

Nome: Numero:

ENGENHARIA INFORMÁTICA

Aplicações e Serviços de Computação em Nuvem

Teste

12 de Janeiro de 2022

Duração: 1h30min

1 A complexidade da replicação de um serviço multi-camada *não varia* de acordo com o componente alvo (p.ex., servidor web, servidor aplicacional, base de dados) a replicar. Indique e **justifique** se concorda ou não com esta afirmação.

Resposta: Discordo da afirmação.

A complexidade da replicação de um serviço multi-camada **varia** significativamente dependendo do componente alvo (como um servidor web, servidor aplicacional ou base de dados) devido às diferentes características e requisitos de cada tipo de componente.

Justificativa:

1. Servidores Web:

- **Menor Complexidade:** A replicação de servidores web geralmente é menos complexa, pois esses componentes tendem a ser **stateless** (sem estado). Cada instância de um servidor web pode atender requisições sem depender do estado mantido por outra instância. Por exemplo, replicar um servidor web como **Nginx** ou **Apache** envolve apenas replicar os arquivos de configuração e ativos estáticos.

2. Servidores Aplicacionais:

- **Complexidade Moderada:** Servidores aplicacionais podem ter **algum estado** ou **dependências** entre camadas, como interações com bancos de dados ou caches. Embora a replicação ainda seja viável, pode haver a necessidade de coordenar estados temporários ou sincronizar sessões do usuário (caso elas não sejam totalmente gerenciadas pelo cliente ou por caches externos).

3. Base de Dados:

- **Alta Complexidade:** A replicação de bases de dados é a mais complexa devido à necessidade de **consistência dos dados**. Bancos de dados geralmente mantêm estados persistentes e, ao replicá-los, é necessário escolher um modelo de replicação que equilibre **consistência, disponibilidade e tolerância a partições (CAP)**.
 - **Replicação Síncrona:** Garante consistência, mas pode impactar o desempenho.
 - **Replicação Assíncrona:** Oferece alta disponibilidade, mas corre o risco de inconsistências temporárias.
- Além disso, bases de dados podem necessitar de estratégias de particionamento (sharding) e mecanismos para lidar com conflitos em replicações bidirecionais.
- **Exemplo:** Replicar um banco de dados como **MySQL**, **PostgreSQL** ou **MongoDB** requer configuração e monitoramento cuidadosos para garantir que não haja perda de dados ou inconsistências.

Conclusão:

A complexidade da replicação **depende diretamente do componente** devido às diferenças em como cada um lida com estado e consistência. Servidores web tendem a ser os mais simples de replicar, enquanto bases de dados são os mais desafiadores, devido à sua natureza **stateful** e à necessidade de gerenciar **consistência e disponibilidade**.

Nome: Numero:

ENGENHARIA INFORMÁTICA

Aplicações e Serviços de Computação em Nuvem

Teste

12 de Janeiro de 2022

Duração: 1h30min

2 As infraestruturas de computação em nuvem promovem uma utilização eficiente de recursos computacionais, de rede e armazenamento através da utilização de virtualização. Descreva, e **justifique**, **duas vantagens** e **duas desvantagens** da utilização desta tecnologia.

Duas Vantagens da Virtualização:

1. Melhor Utilização de Recursos:

- **Descrição:** A virtualização permite que múltiplos sistemas operacionais e aplicações sejam executados em um único servidor físico, utilizando seus recursos de maneira mais eficiente.
- **Justificativa:** Em ambientes não virtualizados, os servidores frequentemente operam abaixo de sua capacidade total. Com a virtualização, os recursos não utilizados podem ser alocados dinamicamente para diferentes máquinas virtuais (VMs), reduzindo custos e aumentando a eficiência.

2. Flexibilidade e Escalabilidade:

- **Descrição:** A virtualização facilita a criação, configuração e migração de máquinas virtuais, permitindo escalabilidade e agilidade no provisionamento de recursos.
- **Justificativa:** Em cenários como expansão de serviços ou recuperação de desastres, máquinas virtuais podem ser replicadas ou movidas rapidamente entre servidores físicos, minimizando o impacto de falhas e permitindo ajustes dinâmicos à carga.

Duas Desvantagens da Virtualização:

1. Overhead de Desempenho:

- **Descrição:** A execução de múltiplas máquinas virtuais em um único hardware físico pode introduzir uma sobrecarga de recursos, resultando em desempenho inferior ao de sistemas não virtualizados.
- **Justificativa:** Cada máquina virtual consome uma fração da CPU, memória e armazenamento do host físico. Em cargas de trabalho intensivas, isso pode levar a gargalos e redução da performance global.

2. Complexidade na Gestão:

- **Descrição:** A virtualização adiciona uma camada de abstração que pode complicar o gerenciamento do ambiente, especialmente em infraestruturas grandes.
- **Justificativa:** Problemas como shadow IT (recursos não documentados), sprawl (crescimento descontrolado de VMs) e dependência de hipervisores tornam a administração mais desafiadora, exigindo ferramentas avançadas e equipes especializadas para manter a eficiência e a segurança.

Nome: Numero:

ENGENHARIA INFORMÁTICA

Aplicações e Serviços de Computação em Nuvem

Teste

12 de Janeiro de 2022

Duração: 1h30min

3 Nas aulas práticas recorreu às ferramentas *ElasticSearch*, *Kibana* e *MetricBeat* para observar a utilização de recursos de diferentes máquinas virtuais. Indique qual a função de cada uma destas ferramentas num ambiente de monitorização.

1. ElasticSearch:

- **Função:** É a ferramenta de **armazenamento e indexação** dos dados coletados para monitorização.
- **Detalhes:** ElasticSearch armazena os logs, métricas e outros dados enviados por ferramentas como o MetricBeat. Ele é projetado para **processar grandes volumes de dados** e permitir **pesquisas rápidas** e complexas por meio de consultas estruturadas. No contexto de monitorização, ElasticSearch organiza e disponibiliza os dados para serem analisados ou visualizados.

2. Kibana:

- **Função:** É a interface de **visualização e análise** dos dados armazenados no ElasticSearch.
- **Detalhes:** Kibana permite criar **dashboards interativos**, gráficos, e tabelas para interpretar visualmente as métricas e logs. Ele ajuda os usuários a identificar tendências, anomalias ou problemas relacionados à utilização de recursos em máquinas virtuais, de maneira intuitiva.

3. MetricBeat:

- **Função:** É a ferramenta de **coleta e envio de métricas** sobre a utilização de recursos.
- **Detalhes:** MetricBeat monitora o uso de CPU, memória, disco, rede e outros parâmetros nas máquinas virtuais e envia essas métricas para o ElasticSearch. Ele atua como o agente responsável por capturar os dados em tempo real e disponibilizá-los para análise.

Resumo da Interação:

- **MetricBeat** coleta as métricas de utilização das máquinas.
- **ElasticSearch** armazena e organiza essas métricas.
- **Kibana** apresenta os dados visualmente, permitindo análises mais acessíveis e rápidas.

Nome: Numero:

ENGENHARIA INFORMÁTICA

Aplicações e Serviços de Computação em Nuvem

Teste

12 de Janeiro de 2022

Duração: 1h30min

4 Imagine que a Universidade do Minho lhe atribui a responsabilidade de instalar a aplicação WikiJS para servir todos os alunos da universidade. No entanto, antes do processo de instalação, o gestor financeiro da universidade pergunta-lhe quais os recursos computacionais que prevê serem necessários para esta instalação. Que estratégia(s) pode aplicar para responder a esta pergunta e ter um elevado grau de confiança que a aplicação será capaz de responder à carga computacional imposta em produção?

1. Realizar Testes de Benchmarking e Simulação de Carga

- **Descrição:** Configure um ambiente de pré-produção que replica o mais próximo possível o ambiente esperado em produção. Utilize ferramentas de benchmarking como **Apache JMeter**, **K6** ou **Locust** para simular a carga real que será imposta pela comunidade de alunos.
- **Como aplicar:**
 1. Configure a aplicação WikiJS com uma infraestrutura inicial básica.
 2. Simule cenários típicos de uso, como consultas simultâneas, edições de páginas, ou uploads.
 3. Meça o consumo de CPU, memória, disco e rede sob diferentes níveis de carga.
- **Vantagem:** Os resultados indicam claramente como a aplicação responde a diferentes volumes de acessos e qual é o ponto de saturação dos recursos.

2. Análise de Casos de Uso e Dimensionamento com Base em Métricas de Aplicações Similares

- **Descrição:** Pesquise implementações de **WikiJS** ou aplicações semelhantes em instituições ou organizações com cargas de trabalho comparáveis.
- **Como aplicar:**
 1. Consulte a documentação do WikiJS sobre requisitos mínimos e recomendados.
 2. Compare o número de usuários esperados na Universidade do Minho com os casos relatados.
 3. Utilize esses dados para prever os requisitos computacionais.
- **Vantagem:** Reduz incertezas ao se basear em experiências práticas de cenários semelhantes.

4. Escalabilidade Vertical e Horizontal

- **Descrição:** Dimensione os recursos iniciais considerando uma infraestrutura que permita **escalabilidade vertical** (aumento de recursos em uma máquina) e **horizontal** (adição de mais servidores).
- **Como aplicar:**
 1. Configure a aplicação em um ambiente virtualizado ou em contêineres (Docker/Kubernetes) que permita redimensionamento dinâmico.
 2. Comece com uma configuração básica e adicione recursos conforme necessário, com base nos resultados dos testes.
- **Vantagem:** Reduz o risco de subdimensionamento ou desperdício de recursos, mantendo flexibilidade para adaptar-se à carga real.

isto nao é bem pre instalação

Conclusão:

Uma abordagem eficiente combina **testes práticos (benchmarking e piloto)** com **análise de casos semelhantes e planejamento para escalabilidade**. Isso garante um dimensionamento inicial confiável e flexível, permitindo que a aplicação suporte a carga de trabalho real sem falhas ou desperdício de recursos.