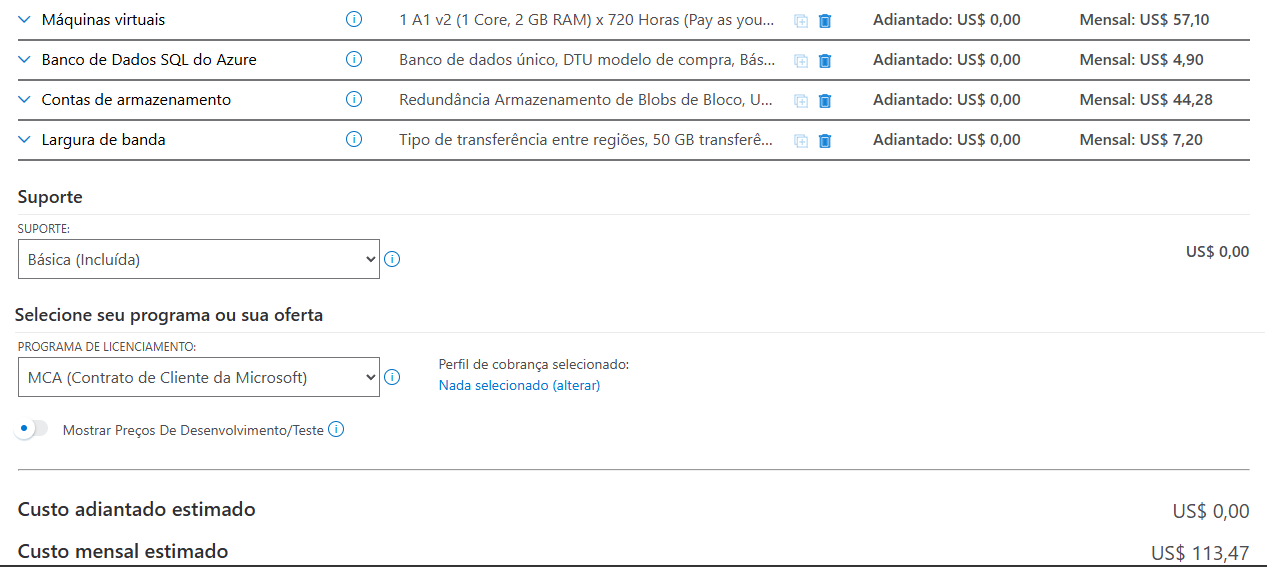
# Introdução

* Objetivo: "Simular custos de infraestrutura para um site de e-commerce básico no Azure."
* Componentes: Liste os serviços usados (VM, SQL Database, Blob Storage, etc.)

## Premissas

* Exemplo:
  + *"VM operando 24/7 durante o mês."*
  + *"Tráfego estimado de 50 GB/mês."*
  + *"Armazenamento de logs não críticos (Hot Storage)."*

## Tabela de custos



|  |  |  |  |
| --- | --- | --- | --- |
| Serviço | Configuração | Região | Custo Mensal (USD) |
| Máquinas Virtuais | 1 A1 v2 (1 Core, 2 GB RAM) x 720 Horas (Pay as you go), Linux, (Pay as you go); 0 managed disks – S4; Inter Region transfer type, 5 GB outbound data transfer from to East Asia | **Brazil Southeast** | **57,10** |
| Banco de dados SQL do Azure | Banco de dados único, DTU modelo de compra, Básico camada, B: 5 DTUs, 2 GB de armazenamento incluído por BD, 1 bancos de dados x 730 Horas, Armazenamento de 2 GB, RA-GRS Redundância de Armazenamento de Backup, 0 x 5 GB Retenção de Longo Prazo | **East US** | **4,90** |
| Contas de Armazenamento | Redundância Armazenamento de Blobs de Bloco, Uso geral V2, Namespace simples, LRS, Quente Camada de Acesso, Capacidade de 1.000 GB - PAGO CONFORME O USO, 10 x 10.000 operações de Gravação, 10 x 10.000 Operações de Lista e Criação de Contêiner, 10 x 10.000 operações de Leitura, 1 x 10.000 Outras operações. 1.000 GB Recuperação de Dados, 1.000 GB Gravação de Dados, SFTP desabilitado  ClonarExcluir | **Brazil Southeast** | **44,28** |
| Largura de Banda | Tipo de transferência entre regiões, 50 GB transferência de dados de saída de Brazil South para East Asia  ClonarExcluir |  | **7,20** |
| Total | |  | **113,48** |

## Análise de otimização

Melhorias para reduzir custo

* Sugestão 01: Utilizar a origem dos dados alguma região dos Estados Unidos, pode haver uma redução de até 50% para Bandwith;
* Sugestão 02: Reduzir as horas de operação da VM para 12h/dia, redução de 50% do serviço;
* Sugestão 03: Utilizar a região South Central US e reduzir o armazenamento, redução de até 50% do serviço;

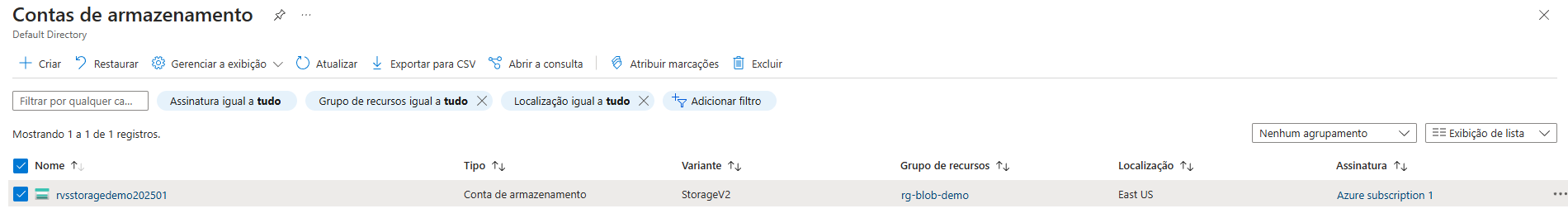
## Conclusão

Usar como origem o Brazil para os serviços contratados por ter um custo muito alto, se o projeto não tiver nenhum impedimento legal, podemos reduzir em 50% utilizando outras regiões, por exemplo: South Central US.

# Azure

## Criação da conta de armazenamento e container

Após estimativas, foi realizado a criação da conta de armazenamento ‘rvsstoragedemo202501’ e do Container ‘rvsarquivos’



## Script para upload e Download do arquivo CSV no container

#Instalação do pacote da azure -- executar se for a primeira vez utilizando

pip install azure-storage-blob

#Script Python para Upload do Arquivo

from azure.storage.blob import BlobServiceClient

# Configurações da conta de armazenamento

connection\_string = " SUA\_STRING\_DE\_CONEXÃO\_AQUI"  # Substitua pela sua connection string

container\_name = "rvsarquivos"  # Nome do container de destino

blob\_name = "incident\_event\_log.csv"  # Nome do arquivo no Blob Storage

local\_file\_path = "./incident\_event\_log.csv"  # Caminho local do arquivo

# Conectar ao Blob Storage

blob\_service\_client = BlobServiceClient.from\_connection\_string(connection\_string)

container\_client = blob\_service\_client.get\_container\_client(container\_name)

blob\_client = container\_client.get\_blob\_client(blob\_name)

# Fazer o upload do arquivo

try:

    with open(local\_file\_path, "rb") as data:

        blob\_client.upload\_blob(data, overwrite=True)  # Sobrescreve se o arquivo já existir

    print(f"Arquivo '{blob\_name}' enviado com sucesso para o container '{container\_name}'!")

except Exception as e:

    print(f"Erro: {e}")

#Script Python para DOWNLOAD do Arquivo

from azure.storage.blob import BlobServiceClient

# Configurações corrigidas

connection\_string = "SUA\_STRING\_DE\_CONEXÃO\_AQUI"

container\_name = "rvsarquivos"

blob\_name = "incident\_event\_log.csv" #nome do arquivo que foi realizado o

local\_file\_path = "./incident\_event\_log.csv"

# Conectar ao Blob Storage

blob\_service\_client = BlobServiceClient.from\_connection\_string(connection\_string)

container\_client = blob\_service\_client.get\_container\_client(container\_name)

blob\_client = container\_client.get\_blob\_client(blob\_name)

# Fazer o download

try:

    with open(local\_file\_path, "wb") as file:

        download\_stream = blob\_client.download\_blob()

        file.write(download\_stream.readall())

    print(f"Arquivo baixado: {local\_file\_path}")

except Exception as e:

    print(f"Erro: {e}")

Solução do Erro 403 (Acesso Negado):

* Confirme se seu IP está autorizado no firewall da conta de armazenamento (como discutido anteriormente).

Se não estiver acessar o container, no menu de Segurança + Rede, habilitar ‘Habilitado de redes virtuais e endereços IP selecionados’ e informar o endereço IP da sua máquina