Laboratório 01 Azure – Projeto PoccoBank I

São Paulo, 01 de julho de 2022

Autor	**Versão**	**Data**	**Descrição**
Renan Da Silva Ramos	1.0	01/07/2022	Criação do documento

1. INTRODUÇÃO

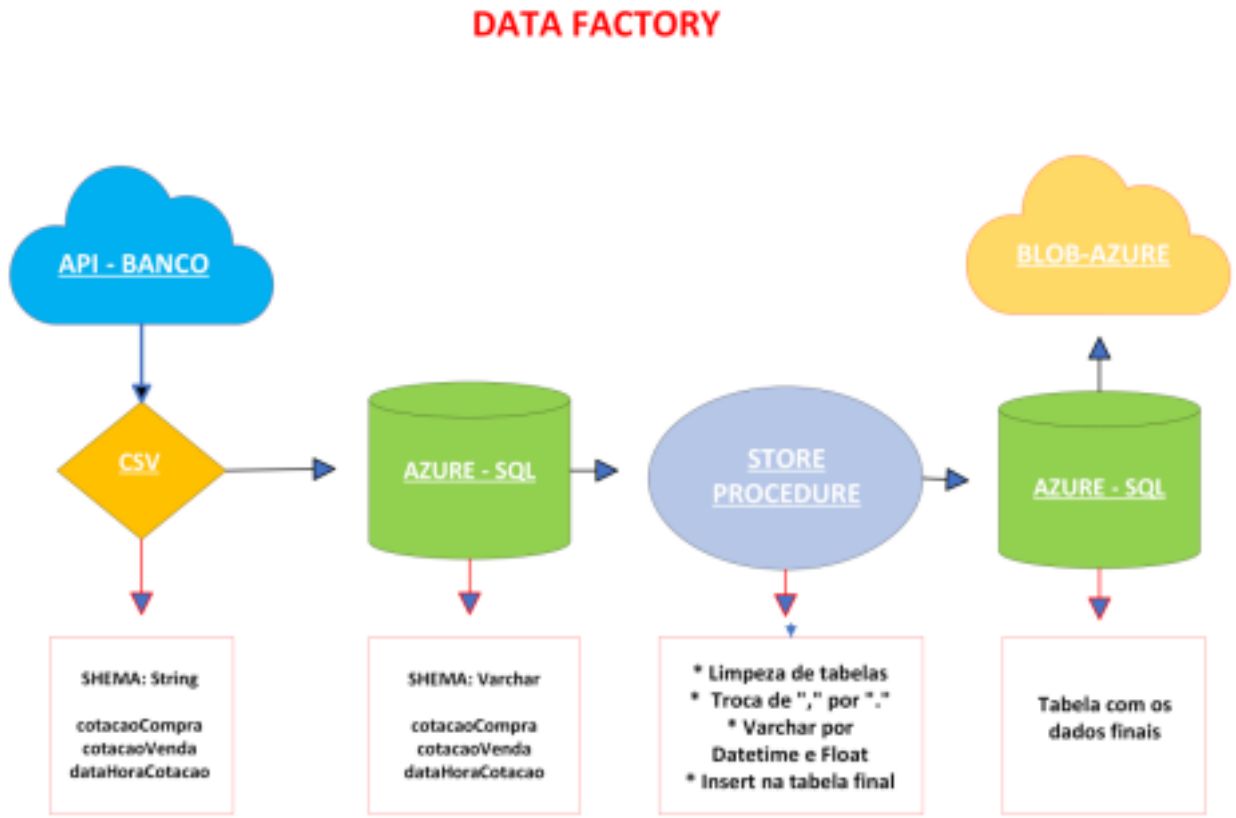
Este documento visa detalhar tecnicamente as etapas utilizadas no cumprimento do projeto do cliente Pocco Bank.

2. SOLICITAÇÃO

O cliente deseja obter o resultado das variações do dólar através de uma tabela simples e objetiva, contendo 3 colunas; cotacaoCompra, cotacaoDolar e dataHoraCotacao. Os valores serão carregados através de uma API em específico pelo método (GET). Após a obtenção dos dados da API, será implementado um pipeline no ambiente do (Data Factory Azure), juntamente com o (Azure Sql) na manipulação das tabelas e dados, e por fim será armazenado o valor final em formato parquet em um container (Blob Azure) para utilização do cliente.

3. MODELO DA ARQUITETURA

A figura abaixo representa a arquitetura e os principais processos para atender as demandas do cliente



4. DADOS INICIAIS - ORIGEM

A *API utilizada* como fonte dos dados originais, retorna uma tabela com 3 colunas. Estas colunas detalham o valor de compra, venda e data da cotação do dólar.Requisitamos essa tabela como um arquivo *CSV e* enviamos o seu retorno para o (Data Factor) através de (links de conexões) que a plataforma (Azure) oferece.

Abaixo, uma breve ilustração dos dados originais:

cotacaoCompra	cotacaoVenda	dataHoraCotacao
3,8589	3,8595	2019-01-02 13:04:46.568
3,7677	3,7683	2019-01-03 13:04:50.817
3,7621	3,7627	2019-01-04 13:06:29.332

Simultaneamente à extração da API, criamos uma tabela (STAGE) no (AZURE SQL) para guardar esses dados iniciais e poder fazer alterações futuras.

5. MANIPULAÇÃO DO AZURE SQL

Nos requisitos do projeto foi necessário a criação de duas tabelas, “dolar_stage” e “dolar_*final” no (Azure Sql).* A tabela “dolar_stage”, contém os dados (brutos) da extração da API. Já a tabela “dolar_final”, terá os dados finais manipulados conforme as especificações que foram dadas no objetivo do projeto. Também foi necessária a criação de uma procedure para a transformação dos dados.

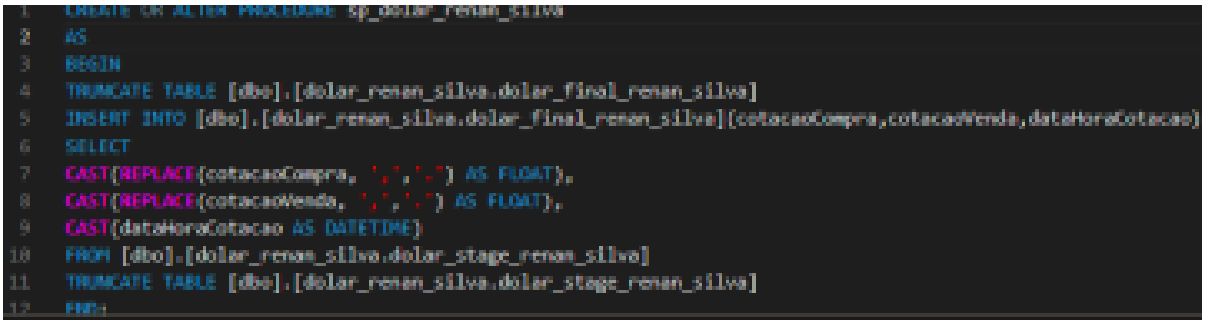
Schema da tabela dolar_stage :

Dolar_Stage	
cotacaoCompra	VARCHAR
cotacaoVenda	VARCHAR
dataHoraCotacao	VARCHAR

Schema da tabela dolar_final :

Dolar_Final	
cotacaoCompra	FLOAT
cotacaoVenda	FLOAT
dataHoraCotacao	DATETIME

Estrutura da Procedure :



```
CREATE OR ALTER PROCEDURE sp_dolar_renan_silva
AS
BEGIN
TRUNCATE TABLE [dbo].[dolar_renan_silva.dolar_final_renan_silva]
INSERT INTO [dbo].[dolar_renan_silva.dolar_final_renan_silva](cotacaoCompra,cotacaoVenda,dataHoraCotacao)
SELECT
CAST(REPLACE(cotacaoCompra, ',','.') AS FLOAT),
CAST(REPLACE(cotacaoVenda, ',','.') AS FLOAT),
CAST(dataHoraCotacao AS DATETIME)
FROM [dbo].[dolar_renan_silva.dolar_stage_renan_silva]
TRUNCATE TABLE [dbo].[dolar_renan_silva.dolar_stage_renan_silva]
END;
```

6. CONSTRUÇÃO DO DATA FACTORY (PROCESSAMENTO)

O Azure Data Factory é um administrador de todos os processos da pipeline. Basicamente nossa pipeline é dividida em 3 processos. O 1 processo é a requisição da API através do método get , onde será enviado os dados em csv para tabela inicial (dolar_stage). Já o 2 processo, a store-procedure entra em ação e faz a manipulação dos dados da tabela inicial, transformando a tipagem de dados e também substituindo o padrão de (,) por (.) nos valores numéricos e ao final disso é enviado para a tabela final (dolar_final). Por fim, temos o 3 processo, que levará a tabela final para o (container – blob do azure) onde conterá os dados tratados e finais para utilização do cliente.

Ilustração do 1 processo – obtendo API (GET) :

Ilustração do 2 processo – transformação dos dados (Store Procedure) :

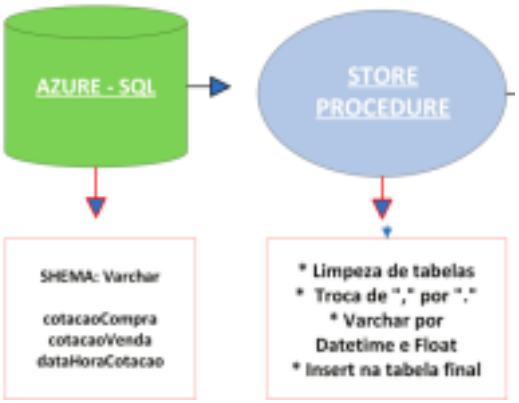
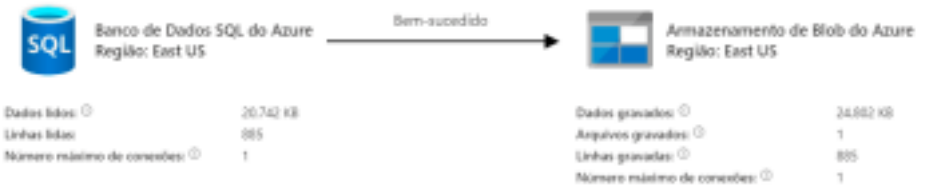


Ilustração do 3 processo – dados finais (dolar_stage) :

cotacaoCompra	cotacaoVenda	dataHoraCotacao
3.8589	3.8595	2019-01-02T13:04:46.5670000
3.7677	3.7683	2019-01-03T13:04:50.8170000
3.7821	3.7827	2019-01-04T13:06:26.3330000
3.7056	3.7062	2019-01-07T13:09:39.6530000

Armazenando os dados no (container – blob azure) :



Resultado final do pipeline no data factor :