

# Secure App with OpenTelemetry Observability

Arquitetura Software

2023/2024

Pompeu Costa, nº 103294

## Column Masking

A tabela *Client* contém três colunas mascaradas (PhoneNumber, DiagnosisDetails e TreatmentPlan) como mostrado na Figura 1 e na Figura 2. PhoneNumber tem quatro “x” seguido dos últimos três números do número de telefone. DiagnosisDetails e TreatmentPlan são mascarados com vários “x”.

O tipo de masking usado é *default* para DiagnosisDetails e para o TreatmentPlan. Para o PhoneNumber é usado *partial(0, "xxxx", 3)* como mostra a Figura 1.

```
CREATE TABLE [dbo].[Clients] (  
    [ClientID] INT  
    [UserID] INT  
    [FullName] NVARCHAR (MAX)  
    [PhoneNumber] NVARCHAR (MAX) MASKED WITH (FUNCTION = 'partial(0, "xxxx", 3)')  
    [MedicalRecordNumber] INT  
    [DiagnosisDetails] NVARCHAR (MAX) MASKED WITH (FUNCTION = 'default()')  
    [TreatmentPlan] NVARCHAR (MAX) MASKED WITH (FUNCTION = 'default()')  
    [AccessCode] NVARCHAR (MAX)
```

Figura 1: Código de construção da tabela *Client*

ClientID	FullName	PhoneNumber	MedicalRecordNumber	DiagnosisDetails	TreatmentPlan
11	ana	xxxx322	3	xxxx	xxxx

Figura 2: Exemplo de uma query com as colunas mascaradas

## Access Controls

Em modo *Debug* qualquer utilizador (Cliente ou Helpdesk) pode pedir dados de um cliente, seja dele mesmo (no caso de um cliente) ou de outro. No entanto, se o utilizador não estiver a pedir os seus próprios dados (no caso de um cliente), é necessário o código de acesso. Se o código de acesso não for dado ou for inválido então os campos mencionados acima são mascarados.

A veracidade do código introduzido é feita através de um *stored procedure* em SQL chamado *IsAllowedToUnmask* (script disponível na pasta extras). Se o código for válido (ou o utilizador está a pedir os seus próprios dados) então a *query* é executada com o utilizador *Client*, caso contrário é executada com o utilizador *HelpDesk*, cujo não tem acesso para ver os dados unmasked.

A *API* para obter os dados de clientes chama um *stored procedure* chamado *GetClientDetails* (script disponível na pasta extras) e tem de passar o ID do utilizador que está a pedir (ID da tabela User), o ID do cliente do qual quer os dados e, opcionalmente, o código de acesso. Este *sp* retorna os dados de um cliente se este existir e não devolve nada se não existir. As colunas são ou não mascaradas dependendo do resultado da chamada ao *sp IsAllowedToUnmask*.

## Test Cases

Para testar os *sps* criados em SQL foram criados quatro testes (disponíveis em AS\_Solo\_Proj1\_Tests -> TestDatabase.cs):

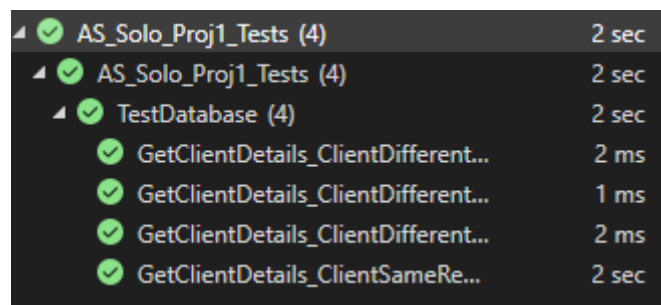
GetClientDetails\_ClientSameRequester\_NoCode\_Unmasked: simula um cliente a pedir os seus próprios dados

GetClientDetails\_ClientDifferentRequester\_NoCode\_Masked: simula um utilizador helpdesk a pedir dados de um cliente e não introduz código de acesso

GetClientDetails\_ClientDifferentRequester\_WrongCode\_Masked: simula um utilizador helpdesk a pedir dados de um cliente e introduz o código errado

GetClientDetails\_ClientDifferentRequester\_RightCode\_Unmasked: simula um utilizador helpdesk a pedir dados de um cliente e introduz o código certo

A Figura 3 mostra os quatro testes a passar.



AS_Solo_Proj1_Tests (4)	2 sec
AS_Solo_Proj1_Tests (4)	2 sec
TestDatabase (4)	2 sec
GetClientDetails_ClientDifferent...	2 ms
GetClientDetails_ClientDifferent...	1 ms
GetClientDetails_ClientDifferent...	2 ms
GetClientDetails_ClientSameRe...	2 sec

Figura 3: Testes a passar