

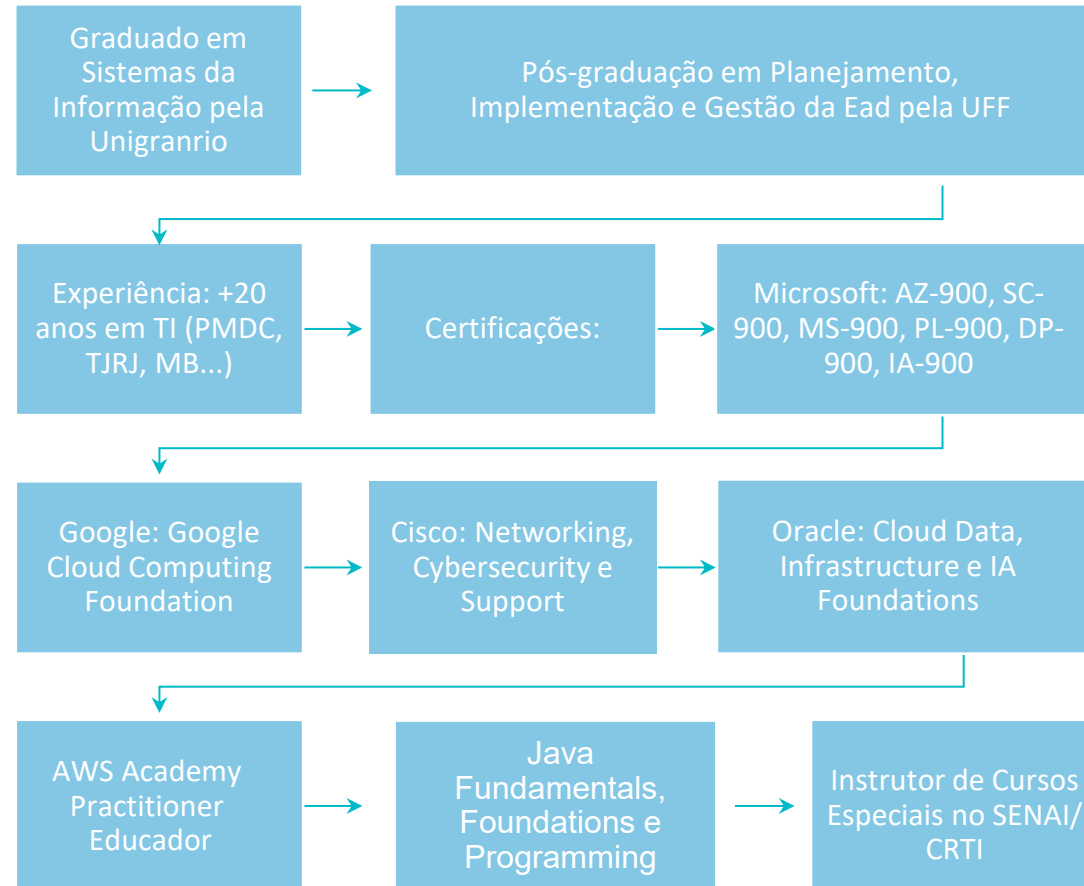
Do Push ao Deploy:

Publicando seu projeto com GitHub e Microsoft Azure AZ-900





Quem é Raphael Barreto?



Programação

Parte 1 (18h às 20h)	Parte 2 (20h30min às 22h)
Sessão 1: Análise de mercado e perspectivas profissionais na área de computação em nuvem.	Sessão 6: Versionamento de aplicações com GitHub Desktop
Sessão 2: Conceitos básicos de nuvem.	Sessão 7: Deploy da aplicação na nuvem com PaaS
Sessão 3: Arquitetura do Azure e serviços de computação do Azure	Sessão 8: Perguntas rápidas
Sessão 4: Criando sua conta gratuita no Azure e explorando o portal	
Sessão 5: Repositório com GitHub	

Material do Workshop: <https://dontpad.com/workshop>

Análise de mercado

Vamos agora entender por que investir em aprendizado sobre nuvem é uma excelente escolha hoje. A área de tecnologia está em plena transformação, e a computação em nuvem é um dos pilares principais dessa nova fase.

Independente de você ser desenvolvedor, analista de dados, administrador de redes, ou mesmo um profissional em transição de carreira, saber usar serviços de nuvem como o Microsoft Azure, AWS ou Google Cloud te coloca em uma posição de vantagem competitiva.

Mas... será que isso é só um discurso bonito?
Vamos aos números!

Análise de vagas no LinkedIn



70%

SQL

Recentemente, um especialista chamado **Paulo Schmitz Junior** fez um levantamento muito interessante. Ele coletou e analisou mais **de 7.000 vagas** da área de dados no **LinkedIn**, ao longo **de 30 dias**.

65%

Python

Veja que **cloud computing** já está presente na análise de vagas e que Azure aparece como uma outra habilidade destacada.

40%

Cloud

Ou seja, é um pré-requisito em crescimento acelerado.

40%

Power BI

Outro dado curioso, para vagas de Analista de Dados, o **Python** já aparece mais que o Excel. Isso mostra como a tecnologia está migrando para algo mais com código e infraestrutura.

Oportunidades no Mercado de Cloud no Brasil

Outro ponto interessante: As vagas presenciais ainda são maioria, mas, quando falamos de cargos que operam em nuvem, o cenário se inverte: **vagas remotas** dominam para **especialistas em cloud, dados e segurança**.

E quando analisamos as plataformas de nuvem mais citadas nas vagas, vemos que o **Microsoft Azure** está praticamente empatado com a **AWS**. Na verdade, grandes empresas do **setor financeiro, público e corporativo no Brasil usam Azure fortemente** - até por conta de compliance e integração com ferramentas Microsoft.

[Acesse aqui o Dashboard!](#)

Revisão da Sessão 01

- ✓ Análise de mercado
- ✓ perspectivas profissionais na área de computação em nuvem.

Conceitos básicos de nuvem

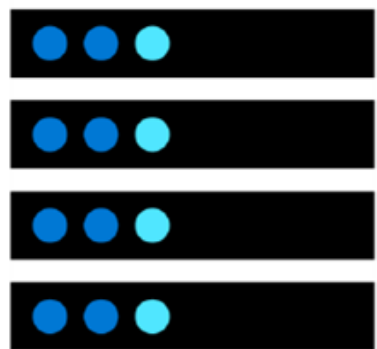




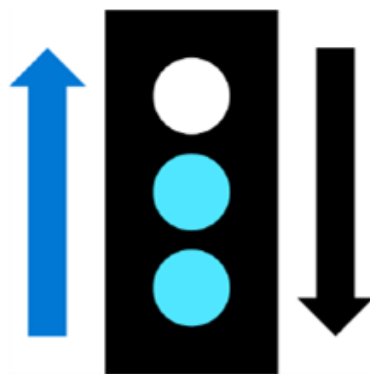
Objetivo de aprendizagem: Computação na nuvem

O que é computação na nuvem?

A **computação na nuvem** é a entrega de serviços de computação pela Internet, permitindo inovações mais rápidas, recursos flexíveis e economias de escala.



Computação



Rede



Armazenamento

Computação: CPU e memória sob demanda, escaláveis.

Rede: conexão entre recursos, sub-redes, IPs e segurança.

Armazenamento: dados e discos virtuais na nuvem, com foco em soberania e conformidade.

Nuvem Privada

A **nuvem privada** é exclusiva de uma organização. Os recursos de computação são hospedados no data center da própria empresa e acessados apenas internamente.

É um ambiente **fechado**, que oferece maior controle e segurança, mas exige mais investimentos em infraestrutura e gerenciamento.



Nuvem Pública

A **nuvem pública** é operada por um provedor de serviços (como Microsoft, Google ou AWS) e disponibiliza recursos e serviços de computação para múltiplos clientes via conexão segura pela internet.

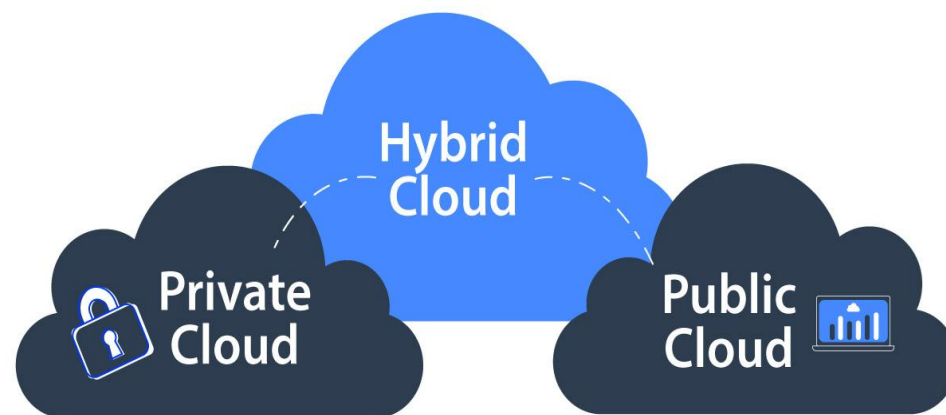
É um modelo **compartilhado**, escalável e de fácil acesso, ideal para organizações que buscam agilidade e redução de custos com infraestrutura própria.



Nuvem Híbrida

A **nuvem híbrida** combina recursos da nuvem pública e da nuvem privada, permitindo executar aplicações no ambiente mais adequado conforme necessidades de desempenho, segurança ou conformidade.

Esse modelo oferece **flexibilidade e otimização de custos**, ao mesmo tempo em que exige **habilidade técnica** para integrar e gerenciar os dois ambientes de forma eficiente.



Comparação de modelos de nuvem

Nuvem pública

Sem despesas de capital (CapEx); provisionamento rápido; pagamento por uso; responsabilidade compartilhada.

Nuvem privada

Exige compra e manutenção de hardware; controle total de segurança; previsibilidade de crescimento; gestão própria de atualizações.

Nuvem Híbrida

Combina vantagens da pública e privada; maior flexibilidade; permite alocar aplicações conforme requisitos técnicos, legais e de conformidade. |

Comparação de CapEx vs. OpEx

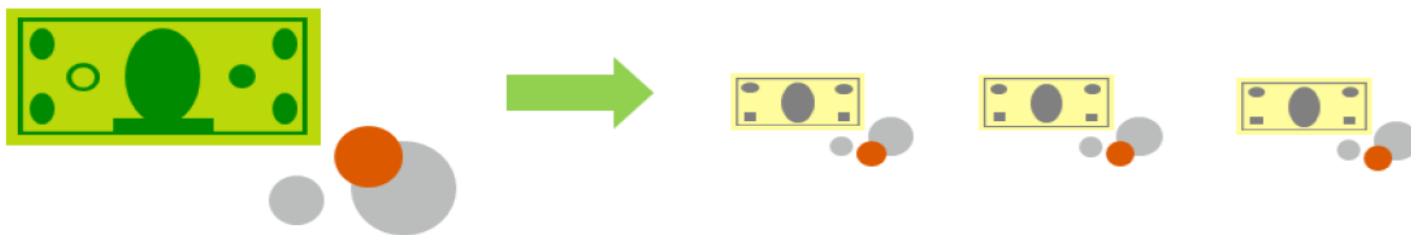
CapEx (Despesa de Capital):

Investimento antecipado em infraestrutura física; custo alto inicial com depreciação ao longo do tempo; dedução fiscal distribuída por anos; exige planejamento de longo prazo.

OpEx (Despesa Operacional):

Pagamento por uso sob demanda; sem custo inicial; dedução fiscal imediata no mesmo ano; ideal para escalabilidade e flexibilidade de curto prazo.

Esse comparativo é essencial para entender os impactos financeiros ao adotar soluções em nuvem.



Modelo com base no consumo

O modelo **baseado no consumo** permite pagar apenas pelos recursos utilizados, sem necessidade de investir em infraestrutura própria.

Vantagens:

- Sem custos iniciais
- Escalabilidade sob demanda
- Ideal para projetos de Pesquisa e Desenvolvimento, IA e ambientes de alta variabilidade
- Possibilidade de pausar ou desligar recursos quando não forem mais necessários
- Evita desperdício e otimiza o uso de recursos



Esse modelo difere de simplesmente “hospedar” algo: ele oferece **acesso sob demanda a recursos de alto valor tecnológico**, com agilidade e controle total de uso.

Objetivo de aprendizagem: Benefícios da Nuvem

Benefícios da Nuvem



Alta disponibilidade



Elasticidade



Escalabilidade

A nuvem oferece **benefícios estratégicos** para empresas e projetos:

Alta disponibilidade: acesso contínuo aos recursos, com SLAs que chegam a 99,99% de uptime; ideal para sistemas críticos.

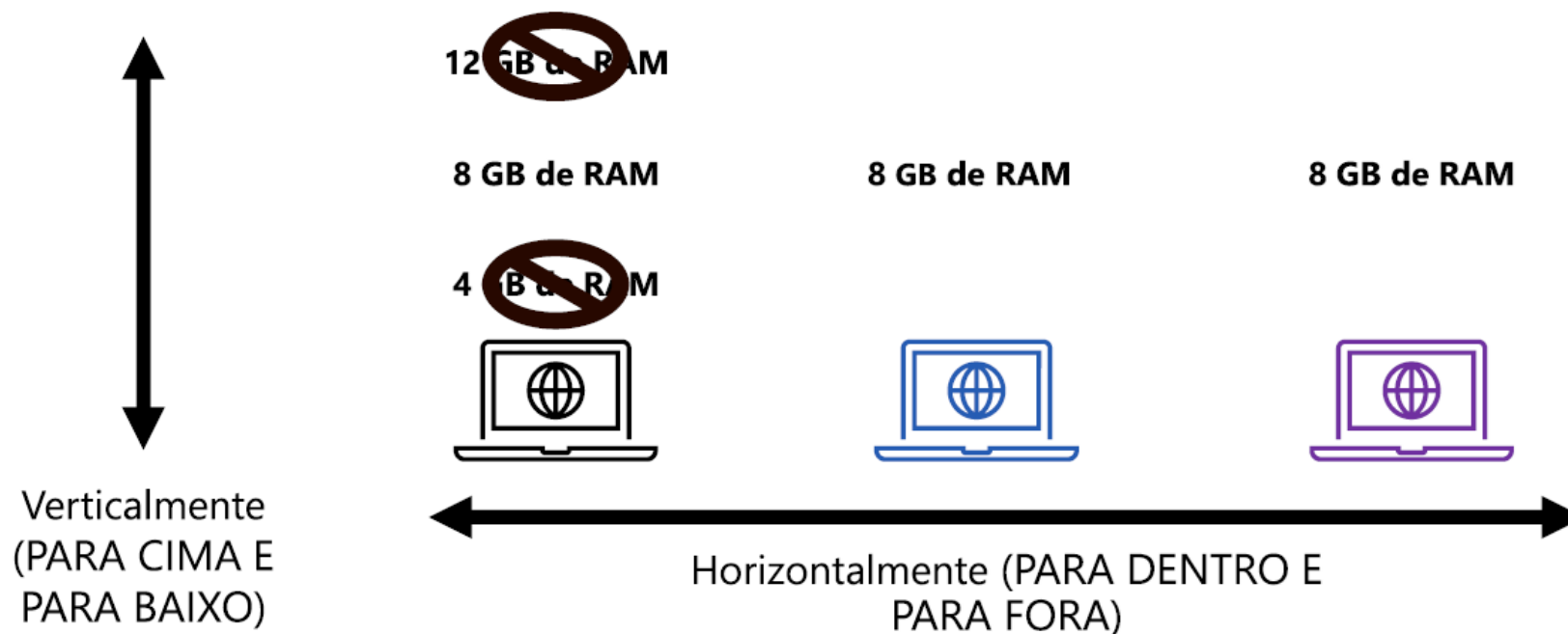
Elasticidade: ajuste automático da capacidade com base na demanda (escala lateral), otimizando recursos conforme o uso.

Escalabilidade: capacidade de aumentar ou reduzir manualmente o poder computacional (escala vertical), ideal para crescimento planejado.

A infraestrutura em nuvem permite obter esses benefícios **sem investimento inicial**, de forma prática e sob demanda.

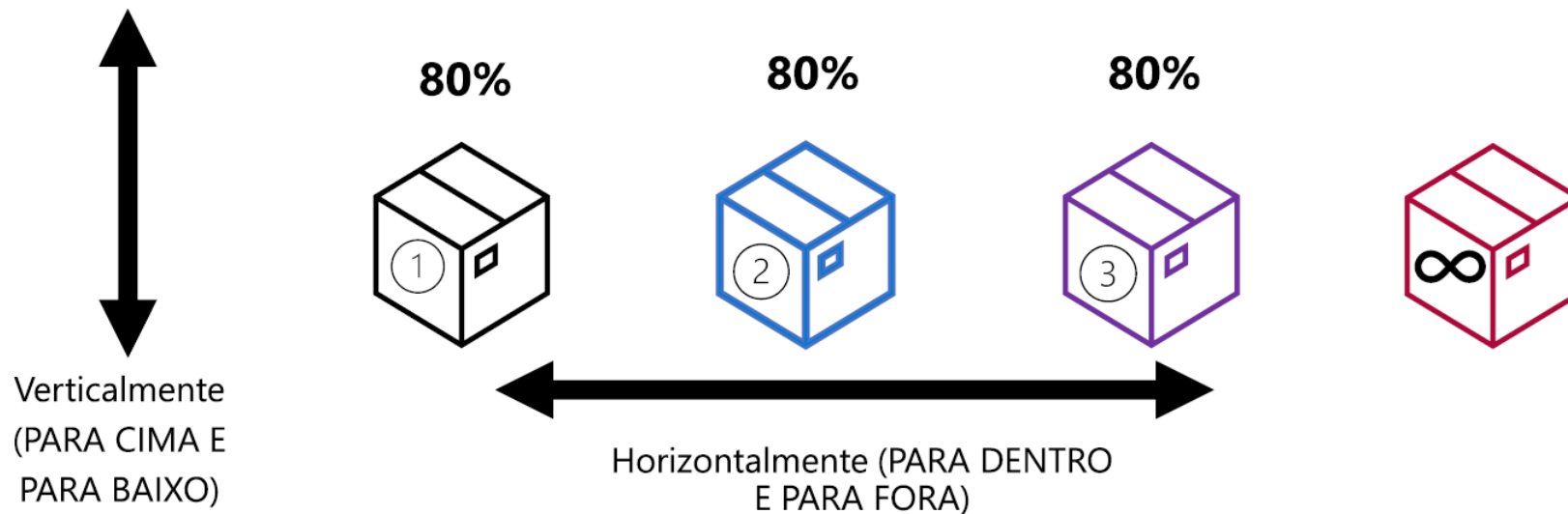
Escalabilidade e elasticidade

- ✓ **Escalabilidade:** a capacidade de escalar verticalmente (para cima ou para baixo) e horizontalmente (para dentro ou para fora)
- ✓ **Elasticidade:** a capacidade de escalar automaticamente



Escalabilidade e elasticidade

- ✓ **Escalabilidade:** a capacidade de escalar verticalmente (para cima ou para baixo) e horizontalmente (para dentro ou para fora)
- ✓ **Elasticidade:** a capacidade de escalar automaticamente



Benefícios da Nuvem



Alta disponibilidade



Elasticidade



Escalabilidade



Confiabilidade



Previsibilidade



Segurança



Governança



Capacidade de gerenciamento

Objetivo de aprendizagem: Serviço de Nuvem

IaaS (Infraestrutura como Serviço)

IaaS (Infraestrutura como Serviço): é o modelo mais básico da computação em nuvem.

O cliente **aluga recursos de infraestrutura** como máquinas virtuais, armazenamento, redes e sistemas operacionais.

A infraestrutura é **provisionada via internet**, e o gerenciamento de hardware, data centers e rede fica a cargo do provedor.

Exemplos no Azure: **Máquinas Virtuais (VMs)**, firewalls, e recursos de rede.

O cliente tem mais controle e também **mais responsabilidade de configuração, atualização e segurança** do que em modelos mais gerenciados.



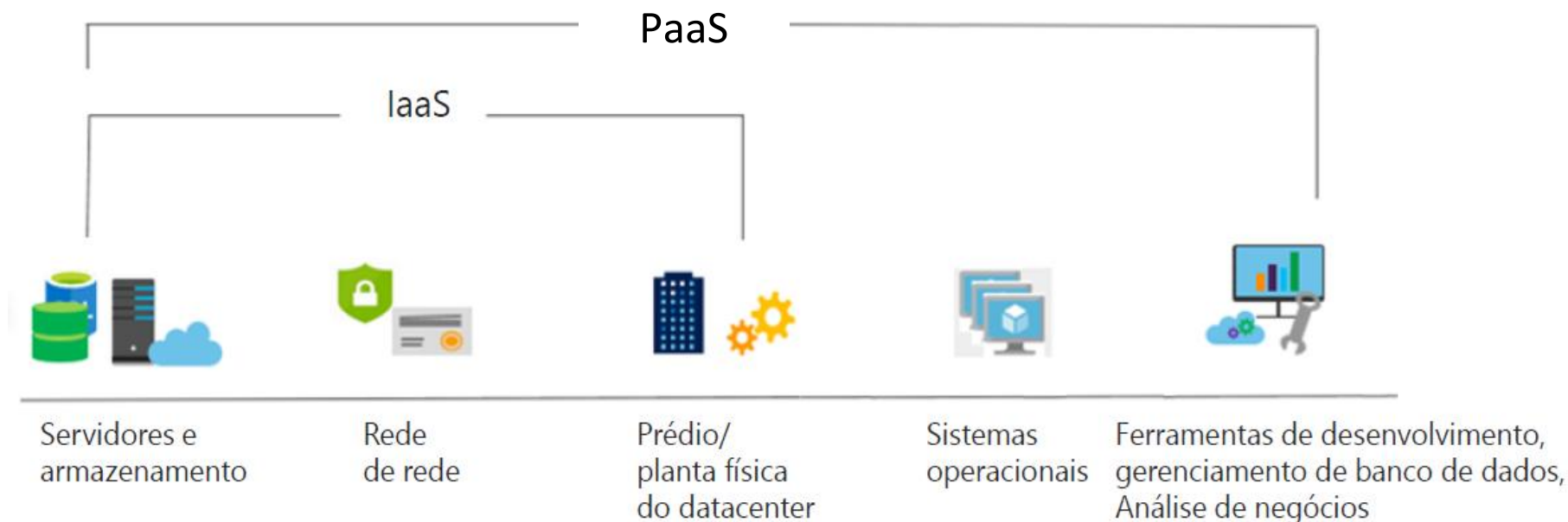
PaaS (Plataforma como Serviço)

PaaS (Plataforma como Serviço): oferece um ambiente completo para **desenvolver, testar e implantar aplicações**, sem a necessidade de gerenciar servidores, sistemas operacionais ou atualizações.

O provedor cuida da infraestrutura e da manutenção, permitindo ao desenvolvedor **focar exclusivamente na lógica e entrega da aplicação**.

Exemplo no Azure: **App Service**, onde o desenvolvedor simplesmente publica seu código, sem se preocupar com a VM por trás.

Ideal para quem deseja **diminuir responsabilidades técnicas** e aumentar a agilidade no desenvolvimento.



Software como serviço (SaaS)

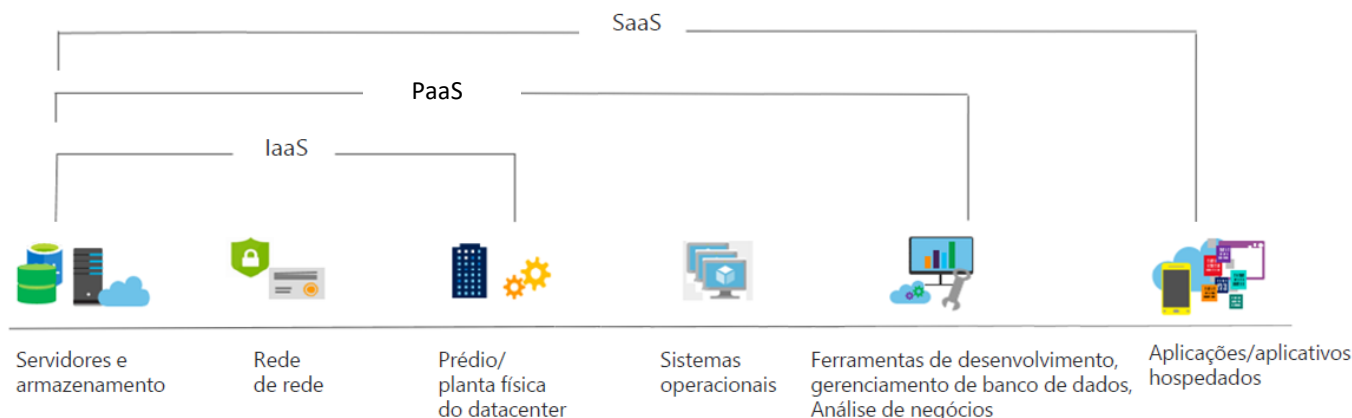
o software completo é entregue como serviço via internet, sem necessidade de instalação ou manutenção pelo cliente.

O provedor é responsável por toda a infraestrutura, atualizações e suporte da aplicação.

O cliente gerencia apenas **usuários e dados**, com acesso imediato ao sistema.

Exemplo: **Microsoft 365**, com e-mail, calendário e ferramentas online hospedadas nos data centers da Microsoft.

Ideal para uso prático, com **mínima responsabilidade técnica e máxima conveniência**.




Comparação dos serviços de nuvem


IaaS	PaaS	SaaS
<p>Maior flexibilidade, o usuário gerencia servidores, rede e armazenamento.</p> <p>Responsabilidade maior é do cliente.</p>	<p>Foca no desenvolvimento, o provedor cuida da infraestrutura e sistema operacional.</p> <p>provedor assume mais responsabilidade, mas identidade e diretórios ainda são compartilhados.</p>	<p>O usuário consome o software pronto, paga por uso ou assinatura, e gerencia apenas os dados e acessos.</p> <p>O provedor é responsável por quase tudo, exceto usuários e dados.</p>


Modelo de responsabilidade compartilhada

O modelo de responsabilidade compartilhada define quem é responsável por cada parte do ambiente de nuvem: cliente, provedor (ex: Microsoft) ou ambos.

Responsabilidade		SaaS	PaaS	IaaS	Na infraestrutura local
Responsabilidade sempre Retido pelo cliente	Informações e dados	Cliente	Cliente	Cliente	Cliente
	Dispositivos (móveis e PCs)	Cliente	Cliente	Cliente	Cliente
	Contas e identidades	Cliente	Cliente	Cliente	Cliente
Responsabilidade varia por tipo	Identidade e infraestrutura do diretório	Compartilhado	Compartilhado	Cliente	Cliente
	Aplicações	Compartilhado	Compartilhado	Cliente	Cliente
	Controles de rede	Compartilhado	Compartilhado	Cliente	Cliente
	Sistema operacional	Compartilhado	Compartilhado	Cliente	Cliente
Transferências de responsabilidade para o provedor de nuvem	Hosts físicos	Compartilhado	Compartilhado	Compartilhado	Cliente
	Rede física	Compartilhado	Compartilhado	Compartilhado	Cliente
	Datacenter físico	Compartilhado	Compartilhado	Compartilhado	Cliente

 Microsoft

 Cliente

 Compartilhado

Revisão da Sessão 02

- ✓ O modelo de responsabilidade compartilhada
- ✓ Nuvem pública, privada e híbrida
- ✓ Benefícios da computação na nuvem
- ✓ Tipos de serviço de nuvem

Arquitetura do Azure e serviços de computação do Azure



Objetivo de aprendizagem: componentes da arquitetura do Azure

Regiões

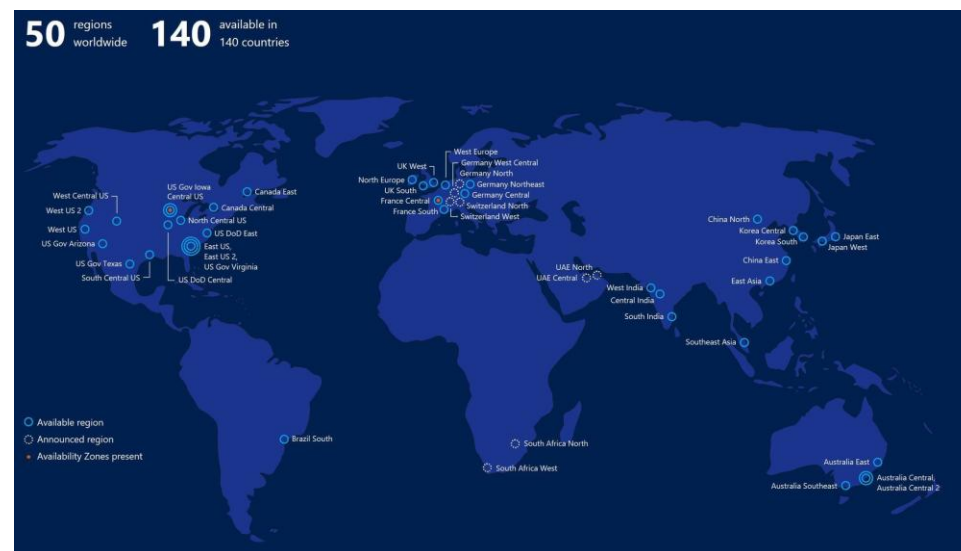
Região: conjunto de datacenters interconectados em uma área geográfica específica.

Cada **região do Azure** fornece **escala, flexibilidade** e ajuda a **preservar a residência dos dados**.

É importante escolher **regiões próximas aos usuários finais** para melhor desempenho.

Nem todos os recursos estão disponíveis em todas as regiões — atenção à **disponibilidade regional**.

Algumas funcionalidades são **globais**, enquanto outras dependem da região escolhida.



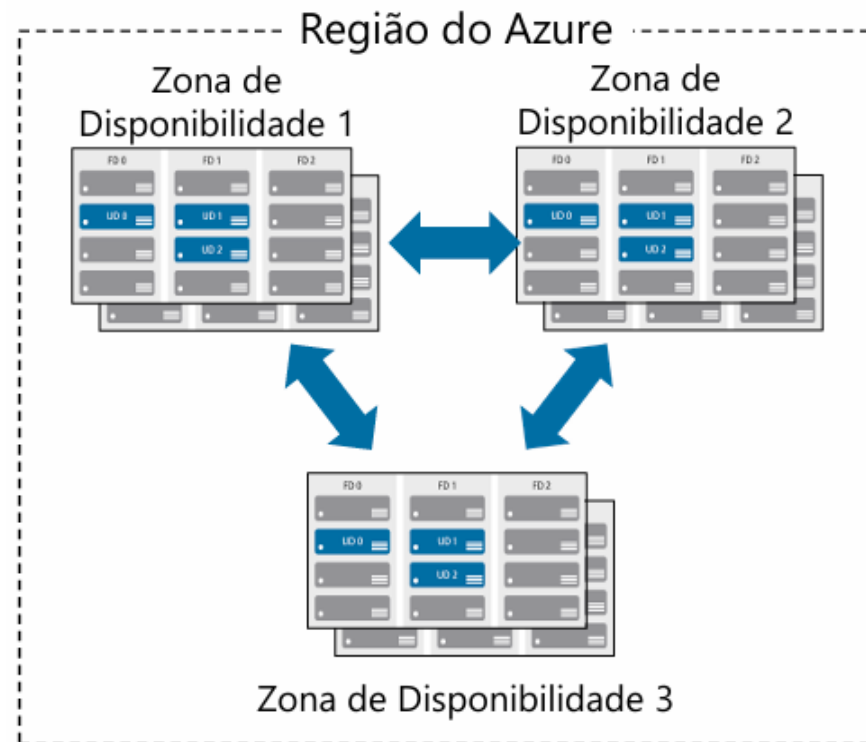
Zonas de disponibilidade

Fornecer proteção contra o tempo de inatividade devido a falhas no datacenter

Datacenters separados fisicamente na mesma região

Cada datacenter é equipado com energia, resfriamento e rede independentes.

Conectado por meio de redes privadas de fibra óptica



Pares de regiões

Pelo menos 300 milhas de separação entre os pares de regiões.

Replicação automática para alguns serviços

Recuperação priorizada da região em caso de interrupção.

As atualizações são lançadas sequencialmente para minimizar o tempo de inatividade.

Região		Região
Centro-Norte dos EUA		Centro-Sul dos EUA
Leste dos EUA		Oeste dos EUA
Oeste dos EUA 2		Centro-Oeste dos EUA
EUA Leste 2		EUA Central
Canadá Central		Leste do Canadá
Norte da Europa		Oeste da Europa
Oeste do Reino Unido	↔	Sul do Reino Unido
Alemanha Central		Nordeste da Alemanha
Sudeste da Ásia		Leste da Ásia
Leste da China		Norte da China
Leste do Japão		Oeste do Japão
Sudeste da Austrália		Leste da Austrália
Sul da Índia		Índia Central
Sul do Brasil (primário)		Centro-Sul dos EUA

Regiões Soberanas do Azure (serviços do governo dos EUA)

Atende às necessidades de segurança e conformidade de agências federais dos EUA, governos estaduais e locais e seus provedores de soluções.

Azure Governamental:

- Instância separada do Azure
- Fisicamente isolada de implantações governamentais que não são dos EUA
- Acessível apenas para funcionários selecionados e autorizados



Regiões Soberanas do Azure (Azure China)

A Microsoft é o primeiro provedor estrangeiro de serviços de nuvem pública da China, em conformidade com os regulamentos governamentais.

Recursos do Azure China:

- Instância fisicamente separada dos serviços de nuvem do Azure operados pela 21Vianet
- Todos os dados permanecem dentro da China para garantir a conformidade



Demonstração

Explore a infraestrutura global do Azure, acesse:

<https://datacenters.microsoft.com/globe/explore>

- Demonstração prática do **mapa interativo do Azure**, mostrando a infraestrutura global.
- Visualização por **geografia e regiões** (ex: Brasil Sul – com 3 zonas de disponibilidade, localizada em SP).
- Brasil Sul é a única região que **replica para fora da própria geografia (EUA)**.
- É possível verificar **produtos disponíveis, certificações de conformidade (ISO, etc.) e requisitos legais/industriais** por região.
- **Pontos de presença (PoP)** são mostrados como **bolas menores**, indicando **entradas de rede** que melhoram a **latência e desempenho**.
- O site também permite **filtrar regiões por conformidade, sustentabilidade, disaster recovery e disponibilidade**.
- Há ainda um **tour virtual pelos datacenters** da Microsoft, com visão completa da operação.

Recursos do Azure

Os recursos do Azure são componentes como armazenamento, máquinas virtuais e redes que estão disponíveis para criar soluções de nuvem.



Máquinas Virtuais



Contas de armazenamento



Redes virtuais



Serviços de Aplicativos



Bancos de dados SQL



Functions

Grupos de Recursos

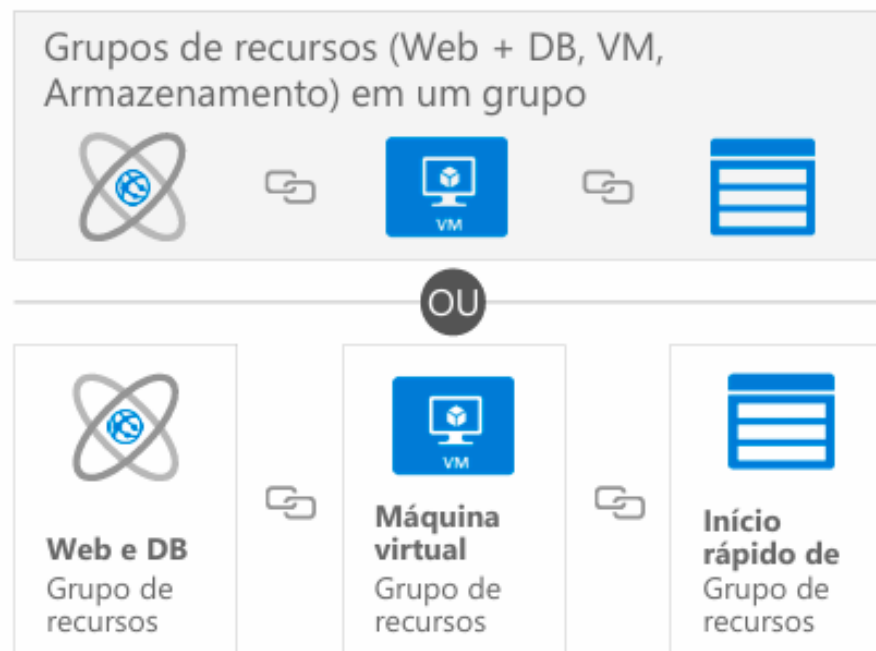
Um **grupo de recursos** é um contêiner para gerenciar e agregar recursos em uma única unidade.

Os recursos podem existir apenas em um grupo.

Os recursos podem existir em diferentes regiões.

Os recursos podem ser movidos para diferentes grupos de recursos.

As aplicações podem utilizar vários grupos de recursos.

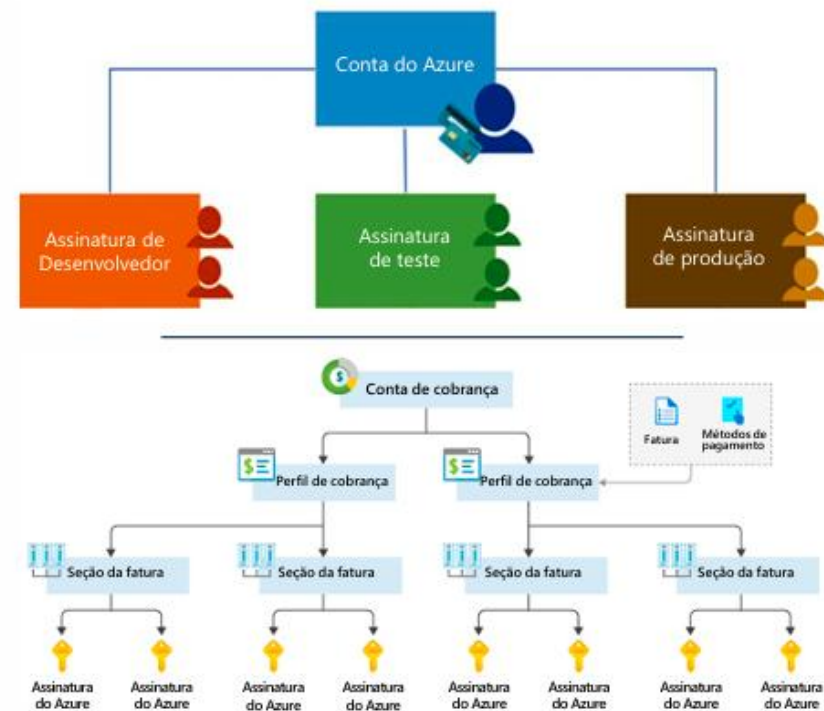


Assinaturas do Azure

Uma **assinatura do Azure** fornece acesso autenticado e autorizado a contas do Azure.

Limite de cobrança: gere relatórios de cobrança e faturas separados para cada assinatura

Limite de controle de acesso: gerencie e controle o acesso aos recursos que os usuários podem provisionar com assinaturas específicas.



Grupos de gerenciamento

Grupos de gerenciamento podem incluir várias assinaturas do Azure

As assinaturas herdam as condições aplicadas ao grupo de gerenciamento

10.000 grupos de gerenciamento podem ser compatíveis em um único diretório

Uma árvore de grupo de gerenciamento pode oferecer suporte a até seis níveis de profundidade



Objetivo de aprendizagem: Computação

Serviços de computação do Azure

A **computação** do Azure é um serviço de computação sob demanda que fornece recursos de computação, como discos, processadores, memória, rede e sistemas operacionais.



Máquinas
Virtuais



Serviços de
Aplicativos



Instâncias de
Contêiner



Serviços do
Kubernetes do
Azure (AKS)

Máquinas virtuais do Azure

Máquinas Virtuais (VM) do Azure são emulações de software de computadores físicos.

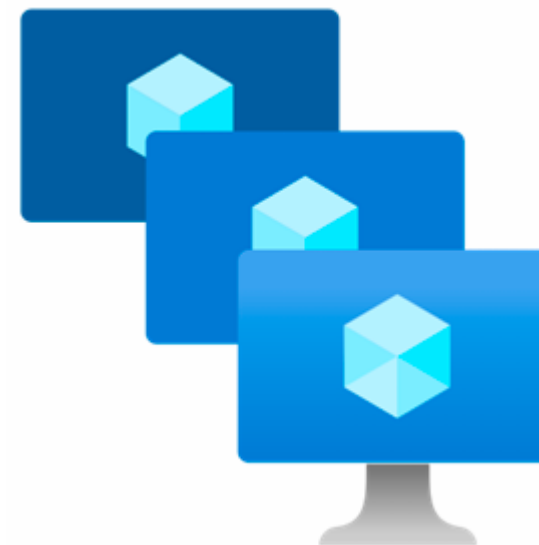
- Inclui memória, armazenamento, rede e processador virtual.
- Oferta de IaaS que fornece controle total e personalização.



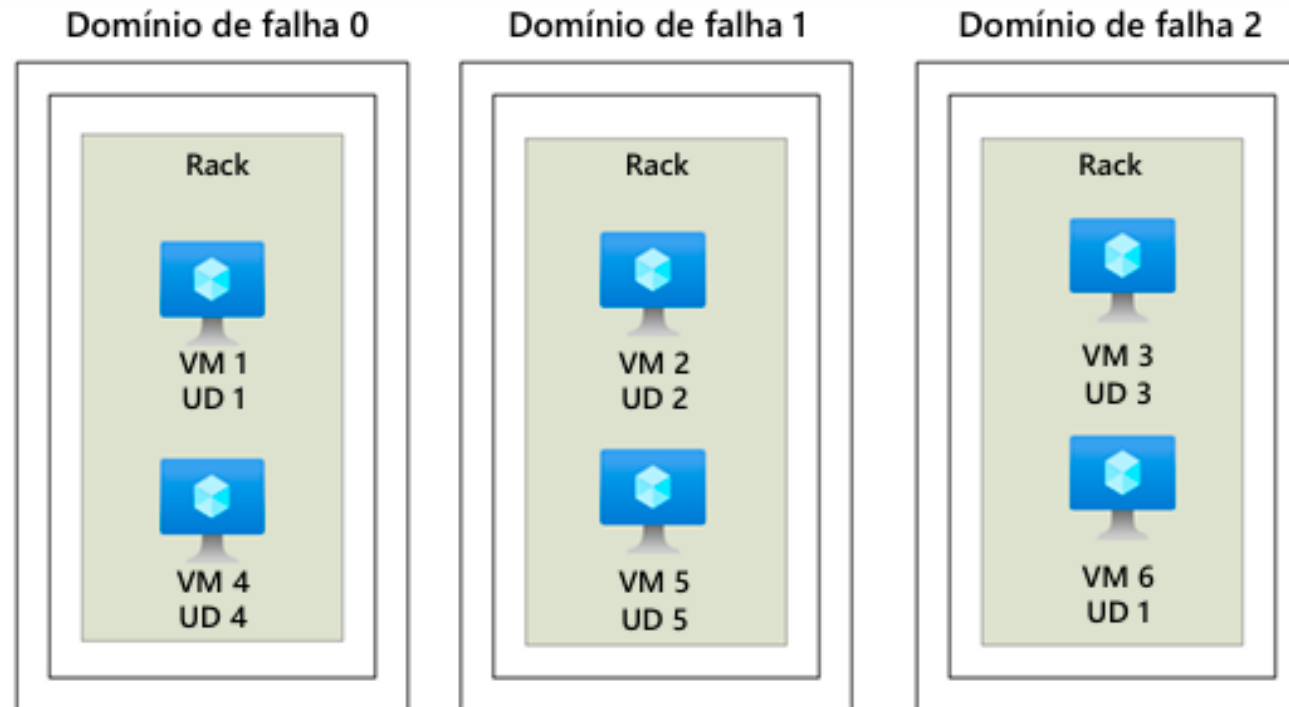
Conjuntos de escalonamento de VMs

Os conjuntos de escalas fornecem uma oportunidade com balanceamento de carga para escalar recursos automaticamente.

- Expanda quando as necessidades de recursos aumentarem.
- Reduza quando as necessidades de recursos forem menores.



Conjuntos de disponibilidade de VM



Serviços de Contêiner do Azure

Os **Contêineres** do Azure são um ambiente virtualizado leve que não requer gerenciamento do sistema operacional e podem responder a alterações sob demanda.

- **Instâncias de Contêiner do Azure:** uma oferta de PaaS que executa um contêiner ou pod de contêineres no Azure.
- **Aplicativos de Contêiner do Azure:** uma oferta de PaaS como instâncias de contêiner que podem equilibrar e escalar.
- **Serviço de Kubernetes do Azure:** um serviço de orquestração para contêineres com arquiteturas distribuídas e grandes volumes de contêineres.



Serviços de Contêiner do Azure

1. Instâncias de Contêiner do Azure (ACI)

O que é?

É a forma mais simples e rápida de rodar um contêiner no Azure.

Analogia: Pense nisso como pedir uma **pizza inteira e pronta**. Você apenas diz o sabor (o contêiner) e a quantidade, e o Azure se vira para assar e entregar. Você não se preocupa em ligar o forno, comprar os ingredientes ou gerenciar a cozinha.

Ideal para: Tarefas simples e rápidas, como processar um único lote de dados, rodar um script ou hospedar um serviço pequeno que não precisa de muitos recursos.

Serviços de Container do Azure

2. Aplicativos de Container do Azure

O que é?

Um serviço mais avançado para rodar containers que precisam de escala e gerenciamento de tráfego.

Analogia: É como ter uma **pizzaria semi-automática**. Você ainda não se preocupa com o forno, mas pode configurar regras para a pizzaria: "se chegarem muitos pedidos, faça mais pizzas automaticamente".

Ideal para: Microsserviços e aplicações web que precisam de um gerenciamento mais robusto de tráfego e que escalem (aumentem a capacidade) automaticamente conforme a demanda.

Serviços de Container do Azure

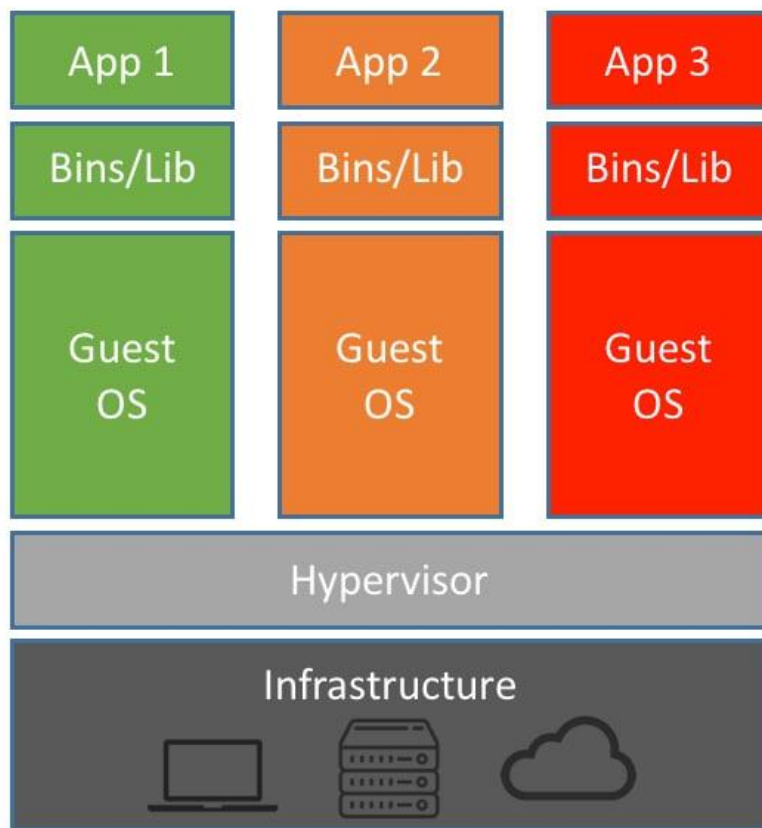
3. Serviço de Kubernetes do Azure (AKS)

O que é? O serviço mais poderoso e completo, usado para orquestrar grandes volumes de contêineres.

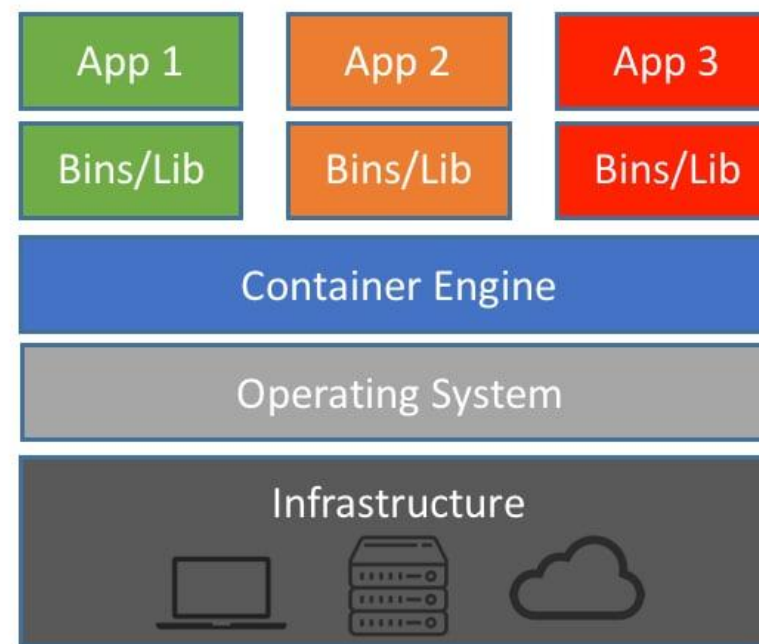
Analogia: É como ter uma **fábrica de pizzas super moderna e automatizada**. Você é o gerente e tem controle total. Você configura máquinas, define fluxos de trabalho e monitora tudo. Se uma máquina falhar, a fábrica sabe como ligar outra automaticamente.

Ideal para: Aplicações complexas e distribuídas, que usam muitos contêineres e precisam de controle total sobre o gerenciamento, escalabilidade e comunicação entre eles. É a solução mais robusta para ambientes de produção em larga escala.

Serviços de Contêiner do Azure



Machine Virtualization



Containers

Comparação das opções de computação

Máquinas virtuais

Servidor baseado em nuvem que oferece suporte a ambientes Windows ou Linux.

Útil para migrações lift-and-shift para a nuvem.

Pacote completo do sistema operacional, incluindo o sistema operacional do host.

Contêineres

Ambiente leve em miniatura adequado para a execução de microsserviços.

Projetado para escalabilidade e resiliência por meio da orquestração.

Aplicações e serviços são empacotados em um contêiner que fica na parte superior do sistema operacional do host. Vários contêineres podem estar em um sistema operacional do host.

Serviços de Aplicativos Azure

Os **Serviços de Aplicativo** do Azure são uma plataforma totalmente gerenciada para criar, implantar e escalar aplicativos Web e APIs rapidamente.



- Funciona com .NET, .NET Core, Node.js, Java, Python ou php.
- Oferta de PaaS com requisitos de performance, segurança e conformidade de nível empresarial.

Revisão da Sessão 03

- ✓ Infraestrutura física e de gerenciamento do Microsoft Azure Nuvem pública, privada e híbrida
- ✓ Máquinas virtuais do Azure da computação na nuvem
- ✓ Serviços de contêiner do Azure
- ✓ Comparação do serviço de computação do Azure

Redes do Azure



Objetivo de aprendizagem:

Redes

Serviços de rede do Azure

A **Rede Virtual do Azure** permite que os recursos do Azure se comuniquem uns com os outros, com a Internet e com as redes na infraestrutura local.



- Pontos de extremidade públicos, acessíveis de qualquer lugar da Internet
- Pontos de extremidade privados, acessíveis somente de dentro da sua rede
- Sub-redes virtuais, segmentadas para atender às suas necessidades
- Emparelhamento de rede, conectando suas redes privadas

DNS do Azure

- **DNS (Domain Name System)** traduz nomes de domínio (ex: www.microsoft.com) para endereços IP;
- No Azure, o serviço de DNS é **gerenciado e distribuído globalmente**, garantindo:
 - Alta disponibilidade;**
 - Desempenho** otimizado;
 - Segurança** e;
 - Facilidade de uso.**

Você pode **delegar a resolução DNS pública** de seus domínios para o Azure.

Também é possível configurar um **DNS privado interno**, para resolução de nomes entre **máquinas virtuais** e recursos **dentro da sua rede virtual**.

Azure oferece suporte tanto para **DNS público (internet)** quanto **privado (intranet)**.

Revisão da Sessão 04

- ✓ Redes Virtuais
- ✓ DNS do Azure

Criando sua conta Gratuita no Azure

Crie sua conta gratuita no Azure

1. Azure for Students

- Para estudantes com e-mail acadêmico;
- 100 USD de crédito e nenhum cartão de crédito necessário;
- Acesse: <https://azure.microsoft.com/pt-br/free/students>

2. Azure Free Trial (público geral)

- 200 USD de crédito por 30 dias;
- Requer cartão de crédito para verificação (não cobra, só valida);
- Acesse: <https://azure.microsoft.com/pt-br/free>

Revisão da Sessão 05

- ✓ Azure for students
- ✓ Azure free trial (Público geral)

Repositório com GitHub

Passo a passo para criar a conta no GitHub

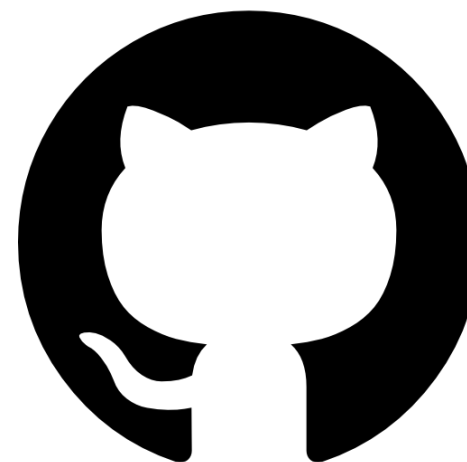
O **repositório** é a pasta do seu projeto, onde o Git armazena todas as versões.

GitHub é uma plataforma na web para hospedar e gerenciar esses repositórios.

Acesse o site: <https://github.com>

Na página inicial, você verá:

- Um campo para digitar seu **email** e iniciar o cadastro;
- No canto superior direito, os botões: **Sign In** (entrar) e **Sign Up** (criar conta);
- Clique em **Sign Up** para criar sua conta;
- OBS: Importante configurar métodos de 2 fatores para autenticação para segurança e configuração em outras plataformas!



Revisão da Sessão 06

✓ Repositório com GitHub

Versionamento de aplicações com GitHub Desktop

Versionamento de aplicações

O que é Versionamento e por que ele é Essencial?

Imagine uma máquina do tempo para o seu código!

O versionamento é o ato de registrar e rastrear todas as alterações no seu projeto ao longo do tempo.

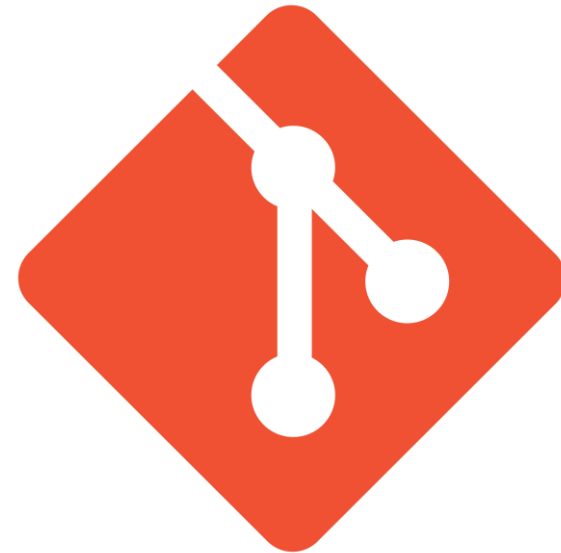
A ferramenta mais popular para isso é o **Git**.

Benefícios Chave:

Colaboração: Permite que várias pessoas trabalhem no mesmo projeto sem conflitos.

Histórico: Você tem acesso total a todas as versões, podendo voltar a um ponto anterior a qualquer momento.

Segurança: Protege seu trabalho, garantindo que nenhuma alteração seja perdida.



Principais comandos Git

git init

Cria um novo repositório Git vazio ou reinicializa um já existente na pasta atual.

Ex: git init

git clone [url]

Clona um repositório remoto para sua máquina local, incluindo todo o histórico de versões.

Ex: git clone https://github.com/seu-usuario/seu-projeto.git

git add .

Adiciona todas as alterações de arquivos modificados para a área de preparação (staging area).

Ex: git add .

git commit -m "[mensagem]"

Salva as alterações da área de preparação no histórico do repositório local com uma mensagem descritiva.

Ex: git commit -m "Adiciona a nova tela de login"

git status

Mostra o estado atual do repositório, indicando arquivos modificados, na área de preparação e não rastreados.

Ex: git status

git push

Envia os commits do seu repositório local para o repositório remoto

Ex: git push origin main

git pull

Baixa e integra as alterações do repositório remoto para o seu repositório local.

Ex: git pull origin main

git branch

Lista todos os branches do repositório. Pode ser usado para criar ou deletar branches.

Ex: git branch nova-feature

git checkout [branch]

Muda para um branch específico. Use git checkout -b [novo-branch] para criar e mudar de branch.

Ex: git checkout nova-feature

git merge [branch]

Integra as alterações de um branch especificado no branch atual.

Ex: git merge nova-feature

Download: <https://git-scm.com/downloads>

GitHub Desktop

O **GitHub Desktop** é a forma mais fácil e intuitiva de usar o Git, especialmente para quem está começando. Ele elimina a necessidade de decorar comandos de linha e permite que você se concentre no que realmente importa: seu código.

Por que usar o GitHub Desktop?

Interface Visual: Em vez de digitar comandos, você gerencia seu projeto através de botões e menus simples. É como usar um aplicativo de edição de texto, mas para o seu repositório.

Controle de Versão Simplificado: Operações complexas como **commit**, **push** e **pull** se tornam cliques de botão. Você vê exatamente quais arquivos foram alterados e quais estão prontos para serem salvos.

Fluxo de Trabalho Descomplicado: Gerenciar branches, mesclar alterações e resolver conflitos é feito de maneira visual. O GitHub Desktop mostra o histórico do seu projeto de forma clara, ajudando a entender a evolução do seu código.

Download: <https://desktop.github.com/download/>



Visual Studio Code

Visual Studio Code: Seu Aliado no Versionamento

Visual Studio Code (VS Code) é um editor de código-fonte leve, mas poderoso, desenvolvido pela Microsoft. Ele é multiplataforma, rodando em Windows, macOS e Linux, e é conhecido por sua flexibilidade e extensibilidade, suportando diversas linguagens de programação e oferecendo uma vasta gama de extensões para personalizar e aprimorar a experiência do usuário.

O Visual Studio Code (VS Code) possui uma integração nativa e poderosa com o Git.

A aba de Controle do Código-Fonte (o ícone de ramificação) permite que você:

- Visualize e gerencie todas as alterações pendentes;
- Faça **commits** e **pushes** com apenas um clique;
- Compare versões do seu código facilmente.

Isso torna o processo de versionamento mais fluido e visual, sem a necessidade de comandos complexos.

Download: <https://code.visualstudio.com/download>

Atalhos e Dicas Essenciais para o VS Code

Dominar alguns atalhos e funcionalidades do VS Code pode acelerar muito o seu fluxo de trabalho. Vamos explorar os que você mencionou e adicionar outros que são fundamentais.

1. Geração de Texto de Preenchimento (Lorem Ipsum)

Essa é uma funcionalidade do Emmet, que já vem integrada ao VS Code.

Como funciona: Digite `lorem` em um arquivo HTML e pressione a tecla Tab. O VS Code irá gerar um parágrafo completo de texto fictício (Lorem Ipsum).

Variação: Você pode especificar o número de palavras. Por exemplo, `lorem10` gera um parágrafo com 10 palavras.

2. Estrutura Básica de um HTML5

Este é um dos atalhos mais úteis para iniciar um novo projeto HTML.

Como funciona: Em um arquivo `.html` vazio, digite `!` e pressione a tecla Tab ou Enter. O VS Code irá inserir automaticamente todo o boilerplate de um documento HTML5

Atalhos e Dicas Essenciais para o VS Code

3. Criação Rápida de Arquivos

Essa funcionalidade é uma ótima maneira de criar arquivos relacionados diretamente do seu código.

Como funciona: Se você digitar `<link rel="stylesheet" href="css/style.css">` no seu `index.html` e o arquivo `style.css` ainda não existir, você pode segurar a tecla `Ctrl` (Windows/Linux) ou `Cmd` (macOS) e clicar no caminho `"css/style.css"`. Uma caixa de diálogo irá perguntar se você deseja criar o arquivo, e ele será criado na pasta correta.

4. Seletor de Cores (Color Picker)

Essa funcionalidade ajuda muito na hora de trabalhar com cores em CSS.

Como funciona: Simplesmente passe o mouse sobre qualquer valor de cor em um arquivo CSS (seja ele em formato HEX, RGB, HSL, etc.). O VS Code exibirá um pequeno seletor de cores que permite visualizar e ajustar a cor facilmente.

Atalhos e Dicas Essenciais para o VS Code

5. Seleção de Múltiplas Linhas ou Palavras (Multi-cursor)

Esta funcionalidade é extremamente poderosa para fazer edições em massa.

Como funciona:

Para adicionar múltiplos cursores em diferentes pontos do código: Segure a tecla Alt (Windows/Linux) ou Option (macOS) e clique nos locais onde deseja adicionar um cursor.

Para selecionar a próxima ocorrência de uma palavra: Selecione a palavra e pressione Ctrl + D (Windows/Linux) ou Cmd + D (macOS).

6. Formatar o Código (Organizar)

Esse é um atalho essencial para manter o código limpo e legível.

Como funciona: Pressione Shift + Alt + F (Windows/Linux) ou Shift + Option + F (macOS). O VS Code irá formatar automaticamente o código do documento atual, alinhando a indentação e a estrutura.

Atalhos e Dicas Essenciais para o VS Code

7. Paleta de Comandos

A Paleta de Comandos é uma ferramenta central no VS Code para executar praticamente qualquer ação.

Como funciona: Pressione Ctrl + Shift + P (Windows/Linux) ou Cmd + Shift + P (macOS). Isso abre uma barra de pesquisa que permite acessar qualquer comando, de "Format Document" a "Abrir Terminal", sem precisar usar o mouse.

8. Comentar/Descomentar Linhas

Esse atalho facilita a habilitação ou desabilitação de trechos de código rapidamente.

Como funciona: Selecione a(s) linha(s) que você quer comentar e pressione Ctrl + / (Windows/Linux) ou Cmd + / (macOS). O atalho funciona como um "liga/desliga", ou seja, se você o pressionar novamente, ele descomenta.

Atalhos e Dicas Essenciais para o VS Code

9. Duplicar Linha

Uma forma rápida de replicar código sem precisar copiar e colar.

Como funciona: Posicione o cursor em uma linha e pressione Shift + Alt + Seta para cima/baixo (Windows/Linux) ou Shift + Option + Seta para cima/baixo (macOS). Isso duplicará a linha inteira para a direção que você escolher.

10. Mover Linha

Ideal para reordenar linhas de código sem bagunçar a indentação.

Como funciona: Posicione o cursor em uma linha e pressione Alt + Seta para cima/baixo (Windows/Linux) ou Option + Seta para cima/baixo (macOS). Isso moverá a linha para cima ou para baixo, alterando a sua posição no código

Atalhos e Dicas Essenciais para o VS Code

11. Abrir Terminal Integrado

Acesso rápido ao terminal do sistema dentro do próprio editor.

Como funciona: Pressione a tecla Ctrl + ` (Windows/Linux) ou Cmd + ` (macOS). Isso abrirá ou fechará o terminal integrado do VS Code, permitindo que você execute comandos sem sair do ambiente de trabalho.

12. Pesquisa Global

Uma ferramenta essencial para encontrar e substituir texto em todo o seu projeto.

Como funciona: Pressione Ctrl + Shift + F (Windows/Linux) ou Cmd + Shift + F (macOS). Isso abre a barra de pesquisa para buscar um texto em todos os arquivos do seu projeto, facilitando a navegação e edição em larga escala.

Demonstração – Aplicação

Passo a passo:

- Acessar e realizar o download do projeto **html**:
<https://github.com/raphaelbarretopro/html/archive/refs/heads/main.zip>
- Descompactar o arquivo **html-main** na pasta meus documentos e renomear para html
- Criar um novo repositório no Github Desktop com as seguintes informações:
 - nome: html
 - Local: documentos/html
 - README
 - MIT License
 - Publish repositior
 - Desmarcar Private
 - Realizar o Commit
 - Realizar o Push
- Abrir o github: <https://github.com>

Demonstração - Aplicação

Passo a passo:

- Abrir o projeto no VS Code: Ctrl + Shift + A
- Habilitar o auto Save
- Preferences > themes > Color Theme
- Fazer uma alteração do título;
- Abrir o Github Desktop;
- Realizar o Commit
- Realizar o Push
- Abrir o repositório no github: <https://github.com>
- Criar uma new Branch com o nome de **dev**
- Publicar a Branch
- Visualizar no github
- Abrir o Visual Studio
- Alterar o cabeçalho
- Abrir o github desktop
- Realizar o Commit
- Realizar o Push
- Abrir Github para verificar a alteração
- Merge into current Branch
- Selecione o Dev
- Criar a Merge Commit
- Realizar Push

Atividade com Issues

Melhoria Visual: Atualizar texto e cor de fundo do rodapé do projeto

O Que Precisamos Fazer?

Nossa missão está dividida em duas partes:

Atualizar o Texto: O texto atual do rodapé precisa ser atualizado para incluir o ano e o símbolo de copyright.

Texto Atual: Simulado AZ-900 - Desenvolvido por Raphael Barreto

Novo Texto Sugerido: © 2025 Simulado AZ-900 - Desenvolvido por "Nome do Aluno"

Mudar a Cor de Fundo: Para destacar o rodapé, vamos adicionar uma cor de fundo a ele.

Cor Sugerida: Cinza claro (#f2f2f2), para ser sutil e profissional.

Criando uma Página com GitHub Pages

Melhoria Visual: Atualizar texto e cor de fundo do rodapé do projeto

1. Hospedagem Gratuita e Simples

Use o **GitHub Pages**, um serviço do GitHub para hospedar sites estáticos (HTML, CSS e JavaScript).

É ideal para portfólios, blogs pessoais, documentação de projetos e landing pages.

2. O Processo

Crie um Repositório: Suba os arquivos do seu site para um novo repositório público no GitHub.

Ajuste as Configurações: Vá em **Settings** > **Pages**.

3. Selecione a Branch: Escolha a branch onde estão seus arquivos para que o GitHub saiba de onde "ler" seu site.

Ponto-chave: Em poucos minutos, seu site estará no ar com um endereço no formato `[seu-usuario].github.io/[nome-do-repositorio]`.

Revisão da Sessão 07

- ✓ Versionamento de aplicações com GitHub Desktop
- ✓ Atalhos e Dicas essenciais para VS Code
- ✓ Atividade com Issues
- ✓ Criando uma página com GitHub Pages

Deploy da aplicação na nuvem com PaaS

O que é Deploy?

Deploy de uma aplicação na nuvem, também chamado de implantação na nuvem, é o processo de disponibilizar uma aplicação desenvolvida para uso em um ambiente de computação em nuvem. Esse processo envolve a transferência dos arquivos e recursos da aplicação para um provedor de nuvem, onde ela será executada e acessível aos usuários finais.

Demonstração – Deploy App com PaaS

Passo a Passo: Deploy de Aplicação Web com PaaS no Azure

Agora, vamos demonstrar como realizar o deploy de uma aplicação estática utilizando o modelo de **PaaS (Platform as a Service)** no portal do Azure. Este método simplifica o processo, abstraindo a infraestrutura e permitindo que nos concentremos apenas no código da aplicação.

1. Criação do Aplicativo Web Estático

O primeiro passo é buscar e criar o recurso. No portal do Azure, vamos procurar por "Aplicativos Web Estáticos" e clicar no botão "Criar". Isso nos levará à tela de configuração, onde preencheremos os detalhes da nossa aplicação.

Demonstração – Deploy App com PaaS

Passo a Passo: Deploy de Aplicação Web com PaaS no Azure

Agora, vamos demonstrar como realizar o deploy de uma aplicação estática utilizando o modelo de **PaaS (Platform as a Service)** no portal do Azure. Este método simplifica o processo, abstraindo a infraestrutura e permitindo que nos concentremos apenas no código da aplicação.

1. Criação do Aplicativo Web Estático

O primeiro passo é buscar e criar o recurso. No portal do Azure, vamos procurar por "Aplicativos Web Estáticos" e clicar no botão "Criar". Isso nos levará à tela de configuração, onde preencheremos os detalhes da nossa aplicação.

Demonstração – Deploy App com PaaS

2. Configurações da Tela "Básico"

A primeira tela de configuração, conforme a imagem, é a de "**Básico**", onde definimos os detalhes do nosso projeto e a origem do código.

Detalhes do Projeto:

- **Assinatura:** Selecionamos a assinatura do Azure que desejamos usar. No exemplo, "Azure for Students" é a opção escolhida.
- **Grupo de Recursos:** Criamos um novo grupo de recursos, neste caso, chamado "Workshop". Um grupo de recursos é um contêiner para gerenciar recursos do Azure, o que facilita a organização e o gerenciamento do ciclo de vida dos recursos relacionados.
- **Nome:** Damos um nome para a nossa aplicação. Aqui, o nome "**html**" é o escolhido. Este nome será parte da URL de acesso à nossa aplicação.

Demonstração – Deploy App com PaaS

Plano de Hospedagem:

- **Tipo de plano:** Selecionamos a opção "Gratuito", ideal para projetos de hobby ou para fins de demonstração, como é o nosso caso. O plano gratuito oferece recursos suficientes sem custos.

Detalhes da Implantação:

- **Origem:** A origem do código é o "GitHub". Esta é uma das principais vantagens do PaaS para aplicativos estáticos, pois o Azure se integra diretamente com o GitHub para automatizar o processo de deploy.

Conta do GitHub: Conectamos nossa conta do GitHub, que no exemplo é "raphaelbarretopro", para que o Azure tenha permissão de acesso ao nosso repositório.

Demonstração – Deploy App com PaaS

Detalhes do Repositório:

- **Organização:** Selecionamos a organização do GitHub, que é "raphaelbarretopro".
- **Repositório:** Apontamos para o repositório específico que contém nosso código, que é "html".
- **Filial (Branch):** Escolhemos a branch do repositório que será utilizada para o deploy. No nosso caso, é a branch "main".

Detalhes do Build:

- **Predefinições de Build:** A configuração "Custom (detectado)" indica que o Azure detectará automaticamente o tipo de aplicativo.
- **Local do aplicativo:** Indicamos o diretório onde o código da nossa aplicação estática se encontra. O valor / significa que o código está na raiz do repositório.
- **Local de saída:** O valor . (ponto) indica que a pasta de saída do build é a própria raiz do repositório.

Demonstração – Deploy App com PaaS

3. Configurações da Tela "Configuração de Implantação"

Na segunda tela, "**Configuração de implantação**", conforme a imagem, temos a opção de escolher a política de autorização.

Política de autorização de implantação: Selecionamos a opção "**GitHub**". Esta opção permite que o Azure utilize o GitHub Identity para a implantação segura do código, o que simplifica o processo de autenticação e autorização.

Demonstração – Deploy App com PaaS

4. Revisão e Criação

A terceira e última etapa, "**Revisar + criar**", nos apresenta um resumo de todas as configurações que fizemos. É aqui que podemos revisar todos os detalhes antes de finalizar a criação do aplicativo.

Após revisar e confirmar que todas as informações estão corretas, clicamos em "Criar". O Azure então iniciará o processo de deploy, que incluirá a criação do recurso, a conexão com o GitHub e a execução de um workflow de build e deploy para publicar o seu site.

Este processo demonstra como o PaaS do Azure simplifica drasticamente a publicação de aplicações web, eliminando a necessidade de gerenciar a infraestrutura subjacente.

Demonstração – Deploy App com PaaS

5. Configuração de Domínio Personalizado

Depois que a aplicação for criada, você pode acessá-la e, no menu lateral, em "Configurações", clicar em "Domínios personalizados". Nesta tela, você terá a opção de adicionar um domínio personalizado à sua aplicação estática, tornando-a acessível por um nome de domínio amigável (por exemplo, meusite.com.br) em vez da URL padrão do Azure.

O Azure oferece três métodos principais para adicionar um domínio personalizado:

Domínio personalizado no DNS do Azure

Explicação: Esta opção é ideal se você já tem uma zona de DNS gerenciada diretamente no Azure. Ao selecionar esta opção, o Azure irá guiar você na criação dos registros DNS (como um registro A ou CNAME) automaticamente dentro da sua zona de DNS do Azure, simplificando todo o processo de configuração.

Demonstração – Deploy App com PaaS

Domínio personalizado em outro DNS

Explicação: Esta é a opção mais comum para quem tem um domínio registrado em um provedor externo, como Hostinger, HostGator ou Registro.br. O Azure irá fornecer os registros DNS que você precisa adicionar (como um registro A ou CNAME para apontar para a sua aplicação), mas você será responsável por adicionar manualmente esses registros no painel de controle do seu provedor de DNS externo.

Novo domínio personalizado

Explicação: Use esta opção se você ainda não tem um domínio registrado e deseja criar um novo. O Azure pode te ajudar a registrar um novo domínio e, em seguida, gerenciá-lo diretamente como uma zona de DNS do Azure. Isso integra a compra e a configuração do domínio em um único fluxo de trabalho.

O processo de adicionar um domínio personalizado é crucial para dar uma identidade profissional à sua aplicação web.

Revisão da Sessão 08

✓ Deploy da aplicação na nuvem com PaaS

Certificações Microsoft

Perguntas?