

**INSTITUTO INFNET
ESCOLA SUPERIOR DE TECNOLOGIA DA
INFORMAÇÃO
GRADUAÇÃO EM GESTÃO DA TECNOLOGIA DA
INFORMAÇÃO**



Projeto em Arquitetura de Infraestrutura de Aplicações

Aluno: Alexandre Garcia Barbeito Ferreira

Matrícula: 69165041100

E-MAIL: alexandre.pferreira@al.infnet.edu.br

TESTE DE PERFORMANCE 09

Sumário

1	Introdução	3
1.1	Objetivo	3
2	Referencial Teórico	4
2.1	Arquitetura de Aplicações	4
2.2	Arquitetura de Infraestrutura	9
2.2.1	Arquitetura e design: Hyper-V vs VMware	9
2.2.2	Sistemas operacionais suportados - Hyper-V vs VMware.....	12
2.2.3	Diferenças de desempenho: Hyper-V vs VMware	13
2.2.4	Comparação de memória entre Hyper-V e VMware	14
3	Desenvolvimento e Implantação	16
3.1	Implantação do Ambiente.....	19
3.1.1	Implantação da Infraestrutura	19
3.1.1.1	Instalação do ESXi.....	19
3.1.1.2	Instalação Criação da Máquina Virtual e Instalação do Ubuntu	22
3.1.2	Implantação da Aplicação	24
3.1.2.1	Instalação do Ansible	24
3.1.2.2	Instalação do Docker	24
3.1.2.3	Instalação dos Containers MySql e WordPress.....	26
3.2	Oportunidades de melhoria	28
3.3	Cronograma.....	30
4	Conclusão	32
	Referência Bibliográfica	33

1 Introdução

A AMAZONIA UMA VISAO PARA TODOS é uma ONG que tem como objetivo apresentar ao mundo uma visão realista sem distorção política ideológica do que é a Amazônia e seus problemas sociais, econômicos e ecológicos. Para isso ela utiliza dados de profissionais voluntários, publicações científicas além de outras fontes, com a intensão de se tornar a mais isenta possível e livre de qualquer tipo de censura. Por isso, optou-se por implantar um blog onde será realizada a interação entre a sua visão e as informações captadas pela ONG com a sociedade interessada, ao invés de simplesmente realizar a interação através de redes sociais. Com a criação do blog, debates sobre o assunto poderão ser realizados e até eventos de esclarecimentos poderão ser organizados.

Dessa forma, nesse projeto iremos tratar a escolha da aplicação adequada e sua infraestrutura para atender às necessidades da ONG AMAZONIA UMA VISAO PARA TODOS.

No projeto descreveremos como será realizada a implantação da arquitetura de virtualização e compararemos com algumas outras abordagens de *Hypervisor*. Iremos tratar também de como realizar a instalação da aplicação distribuída WordPress de forma automatizada utilizando um *playbook* Ansible.

Todas as referências as versões/downloads de todos os componentes do WordPress e etapas de sua implantação na infraestrutura de virtualização constaram desse projeto.

1.1 Objetivo

Esse projeto tem como objetivo apresentar de forma clara e organizada as características infraestrutura, desenvolvimento, implantação e configuração do WordPress e os detalhes técnico envolvidos.

Nesse documento serão mostrados os passos para instalação automatizada da aplicação escolhida através do Ansible e seu cronograma estimado para execução e configuração de cada atividade até o funcionando do WordPress.

2 Referencial Teórico

Como solução foi escolhida uma infraestrutura *on premises* utilizado o WordPress como aplicação distribuída onde será abrigado o blog da organização.

2.1 Arquitetura de Aplicações

O WordPress é uma ferramenta robusta que possui várias camadas, como por exemplo uma camada WEB, uma camada de banco de dados entre outras.

O banco de dados (BD) do WordPress armazena todos os dados de uma plataforma *Content Management System* (CMS). Não apenas as informações básicas, mas também as credenciais de login. A figura 2.1.1 mostra as diversas informações que o BD do WordPress pode armazenar.



Figura 2.1.1 - WordPress

Um middleware é desenvolvido para que a comunicação entre o banco de dados e a camada web ou *frontend* seja transparente para o usuário (figura 2.1.2).

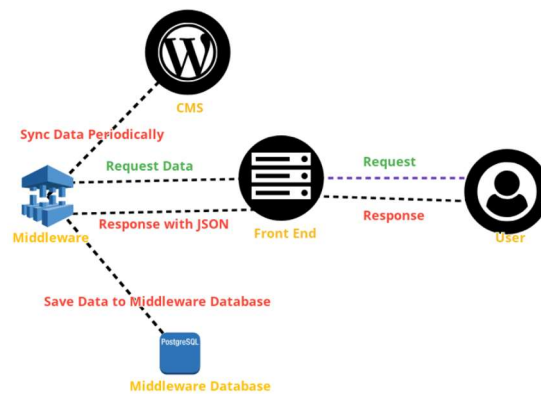


Figura 2.1.2 - Middleware WordPress

A aplicação Web pode estar em um servidor e o banco de dados em outro. Distribuindo o serviço, é possível balancear a carga e separar os servidores de acordo com a sua atividade, por exemplo, banco de dados e aplicações web. Entretanto no momento não vimos necessidade para implantação ser dividida em servidores diferentes. Porém, havendo necessidade essa divisão é totalmente possível e rápida utilizando as facilidades da imagem Docker explicado mais adiante nesta seção.

O WordPress é uma ferramenta extremamente versátil e pode ser adaptado para uma infinidade de propósitos. No caso da AMAZONIA UMA VISAO PARA TODOS ela será usada como BLOG.

Como características/vantagens do WordPress temos:

- **Facilidade em operar** – não precisa ser um expert em internet para administrar ou publicar conteúdo em um site WordPress.
- **Amigável aos mecanismos de busca** – a estrutura padrão das páginas e posts já possui todos os elementos html que são necessários para o Google e outros buscadores entendam e indexem seu conteúdo.
- **Pode ser exportado/migrado com facilidade** – se em qualquer momento for preciso mudar de plano de hospedagem, é muito fácil migrar a instalação do WordPress de um servidor para outro. Algumas empresas de hospedagem até oferecem o procedimento como um serviço gratuito.
- **É suportado por uma comunidade gigantesca** – respostas para praticamente qualquer questão na documentação ou no fórum oficial

da plataforma. No Google é possível encontrar uma infinidade de blogs e sites falando sobre o assunto.

- **Possui milhares de especialistas** – se em algum momento for necessária a ajuda profissional, ou seja, contratar alguém para customizar ou incrementar o WordPress, existem muitos profissionais, entre designers, desenvolvedores e programadores especialistas na plataforma.
- **Adaptável a qualquer cenário** – O WordPress pode ser usado como fórum online, loja virtual, área exclusiva para determinados assinantes. Apesar de não ser a solução para todas as situações, o que seria impossível, o WordPress é bastante flexível e pode se transformar em diferentes ferramentas.

São recursos nativos da ferramenta:

- Editor de páginas e posts com recursos avançados de formatação de textos;
- Biblioteca de mídia com recursos para edição simples de imagens (ex.: recorte e giro);
- Gerenciador de comentários com moderação incluída por padrão;
- Editor de aparência do site, onde é possível editar diversos itens do templates, como cores, marca e menus;
- Editor de widgets, que são áreas do site para inclusão de informações especiais, como menus personalizados, categorias, agenda e uma infinidade de outros recursos;
- Gerenciador de plugins;
- Gerenciador de temas;
- Gerenciador de usuários, com definição de capacidades por usuário (ex.: administrador, editor, colaborador etc.);
- Configuração de URLs amigáveis;
- Plugin anti-spam (Akismet) instalado por padrão, para filtrar spam nos comentários.

A entrega da aplicação é feita por qualquer browser através do endereço www.amazoniaumavisaoparatodos.org.br que pode ser colocado como página default através de políticas de segurança ou simplesmente distribuído e

divulgado através de e-mail. A divulgação também poderá feita através de redes sociais e Mailing List.

Seu código fonte é aberto e se for necessário navegar no código fonte e na documentação do WordPress, é possível ser feito pela [referência prática para desenvolvedores](#) e por um [navegador de código](#). Também existem guias para o [sistema de controle de versão](#) e para a [contribuição com o Git](#).

A fonte compilada do WordPress, [licenciada](#) sob a versão 2 (ou versão mais recente) da licença GNU (General Public License), pode ser [consultada online](#) ou vista localmente com o *subversion* ou Git:

- *Subversion*: <https://core.svn.wordpress.org/>
- Espelho do Git: <git://core.git.wordpress.org/>

Os arquivos básicos do WordPress em JavaScript são minificados com o UglifyJS, e os em CSS, com o clean-css, tudo pelo [Grunt](#), um executor de tarefas baseado em JavaScript. O código de desenvolvimento, que inclui as versões originais dos arquivos e scripts de compilação, pode ser visto online ou baixado via *subversion* ou Git:

- *Subversion*: <https://develop.svn.wordpress.org/>
- Espelho do [Git](#): <git://develop.git.wordpress.org/>

O código fonte de qualquer dos binários do programa ou dos scripts externos comprimidos que são incluídos no WordPress pode ser livremente baixado do [repositório de fontes](#) do próprio WordPress.

Foi escolhido o WordPress em detrimento do Joomla!, Ghost, Textpattern, Drupal entre outros por sua robustez, variedade de widgets e plugins disponíveis, quantidade de comunidades de apoio além de sua versatilidade de adaptação. O WordPress é o mais popular e responsável por mais da metade das páginas na Internet indexadas.

O Joomla! foi preterido, pois apesar de semelhante foi projetado para sites maiores, seu código e interface foram projetados para facilitar o gerenciamento de um grande número de páginas e postagens o que não acontecerá em nossa solução.

Da mesma forma que o Joomla! o Drupal foi projetado para projetos maiores e possui uma complexidade maior que o WordPress para desenvolvê-lo.

O Docker é uma tecnologia de software que fornece contêineres, promovido pela empresa Docker, Inc. O Docker fornece uma camada adicional de abstração e automação de virtualização de nível de sistema operacional no Windows e no Linux. A plataforma usa as características de isolamento de recurso do núcleo do Linux como cgroups e espaços de nomes do núcleo, e um sistema de arquivos com recursos de união, como OverlayFS e outros para permitir "contêineres" independentes para executar dentro de uma única instância Linux, evitando a sobrecarga de iniciar e manter máquinas virtuais (VMs). Quando falamos em Docker, estamos falando em Docker Engine. O Docker Engine permite criar e executar contêineres. No entanto esses contêineres devem iniciar a partir de um arquivo Docker. O arquivo Docker define tudo o que é necessário para executar a imagem, incluindo as especificações de rede do sistema operacional e os locais dos arquivos, com todas essas definições é possível criar a imagem Docker, que é o componente estático portátil executado no Docker Engine.

Nesta tecnologia, o kernel da máquina hospedeira é compartilhado com o software em operação, portanto o desenvolvedor pode agregar a seu software a possibilidade de levar as bibliotecas e outras dependências do seu programa junto ao software com menos perda de desempenho do que a virtualização do hardware de um servidor completo. Assim, o Docker torna operações em uma infraestrutura como serviços web mais intercambiável, eficientes e flexíveis. (REDHAT, 2020)

O Kubernetes (comumente estilizado como K8s) é um sistema de orquestração de contêineres open-source que automatiza a implantação, o dimensionamento e a gestão de aplicações em contêineres. Foi originalmente projetado pelo Google e agora é mantido pela Cloud Native Computing Foundation. Funciona com uma variedade de ferramentas de containerização, incluindo Docker. (KUBERNETES, 2020) (REDHAT, 2020)

No nosso ambiente não encontramos necessidade para utilização do Kubernetes nesse momento.

O ansible é um mecanismo de automação de TI radicalmente simples que automatiza o provisionamento em nuvem, o gerenciamento de configurações, a implantação de aplicativos, a orquestração de infraestrutura como serviço e muitas outras necessidades de TI.

O mecanismo do ansible funciona conectando-se aos nós e executando programas pequenos, chamados "Módulos Ansible". Esses programas foram criados para serem modelos de recursos do estado desejado do sistema. O ansible executa esses módulos (por SSH por padrão) e os remove quando finalizados. (RED HAT ANSIBLE)

2.2 Arquitetura de Infraestrutura

Atualmente, os dois principais produtos para virtualização em datacenters corporativos são os *hypervisors* Microsoft Hyper-V e VMware vSphere. Ao analisar o Hyper-V e VMware, cada um apresentou pontos forte e considerações em usar um ou outro. No nosso caso a escolha foi em função do maior conhecimento na implantação da solução da VMware.

VMware vSphere é um termo que abrange as principais soluções de virtualização para gerenciar, monitorar e configurar um data center virtual. O núcleo central do vSphere é o próprio *hypervisor*. O *hypervisor* ESXi da VMware é o *hypervisor* de virtualização produzido pela VMware que é executado no servidor bare metal. Expandindo a partir do *hypervisor* principal do ESXi, o vSphere é composto por um conjunto de produtos de virtualização de data center corporativo. O VMware inclui uma gama de ferramentas adicionais que adicionam funcionalidade ao produto principal do VMware vSphere. Isso inclui VMware vSAN, VMware NSX, VMware vRealize Operations Management. (REED, 2018)

O Hyper-V é uma plataforma de *hypervisor* de data center corporativa lançada pela Microsoft a partir do Windows Server 2008. O Hyper-V é um *hypervisor* de tipo 1 que é executado como uma função instalada na plataforma Windows Server. O Hyper-V pode ser executado como um servidor autônomo ou como parte de um cluster em execução nos serviços de cluster de failover do Windows com armazenamento compartilhado. (LEE, 2019)

2.2.1 Arquitetura e design: Hyper-V vs VMware

O Hyper-V é um *hypervisor* Tipo 1. Como o Hyper-V é instalado como uma função dentro do Windows Server, muitos concluíram que esse seria um *hypervisor* Tipo 2; no entanto, a Microsoft realiza uma engenharia bastante

eficiente aqui, depois que a função Hyper-V é ativada, o Hyper-V é realmente instanciado primeiro, e o sistema operacional host é executado sobre o Hyper-V.

O Hyper-V virtualiza processadores e memória e usa várias partições para expor serviços e dispositivos de E/S às máquinas virtuais convidadas, além de oferecer suporte ao isolamento.

Essas partições são unidades lógicas de isolamento onde o sistema operacional é executado. As partições não têm acesso direto ao processador físico, mas têm uma visão virtual da CPU. O Hyper-V manipula interrupções para o processador e as redireciona para a respectiva partição necessária. (NOVELLO, 2017)

- Raiz - Partição executando o Microsoft Windows e é a partição original que inicia o *hypervisor*;
- Pai - É partição que a pilha de virtualização existe, bem como o gerenciador de memória para VMs, APIs de gerenciamento e dispositivos de E/S virtualizados. Os dispositivos emulados também são atendidos nesta partição. Isso é realizado com os *Virtualization Service Providers* (VSPs) que se comunicam pelo VMBus para manipular as várias solicitações de dispositivo das partições filha;
- Partições filha - A partição filha hospeda os sistemas operacionais da máquina virtual convidada. Eles são criados usando a API *hypercall*.

O Hyper-V requer um processador que inclua virtualização assistida por hardware. Isso inclui a tecnologia Intel VT ou AMD Virtualization (AMD-V).

Embora o Hyper-V possa ser executado em um servidor autônomo, para resiliência e alta disponibilidade, o Hyper-V é executado em vários servidores que fazem parte de um Cluster de *Failover* do Windows. As máquinas virtuais em execução nos clusters Hyper-V são executadas como um recurso de cluster, permitindo alta disponibilidade. A execução de hosts Hyper-V em um cluster de *failover* do Windows com algum tipo de armazenamento compartilhado permite recursos como o *Live Migration*, em que a computação e a memória da VM podem ser transferidas entre os hosts Hyper-V no cluster. A Microsoft também trabalhou bastante na área do Cluster de *Failover* do Windows. Agora, com o

Windows Server 2019, torna-se possível migrar VMs Hyper-V entre clusters. Isso permite estruturar o *Windows Failover Clustering* de maneiras mais estratégicas para satisfazer as necessidades de negócios. (LEE, 2019)

O ESXi da VMware é um sistema operacional semelhante ao POSIX desenvolvido que fornece recursos e funcionalidades muito semelhantes a outros sistemas operacionais. No entanto, ele foi projetado e desenvolvido especificamente para executar máquinas virtuais. O ESXi é um software para ser instalado em um servidor bare-metal que serve como um *hypervisor* Tipo 1.

O ESXi é uma evolução do *hypervisor* original ESXi ou "Elastic Sky X" que incluía um console de serviço. A pilha de software ESXi cria uma camada de abstração que virtualiza o hardware do servidor físico. Isso permite que as máquinas virtuais usem os recursos de hardware do servidor físico totalmente inconscientes das outras VMs em execução no sistema. Um dos pontos fortes do VMware ESXi é a pegada extremamente pequena quando instalada. O ESXi é tão pequeno que pode ser executado inteiramente na memória. Todas as operações são executadas no sistema de arquivos na memória, que basicamente armazena todos os arquivos de configuração que permitem as principais funcionalidades do ESXi. (LEE, 2019)

O *hypervisor* VMware ESXi é composto de um conjunto de componentes principais, incluindo:

- VMkernel;
- Direct Console User Interface (DCUI) no User World;
- Monitor de Máquina Virtual;
- Vários agentes;
- Sistema Modelo de Informação Comum (CIM).

O VMkernel é a principal força de trabalho do ESXi, responsável pelo agendamento de todos os recursos no sistema, com os recursos virtuais sendo solicitados. O VMkernel cria uma camada de abstração necessária para que as máquinas virtuais usem os recursos no sistema como usariam em uma máquina física e fazem isso de maneira isolada das outras máquinas virtuais em execução no host VMware ESXi.

O VMkernel também é composto de uma camada de armazenamento que utiliza o sistema de armazenamento de arquivos proprietário da VMware

chamado Sistema de Arquivos da Máquina Virtual ou mais conhecido como VMFS. Além disso, há uma pilha de rede virtual responsável pela criação de comutadores virtuais que permite que as máquinas virtuais interajam e se comuniquem com a rede física.

O *User World* é o conjunto de *end points* da API que permitem que outros mecanismos interajam com o VMkernel, incluindo o DCUI, os agentes de gerenciamento, os processos auxiliares do mundo do usuário do VMX e muitos outros. Conforme descrito, o *User World* é a camada de software que permite a interação do usuário e outros processos relacionados ao usuário.

A Interface do usuário do console direto (DCUI) é a interface de baixo nível que permite aos administradores executar tarefas de configuração e gerenciamento e é acessível através do console do servidor.

Além disso, duas APIs do *User World* extremamente importantes são os agentes de gerenciamento - *hostd* e *vpix*. Esses dois agentes de gerenciamento são responsáveis por ajudar a enviar comandos para o *hypervisor* a partir de ferramentas de gerenciamento como o vSphere Client ou o vCenter Server.

2.2.2 Sistemas operacionais suportados - Hyper-V vs VMware

O ponto principal dos recursos oferecidos pelo *hypervisor* é executar sistemas operacionais convidados.

A [documentação mais recente da Microsoft](#) sobre os sistemas operacionais convidados Windows Server 2019 Hyper-V com suporte inclui uma ampla mistura de sistemas operacionais Windows nativos e a maioria dos sistemas operacionais Linux padrão.

Os sistemas operacionais Windows suportados incluem (MICROSOFT, 2019):

- Windows Server 2019;
- Windows Server 2016;
- Windows Server 2012 R2;
- Windows Server 2012;
- Windows Server 2008 R2;
- Windows Server 2008 com SP2;
- Windows 10;

- Windows 8.1;
- Windows 8;
- Windows 7 com SP1.

Os sistemas operacionais suportados pelo Linux incluem:

- Centos;
- Debian;
- Oráculo;
- SUSE;
- Ubuntu;
- FreeBSD.

O VMware fornece um conjunto muito mais diversificado de sistemas operacionais convidados suportados que podem ser executados com a versão mais recente do VMware vSphere 6.7 Update 2.

O VMware suporta todas as versões mais recentes do sistema operacional Windows Server e do cliente Windows, mas também de várias variantes do Linux, incluindo variantes de nuvem como o Amazon Linux 2. Além disso, outros sistemas operacionais convidados, como o macOS Desktop e o macOS Server que são suportados no VMware vSphere e não no Hyper-V. (LEE, 2019)

2.2.3 Diferenças de desempenho: Hyper-V vs VMware

O Hyper-V, por meio da versão mais recente do Windows Server 2019, bem como o VMware vSphere 6.7 Update 2, são plataformas poderosas nas quais são executadas cargas de trabalho críticas aos negócios. Ambos os *hypervisors* possuem tecnologias de aprimoramento de desempenho que podem ajudar a melhorar o desempenho dos sistemas operacionais convidados.

O Windows Server 2019 Hyper-V e o VMware vSphere 6.7 oferecem suporte à memória persistente, que é uma nova tecnologia de armazenamento que pode reduzir significativamente a latência de armazenamento.

Tanto o Hyper-V quanto o VMware ESXi, em suas variantes mais recentes, entendem a arquitetura Non-uniform memory access ([NUMA](#)) e têm recursos muito bons para lidar com eficiência com a arquitetura virtual [NUMA](#).

A versão mais recente da VMware no vSphere 6.7 divulgou muitas melhorias no agendamento da CPU, incluindo a eliminação do último bloqueio global, que permite ainda mais mundos relacionados à CPU virtual, o que ajuda a aumentar o desempenho em tecnologias como contêineres e micros serviços.

No mundo da HCI, o Hyper-V no Windows Server 2019 introduziu muitos grandes aprimoramentos à tecnologia Storage Spaces Direct em comparação com a versão anterior no Windows Server 2016. Eles adicionaram suporte à compactação e “desduplicação” de ReFS e dois clusters de nós. No entanto, a Microsoft ainda está tentando recuperar o atraso com o VMware vSAN como uma plataforma HCI. A VMware tem a vantagem definitiva aqui da plataforma mais madura e amplamente adotada para executar uma solução de armazenamento definida por software. (LEE, 2019)

2.2.4 Comparação de memória entre Hyper-V e VMware

O Hyper-V e o VMware têm técnicas e configurações de gerenciamento de memória muito diferentes à disposição dos que administram os dois *hypervisors*.

A principal técnica que o Hyper-V usa para gerenciar é o gerenciamento dinâmico de memória. Essa técnica é usada para adicionar dinamicamente mais RAM a uma máquina virtual em execução na infraestrutura do Hyper-V e recuperar a memória não utilizada quando a memória não está sendo usada pela VM. O componente de memória dinâmica do Hyper-V contém as seguintes configurações que podem ser ajustadas:

- RAM de inicialização
- RAM mínima
- RAM máxima
- Buffer de memória
- Peso da memória

Usando as configurações de memória dinâmica, a memória das máquinas virtuais Hyper-V pode ser adicionada ou liberada da máquina virtual de volta ao host Hyper-V.

Há uma limitação na memória dinâmica do Hyper-V que precisa ser observada. A memória dinâmica não pode ser usada com o NUMA virtual, o que

quer dizer que uma máquina virtual com memória dinâmica ativada será essencialmente bloqueada em um nó NUMA virtual e nenhuma topologia NUMA é apresentada à máquina virtual, independentemente das configurações virtuais do NUMA.

Com o VMware, o VMkernel gerencia a RAM física no host. O VMkernel aloca uma certa quantidade de RAM física para si e o restante é usado para máquinas virtuais. O espaço de memória física e virtual é dividido em blocos chamados páginas. Quando a memória física está cheia, os dados das páginas virtuais que não existem na memória física são armazenados em disco.

Alguns componentes de memória específicos da VMware:

- A memória da máquina virtual é alocada, o que inclui a quantidade de memória configurada na máquina virtual mais a sobrecarga adicional de memória necessária para virtualização;
- Compromisso de memória - Cada máquina virtual em execução recebe RAM reservada para todas as reservas presentes, além de sobrecarga de virtualização;
- Compartilhamento de memória - O compartilhamento de memória é uma técnica específica do VMware ESXi que ajuda a obter taxas de densidade mais altas de máquinas virtuais para um host. Você pode pensar no compartilhamento de memória um pouco como a “desduplicação” no nível da memória. O compartilhamento de memória pressupõe que várias máquinas virtuais possam estar executando o mesmo sistema operacional convidado e compartilhando aplicativos ou componentes semelhantes carregados. O mecanismo de compartilhamento de página transparente da VMware elimina cópias redundantes de páginas de memória. Isso resulta em níveis muito mais altos de comprometimento excessivo.

Outras técnicas de gerenciamento de memória VMware:

- Balão;
- Troca de *hypervisor*;
- Compressão de memória.

A principal técnica do Hyper-V com gerenciamento de memória é a memória dinâmica. No entanto, o VMware usa várias técnicas que incluem vários componentes para controlar adequadamente a memória e o comprometimento excessivo no VMware ESXi. (LEE, 2019)

3 Desenvolvimento e Implantação

A aplicação será implantada em um datacenter tipo SDDC (*Software Defined Data Center*) *on premises*, baseada em VMware ESXi 6.7.0 com um servidor virtualizado rodando Linux Ubuntu 18.04.4 LTS onde serão levantados container com SQL e Wordpress que darão suporte aplicação. Essa solução substituiu a AWS no projeto em virtude de não haver mais possibilidade em realizar os testes necessários devido ao término das horas gratuitas oferecidas na plataforma.

Nesse datacenter teremos:

- 01 Roteador de borda;
- 01 Servidor de firewall;
- 04 Switch;
- 01 Servidor de virtualização;
- 01 Storage;
- 01 Nobreaks;
- 01 Rack.

Essa infraestrutura terá como objetivo principal atender as necessidades desse projeto.

O roteador receberá um link de internet que realizará a ligação entre os clientes externos e o datacenter. Após esse roteador teremos servidor firewall que ficará responsável por realizar a proteção do ambiente do datacenter. Seguindo ao firewall teremos os switches que redistribuirão o acesso para toda a rede.

Conectado aos switches teremos o servidor de virtualização rodando o sistema operacional *Hypervisor* da VMWare. O ESXi 6.7 irá gerenciar as VMs do datacenter, que nesse momento será apenas uma (figura 3.1).

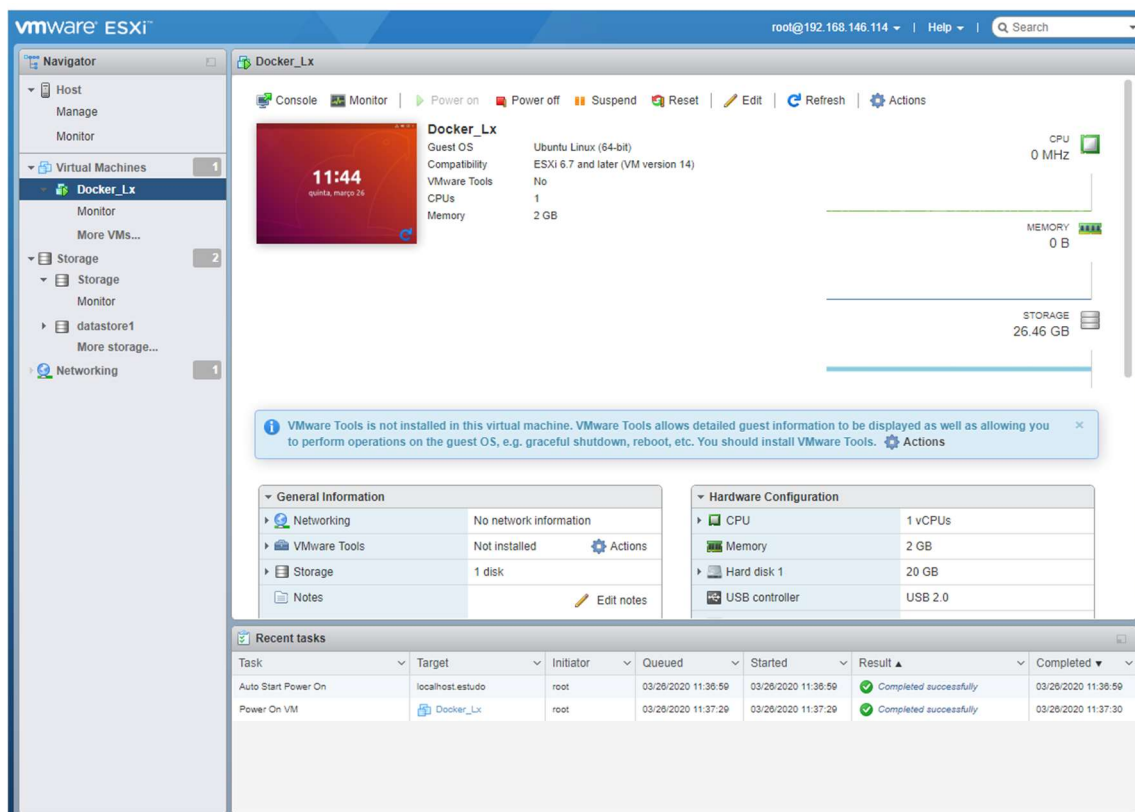


Figura 3.1 - ESXi

O datacenter contará em sua infraestrutura um sistema de ventilação que permita o ambiente mantenha uma temperatura entre 20°C e 25°C conforme orientado pela TIA 942. (TIA-942 GLOBAL REGISTRY)

Para garantir segurança de energia ininterrupta ao datacenter, este estará ligado a nobreaks que garantam o fornecimento de energia por pelo menos 5 horas.

Na figura 3.2 podemos observar um diagrama simplificado da estrutura do datacenter.

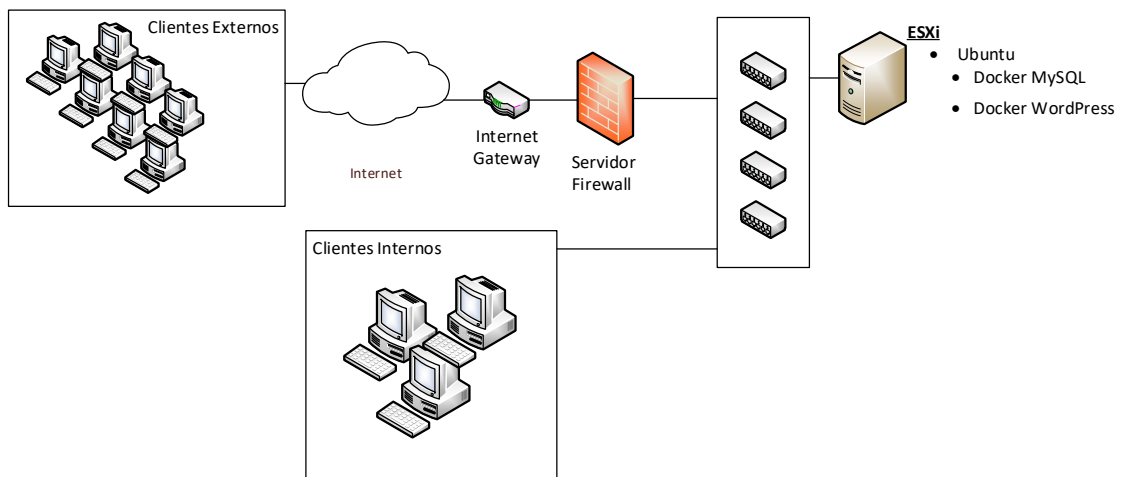


Figura 3.2 - Diagrama simplificado

Os recursos necessários recomendados pelo desenvolvedor para armazenar esse serviço são:

- Servidor baseado em UNIX/Linux1;
- PHP versão 7 ou superior;
- MySQL versão 5.6 ou superior OU MariaDB versão 10.0 ou superior;
- Memória para o PHP de pelo menos 64 MB (Somente para o software WordPress, sem plugins adicionais).

Como recursos extra:

- Memória para o PHP de pelo menos 256 MB;
- Apache ou Nginx;
- Módulo `mod_rewrite` do Apache ativo;
- Extensões PHP como `php_exif`, `php_GD` etc (recursos nativos e de plugins).

As atualizações são feitas na própria console de atualização do WordPress, podendo ser configuradas para serem feitas de forma manual ou automática. A console administrativa do WordPress faz o controle de versão da aplicação.

Os arquivos de código do WordPress e de criação dos contêineres via Ansible serão armazenados no GitHub para o gerenciamento de versões das aplicações e dos contêineres.

3.1 Implantação do Ambiente

O ambiente será implantado com base no sistema de virtualização da VMware. Para isso, usaremos o sistema operacional VMware ESXi 6.7.0 como host da nossa máquina virtual que irá rodar a distribuição Linux Ubuntu 18.04.4 LTS. Nessa máquina iremos instalar o SQL e o WordPress como contêineres Docker utilizando o Ansible.

A utilização de contêineres permite que uma rápida reimplantação dos sistemas e portabilidade para nuvem.

O VMware ESXi foi sistema de virtualização escolhido para criar a máquina virtual segura e totalmente isoladas que irá encapsular o Ubuntu onde rodará Docker com seus contêineres. A camada de virtualização do VMware mapeia os recursos físicos de hardware para os recursos da máquina virtual, de forma que a máquina virtual tenha seus próprios componentes, como CPU, memória, discos e dispositivos de entrada/saída.

3.1.1 Implantação da Infraestrutura

Ao final da implantação da infraestrutura o ambiente será capaz de oferecer um ambiente para instalação de máquinas virtuais que poderão ser gerenciadas conforme necessário.

3.1.1.1 Instalação do ESXi

Os requisitos mínimos para instalação do ESXi 6.7.0 são:

- Plataforma de servidor compatível. Para obter uma lista de plataformas suportadas, consulte o Guia de compatibilidade da VMware em <http://www.vmware.com/resources/compatibility>.
- O ESXi 6.7 requer um computador host com pelo menos dois núcleos de CPU.
- O ESXi 6.7 é compatível com processadores x86 de 64 bits desde setembro de 2006. Isso inclui uma ampla variedade de processadores com vários núcleos. Para obter uma lista completa dos processadores suportados, consulte o Guia de compatibilidade da VMware em <http://www.vmware.com/resources/compatibility>.

- O ESXi 6.7 requer que o bit NX / XD seja ativado para a CPU no BIOS.
- O ESXi 6.7 requer no mínimo 4 GB de RAM física. É recomendável fornecer pelo menos 8 GB de RAM para executar máquinas virtuais em ambientes típicos de produção.
- Para oferecer suporte a máquinas virtuais de 64 bits, o suporte à virtualização (Intel VT-x ou RVI AMD) deve estar ativado nas CPUs x64.
- Um ou mais controladores Gigabit ou Ethernet mais rápidos. Para obter uma lista dos modelos de adaptador de rede suportados, consulte o VMware Compatibility Guide em <http://www.vmware.com/resources/compatibility> .
- Disco SCSI ou um RAID LUN local sem rede com espaço não particionado para máquinas virtuais.
- Para Serial ATA (SATA), um disco conectado via controladores SAS compatíveis ou controladores SATA integrados compatíveis. Os discos SATA são considerados remotos, não locais. Esses discos não são usados como partição do zero por padrão, pois são considerados remotos.

O download do ESXi pode ser realizado através do link: <https://my.vmware.com/web/vmware/details?productId=757&downloadGroup=ESXi670>

Para instalação do ESXi seguiremos as imagens de 3.1.1.1.1 a 3.1.1.1.4.



Figura 3.1.1.1.1 - Instalação ESXi



Figura 3.1.1.1.2 - Instalação ESXi



Figura 3.1.1.1.3 - Instalação ESXi

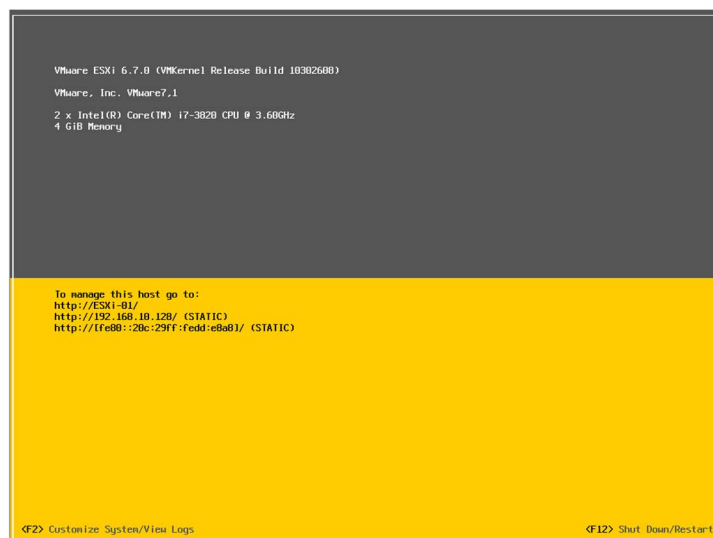


Figura 3.1.1.1.4 - Instalação ESXi

A máquina onde o ESXi irá ser instalado deverá ter pelo menos 03 discos, sendo o disco onde a VM será instalada espelhado via hardware para mitigação de riscos de falhas.

A memória RAM será de pelo menos 8 GB para atender os recursos da VM hospedada no ESXi.

No disco espelhado criaremos uma pasta chamada ISOS onde será armazenada ISO para instalação do Ubuntu (figura 3.1.1.1.5). O download pode ser feito através do link: <https://ubuntu.com/download/desktop/thank-you?version=18.04.4&architecture=amd64>

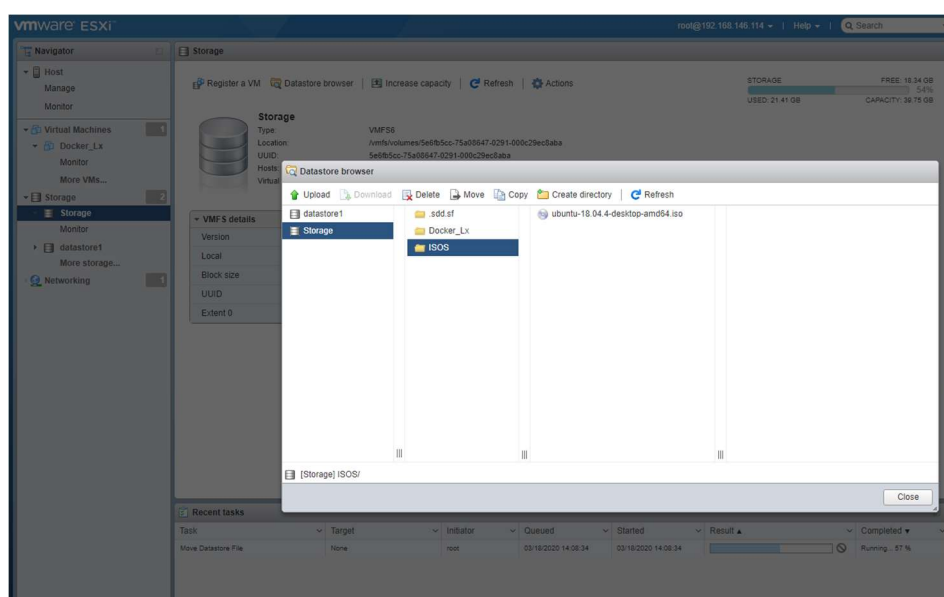


Figura 3.1.1.1.5 - ISO de Instalação do Ubuntu

Por não possuir os recursos necessários para demonstrar conforme a necessidade do ambiente para produção, usarei somente dois discos no ESXi, um para armazenamento do ESXi e outro para armazenar as ISOS e as VMs.

3.1.1.2 Instalação Criação da Máquina Virtual e Instalação do Ubuntu

Após realizada a instalação do ESXi, iniciaremos a instalação da máquina virtual através portal de gerenciamento do ESXi (figura 3.1.1.2.1) selecionado *Create/Register VM*.

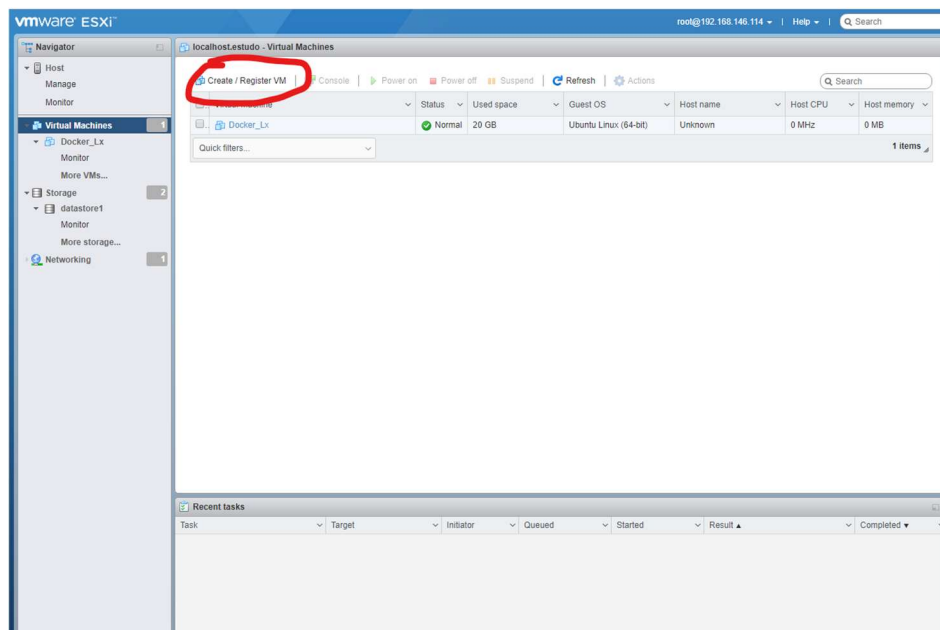


Figura 3.1.1.2.1 - Portal ESXi

A máquina virtual (VM) deverá seguir as configurações conforme a figura 3.1.1.2.2.

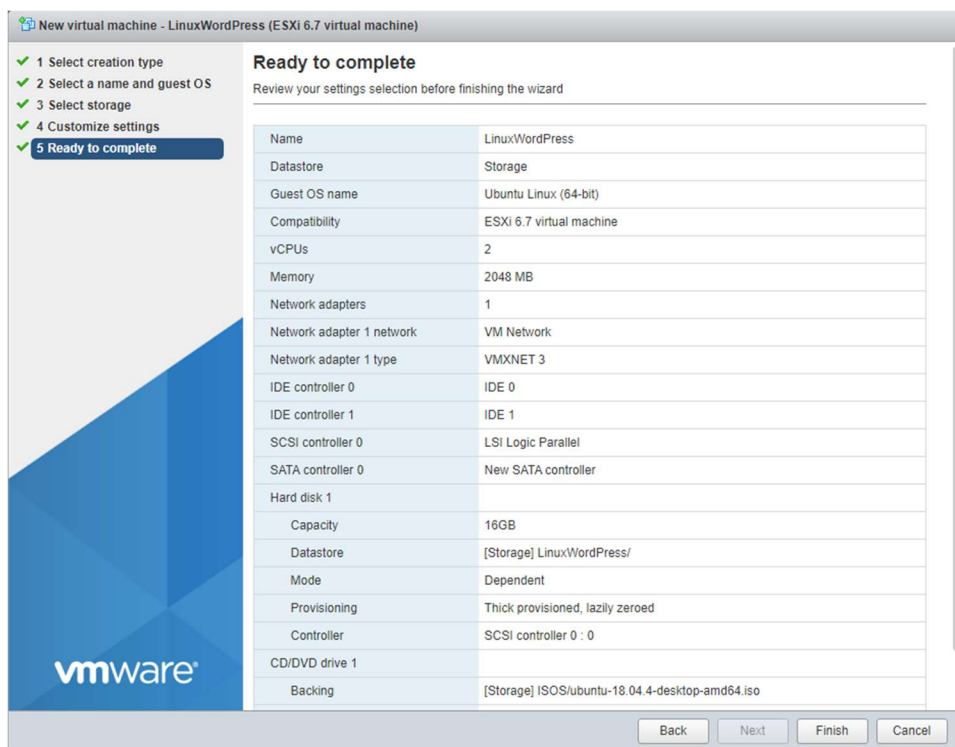


Figura 3.1.1.2.2 - Máquina Virtual

Após feita a criação da VM, está deverá ser inicializada e dado prosseguimento a instalação do Ubuntu.

3.1.2 Implantação da Aplicação

Para disponibilizar o WordPress será utilizado o Ansible e o Docker para automatização da instalação da aplicação.

3.1.2.1 Instalação do Ansible

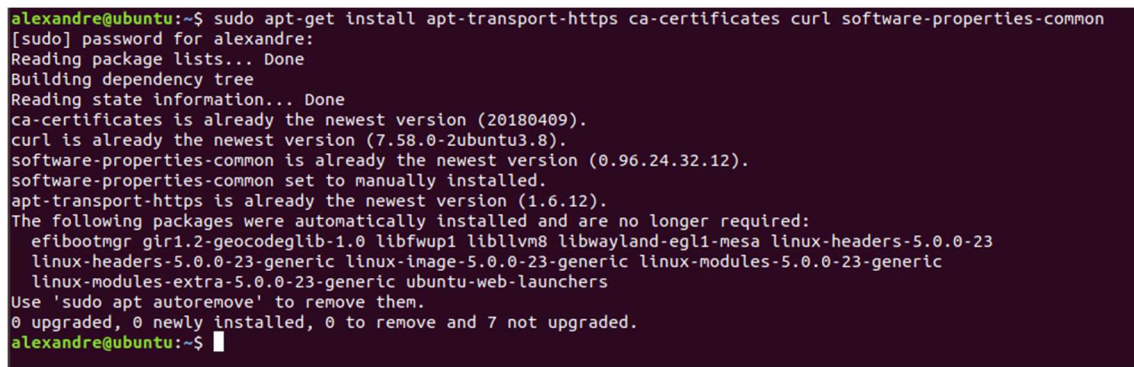
Para instalação do Ansible os seguintes comandos devem ser executados:

```
sudo apt-add-repository ppa:ansible/Ansible
sudo apt-get install software-properties-common
sudo apt-get update
sudo apt-get install ansible
```

3.1.2.2 Instalação do Docker

A plataforma Docker é hoje a plataforma mais popular do mercado.

Para instalação do Docker devemos seguir alguns comandos. Primeiro precisamos instalar os pacotes que irão executar as funções de assinatura digital e de pacotes necessários para o funcionamento de toda estrutura do Docker (figura 3.1.2.2.1).



```
alexandre@ubuntu:~$ sudo apt-get install apt-transport-https ca-certificates curl software-properties-common
[sudo] password for alexandre:
Reading package lists... Done
Building dependency tree
Reading state information... Done
ca-certificates is already the newest version (20180409).
curl is already the newest version (7.58.0-2ubuntu3.8).
software-properties-common is already the newest version (0.96.24.32.12).
software-properties-common set to manually installed.
apt-transport-https is already the newest version (1.6.12).
The following packages were automatically installed and are no longer required:
  efibootmgr gir1.2-geocodeglib-1.0 libfwup1 liblvm8 libwayland-egl1-mesa linux-headers-5.0.0-23
  linux-headers-5.0.0-23-generic linux-image-5.0.0-23-generic linux-modules-5.0.0-23-generic
  linux-modules-extra-5.0.0-23-generic ubuntu-web-launchers
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 7 not upgraded.
alexandre@ubuntu:~$
```

Figura 3.1.2.2.1 - Instalação pacotes

Na figura 3.1.2.2.2 o comando executa o download da chave de assinatura digital do projeto Docker, permitindo a autenticação dos programas a serem baixados e neste mesmo comando acionamos o gerenciador de pacotes. Dessa forma, tornamos possível autenticar os programas baixados do projeto Docker.


```
alexandre@ubuntu:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
OK
alexandre@ubuntu:~$
```

Figura 3.1.2.2.2 - Chave de assinatura e gerenciador de pacotes

O comando mostrado na figura 3.1.2.2.3 expõe a chave de identificação adicionada no comando anterior.

```
alexandre@ubuntu:~$ sudo apt-key fingerprint 0EBFCD88
pub  rsa4096 2017-02-22 [SCEA]
     9DC8 5822 9FC7 DD38 854A  E2D8 8D81 803C 0EBF CD88
uid  [ unknown] Docker Release (CE deb) <docker@docker.com>
sub  rsa4096 2017-02-22 [S]

alexandre@ubuntu:~$
```

Figura 3.1.2.2.3 - Chave de identificação

A figura 3.1.2.2.4 mostra a adição dos repositórios do projeto Docker. Dessa forma, será possível baixar e instalar os softwares disponíveis no projeto, inclusive o Docker.

```
alexandre@ubuntu:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable"
Hit:1 https://download.docker.com/linux/ubuntu bionic InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Hit:3 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:5 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelease
Reading package lists... Done
alexandre@ubuntu:~$
```

Figura 3.1.2.2.4 – Repositórios

Após esses comandos precisamos realizar o update (figura 3.1.2.2.5) para fazer o download da lista de pacotes disponíveis nos repositórios configurados. Esse comando é necessário pois assim poderemos receber a lista de pacotes disponíveis no repositório Docker que acabamos de adicionar.

```
alexandre@ubuntu:~$ sudo apt-get update
Hit:1 https://download.docker.com/linux/ubuntu bionic InRelease
Hit:2 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu bionic InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:5 http://us.archive.ubuntu.com/ubuntu bionic-backports InRelease
Reading package lists... Done
alexandre@ubuntu:~$
```

Figura 3.1.2.2.5 - Atualização de lista

Para concluir a instalação do Docker executaremos o comando conforme a figura 3.1.2.2.6.

```
alexandre@ubuntu:~$ sudo apt-get install docker-ce
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  efibootmgr gir1.2-geocodeglib-1.0 libfwup1 libllvm8 libwayland-egl1-mesa linux-headers-5.0.0-23
  linux-headers-5.0.0-23-generic linux-image-5.0.0-23-generic linux-modules-5.0.0-23-generic
  linux-modules-extra-5.0.0-23-generic ubuntu-web-launchers
Use 'sudo apt autoremove' to remove them.
Recommended packages:
  aufs-tools cgroupfs-mount | cgroup-lite git pigz
The following packages will be upgraded:
  docker-ce
1 upgraded, 0 newly installed, 0 to remove and 6 not upgraded.
Need to get 22.9 MB of archives.
After this operation, 23.6 kB of additional disk space will be used.
Get:1 https://download.docker.com/linux/ubuntu bionic/stable amd64 docker-ce amd64 5:19.03.8~3-0~ubuntu-bionic [2
2.9 MB]
Fetched 22.9 MB in 1s (22.8 MB/s)
(Reading database ... 199496 files and directories currently installed.)
Preparing to unpack .../docker-ce_5%3a19.03.8~3-0~ubuntu-bionic_amd64.deb ...
Unpacking docker-ce (5:19.03.8~3-0~ubuntu-bionic) over (5:19.03.7~3-0~ubuntu-bionic) ...
Setting up docker-ce (5:19.03.8~3-0~ubuntu-bionic) ...
Processing triggers for systemd (237-3ubuntu10.39) ...
Processing triggers for ureadahead (0.100.0-21) ...
alexandre@ubuntu:~$
```

Figura 3.1.2.2.6 - Instalação do Docker

3.1.2.3 Instalação dos Containers MySQL e WordPress

Para execução da aplicação distribuída iremos instalar os contêineres através do Docker usando para implantação o Ansible. Os contêineres instalados serão os seguintes:

- WordPress;
- MySQL.

Os arquivos para implantação do WordPress e SQL via Ansible estarão disponíveis no GitHub.

playbook.yml

```
---
- hosts: wpdocker
  remote_user: alexandre
  become: yes
```

```

tasks:
  - name: "Executa o container MySQL"
    docker_container:
      name: banco
      image: mysql:5.7
      env:
        MYSQL_ROOT_PASSWORD: senha123
  - name: "Executa o container WordPress"
    docker_container:
      name: wordpress
      image: wordpress
      links:
        - "banco:mysql"
      ports:
        - "8080:80"

```

<https://github.com/agbarbeito/wpdocker>

O Ansible pode trabalhar com vários nós gerenciados ou *hosts* em sua estrutura ao mesmo tempo, usando uma lista ou grupo de listas conhecidas como inventário. Na figura 3.1.2.3.1 temos a configuração do arquivo de inventário necessário para indicar onde será feita a instalação.



Figura 3.1.2.3.1 - Arquivo de Inventário Host

Para execução do playbook do Ansible o comando conforme a figura 3.1.2.3.2 deve ser executado.

```
alexandre@ubuntu:~/wpdocker$ ansible-playbook -i hosts playbook.yml

PLAY [wpdocker] *****

TASK [Gathering Facts] *****
[DEPRECATION WARNING]: Distribution Ubuntu 18.04 on host 127.0.0.1 should use /usr/bin/python3, but is using /usr/bin/python for backward compatibility with prior Ansible releases. A future Ansible release will default to using the discovered platform python for this host. See https://docs.ansible.com/ansible/2.9/reference_appendices/interpreter_discovery.html for more information. This feature will be removed in version 2.12. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
ok: [127.0.0.1]

TASK [Executa o container MySQL] *****
changed: [127.0.0.1]

TASK [Executa o container WordPress] *****
changed: [127.0.0.1]

PLAY RECAP *****
127.0.0.1 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
```

Figura 3.1.2.3.2 - Execução do playbook

Dessa forma, a aplicação irá responder na porta 8080 do host e está solicitação será reencaminhada para porta 80 do container, como mostrado na imagem 3.1.2.3.3.

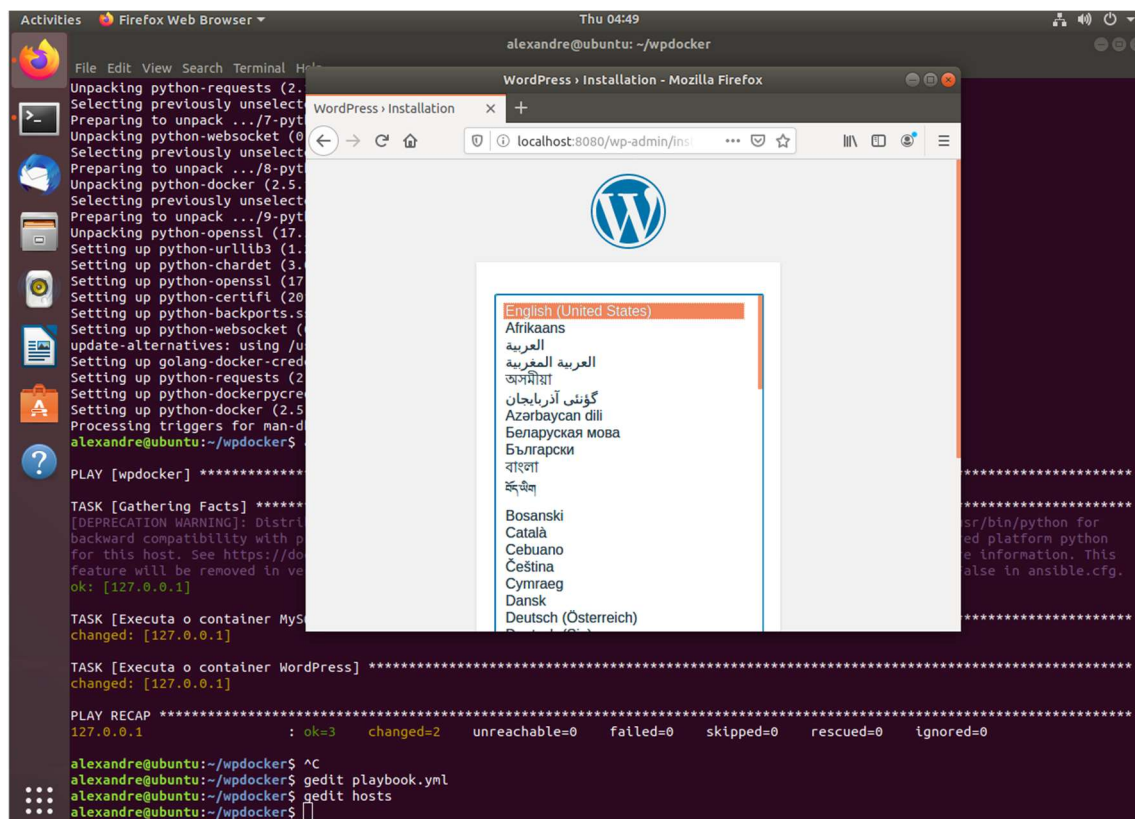


Figura 3.1.2.3.3 - Aplicação em Execução no Browser

3.2 Oportunidades de melhoria

O VMware ESXi é uma solução de fácil gerenciamento e que permite uma boa escalabilidade. A curto prazo a ONG pretende colocar mais um ESXi que vai permitir o melhor gerenciamento da aplicação podendo migrar a VM com as aplicações de um ESXi para outro em caso problema, necessidade de

manutenção ou upgrade de algum hardware utilizando o vMotion. O vMotion permite a migração da VM de um servidor físico para outro em tempo real sem interrupção dos serviços, de forma que os usuários da aplicação nem perceberam a movimentação.

Como forma de mitigação de riscos deve ser implantado uma solução de backup para máquinas virtuais, o espelhamento do disco e a configuração de um script para execução de commit dos *containers*.

Como solução de backup recomendamos utilização do Vinchin Backup & Recovery para VMware. Como características do Vinchin Backup & Recovery temos:

- Backup de VM quente, sem necessidade de desligar a VM durante o processo de backup.
- Backup de VM sem agente, sem necessidade de instalar nenhum agente no SO convidado.
- O backup sem LAN, melhora extremamente a velocidade de transmissão e reduz a influência no sistema do negócio.
- A tecnologia VMware CBT, captura rapidamente dados alterados, reduz a capacidade dos dados de backup.
- Recuperação instantânea de segundo nível, recupera toda a VM em 15 segundos.
- Agenda múltipla de backup inteligente, suporta backup completo, backup incremental e backup diferencial.
- Suporta diretiva de backup incremental permanente e política de retenção de backup.
- A deduplicação e compactação de dados que reduzem extremamente o espaço e os custos de armazenamento.

A utilização de snapshots das VMs também deverá ser utilizada como forma de garantir pontos de recuperação caso necessário.

O monitoramento das VM ocorrerá através das ferramentas do ESXi ou vSphere. Através de gráficos de desempenho para CPU, memória, E/S de disco, rede e armazenamento (figura 3.2.1).

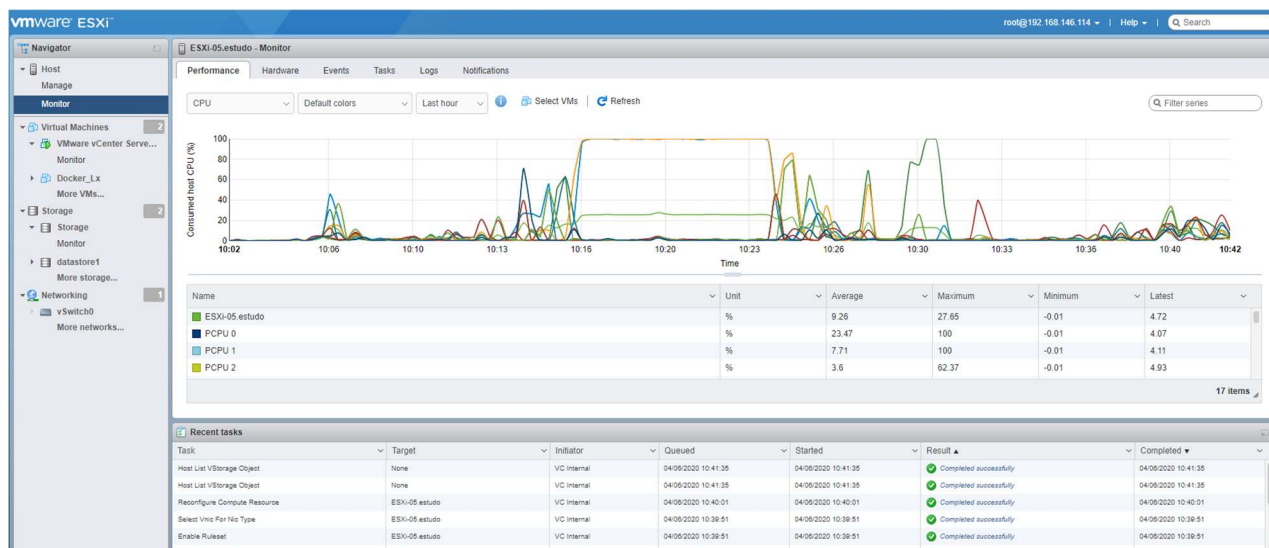


Figura 3.2.1 - Monitoramento

A VMware fornece várias ferramentas para monitorar o ambiente virtual e localizar a fonte de possíveis problemas. O Gráficos de desempenho permite ver dados de desempenho em uma variedade de recursos do sistema, incluindo CPU, Memória, Armazenamento e outros. Integridade do Host Permite identificar rapidamente quais hosts estão saudáveis e quais estão tendo problemas. Eventos, alertas e alarmes permitem configurar alertas e alarmes e especificar as ações que o sistema deve executar quando forem acionadas. Dessa forma é possível monitorar a saúde e o desempenho de dispositivos virtuais no ambiente virtual.

3.3 Cronograma

Partindo do ponto onde temos toda infraestrutura de hardware pronta e disponível, segue abaixo (figura 3.3.1) o cronograma para solução apresentada.

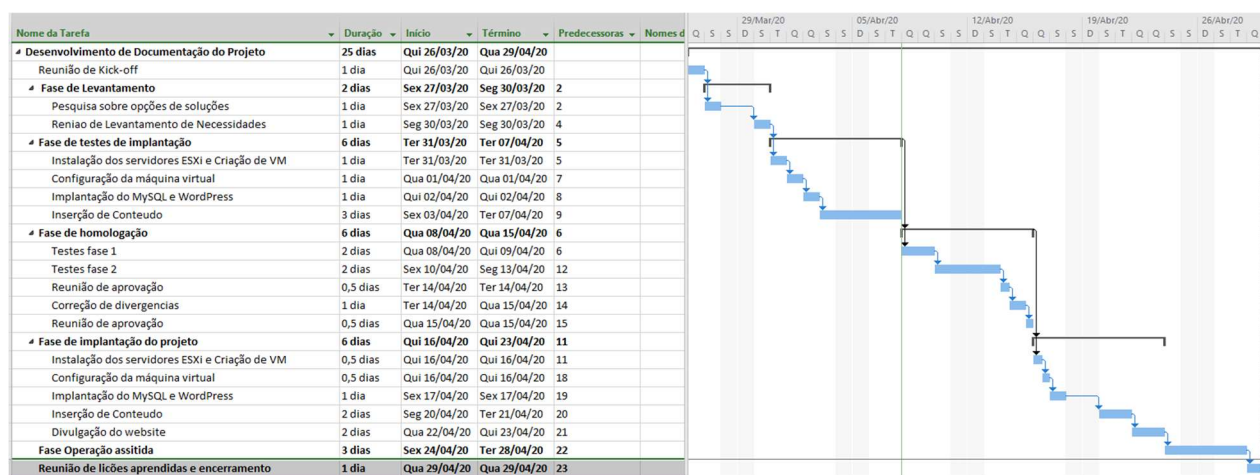


Figura 3.3.1 – Cronograma

O projeto terá uma duração total de 25 dias úteis, sendo dividido em cinco grandes fases e duas grandes reuniões.

A primeira reunião acontecerá no início do projeto e serão definidas as expectativas e premissas para o projeto. A segunda ocorrerá ao final do projeto e serão apresentadas as lições aprendidas e será realizado o encerramento do projeto.

A primeira fase, fase de levantamento, será realizada pesquisas sobre as opções de soluções e as necessidades para execução do projeto e levará 2 dias.

Na segunda fase, que durará 6 dias, será realizada uma fase de testes de implantação com instalação de servidores, máquinas virtuais, aplicações e inserção de conteúdo.

A homologação acontecerá na terceira fase. Nesse momento serão realizados os testes de execução e produção, a duração será de 6 dias. Os testes fase 1 serão realizados pela equipe de tecnologia da informação de maneira isolada. O teste fase 2 serão realizados com os mantenedores de conteúdo interno. Após os testes uma pequena reunião será realizada para aprovação e levantamento de possíveis problemas, caso problemas sejam encontrados, será reservado um período para correções e uma nova reunião para aprovação.

A quarta fase, consistira na implantação e divulgação do website.

E a última fase será uma operação assistida entre a equipe de TI e todos os clientes internos.

4 Conclusão

O desenvolvimento e implantação da aplicação foi orientado pela disponibilidade dos recursos na AMAZONIA UMA VISAO PARA TODOS. Foi utilizando os recursos disponíveis no momento que o desenvolvimento e implantação da solução aconteceu.

Como forma de mitigação de riscos é altamente recomendado que uma solução de Backup seja implantada, assim como a configuração de alertas de monitoramento no ESXi.

A instalação de um segundo ESXi dará maior escalabilidade e segurança.

A escassez de recursos não permitiu que outras funções fossem exploradas tais como a instalação do vSphere e a implantação da solução na nuvem (Ex: AWS), onde poderíamos utilizar recursos muito mais robustos como EC2 para disponibilizar capacidade computacional segura e redimensionável como uma infraestrutura altamente confiável e escalável. Com a utilização do EC2 na AWS não existiria a preocupação com hardware e o foco seria apenas o desenvolvimento da aplicação.

Referência Bibliográfica

- BARATA, J. Virtualização com o Hyper-V - Artigo Revista Infra Magazine. **DevMedia**, 2011. Disponível em: <<https://www.devmedia.com.br/virtualizacao-com-o-hyper-v-artigo-revista-infra-magazine-1/20821>>. Acesso em: 08 Abril 2020.
- CRAMPTON, C. How To Automate Installing WordPress on Ubuntu 14.04 Using Ansible. **Digitalocean**, 25 Junho 2015. Disponível em: <<https://www.digitalocean.com/community/tutorials/how-to-automate-installing-wordpress-on-ubuntu-14-04-using-ansible>>. Acesso em: 20 março 2020.
- JETHA, H. How to Set Up SSH Keys on Ubuntu 18.04. **Digitalocean**, 27 Abril 2018. Disponível em: <<https://www.digitalocean.com/community/tutorials/how-to-set-up-ssh-keys-on-ubuntu-1804>>. Acesso em: 20 março 2020.
- KUBERNETES. What is Kubernetes. **Kubernetes**, 16 Março 2020. Disponível em: <<https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/>>. Acesso em: 22 março 2020.
- LEE, B. Difference between VMware and Hyper-V: A Complete guide. **Backup & Disaster Recovery**, 2 Julho 2019. Disponível em: <<https://www.vembu.com/blog/hyper-v-vs-vmware/>>. Acesso em: 20 março 2020.
- LONG, J. Citrix XenServer vs VMware ESXi honest advice. **Spiceworks**, 28 Dezembro 2014. Disponível em: <<https://community.spiceworks.com/topic/709528-citrix-xenserver-vs-vmware-esxi-honest-advice>>. Acesso em: 27 março 2020.
- MICROSOFT. Supported Windows guest operating systems for Hyper-V on Windows Server. **Microsoft**, 2019. Disponível em: <<https://docs.microsoft.com/en-us/windows-server/virtualization/hyper-v/supported-windows-guest-operating-systems-for-hyper-v-on-windows>>. Acesso em: 20 Março 2020.
- MOURO, A. Microsoft Hyper-V 2019 vs. VMware vSphere 6.7. **vinfrastructure**, 31 Outubro 2019. Disponível em: <<https://vinfrastructure.it/2018/10/microsoft-hyper-v-2019-vs-vmware-vsphere-6-7>>. Acesso em: 27 março 2020.
- NEMETH, E.; SNTDER, G.; HEIN, T. R. **Manual Completo do Linux**. 2ª Edição. ed. São Paulo: Person Prentice Hall, 2007.
- NOVELLO, A. Arquitetura do Hyper-V. **André Novello**, 2017. Disponível em: <<https://andrenovello.wordpress.com/2017/01/28/arquitetura-do-hyper-v/>>. Acesso em: 08 Abril 2020.
- PHILLIPS, G. VirtualBox vs. VMWare vs. Hyper-V: What's the Best Virtual Machine? **Make Use Of**, 2019 Dezembro 2019. Disponível em: <<https://www.makeuseof.com/tag/virtualbox-vs-vmware-vs-hyper-v/>>. Acesso em: 27 março 2020.
- POSEY, B. vSphere vs. Hyper-V: Hypervisors Battle for the Enterprise. **Redmond magazine**, 04 Abril 2015. Disponível em: <<https://redmondmag.com/articles/2015/05/01/vsphere-vs-hyperv.aspx>>. Acesso em: 27 março 2020.
- POSEY, B. Virtual Infrastructures: Hyper-V vs VMware. **Solarwinds msp**, 29 Agosto 2017. Disponível em: <<https://www.solarwindmsp.com/blog/virtual-infrastructures-hyper-v-vs-vmware>>. Acesso em: 27 março 2020.

RED HAT ANSIBLE. Red hat ansible. **Red hat ansible**. Disponível em: <<https://www.ansible.com/>>. Acesso em: 08 Abril 2020.

REDHAT. Kubernetes e a tecnologia de containers. **Red Hat**, 26 Março 2020. Disponível em: <<https://www.redhat.com/pt-br/topics/containers/what-is-kubernetes>>. Acesso em: 24 março 2020.

REED, J. Hyper-V vs VMware: Which One to Choose? **Nakivo**, 26 Dezembro 2018. Disponível em: <<https://www.nakivo.com/blog/hyper-v-vmware-complete-comparison/>>. Acesso em: 22 março 2020.

ROUSE, M. vNUMA (virtual NUMA). **TechTarget**, 2013. Disponível em: <<https://searchvmware.techtarget.com/definition/vNUMA-virtual-NUMA>>. Acesso em: 20 Março 2020.

SIMPLILEARN. Citrix Xenserver Vs Vmware vSphere. **Simplilearn**, 03 Dezembro 2019. Disponível em: <<https://www.simplilearn.com/citrix-xenserver-vs-vmware-vsphere-rar338-article>>. Acesso em: 27 março 2020.

SINGH, R. How to Use an Ansible Playbook To Install WordPress on Ubuntu 18.04 and 16.04. **Dotlayer**, 22 Novembro 2018. Disponível em: <<https://dotlayer.com/how-to-use-an-ansible-playbook-to-install-wordpress>>. Acesso em: 18 março 2020.

TANENBAUM, A. S.; STEEN, M. V. **Sistemas Distribuídos**. 2ª Edição. ed. São Paulo: Pearson Prentice Hall, 2007.

TIA-942 GLOBAL REGISTRY. **TIA-942 Global Registry**. Disponível em: <<http://www.tia-942.org/>>. Acesso em: 22 Março 2020.

VINCHIN. Vinchin. **Vinchin**, 2020. Disponível em: <https://www.vinchin.com/en/res/pdf/Datasheet_Vinchin_Backup_&_Recovery_v5.0.pdf>. Acesso em: 06 Abril 2020.

WIKIPEDIA. Virtualização em nível de sistema operacional. **Wikipedia**, 13 Agosto 2019. Disponível em: <https://pt.wikipedia.org/wiki/Virtualiza%C3%A7%C3%A3o_em_n%C3%ADvel_de_sistema_operacional>. Acesso em: 27 março 2020.