

Processamento de Linguagens  
(3º ano de Licenciatura em Engenharia  
Informática)  
**Trabalho Prático 1**  
Relatório de Desenvolvimento

André Gerlades (67673)      Patrícia Barros (67665)  
Sandra Ferreira (67709)

31 de Março de 2015

## **Resumo**

Este relatório descreve todo o processo de desenvolvimento e decisões tomadas para a realização do primeiro trabalho prático da Unidade Curricular de Processamento de Linguagens.

O problema a resolver consiste no desenvolvimento de um Filtro de Texto, utilizando Flex, para processar ficheiros XML com informações sobre fotografias e gerar um álbum HTML a partir delas.

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Análise e Especificação</b>	<b>3</b>
2.1	Descrição informal do problema . . . . .	3
2.2	Especificação do Requisitos . . . . .	3
2.2.1	Dados . . . . .	3
2.2.2	Pedidos . . . . .	3
2.2.3	Relações . . . . .	3
<b>3</b>	<b>Concepção/desenho da Resolução</b>	<b>4</b>
3.1	Expressões Regulares . . . . .	4
3.2	Estados da Aplicação . . . . .	4
3.3	Módulos da Aplicação . . . . .	4
3.4	Estruturas de Dados . . . . .	4
<b>4</b>	<b>Codificação e Testes</b>	<b>5</b>
4.1	Alternativas, Decisões e Problemas de Implementação . . . . .	5
4.2	Testes realizados e Resultados . . . . .	5
<b>5</b>	<b>Conclusão</b>	<b>6</b>
<b>A</b>	<b>Código do Programa</b>	<b>7</b>

# Capítulo 1

## Introdução

A resolução deste trabalho prático passa pelo desenvolvimento de um Filtro de Texto em Flex para gerar ficheiros em HTML. Para isso utilizamos as técnicas leccionadas nas aulas da Unidade Curricular de Processamento de Linguagens. Pretendemos portanto com este trabalho aprimorar as nossas capacidades de escrever *Expressões Regulares (ER)* e também a nossa experiência na utilização da linguagem de programação C.

Neste relatório apresentamos todos os passos e decisões tomadas durante todo o processo, descrevemos as estruturas criadas para guardar o texto extraído pelo filtro e também uma descrição do produto final (em HTML) obtido com a utilização do filtro criado por nós.

### Estrutura do Relatório

explicar como está organizado o documento, referindo os capítulos existentes e a sua articulação explicando o conteúdo de cada um. No capítulo 2 faz-se uma análise detalhada do problema proposto de modo a poder-se especificar as entradas, resultados e formas de transformação.

etc. . . .

No capítulo 5 termina-se o relatório com uma síntese do que foi dito, as conclusões e o trab futuro

## Capítulo 2

# Análise e Especificação

### 2.1 Descrição informal do problema

Dos enunciados propostos o que escolhemos seguir foi o "Museu da Pessoa - tratamento de fotografias".

Neste enunciado foi-nos pedido que realizássemos um Filtro de Texto em Flex para, através de um ficheiro XML com informações (onde, quando, quem e facto) sobre fotografias que fazem parte do Museu da Pessoa, gerar um álbum em HTML em que sejam mostradas as fotografias em questão, por ordem cronológica, sendo que o título de cada fotografia será o campo "facto". Foi-nos ainda pedido que apresentássemos um índice no início com o nome de todas as pessoas retratadas.

### 2.2 Especificação do Requisitos

#### 2.2.1 Dados

#### 2.2.2 Pedidos

#### 2.2.3 Relações

## Capítulo 3

# Concepção/desenho da Resolução

Para a resolução do problema foi necessário definirmos *Expressões Regulares*, estados no Flex e ainda estruturas de dados para guardar a informação filtrada. Nos próximos capítulos iremos descrever detalhadamente cada um destes pontos.

### 3.1 Expressões Regulares

As expressões regulares definidas visam extrair do ficheiro XML as informações necessárias para a construção da página HTML: foto, quando, onde, quem e facto. Após analisar cuidadosamente a estrutura dos ficheiros XML em questão chegamos às seguintes *Expressões Regulares*:

**QUEM** <"(?i:QUEM)>"  
**FACTO** <"(?i:FACTO)>"  
**FOTO** <"(?i:FOTO)(?i:FICHEIRO)"= "  
**QUANDO** <"(?i:QUANDO)(?i:DATA)"= "  
**ONDE** ????

### 3.2 Estados da Aplicação

### 3.3 Módulos da Aplicação

### 3.4 Estruturas de Dados

## Capítulo 4

# Codificação e Testes

### 4.1 Alternativas, Decisões e Problemas de Implementação

### 4.2 Testes realizados e Resultados

Mostram-se a seguir alguns testes feitos (valores introduzidos) e os respectivos resultados obtidos:

## Capítulo 5

# Conclusão

Síntese do Documento.

Estado final do projecto; Análise crítica dos resultados.

Trabalho futuro.



## Apêndice A

# Código do Programa

---

```
1  %{
2  /* Declaracoes C diversas */
3  #include <stdio.h>
4  #include <string.h>
5  #include "html.h"
6  char * foto;
7  char * facto;
8  char * quem;
9  char * quando;
10 %}
11
12
13 %x QUEM1
14 %x FACTO1
15 %x FOTO1
16 %x QUANDO1
17
18 %%
19 "<"(?i:QUEM)">" {
20     BEGIN QUEM1; }
21 <QUEM1>"<" { BEGIN INITIAL; }
22 <QUEM1>[^<]+ { quem
23     = strdup(yytext); printf("%s\n", quem);}
24
25
26
27 "<"(?i:FACTO)">" {
28     BEGIN FACTO1; }
29 <FACTO1>"<" { BEGIN INITIAL; }
30 <FACTO1>[^<]+ {
31     facto = strdup(yytext); printf("%s\n", facto); }
32
33
34
35 "<"(?i:FOTO)" "<"(?i:FICHEIRO)"=\\>" { BEGIN FOTO1; }
```

```

28 <FOTO1>"\"
    { BEGIN INITIAL; }
29 <FOTO1>[^\"]+
    = strdup(yytext); printf("%s\n", foto); } { foto
30
31 "<(?i:QUANDO)" "(?i:DATA)"=\" { BEGIN QUANDO1; }
32 <QUANDO1>"\" {
    BEGIN INITIAL; }
33 <QUANDO1>[^\"]+ {
    quando = strdup(yytext); printf("%s\n", quando); }
34
35 .|\n
    { ; }
36 %%
37
38 int yywrap()
39 {
40     return(1);
41 }
42
43 int main()
44 {
45     FILE * html;
46     html = fopen("new.html","w");
47     newHeader("Museu da Pessoa", html);
48     yylex();
49     return 0;
50 }

```

---

# Bibliografia