

Tarefa 1

1.

1.1

Os problemas de interoperabilidade incluem a diferença nos nomes dos campos, como por exemplo **Cliente_ID**, **ID_Cliente** ou **UserID**, a redundância da informação, como acontece com **Nome**, **NomeCompleto** ou **Nome**, a existência de diferentes identificadores para o mesmo conceito, como **NIF**, **NumeroContribuinte** ou **TaxID**, e a falta de padronização na nomenclatura e na estrutura dos dados.

1.2

Tabela:

Cliente

ID_Cliente(PK)	Nome	Nif
----------------	------	-----

Tabela:

Alias_Cliente

ID_Alias(PK)	ID_Cliente(FK)	SistemaOrigem	Identificador_Origem
--------------	----------------	---------------	----------------------

Sendo o SistemaOrigem, um identificador de onde estão a vir os dados, podendo ser eles **Empresa**, **Cliente** ou **Loja Online**, e o Identificador_Origem sendo a identificação do cliente dentro de onde foi selecionado

Ex: O João na Empresa tem o identificador **nº53**, enquanto como Cliente tem um identificador como **nº1007**, e na Loja Online tem o **nº231**, irei identificar na tabela abaixo como o João com **ID_Cliente** da tabela **Cliente** como **1**.

Ex Tabela

Alias_Cliente

ID_Alias	ID_Cliente	SistemaOrigem	Identificador_Origem
1	1	Empresa	53
2	1	Cliente	1007
3	1	Loja Online	231

2.

- Relatórios de vendas em CSV

Estruturada (dados organizados em linhas e colunas).

- Comentários/menções em redes sociais

Não estruturada (texto livre, linguagem natural).

- Vídeos promocionais

Não estruturada (conteúdo multimédia sem estrutura definida).

- Dados demográficos do formulário

Semiestruturada (podem vir em JSON/XML, têm estrutura, mas não são totalmente relacionais).

- Emails

Semiestruturada (cabeçalhos estruturados, corpo de texto livre).

3.

3.1 Tabela Excel com notas dos alunos – **Estruturada**

3.2 Ficheiro Json com informações de Clientes – **Semiestruturada**

3.3 Video de vigilância – **Não estruturada**

3.4 Log de acesso ao servidor (.log) – **Semiestruturada**

3.5 Contrato em PDF digitalizado – **Não estruturada**

4.

4.1 Banco a analisar transações em tempo real – **Streaming, porque precisa de deteção imediata de fraudes.**

4.2 Universidade a consolidar dados no final do semestre - **Batch, porque os dados só são necessários periodicamente.**

4.3 Startup de meteorologia a disponibilizar previsões - **Serviços Web/API, porque outras aplicações vão aceder em tempo real via integração direta.**

5.

5.1 Hospital a consultar historial clínico em tempo real - **Serviços Web/API**

5.2 Empresa a atualizar Data Warehouse todas as noites – **Batch**

5.3 Aplicação a detetar uso suspeito de cartão de crédito – **Streaming**

5.4 Site a calcular custo de envio via DHL - **Serviços Web/API**