

DOCUMENTO “Vamos interligar as coisas com a nuvem!”

Desenvolvedor : André Luis Gonçalves Carvalhal
Matrícula: 202203185403

BRIEFING:

Nesta Missão Prática você aprenderá a visualizar dados em tempo real provenientes de sensores conectados ao seu hub IoT. Isso será realizado por meio da execução de um aplicativo web Node.js em seu computador local. Após a configuração bem-sucedida e execução do aplicativo web local, você terá a opção de hospedá-lo no Serviço de Aplicativo do Azure para facilitar o acesso e a escalabilidade.

O fluxo de dados seguirá o caminho delineado na figura abaixo. O dispositivo simulado coletará dados de temperatura e umidade, os quais serão enviados para o Azure IoT Hub e exibidos através do Serviço de Aplicativo do Azure (Web App).

TESTANDO:

É possível executar o **Microsoft IOT Pusher.py** disponível neste repositório para enviar pacotes para a Cloud do Azure e então acessar a página de diagnóstico pelo endereço:

<https://viaccanapp.azurewebsites.net/>

EXECUÇÃO:

3. Adicionar um grupo de consumidores ao hub IoT é essencial para fornecer visualizações independentes do fluxo de eventos. Essas visualizações permitem que aplicativos e serviços do Azure consumam dados do mesmo ponto de extremidade do hub de eventos. Nesta etapa, você configurará um grupo de consumidores no ponto de extremidade integrado do seu hub IoT, que será utilizado pelo aplicativo web para ler dados. Execute o seguinte comando na CLI do Azure para adicionar um grupo de consumidores ao ponto de extremidade interno do hub IoT:

`az iot hub consumer-group create --hub-name YOUR_IOT_HUB_NAME --name YOUR_CONSUMER_GROUP_NAME`

```

$ /home/andre> az configure --defaults group:RPG0027
$ /home/andre> az iot hub consumer-group create --hub-name VIACCAN --name VIACCANCONSUMERS

{"etag": null,
 "id": "/subscriptions/e0c84085-1170-4c3d-9087-cf0e7abab7d0/resourceGroups/RPG0027/providers/Microsoft.Devices/IotHubs/VIACCAN/eventHubEndpoints/events/ConsumerGroups/VIACCANCONSUMERS",
 "name": "VIACCANCONSUMERS",
 "properties": {
   "created": "Fri, 14 Jun 2024 18:40:44 GMT",
   "properties": {
     "name": "VIACCANCONSUMERS"
   }
 }
}

```

5. Obter uma cadeia de conexão de serviço para seu hub IoT é um passo crucial para conceder permissões necessárias a serviços que precisam ler e gravar nos pontos de extremidade do hub IoT. Os hubs IoT são criados com várias políticas de acesso padrão, e uma dessas políticas é a política de serviço. Execute o seguinte comando para obter uma cadeia de conexão para o hub IoT, seguindo a política de serviço:

az iot hub show-connection-string --hub-name YOUR_IOT_HUB_NAME --policy-name service

```

$ /home/andre> az iot hub show-connection-string --hub-name VIACCAN --policy-name service
This command has been deprecated and will be removed in a future release. Use "IoT Extension (azure-iot) connection-string command (az iot hub connection-string show)" instead.
[Warning] This output may compromise security by showing the following secrets: connectionString. Learn more at: https://go.microsoft.com/fwlink/?linkid=2258669
{
  "connectionString": "HostName=VIACCAN.azure-devices.net;SharedAccessKeyName=service;SharedAccessKey=9EG7bCINMy1hUCHeF0z+Gd1Sh8TTfHA/a3AIoTHqhGqg="
}

```

6. Para definir as variáveis de ambiente, utilize os comandos abaixo na janela de comando, substituindo os espaços reservados pela cadeia de conexão de serviço para o hub IoT e o nome do grupo de consumidores criado anteriormente. Não é necessário colocar as cadeias de caracteres entre aspas.

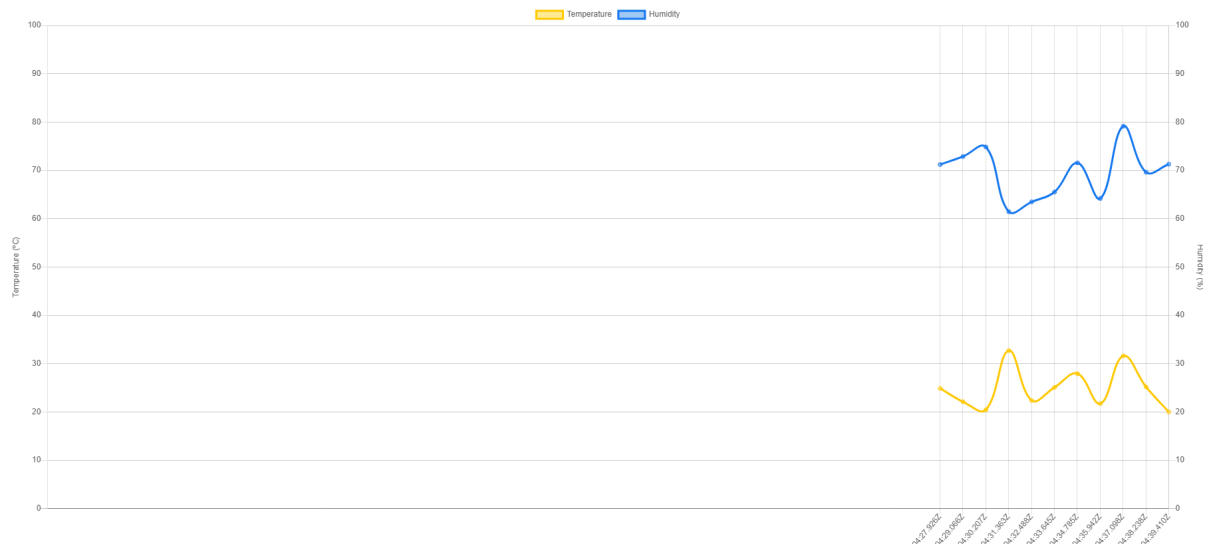
R.: Optei por instalar o pacote dotenv e implementar o mesmo no server.js, assim já fiz o upload do git com as variáveis de ambiente automaticamente configuradas.

8. Com todo o ambiente configurado, agora é possível visualizar a coleta de dados através de um navegador. Para visualizar os dados do hub IoT, abra uma página da web acessando `http://localhost:3000` em seu navegador.

1 device

Dispositivo_RPG0027

Temperature & Humidity Real-time Data



9. Um plano de Serviço de Aplicativo define um conjunto de recursos de computação para a execução de um aplicativo hospedado no Serviço de Aplicativo. Para criar um novo plano do Serviço de Aplicativo usando o nível gratuito do Windows, utilize o comando **az appservice plan create**. Certifique-se de usar o mesmo grupo de recursos em que seu hub IoT está localizado. O nome do plano do serviço pode conter letras maiúsculas e minúsculas, números e hifens. **Optei por configurar o default group no Azure Shell neste passo.**

```

Read more about the command in reference docs
PS /home/andre> az appservice plan create --name VIACCANPLAN --sku FREE
Readonly attribute name will be ignored in class <class 'azure.mgmt.web.v2023_01_01.models._models_py3.AppServicePlan'>
Resource provider 'Microsoft.Web' used by this operation is not registered. We are registering for you.
Registration succeeded.
{
  "elasticScaleEnabled": false,
  "extendedLocation": null,
  "freeOfferExpirationTime": null,
  "geoRegion": "East US",
  "hostingEnvironmentProfile": null,
  "hyperV": false,
  "id": "/subscriptions/e0c84085-1170-4c3d-9087-cf0e7abab7d0/resourceGroups/RPG0027/providers/Microsoft.Web/serverfarms/VIACCANPLAN",
  "isSpot": false,
  "isXenon": false,
  "kind": "app",
  "kubeEnvironmentProfile": null,
  "location": "eastus",
  "maximumElasticWorkerCount": 1,
  "maximumNumberOfWorkers": 0,
  "name": "VIACCANPLAN",
  "numberOfSites": 0,
  "numberOfWorkers": 0,
  "perSiteScaling": false,
  "provisioningState": "Succeeded",
  "reserved": false,
  "resourceGroup": "RPG0027",
  "sku": {
    "capabilities": null,
    "capacity": 0,
    "family": "F",
    "locations": null,
    "name": "F1",
    "size": "F1",
    "skuCapacity": null,
    "tier": "Free"
  },
  "spotExpirationTime": null,
  "status": "Ready",
  "subscription": "e0c84085-1170-4c3d-9087-cf0e7abab7d0",
  "tags": null,
  "targetWorkerCount": 0,
  "targetWorkerSizeId": 0,
  "type": "Microsoft.Web/serverfarms",
  "workerTierName": null,
  "zoneRedundant": false
}
PS /home/andre>

```

11. Utilize o comando **az webapp config appsettings set** para adicionar configurações de aplicativo referentes às variáveis de ambiente que especificam a cadeia de conexão do hub IoT e o grupo de consumidores do hub de eventos.

```
az webapp config appsettings set -n YOUR_WEB_APP_NAME -g  
YOUR_RESOURCE_GROUP_NAME --settings  
EventHubConsumerGroup=YOUR_CONSUMER_GROUP_NAME
```

```
PS /home/andre> az webapp config set -n viaccanapp --web-sockets-enable true
```

```
{
  "acrUseManagedIdentityCreds": false,
  "acrUserManagedIdentityId": null,
  "alwaysOn": false,
  "apiDefinition": null,
  "apiManagementConfig": null,
  "appCommandLine": "",
  "appSettings": null,
  "autoHealEnabled": false,
  "autoHealRules": null,
  "autoSwapSlotName": null,
  "azureStorageAccounts": {},
  "connectionStrings": null,
  "cors": null,
  "defaultDocuments": [
    "Default.htm",
    "Default.html",
    "Default.asp",
    "index.htm",
    "index.html",
    "iisstart.htm",
    "default.aspx",
    "index.php",
    "hostingstart.html"
  ],
  "detailedErrorLoggingEnabled": false,
  "documentRoot": null,
  "elasticWebAppScaleLimit": 0,
  "experiments": {
    "rampUpRules": []
  },
  "ftpsState": "FtpsOnly",
  "functionAppScaleLimit": null,
  "functionsRuntimeScaleMonitoringEnabled": false,
  "handlerMappings": null,
  "healthCheckPath": null,
  "http20Enabled": true,
```

```

bad more about the command in reference docs
PS /home/andre> az webapp update -n viaccanapp --https-only true

"availabilityState": "Normal",
"clientAffinityEnabled": true,
"clientCertEnabled": false,
"clientCertExclusionPaths": null,
"clientCertMode": "Required",
"cloningInfo": null,
"containerSize": 0,
"customDomainVerificationId": "CE9CE6B7595407F95DBAD0A6B5514B70D5F31A7341152FC359D8BE2ED035F72E",
"dailyMemoryTimeQuota": 0,
"daprConfig": null,
"defaultHostName": "viaccanapp.azurewebsites.net",
"enabled": true,
"enabledHostNames": [
  "viaccanapp.azurewebsites.net",
  "viaccanapp.scm.azurewebsites.net"
],
"extendedLocation": null,
"hostNameSslStates": [
  {
    "certificateResourceId": null,
    "hostType": "Standard",
    "ipBasedSslResult": null,
    "ipBasedSslState": "NotConfigured",
    "name": "viaccanapp.azurewebsites.net",
    "sslState": "Disabled",
    "thumbprint": null,
    "toUpdate": null,
    "toUpdateIpBasedSsl": null,
    "virtualIPv6": null,
    "virtualIp": null
  },
  {
    "certificateResourceId": null,
    "hostType": "Repository",
    "ipBasedSslResult": null,
    "ipBasedSslState": "NotConfigured",
    "name": "viaccanapp.scm.azurewebsites.net",
    "sslState": "Disabled",
    "thumbprint": null,
  }
]

```

13. Execute o comando a seguir para consultar o status do seu aplicativo web e assegurar-se de que está em execução, lembrando de ajustar os parâmetros.

```

PS /home/andre> az webapp show -n viaccanapp --query state
"Running"

```

14. Após isso, acesse <https://<seu nome de aplicativo web>.azurewebsites.net> em um navegador. Você verá uma página da web semelhante àquela que visualizou ao executar o aplicativo web localmente. Supondo que o dispositivo esteja em execução e enviando dados, você deverá observar um gráfico em tempo real das 50 leituras mais recentes de temperatura e humidade enviadas pelo dispositivo.

