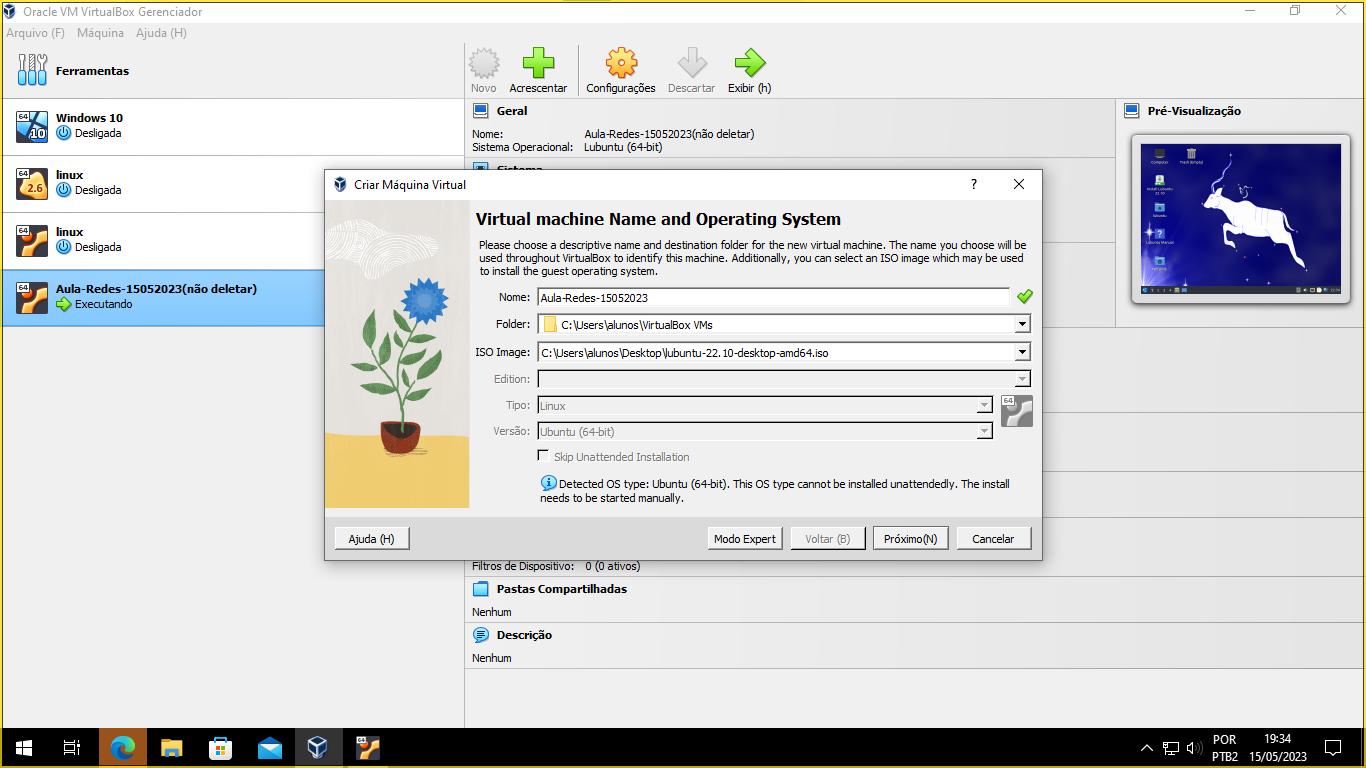
**João Henrique Rodrigues Batista – RGM 11231103959  
Pedro Henrique Santos de Andrade – RGM 11231103572**

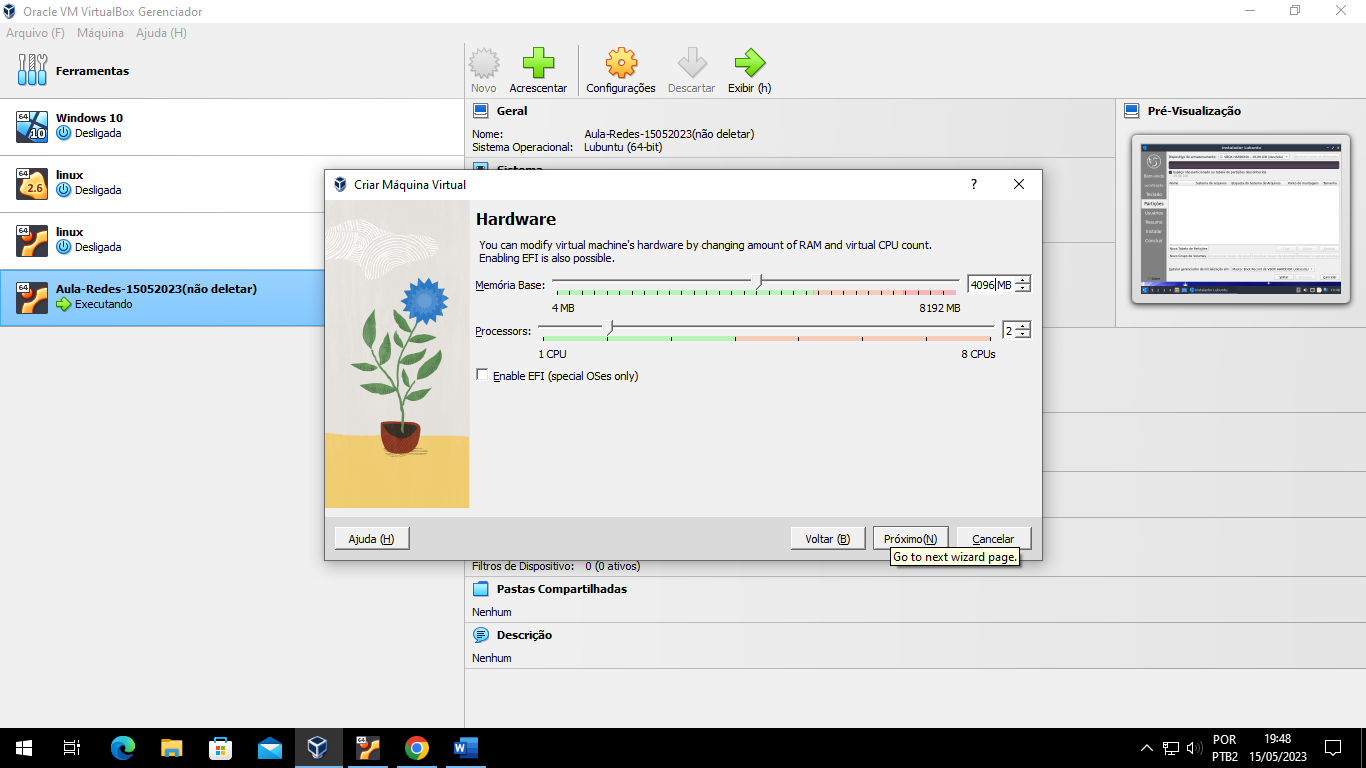
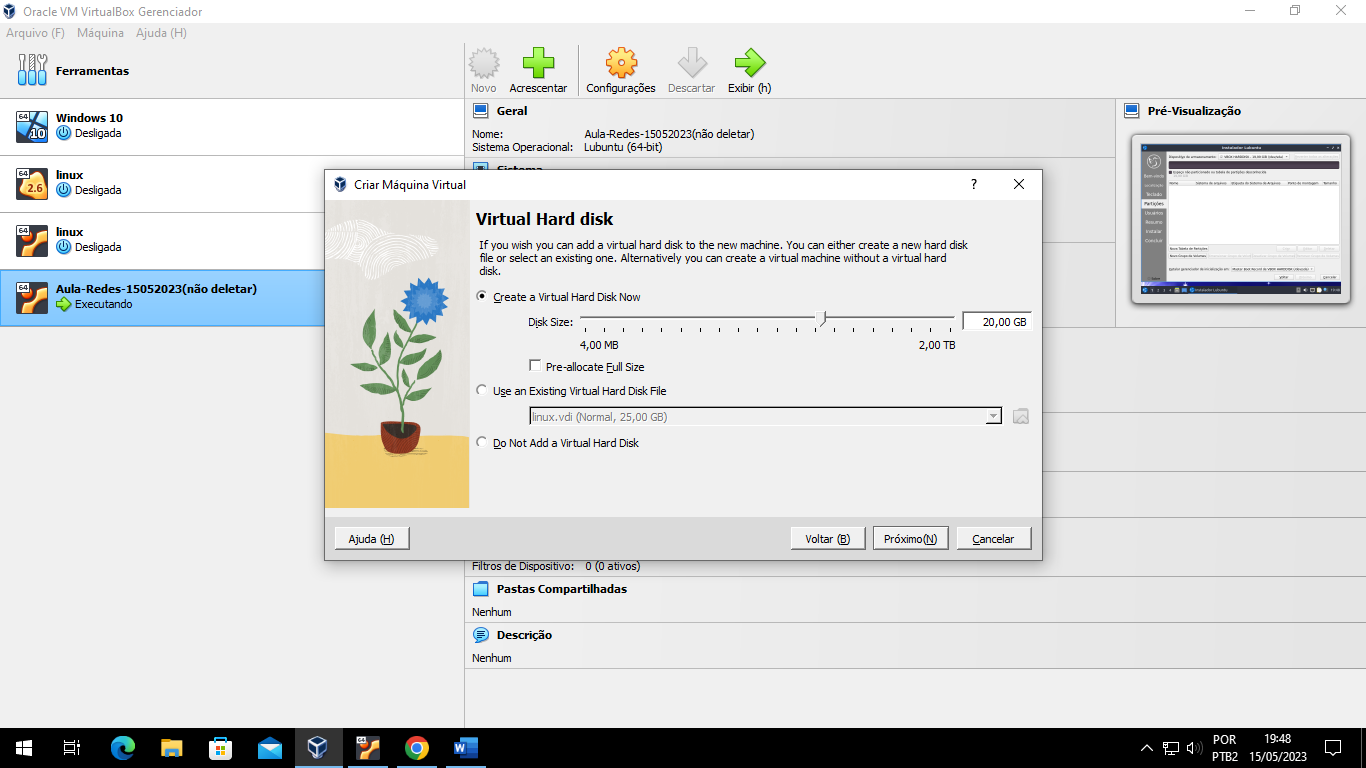
**Atividade de Instalação e manipulação de VM.**

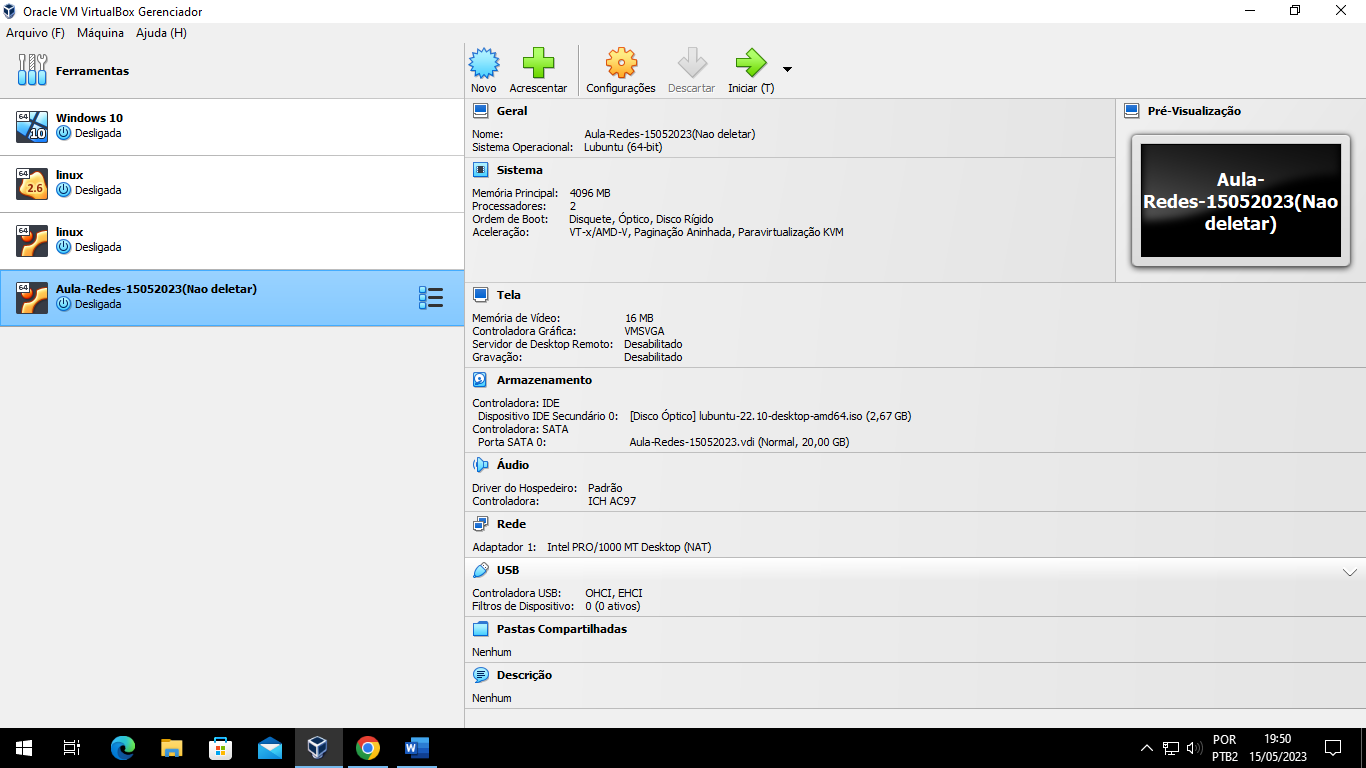
**1º Passo:**

Preencha os campos abaixo e clique em próximo.



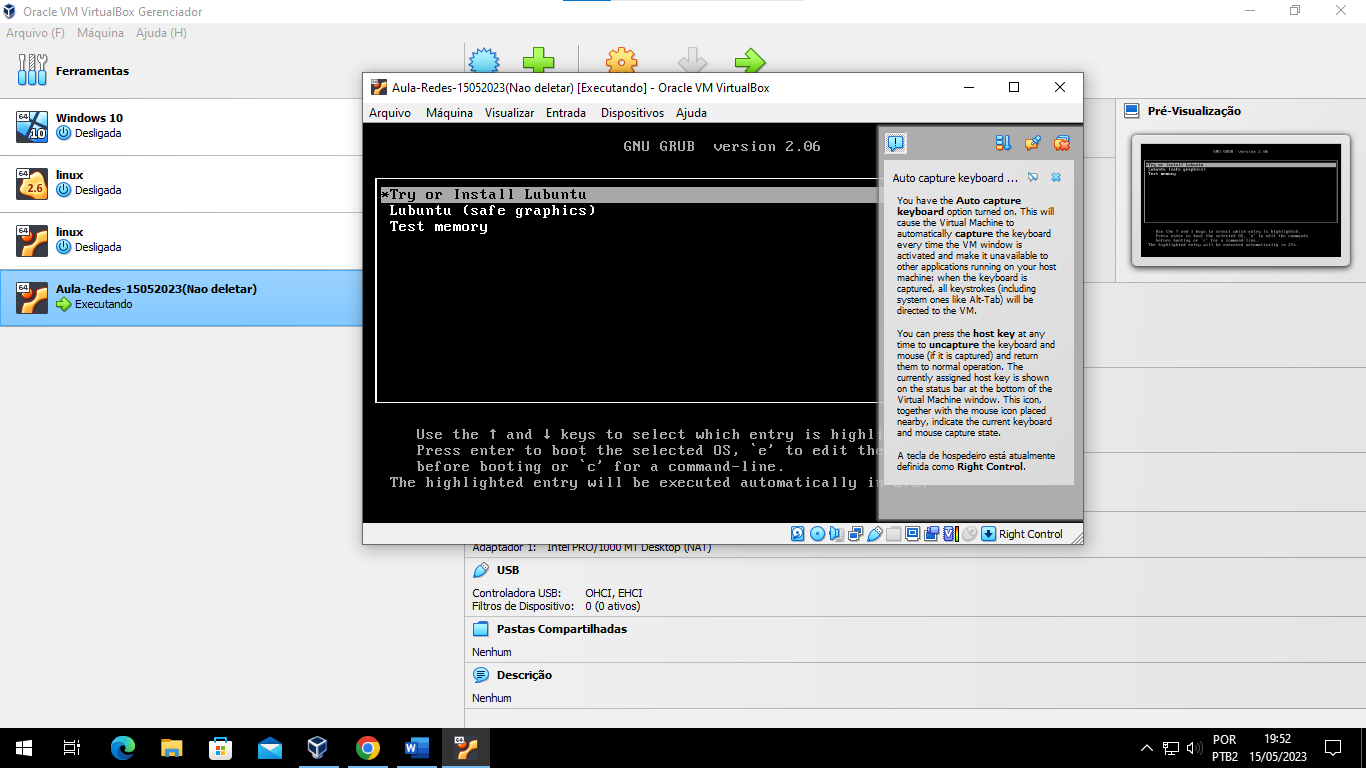
Após esse passo, coloque os requisitos de sua VM.

Máquina virtual criada.  


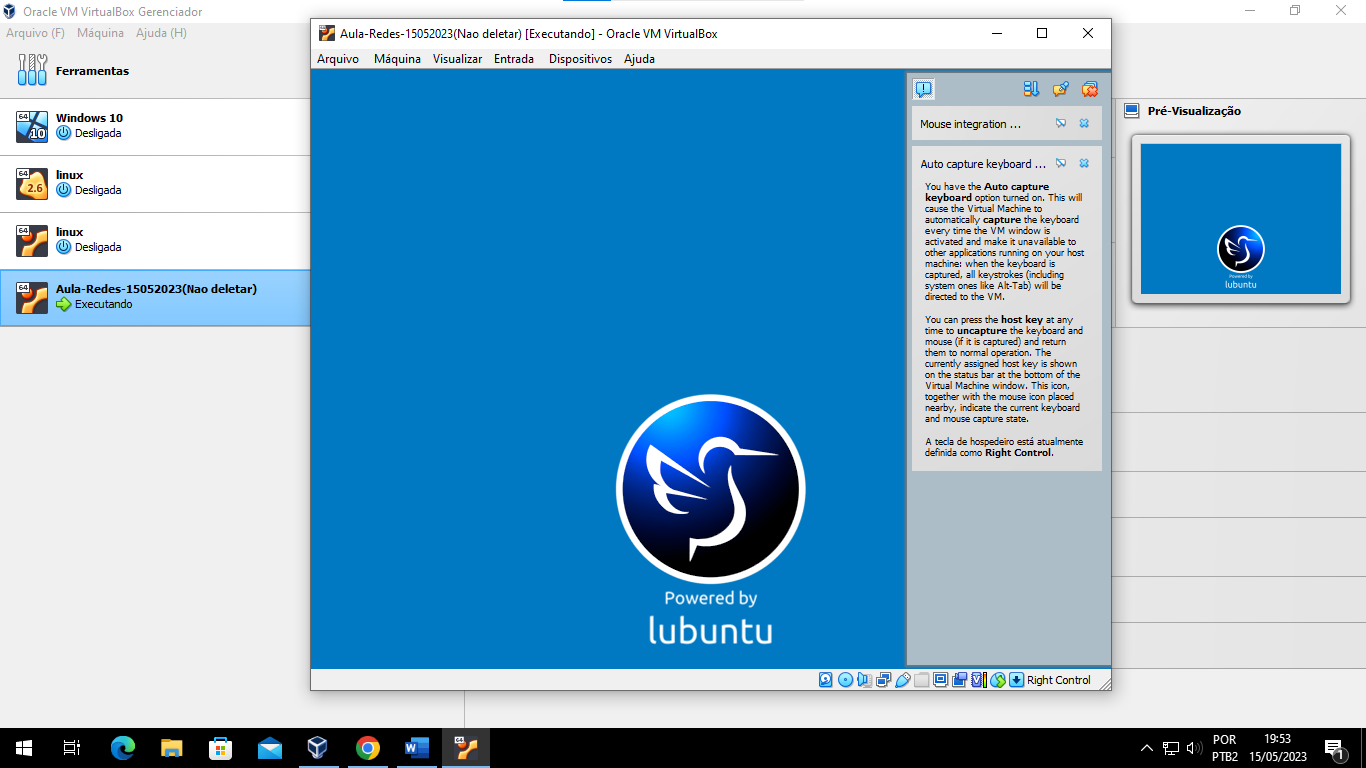
**2º Passo:**

Executando a máquina e instalando suas dependências.

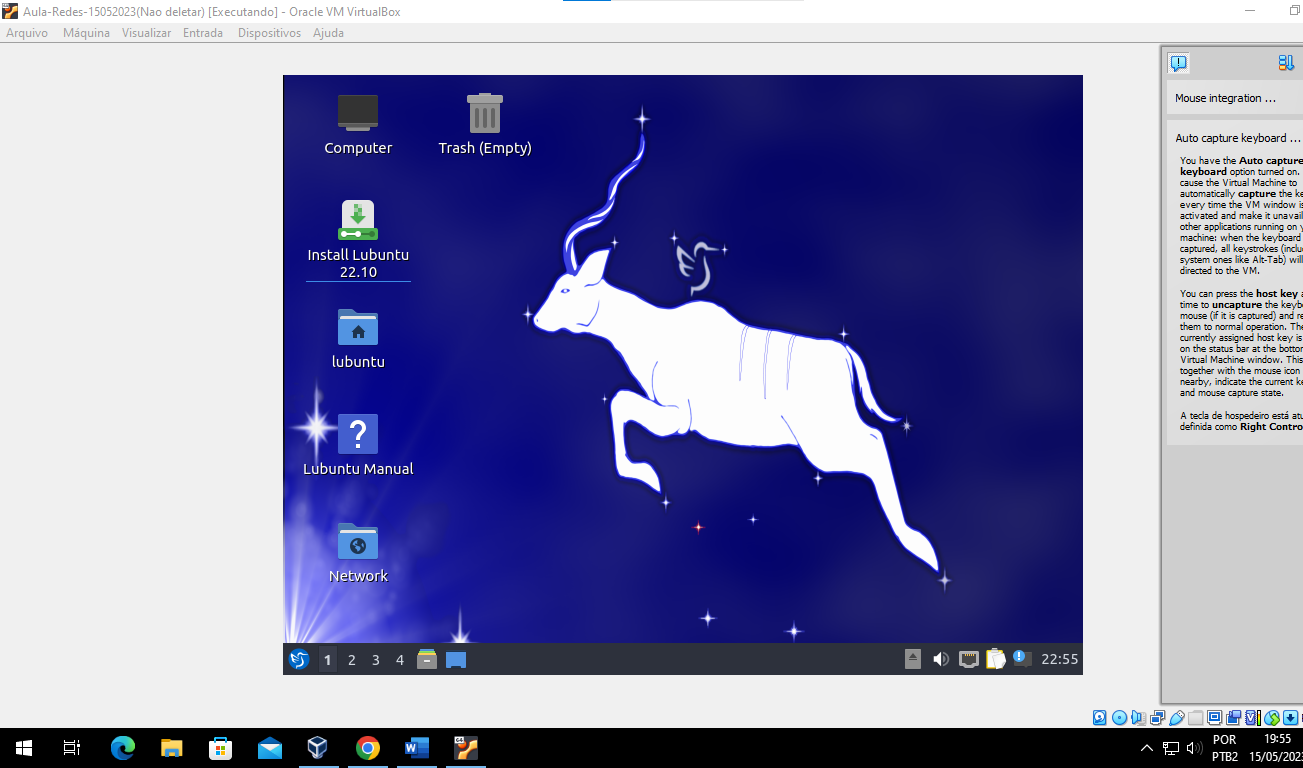


Na imagem acima, selecione a primeira opção.

Começou a rodar...

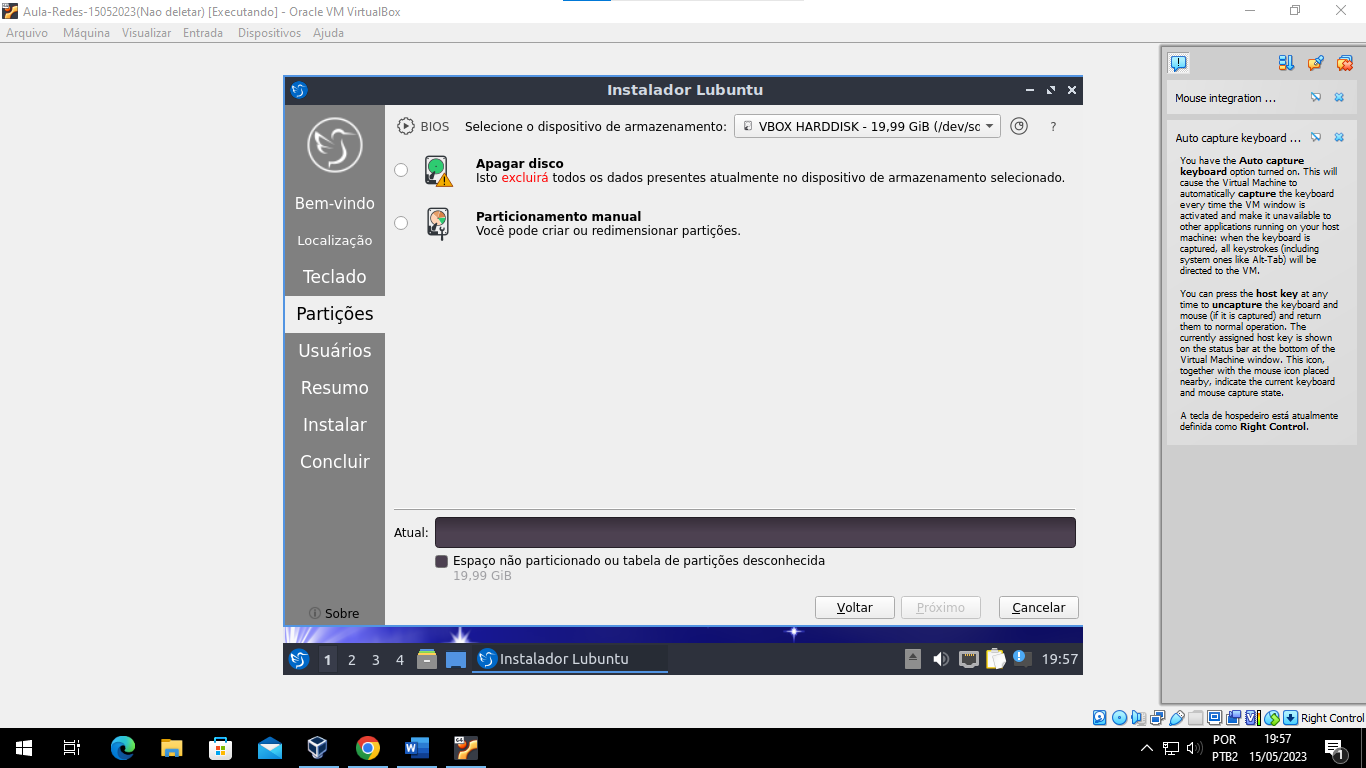


Clique em Install Lubuntu 22.10

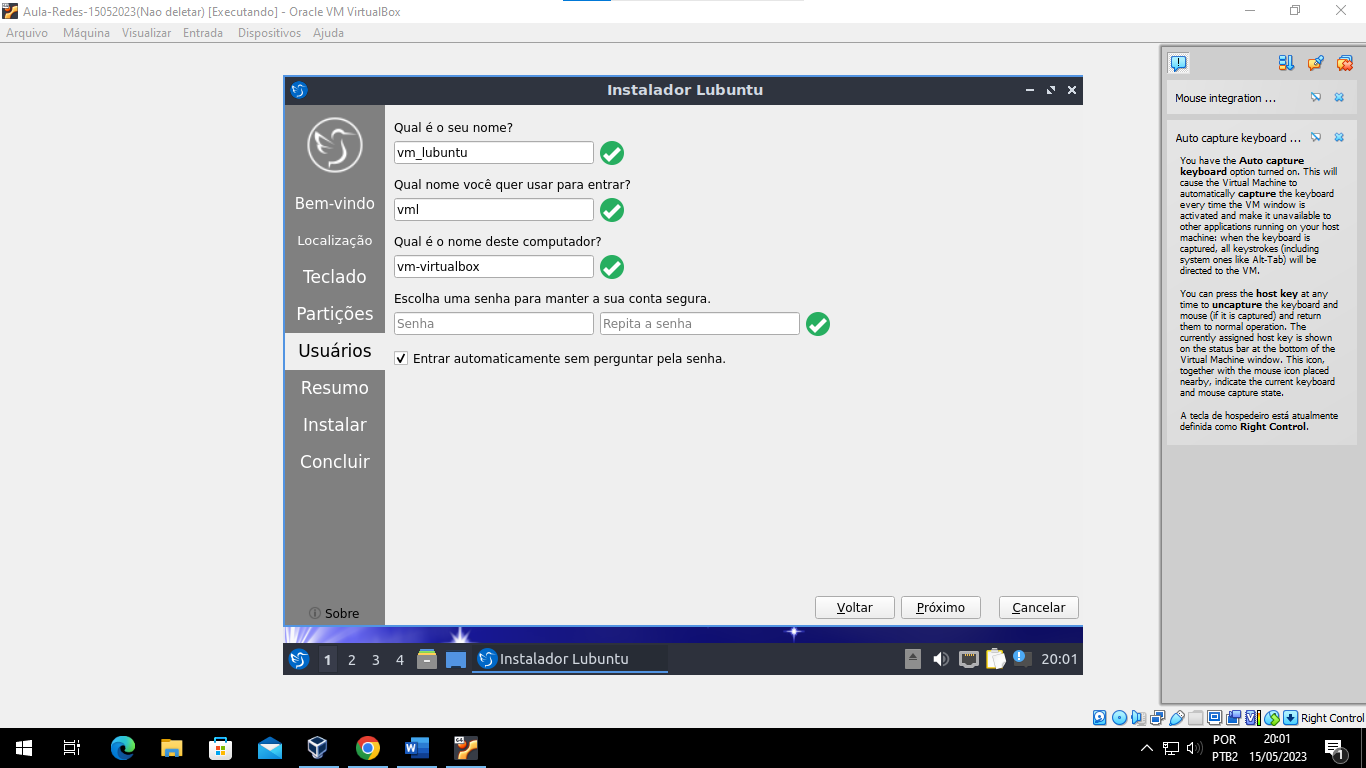


Após esse passo são necessários alguns detalhes, como idioma, localização, teclado, configure como desejado e clique em próximo.

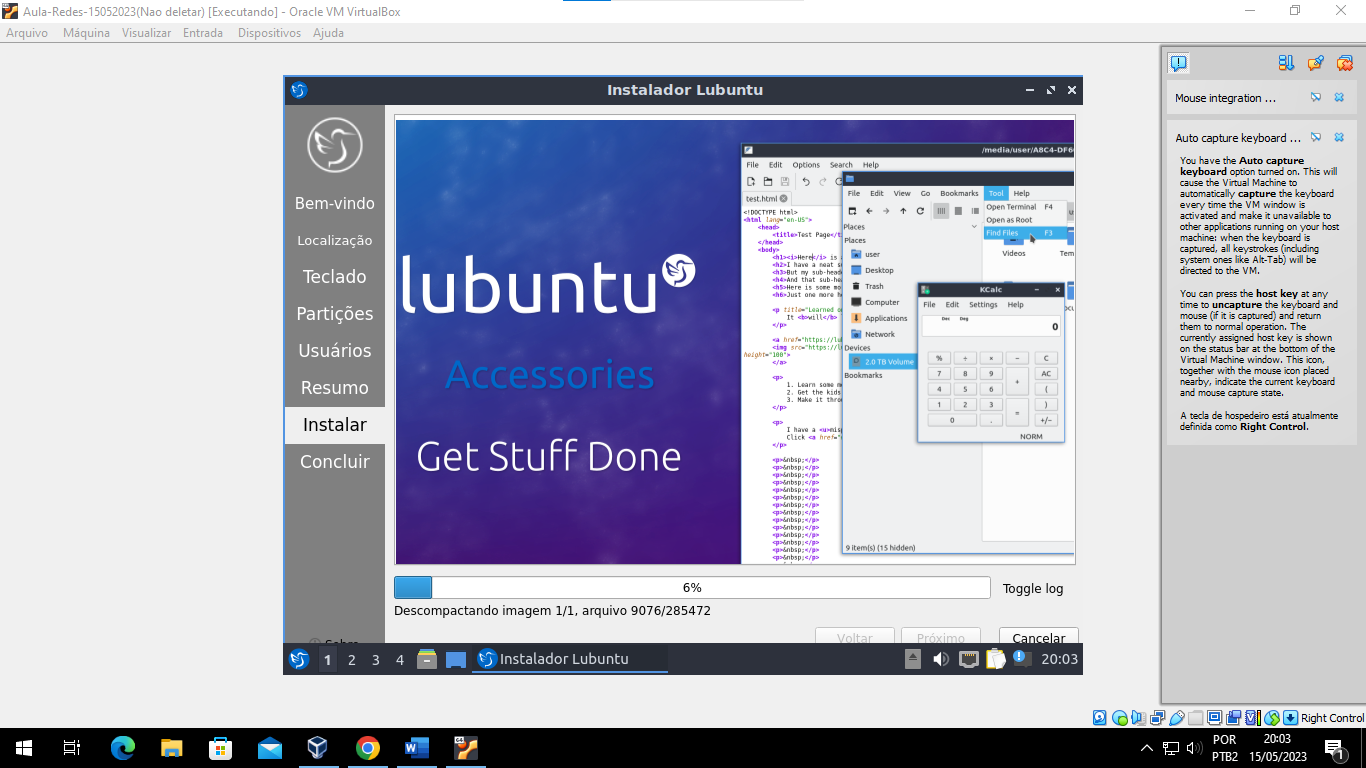
Ao chegar nessa tela, selecione a primeira opção.

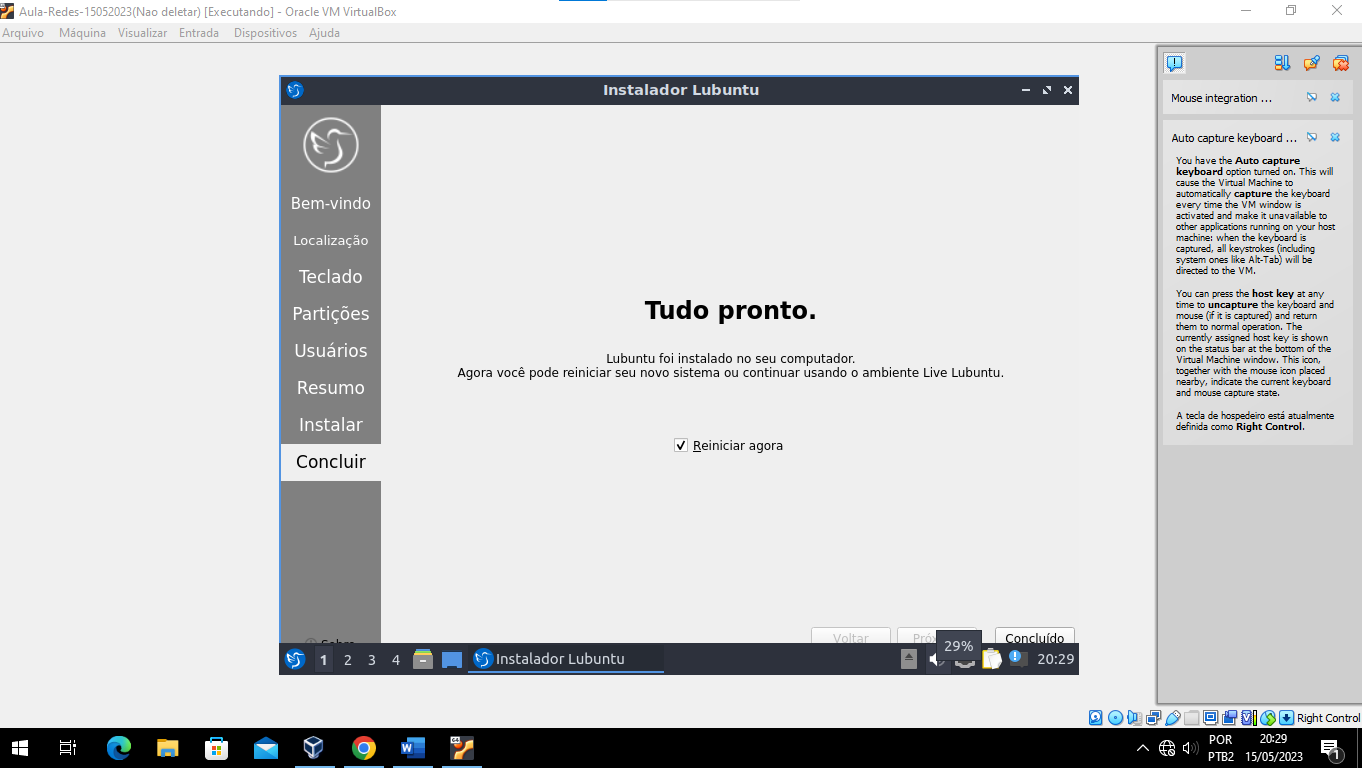


Nessa parte crie o usuário de acesso.

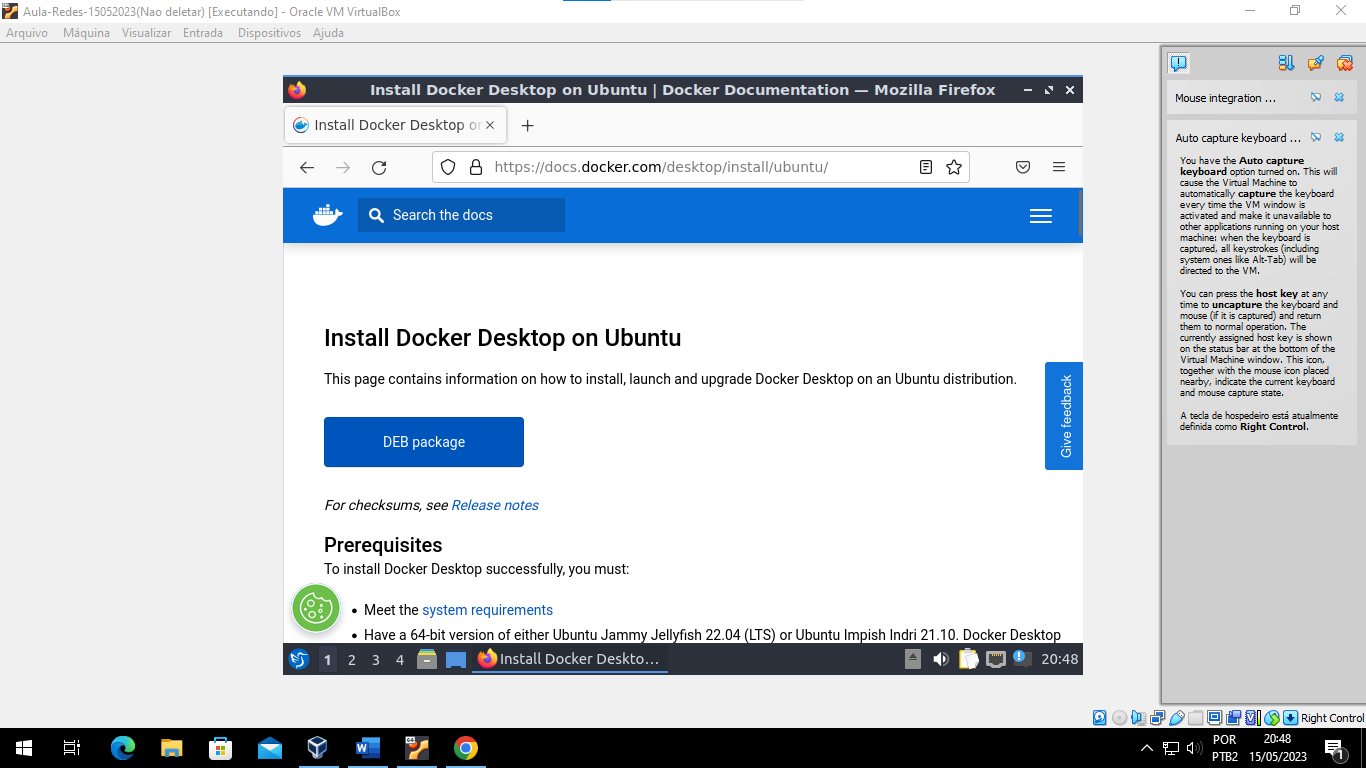


Leia o resumo, e confirme se todas as especificações estão corretas. Após isso clique em instalar, e aguarde o procedimento.



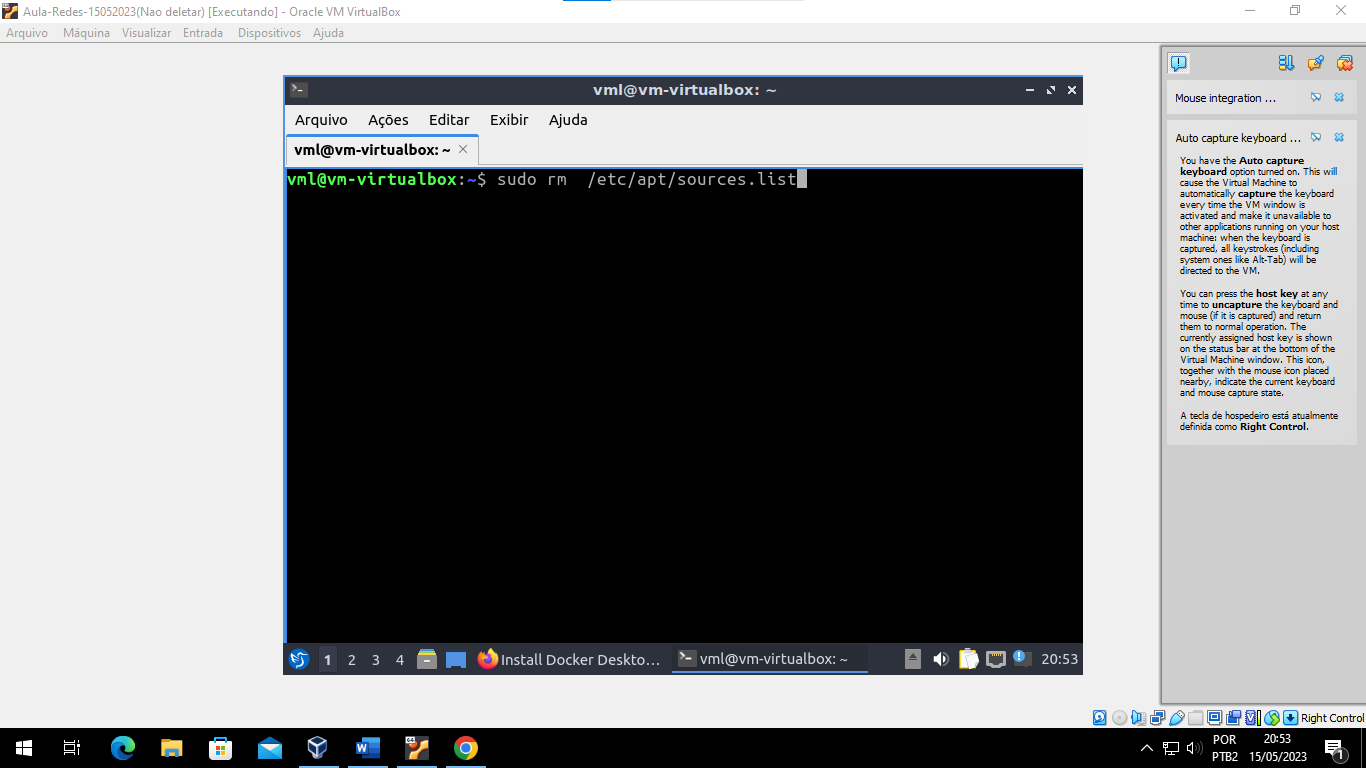


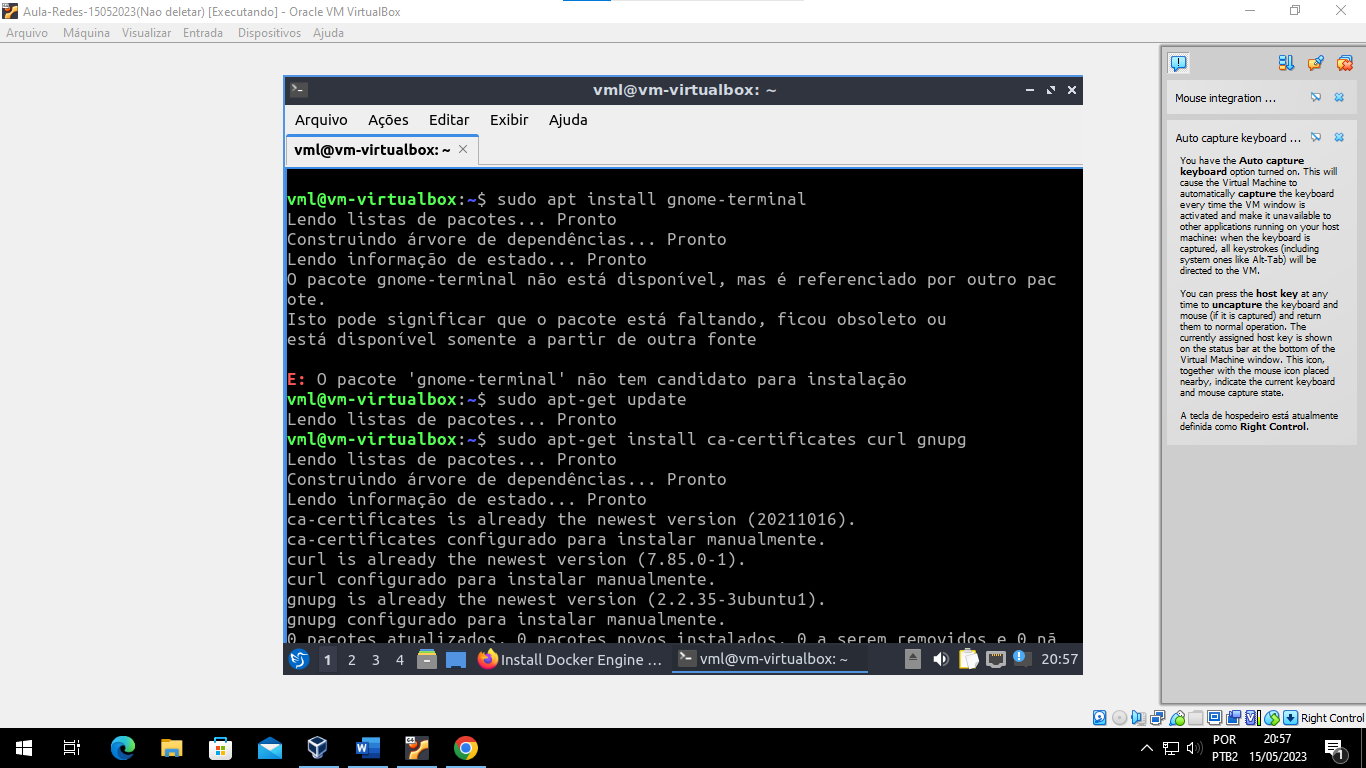
Abra a página lubuntu manual, e pesquise por lubuntu install Docker, acesse o primeiro site.



**3º Passo**

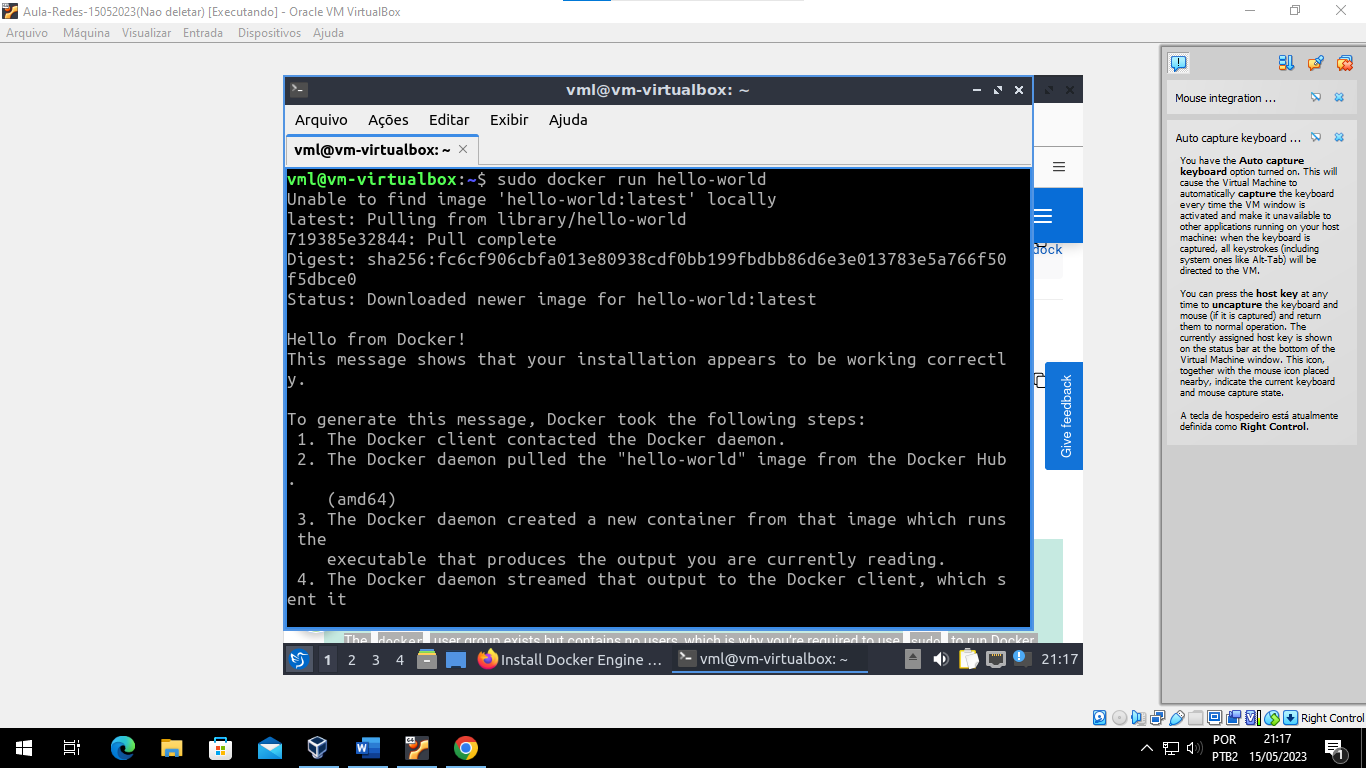
Instalação do Docker

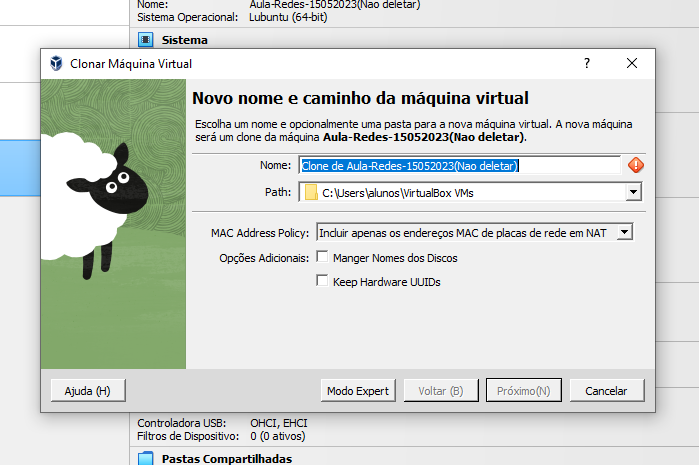
Escreva o código a seguir, dentro do terminal.

Digite e execute os comandos necessários como instruído no site do Docker, dentro do terminal.

**4º Passo:**

Suba o contêiner Hello-World ao Docker. Como foi adicionado o comando **run** na linha, não é necessário dar o start.

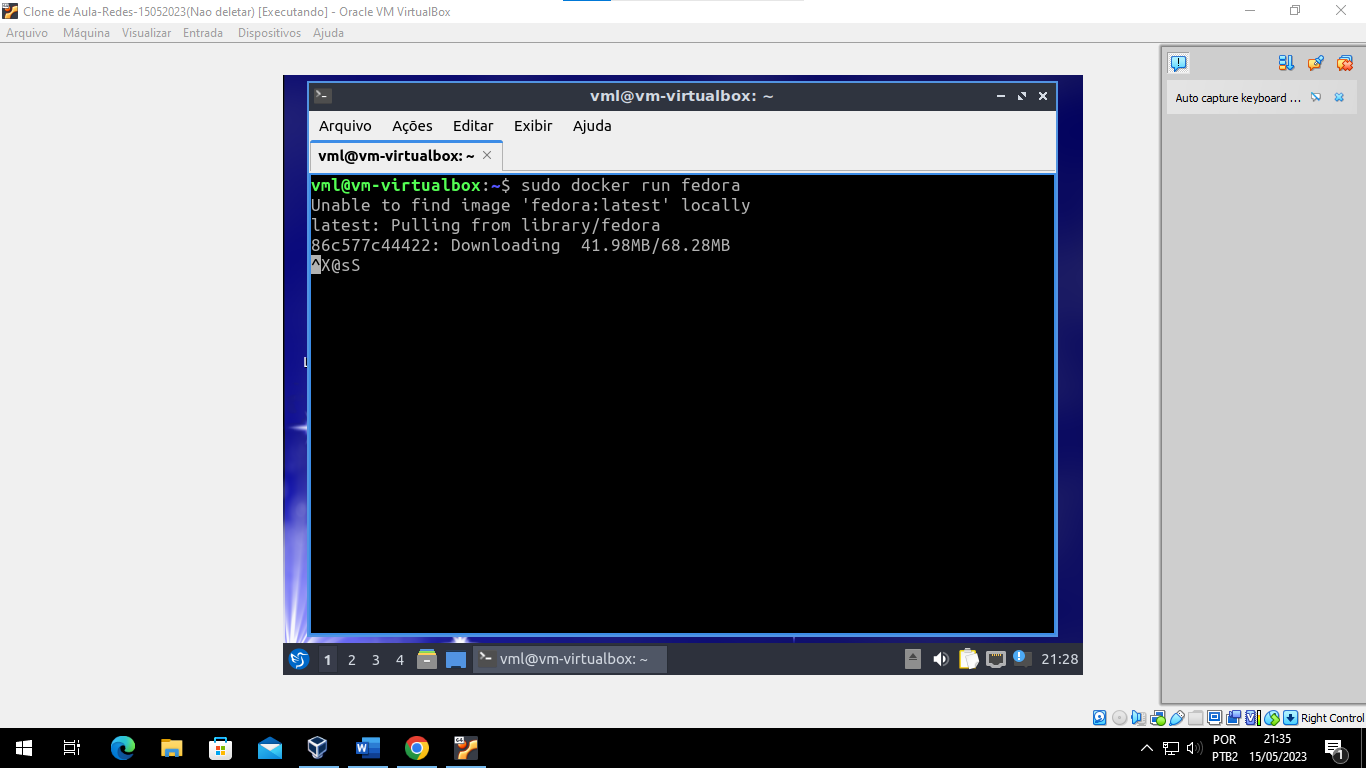


**5º Passo:**Clonando a máquina

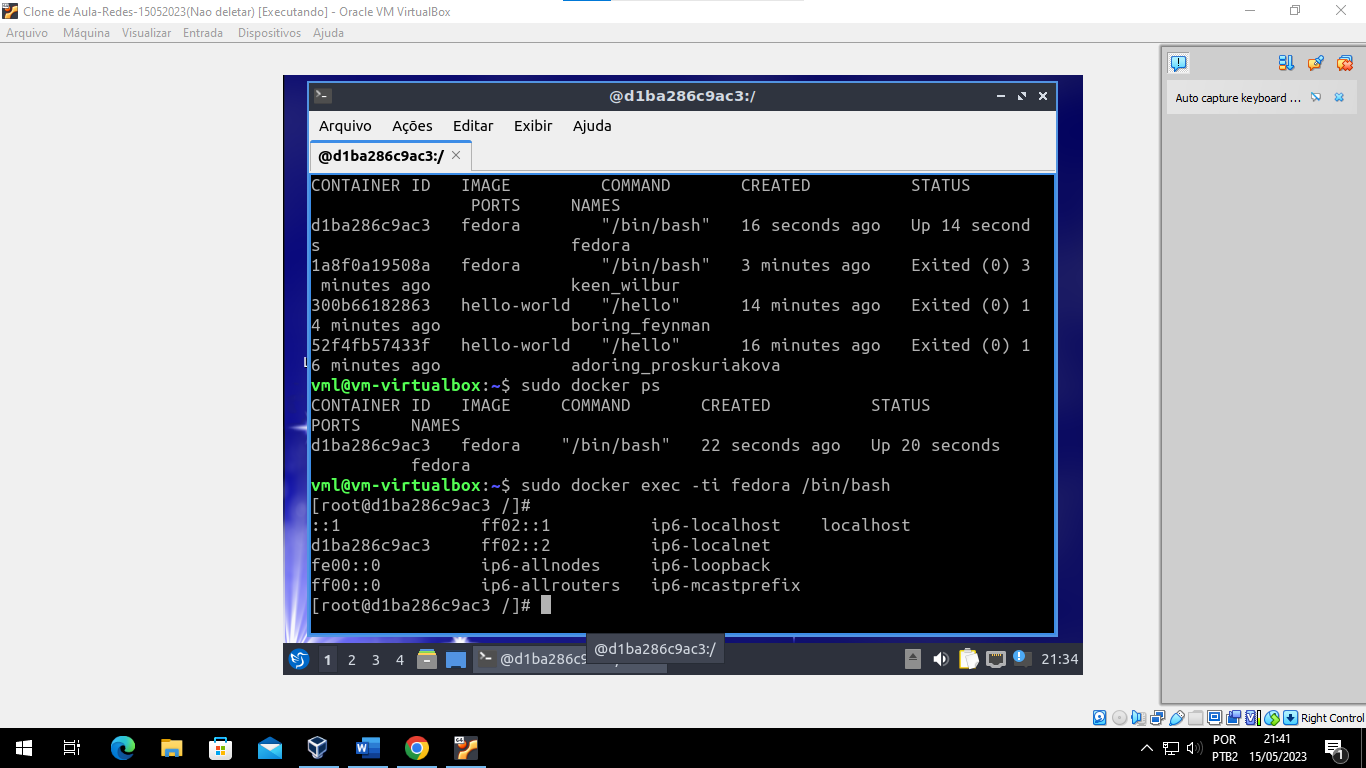
Acesse a máquina desejada, clique com o botão direito e selecione a segunda opção (clonar), após isso é só configurar como desejado e clicar em próximo.

**PARTE 2:**

Adicione a imagem do fedora:



Inicie o container Fedora em daemon (-d) e acesse o container.



**Parte 3: Adicionar execução dos slides**

Execução dos comandos:

docker pull hello-world

docker run hello-world

docker run alpine

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Execução dos seguintes comandos:

docker ps

docker ps -a

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Execução dos seguintes comandos:

docker run alpine echo “Ola Mundo”

docker run ubuntu:18.04 cat /etc/issue

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Execução dos seguintes comandos:

docker run -d alpine sleep 20

docker run -rm alpine echo “Ola Mundo”

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Exercício Pagina 43

1. Crie um container CentOS
2. Execute o comando sleep e suspenda o SO por 2000 segundos
3. Verifique os containers em execução
4. Pare o container que esta executando o comando sleep

Interface gráfica do usuário, Texto

Descrição gerada automaticamente Interface gráfica do usuário, Texto

Descrição gerada automaticamente

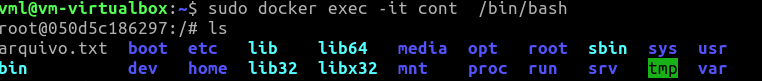
Exercício pagina 45

1. Crie um container Ubuntu no modo iterativo
2. Crie um arquivo de texto dentro do container com o comando touch
3. Saia do container
4. Utilize o comando run novamente para tentar “acessar” o container passado
5. Busque pelo arquivo criado anteriormente com o comando ls

Texto

Descrição gerada automaticamente





Exercício pagina 49

1. Verifique qual a versão do Docker Server Engine que esta sendo executado na sua maquina
2. Verifique quantas imagens estão disponíveis no Docker Host
3. Execute um container utilizando uma imagem do Redis
4. Pare o container que você acabou de criar
5. Verifique quantas imagens estão sendo executadas no momento
6. Crie os seguintes containers
7. Alpine com sleep de 1000s
8. Nginx:alpine com sleep de 900s
9. Nginx:alpine com sleep de 1500s
10. Ubuntu com sleep de 1000s
11. Alpine
12. Redis

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente  
Texto

Descrição gerada automaticamente





Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente