



Relatório Trabalho Prático

João Pedro Araújo

14363

Engenharia de Sistemas Informáticos

Integração de Sistemas de Informação

Professor: Luís Ferreira

2020/2021

Índice

Introdução.....	3
Arquitetura da solução.....	4
Conclusão	11
Bibliografia	12

Introdução

Unidade Curricular de Integração de Sistemas de Informação lecionada pelo docente Luís Ferreira, foi nos proposto um trabalho pratico para desenvolver alguns projetos de integração de serviços (ETL), utilizando algumas ferramentas fornecidas pelo professor como Pentaho Kettle ou Microsoft SQL Server Integration Services (MSSIS).

O papel de descrever este relatório tem objetivo de enriquecer e desenvolver competências, evoluir a nossa formação com projetos ETL desenvolvidos através da plataforma ou ferramenta Pentaho Kettle.

O projeto desenvolvido é constituído por três ficheiros *.ktr* e por um ficheiro *.kjb* gerados pelo Pentaho Kettle, toda esta informação é derivada de uma *.csv* com dados simulatórios de uma empresa fictícia, três ficheiros *.xsl* usados para fazer uma transformação XSL. Perante esta aplicação são gerados, com a ferramenta Penath Kettle, três ficheiros *.xml* e posteriormente dois ficheiros *.html*. Em busca de conhecimento e aprofundamento desta ferramenta é também convertido cada um dos ficheiros *.xml* em ficheiro *.json* e gerado um Excel com informação tabelada.

Arquitetura da solução

A Arquitetura é agrupada por um job (.kjb) e três transformation (.ktr). Nas transformation são extraídos os dados dentro de um ficheiro .csv e depois de alguns processos as três transformation geram um documento .xml e parcialmente gera um .json. No job são usadas as três transformation, no qual de seguida há uma transformação XSL do qual é recebido três ficheiros HTML.

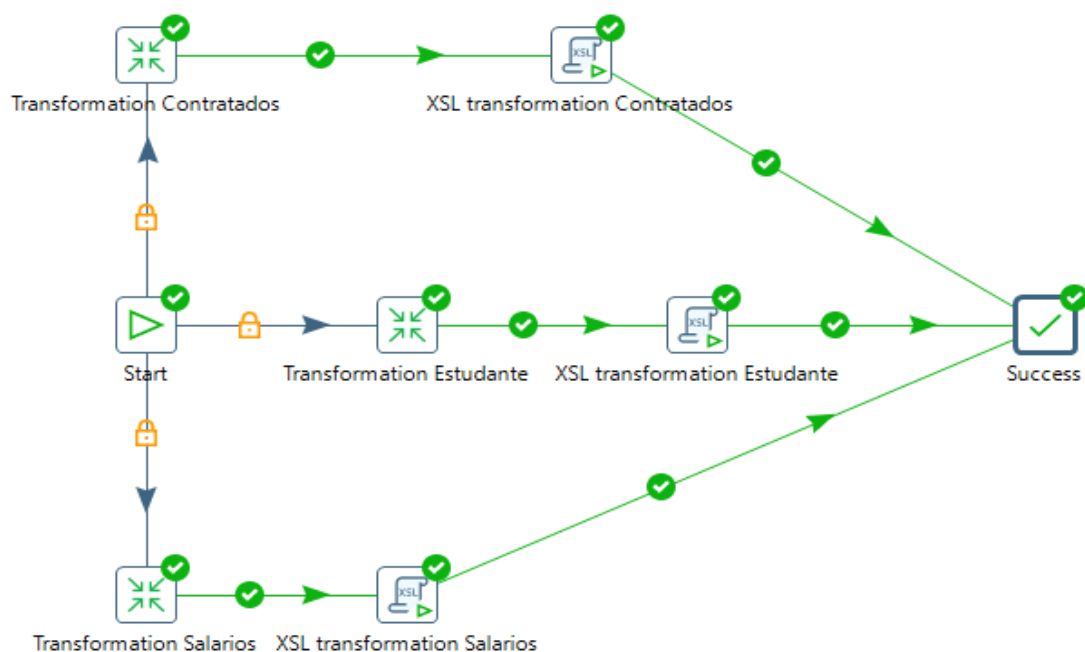


Figura 1 - Job

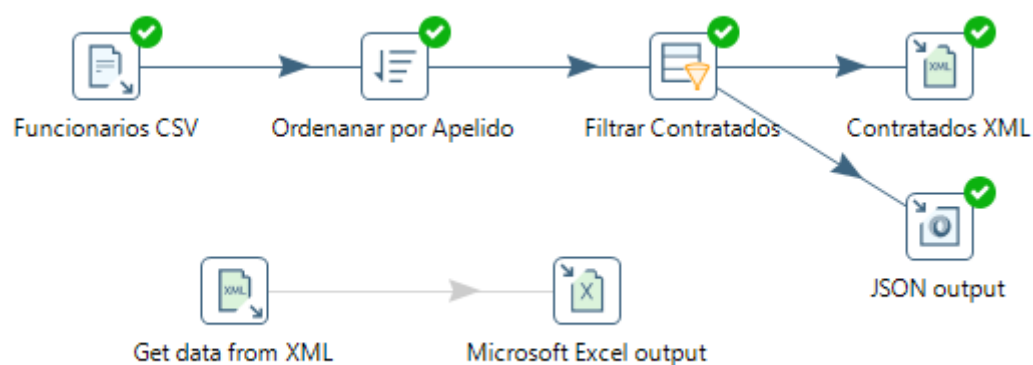


Figura 2 - Transformation para filtrar Contratados

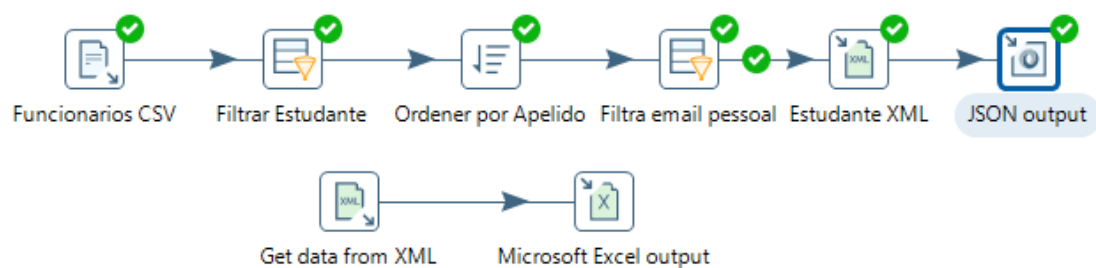


Figura 3 - Transformation para filtrar Estudantes

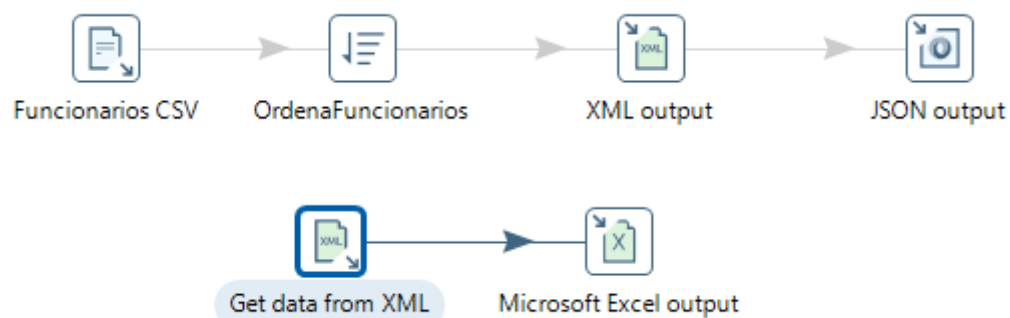


Figura 5 – Transformation para organizar salários / pagamentos

Processo de desenvolvimento do Projeto

Na inicialização do desenvolvimento do projeto foi necessário verificar e analisar qual e como o modo de gerar dados para o processo. Usando ocorrências diárias de empresas foi criado um .csv no qual passou-se a extrair dados.

O .csv que contém dados relativamente a cada funcionário da empresa que é designado Nome, Apelido, CC, Idade, Contacto, Email, Contratado, Estudante, IBAN, Salario.

#	Fieldname	Ascending	Case sensitive compare?	Sort based on current locale?	Collator Strength	Presorted?
1	Apelido	S	N	N	0	N

Figura 4 - Ordenar por Apelido

Na transformation contratados ordenamos primeiro pelo Apelido e de seguida uma filtragem dos contratados.

The condition:

Contratado = (String)

Figura 6 – Filtrar apenas contratados

O resultado desta operação obtivemos um ficheiro .xml – Contratados.xml e um ficheiro .json. De modo a que a informação esteja disponível e de fácil interpretação é gerado como output um excel tabelado.

Na transformation Estudante iniciamos com uma Filtragem de estudantes, de seguida uma ordenação por Apelido novamente e por fim uma filtragem aos emails no qual é usada uma expressão regular. Na conclusão da transformação estudante é dado como output um ficheiro .xml, um ficheiro .json e um excel para fácil interpretação dos dados.

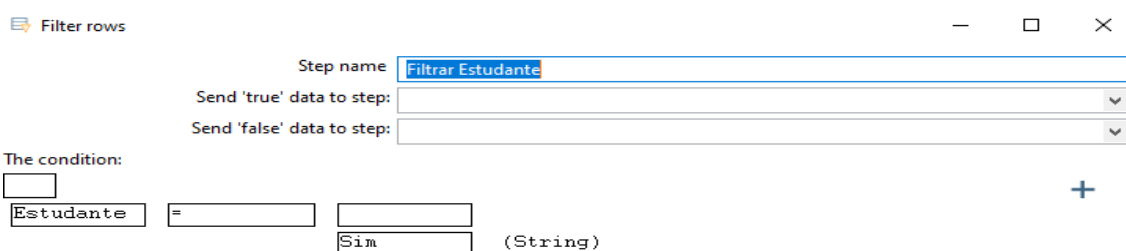


Figura 5 – Filtrar apenas Estudantes

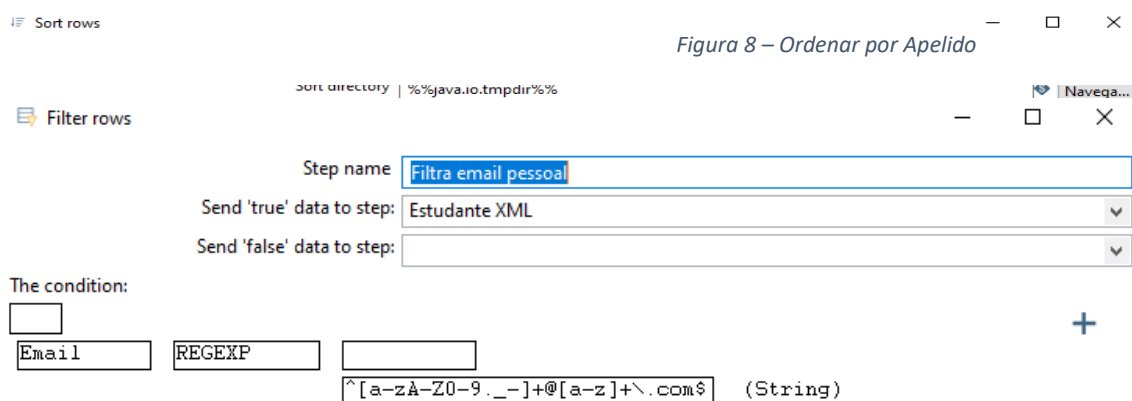


Figura 9 – Filtrar email pessoal

Na ultima transformation Salários apenas é ordenado todos os funcionários da empresa fictícia. Esta organização é para facilitar a identificação de cada funcionário e também para ver logo de imediato dados pessoais.

Na conclusão da transformação estudante é dado como output um ficheiro .xml, um ficheiro .json e um excel para fácil interpretação dos dados.

Sort rows

Nome do Step

Sort directory

TMP-file prefix

Sort size (rows in memory)

Free memory threshold (in %)

Compress TMP Files? ☐

Only pass unique rows? (verifies keys only) ☐

Fields :

#	Fieldname	Ascending	Case sensitive compare?	Sort based on current locale?	Collator Strength	Presorted?
1	Nome	S	N	N	0	N
2	Apelido	S	N	N	0	N
3	Cc	S	N	N	0	N
4	IBAN	S	N	N	0	N
5	Salario	S	N	N	0	N

Figura 10 - Ordenar Funcionarios

Três transformações XSL de forma a exportar os dados todos para três documentos .html.

Estes documentos são originados das três transformation realizadas no trabalho, os contratados, os estudantes e os salários.

```
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:output method="html" encoding="iso-8859-1"/>

<xsl:template match="/">
<html>
<body>
<table> Salarios </table>
<table border= "1">
<xsl:apply-templates select="Rows"/>
</table>
</body>
</html>
</xsl:template>

<xsl:template match="Rows">
<xsl:for-each select ="Row">
<xsl:if test="position()=1">
<xsl:attribute name="bgcolor">red</xsl:attribute>
<tr>
<xsl:for-each select="/*">
<td>
<xsl:value-of select="name()"/>
</td>
</xsl:for-each>
</tr>
</xsl:if>
<tr>
<xsl:if test="position() mod 2 =0">
<xsl:attribute name="bgcolor">Red</xsl:attribute>
</xsl:if>
<xsl:if test="position() mod 2 !=0">
<xsl:attribute name="bgcolor">White</xsl:attribute>
</xsl:if>
```

Figura 7 – XSL cria uma tabela em html

Salarios

Nome	Apelido	Cc	IBAN	Salario
Ana	Barros	35456765	PT50214215879654123654120	1414
Andre	Costa	53423123	PT50987456321741236987412	0595
Bruna	Estefania	13232345	PT50323232323256565889789	0655
Carolina	Barros	12313344	PT50214587896536523654789	0765
Iolanda	Faria	45545433	PT50214569896325698741236	0788
Joana	Barros	14475343	PT50123654789656989632531	0998
Joao	Barros	12312123	PT50123454312345232453155	0745
Leticia	Barros	33221155	PT50112200445588996633220	0852
Luis	Barros	12345642	PT50125478963214587410236	1200
Luisa	Cunha	22124567	PT50123654789654123656987	0850
Mariana	Silva	33445566	PT50147414741478965236520	1455
Megui	Barros	21271625	PT50214215698741236547853	0755
Miguel	Araujo	76545675	PT50000022114545478745200	0452
Pedro	Barros	14322121	PT5120102541254125412541	9581
Ricardo	Esteves	22334567	PT50214563210000102032101	0857
Sonia	Almeida	11324354	PT50121212125252526363699	0951
Soraia	Costa	08778908	PT50336655441122336655881	0741
Tiago	Barroso	14456456	PT50999887744556632145630	0963
Vera	costa	11113433	PT50352523655125364587895	0949

Figura 8 – HTML gerado apartir dos Salarios.xml

Conclusão

Com a elaboração deste trabalho foi muito enriquecedora e valorizada por mim pois consegui superar algumas dificuldades, pois nunca tinha utilizado nem sabia da existência desta ferramenta Kettle, aparecendo com o decorrer do trabalho e adquirir algum conhecimento significativamente no desenvolvimento de software, assim como na implementação de processos ETL.

Esta ferramenta nos dias de hoje permite-nos de uma forma intuitiva simplificar um input desconhecido e desorganizado para um output simplificado e fácil compreensão.

De um modo geral, posso dar este trabalho como concluído com sucesso, pois foram atingidos todos os objetivos propostos.

Embora não foram aplicadas todas as funcionalidades devido a desconhecimento ou não necessárias para esta elaboração, mas sim no futuro.

Bibliografia

<https://wiki.pentaho.com/display/EAI/XSL+Transformation>

<https://en.wikipedia.org/wiki/XSLT>

https://www.tutorialspoint.com/xslt/xslt_quick_guide.htm

Documentação Moodle