

2026



# Cockpit Project

UFCD: 0826 – REDES - INSTALAÇÃO E CONFIGURAÇÃO  
TIAGO SILVA & RUI SOUSA

## Indíce

<b>Introdução .....</b>	2
<b>Apresentação da ferramenta.....</b>	35
<b>Arquitetura e funcionamento.....</b>	37
<b>Casos de uso .....</b>	41
<b>Iniciantes em Linux / utilizadores pouco avançados .....</b>	41
<b>Administradores experientes .....</b>	41
<b>Mercado de trabalho.....</b>	41
<b>Vantagens .....</b>	42
<b>Limitações.....</b>	42
<b>Comparação com alternativas .....</b>	43
<b>Cockpit vs Webmin.....</b>	43
<b>Cockpit .....</b>	43
<b>Webmin .....</b>	43
<b>Laboratórios propostos .....</b>	44
<b>Instrumentos de avaliação.....</b>	44
<b>Conclusão.....</b>	45
<b>Referências bibliográficas .....</b>	46

## Introdução

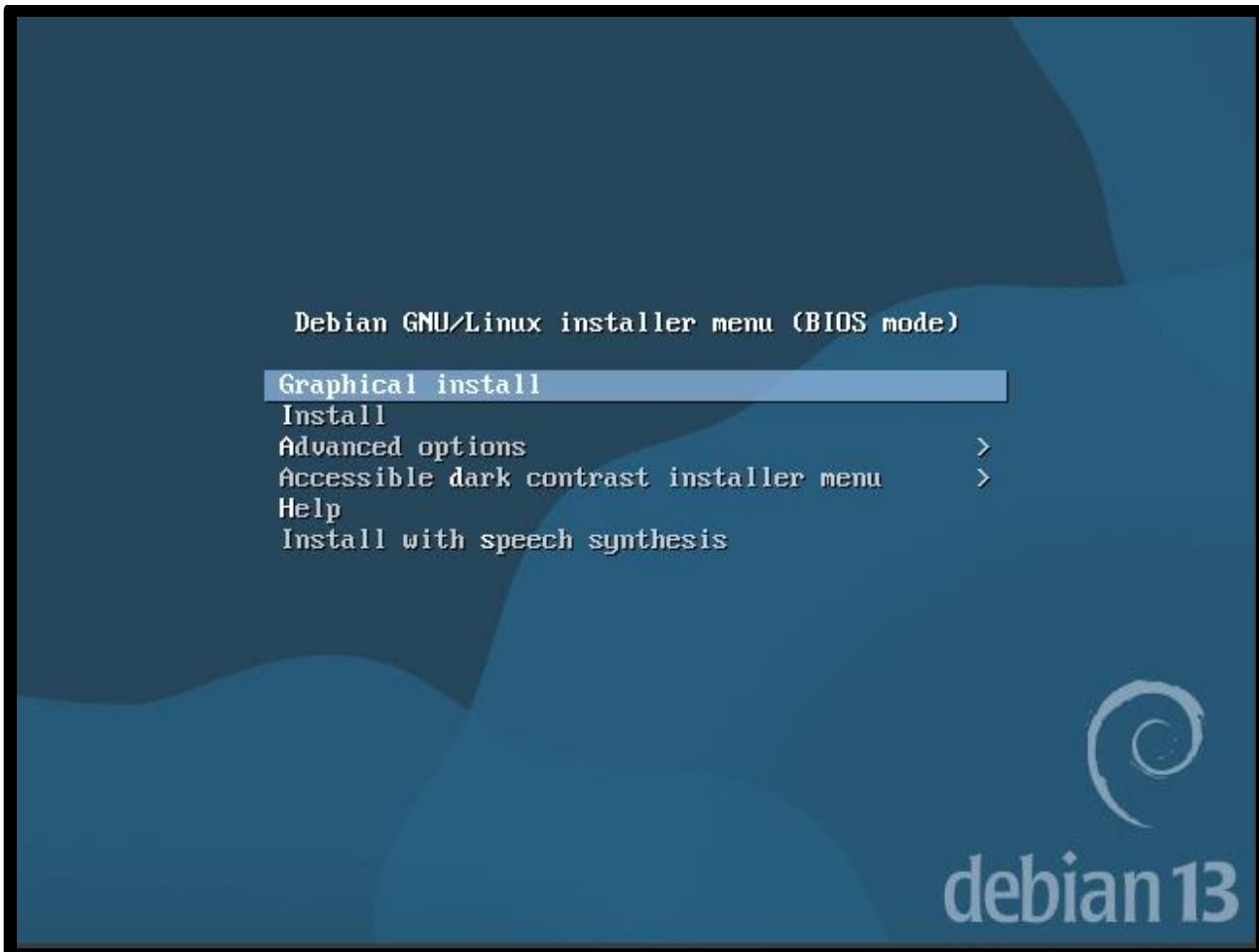
Este relatório tem como objetivo principal fornecer informação sobre o **Cockpit**, e explorar as suas funcionalidades que nos propõe, o seu impacto no seu uso para novos administradores e experientes, e a sua integração com os fluxos de trabalho Linux existentes.

No Cockpit podemos fazer a gestão de serviços, visualização de *logs* e configuração de rede até à administração de armazenamento e contentores (como o Podman), o Cockpit demonstra ser uma ferramenta **não intrusiva e modular** que utiliza as APIs (Application Programming Interface é um conjunto de regras, protocolos e ferramentas que permite que diferentes softwares se comuniquem entre si ou com o próprio sistema operacional) do sistema de forma nativa.

O Cockpit não pretende substituir as ferramentas de linha de comandos, mas sim complementá-las e tornar o sistema Linux mais **acessível e de fácil descoberta**. A sua arquitetura leve e a capacidade de gerir múltiplos servidores a partir de um único painel de controlo (*dashboard*) o que com que seja um recurso bastante útil na administração moderna de infraestruturas Linux.

Agora vamos mostrar passo a passo de como instalar o Cockpit Project numa máquina Linux Debian 13 com GUI (ambiente gráfico):

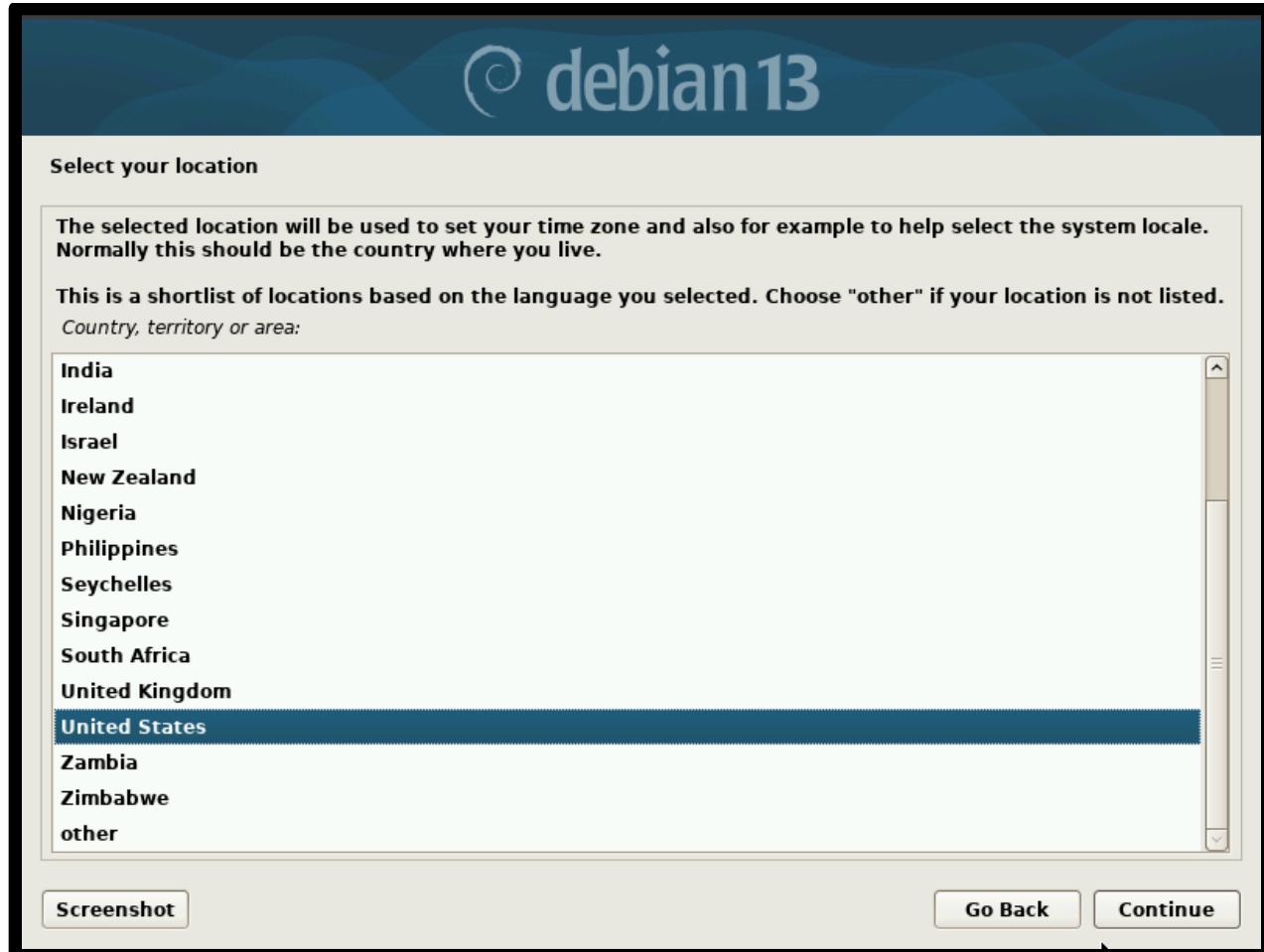
1º Passo – Aqui selecionamos a primeira opção “Graphical Install”.



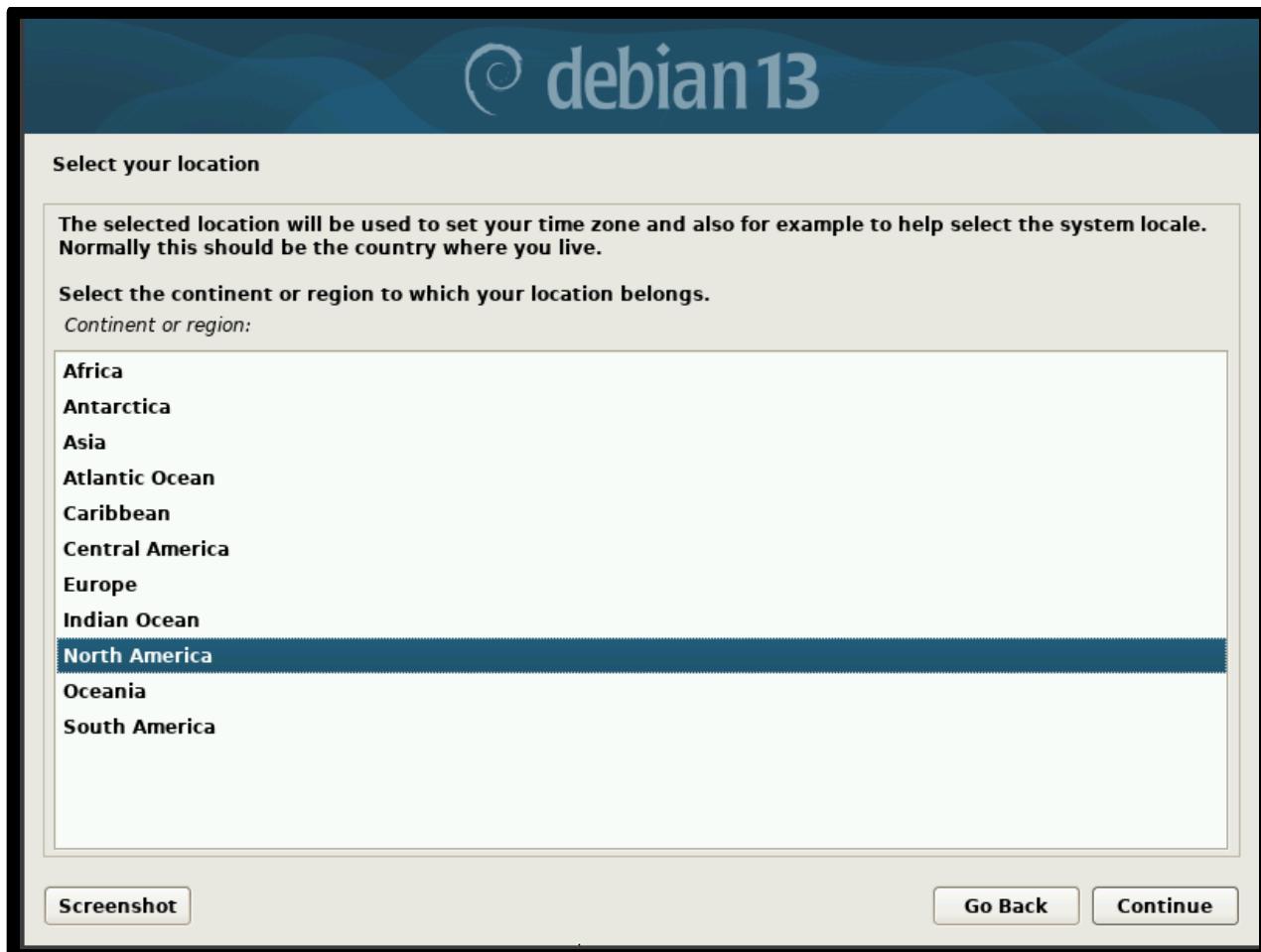
2º Passo – Nesta parte da instalação depende de cada um, pois é a língua que a nossa máquina vai ter, por defeito está escolhido “English – English”, porque é uma língua muito falada.



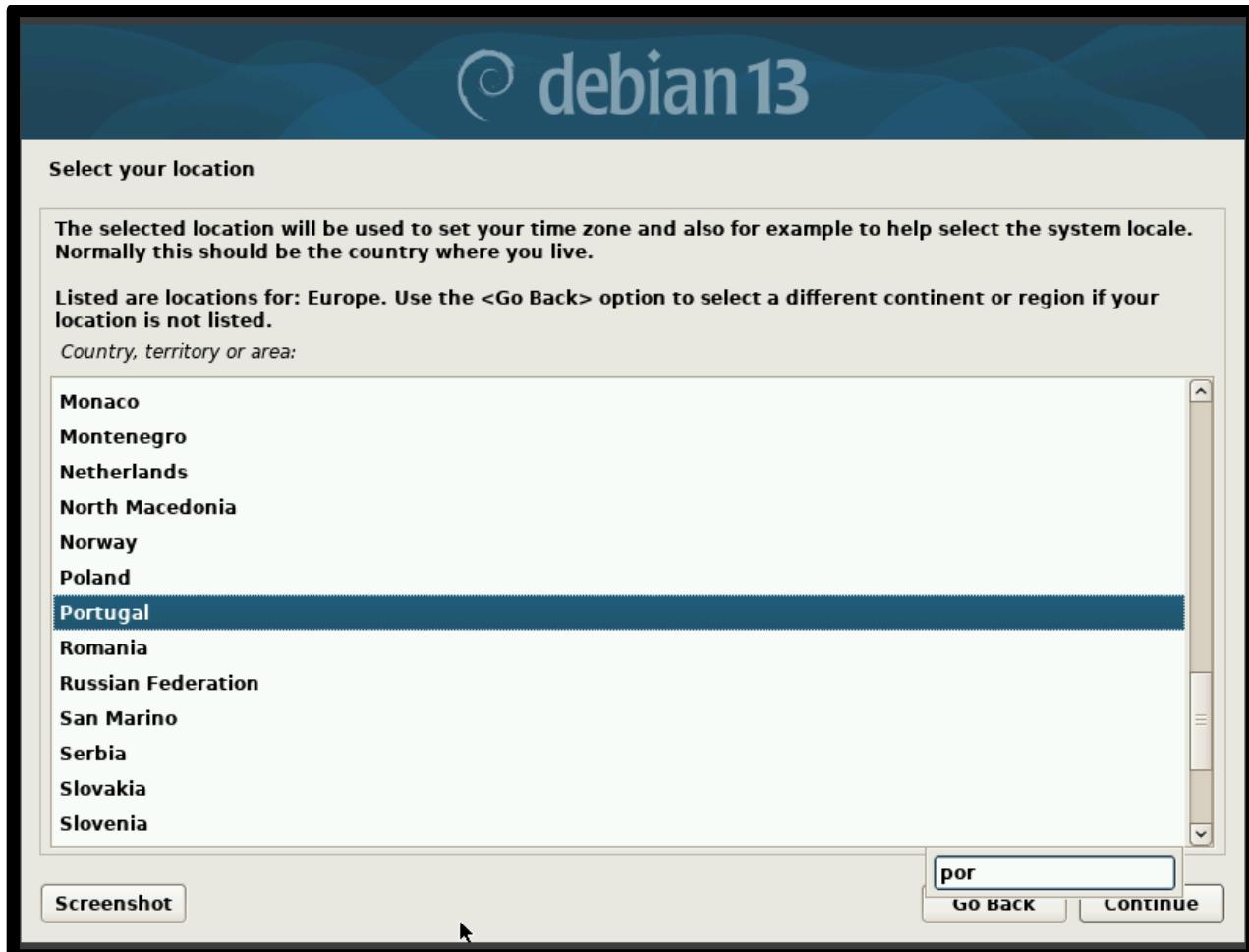
3º Passo – Agora selecionamos onde nos encontramos no momento (país) , ou seja, caso queiramos selecionar Portugal temos de clicar em “other”



Aqui selecionamos Europe e procuramos por Portugal.



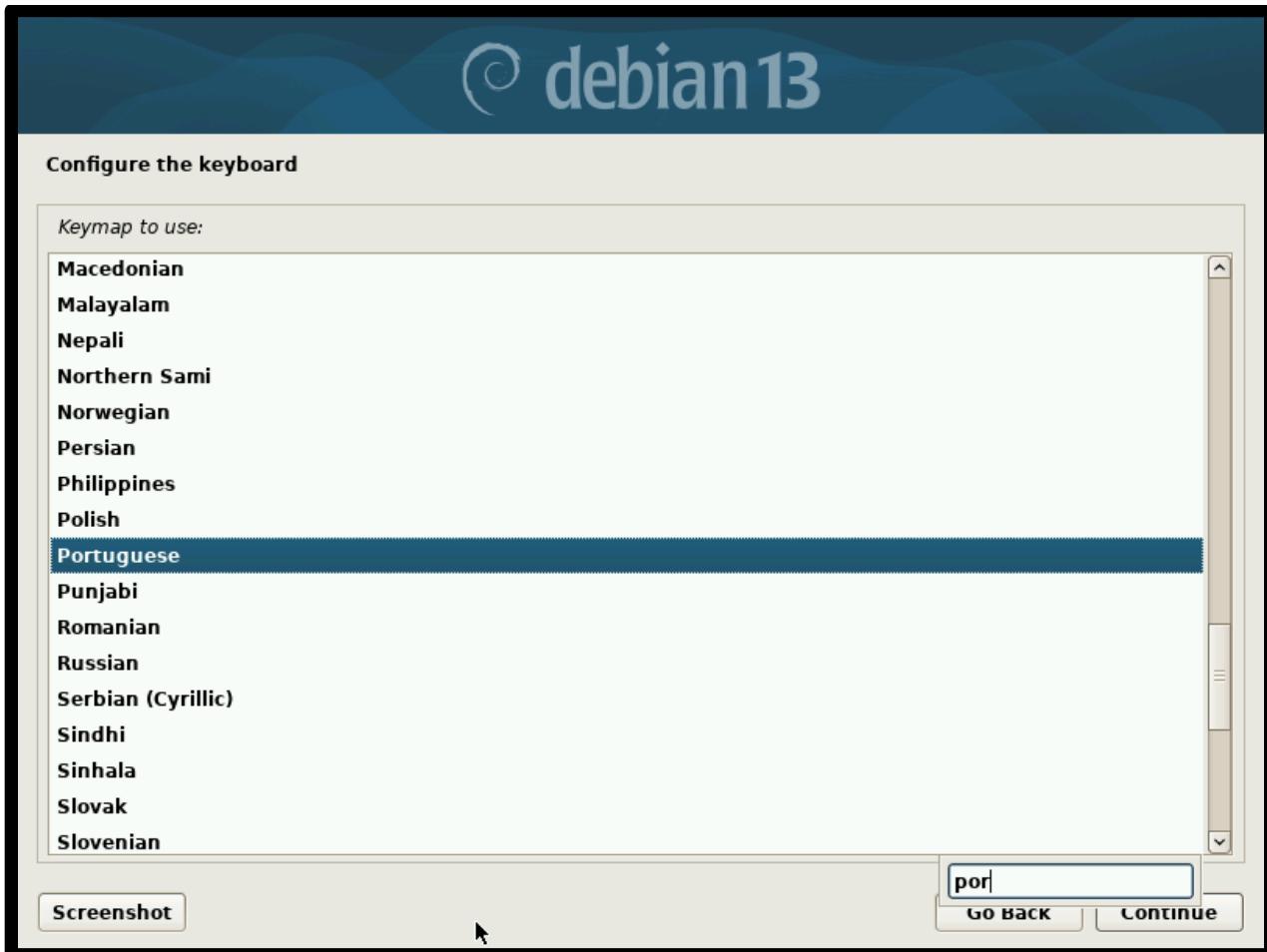
Para ser mais fácil a procura podemos pesquisar, como podemos pesquisar basta carregar nas letras iniciais do país ou escrever o nome todo do país, no exemplo abaixo basta escrever “Por” e o sistema encontra logo Portugal. Depois de selecionarmos a nossa localização podemos continuar a instalação.



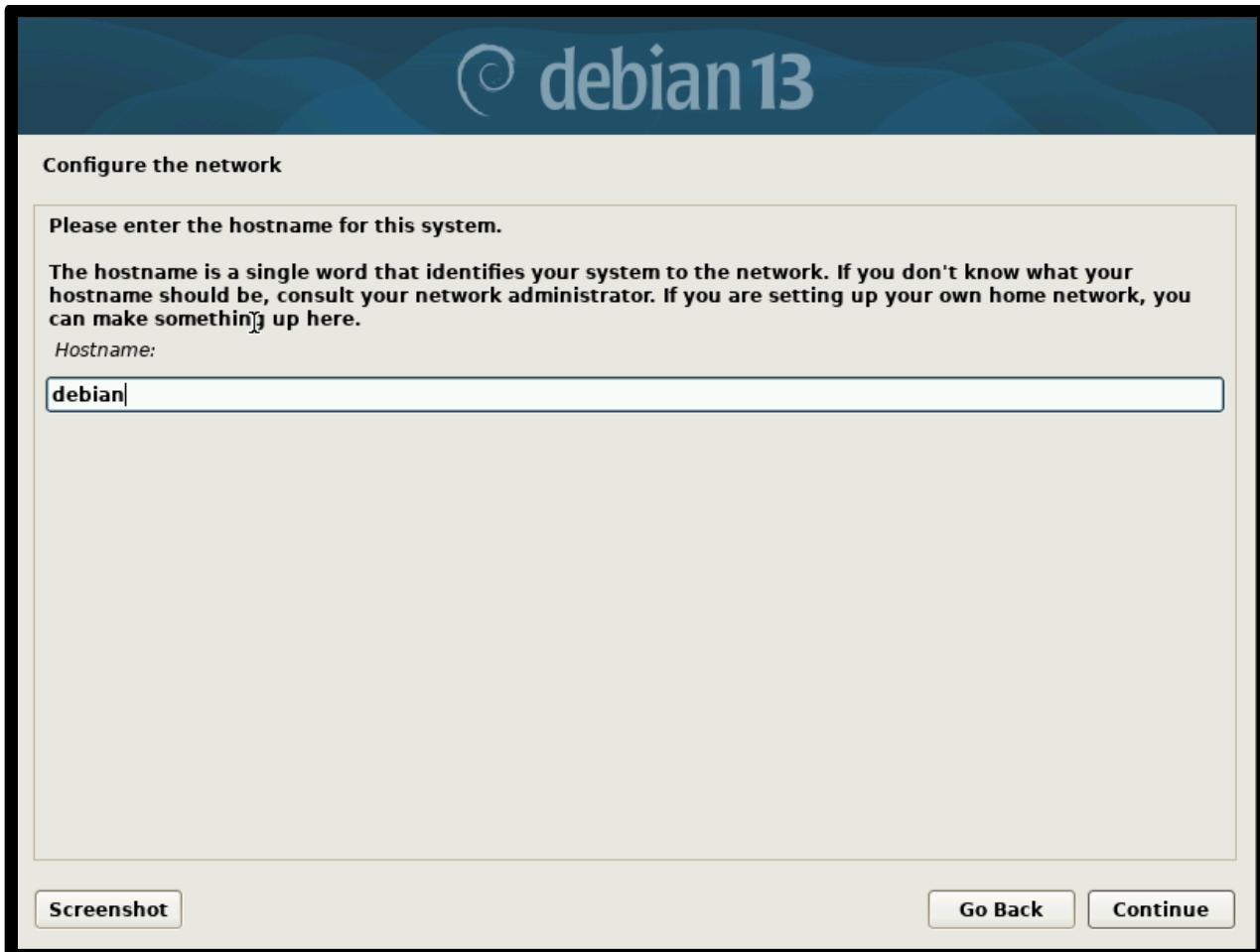
4º Passo – Aqui normalmente é sempre selecionado o que está por defeito que é “United States”.



5º Passo – Neste passo agora vamos configurar a linguagem teclado é como na localização depende de cada um e caso queiramos uma língua em específico basta clicarmos nas iniciais que queremos por “por” e aparece o teclado português. Mas claro que formos de outro país vamos escolher a linguagem do país em que nos encontramos.



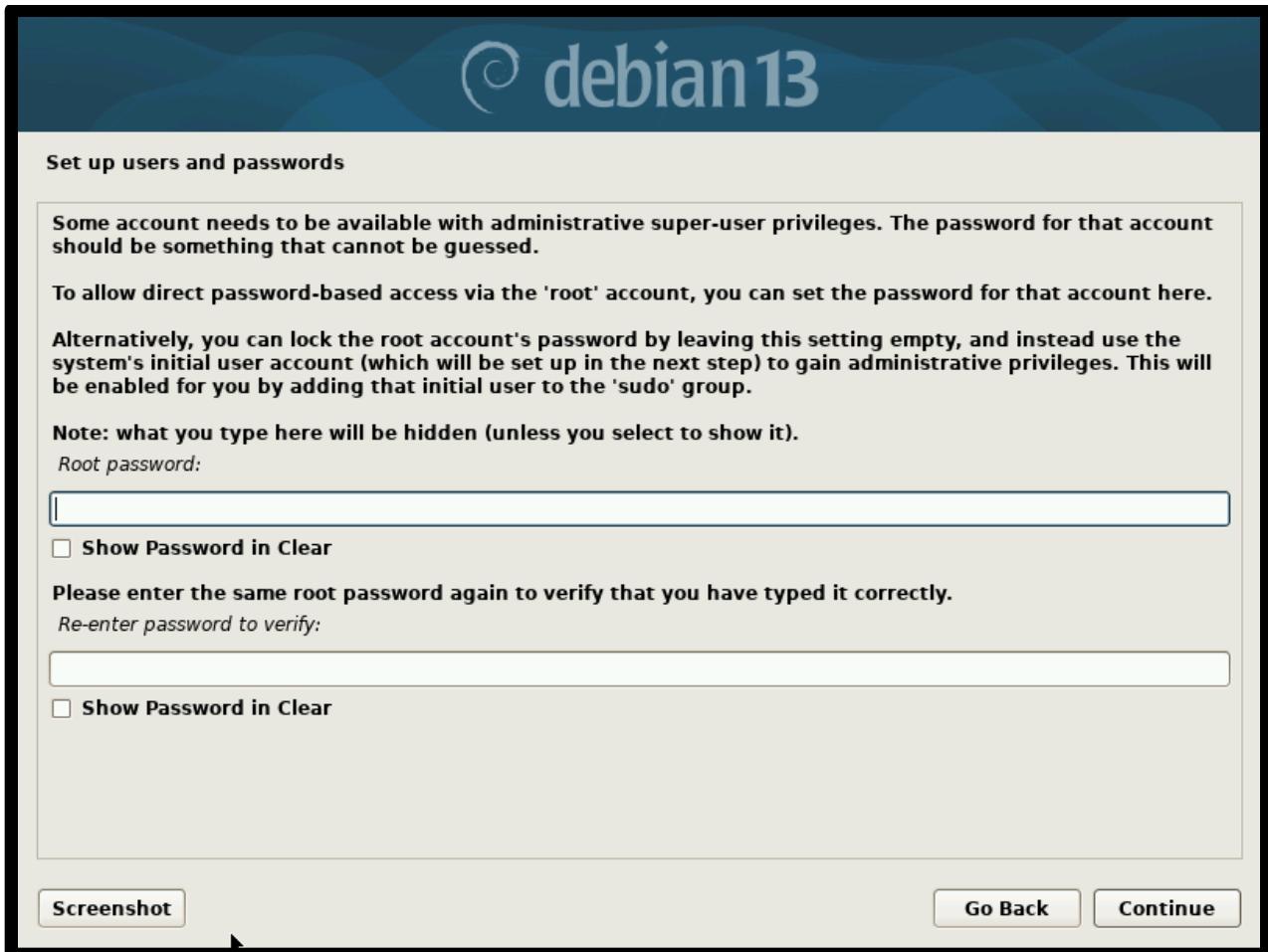
6º Passo – Agora vamos começar a configurar a *Network* da máquina e começa por nos pedir um *hostname* podemos colocar o nome que quisermos, mas claro se nos for exigido um nome devemos colocar o nome exigido. Por defeito o hostname que está é ***Debian***



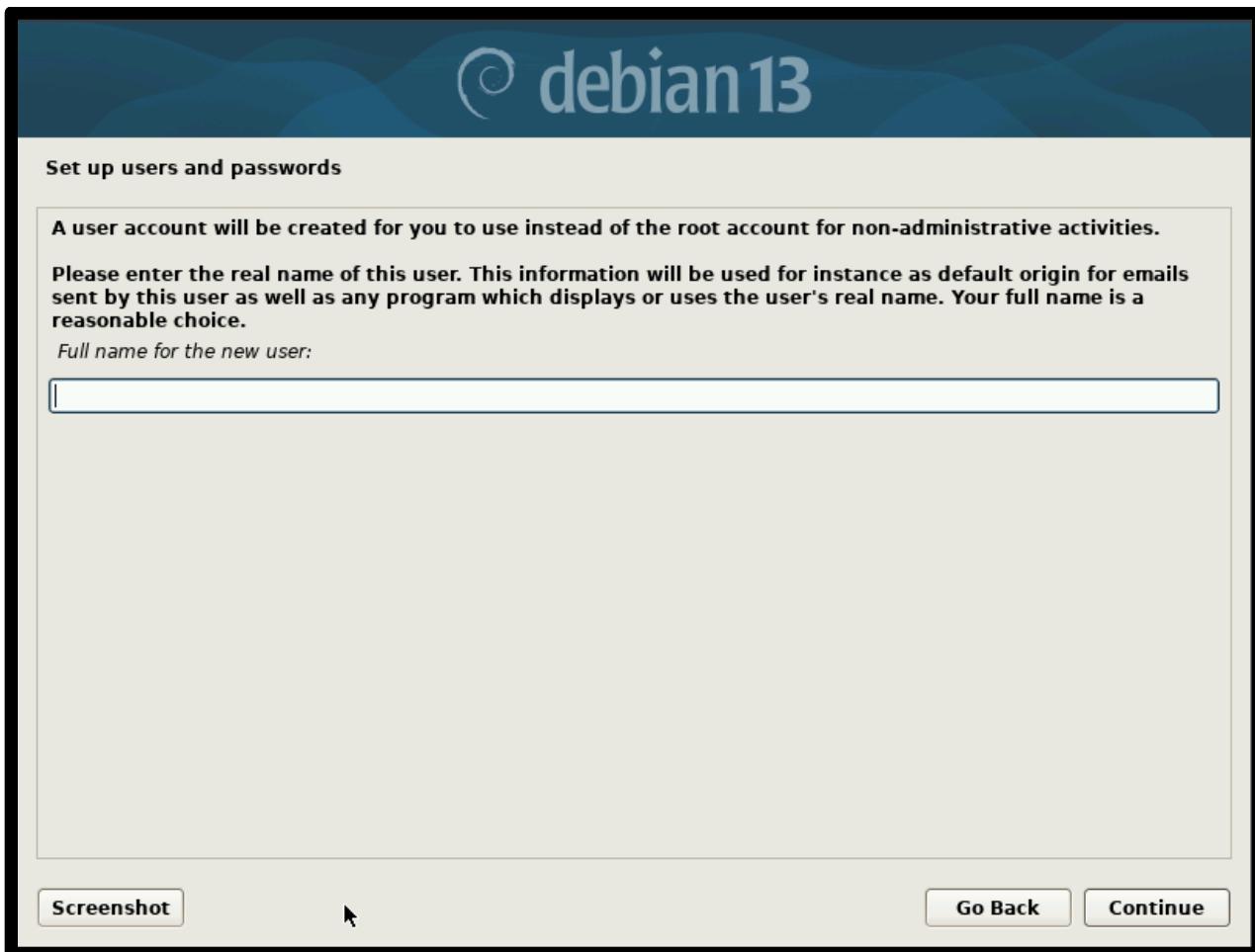
7º Passo – Agora vamos configurar o domínio, por defeito está escrito o domínio em que a máquina que está a usar se encontra.



8º Passo – Neste próximo passo é algo bastante simples temos de criar uma palavra-passe para o Utilizador Root que é o Administrador da nossa máquina. É aconselhável usar palavras-passes grandes com letras maiúsculas e minúsculas e com caracteres especiais tais como: @, !, #, £, \$, §, %, &, \*, etc.



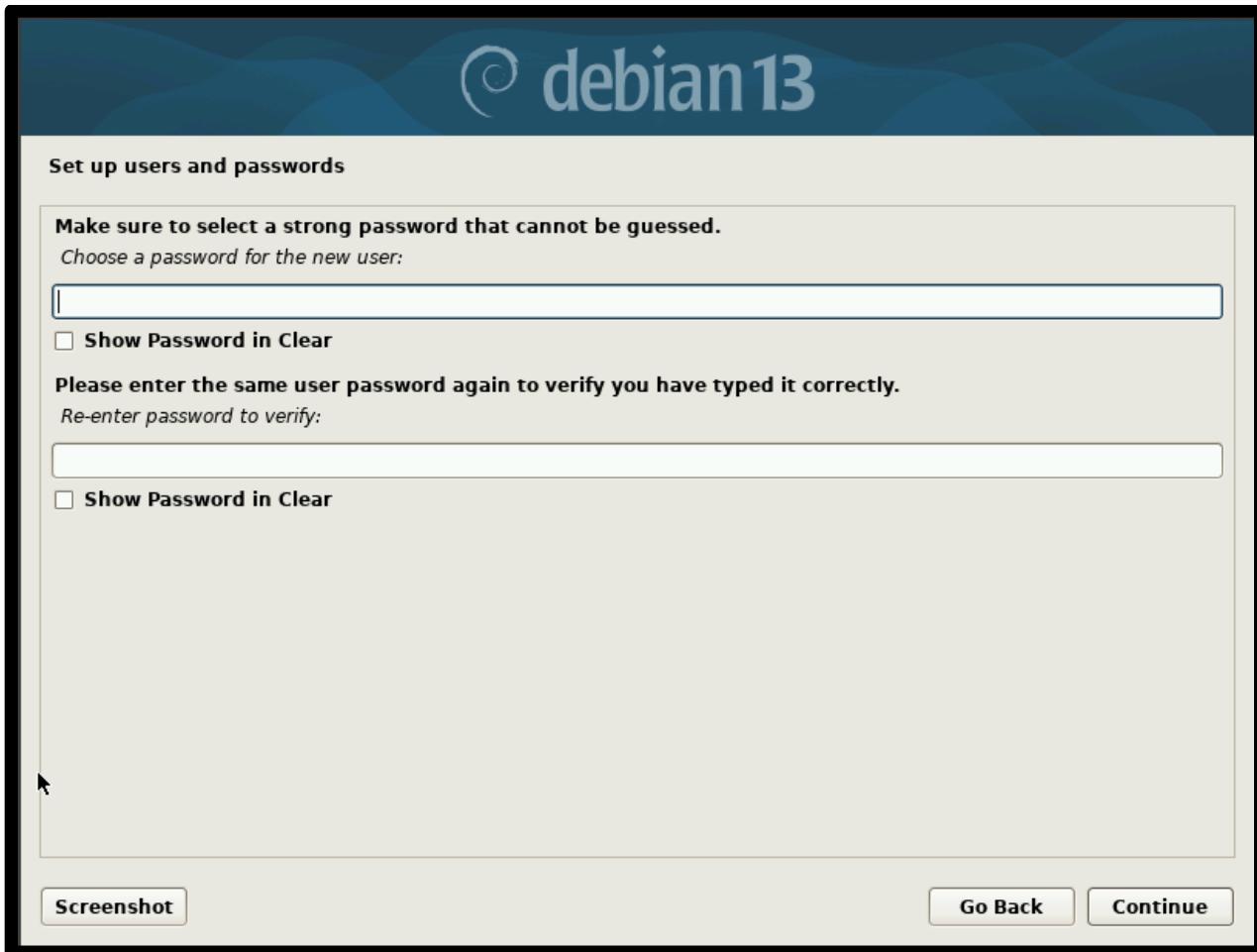
9º Passo – Temos de criar um Utilizador novo o nome de Utilizador vai de cada um podemos colocar o que quisermos.



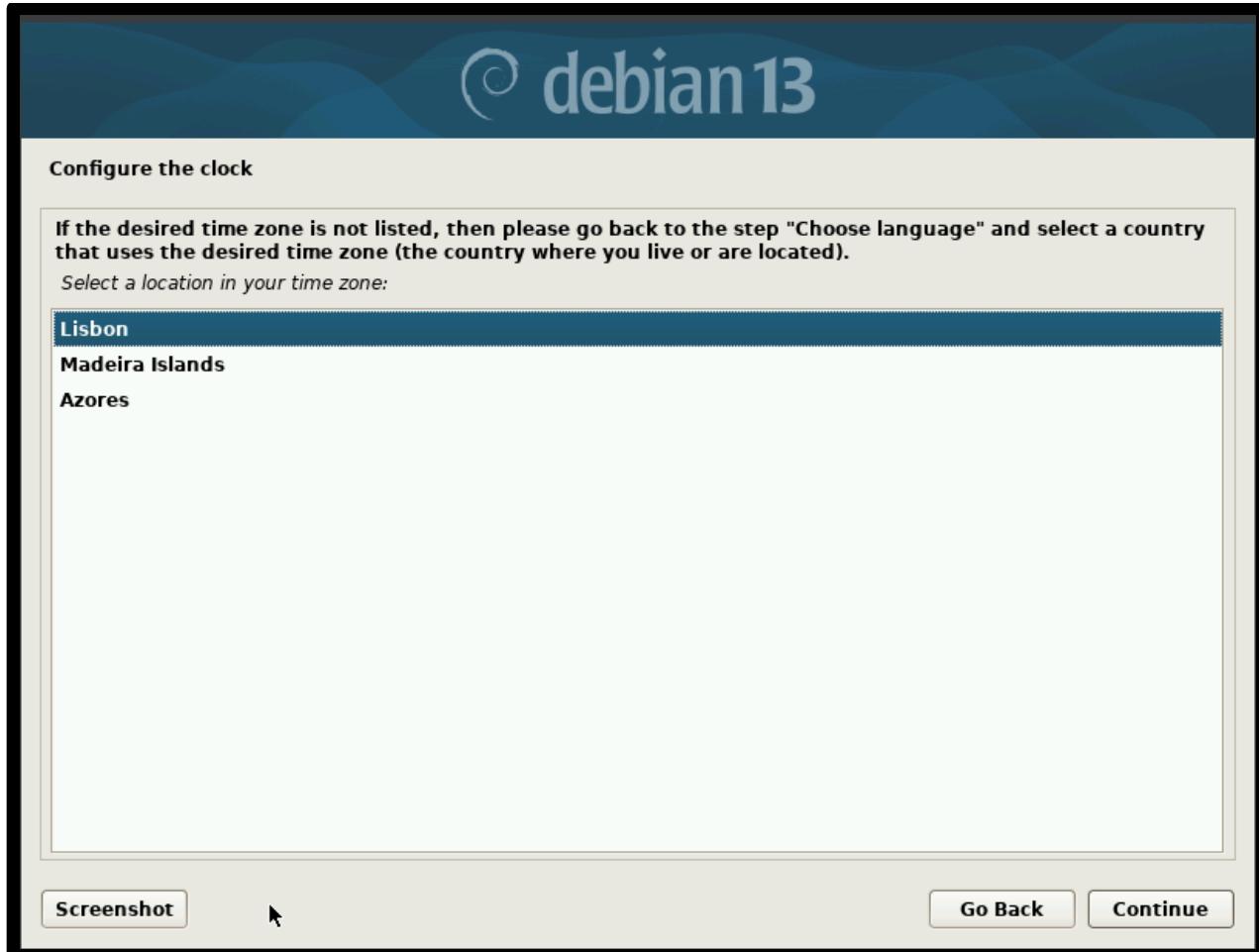
Depois vai nos pedir um *username* para o nosso utilizador normalmente deixa-se o que está por defeito que é o nome que escolhemos para o nosso utilizador, neste caso escolhemos teste



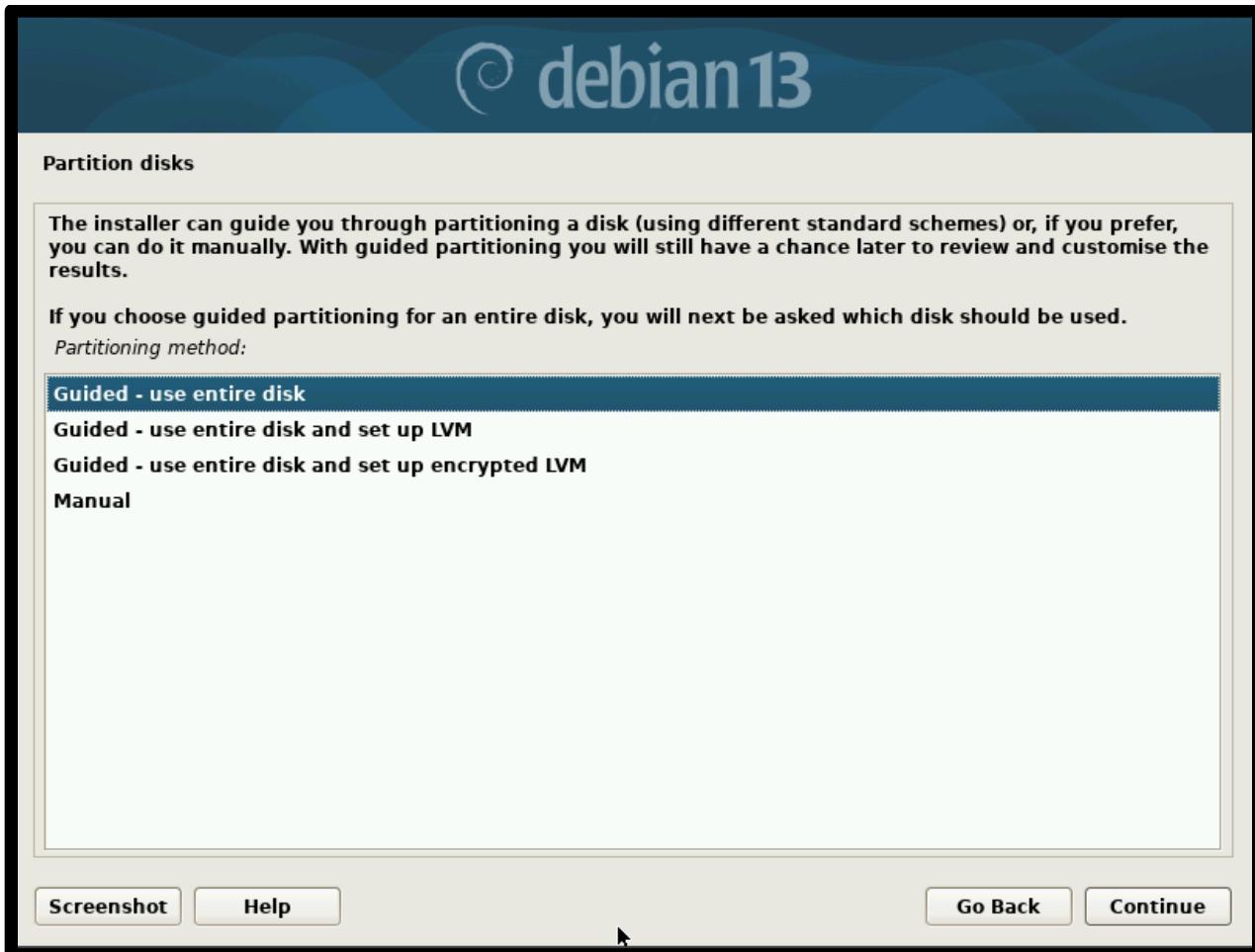
Depois vamos ter de configurar uma palavra-passe para o utilizador também que vai de cada um também escolhemos aquilo que queremos



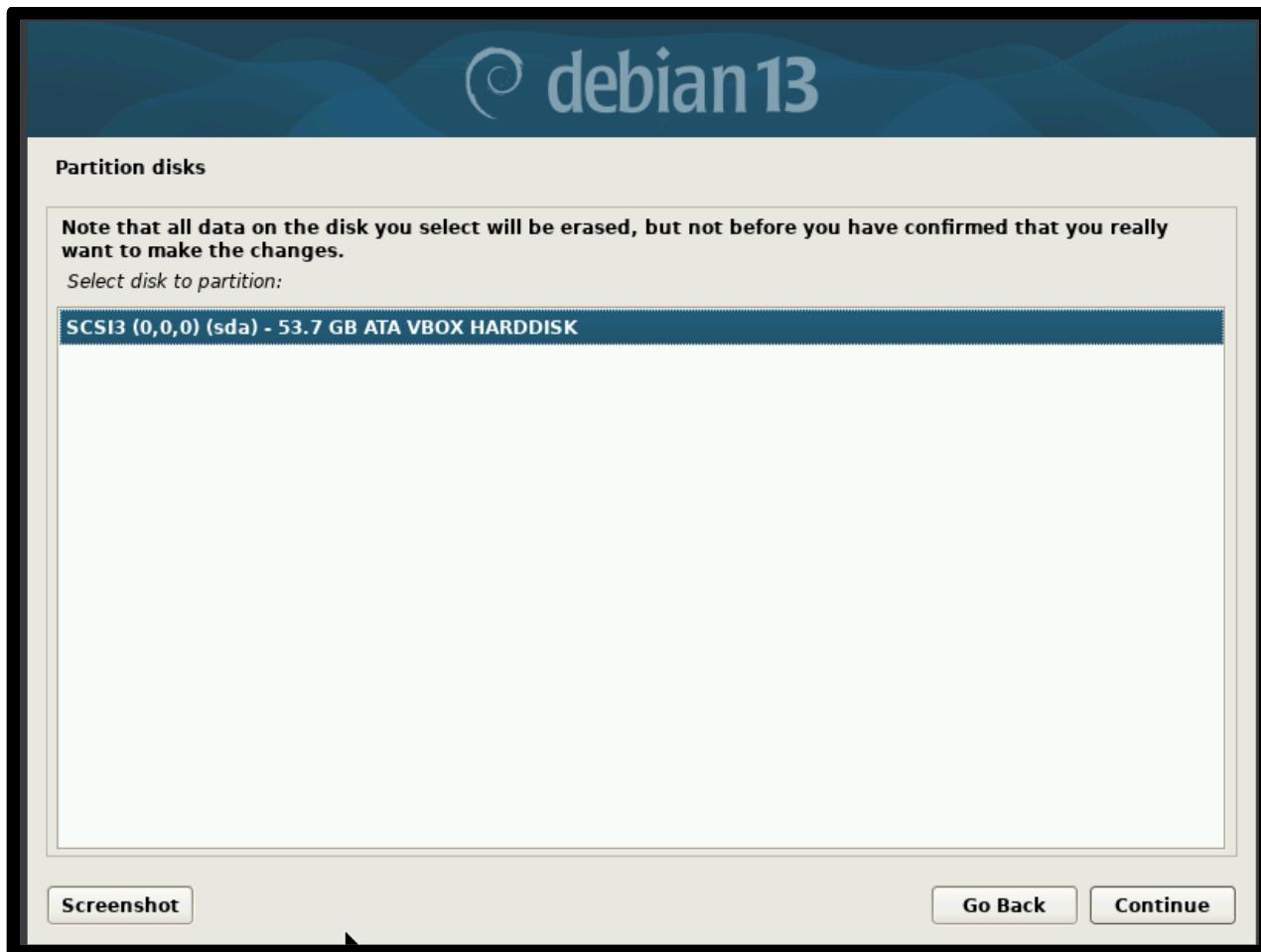
10º Passo – Neste passo, como nós escolhemos Portugal vai nos aparecer umas opções, mas se tivéssemos selecionado outro país iam aparecer outras opções, aqui cada um escolhe o que quer. Normalmente é selecionado o sítio onde o horário coincide com a hora do local onde nos encontramos.



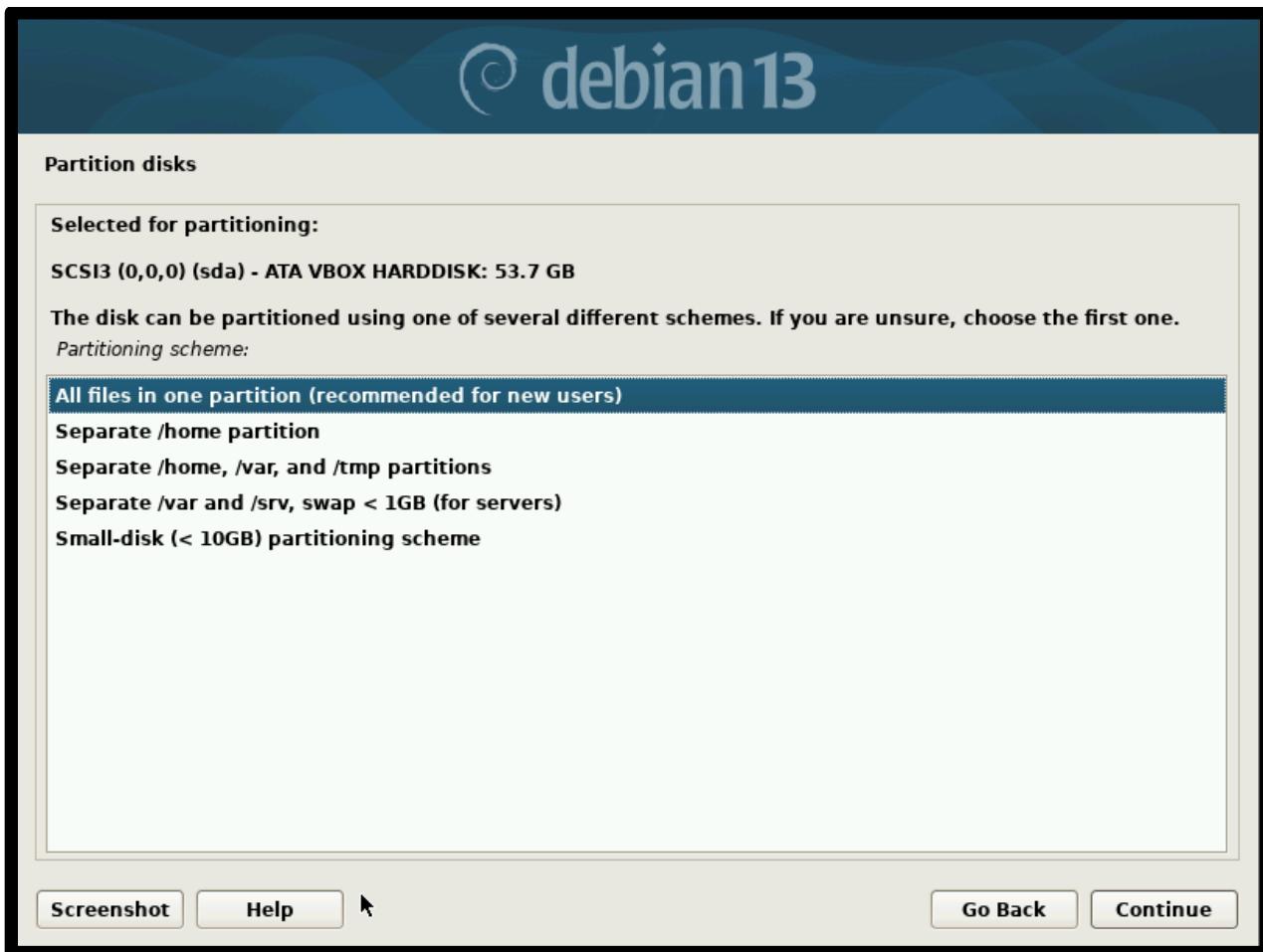
11º Passo – Neste passo vamos configurar as partições dos nossos discos o normalmente é selecionado é sempre a 1º opção que é a que vem por defeito.



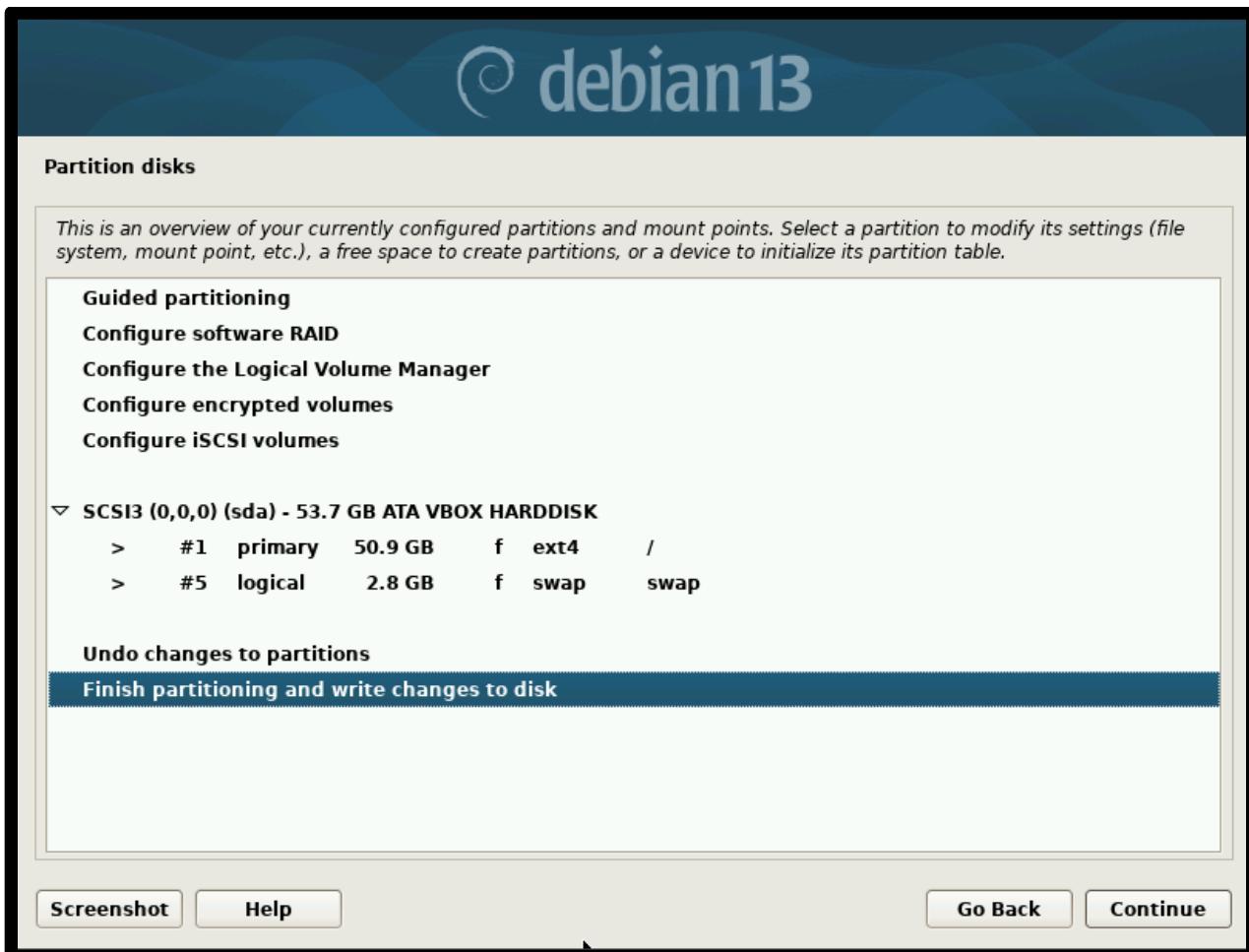
Aqui selecionamos o disco que tivermos.



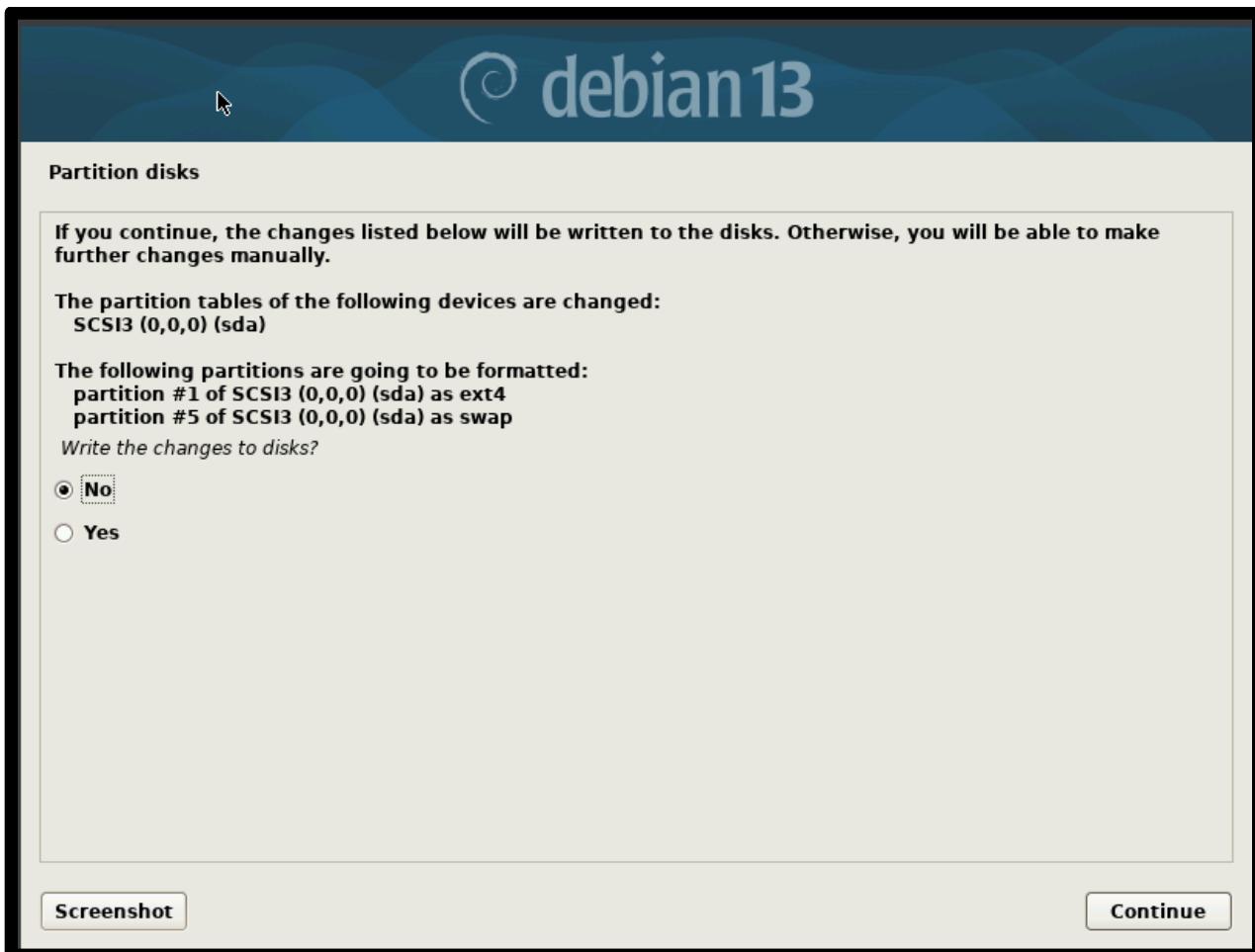
Aqui normalmente é usado a 1º opção mas cada um escolhe aquilo que pretende.



Clicamos em continue pois não queremos voltar atrás só se tivermos feito algum passo de errado.



Mudamos para yes e depois clicamos em continue pois não queremos fazer alterações.



12º Passo – Clicamos em continue, pois isto não é necessário



Depende do país em que estivermos



Aqui deixamos estar o que está por defeito.



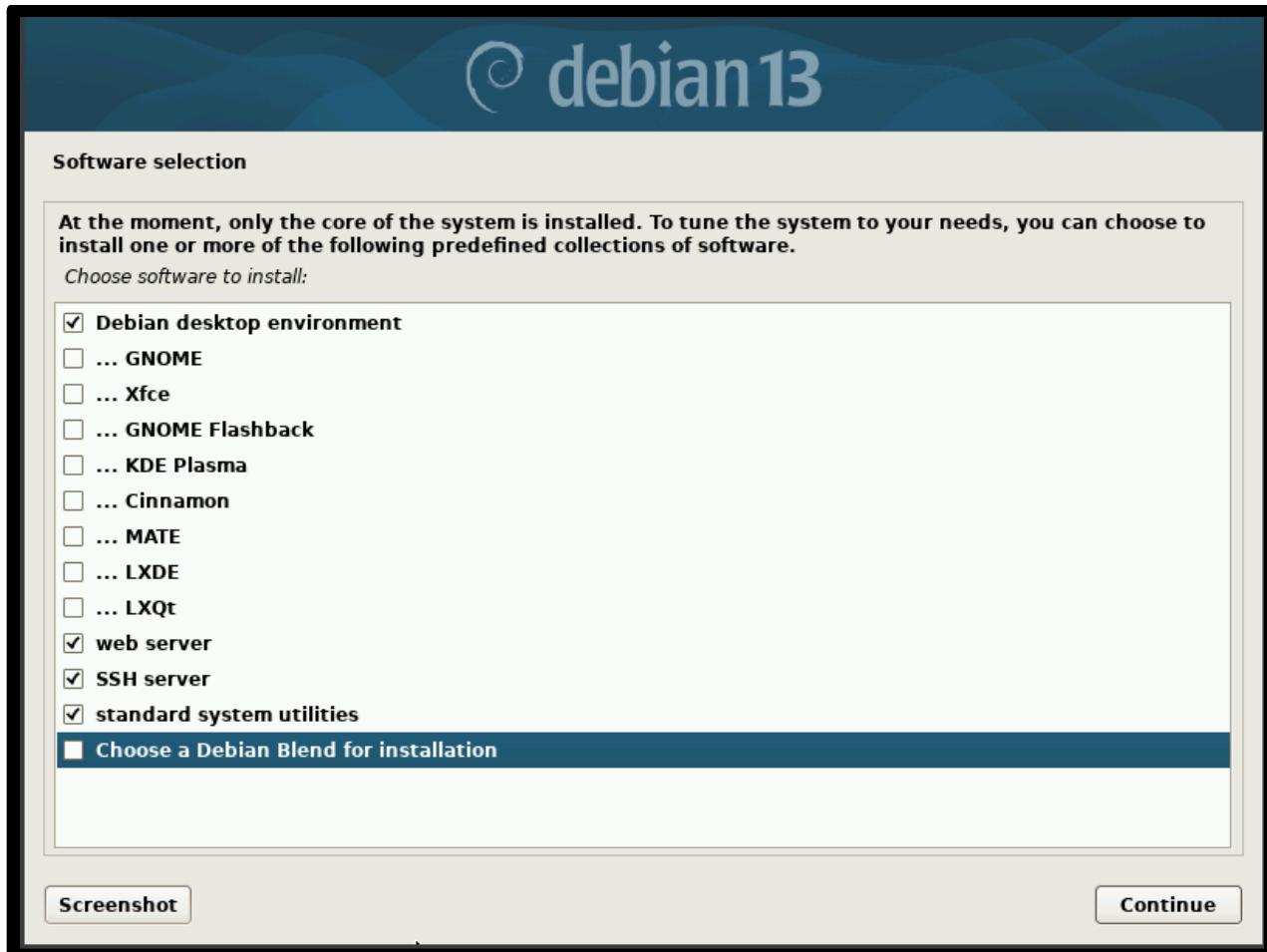
Aqui é caso queiramos configurar proxy, não é necessário, mas cada um configura como quer.



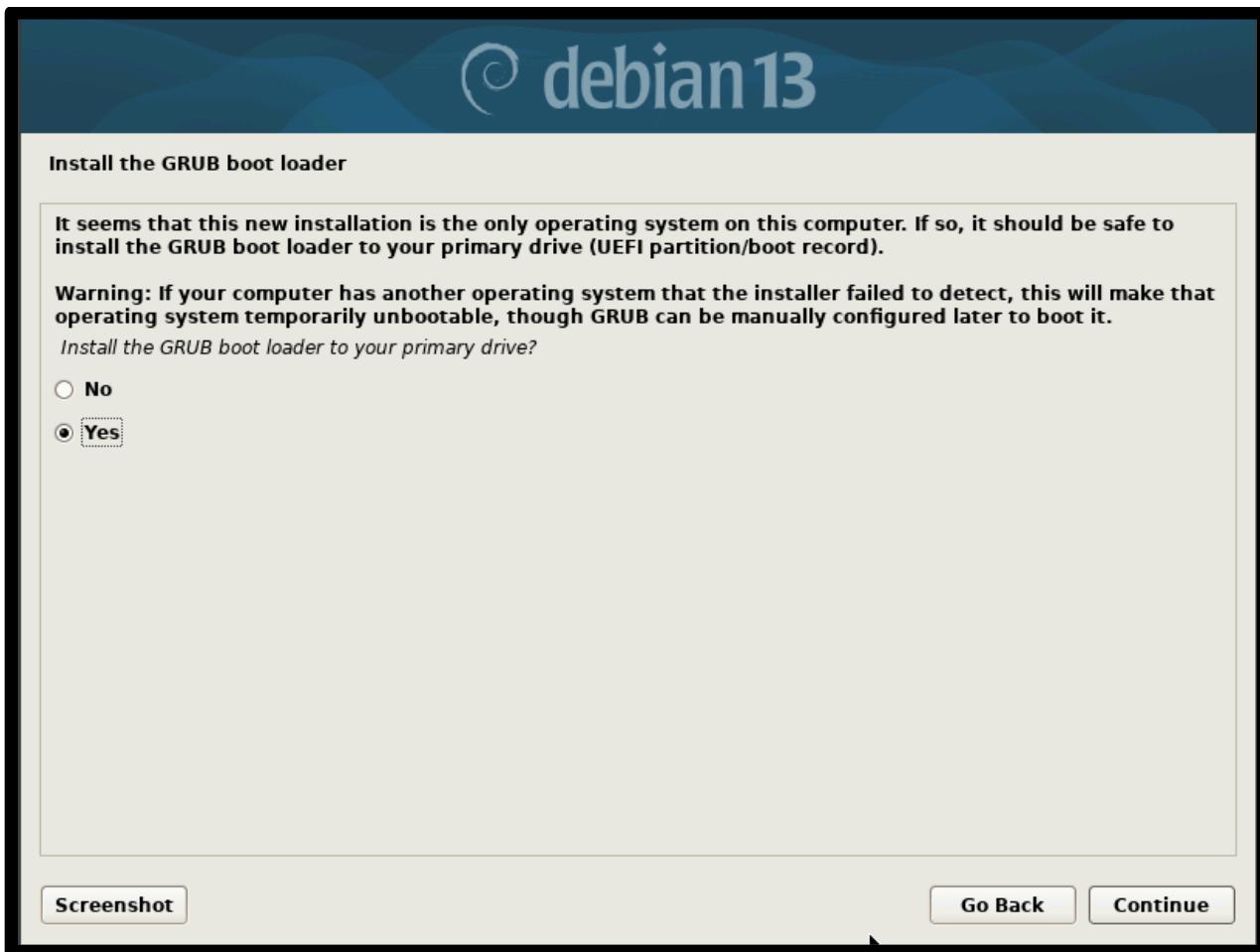
13º Passo – Aqui deixamos a opção no, pois é para participar em um survey.



15º Este é o passo mais importante, pois nós queremos uma máquina com GUI (Graphical User Interface), ou seja, com ambiente gráfico, nós não queremos por CLI (Command Line), ou seja, é uma tela preta e só podemos executar comandos. Selecionamos como está na imagem abaixo.



16º Passo – Aqui deixamos o yes.



Aqui é algo muito importante temos de ter em atenção porque devemos sempre escolher a 2ºopção, porque se escolhermos a 1º vamos ficar numa tela preta infinitamente.

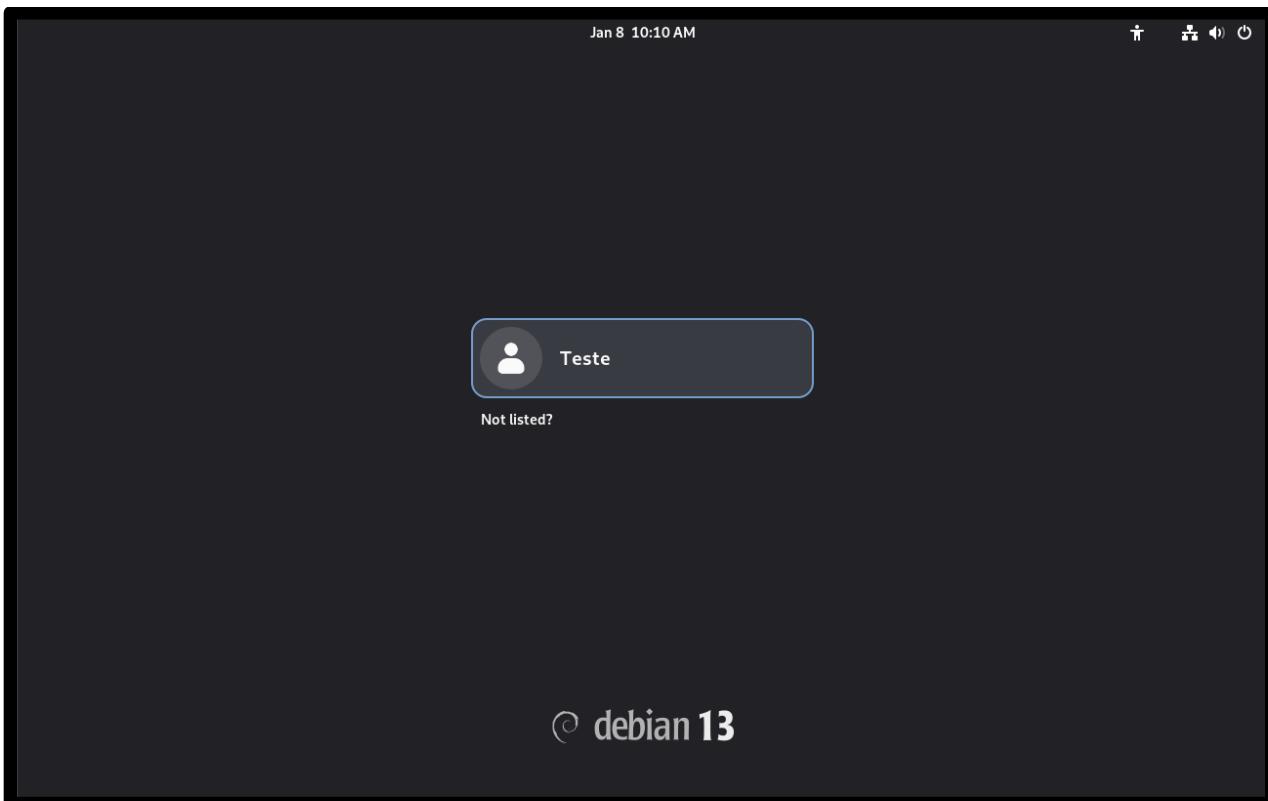


Clicamos em continue

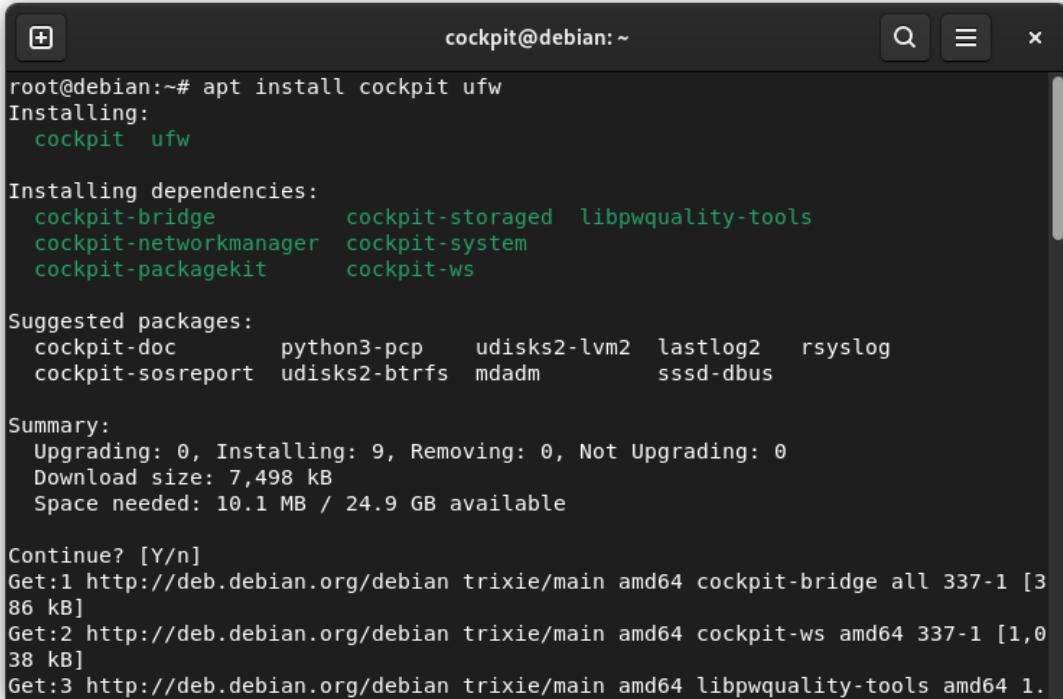


Fazendo estes passos todos temos a nossa máquina pronta a usar.

Já dentro da máquina.



Com o usuário **root** vamos instalar os pacotes **cockpit** e **ufw**.



```
cockpit@debian:~# apt install cockpit ufw
Installing:
  cockpit  ufw

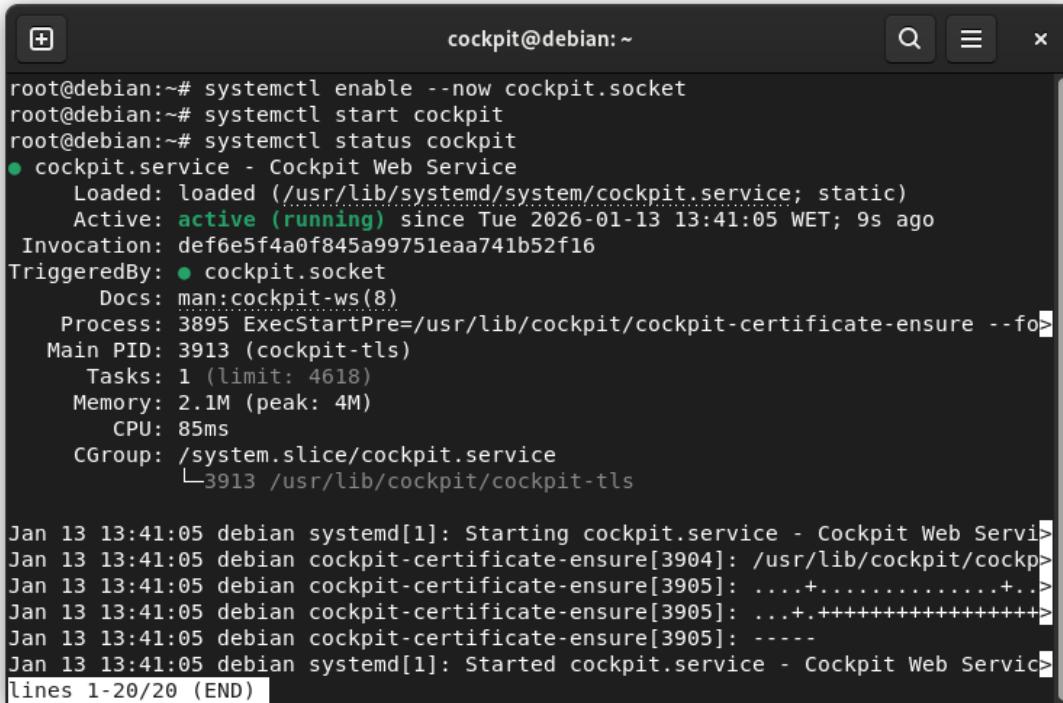
Installing dependencies:
  cockpit-bridge    cockpit-storaged  libpwquality-tools
  cockpit-networkmanager  cockpit-system
  cockpit-packagekit   cockpit-ws

Suggested packages:
  cockpit-doc      python3-pcp      udisks2-lvm2  lastlog2  rsyslog
  cockpit-sosreport  udisks2-btrfs  mdadm        sssd-dbus

Summary:
  Upgrading: 0, Installing: 9, Removing: 0, Not Upgrading: 0
  Download size: 7,498 kB
  Space needed: 10.1 MB / 24.9 GB available

Continue? [Y/n]
Get:1 http://deb.debian.org/debian trixie/main amd64 cockpit-bridge all 337-1 [386 kB]
Get:2 http://deb.debian.org/debian trixie/main amd64 cockpit-ws amd64 337-1 [1,038 kB]
Get:3 http://deb.debian.org/debian trixie/main amd64 libpwquality-tools amd64 1.
```

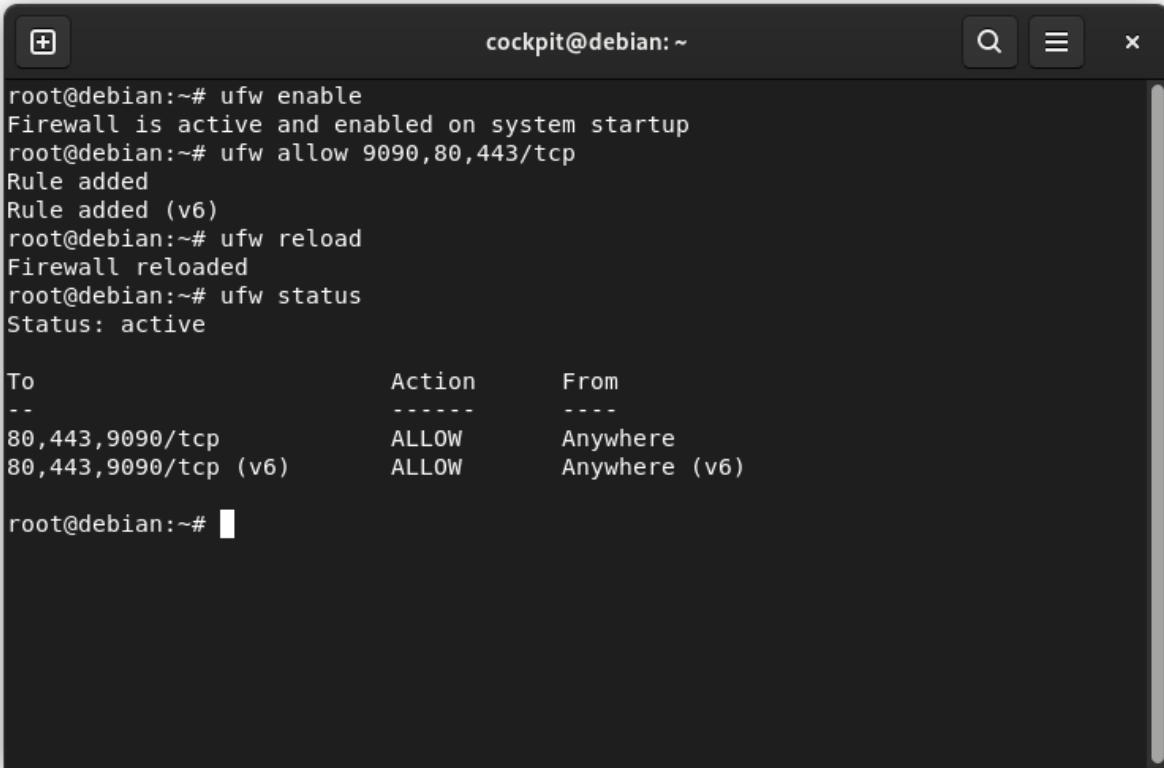
Vamos agora ativar o serviço **cockpit**.



```
cockpit@debian:~# systemctl enable --now cockpit.socket
cockpit@debian:~# systemctl start cockpit
cockpit@debian:~# systemctl status cockpit
● cockpit.service - Cockpit Web Service
  Loaded: loaded (/usr/lib/systemd/system/cockpit.service; static)
  Active: active (running) since Tue 2026-01-13 13:41:05 WET; 9s ago
    Invocation: def6e5f4a0f845a99751eaa741b52f16
  TriggeredBy: ● cockpit.socket
    Docs: man:cockpit-ws(8)
  Process: 3895 ExecStartPre=/usr/lib/cockpit/cockpit-certificate-ensure --fo>
  Main PID: 3913 (cockpit-tls)
    Tasks: 1 (limit: 4618)
   Memory: 2.1M (peak: 4M)
     CPU: 85ms
    CGroup: /system.slice/cockpit.service
              └─3913 /usr/lib/cockpit/cockpit-tls

Jan 13 13:41:05 debian systemd[1]: Starting cockpit.service - Cockpit Web Service
Jan 13 13:41:05 debian cockpit-certificate-ensure[3904]: /usr/lib/cockpit/cockp>
Jan 13 13:41:05 debian cockpit-certificate-ensure[3905]: ....+.....+...>
Jan 13 13:41:05 debian cockpit-certificate-ensure[3905]: ....+.+++++++=+>
Jan 13 13:41:05 debian cockpit-certificate-ensure[3905]: -----
Jan 13 13:41:05 debian systemd[1]: Started cockpit.service - Cockpit Web Service
lines 1-20/20 (END)
```

Agora vamos ativar a **firewall ufw** e alterar as suas regras para permitir as portas 9090 (utilizada pelo cockpit), 80 (HTTP) e 443 (HTTPS).



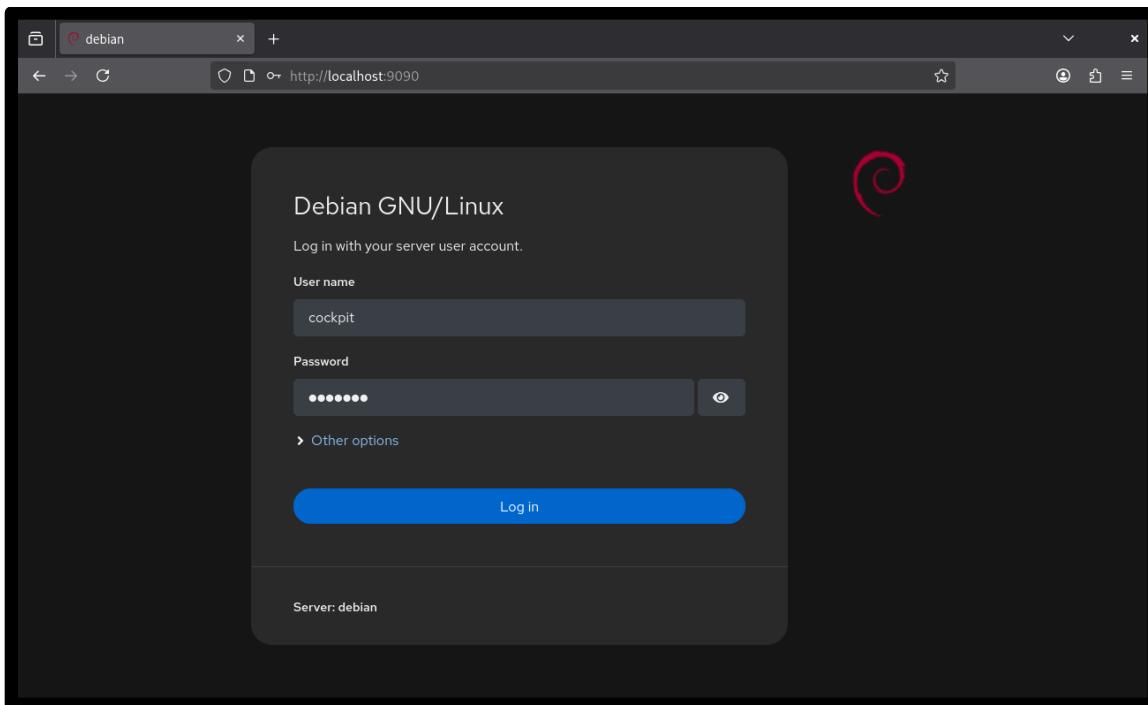
The screenshot shows a terminal window titled "cockpit@debian: ~". The terminal displays the following commands and output:

```
root@debian:~# ufw enable
Firewall is active and enabled on system startup
root@debian:~# ufw allow 9090,80,443/tcp
Rule added
Rule added (v6)
root@debian:~# ufw reload
Firewall reloaded
root@debian:~# ufw status
Status: active

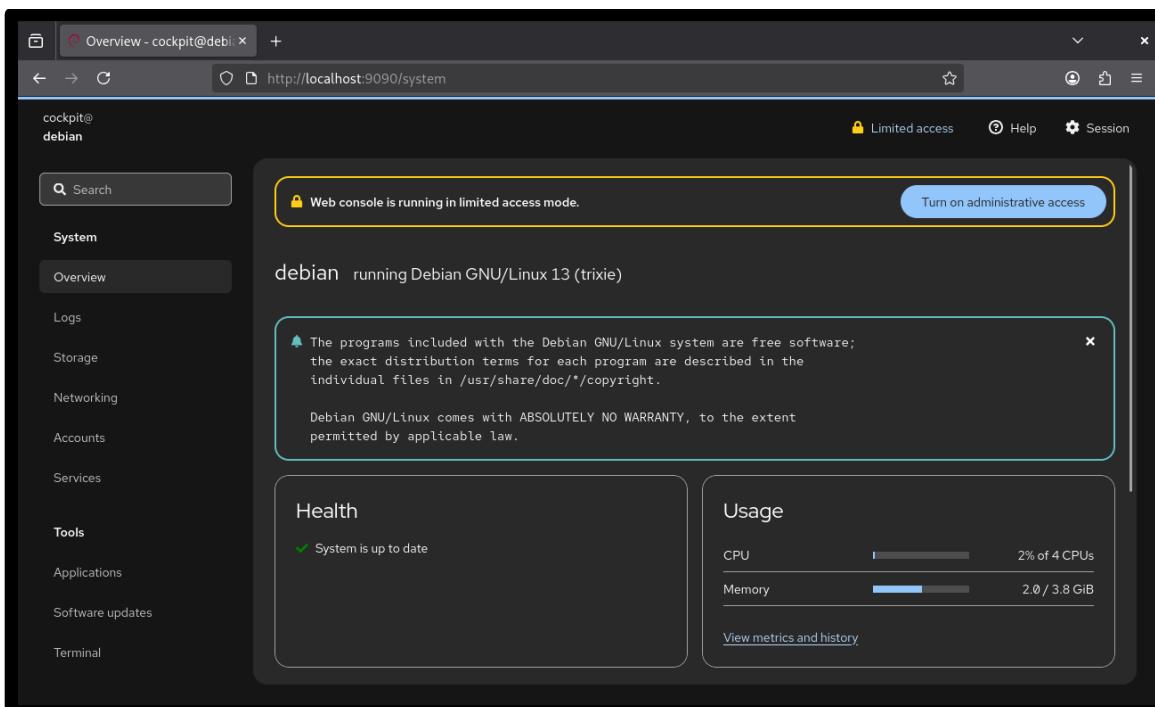
To           Action      From
--           ----       ---
80,443,9090/tcp    ALLOW      Anywhere
80,443,9090/tcp (v6) ALLOW      Anywhere (v6)

root@debian:~#
```

Agora em um navegador (Firefox), na barra de pesquisa introduzir **localhost:9090** ou **<IP da máquina>:9090**.



Deverá aparecer um ecrã assim:



## Apresentação da ferramenta

O **Cockpit Project** é um software que facilita a administração de servidores em Linux através de um **browser**. Quando iniciamos um serviço através do Cockpit, podemos pará-lo através do terminal. Além disso, se ocorrer algum erro no terminal, o erro é exibido no separador **logs** do Cockpit.

O Cockpit é um projeto interessante para administração Linux baseada na web que recebeu relativamente pouca atenção ao longo dos anos. Parte disso pode por causa da estratégia do projeto de lançamentos menores a cada duas semanas, em vez de lançamentos maiores com muitos recursos novos. Embora a estratégia tenha contribuído pouco para ganhar destaque, proporcionou uma ferramenta útil e extensível para observar, **gerir** e solucionar problemas em servidores Linux.

O projeto Cockpit começou em 2013, patrocinado pela Red Hat. Era voltado principalmente para administradores de sistemas novos em Linux, como administradores de Windows que precisavam de se adaptar à crescente presença do Linux no mercado de servidores. Foi concebido para ser independente de distribuição e está disponível e testado em muitas distribuições populares de Linux, incluindo Arch Linux, Debian, Fedora, openSUSE Tumbleweed e Red Hat Enterprise Linux (RHEL) e os seus derivados.

Um dos ideais do projeto é que os utilizadores “sintam que estão a interagir com o sistema operativo do servidor subjacente” e que a sua interface de utilizador mostre o nome do servidor ou do sistema operativo, em vez da marca Cockpit. Na verdade, os utilizadores que estão a interagir com o sistema operativo subjacente quando utilizam o Cockpit: utilizam ferramentas de sistema padrão em segundo plano para as suas tarefas, caso estejam disponíveis.

Embora o Cockpit seja independente da distribuição, a sua herança transparece um pouco nas escolhas das ferramentas padrão. O Cockpit utiliza o **NetworkManager** para gerir configurações de rede, o **firewalld** para gerir a **firewall** do sistema, o **storaged** para trabalhar com o armazenamento do sistema, o **SOS report** para criar relatórios de diagnóstico e assim por diante, se estes estiverem presentes no sistema.

Se uma distribuição utilizar o **networkd** em vez de utilizar o NetworkManager como padrão, as opções passam por instalar o NetworkManager, deixar de utilizar o Cockpit para a administração de rede ou realizar o trabalho necessário e contribuir com essa funcionalidade para o projeto Cockpit. Na maioria das vezes, porém, poucos utilizadores contribuem com funcionalidades para o projeto — a maioria das contribuições para o Cockpit provém de um punhado de funcionários da Red Hat.

Se formos a analisar a atividade dos dois últimos anos no repositório principal do Cockpit, a maioria dos *commits* (é o registo permanente de um conjunto de alterações num projeto de software) pertence a Martin Pitt, Allison Karlitskaya, Jelle van der Waa, Marius Vollmer e Katerina Koukiou, todos funcionários da Red Hat. Isto não quer dizer que as contribuições não sejam bem-vindas. O Cockpit possui um guia detalhado de contribuições e um guia do programador para aqueles que estejam interessados em contribuir ou reutilizar os componentes do Cockpit.

## Arquitetura e funcionamento

O Cockpit é licenciado pela LGPL v2.1 (ou posterior) e consiste no *frontend* web, juntamente com um serviço web para comunicação entre o *frontend* e o **cockpit-bridge**. O programa *cockpit-bridge* é responsável por retransmitir comandos via DBus para o systemd, storaged, NetworkManager e outras ferramentas do sistema utilizadas pelo Cockpit. Os seus componentes web são escritos em JavaScript, estando os outros componentes escritos principalmente em C ou Python. O componente *cockpit-bridge* tem implementações em C e Python. A ponte Python foi introduzida em junho de 2023, na versão Cockpit 294. As distribuições recentes baseadas em RHEL e Fedora mudaram para a implementação Python, enquanto o Debian 12 e o Ubuntu 22.04 ainda mantêm a versão C do *cockpit-bridge*.

Depois de o Cockpit estar instalado e em execução num sistema, a sua interface web estará disponível na porta 9090 por predefinição. A porta pode ser alterada, se necessário, ou o serviço web pode ser desativado completamente; os clientes do Cockpit podem **ligar-se** via SSH usando um cliente de desktop, um contentor que execute o serviço web do Cockpit ou fazendo *login* noutro servidor que execute o Cockpit e, em seguida, **ligando-se** a um servidor secundário.

Podem esperar-se problemas ao misturar e combinar versões do Cockpit mas, como este é usado em tantas distribuições Linux com calendários de lançamento variados, a compatibilidade com versões anteriores é outro dos ideais do projeto. Se um utilizador estiver a executar o Cockpit no Debian Bookworm, deverá ainda conseguir **ligar-se** a um servidor que execute o Fedora Rawhide sem problemas, ou vice-versa. Tendo usado o Cockpit durante anos em vários sistemas que executam Alma Linux, CentOS, Debian, Fedora, RHEL e Ubuntu, posso confirmar que o projeto faz um excelente trabalho na manutenção da compatibilidade entre as versões.

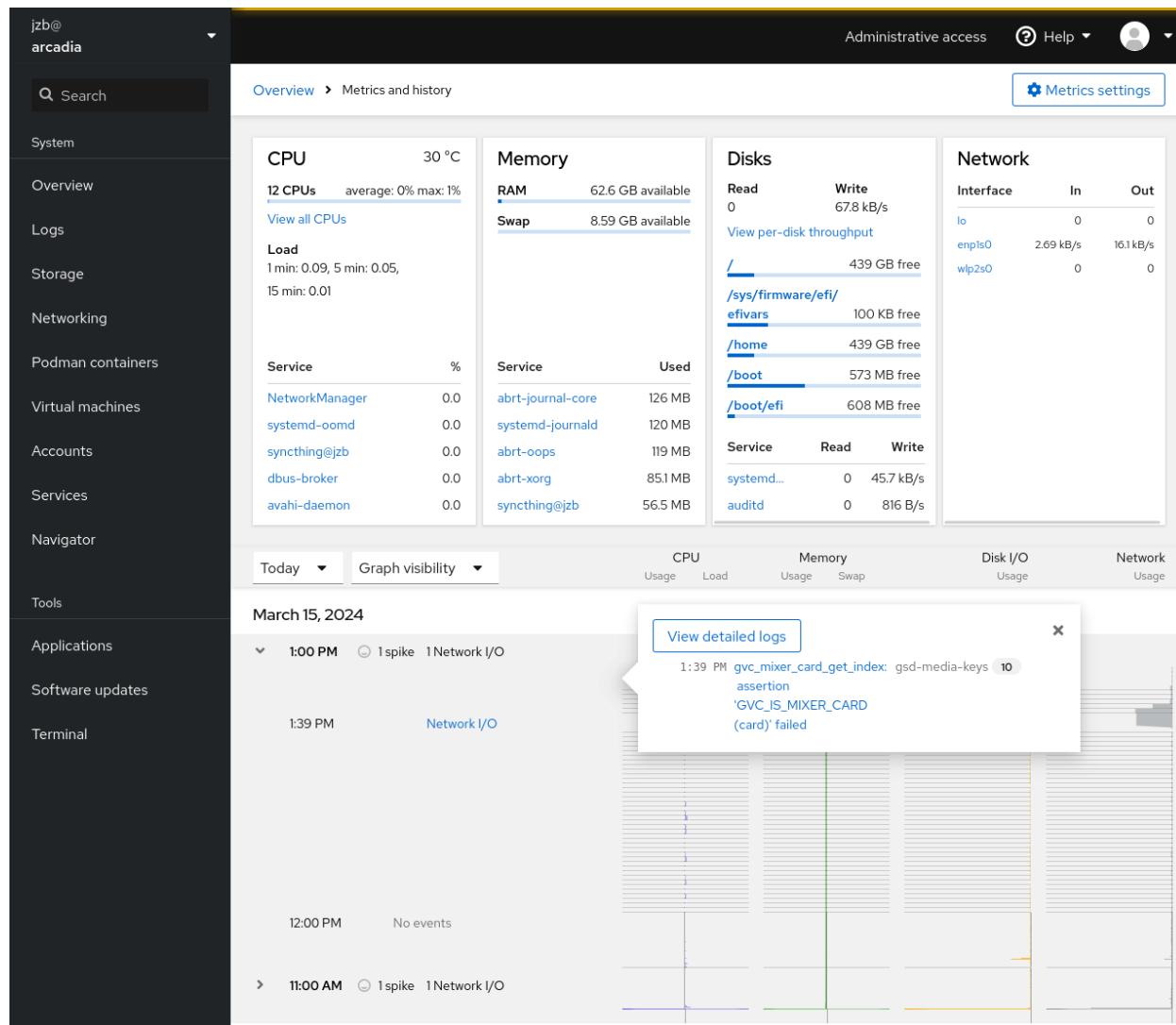
É recomendável o uso do Firefox ou do Chrome e possui testes automatizados para garantir a compatibilidade com ambos os navegadores. O projeto também afirma que testa “periodicamente” o Edge, o Safari e o navegador web Epiphany do GNOME. Os utilizadores que tentam **ligar-se** ao Cockpit com o Safari no iOS, iPadOS e Macs baseados em ARM podem ter alguma dificuldade. O Safari nessas plataformas não oferece suporte aos certificados que o Cockpit utiliza por predefinição. Os utilizadores podem tentar contornar esta situação ou simplesmente utilizar um navegador compatível.

O projeto tem uma lista de correio para desenvolvimento, mas não é uma lista muito ativa. Duas mensagens foram enviadas em 2024, uma em janeiro e outra em fevereiro. A página de discussões do GitHub é mais ativa, e a maior parte do trabalho acontece no rastreador de problemas (*issue tracker*) do Cockpit no GitHub.

O ritmo acelerado dos lançamentos significa que a versão *upstream* do Cockpit pode ser significativamente mais recente do que a versão empacotada para as versões estáveis ou de suporte a longo prazo (LTS) do Debian e do Ubuntu. O projeto recomenda a instalação a partir dos repositórios *backports* do Debian ou do Ubuntu para utilizar uma versão mais atualizada, se possível.

O Cockpit pode ser configurado para utilizar a **palavra-passe** que vem por defeito do sistema ou a autenticação Kerberos ao fazer *login* diretamente. Pode utilizar a autenticação baseada em chave SSH ao **ligar-se** via SSH a servidores secundários. Ao contrário de ferramentas como o Webmin, o Cockpit não mantém um conjunto de utilizadores independente do sistema; os utilizadores fazem *login* como eles próprios e têm as mesmas permissões e privilégios como se tivessem **efetuado a ligação** via SSH ou feito *login* na consola. Os utilizadores também podem obter acesso administrativo se tiverem privilégios de **sudo** no *host*.

Após o *login*, os utilizadores verão uma página de visão geral do sistema com um painel que exibe a integridade do sistema, o uso da CPU e da memória, informações básicas do sistema e navegação para páginas para trabalhar com contas de utilizador, registos do sistema, serviços do sistema, rede e outras ferramentas. Os serviços exatos disponíveis através do Cockpit variam de acordo com os serviços disponíveis no sistema operativo do *host* e quais os componentes opcionais que estão instalados. Isto simplifica o *login* e permite ver rapidamente se um sistema tem atualizações de software pendentes, se registou algum erro preocupante e muito mais.



Ao aprofundar-se na página de métricas e histórico, o Cockpit oferece um conjunto mais detalhado de métricas sobre a carga do sistema, quais as aplicações que estão a consumir mais memória, o uso e a taxa de transferência do disco, as interfaces de rede e o consumo de largura de banda. Note-se que as informações de utilização de memória do Cockpit provêm de grupos de controlo (**cgroups**) para fornecer uma visão geral rápida do uso do sistema, em vez de uma contabilidade completa do conjunto residente e da memória virtual por processo.

Para dados históricos, o Cockpit utiliza o **Performance Co-Pilot (PCP)**, se este estiver disponível. Se não estiver disponível, o Cockpit oferece uma opção para instalar o PCP para começar a recolher métricas e, em seguida, exibir o uso e a carga do sistema ao longo do tempo. O Cockpit consegue inclusivamente observar picos

na carga ou E/S (*I/O*), apresentando ligações para os registo (*logs*) desse período que podem ajudar a esclarecer a causa.

O separador **Serviços** permite que um administrador navegue pelos serviços, destinos, temporizadores, **sockets** ou caminhos disponíveis no **systemd**. A partir daí, os administradores podem gerir os vários serviços do systemd, ver as suas relações com outros serviços e visualizar os seus registo. Se for relevante para um serviço, o Cockpit também mostrará a sua utilização de memória. Por exemplo, ao navegar pelo apache2.service no meu servidor Debian, que executa alguns blogs WordPress, é mostrado que 628 MB de memória estão em uso.

Como seria de esperar, o separador **Contas** do Cockpit permite aos administradores visualizar os utilizadores no sistema e trabalhar com utilizadores e grupos. Além das operações básicas, como adicionar ou remover utilizadores, alterar **palavras-passe** e gerir grupos de utilizadores, os administradores também podem navegar pelo histórico de *login* dos utilizadores e gerir chaves SSH autorizadas.

## Requisitos técnicos

- Sistema operativo Linux (recomendados: Fedora, Fedora CoreOS, Red Hat Enterprise Linux, CentOS, Debian, archlinux, ubuntu, Tumbleweed ou SUSE Linux Enterprise Micro)
- Um navegador web (recomendados: Chrome ou Firefox)
- Espaço em disco: Apenas o suficiente para o SO, o Cockpit e os logs do sistema
- Memória RAM: Como o Cockpit é bastante otimizado e requer pouco do sistema, o mínimo de RAM recomendado para o SO é o suficiente

## Casos de uso

### Iniciantes em Linux / utilizadores pouco avançados

O Cockpit oferece uma interface gráfica intuitiva e de fácil utilização, sendo especialmente recomendado para utilizadores que ainda não estão familiarizados com o uso do terminal. Permite realizar tarefas comuns de administração como, monitorização de recursos, gestão de serviços e utilizadores, sem necessidade de recorrer a comandos complexos.

### Administradores experientes

Mesmo para administradores com vasta experiência em linha de comandos e que utilizam ferramentas mais avançadas, o Cockpit continua a ser útil pois oferece uma visão geral bem organizada do servidor e dos sistemas a ele ligados, facilitando a monitorização rápida do estado geral da infraestrutura.

Devido à utilização de ferramentas e APIs já existentes no sistema na sua construção, o Cockpit tem uma grande compatibilidade com outras ferramentas de gestão de servidores.

### Mercado de trabalho

O Cockpit é bastante utilizado no mercado de trabalho na área de administração de sistemas e servidores devido a reduzir bastante o trabalho manual para tarefas simples com a sua GUI de fácil uso.

A integração com praticamente qualquer ferramenta de administração de servidores é também um grande incentivo ao seu uso pois não requer a instalação de serviços externos.

## Vantagens

- Interface gráfica simples, moderna e de fácil compreensão, mesmo para utilizadores inexperientes
- Visão geral completa do sistema (CPU, memória, armazenamento, rede e serviços)
- Boa compatibilidade com outras ferramentas de administração e gestão de servidores
- Consumo reduzido de recursos
- Não corre permanentemente em segundo plano, estando apenas ativo quando utilizado

## Limitações

- Devido às ferramentas mais básicas que apresenta o Cockpit não consegue realizar configurações mais profundas, requerendo o uso de ferramentas mais avançadas na CLI para estas tarefas
- Grandes empresas priorizam mais a automação de ferramentas mais complicadas que a simplicidade da GUI do Cockpit
- Não é recomendado para a aprendizagem pois limita o utilizador a recursos mais básicos

## Comparação com alternativas

### Cockpit vs Webmin

#### Cockpit

- Utiliza APIs e comandos já existentes no sistema, garantindo maior compatibilidade com outras ferramentas
- Software leve, com uma GUI moderna e segura por defeito
- Ideal para uma visão geral rápida do sistema e para tarefas administrativas básicas
- Menos adequado para configurações muito profundas ou altamente específicas

#### Webmin

- Utiliza APIs e comandos próprios, podendo gerar configurações não standard, o que reduz a compatibilidade com outras ferramentas
- Possui módulos maus completos e poderosos, beneficiando do facto de existir há mais tempo e ter uma comunidade maior e mais ativa
- Pode representar maiores riscos de segurança, devido ao elevado número de serviços ativos simultaneamente

## Laboratórios propostos

### Setup inicial do Cockpit

- Instalação do Cockpit num sistema Linux (Debian)
- Configuração inicial do serviço
- Configuração da firewall para permitir acesso à interface web
- Acesso e navegação básica pela interface

### Gestão e monitorização do sistema

- Monitorização de CPU, memória, discos e rede
- Gestão de serviços (start, stop, restart)
- Visualização de logs do sistema
- Criação e gestão de utilizadores e grupos

### Instrumentos de avaliação

- Quiz em Kahoot! (Multipla escolha)
- Perguntas de resposta aberta (Incluídas na Apresentação)

## Conclusão

O Cockpit é uma ferramenta de administração de servidores moderna, leve e eficaz, que se adapta às necessidades tanto de utilizadores iniciantes como a administradores experientes. A sua interface gráfica intuitiva simplifica tarefas comuns de gestão e manutenção, enquanto a sua integração com ferramentas nativas do sistema garante compatibilidade e segurança.

Embora não substitua completamente soluções mais avançadas como o Webmin em cenários de configuração profunda, o Cockpit destaca-se como uma excelente opção para monitorização, gestão básica e provem uma visão geral clara do estado dos sistemas.

## Referências bibliográficas

- Documentação oficial Cockpit Project  
(<https://cockpit-project.org/documentation.html>)
- Blog no site da Red Hat sobre o Cockpit Project  
(<https://www.redhat.com/en/blog/intro-cockpit>)
- Artigo na Wiki Ikoula sobre o Cockpit Project  
([https://pt-wiki.ikoula.com/pt/Descubra\\_o\\_projeto\\_WebUI\\_Cockpit](https://pt-wiki.ikoula.com/pt/Descubra_o_projeto_WebUI_Cockpit))
- Artigo no LWN.NET sobre o Cockpit Project  
(<https://lwn.net/Articles/965434/>)