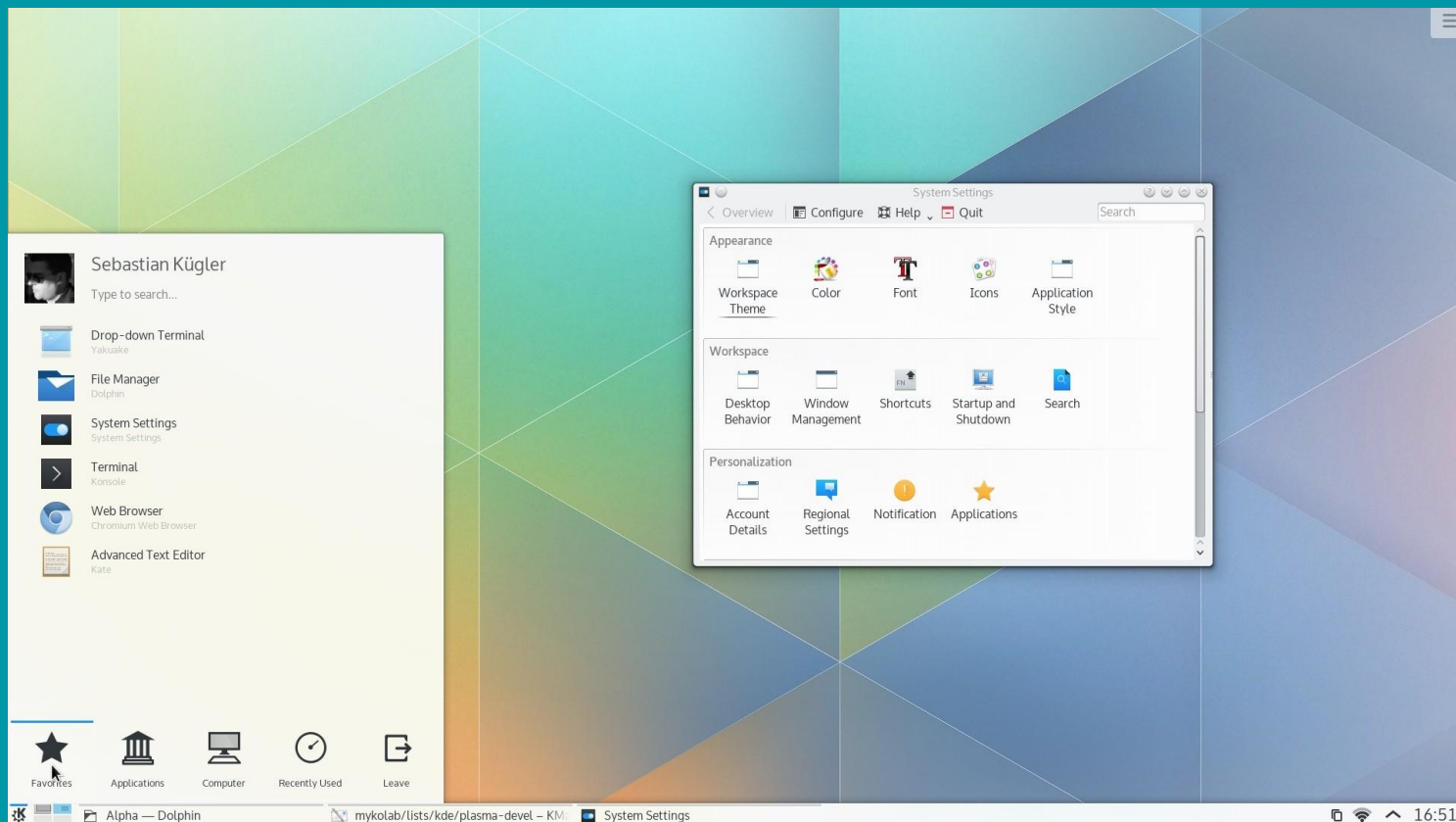
Gnu/Linux - XFCE



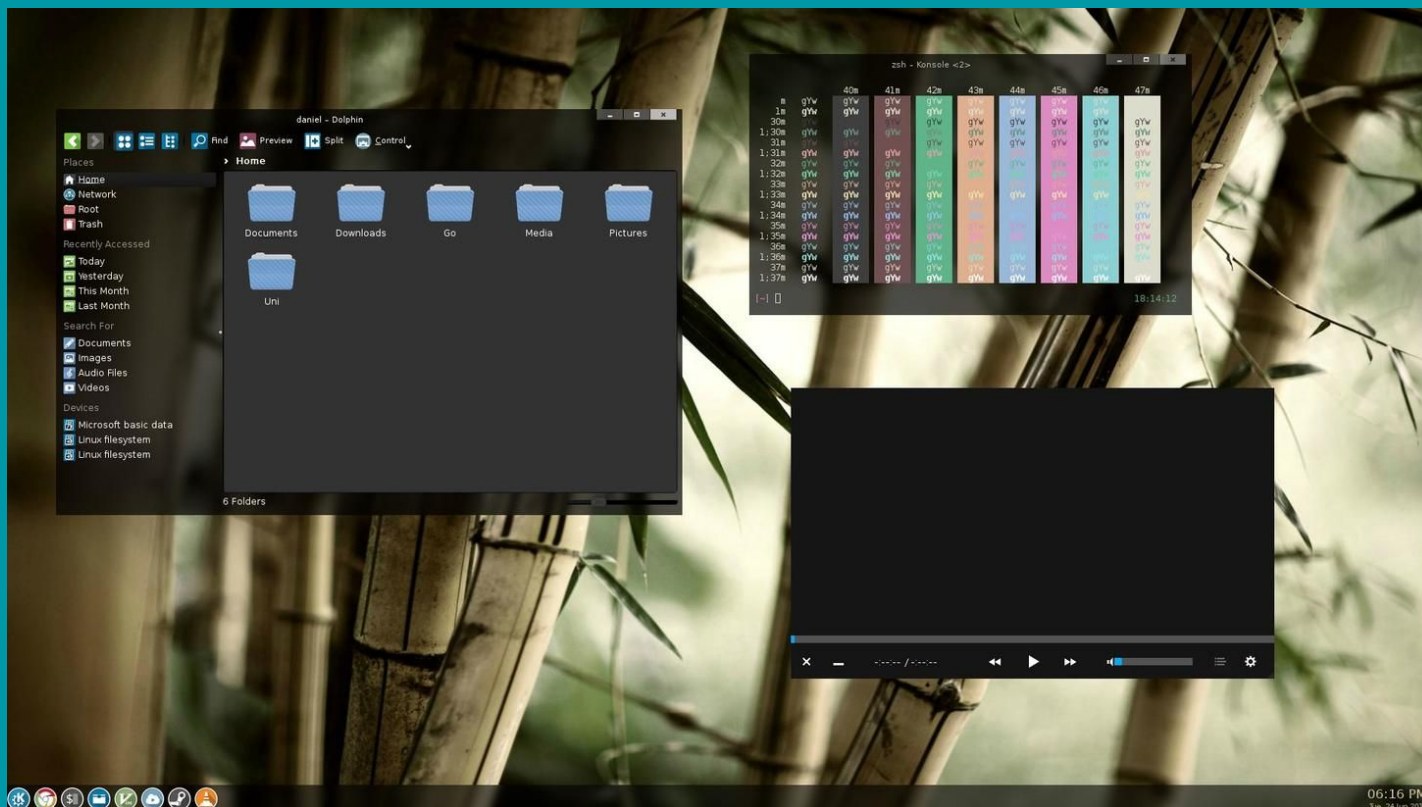
Gnu/Linux - KDE

- > Tem como objetivo ser uma interface ultra moderna e avançada
- > Bastante customizável

Gnu/Linux - KDE



Gnu/Linux - KDE



Feedback do MiniCurso

<https://goo.gl/BiXFz2>

InstallFest

22/03 - 13:00

24/03 - 16:00