

**UNIVERSIDADE FEDERAL RURAL DO RIO DE JANEIRO**  
**1ª PROVA WEB SEMÂNTICA – 2016-2**  
**PROFESSOR SERGIO SERRA**  
**CURSO SISTEMAS DE INFORMAÇÃO**

**Aluno: Aléxis da Silva de Oliveira – 201439001-9**

**1ª Questão (1.5 pontos)**

Os documentos XML são válidos? Estão bem formados? Caso estejam malformados faça os ajustes necessários e descreva tecnicamente quais os erros encontrados e descreva, em detalhes, TODOS os erros e TODOS os ajustes que você fez no código dos 3 documentos.

R.:

Os documentos XML não são válidos porque não foram construídos de acordo com as regras de um esquema (DTD ou XML Schema). Mais especificamente, eles não possuem um DTD ou XML Schema associado.

Nenhum dos três documentos XML estão bem formatados. Os erros de formatação são os seguintes:

Documento1:

- Uma tag “<nNF>” da linha 10 não foi fechada da maneira correta, faltou a barra (“/”) de fechamento. Esse problema fez com que o erro fosse acusado na tag “</ide>”, que é o elemento pai de “<nNF>”.
- O elemento “<nro>” na linha 29 não foi fechado corretamente, nome do elemento foi escrito de maneira errada.
- O elemento “<retirada>” na linha 60 não foi iniciado corretamente, pois possui um erro de sintaxe. O sinal maior-que (>) que compõe o elemento não foi colocado.
- O elemento “<xBairro>” na linha 65 não foi fechado corretamente, pois o nome do elemento foi escrito de maneira incorreta (“<xBarrio>”).
- O elemento “<transporta>” na linha 195 não foi fechado corretamente, pois o nome do elemento foi escrito de maneira errada começando com letra maiúscula. Como ajuste a letra maiúscula foi trocada pela mesma letra, só que minúscula.
- O elemento “<Reference>” na linha 233 não foi fechado corretamente, pois o nome do elemento foi escrito de forma errada com uma das letras maiúscula. Como ajuste a letra foi trocada por uma minúscula.
- O elemento “<SignaturValue>” na linha 237 não foi fechado corretamente, pois o nome do elemento foi escrito de maneira errada com uma letra a mais. Como ajuste a letra foi removida.

Documento2:

- O elemento “<mod>” não foi fechado corretamente, pois sua tag de fechamento foi escrita com uma letra a mais (“<modi>”). Como ajuste, a letra foi removida.
- O elemento “<xCl>” não foi fechado corretamente, pois sua tag de fechamento foi escrita com uma letra a mais (“<xCpl>”). Como ajuste, a letra foi removida.
- O elemento “<ICMS00>” não foi fechado corretamente, pois era esperada a tag de fechamento “</ICMS00>” e não a tag “<ICMS01>” usada. Como ajuste, a tag foi trocada para “</ICMS00>”.
- O elemento “<DigesValue>” não foi fechado corretamente, pois era esperada a tag de fechamento “</DigesValue>” e não a tag “<DigestValue>” usada. Como ajuste, a tag foi trocada para “</DigesValue>”.
- As tags “<DigesValue>” e “<Transforms>” não foram fechadas.

Documento3:

- A tag “<finFe>” não foi fechada corretamente, pois era esperada a tag de fechamento

- “</finFe>” e não a tag “</finNFe>” usada. Como ajuste, a tag foi trocada para “</finFe>”.
- A tag “<CNP>” não foi fechada corretamente, pois era esperada a tag de fechamento “</CNP>” e não a tag “</CNPJ>” usada. Como ajuste, a tag foi trocada para “</CNP>”.
- A tag “<xCpl>” na linha 64 não foi fechada corretamente, pois era esperada a tag de fechamento “</xCpl>” e não a tag “</xCp2>” usada. Como ajuste, a tag foi trocada para “</xCpl>”.
- A tag “<cProd>” na linha 82 não foi fechada corretamente, pois era esperada a tag de fechamento “</cProd>” e não a tag “</cProD>” usada, porque o XML é case sensitive. Como ajuste, a tag foi trocada para “</cProd>”.
- A tag “<xCpl>” na linha 64 não foi fechada corretamente, pois era esperada a tag de fechamento “</xCpl>” e não a tag “</xCp2>” usada. Como ajuste, a tag foi trocada para “</xCpl>”.
- A tag “<COFINS>” na linha 121 não foi fechada corretamente, pois era esperada a tag de fechamento “</COFINS>” e não a tag “</COFI NS>” usada com um espaço entre uma das letras do nome. Como ajuste, a tag foi trocada para “</COFINS>”.
- A tag “<ICMS>” na linha 149 não foi fechada corretamente, pois era esperada a tag de fechamento “</ICMS>” e não a tag “</ICMs>” usada, porque o XML é case sensitive. Como ajuste, a tag foi trocada para “</ICMS>”.
- A tag “<imposto>” não foi fechada corretamente na linha 166, pois era esperada a tag de fechamento “</imposto>” e não a tag “</Imposto>” usada com a primeira letra maiúscula, porque o XML é case sensitive. Como ajuste, a tag foi trocada para “</imposto>”.
- A tag “<nLacre>” na linha 214 não foi fechada corretamente, pois era esperada a tag de fechamento “</nLacre>” e não a tag “</n Lacre>” usada com um espaço entre uma das letras do nome. Como ajuste, a tag foi trocada para “</nLacre>”.
- .

## 2ª Questão (1.5 pontos)

Uma vez ajustados os documentos, desenvolva um ÚNICO DTD e um ÚNICO XML Schema que **validem os três documentos simultaneamente**. Explique em detalhes quais são as principais diferenças entre o DTD e o XML Schema que você desenvolveu. Explique os principais campos do seu arquivo XML Schema.

R.:

----- DTD -----

```
<!ELEMENT xCI ( #PCDATA )>
<!ELEMENT modFrete ( #PCDATA )>
<!ELEMENT tpNF ( #PCDATA )>
<!ELEMENT pesoL ( #PCDATA )>
<!ELEMENT cMunFG ( #PCDATA )>
<!ELEMENT procEmi ( #PCDATA )>
<!ELEMENT vUnTrib ( #PCDATA )>
<!ELEMENT pesoB ( #PCDATA )>
<!ELEMENT mod ( #PCDATA )>
<!ELEMENT tpAmb ( #PCDATA )>
<!ELEMENT marca ( #PCDATA )>
<!ATTLIST det
<!ELEMENT vICMS ( #PCDATA )>
<!ELEMENT vII ( #PCDATA )>
<!ELEMENT xMun ( #PCDATA )>
<!ELEMENT IE ( #PCDATA )>
```

<!ELEMENT vFrete ( #PCDATA )>  
<!ELEMENT DigestMetod EMPTY>  
<!ATTLIST DigestMetod  
<!ELEMENT DigestMethod EMPTY>  
<!ATTLIST DigestMethod  
<!ELEMENT tpImp ( #PCDATA )>  
<!ELEMENT cEANtrib EMPTY>  
<!ELEMENT nro ( #PCDATA )>  
<!ELEMENT pCOFINS ( #PCDATA )>  
<!ELEMENT xProd ( #PCDATA )>  
<!ELEMENT X509Certificate ( #PCDATA )>  
<!ELEMENT CEP ( #PCDATA )>  
<!ELEMENT xCpl ( #PCDATA )>  
<!ELEMENT cDV ( #PCDATA )>  
<!ELEMENT cUF ( #PCDATA )>  
<!ELEMENT vBC ( #PCDATA )>  
<!ELEMENT SignatureMethod EMPTY>  
<!ATTLIST SignatureMethod  
<!ELEMENT pPIS ( #PCDATA )>  
<!ELEMENT vBCST ( #PCDATA )>  
<!ELEMENT cPais ( #PCDATA )>  
<!ELEMENT qTrib ( #PCDATA )>  
<!ELEMENT vProd ( #PCDATA )>  
<!ELEMENT cMun ( #PCDATA )>  
<!ELEMENT tpEmis ( #PCDATA )>  
<!ELEMENT SignatureValue ( #PCDATA )\*>  
<!ELEMENT SignaturValue ( #PCDATA )\*>  
<!ELEMENT vST ( #PCDATA )>  
<!ELEMENT cNF ( #PCDATA )>  
<!ELEMENT indPag ( #PCDATA )>  
<!ELEMENT placa ( #PCDATA )>  
<!ELEMENT nLacre ( #PCDATA )>  
<!ELEMENT qCom ( #PCDATA )>  
<!ELEMENT xBairro ( #PCDATA )>  
<!ELEMENT vUnCom ( #PCDATA )>  
<!ELEMENT fone ( #PCDATA )>  
<!ELEMENT xNome ( #PCDATA )>  
<!ELEMENT vPIS ( #PCDATA )>  
<!ELEMENT orig ( #PCDATA )>  
<!ELEMENT verProc ( #PCDATA )>  
<!ELEMENT Transform EMPTY>  
  
<!ELEMENT esp ( #PCDATA )>  
<!ELEMENT xLgr ( #PCDATA )>  
<!ELEMENT dSaiEnt ( #PCDATA )>  
<!ELEMENT RNTC ( #PCDATA )>  
<!ELEMENT UF ( #PCDATA )>  
<!ELEMENT dEmi ( #PCDATA )>  
<!ELEMENT qVol ( #PCDATA )>  
<!ELEMENT CFOP ( #PCDATA )>  
<!ELEMENT infAdFisco ( #PCDATA )>  
<!ELEMENT nNF ( #PCDATA )>

```

<!ELEMENT vNF ( #PCDATA )>
<!ELEMENT xPais ( #PCDATA )>
<!ELEMENT uTrib ( #PCDATA )>
<!ELEMENT nVol ( #PCDATA )>
<!ELEMENT vDesc ( #PCDATA )>
<!ELEMENT CNPJ ( #PCDATA )*>
<!ELEMENT CNP ( #PCDATA )*>
<!ELEMENT cEAN EMPTY>
<!ELEMENT cProd ( #PCDATA )>
<!ELEMENT vCOFINS ( #PCDATA )>
<!ELEMENT vSeg ( #PCDATA )>
<!ELEMENT finNFe ( #PCDATA )*>

<!ELEMENT total ( ICMSTot )>
<!ELEMENT natOp ( #PCDATA )>
<!ELEMENT vOutro ( #PCDATA )>
<!ELEMENT xFant ( #PCDATA )>
<!ELEMENT CST ( #PCDATA )>
<!ELEMENT CanonicalizationMethod EMPTY>
<!ATTLIST CanonicalizationMethod
<!ELEMENT modBC ( #PCDATA )>
<!ELEMENT xEnder ( #PCDATA )>
<!ELEMENT serie ( #PCDATA )>
<!ELEMENT uCom ( #PCDATA )>
<!ELEMENT vIPI ( #PCDATA )>
<!ELEMENT pICMS ( #PCDATA )>
<!ELEMENT COFINSAliq ( CST , vBC , pCOFINS , vCOFINS )>
<!ELEMENT COFINS ( COFINSAliq )>
<!ELEMENT PISAliq ( CST , vBC , pPIS , vPIS )>
<!ELEMENT PIS ( PISAliq )>
<!ELEMENT ICMS00 ( orig , CST , modBC , vBC , pICMS , vICMS )>
<!ELEMENT ICMS ( ICMS00 )>
<!ELEMENT imposto ( ICMS , PIS , COFINS )>
<!ELEMENT prod ( cProd , cEAN , xProd , CFOP , uCom , qCom , vUnCom , vProd , cEANtrib ,
uTrib , qTrib , vUnTrib )>
<!ELEMENT enderDest ( xLgr , nro , xCpl , xBairro , cMun , xMun , UF , CEP , cPais , xPais ,
fone )>
<!ELEMENT det ( prod , imposto )>
<!ELEMENT dest ((CNPJ | CNP) , xNome , enderDest , IE )>
<!ELEMENT entrega ( CNPJ , xLgr , nro , xCpl , xBairro , cMun , xMun , UF )>
<!ELEMENT retirada ( CNPJ , xLgr , nro , (xCpl | xCpl) , xBairro , cMun , xMun , UF )>
<!ELEMENT enderEmit ( xLgr , nro , xCpl , xBairro , cMun , xMun , UF , CEP , cPais , xPais ,
fone )>
<!ELEMENT emit ( CNPJ , xNome , xFant , enderEmit , IE )>
<!ELEMENT ide ( cUF , cNF , natOp , indPag , mod , serie , nNF , dEmi , dSaiEnt , tpNF , cMunFG ,
tpImp , tpEmis , cDV , tpAmb , (finNFe | finFe) , procEmi , verProc )>
<!ELEMENT X509Data ( X509Certificate )>
<!ELEMENT KeyInfo ( X509Data )>
<!ELEMENT Transforms ( Transform+ )>

<!ELEMENT SignedInfo ( CanonicalizationMethod , SignatureMethod , Reference )>

```

```

<!ELEMENT Signature ( SignedInfo , (SignatureValue | SignaturValue)* , KeyInfo* )>
<!--ATTLIST Signature
<!--ELEMENT transp ( modFrete , transporta , veicTransp , reboque , vol )>
<!--ELEMENT reboque ( placa , UF , RNTC )>
<!--ELEMENT transporta ( CNPJ , xNome , IE , xENDER , xMun , UF )>
<!--ELEMENT infNFe ( ide , emit , dest , retirada , entrega , det+ , total , transp , infAdic* )>
<!--ATTLIST infNFe
<!--ELEMENT NFe ( infNFe , Signature )>
<!--ATTLIST NFe
<!--ELEMENT ICMSTot ( vBC , vICMS , vBCST , vST , vProd , vFrete , vSeg , vDesc , vII , vIPI ,
vPIS , vCOFINS , vOutro , vNF )>
<!--ELEMENT lacres ( nLacre )>
<!--ELEMENT Reference ( Transforms , (DigestMetod | DigestMethod) , (DigesValue |
DigestValue) )>
<!--ATTLIST Reference
<!--ELEMENT veicTransp ( placa , UF , RNTC )>
<!--ELEMENT infAdic ( infAdFisco )>
<!--ELEMENT vol ( qVol , esp , marca , nVol , pesoL , pesoB , lacres )>

```

#### ----- XML SCHEMA -----

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:element name="xCI" type="xs:string"/>
  <xs:element name="modFrete" type="xs:string"/>
  <xs:element name="tpNF" type="xs:string"/>
  <xs:element name="pesoL" type="xs:string"/>
  <xs:element name="cMunFG" type="xs:string"/>
  <xs:element name="procEmi" type="xs:string"/>
  <xs:element name="vUnTrib" type="xs:string"/>
  <xs:element name="pesoB" type="xs:string"/>
  <xs:element name="mod" type="xs:string"/>
  <xs:element name="tpAmb" type="xs:string"/>
  <xs:element name="marca" type="xs:string"/>
  <xs:element name="vICMS" type="xs:string"/>
  <xs:element name="vII" type="xs:string"/>
  <xs:element name="xMun" type="xs:string"/>
  <xs:element name="IE" type="xs:string"/>
  <xs:element name="vFrete" type="xs:string"/>
  <xs:element name="DigestMetod">
    <xs:complexType>
      <xs:attribute name="Algorithm" use="required"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="DigestMethod">
    <xs:complexType>
      <xs:attribute name="Algorithm"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="tpImp" type="xs:string"/>
  <xs:element name="cEANtrib">
    <xs:complexType/>

```

```

</xs:element>
<xs:element name="nro" type="xs:string"/>
<xs:element name="pCOFINS" type="xs:string"/>
<xs:element name="xProd" type="xs:string"/>
<xs:complexType name="X509Certificate">
  <xs:sequence>
    <xs:element ref="X509Certificate"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="X509Certificate" type="xs:string"/>
<xs:element name="CEP" type="xs:string"/>
<xs:element name="xCpl" type="xs:string"/>
<xs:element name="cDV" type="xs:string"/>
<xs:element name="cUF" type="xs:string"/>
<xs:element name="vBC" type="xs:string"/>
<xs:element name="SignatureMethod">
  <xs:complexType>
    <xs:attribute name="Algorithm" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="pPIS" type="xs:string"/>
<xs:element name="vBCST" type="xs:string"/>
<xs:element name="cPais" type="xs:string"/>
<xs:element name="qTrib" type="xs:string"/>
<xs:element name="vProd" type="xs:string"/>
<xs:element name="cMun" type="xs:string"/>
<xs:element name="tpEmis" type="xs:string"/>
<xs:element name="SignatureValue" type="xs:string"/>
<xs:element name="SignaturValue" type="xs:string"/>
<xs:element name="vST" type="xs:string"/>
<xs:element name="cNF" type="xs:string"/>
<xs:element name="indPag" type="xs:string"/>
<xs:element name="placa" type="xs:string"/>
<xs:complexType name="nLacre">
  <xs:sequence>
    <xs:element ref="nLacre"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="nLacre" type="xs:string"/>
<xs:element name="qCom" type="xs:string"/>
<xs:element name="xBairro" type="xs:string"/>
<xs:element name="vUnCom" type="xs:string"/>
<xs:element name="fone" type="xs:string"/>
<xs:element name="xNome" type="xs:string"/>
<xs:element name="vPIS" type="xs:string"/>
<xs:element name="orig" type="xs:string"/>
<xs:element name="verProc" type="xs:string"/>
<xs:element name="Transform">
  <xs:complexType>
    <xs:attribute name="Algorithm" use="required"/>
  </xs:complexType>
</xs:element>

```

```

<xs:element name="esp" type="xs:string"/>
<xs:element name="xLgr" type="xs:string"/>
<xs:element name="dSaiEnt" type="xs:string"/>
<xs:element name="RNTC" type="xs:string"/>
<xs:element name="UF" type="xs:string"/>
<xs:element name="dEmi" type="xs:string"/>
<xs:element name="qVol" type="xs:string"/>
<xs:element name="CFOP" type="xs:string"/>
<xs:complexType name="infAdFisco">
  <xs:sequence>
    <xs:element ref="infAdFisco"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="infAdFisco" type="xs:string"/>
<xs:element name="nNF" type="xs:string"/>
<xs:element name="vNF" type="xs:string"/>
<xs:element name="xPais" type="xs:string"/>
<xs:element name="uTrib" type="xs:string"/>
<xs:element name="nVol" type="xs:string"/>
<xs:element name="vDesc" type="xs:string"/>
<xs:element name="CNPJ" type="xs:string"/>
<xs:element name="CNP" type="xs:string"/>
<xs:element name="cEAN">
  <xs:complexType/>
</xs:element>
<xs:element name="cProd" type="xs:string"/>
<xs:element name="vCOFINS" type="xs:string"/>
<xs:element name="vSeg" type="xs:string"/>
<xs:element name="finNFe" type="xs:string"/>
<xs:element name="finFe" type="xs:string"/>
<xs:element name="total" type="ICMSTot"/>
<xs:element name="natOp" type="xs:string"/>
<xs:element name="vOutro" type="xs:string"/>
<xs:element name="xFant" type="xs:string"/>
<xs:element name="CST" type="xs:string"/>
<xs:element name="CanonicalizationMethod">
  <xs:complexType>
    <xs:attribute name="Algorithm" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="modBC" type="xs:string"/>
<xs:element name="xEnder" type="xs:string"/>
<xs:element name="serie" type="xs:string"/>
<xs:element name="uCom" type="xs:string"/>
<xs:element name="vIPI" type="xs:string"/>
<xs:element name="pICMS" type="xs:string"/>
<xs:complexType name="COFINSAliq">
  <xs:sequence>
    <xs:element ref="COFINSAliq"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="COFINSAliq">

```

```

<xs:complexType>
  <xs:sequence>
    <xs:element ref="CST"/>
    <xs:element ref="vBC"/>
    <xs:element ref="pCOFINS"/>
    <xs:element ref="vCOFINS"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="COFINS" type="COFINSAliq"/>
<xs:complexType name="PISAliq">
  <xs:sequence>
    <xs:element ref="PISAliq"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="PISAliq">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="CST"/>
      <xs:element ref="vBC"/>
      <xs:element ref="pPIS"/>
      <xs:element ref="vPIS"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="PIS" type="PISAliq"/>
<xs:complexType name="ICMS00">
  <xs:sequence>
    <xs:element ref="ICMS00"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ICMS00">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="orig"/>
      <xs:element ref="CST"/>
      <xs:element ref="modBC"/>
      <xs:element ref="vBC"/>
      <xs:element ref="pICMS"/>
      <xs:element ref="vICMS"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ICMS" type="ICMS00"/>
<xs:element name="imposto">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ICMS"/>
      <xs:element ref="PIS"/>
      <xs:element ref="COFINS"/>
    </xs:sequence>
  </xs:complexType>

```



```

</xs:element>
<xs:element name="prod">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="cProd"/>
      <xs:element ref="cEAN"/>
      <xs:element ref="xProd"/>
      <xs:element ref="CFOP"/>
      <xs:element ref="uCom"/>
      <xs:element ref="qCom"/>
      <xs:element ref="vUnCom"/>
      <xs:element ref="vProd"/>
      <xs:element ref="cEANtrib"/>
      <xs:element ref="uTrib"/>
      <xs:element ref="qTrib"/>
      <xs:element ref="vUnTrib"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="enderDest">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="xLgr"/>
      <xs:element ref="nro"/>
      <xs:element ref="xCpl"/>
      <xs:element ref="xBairro"/>
      <xs:element ref="cMun"/>
      <xs:element ref="xMun"/>
      <xs:element ref="UF"/>
      <xs:element ref="CEP"/>
      <xs:element ref="cPais"/>
      <xs:element ref="xPais"/>
      <xs:element ref="fone"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="det">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="prod"/>
      <xs:element ref="imposto"/>
    </xs:sequence>
    <xs:attribute name="nItem" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="dest">
  <xs:complexType>
    <xs:sequence>
      <xs:choice>
        <xs:element ref="CNPJ"/>
        <xs:element ref="CNP"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        <xs:element ref="xNome"/>
        <xs:element ref="enderDest"/>
        <xs:element ref="IE"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="entrega">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="CNPJ"/>
            <xs:element ref="xLgr"/>
            <xs:element ref="nro"/>
            <xs:element ref="xCpl"/>
            <xs:element ref="xBairro"/>
            <xs:element ref="cMun"/>
            <xs:element ref="xMun"/>
            <xs:element ref="UF"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="retirada">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="CNPJ"/>
            <xs:element ref="xLgr"/>
            <xs:element ref="nro"/>
            <xs:choice>
                <xs:element ref="xCi"/>
                <xs:element ref="xCpl"/>
            </xs:choice>
            <xs:element ref="xBairro"/>
            <xs:element ref="cMun"/>
            <xs:element ref="xMun"/>
            <xs:element ref="UF"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="enderEmit">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="xLgr"/>
            <xs:element ref="nro"/>
            <xs:element ref="xCpl"/>
            <xs:element ref="xBairro"/>
            <xs:element ref="cMun"/>
            <xs:element ref="xMun"/>
            <xs:element ref="UF"/>
            <xs:element ref="CEP"/>
            <xs:element ref="cPais"/>
            <xs:element ref="xPais"/>
            <xs:element ref="fone"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>

```

```

</xs:complexType>
</xs:element>
<xs:element name="emit">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="CNPJ"/>
      <xs:element ref="xNome"/>
      <xs:element ref="xFant"/>
      <xs:element ref="enderEmit"/>
      <xs:element ref="IE"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ide">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="cUF"/>
      <xs:element ref="cNF"/>
      <xs:element ref="natOp"/>
      <xs:element ref="indPag"/>
      <xs:element ref="mod"/>
      <xs:element ref="serie"/>
      <xs:element ref="nNF"/>
      <xs:element ref="dEmi"/>
      <xs:element ref="dSaiEnt"/>
      <xs:element ref="tpNF"/>
      <xs:element ref="cMunFG"/>
      <xs:element ref="tpImp"/>
      <xs:element ref="tpEmis"/>
      <xs:element ref="cDV"/>
      <xs:element ref="tpAmb"/>
      <xs:choice>
        <xs:element ref="finNFe"/>
        <xs:element ref="finFe"/>
      </xs:choice>
      <xs:element ref="procEmi"/>
      <xs:element ref="verProc"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:complexType name="X509Data">
  <xs:sequence>
    <xs:element ref="X509Data"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="X509Data" type="X509Certificate"/>
<xs:element name="KeyInfo" type="X509Data"/>
<xs:element name="Transforms">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" ref="Transform"/>
    </xs:sequence>
  </xs:complexType>

```

```

    </xs:complexType>
  </xs:element>
  <xs:element name="DigesValue">
    <xs:complexType mixed="true">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" ref="Transforms"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="DigestValue" type="xs:string"/>
  <xs:element name="SignedInfo">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="CanonicalizationMethod"/>
        <xs:element ref="SignatureMethod"/>
        <xs:element ref="Reference"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Signature">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="SignedInfo"/>
        <xs:choice minOccurs="0" maxOccurs="unbounded">
          <xs:element ref="SignatureValue"/>
          <xs:element ref="SignaturValue"/>
        </xs:choice>
        <xs:element minOccurs="0" maxOccurs="unbounded" ref="KeyInfo"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="transp">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="modFrete"/>
        <xs:element ref="transporta"/>
        <xs:element ref="veicTransp"/>
        <xs:element ref="reboque"/>
        <xs:element ref="vol"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="reboque">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="placa"/>
        <xs:element ref="UF"/>
        <xs:element ref="RNTC"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="transporta">

```

```

<xs:complexType>
  <xs:sequence>
    <xs:element ref="CNPJ"/>
    <xs:element ref="xNome"/>
    <xs:element ref="IE"/>
    <xs:element ref="xENDER"/>
    <xs:element ref="xMun"/>
    <xs:element ref="UF"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="infNFe">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ide"/>
      <xs:element ref="emit"/>
      <xs:element ref="dest"/>
      <xs:element ref="retirada"/>
      <xs:element ref="entrega"/>
      <xs:element maxOccurs="unbounded" ref="det"/>
      <xs:element ref="total"/>
      <xs:element ref="transp"/>
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="infAdic"/>
    </xs:sequence>
    <xs:attribute name="Id" use="required"/>
    <xs:attribute name="versao" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="NFe">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="infNFe"/>
      <xs:element ref="Signature"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:complexType name="ICMSTot">
  <xs:sequence>
    <xs:element ref="ICMSTot"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ICMSTot">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="vBC"/>
      <xs:element ref="vICMS"/>
      <xs:element ref="vBCST"/>
      <xs:element ref="vST"/>
      <xs:element ref="vProd"/>
      <xs:element ref="vFrete"/>
      <xs:element ref="vSeg"/>
      <xs:element ref="vDesc"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

    <xs:element ref="vII"/>
    <xs:element ref="vIPI"/>
    <xs:element ref="vPIS"/>
    <xs:element ref="vCOFINS"/>
    <xs:element ref="vOutro"/>
    <xs:element ref="vNF"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="lacsres" type="nLacre"/>
<xs:element name="Reference">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Transforms"/>
      <xs:choice>
        <xs:element ref="DigestMetod"/>
        <xs:element ref="DigestMethod"/>
      </xs:choice>
      <xs:choice>
        <xs:element ref="DigesValue"/>
        <xs:element ref="DigestValue"/>
      </xs:choice>
    </xs:sequence>
    <xs:attribute name="URI" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="veicTransp">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="placa"/>
      <xs:element ref="UF"/>
      <xs:element ref="RNTC"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="infAdic" type="infAdFisco"/>
<xs:element name="vol">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="qVol"/>
      <xs:element ref="esp"/>
      <xs:element ref="marca"/>
      <xs:element ref="nVol"/>
      <xs:element ref="pesoL"/>
      <xs:element ref="pesoB"/>
      <xs:element ref="lacsres"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

### Diferenças:

- O XML Schema criado é bem maior que o DTD equivalente.
- Diferente do DTD, o XML Schema é escrito em XML.
- O XML Schema criado é mais simples de entender por estar em uma linguagem mais próxima da nossa do que o DTD.
- Embora seja maior e mais trabalhoso de fazer, o XML Schema é menos limitado que o DTD. No XML Schema foi possível, por exemplo, usar namespaces e validar o tipo dos elementos.

### Principais campos do XML Schema:

O documento possui um elemento, que é o mais externo (“<xs:schema>”). Nesse elemento é definido o namespace.

A tag <xs:element> é utilizada para definir um elemento, que deve ter um nome e possuir um tipo (primitivo ou complexo). Para criar um tipo complexo usamos a tag <xs:complexType> e definimos dentro dela os elementos e atributos. A cardinalidade dos elementos é definida pelos atributos xs:minOccurs (número mínimo de vezes que um subelemento pode aparecer) e xs:maxOccurs (número máximo de vezes que um subelemento pode aparecer). Os outros componentes principais são os delimitadores de grupo “<xs:sequence>” (subelementos devem aparecer na instância XML na mesma ordem em que foram declarados no esquema), “<xs:choice>” (somente um dos elementos declarados no grupo pode aparecer na instância) e “<xs:all>” (os elementos do grupo podem aparecer uma vez em qualquer ordem).

### **3ª Questão (2 pontos)**

Desenvolva um arquivo de transformação (via XSLT) para transformar os três documentos bem formados em um arquivo HTML e em TXT. Você deve escolher a formatação de saída e o layout (formatação) do HTML e TXT que desejar, como o XML de origem é longo, seu HTML de saída deverá ter até 20 campos de informação e não menos que 15 campos. A escolha dos campos do XML é livre para cada aluno, sugere-se usar um bloco de informações coerentes. Seja criativo! Não copie do seu colega.

### **4ª Questão (1.5 pontos)**

Desenvolva consultas XPath e Xquery. Use o seu editor de documentos XML, explique o procedimento utilizado e exiba os resultados na sua resposta

Consulta a) Exiba todos os nomes dos pais e avós do elemento <UF>

XPath – //UF/.. | //UF/../../

XQuery :

```
<resultado>
{
for $e in doc('documento1.xml')//UF
return ($e/.., $e/../../)
}
</resultado>
```

Consulta b) Exiba o elemento e o número do nível do elemento mais profundo na estrutura do documento numero 2

Consulta c) Exiba todos os elementos que estão no mesmo nível do elemento ICMS.

XPath – //ICMS/../../\*

XQuery:

```
<resultado>
{
for $e in doc('documento1.xml')//ICMS
return $e/../../*
}
</resultado>
```

**5ª Questão (1.5ponto) Como você observou os documentos XML estão representando um conjunto de tabelas desnormalizadas. Faça um diagrama ER que propõe uma normalização (separação de documento em tabelas). A seguir faça um XQuery que divida o documento de acordo com o projeto que você desenvolveu. Explique o processo em detalhes.**

**6ª Questão (2.0pontos) –**

**a) Quais são as principais características (PRINCÍPIOS) técnicas dos dados abertos que estão relatados nos documentos que você leu sobre o tema?**

1. Completos: Todos os dados públicos estão disponíveis e sem limitações de acesso.
2. Primários: Os dados são apresentados da forma original em que foram coletados, sem nenhuma modificação
3. Atuais: São disponibilizados assim que se tornam necessários.
4. Acessíveis: Estão disponíveis para que sejam alcançados pelo maior número de pessoas possível.
5. Compreensíveis por máquinas: São estruturados para que possam ser processados por máquina de forma automática.
6. Não discriminatórios: Estão disponíveis a todos, sem exceção.
7. Não proprietários: Ninguém controla exclusivamente os dados.
8. Livres de licenças: Não há nenhuma restrição de utilização prevista. Apenas alguns tipos de segurança podem ser usados, como por exemplo, segurança.

**b) Avalie tecnicamente TODOS os aplicativos baseados em dados abertos indicados no site <http://dados.gov.br/aplicativos/>. Faça um resumo de suas funcionalidades e indique quais os dados (datasets) estão sendo utilizados**

- Reclamações Procon:

O sistema apresenta uma interface web simples para pesquisa por nome de empresas, que apresenta basicamente uma caixa de pesquisa e filtros. Ao pesquisar, o sistema apresenta um resultado com visão simplificada com grafos de pizza e circulares, que pode ser filtrado pelo **sexo** dos consumidores (masculino e feminino) e pelo **estado de atendimento** das reclamações pelo fornecedor (atendidas e não atendidas).

O sistema também apresenta o tempo demorado pelas empresas para resolver uma determinada reclamação que são separadas em notas que vão de A à E (começando dos menos demorados).

A base de dados consultados neste projeto são **Dados Abertos** do Cadastro Nacional de Reclamações Fundamentadas de 2011 realizado pelos 25 Procons estaduais e 220 Procons municipais.

- Reclamações BR:

É uma ferramenta para o consumidor verificar o comportamento das marcas presentes no mercado brasileiro. As informações são baseadas nas reclamações dos consumidores. Na página inicial, são apresentados gráficos de barra das empresas com mais reclamações e das que tem o pior índice de solução dos atendimentos de 2009 à 2010. O portal contém uma página com uma listagem de grupos e empresas em que é possível detalhar as informações de cada empresa listada. Ao detalhar as informações de uma determinada empresa, são mostrados os gráficos de Índice de solução dos atendimentos, Problemas mais reclamados, Detalhes e evolução dos atendimentos do grupo e é exibida uma lista das empresas que



compõem o grupo em questão. Os resultados podem ser filtrados por ano e a soma de todos os anos. O total de reclamações também são apresentados.

O portal também criou um ranking das empresas com mais reclamações, com pior índice de solução dos atendimentos e também o perfil dos reclamantes. Os Dados utilizados são do Cadastro Nacional de Reclamações Fundamentadas.

- Para onde foi o meu dinheiro:

É uma ferramenta permite saber como governos gastam os recursos públicos. A página inicial mostra um conjunto de dados baseado em informações anuais sobre a execução do orçamento público. Esses dados são apresentados em forma de gráfico que relaciona o ano com a quantia em bilhões de reais usada. O tamanho do círculo apresentado no gráfico, por município, é baseado na quantidade de dados disponíveis para o banco de dados, que está diretamente ligado ao volume de dinheiro investido.

As fontes de dados utilizadas são as seguintes:

<http://www.portaltransparencia.gov.br/>

<http://www.fazenda.sp.gov.br/download/default.shtm>

[http://sempla.prefeitura.sp.gov.br/orc\\_homenew.php](http://sempla.prefeitura.sp.gov.br/orc_homenew.php)

[http://dados.tce.rs.gov.br/dados\\_siapc.html#empenhos](http://dados.tce.rs.gov.br/dados_siapc.html#empenhos)

- Aeroportos Brasil:

É um aplicativo que mostra o movimento de aeronaves e passageiros nos aeroportos administrados pela Infraero. As tabelas apresentadas pelo aplicativo mostram o movimento de aeronaves, passageiros, carga e mala postal nos 66 aeroportos administrados pela Infraero. Em aeronaves, estão computados pousos e decolagens. Em passageiros, embarques e desembarques. São apresentados três tipos de análises: análises de aeronaves, de passageiros e aeroportos. A análise de aeronaves apresenta gráficos sobre a quantidade de aeronaves por mês e por aeroportos. A análise por passageiros mostra gráficos relacionando a quantidade de passageiros por tipo de voo (doméstico e internacional), por mês, por aeroporto e por região do Brasil. A análise de aeroportos é apresentada por tipos, quantidade por mês e por regiões. Os Dados utilizados são do Movimento dos Aeroportos Administrados pela Infraero

- Siga seu vereador:

O aplicativo possui uma interface simples. Mostra uma lista de filtros para exibição de vereadores. Primeiro selecionamos o estado desejado e depois a cidade e é exibida uma lista de vereadores com o nome, o partido e a coligação de cada um. Ao selecionar um vereador, é exibida uma página contendo o nome, data de nascimento, gênero, número de urna, situação da candidatura e informações pessoais. A página apresenta também outros candidatos do mesmo partido, cidade e estado.

Os Dados utilizados são das Votações dos Vereadores da Câmara Municipal de São Paulo fornecidos pela Justiça Eleitoral.

- Radar Parlamentar:

É um aplicativo bem elaborado, com design inovador e informações detalhadas. O aplicativo ilustra as semelhanças entre partidos políticos com base na análise matemática dos dados de votações que ocorrem na casa legislativa. As semelhanças são apresentadas em um gráfico bidimensional, em que círculos representam partidos ou parlamentares, e a distância entre esses círculos representa o quão parecido os mesmos votam. São apresentados três tipos de gráficos: por gênero, que tem por objetivo fazer uma análise gráfica da participação de mulheres e homens na Câmara Federal; um radar, que apresenta o total de votações existentes e divididos entre a Câmara Municipal de São Paulo, Câmara dos Deputados e Senado; e plenárias, que podem ser filtradas entre a Câmara Municipal de São Paulo e por proposição (PL). A base de dados contém dados da câmara Municipal, da Câmara dos Deputados e do Senado.

- Basômetro:

É um blog do Estadão Dados, núcleo do Estadão especializado em reportagens baseadas em estatísticas e no desenvolvimento de projetos especiais de visualização de dados. Encontramos gráficos e animações sobre temas do noticiário do dia, além de cruzamentos de dados e análises especiais feitas por jornalistas e programadores. Os posts são divididos em três seções: gráfico do dia, permanentes, para séries estatísticas que serão constantemente atualizadas, e séries especiais, focadas em um tema específico. OS Dados utilizados foram as Informações Legislativas da Câmara dos Deputados.

- Painel do PAC:

Este aplicativo mostra, de diversas maneiras, os investimentos realizados nas Obras do PAC (Programa de Aceleração do Crescimento). O objetivo deste aplicativo é demonstrar o que é possível fazer com a utilização de ferramentas livres (Open Source) para sistemas de tomada de decisão. O aplicativo apresenta mapas, painéis e vídeos. Os dados utilizados são das Obras do Programa de Aceleração do Crescimento

**c) PROPONHA um novo tipo de aplicativo que use dados abertos que possam ser utilizados para empoderar o cidadão comum (seja criativo). Explique como ele funcionaria no nível conceitual e indique as fontes de dados.**

Aplicativo que utilize dados do IBGE para verificar os locais mais próximos onde ocorre recolhimento de Lixo eletrônico.