Cambios realizados en el XML Esquema.

Cabecera

Hubo que añadir una cabecera para declarar el documento xsd.

```xml
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    targetNamespace="http://www.uniovi.es"
    xmlns="http://www.uniovi.es"
    elementFormDefault="qualified">
    <xs:element name="circuitoEsquema">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="circuit">
                    <xs:complexType>
                        <xs:sequence>
                            <xs:element ref="date" />
                            <xs:element ref="location" />
                            <xs:element ref="length" />
                            <xs:element ref="width" />
                            <xs:element ref="references" />
                            <xs:element ref="fotos" />
                            <xs:element ref="videos" />
                            <xs:element ref="coordinates" />
                            <xs:element ref="sections" />
                        </xs:sequence>
                        <xs:attribute name="name" type="xs:string" use="required" />
                        <xs:attribute name="numberLaps" type="positiveInt-type" use="required"
                    </xs:complexType>
                </xs:element>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
```

Cambios de datos

Circuit:

- Length: xs:string -> positiveDecimal-type
- Width: xs:string -> positiveDecimal-type
- NumberLaps: xs:string -<positiveInt-type

```xml
<xs:attribute name="name" type="xs:string" use="required" />
<xs:attribute name="length" type="xs:string" use="required" />
<xs:attribute name="width" type="xs:string" use="required" />
<xs:attribute name="numberLaps" type="xs:string" use="required" />
```

```xml
<xs:attribute name="name" type="xs:string" use="required" />
<xs:attribute name="length" type="positiveDecimal-type" use="required" />
<xs:attribute name="width" type="positiveDecimal-type" use="required" />
<xs:attribute name="numberLaps" type="positiveInt-type" use="required" />
```

Length:

- Base:xs:string -> positiveDecimal-type

```
<xs:element name="Length">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
```

```
<xs:element name="length">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="positiveDecimal-type">
```

Width:

- Base: xs:string -> positiveDecimal-type

```
<xs:element name="width">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
```

```
<xs:element name="width">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="positiveDecimal-type">
```

Date:

- Base: xs:string -> xs:date
- Time: xs:string -> xs:time

```
<xs:element name="date">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="time" type="xs:string" use="required" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

```
<xs:element name="date">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:date">
                <xs:attribute name="time" type="xs:time" use="required" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

Section:

- Distance: xs:string -> positiveDecimal-type

Coordinates and section:

- Latitude: xs:string -> latitude-type
- Longitude: xs:string -> longitude-type
- Height: xs:string -> xs:decimal

```xml
<xs:element name="coordinates">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="latitude" type="xs:string" use="required" />
                <xs:attribute name="longitude" type="xs:string" use="required" />
                <xs:attribute name="height" type="xs:string" use="required" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

```xml
<xs:element name="coordinates">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="latitude" type="latitude-type" use="required" />
                <xs:attribute name="longitude" type="longitude-type" use="required" />
                <xs:attribute name="height" type="xs:decimal" use="required" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

```xml
<xs:element name="section">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="distance" type="xs:string" use="required" />
                <xs:attribute name="latitude" type="xs:string" use="required" />
                <xs:attribute name="longitude" type="xs:string" use="required" />
                <xs:attribute name="height" type="xs:string" use="required" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

```xml
<xs:element name="section">
    <xs:complexType>
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="distance" type="positiveDecimal-type" use="required" />
                <xs:attribute name="latitude" type="latitude-type" use="required" />
                <xs:attribute name="longitude" type="longitude-type" use="required" />
                <xs:attribute name="height" type="xs:decimal" use="required" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>
</xs:element>
```

Cambios de rangos

References: [0-*] -> [3-*]

```
<xs:element name="references">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" ref="reference" />
```

```
<xs:element name="references">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="3" maxOccurs="unbounded" ref="reference" />
```

Fotos: [1-*] -> [1-5]

```
<xs:element name="fotos">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="unbounded" ref="foto" />
```

```
<xs:element name="fotos">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="5" ref="foto" />
```

Videos: [0-*] -> [0-3]

```
<xs:element name="videos">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" ref="video" />
```

```
<xs:element name="videos">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="3" ref="video" />
```

Cambios de restricciones

Length, width y distance usan positiveDecimal-type que obliga a que los valores sean positivos.

```
<xs:simpleType name="positiveDecimal-type">
    <xs:restriction base="xs:decimal">
        <xs:minInclusive value="0"></xs:minInclusive>
    </xs:restriction>
</xs:simpleType>
```

NumberLaps usa positiveInt-type que obliga a que el valor sea positivo.

```
<xs:simpleType name="positiveInt-type">
    <xs:restriction base="xs:integer">
        <xs:minInclusive value="0"></xs:minInclusive>
    </xs:restriction>
</xs:simpleType>
```

Latitude y longitude usan sus propios tipos que restringe los valores a los mínimos y máximos de cada uno.

```
<xs:simpleType name="latitude-type">
    <xs:restriction base="xs:decimal">
        <xs:minInclusive value="-90"></xs:minInclusive>
        <xs:maxInclusive value ="90"></xs:maxInclusive>
    </xs:restriction>
</xs:simpleType>
```

```
<xs:simpleType name="longitude-type">
    <xs:restriction base="xs:decimal">
        <xs:minInclusive value="-180"></xs:minInclusive>
        <xs:maxInclusive value ="180"></xs:maxInclusive>
    </xs:restriction>
</xs:simpleType>
```