**🔒 O que é o modelo de responsabilidade compartilhada?**

É um acordo “implícito” entre você (cliente) e o **provedor de nuvem** (AWS, Azure, Google Cloud etc.) sobre **quem cuida de quê** em relação à **segurança e funcionamento** da infraestrutura de nuvem.

**🏢 Data Center tradicional (tudo local)**

* Você é dono do prédio (data center).
* Você paga energia, refrigeração, espaço físico, segurança contra invasão.
* Seu time de TI cuida dos servidores, da rede, do sistema operacional, das aplicações e dos dados.  
  👉 Ou seja: **100% da responsabilidade é sua**.

**☁️ Na Nuvem**

O provedor tira de você uma parte da carga (principalmente a parte física) e divide as obrigações:

* **Provedor de Nuvem SEMPRE cuida de:**
  + Segurança física (predio/data center)
  + Energia, refrigeração e rede física
  + Servidores físicos (hardware)
* **Você (cliente) SEMPRE cuida de:**
  + Seus **dados** (não importa onde estejam)
  + Quem pode acessar (usuários, senhas, permissões, autenticação multifator)
  + Seus dispositivos (notebook, celular, etc.) que acessam a nuvem

**📊 Como varia de acordo com o tipo de serviço?**

1. **IaaS (Infraestrutura como Serviço)**  
   Você aluga a “estrutura básica” (máquinas virtuais, rede, armazenamento).
   * Provedor: cuida do hardware, rede e data center.
   * Você: cuida do sistema operacional, patches, firewall, aplicações e dados.  
     👉 É o modelo mais parecido com o data center tradicional.
2. **PaaS (Plataforma como Serviço)**  
   Você aluga não só a máquina, mas também um **ambiente pronto** (ex: um banco de dados gerenciado, um servidor de aplicação).
   * Provedor: cuida da infraestrutura + sistema operacional + atualizações da plataforma.
   * Você: só cuida das aplicações que constrói e dos dados.  
     👉 Divisão mais equilibrada.
3. **SaaS (Software como Serviço)**  
   Você só usa o software (ex: Outlook online, Google Drive, Salesforce).
   * Provedor: cuida de praticamente tudo (infra, OS, aplicativo, atualizações).
   * Você: só cuida de gerenciar os usuários e os dados que coloca lá.  
     👉 É o modelo mais “tranquilo” pro consumidor.

**📌 Exemplo prático:**

* **IaaS**: Você cria uma máquina virtual na nuvem e instala um SQL Server → você precisa atualizar o Windows, aplicar patch no SQL e cuidar dos dados.
* **PaaS**: Você contrata um serviço de SQL pronto (Azure SQL Database, por exemplo) → o provedor atualiza o banco, mas os dados são sua responsabilidade.
* **SaaS**: Você usa o Office 365 → a Microsoft cuida de tudo, você só cuida dos usuários e documentos que salva.

👉 Em resumo:

* **Quanto mais “baixo nível” o serviço (IaaS), mais responsabilidade é sua.**
* **Quanto mais “alto nível” (SaaS), mais responsabilidade é do provedor.**

**⚡ O que é o modelo baseado em consumo?**

É o jeito que os provedores de nuvem (Azure, AWS, Google Cloud etc.) **cobram e entregam recursos**:  
👉 você paga **somente pelo que usar**, em vez de comprar ou alugar uma estrutura fixa.

É como a conta de luz ou de água:

* Se você usar pouco, paga pouco.
* Se usar muito, paga mais.
* Não precisa comprar a usina de energia (infraestrutura), só pagar pelo consumo.

**🔑 Características principais:**

1. **Escalabilidade**: você pode aumentar ou reduzir a capacidade (CPU, memória, armazenamento) conforme a necessidade, em tempo real.
2. **Sem grandes investimentos iniciais**: diferente de comprar servidores caros, você começa gastando pouco.
3. **Pagamento sob demanda**: o custo é proporcional ao uso (horas de máquina virtual, quantidade de dados armazenados, tráfego de rede etc.).
4. **Transparência de custos**: você consegue prever e controlar quanto está gastando com relatórios detalhados.

**🏢 Comparando com o tradicional:**

* **Modelo tradicional (on-premises)**: você compra servidores para 5 anos, mesmo que use só 30% da capacidade.
* **Modelo baseado em consumo (cloud)**: você paga apenas pelos 30% que realmente usou naquele mês.

**📌 Exemplo prático:**

* Sua empresa precisa rodar um site para um evento que dura **1 semana**.
  + **Antes**: teria que comprar servidores físicos que ficariam parados depois do evento.
  + **Na nuvem**: você contrata recursos só por aquela semana → terminou o evento → desliga → para de pagar.

👉 Em resumo:  
O **modelo baseado em consumo** é o que torna a nuvem tão atraente, porque elimina o desperdício de recursos, dá flexibilidade e reduz custos.

## 📊 Comparação de Custos – On-premises x Nuvem (consumo)

| **Aspecto** | **Data Center Tradicional (On-premises)** | **Nuvem (Modelo de Consumo)** |
| --- | --- | --- |
| **Investimento inicial** | Alto 💰 (compra de servidores, rede, espaço físico, energia, refrigeração) | Zero ou quase zero (sem compra de hardware, só ativar serviço) |
| **Escalabilidade** | Limitada (se precisar de mais recursos, tem que comprar novos servidores → demora semanas/meses) | Imediata (aumenta CPU, memória ou armazenamento em minutos) |
| **Custos fixos** | Sempre altos (mesmo que use pouco, continua pagando energia, manutenção, pessoal de TI) | Proporcionais ao uso (paga só pelo que consumir, pode desligar quando não precisa) |
| **Manutenção** | Responsabilidade da empresa (troca de peças, atualização de hardware, suporte físico) | Responsabilidade do provedor de nuvem |
| **Tempo de uso** | Precisa justificar investimento para 3-5 anos de vida útil do servidor | Pode contratar por minutos, horas, dias ou anos |
| **Exemplo prático** | Comprar 5 servidores para aguentar picos de demanda, mesmo que fiquem 70% ociosos | Pagar apenas pela capacidade extra durante o pico → desligar depois |

### 📌 Cenário numérico simples

* **On-premises**: compra de 5 servidores a R$ 50.000 cada = **R$ 250.000** de investimento inicial + ~R$ 10.000/mês em energia e manutenção.
* **Nuvem**: paga **R$ 15.000/mês** só pelo que usar. Se desligar parte dos recursos em períodos de baixa demanda, pode cair para R$ 8.000/mês.

👉 Em **2 anos**, a diferença de custo pode ser brutal:

* On-premises ≈ R$ 250.000 + (24 × 10.000) = **R$ 490.000**
* Nuvem ≈ média de R$ 12.000 × 24 = **R$ 288.000**

⚡ Ou seja:

* **On-premises** → custo fixo alto, pouca flexibilidade.
* **Nuvem** → custo variável, flexibilidade total.

# Resumo

Concluído100 XP

* 2 minutos

Neste módulo, você aprendeu sobre conceitos gerais de nuvem. Você começou com conceitos como, por exemplo, entender o que é computação em nuvem. Você também aprendeu sobre o modelo de responsabilidade compartilhada e como você e seu provedor de nuvem compartilham a responsabilidade de manter suas informações seguras na nuvem. Você abordou brevemente as diferenças entre os modelos de nuvem (pública, privada, híbrida e multicloud). Em seguida, você concluiu com uma unidade sobre como a nuvem transforma os gastos de TI de uma despesa de capital para uma despesa operacional.

## Objetivos de aprendizagem

Agora você deve ser capaz de:

* Defina computação em nuvem.
* Descreva o modelo de responsabilidade compartilhada.
* Defina modelos de nuvem, incluindo pública, privada e híbrida.
* Identifique casos de uso apropriados para cada modelo de nuvem.
* Descreva o modelo baseado no consumo.
* Compare modelos de preços de nuvem.

## Recursos adicionais

Os seguintes recursos fornecem mais informações sobre tópicos deste módulo ou relacionados a este módulo.

* [Modelo de responsabilidade compartilhada](https://learn.microsoft.com/en-us/azure/security/fundamentals/shared-responsibility) O modelo de responsabilidade compartilhada é o compartilhamento de responsabilidades pela nuvem entre você e seu provedor de nuvem.
* [Introdução ao Azure VMware Solution](https://learn.microsoft.com/en-us/learn/modules/intro-azure-vmware-solution/) é um curso do Microsoft Learn que se aprofunda no Azure VMware Solution.
* [Introdução aos serviços de nuvem híbrida do Azure](https://learn.microsoft.com/en-us/learn/modules/intro-to-azure-hybrid-services/) é um curso do Microsoft Learn que explica a nuvem híbrida com mais detalhes.

## 🌐 **Alta Disponibilidade (High Availability – HA)**

**Benefício principal:** manter os serviços **sempre disponíveis**, mesmo diante de falhas ou imprevistos.

🔑 **Vantagens**:

* **Menos tempo de inatividade (downtime):** garante que os sistemas fiquem online praticamente o tempo todo.
* **Confiabilidade:** usuários conseguem acessar o app/serviço quando precisarem.
* **Redundância automática:** se um servidor falhar, outro assume.
* **SLAs (Acordos de Nível de Serviço):** os provedores de nuvem oferecem garantias de disponibilidade (ex: 99,9%).

👉 Exemplo: se uma máquina virtual cair, outra em outra região pode assumir, mantendo o site funcionando sem o usuário perceber.

## 📈 **Escalabilidade**

**Benefício principal:** ajustar recursos **conforme a demanda**, sem precisar comprar infraestrutura fixa.

🔑 **Vantagens**:

* **Flexibilidade:** aumenta ou reduz capacidade de processamento, memória ou armazenamento de acordo com a necessidade.
* **Economia de custos:** você só paga pelo que usa (modelo de consumo).
* **Atende picos de demanda:** se seu site receber muitos acessos de repente, os recursos aumentam automaticamente para evitar travamentos.
* **Escalabilidade sob demanda:** você pode crescer ou reduzir em tempo real.

### Tipos de escalabilidade:

* **Vertical (scale up/down):** aumentar ou diminuir a capacidade de um recurso existente.
  + Ex: adicionar mais CPUs e RAM a uma VM.
* **Horizontal (scale out/in):** aumentar ou diminuir a quantidade de recursos.
  + Ex: adicionar várias máquinas virtuais ou instâncias de contêineres para dividir a carga.

## 🚀 Resumindo:

* **Alta disponibilidade:** garante que seu serviço **não caia**.
* **Escalabilidade:** garante que seu serviço **aguente crescer ou reduzir** conforme a necessidade, sem desperdício de recursos.

## 🔒 **Confiabilidade na nuvem**

👉 É a **capacidade de um sistema continuar funcionando mesmo diante de falhas**.

**Benefícios:**

* **Resiliência**: se um servidor ou região cair, outro assume.
* **Design descentralizado**: recursos distribuídos em várias regiões do mundo.
* **Failover automático**: em muitos casos, a nuvem transfere seu ambiente para outra região sem intervenção manual.
* **Experiência contínua**: o usuário final quase nunca percebe interrupções.

📌 Exemplo: Um site hospedado no Azure pode continuar funcionando mesmo que o data center de São Paulo caia, porque ele redireciona para outra região disponível.

## 📊 **Previsibilidade na nuvem**

👉 É a **capacidade de prever desempenho e custos**, dando confiança para planejar soluções.

### 🔹 ****Previsibilidade de desempenho****

* **Escalonamento automático**: aumenta recursos em picos e reduz quando a demanda cai.
* **Balanceamento de carga**: distribui tráfego entre servidores para evitar sobrecarga.
* **Alta disponibilidade**: garante consistência na experiência do usuário.

📌 Exemplo: Se sua loja virtual tiver muito acesso numa Black Friday, a nuvem aumenta a capacidade automaticamente, evitando lentidão.

### 🔹 ****Previsibilidade de custos****

* **Monitoramento em tempo real**: acompanhamento detalhado do uso de recursos.
* **Ferramentas de cálculo**: como TCO e Calculadora de Preços, para prever gastos.
* **Eficiência financeira**: você paga só pelo que usa, podendo ajustar para evitar desperdícios.

📌 Exemplo: uma empresa pode simular quanto gastaria migrando seus servidores locais para a nuvem antes de decidir, evitando surpresas.

## 🚀 Resumindo:

* **Confiabilidade** → garante que o sistema **não pare** mesmo em falhas.
* **Previsibilidade** → garante que você consiga **planejar e controlar** tanto o desempenho quanto os custos.

## ⚙️ **Benefícios da Capacidade de Gerenciamento na Nuvem**

### 🔹 1. ****Gestão da nuvem (Cloud Management)****

É sobre **como os recursos da nuvem são administrados automaticamente**.

**Benefícios:**

* **Escalonamento automático:** ajusta recursos de acordo com a demanda (sem precisar de intervenção manual).
* **Modelos pré-configurados:** facilita a criação de ambientes prontos (infra como código).
* **Monitoramento contínuo:** acompanha a saúde dos recursos em tempo real.
* **Autocorreção:** substitui automaticamente recursos com falhas.
* **Alertas automáticos:** notifica sobre desempenho, custos ou problemas antes que virem crises.

📌 Exemplo: se uma máquina virtual falhar, outra pode ser criada automaticamente sem precisar de um técnico agindo na hora.

### 🔹 2. ****Gestão na nuvem (Management in the Cloud)****

É sobre **como você, como administrador, acessa e controla seus recursos**.

**Benefícios:**

* **Portal Web:** interface gráfica amigável e acessível de qualquer navegador.
* **CLI (Command Line Interface):** controle via comandos rápidos e scripts.
* **APIs:** integração com outros sistemas ou automação avançada.
* **PowerShell:** automação de tarefas em massa, muito usado em ambientes Microsoft.

📌 Exemplo: você pode criar 10 máquinas virtuais em segundos com um comando no PowerShell ou clicar em alguns botões no portal web.

## 🚀 Resumindo:

* **Gestão da nuvem:** benefícios automáticos (autoescala, monitoramento, alertas, autocorreção).
* **Gestão na nuvem:** flexibilidade de ferramentas para administrar seus recursos (portal, CLI, APIs, PowerShell).

# Resumo

Concluído100 XP

* 2 minutos

Neste módulo, você aprendeu sobre alguns dos benefícios de operar na nuvem. Você aprendeu sobre alta disponibilidade e confiabilidade e como elas funcionam para manter seus aplicativos em execução. Você também aprendeu como a nuvem pode fornecer um ambiente mais seguro. Por fim, você aprendeu que a nuvem proporciona um ambiente altamente gerenciável para seus recursos.

## Objetivos de aprendizagem

Agora você deve ser capaz de:

* Descreva os benefícios da alta disponibilidade e escalabilidade na nuvem.
* Descreva os benefícios da confiabilidade e previsibilidade na nuvem.
* Descreva os benefícios da segurança e governança na nuvem.
* Descreva os benefícios da gerenciabilidade na nuvem.

## Recursos adicionais

Os seguintes recursos fornecem mais informações sobre tópicos deste módulo ou relacionados a este módulo.

* [Crie ótimas soluções com o Microsoft Azure Well-Architected Framework](https://learn.microsoft.com/en-us/learn/paths/azure-well-architected-framework/) é um curso do Microsoft Learn que apresenta o Microsoft Azure Well-Architected Framework.