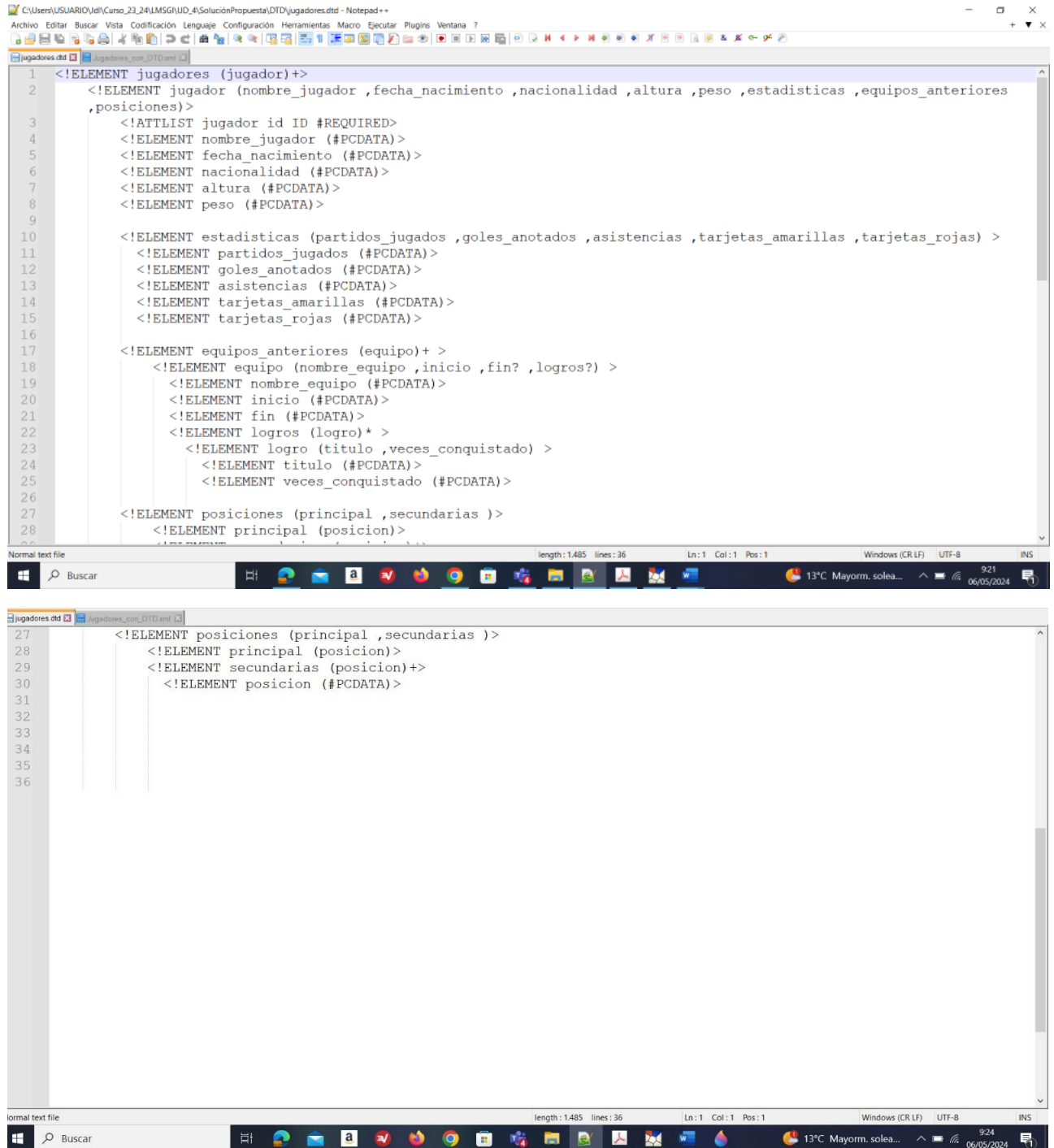


UNA SOLUCIÓN PROPUESTA PARA LA TAREA 4

A continuación, os muestro una solución posible para la tarea 4.

Para la actividad 1 se propone el siguiente documento con un vocabulario DTD.
Documento “jugadores.dtd”:



```
1 <!ELEMENT jugadores (jugador)+>
2 <!-- jugador (nombre_jugador , fecha_nacimiento , nacionalidad , altura , peso , estadisticas , equipos_anteriores , posiciones) -->
3 <!-- jugador id ID #REQUIRED -->
4 <!-- jugador nombre_jugador (#PCDATA) -->
5 <!-- jugador fecha_nacimiento (#PCDATA) -->
6 <!-- jugador nacionalidad (#PCDATA) -->
7 <!-- jugador altura (#PCDATA) -->
8 <!-- jugador peso (#PCDATA) -->
9
10 <!-- jugador estadisticas (partidos_jugados , goles_anoatados , asistencias , tarjetas_amarillas , tarjetas_rojas) -->
11 <!-- jugador estadisticas partidos_jugados (#PCDATA) -->
12 <!-- jugador estadisticas goles_anoatados (#PCDATA) -->
13 <!-- jugador estadisticas asistencias (#PCDATA) -->
14 <!-- jugador estadisticas tarjetas_amarillas (#PCDATA) -->
15 <!-- jugador estadisticas tarjetas_rojas (#PCDATA) -->
16
17 <!-- jugador equipos_anteriores (equipo)+ -->
18 <!-- jugador equipos_anteriores equipo (nombre_equipo , inicio , fin? , logros?) -->
19 <!-- jugador equipos_anteriores equipo nombre_equipo (#PCDATA) -->
20 <!-- jugador equipos_anteriores equipo inicio (#PCDATA) -->
21 <!-- jugador equipos_anteriores equipo fin (#PCDATA) -->
22 <!-- jugador equipos_anteriores equipo logros (logro)* -->
23 <!-- jugador equipos_anteriores equipo logro (titulo , veces_conquistado) -->
24 <!-- jugador equipos_anteriores equipo logro titulo (#PCDATA) -->
25 <!-- jugador equipos_anteriores equipo logro veces_conquistado (#PCDATA) -->
26
27 <!-- jugador posiciones (principal , secundarias) -->
28 <!-- jugador posiciones principal (posicion) -->
29 <!-- jugador posiciones secundarias (posicion)+ -->
30 <!-- jugador posiciones secundarias posicion (#PCDATA) -->
31
32
33
34
35
36
```

Para la actividad 2 habría que modificar el documento XML, jugadores.xml, para asociarle el vocabulario DTD propuesto. Se propone esta cabecera sin alterar el resto del documento.
Documento “jugadores_con_DTD.xml”

```
1 <?xml version='1.0' encoding='UTF-8' standalone='no' ?>
2 <!DOCTYPE jugadores SYSTEM "jugadores.dtd">
3 <jugadores>
4   <jugador id="J0001/01">
5     <nombre_jugador>Lionel Messi</nombre_jugador>
6     <fecha_nacimiento>1987-06-24</fecha_nacimiento>
7     <nacionalidad>Argentina</nacionalidad>
8     <altura>170</altura>
9     <peso>72</peso>
10
11     <estadisticas>
12       <partidos_jugados>800</partidos_jugados>
13       <goles_anoatados>700</goles_anoatados>
14       <asistencias>300</asistencias>
15       <tarjetas_amarillas>30</tarjetas_amarillas>
16       <tarjetas_rojas>1</tarjetas_rojas>
17     </estadisticas>
18
19     <equipos_anteriores>
20       <equipo>
21         <nombre_equipo>FC.Barcelona</nombre_equipo>
22         <inicio>2004</inicio>
23         <fin>2021</fin>
24         <logros>
25           <logro>
26             <titulo>Liga Española</titulo>
27             <veces_conquistado>10</veces_conquistado>
28           </logro>
29           <logro>
```

Para la actividad 3 se propone el siguiente schema. Documento “jugadores.xsd”:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
3
4   <xs:element name="jugadores">
5     <xs:complexType>
6       <xs:sequence>
7         <xs:element ref="jugador" minOccurs="1" maxOccurs="unbounded"/>
8       </xs:sequence>
9     </xs:complexType>
10  </xs:element>
11
12  <!-- Se describe el elemento "jugador" -->
13
14  <xs:element name="jugador">
15    <xs:complexType>
16      <xs:sequence>
17        <xs:element name="nombre_jugador" type="xs:string"/>
18        <xs:element name="fecha_nacimiento" type="xs:date"/>
19        <xs:element name="nacionalidad" type="xs:string"/>
20        <xs:element name="altura" type="TipoAltura"/>
21        <xs:element name="peso" type="TipoPeso"/>
22        <xs:element ref="estadisticas"/>
23        <xs:element ref="equipos_anteriores"/>
24        <xs:element ref="posiciones"/>
25      </xs:sequence>
26      <xs:attribute name="id" type="TipoID" use="required"/>
27    </xs:complexType>
28  </xs:element>
29
```

```
Jugadores.xsd [3]  Jugadores_con_XSD.xml [3]
28 </xs:element>
29
30 <!-- Se describe el elemento "estadisticas" -->
31
32 <xs:element name="estadisticas">
33   <xs:complexType>
34     <xs:sequence>
35       <xs:element name="partidos_jugados" type="TipoEstadistica"/>
36       <xs:element name="goles_anotados" type="TipoEstadistica"/>
37       <xs:element name="asistencias" type="TipoEstadistica"/>
38       <xs:element name="tarjetas_amarillas" type="TipoEstadistica"/>
39       <xs:element name="tarjetas_rojas" type="TipoEstadistica"/>
40     </xs:sequence>
41   </xs:complexType>
42 </xs:element>
43
44 <!-- Se describe el elemento "equipos anteriores" -->
45
46 <xs:element name="equipos_anteriores">
47   <xs:complexType>
48     <xs:sequence>
49       <xs:element ref="equipo" maxOccurs="unbounded"/>
50     </xs:sequence>
51   </xs:complexType>
52 </xs:element>
53 <xs:element name="equipo"> <!-- Se describe el elemento "equipo" -->
54   <xs:complexType>
55     <xs:sequence>
56       <xs:element name="nombre_equipo" type="xs:string"/>
57       <xs:element name="inicio" type="TipoYear"/>
58       <xs:element name="fin" type="xs:string"/>
59       <!-- Comentario Juan Diego: Efectivamente el tipo "xs:nonNegativeInteger" no es válido la ausencia de un
60        valor. Para solventar el tema, una de dos, la solución que has utilizado o tal y como especifica el rango de
61        valores permitido sería dar un valor de año de "fin" menor que al año de "inicio" en el XML lo que se podría
62        interpretar como que no ha concluido el contrato. Solventado este punto la validación del esquema es
63        totalmente correcta -->
64       <xs:element ref="logros"/>
65     </xs:sequence>
66   </xs:complexType>
67 </xs:element>
68 <xs:element name="logros"> <!-- Se describe el elemento "logros" -->
69   <xs:complexType>
70     <xs:sequence>
71       <xs:element ref="logro" minOccurs="0" maxOccurs="unbounded"/>
72     </xs:sequence>
73   </xs:complexType>
74 </xs:element>
75 <xs:element name="logro"> <!-- Se describe el elemento "logros" -->
76   <xs:complexType>
77     <xs:sequence>
78       <xs:element name="titulo" type="xs:string"/>
79       <xs:element name="veces_conquistado" type="TipoEstadistica"/>
80     </xs:sequence>
81   </xs:complexType>
82 </xs:element>
83 </xs:schema>
```

```
Jugadores.xsd [3]  Jugadores_con_XSD.xml [3]
55 <xs:sequence>
56   <xs:element name="nombre_equipo" type="xs:string"/>
57   <xs:element name="inicio" type="TipoYear"/>
58   <xs:element name="fin" type="xs:string"/>
59   <!-- Comentario Juan Diego: Efectivamente el tipo "xs:nonNegativeInteger" no es válido la ausencia de un
60    valor. Para solventar el tema, una de dos, la solución que has utilizado o tal y como especifica el rango de
61    valores permitido sería dar un valor de año de "fin" menor que al año de "inicio" en el XML lo que se podría
62    interpretar como que no ha concluido el contrato. Solventado este punto la validación del esquema es
63    totalmente correcta -->
64   <xs:element ref="logros"/>
65 </xs:sequence>
66 </xs:complexType>
67 </xs:element>
68 <xs:element name="logros"> <!-- Se describe el elemento "logros" -->
69   <xs:complexType>
70     <xs:sequence>
71       <xs:element ref="logro" minOccurs="0" maxOccurs="unbounded"/>
72     </xs:sequence>
73   </xs:complexType>
74 </xs:element>
75 <xs:element name="logro"> <!-- Se describe el elemento "logros" -->
76   <xs:complexType>
77     <xs:sequence>
78       <xs:element name="titulo" type="xs:string"/>
79       <xs:element name="veces_conquistado" type="TipoEstadistica"/>
80     </xs:sequence>
81   </xs:complexType>
82 </xs:element>
83 </xs:schema>
```

```
Jugadores.xsd  jugadores_con_XSD.xml
78 </xs:element>
79
80 <!-- Se describe el elemento "posiciones" -->
81
82 <xs:element name="posiciones">
83 <xs:complexType>
84 <xs:sequence>
85 <xs:element ref="principal"/>
86 <xs:element ref="secundarias"/>
87 </xs:sequence>
88 </xs:complexType>
89 </xs:element>
90 <xs:element name="principal"> <!-- Se describe el elemento "principal" -->
91 <xs:complexType>
92 <xs:sequence>
93 <xs:element name="posicion" type="TipoPosicion"/>
94 </xs:sequence>
95 </xs:complexType>
96 </xs:element>
97 <xs:element name="secundarias"> <!-- Se describe el elemento "secundarias" -->
98 <xs:complexType>
99 <xs:sequence>
100 <xs:element name="posicion" type="TipoPosicion" minOccurs="1" maxOccurs="unbounded"/>
101 </xs:sequence>
102 </xs:complexType>
103 </xs:element>
104
105 <!-- Definición de tipos de datos -->
106
/xs:schema/xs:element[principal]/xs:complexType/xs:sequence length: 6.677 lines: 154 Ln: 93 Col: 58 Pos: 3.646 Windows (CR LF) UTF-8-BOM INS
9:39 13°C Mayorm. solea... 06/05/2024
```

```
Jugadores.xsd  jugadores_con_XSD.xml
105 <!-- Definición de tipos de datos -->
106
107 <xs:simpleType name="TipoID"> <!-- Definición de TipoID -->
108 <xs:restriction base="xs:string">
109 <!-- Debería ser base="xs:ID" ya que se trata del atributo identificador, pero hay un problema que impide
utilizar el "pattern" que se pide si se trata de un dato "xs:ID" : ese tipo de dato no admite el caracter "/" como
válido. Una solución pasa por cambiar en el XML el caracter por otro, por ejemplo, "-" . Otra alternativa pasa por
definir el tipo base como "xs:string" -->
110 <xs:pattern value="J[0-9]{4}/[0-9]{2}"/> <!-- Patrón permitido-->
111 </xs:restriction>
112 </xs:simpleType>
113
114 <xs:simpleType name="TipoAltura"> <!-- Definición de TipoAltura-->
115 <xs:restriction base="xs:nonNegativeInteger">
116 <xs:minInclusive value="150"/> <!-- Valores permitidos-->
117 <xs:maxInclusive value="210"/>
118 </xs:restriction>
119 </xs:simpleType>
120
121 <xs:simpleType name="TipoPeso"> <!-- Definición de TipoPeso-->
122 <xs:restriction base="xs:nonNegativeInteger">
123 <xs:minInclusive value="60"/> <!-- Valores permitidos-->
124 <xs:maxInclusive value="90"/>
125 </xs:restriction>
126 </xs:simpleType>
127
128 <xs:simpleType name="TipoEstadistica"> <!-- Definición de TipoEstadistica-->
129 <xs:restriction base="xs:positiveInteger"/>
130 </xs:simpleType>
131
/xs:schema/xs:element[principal]/xs:complexType/xs:sequence length: 6.677 lines: 154 Ln: 93 Col: 58 Pos: 3.646 Windows (CR LF) UTF-8-BOM INS
9:40 13°C Mayorm. solea... 06/05/2024
```



```
129 <xs:restriction base="xs:positiveInteger"/>
130 </xs:simpleType>
131
132 <xs:simpleType name="TipoYear"> <!-- Definición de TipoYear-->
133 <xs:restriction base="xs:integer">
134 <xs:minInclusive value="2000"/> <!-- Valores permitidos-->
135 <xs:maxInclusive value="2100"/>
136 </xs:restriction>
137 </xs:simpleType>
138 <!-- El tipo de datos numérico como, xs:integer, no permite el no valor para un elemento como es el caso de "fin".
139 Para solventar esta situación podemos optar por varias alternativas. La que propongo de manera más sencilla pasa por
140 introducir una modificación en el XML y dar un valor al elemento "fin" en aquellos casos en los que elemento aparece
141 vacío, inferior al valor que tenga el elemento "inicio" y así deducir que no ha concluido la estancia del jugador en
142 el equipo -->
143
144 <xs:simpleType name="TipoPosicion"> <!-- Definición de TipoPosicion-->
145 <xs:restriction base="xs:string">
146 <!-- Valores permitidos entre una lista cerrada de valores -->
147 <xs:enumeration value="portero"/>
148 <xs:enumeration value="defensa"/>
149 <xs:enumeration value="mediocentro"/>
150 <xs:enumeration value="delantero"/>
151 <xs:enumeration value="extremo"/>
152 <xs:enumeration value="mediapunta"/>
153 <xs:enumeration value="segundo delantero"/>
154 <xs:enumeration value="interior"/>
155 </xs:restriction>
156 </xs:simpleType>
157 </xs:schema>
```

Para la actividad 4, hay que enlazar el documento con schema propuesto modificando la cabecera del XML original. Documento “jugadores_con_XSD.xml”. Incluyo copia de la cabecera y el propio documento xml modificado:

```
1 <?xml version='1.0' encoding='UTF-8' standalone='no' ?>
2 <jugadores xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3   xsi:noNamespaceSchemaLocation="Jugadores.xsd">
4   <jugador id="J0001/01">
5     <nombre_jugador>Lionel Messi</nombre_jugador>
6     <fecha_nacimiento>1987-06-24</fecha_nacimiento>
7     <nacionalidad>Argentina</nacionalidad>
8     <altura>170</altura>
9     <peso>72</peso>
10
11     <estadisticas>
12       <partidos_jugados>800</partidos_jugados>
13       <goles_anotados>700</goles_anotados>
14       <asistencias>300</asistencias>
15       <tarjetas_amarillas>30</tarjetas_amarillas>
16       <tarjetas_rojas>1</tarjetas_rojas>
17     </estadisticas>
18
19     <equipos_anteriores>
20       <equipo>
21         <nombre_equipo>FC.Barcelona</nombre_equipo>
22         <inicio>2004</inicio>
23         <fin>2021</fin>
24         <logros>
25           <logro>
26             <titulo>Liga Española</titulo>
27             <veces_conquistado>10</veces_conquistado>
28           </logro>
29           <logro>
```

Para validar el anterior schema XSD y comprobar que el documento XML propuesto está de acuerdo con el vocabulario XSD creado, utilizo el validador que se incluye como herramienta con el Notepad++, dentro de las “plugins XML tools” de la herramienta:

