



Estácio

Missão Prática | Nível 4 | Mundo 4

RPG0026 – Tirando proveito da nuvem para projetos de software!

Rian Joseph Ramos Felizardo - 202202923931

POLO BARREIRO - Belo Horizonte, MG

Nível 4 – Tirando proveito da nuvem para projetos de software! – 2025.1 – 5º Semestre Letivo

Objetivo da Prática

- Demonstrar habilidade na criação e gerenciamento de recursos na Nuvem Azure, adquirindo conhecimento sobre a estrutura básica da plataforma Azure
- Utilizar efetivamente o portal Azure para criar e configurar uma Máquina Virtual (VM), demonstrando compreensão dos recursos e suas funções.
- Configurar regras de rede e grupos de segurança, adquirindo conhecimento sobre a estrutura das regras de rede na Nuvem Azure.
- Importar um arquivo .bacpac para um banco de dados no Banco de Dados SQL do Azure.
- Criar e configurar um aplicativo web no Azure, demonstrando compreensão do mecanismo de hospedagem e implantação de aplicativos web.

Contextualização

A LogiMove Transportes, uma empresa renomada no setor de logística de transporte, enfrenta desafios significativos em sua operação diária. A coordenação entre agendadores, despachantes, motoristas e clientes é crucial, mas o processo atual, baseado em formulários de papel e comunicações telefônicas, tem se mostrado ineficiente.

Problemas Identificados:

- Excesso de papelada, muitas vezes incompleta ou sem assinaturas.
- Dificuldade na disponibilidade dos distribuidores, resultando em atrasos.
- Motoristas frequentemente parados, esperando por coordenação.
- Atraso nas remessas, afetando negativamente a satisfação do cliente e os negócios recorrentes.

Solução Proposta:

Para resolver esses desafios, a empresa decide migrar para um sistema digital, substituindo formulários de papel e chamadas telefônicas por documentos digitais e comunicação online. A implementação de autenticação digital permitirá uma coordenação e acompanhamento eficazes das remessas, acessíveis via navegador web ou aplicativo móvel.

Projeto de Banco de Dados:

Como líder de desenvolvimento de software, você propõe o desenvolvimento de um protótipo que inclui a criação de um banco de dados no Azure SQL. Este banco de dados será projetado para armazenar informações cruciais, incluindo:

- Dados dos motoristas: informações pessoais, qualificações, histórico de viagens.
- Informações dos clientes: detalhes de contato, histórico de pedidos, preferências.
- Detalhes dos pedidos: informações do pedido, status, cronograma de entrega.
- O protótipo servirá como base para o aplicativo de produção futuro. Portanto, as escolhas tecnológicas feitas agora devem ser escaláveis e compatíveis com as soluções finais.

Objetivos do Projeto:

- Desenvolver um banco de dados robusto e seguro no Azure SQL
- Desenvolver um banco de dados robusto e seguro no Azure SQL.
- Garantir que o banco de dados possa escalar conforme a empresa cresce.
- Facilitar a integração com outras plataformas e serviços.

O projeto visa transformar radicalmente a maneira como a LogiMove Transportes opera, aumentando a eficiência, reduzindo atrasos e melhorando a satisfação do cliente. A adoção de uma solução baseada em Azure SQL é um passo significativo em direção à digitalização e modernização das operações da empresa.

Diagramação Banco de Dados:

Tabela **Clients**


Clients	
ClientID 	int
Nome	varchar(100) NN
Empresa	varchar(100)
Endereco	varchar(200)
Contato	varchar(50)

Tabela **Drivers**


Drivers	
DriverID 	int
Nome	varchar(100) NN
CNH	varchar(20) NN
Endereco	varchar(200)
Contato	varchar(50)

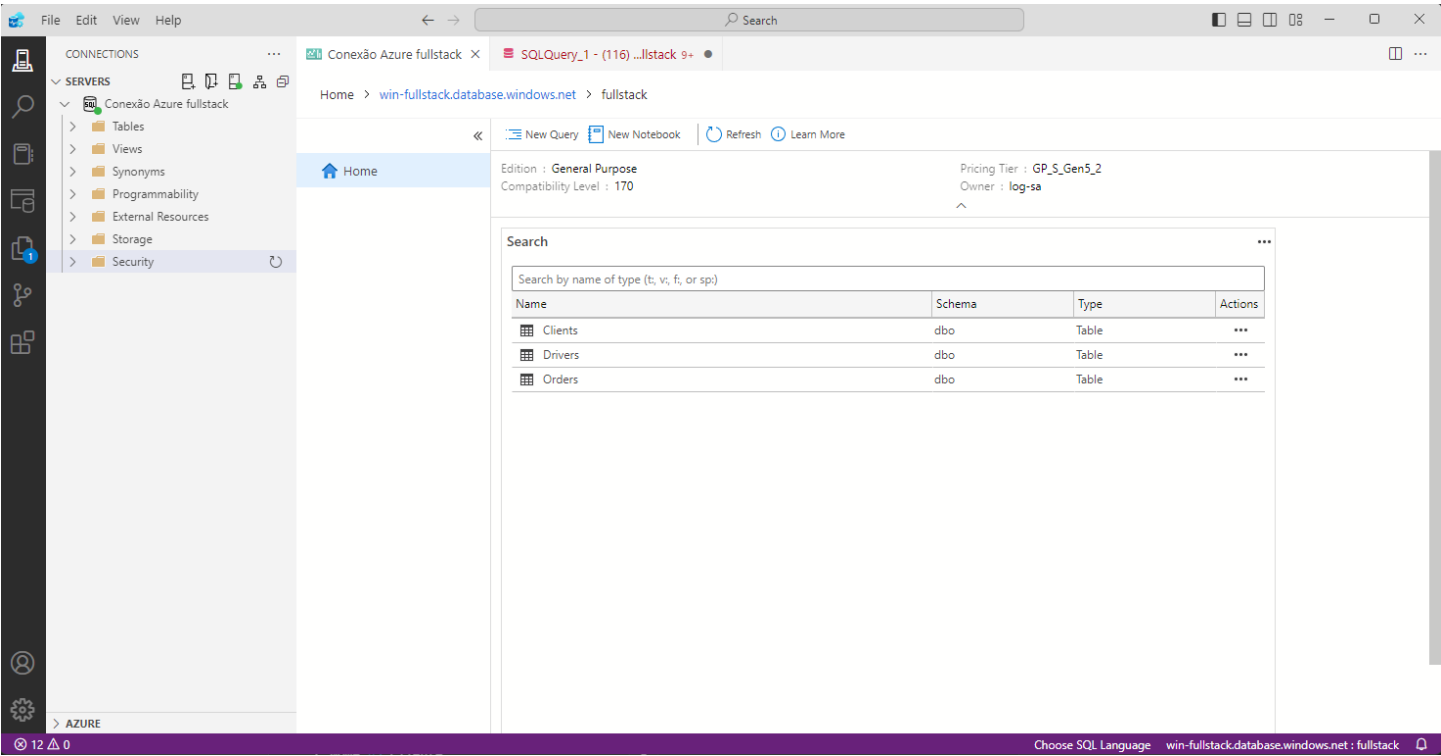
Tabela **Orders**

Orders	
OrderID	int
ClientID	int NN
DriverID	int NN
DetalhesPedido	varchar(500)
DataEntrega	date
Status	varchar(50)

Código inserido no **dbdiagram.io**

```
1  Table Drivers {
2    DriverID int [pk, increment]
3    Nome varchar(100) [not null]
4    CNH varchar(20) [not null]
5    Endereco varchar(200)
6    Contato varchar(50)
7  }
8
9  Table Clients {
10   ClientID int [pk, increment]
11   Nome varchar(100) [not null]
12   Empresa varchar(100)
13   Endereco varchar(200)
14   Contato varchar(50)
15 }
16
17 Table Orders {
18   OrderID int [pk, increment]
19   ClientID int [not null, ref: > Clients.ClientID]
20   DriverID int [not null, ref: > Drivers.DriverID]
21   DetalhesPedido varchar(500)
22   DataEntrega date
23   Status varchar(50)
24 }
25
```

Conexão feita dentro do Azure Data Studio, Tabelas Criadas:



Ao concluir a atividade espera-se que o aluno provisione um banco de dados para a LogiMove Transportes e que esteja funcional e otimizado para operações diárias e preparado para escalabilidade futura. Para isso as seguintes etapas deverão ser concluídas e apresentadas:

1. Configuração e Acesso ao Banco de Dados e Criação e Estruturação das Tabelas:

```
Run Cancel Disconnect Change Database: fullstack
1 CREATE TABLE Drivers (
2     DriverID INT PRIMARY KEY IDENTITY(1,1),
3     Nome VARCHAR(100) NOT NULL,
4     CNH VARCHAR(20) NOT NULL,
5     Endereco VARCHAR(200),
6     Contato VARCHAR(50)
7 );
8
9 CREATE TABLE Clients (
10    ClientID INT PRIMARY KEY IDENTITY(1,1),
11    Nome VARCHAR(100) NOT NULL,
12    Empresa VARCHAR(100),
13    Endereco VARCHAR(200),
14    Contato VARCHAR(50)
15 );
16
17 CREATE TABLE Orders (
18     OrderID INT PRIMARY KEY IDENTITY(1,1),
19     ClientID INT NOT NULL,
20     DriverID INT NOT NULL,
21     DetalhesPedido TEXT,
22     DataEntrega DATE,
23     Status VARCHAR(50),
24     FOREIGN KEY (ClientID) REFERENCES Clients(ClientID),
25     FOREIGN KEY (DriverID) REFERENCES Drivers(DriverID)
26 );
27
28 GO
```

2. Inserção e Gestão de Dados:

▶ Run Cancel Disconnect Change | Database: fullstack | Estimated Plan Enable Actual Plan Parse Enable SQLCMD To Notebook

1 INSERT INTO Drivers (Nome, CNH, Endereco, Contato) VALUES ('João Silva', '1234567890', 'Rua A, 100', '99999-9999');

2 INSERT INTO Drivers (Nome, CNH, Endereco, Contato) VALUES ('Marcos Paulo', '1234567890', 'Rua B, 100', '88888-8888');

3

4 INSERT INTO Clients (Nome, Empresa, Endereco, Contato) VALUES ('Cliente A', 'LogiMove', 'Av. B, 200', '77777-7777');

5 INSERT INTO Clients (Nome, Empresa, Endereco, Contato) VALUES ('Cliente B', 'MoveLog', 'Av. C, 300', '66666-6666');

6

7 INSERT INTO Orders (ClientID, DriverID, DetalhesPedido, DataEntrega, Status) VALUES (1, 1, 'Entrega de caixas', '2025-06-01', 'Pendente');

8 INSERT INTO Orders (ClientID, DriverID, DetalhesPedido, DataEntrega, Status) VALUES (2, 2, 'Entrega de madeiras', '2025-07-01', 'Feito');

9

10 GO

Messages

9:52:27 PM

Started executing query at Line 1

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)

(1 row affected)

Total execution time: 00:00:00.089

3. Execução e Validação de Consultas:

RunCancelDisconnectChangeDatabase:fullstackEstimated PlanEnable Actual PlanParseEnable SQLCMDTo Notebook

1234567891011

```
SELECT DriverID, Nome, CNH, Contato FROM Drivers;
GO

SELECT ClientID, Nome, Empresa, Contato FROM Clients;
GO

SELECT OrderID, ClientID, DriverID, DetalhesPedido, DataEntrega, Status FROM Orders;
GO
```

ResultsMessages

	DriverID	Nome	CNH	Contato
1	1	João Silva	1234567890	99999-9999
2	2	Marcos Paulo	1234567890	88888-88888

	ClientID	Nome	Empresa	Contato
1	1	Cliente A	LogiMove	77777-77777
2	2	Cliente B	MoveLog	66666-66666

	OrderID	ClientID	DriverID	DetalhesPedido	DataEntrega	Status
1	1	1	1	Entrega de caixas	2025-06-01	Pendente
2	2	2	2	Entrega de madeiras	2025-07-01	Feito

RunCancelDisconnectChangeDatabase:fullstackEstimated PlanEnable Actual PlanParseEnable SQLCMDTo Notebook

1234567891011121314

```
SELECT
  o.OrderID,
  c.Nome AS NomeCliente,
  c.Empresa,
  d.Nome AS NomeMotorista,
  o.DetalhesPedido,
  o.DataEntrega,
  o.Status
FROM Orders o
INNER JOIN Clients c ON o.ClientID = c.ClientID
INNER JOIN Drivers d ON o.DriverID = d.DriverID
ORDER BY o.DataEntrega ASC;
GO
```

ResultsMessages

	OrderID	NomeCliente	Empresa	NomeMotorista	DetalhesPedido	DataEntrega	Status
1	1	Cliente A	LogiMove	João Silva	Entrega de caixas	2025-06-01	Pendente
2	2	Cliente B	MoveLog	Marcos Paulo	Entrega de madeiras	2025-07-01	Feito

4. Operações CRUD Eficientes:

RunCancelDisconnectChangeDatabase:fullstackEstimated PlanEnable Actual PlanParseEnable SQLCMDTo Notebook

12345

```
UPDATE Drivers
SET Endereco = 'Rua C'
WHERE DriverID = 3;

GO
```

Messages

10:00:21 PM

Started executing query at Line 1

(1 row affected)

Total execution time: 00:00:00.046

Resultado:

Run

Cancel

Disconnect

Change

Database: fullstack

Estimated Plan

Enable Actual Plan

Parse

Enable SQLCMD

To Notebook

```
1 SELECT DriverID, Nome, CNH, Contato, Endereco FROM Drivers;
2 GO
3
4 SELECT ClientID, Nome, Empresa, Contato FROM Clients;
5 GO
6
7 SELECT OrderID, ClientID, DriverID, DetalhesPedido, DataEntrega, Status FROM Orders;
8 GO
9
```

Results

Messages

	DriverID	Nome	CNH	Contato	Endereco
1	1	João Silva	1234567890	99999-9999	Rua A, 100
2	2	Marcos Paulo	1234567890	88888-88888	Rua B, 100
3	3	Carlos Mendes	9876543210	91234-5678	Rua C

	ClientID	Nome	Empresa	Contato
1	1	Cliente A	LogiMove	77777-77777
2	2	Cliente B	MoveLog	66666-66666
3	3	Fernanda Lima	TransPro	98888-1111

	OrderID	ClientID	DriverID	DetalhesPedido	DataEntrega	Status
1	2	2	2	Entrega de madeiras	2025-07-01	Feito
2	3	1	3	Entrega de eletrônicos	2025-08-01	Pendente

Repositório

[rianjsp/missao-pratica-4-mundo-4](#): Este projeto foi desenvolvido com o objetivo de estruturar um banco de dados relacional funcional e escalável para a empresa fictícia LogiMove Transportes, utilizando T-SQL e hospedagem no Azure SQL Database.