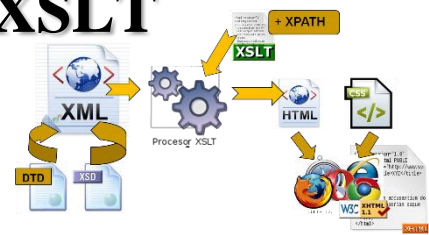




## 3º Trabalho Prático – XPath e XSLT

### *Social Tube XML*



#### Objetivos

Utilização de expressões XPath para obter informação em ficheiros XML. Utilização de transformações XSLT para gerar documentos em vários formatos (XHTML e XML) a partir de documentos já existentes. Utilização de um *framework* CSS.

#### Descrição:

Com base nos desenvolvimentos realizados nos trabalhos anteriores implemente os grupos que se seguem.

Os ficheiros para este trabalho prático devem ser colocados dentro de uma diretoria com nome **tps\tp3**.

## Grupo 1 – Expressões XPath

Indique as expressões XPath que com base no ficheiro XML, da rede social, permitam obter as seguintes informações:

1. Obter o número total de vídeos publicados;
2. Obter a lista de vídeos publicados por um utilizador;
3. Obter o número de listas subscritas por um utilizador;
4. Obter o título do último vídeo publicado por um utilizador;
5. Obter o número total de gostos que têm todos os vídeos de um utilizador;
6. Obter os títulos dos vídeos de que um utilizador indicou “Gosto”;
7. Obter os nomes dos utilizadores com mais de 5 vídeos publicados;
8. Listar os comentários (**1º nível**), do mês de fevereiro deste ano, do(s) vídeo(s) com um dado título (exº: “A day made of glass”);
9. Obter os nomes dos utilizadores que fizeram comentários, em qualquer nível de profundidade, num vídeo com um dado título.

Nota: A indicação do utilizador em questão deverá ser realizada pelo seu *username*. As datas devem ser descritas segundo este exemplo: “2016/01/31”.

## Grupo 2 – Transformações XSLT

Neste grupo pretende-se a realização de transformações XSLT, de XML para XHTML e de XML para XML.

### a) Transformações de XML para XHTML

Pretende-se que para cada alínea que seguidamente se apresenta se crie uma transformação XSLT para gerar um documento XHTML, baseado nos resultados do grupo anterior, com a informação pedida e que utilize estilos suportados em ficheiros externos CSS, que a permita a sua visualização num *browser*. Deve-se utilizar a nomenclatura, já estipulada, de endereços das páginas dos utilizadores, dos vídeos e das listas de vídeos. Deve-se colocar sempre *links* para a informação em todos os locais possíveis. Sugere-se a estruturação dos *templates* de forma a permitir a sua reutilização (`xsl:import` ou `xsl:include`), para tal sugere-se também que eles sejam concebidos como *templates* com “mode” ou com “name” e parâmetros se necessário. As transformadas baseadas em nome de utilizador, identificador (id), etc., devem, elas, receber esse(s) valor(es) por parâmetro (`xsl:param`). Deve-se consultar os exemplos disponibilizados no moodle.

1. Transformadas para informação reduzida, com base no id – para utilizadores, vídeos publicados e listas de vídeos (playlists). Criar uma página para testar cada uma destas transformadas e que receba o identificador como argumento.
2. Transformadas para informação média, com base no id – para utilizadores, vídeos publicados e listas de vídeos (playlists). Estas transformadas devem utilizar as transformadas de formato reduzido para a informação que se queira mais sintética. Criar uma página para testar cada uma destas transformadas e que receba o identificador como argumento. Criar as páginas que permitam mostrar as listagens geral com todos os utilizadores, todos os vídeos publicados, e todas as listas de vídeos (playlists) (em formato médio).
3. Transformadas para informação detalhada, com base no id – para utilizadores, vídeos publicados e listas de vídeos (playlists). Estas transformadas devem utilizar as transformadas de formato médio ou reduzido para a informação que se queira mais sintética. Criar uma página para testar cada uma destas transformadas e que receba o identificador como argumento.
4. Transformada para gerar a página de apresentação do *site*, que deve conter links para todas as páginas mencionadas na seguinte alínea;
5. Uma transformação (com `xsl:result-document`) que produza todo o *site*, chamando transformadas para tal. Deve produzir:
  - ❏ Página de apresentação;
  - ❏ Listas de utilizadores, vídeos e listas de vídeos;
  - ❏ Páginas detalhadas de cada utilizador, de cada vídeo e de cada lista de vídeos;
  - ❏ Página de estatísticas com os resultados do grupo 1 deste TP;
  - ❏ Página com *links* para as páginas resultantes das alíneas seguintes e gerar essas páginas.

6. Transformada para gerar uma página com os utilizadores, em formato médio, que tenham publicado mais de 5 vídeos
7. Transformada para gerar uma página com os utilizadores, em formato reduzido, ordenados decrescentemente pelo número de vídeos publicados, em que a cor de fundo alterne em uma cor e outra (como se faz nas listas)
8. Transformada para gerar uma página com os vídeos, em formato médio, que tenham mais de 6 gostos diretamente neles, apresentando-os por ordem decrescente do nº de gostos;
9. Transformada para gerar uma página com as listas de vídeos, em formato médio, e para cada uma delas o último vídeo publicado na lista, em formato médio;
10. Transformada que gere uma página com os vídeos, em formato médio, que tenham mais de 10 comentários (incluindo comentários aos comentários) e os respetivos comentários.

## b) Transformação de XML para XML

Pretende-se uma transformação XSLT que, a partir do conteúdo do ficheiro `SocialTube.xml`, gere um novo documento XML com a lista dos utilizadores e os seus vídeos publicados. O documento XML gerado deverá ser validado pela definição DTD (`userInfo.dtd`) que se apresenta na Listagem 1.

### Listagem 1 – Ficheiro DTD `userInfo.dtd` de validação de dados

```
<!ELEMENT XMLTubeUserInfo (user*)>
<!ELEMENT user (name, publishedVideo*)>
<!ATTLIST user username ID #REQUIRED>
<!ATTLIST user birthdate CDATA #IMPLIED>
<!ATTLIST user numberOfComments CDATA #REQUIRED>
<!ELEMENT name (#PCDATA)>
<!ELEMENT publishedVideo (titulo, videoLink)>
<!ATTLIST publishedVideo id ID #REQUIRED>
<!ELEMENT titulo (#PCDATA)>
<!ELEMENT videoLink (#PCDATA)>
```

## Grupo 3 – Framework CSS

Numa directoria `tps/tp3/frameworkCSS`, coloque todos os ficheiros de forma a apresentar uma solução equivalente à do grupo 2, mas utilizando um *framework* CSS (Bootstrap, w3.css, Materialize, ...) em todo o *site*.

### Agenda de execução do trabalho:

1ª semana: grupo 1, expressões XPath;

2ª semana: grupo 2, a) transformadas XSLT [1-3];

3ª semana: grupo 2, a) transformadas XSLT [4-10];

4ª semana: grupo 2, b) transformação XML para XML; grupo 3; e relatório.

### Sobre a realização do trabalho e do relatório

A realização do trabalho é em grupo e conjunta, ficando cada um dos elementos do grupo corresponsável técnico da solução adotada. O relatório deve ser entregue em formato PDF e a sua conceção e entrega deve ser acordada com o respetivo professor.

**Data limite para entregar o relatório (e ficheiros): dia 16 de dezembro de 2018.**

Os docentes da Unidade Curricular

*António Teófilo, Carlos Gonçalves e Diogo Remédios*