University of Dublin

Trinity College

Rohan Taneja
19323238
B.A.I. Computer Engineering

CS4D2b – Knowledge Engineering

Final Assignment Submission

School of Computer Science and Statistics
O'Reilly Institute, Trinity College, Dublin 2, Ireland

Declaration

I understand that this is an individual assessment and that collaboration is not permitted. I have not received any assistance with my work for this assessment. Where I have used the published work of others, I have indicated this with appropriate citation.

I have not and will not share any part of my work on this assessment, directly or indirectly, with any other student. I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year, found at http://www.tcd.ie/calendar. I have also completed the Online Tutorial on avoiding plagiarism 'Ready Steady Write', located at http://tcdie.libguides.com/plagiarism/ready-steady-write."

I understand that by returning this declaration with my work, I am agreeing with the above statement. ✅

Name: Rohan Taneja
Date: 23rd May 2021

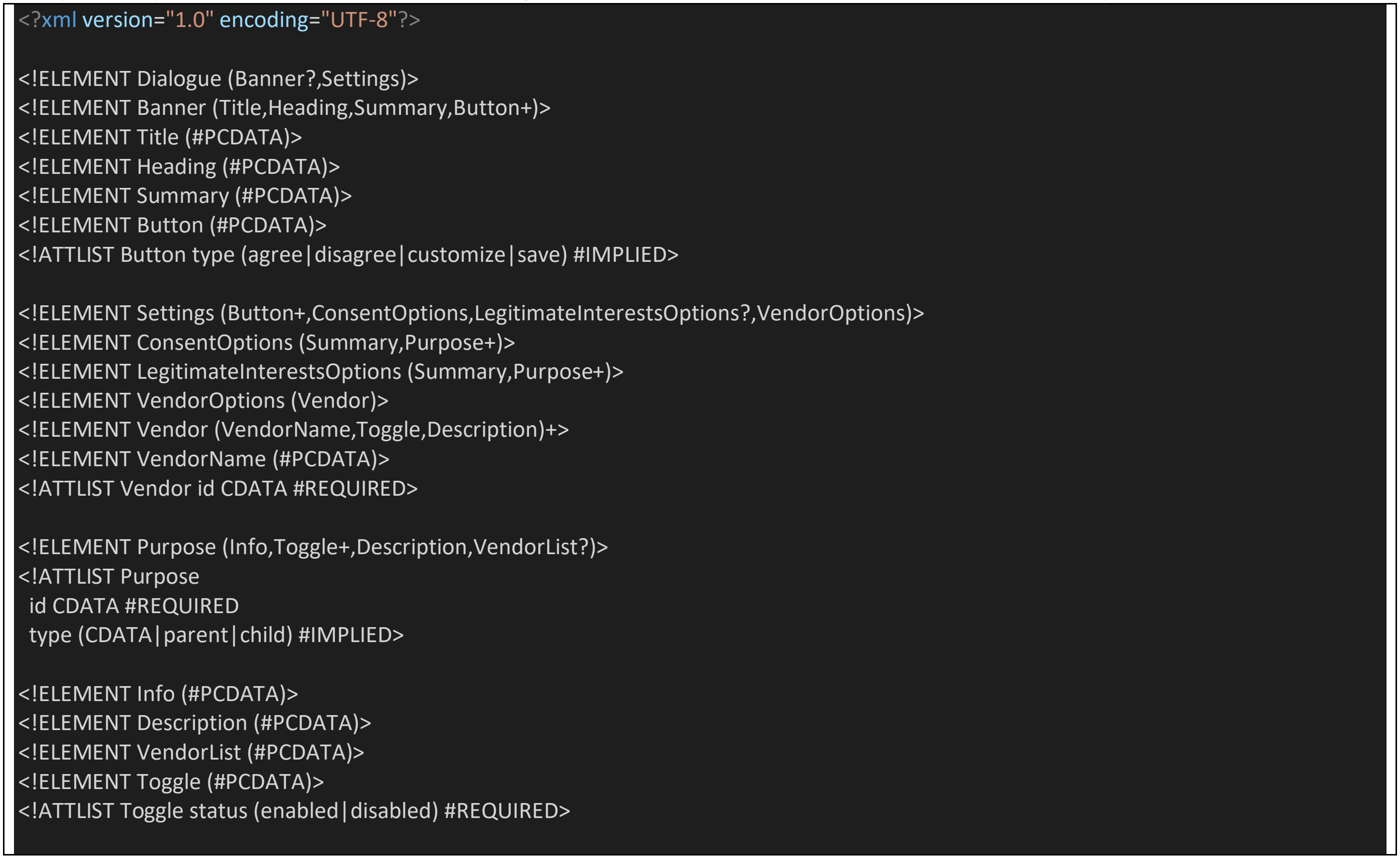
Table of Content

Data Privacy Vocabulary – Consent Right

An XML vocabulary for generalising consent dialogues generated on websites based on Transparency & Consent Framework specification for modern Consent Management Platforms.

a) Document Type Definition

The DTD represents a new XML vocabulary for a Consent Dialogue generated when we access a website. It contains generated banner and customizable settings. The banner is structured to nest a title, heading and summary followed by buttons to trigger certain action. The consent customizing settings provide options with provisional User Specified Consent Options, Publisher Interested Legitimate Options and Name of vendors involved in collection of user data. Consent Options are compulsory and have numerous purpose(+) which are supposedly one-or-more directing with a type parent or child purpose which can be selectively handled. Similarly, for Legitimate Interest Options but can be neglected which has Purpose definition exclusive of type. The purposes present description, toggle and a list of vendors. The vendor list is separately available to allocate sharing of data with certain vendor as authorised by the user.

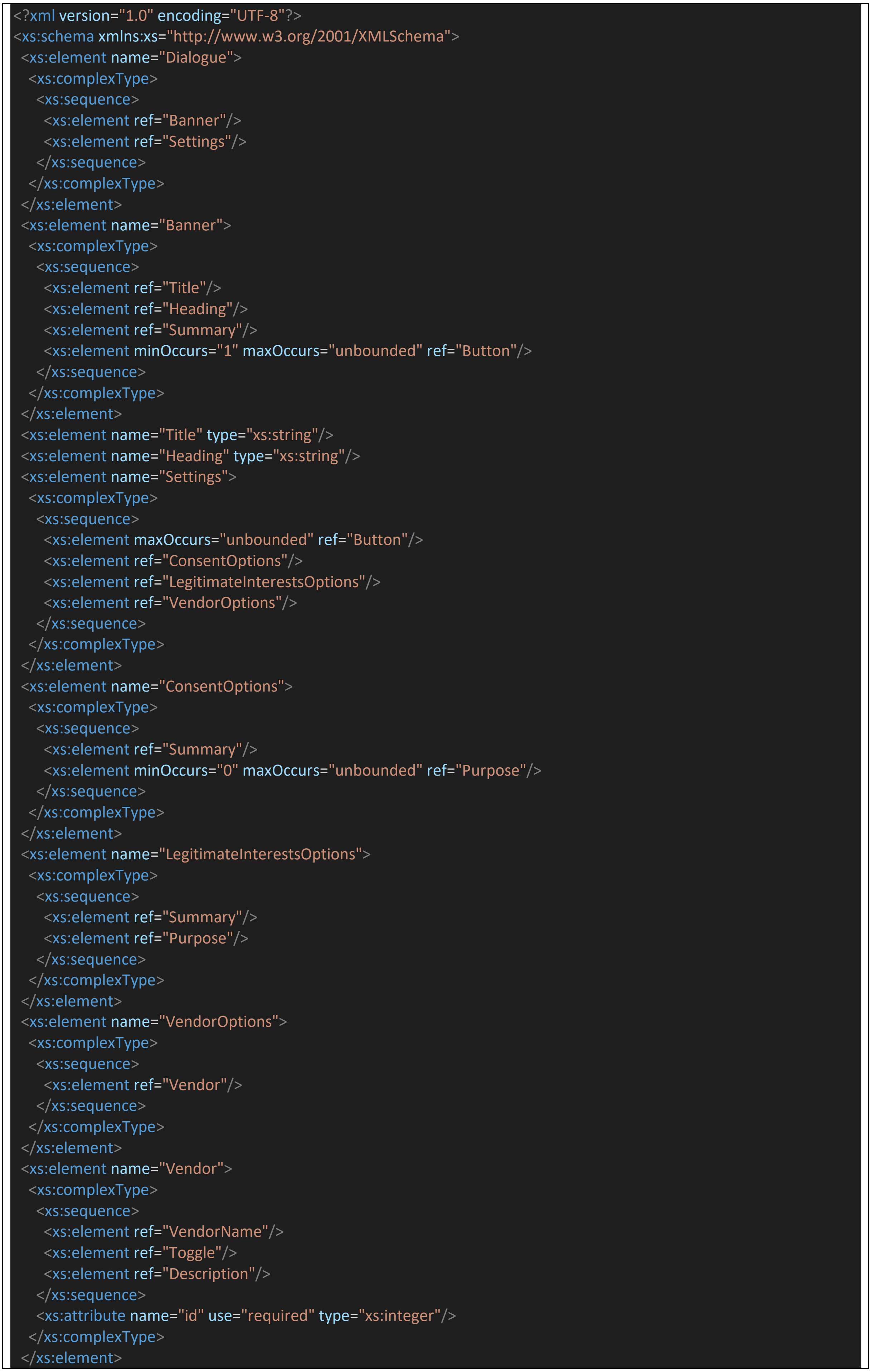


```
<?xml version="1.0" encoding="UTF-8"?>

<!ELEMENT Dialogue (Banner?,Settings)>
<!ELEMENT Banner (Title,Heading,Summary,Button+)>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Heading (#PCDATA)>
<!ELEMENT Summary (#PCDATA)>
<!ELEMENT Button (#PCDATA)>
<!ATTLIST Button type (agree|disagree|customize|save) #IMPLIED>

<!ELEMENT Settings (Button+,ConsentOptions,LegitimateInterestsOptions?,VendorOptions)>
<!ELEMENT ConsentOptions (Summary,Purpose+)>
<!ELEMENT LegitimateInterestsOptions (Summary,Purpose+)>
<!ELEMENT VendorOptions (Vendor)>
<!ELEMENT Vendor (VendorName,Toggle,Description)+>
<!ELEMENT VendorName (#PCDATA)>
<!ATTLIST Vendor id CDATA #REQUIRED>

<!ELEMENT Purpose (Info,Toggle+,Description,VendorList?)>
<!ATTLIST Purpose
 id CDATA #REQUIRED
 type (CDATA|parent|child) #IMPLIED>

<!ELEMENT Info (#PCDATA)>
<!ELEMENT Description (#PCDATA)>
<!ELEMENT VendorList (#PCDATA)>
<!ELEMENT Toggle (#PCDATA)>
<!ATTLIST Toggle status (enabled|disabled) #REQUIRED>
```

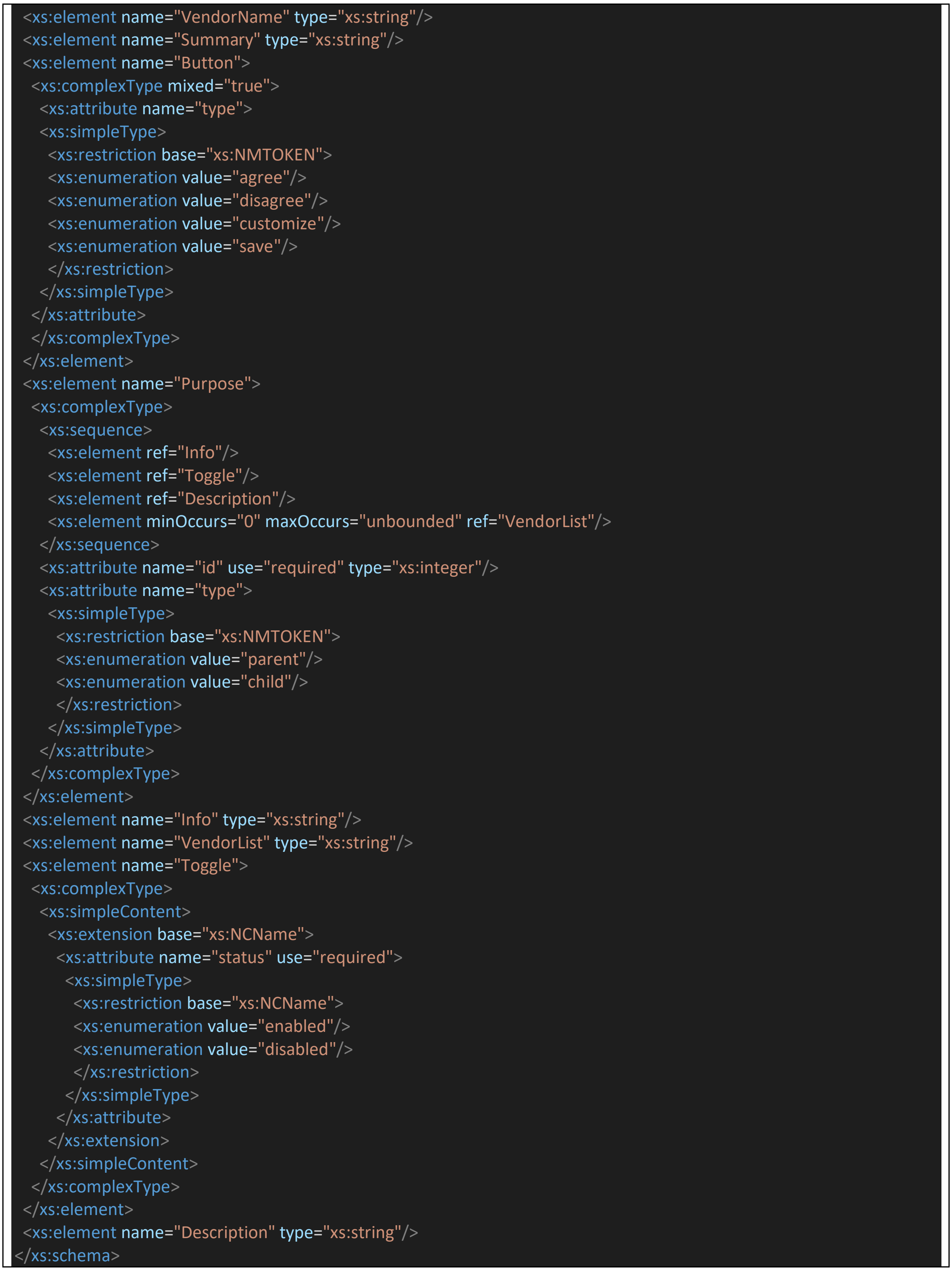
b) XSD Conversion from DTD

To convert the designed DTD to an XSD formal definition of elements and attributes is required in context with its XML. Also, it helps in figuring out a relationship between data and elements defined for validating the structured XML and vocabulary. Various cardinalities defined in DTD are present using minOccurs & maxOccurs corresponding to occurrences of the attributed elements respectively.

For e.g. type definition of button, type of Purpose – PARENT or CHILD and id of Purpose for unique identification. This has helped in separating the type of information accepted by the element based on unique identifiers and further enhance the validation by restricting using appropriate filtering.



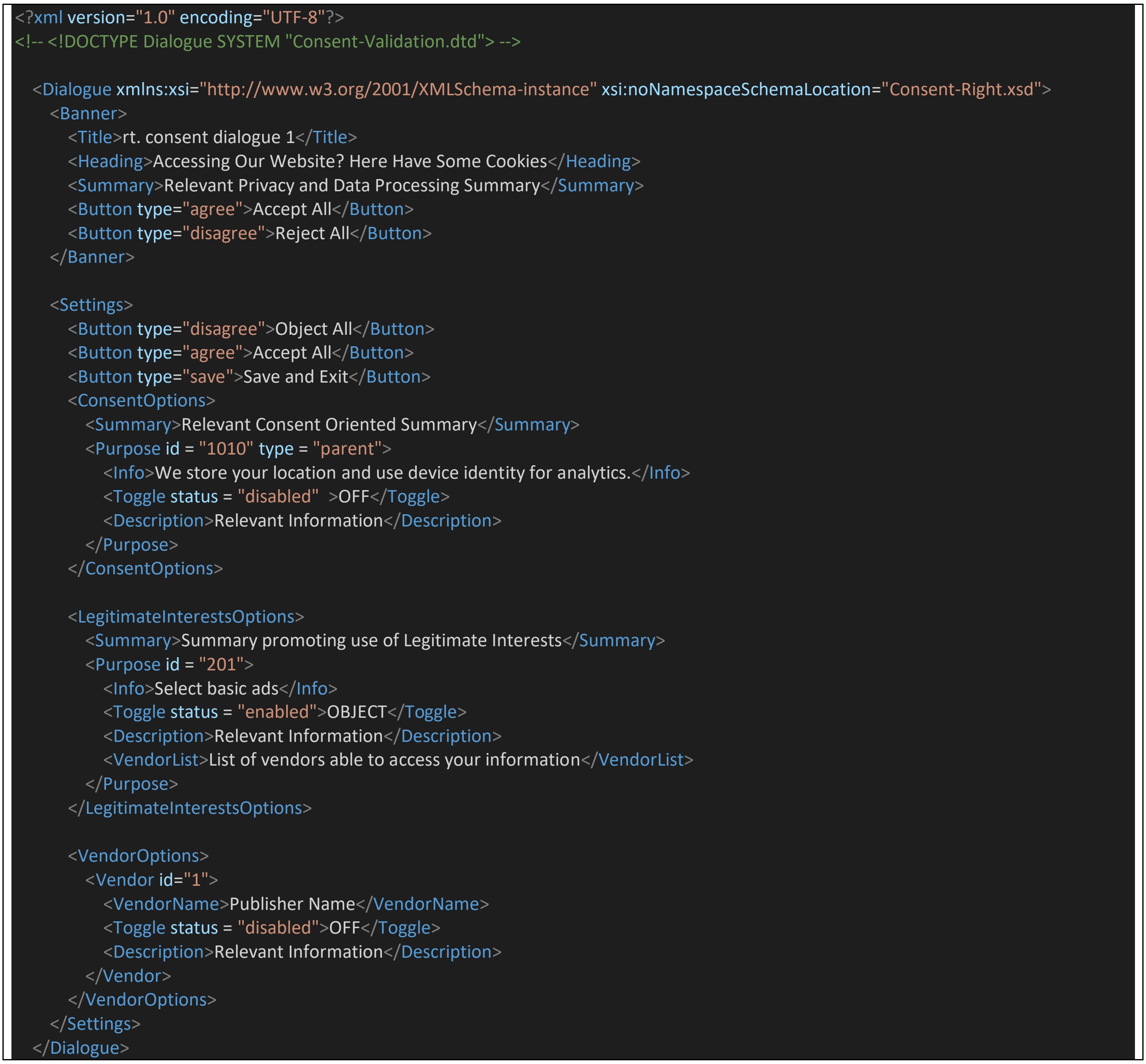
```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
 <xs:element name="Dialogue">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Banner"/>
    <xs:element ref="Settings"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="Banner">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Title"/>
    <xs:element ref="Heading"/>
    <xs:element ref="Summary"/>
    <xs:element minOccurs="1" maxOccurs="unbounded" ref="Button"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="Title" type="xs:string"/>
 <xs:element name="Heading" type="xs:string"/>
 <xs:element name="Settings">
  <xs:complexType>
   <xs:sequence>
    <xs:element maxOccurs="unbounded" ref="Button"/>
    <xs:element ref="ConsentOptions"/>
    <xs:element ref="LegitimateInterestsOptions"/>
    <xs:element ref="VendorOptions"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="ConsentOptions">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Summary"/>
    <xs:element minOccurs="0" maxOccurs="unbounded" ref="Purpose"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="LegitimateInterestsOptions">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Summary"/>
    <xs:element ref="Purpose"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="VendorOptions">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Vendor"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
 <xs:element name="Vendor">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="VendorName"/>
    <xs:element ref="Toggle"/>
    <xs:element ref="Description"/>
   </xs:sequence>
   <xs:attribute name="id" use="required" type="xs:integer"/>
  </xs:complexType>
 </xs:element>
```



```
 <xs:element name="VendorName" type="xs:string"/>
 <xs:element name="Summary" type="xs:string"/>
 <xs:element name="Button">
  <xs:complexType mixed="true">
   <xs:attribute name="type">
   <xs:simpleType>
    <xs:restriction base="xs:NMTOKEN">
    <xs:enumeration value="agree"/>
    <xs:enumeration value="disagree"/>
    <xs:enumeration value="customize"/>
    <xs:enumeration value="save"/>
    </xs:restriction>
   </xs:simpleType>
  </xs:attribute>
  </xs:complexType>
 </xs:element>
 <xs:element name="Purpose">
  <xs:complexType>
   <xs:sequence>
    <xs:element ref="Info"/>
    <xs:element ref="Toggle"/>
    <xs:element ref="Description"/>
    <xs:element minOccurs="0" maxOccurs="unbounded" ref="VendorList"/>
   </xs:sequence>
   <xs:attribute name="id" use="required" type="xs:integer"/>
   <xs:attribute name="type">
    <xs:simpleType>
     <xs:restriction base="xs:NMTOKEN">
     <xs:enumeration value="parent"/>
     <xs:enumeration value="child"/>
     </xs:restriction>
    </xs:simpleType>
   </xs:attribute>
  </xs:complexType>
 </xs:element>
 <xs:element name="Info" type="xs:string"/>
 <xs:element name="VendorList" type="xs:string"/>
 <xs:element name="Toggle">
  <xs:complexType>
   <xs:simpleContent>
    <xs:extension base="xs:NCName">
     <xs:attribute name="status" use="required">
      <xs:simpleType>
       <xs:restriction base="xs:NCName">
       <xs:enumeration value="enabled"/>
       <xs:enumeration value="disabled"/>
       </xs:restriction>
      </xs:simpleType>
     </xs:attribute>
    </xs:extension>
   </xs:simpleContent>
  </xs:complexType>
 </xs:element>
 <xs:element name="Description" type="xs:string"/>
</xs:schema>
```

c) XML Document Validation & Testing Invalid Input

1. Two Sample XML Documents that valid against the XSD are as follows –



```
<?xml version="1.0" encoding="UTF-8"?>
<!-- <!DOCTYPE Dialogue SYSTEM "Consent-Validation.dtd"> -->

  <Dialogue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="Consent-Right.xsd">
    <Banner>
      <Title>rt. consent dialogue 1</Title>
      <Heading>Accessing Our Website? Here Have Some Cookies</Heading>
      <Summary>Relevant Privacy and Data Processing Summary</Summary>
      <Button type="agree">Accept All</Button>
      <Button type="disagree">Reject All</Button>
    </Banner>

    <Settings>
      <Button type="disagree">Object All</Button>
      <Button type="agree">Accept All</Button>
      <Button type="save">Save and Exit</Button>
      <ConsentOptions>
        <Summary>Relevant Consent Oriented Summary</Summary>
        <Purpose id = "1010" type = "parent">
          <Info>We store your location and use device identity for analytics.</Info>
          <Toggle status = "disabled"  >OFF</Toggle>
          <Description>Relevant Information</Description>
        </Purpose>
      </ConsentOptions>

      <LegitimateInterestsOptions>
        <Summary>Summary promoting use of Legitimate Interests</Summary>
        <Purpose id = "201">
          <Info>Select basic ads</Info>
          <Toggle status = "enabled">OBJECT</Toggle>
          <Description>Relevant Information</Description>
          <VendorList>List of vendors able to access your information</VendorList>
        </Purpose>
      </LegitimateInterestsOptions>

      <VendorOptions>
        <Vendor id="1">
          <VendorName>Publisher Name</VendorName>
          <Toggle status = "disabled">OFF</Toggle>
          <Description>Relevant Information</Description>
        </Vendor>
      </VendorOptions>
    </Settings>
  </Dialogue>
```

First sample XML document written is an example of an ideal consent dialogue box. With specified structure all cardinalities are met with correctly nested elements validated against the formed XSD.

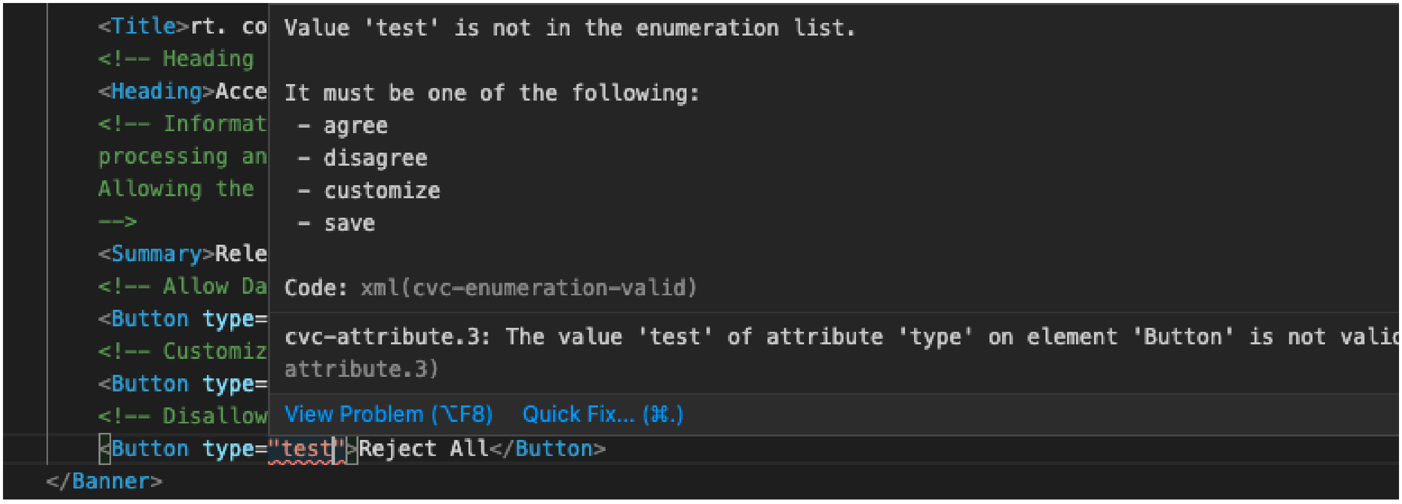
Now we write another sample XML document removing the legitimate interest options which is not mandatory and including child purposes in the consent options as show below keeping the overall structure valid.

```xml
  <Dialogue xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="Consent-Right.xsd">
    <Banner>
      <Title>rt. consent dialogue 2</Title>
      <Heading>Accessing Our Website? Here Have Some Cookies</Heading>
      <Summary>Some Data Processing Summary</Summary>
      <Button type="agree">Accept All</Button>
      <Button type="disagree">Reject All</Button>
    </Banner>

    <Settings>
      <Button type="save">Save and Exit</Button>
      <ConsentOptions>
        <Summary>Relevant Consent Oriented Summary</Summary>
        <Purpose id = "1010" type = "parent">
          <Info>We store your location and use device identity for analytics.</Info>
          <Toggle status = "disabled"  >OFF</Toggle>
          <Description>Relevant Information</Description>
        </Purpose>
        <Purpose id = "1011"  type = "child">
          <Info>Use precise geolocation data</Info>
          <Toggle status = "enabled" >ON</Toggle>
          <Description>Relevant Information</Description>
          <VendorList>List of vendors able to access your information</VendorList>
        </Purpose>
        <Purpose id = "1012" type = "child" >
          <Info>Actively scan device characteristics for identification</Info>
          <Toggle status = "disabled">OFF</Toggle>
          <Description>Relevant Information</Description>
          <VendorList>List of vendors able to access your information</VendorList>
        </Purpose>
      </ConsentOptions>

      <VendorOptions>
        <Vendor id="1">
          <VendorName>Publisher Name</VendorName>
          <Toggle status = "disabled">OFF</Toggle>
          <Description>Relevant Information</Description>
        </Vendor>
      </VendorOptions>
    </Settings>
  </Dialogue>
```

Err… cases

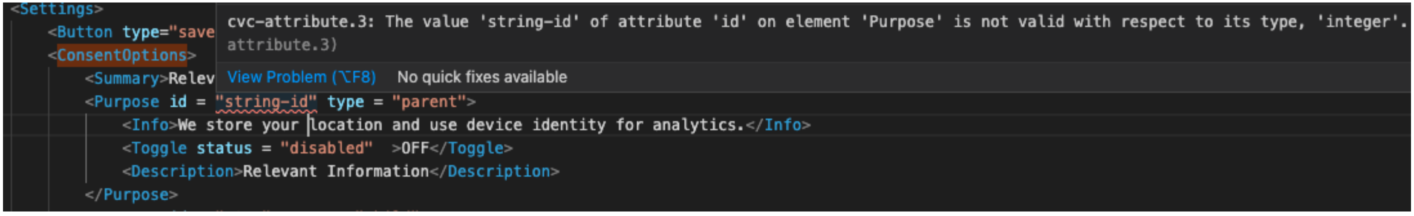
Now we test some cases that defy the structure –

Case 1: Changing the Type of Button. Which is restricted to prevent uncertainty in the knowledge of the consent dialogue structure.



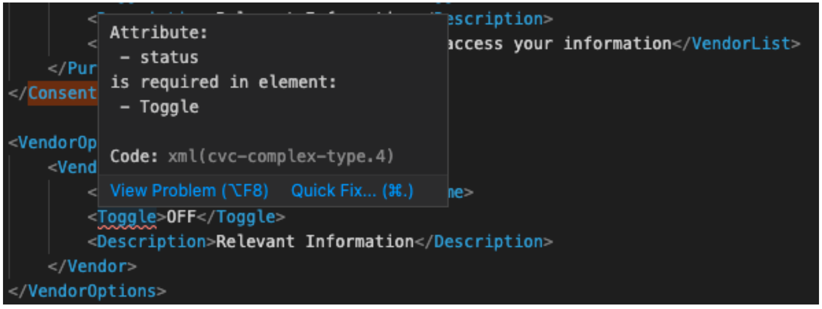
```
    <Title>rt. co  Value 'test' is not in the enumeration list.
    <!-- Heading
    <Heading>Acce  It must be one of the following:
    <!-- Informat    - agree
    processing an    - disagree
    Allowing the     - customize
    -->              - save
    <Summary>Rele
    <!-- Allow Da  Code: xml(cvc-enumeration-valid)
    <Button type=
    <!-- Customiz  cvc-attribute.3: The value 'test' of attribute 'type' on element 'Button' is not valid
    <Button type=  attribute.3)
    <!-- Disallow  View Problem (⌥F8)   Quick Fix... (⌘.)
    <Button type="test">Reject All</Button>
</Banner>
```

Case 2: Purpose is unique identifier based on integer value. Hence, breaks on defining with a character input.



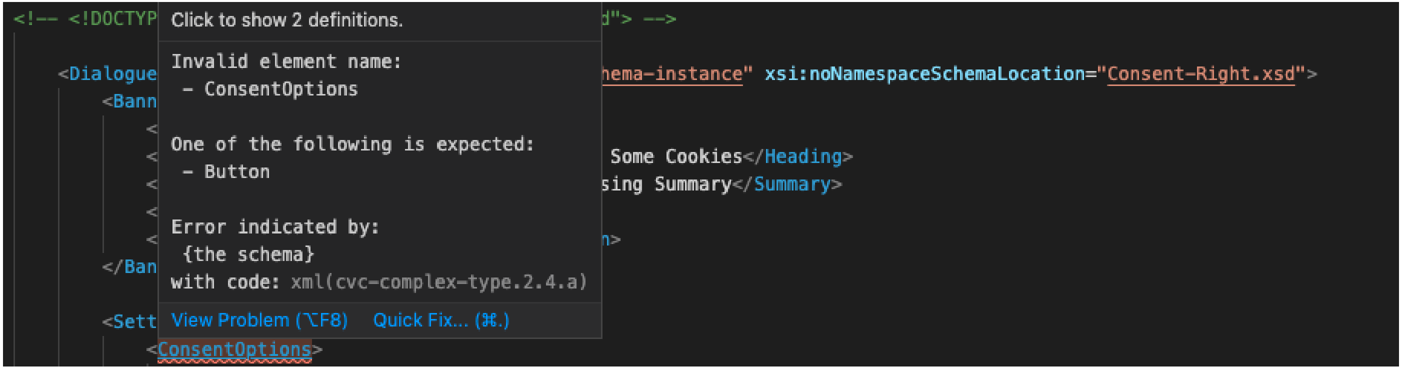
```
<Settings>
    <Button type="save  cvc-attribute.3: The value 'string-id' of attribute 'id' on element 'Purpose' is not valid with respect to its type, 'integer'.
    <ConsentOptions>    attribute.3)
        <Summary>Relev  View Problem (⌥F8)   No quick fixes available
        <Purpose id = "string-id" type = "parent">
            <Info>We store your location and use device identity for analytics.</Info>
            <Toggle status = "disabled"  >OFF</Toggle>
            <Description>Relevant Information</Description>
        </Purpose>
```

Case 3: Toggle Element has a required attribute status which is must present in order keep the integrity. It must be set to disable by default so that user makes a choice upfront.



```
            <             escription>
            < Attribute:  access your information</VendorList>
                 - status
        </Pur  is required in element:
    </Consent    - Toggle

    <VendorOp
        <Vend  Code: xml(cvc-complex-type.4)
            <  View Problem (⌥F8)   Quick Fix... (⌘.)   ne>
            <Toggle>OFF</Toggle>
            <Description>Relevant Information</Description>
        </Vendor>
    </VendorOptions>
```

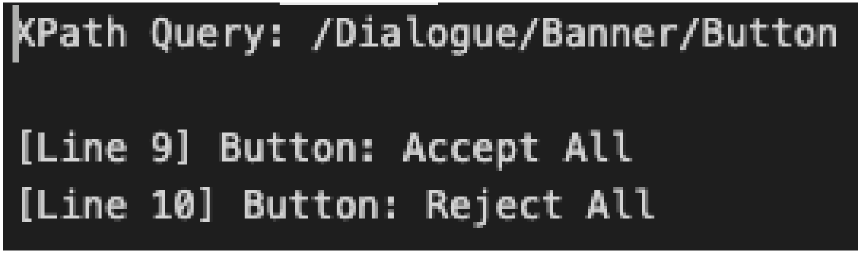
Case 4: Button must be present in Customizable Settings to provide users an option to trigger the changes.



```
<!-- <!DOCTYP  Click to show 2 definitions.   d"> -->

    <Dialogue  Invalid element name:         hema-instance" xsi:noNamespaceSchemaLocation="Consent-Right.xsd">
        <Bann   - ConsentOptions
            <
            <  One of the following is expected:   Some Cookies</Heading>
            <   - Button                          sing Summary</Summary>
            <
            <  Error indicated by:               n>
        </Ban   {the schema}
               with code: xml(cvc-complex-type.2.4.a)
        <Sett  View Problem (⌥F8)   Quick Fix... (⌘.)
            <ConsentOptions>
```

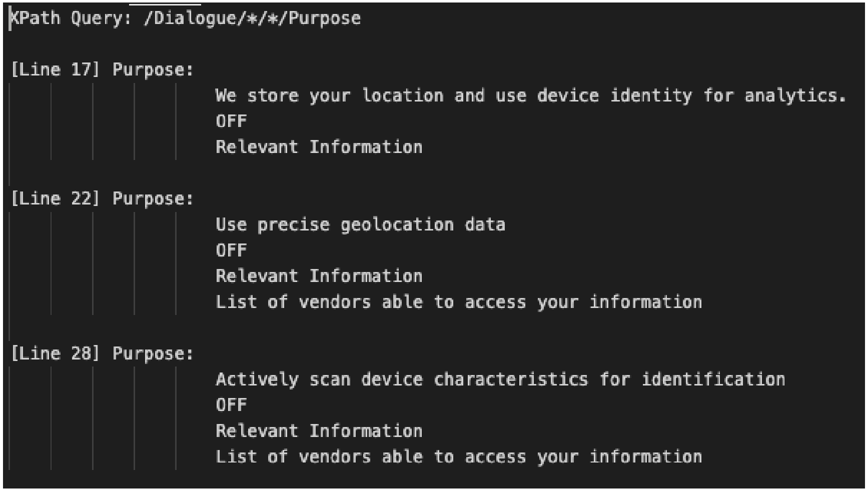
d) XPath on XML Documents

XPath Query 1 – XPath can be useful to select certain button which can be used to automate some workflow. We are able to retrieve the buttons with certain type (if specified).



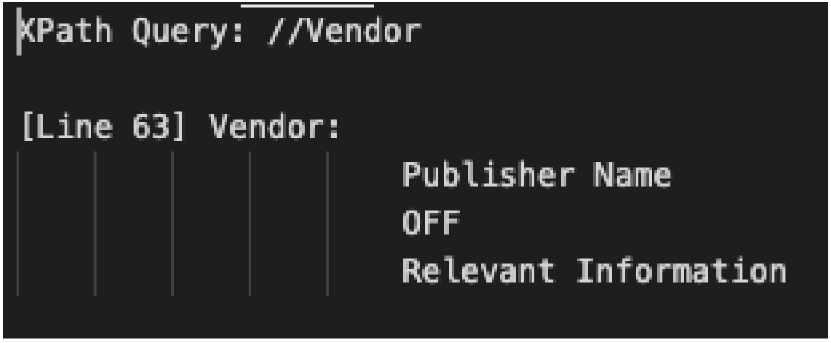
```
XPath Query: /Dialogue/Banner/Button

[Line 9] Button: Accept All
[Line 10] Button: Reject All
```

XPath Query 2 – Using XPath, we can learn more about the purposes and toggles if enabled in the consent dialogue for ease of understanding & information retrieval.



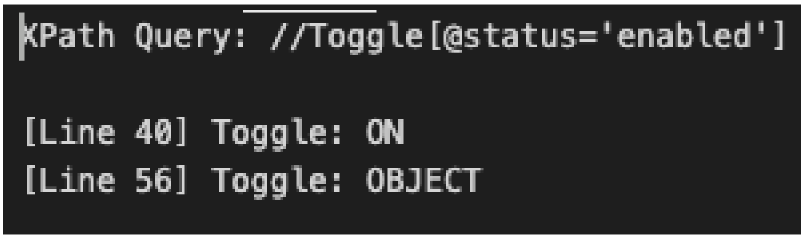
```
XPath Query: /Dialogue/*/*/Purpose

[Line 17] Purpose:
                    We store your location and use device identity for analytics.
                    OFF
                    Relevant Information

[Line 22] Purpose:
                    Use precise geolocation data
                    OFF
                    Relevant Information
                    List of vendors able to access your information

[Line 28] Purpose:
                    Actively scan device characteristics for identification
                    OFF
                    Relevant Information
                    List of vendors able to access your information
```

XPATH Query 3 – // Can be used to List All the Vendors to easily figure out enrolled partners



```
XPath Query: //Vendor

[Line 63] Vendor:
                    Publisher Name
                    OFF
                    Relevant Information
```

XPATH Query 4 – Using @attribute=value can be used to find which toggle is enabled and consent dialogue requires it to be disabled.



```
XPath Query: //Toggle[@status='enabled']

[Line 40] Toggle: ON
[Line 56] Toggle: OBJECT
```

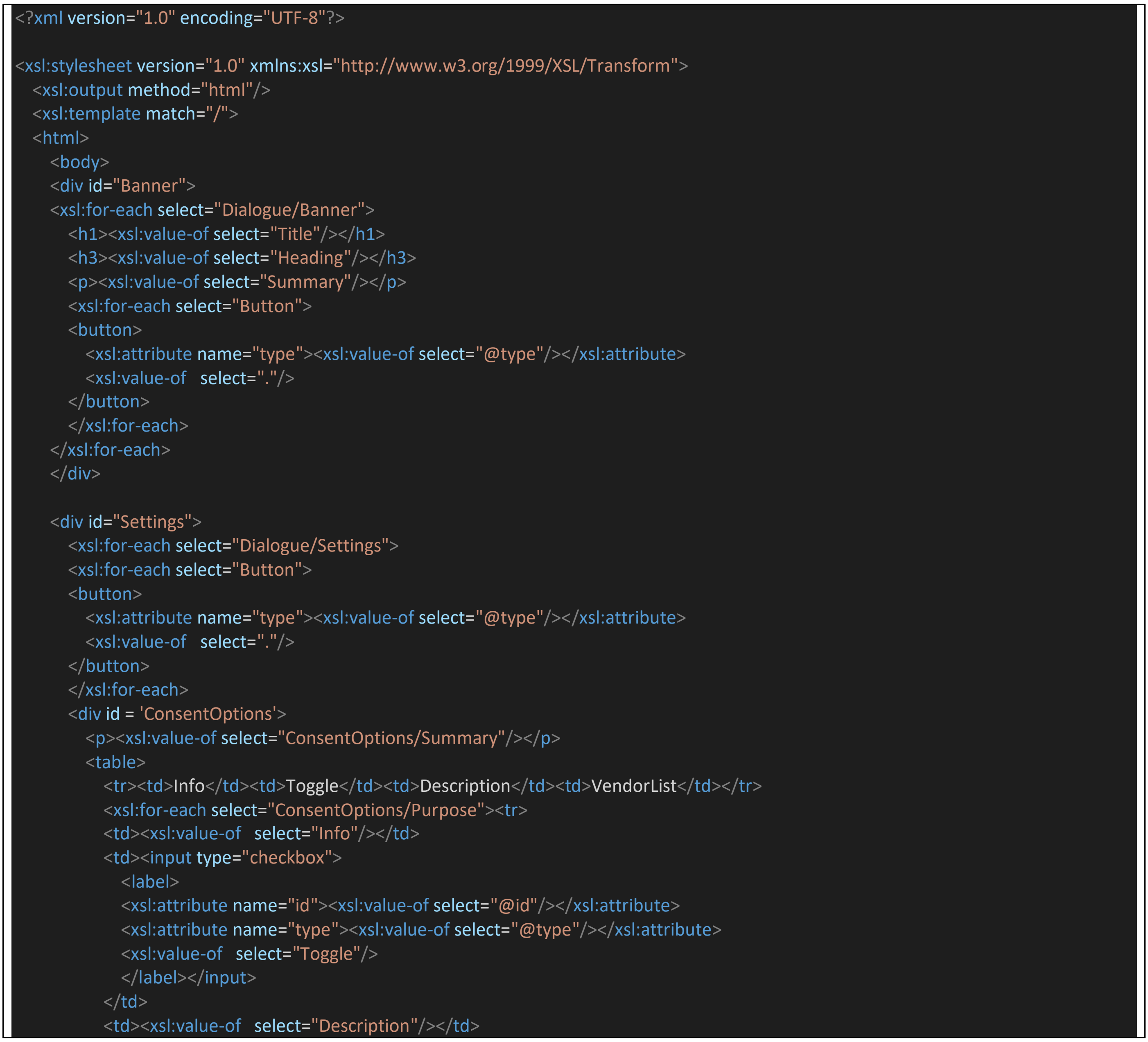
e) HTML Transformation from XSD creating XSLT

The XSLT created to transform our XSD to HTML is progressed. The translation of a XML document as discussed in Part c) of the question is taken from a unified XML developed to idealise the knowledge of consent dialogue generation on web pages. With initial declaration of XSL parameters referring to its stylesheet and output as HTML document. The procedure is discussed as follows –

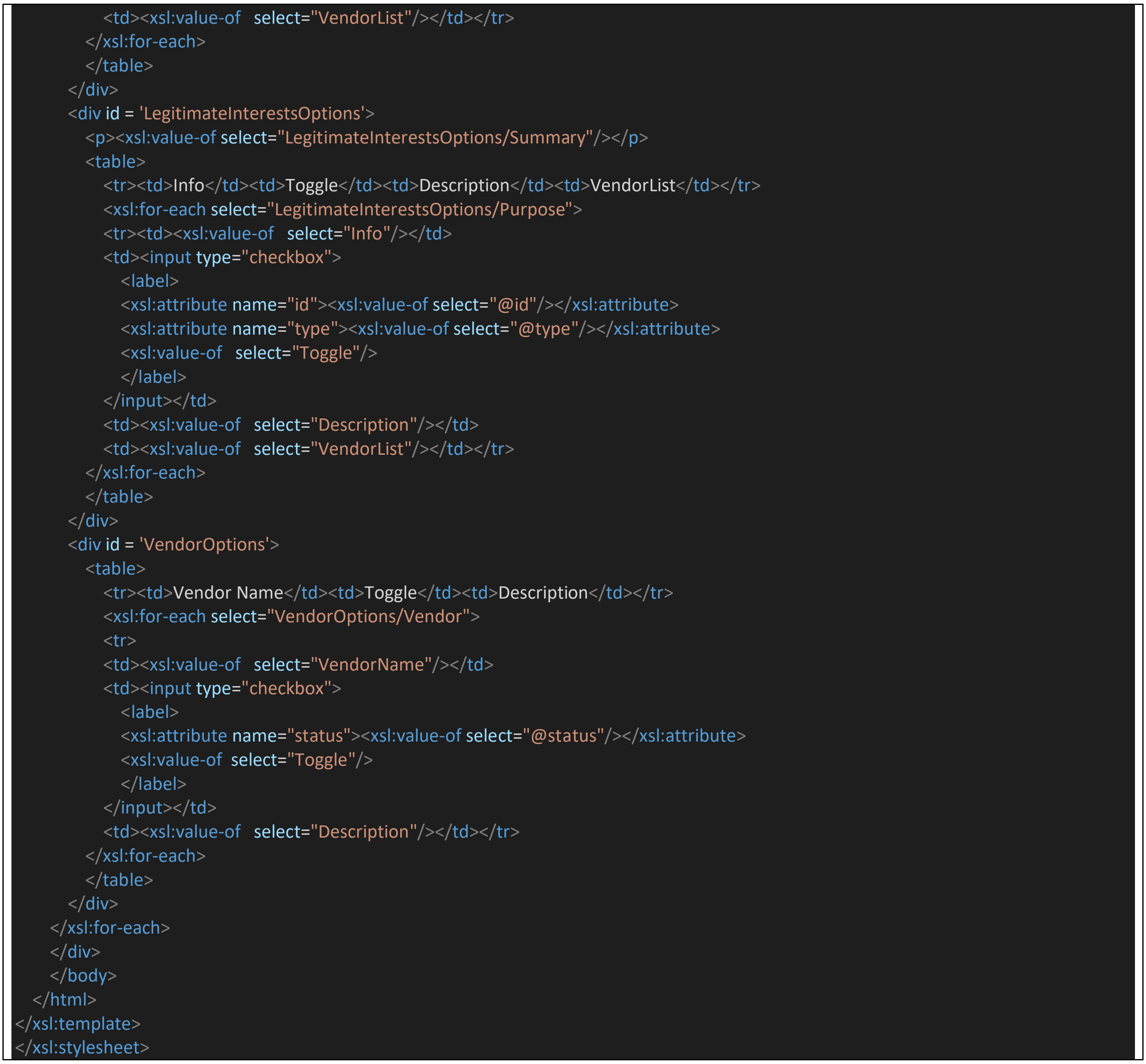
i) A <body> consists of two <div> corresponding to the Banner and Settings which can be handled by background JavaScript to appear based on Button clicked.

ii) Within the Banner, the title, heading and summary is inherited from the XML document based on XSD validation and all the buttons with their relevant trigger corresponding to their type attributes are defined.

iii) Within the Settings, segregation of 3 <div> respective of their elements is presented. Similar to button transformation applied in the Banner all specified buttons in XML are retrieved. Consent Options are mandatory and have purposes with relevant information to consent is defined. This is structured in table which organises the purpose information, toggle and its type attribute where parent properties are inherited in the child type purpose. Lastly, the VendorOptions which comprises of Vendor Name, Toggle and Description is defined.

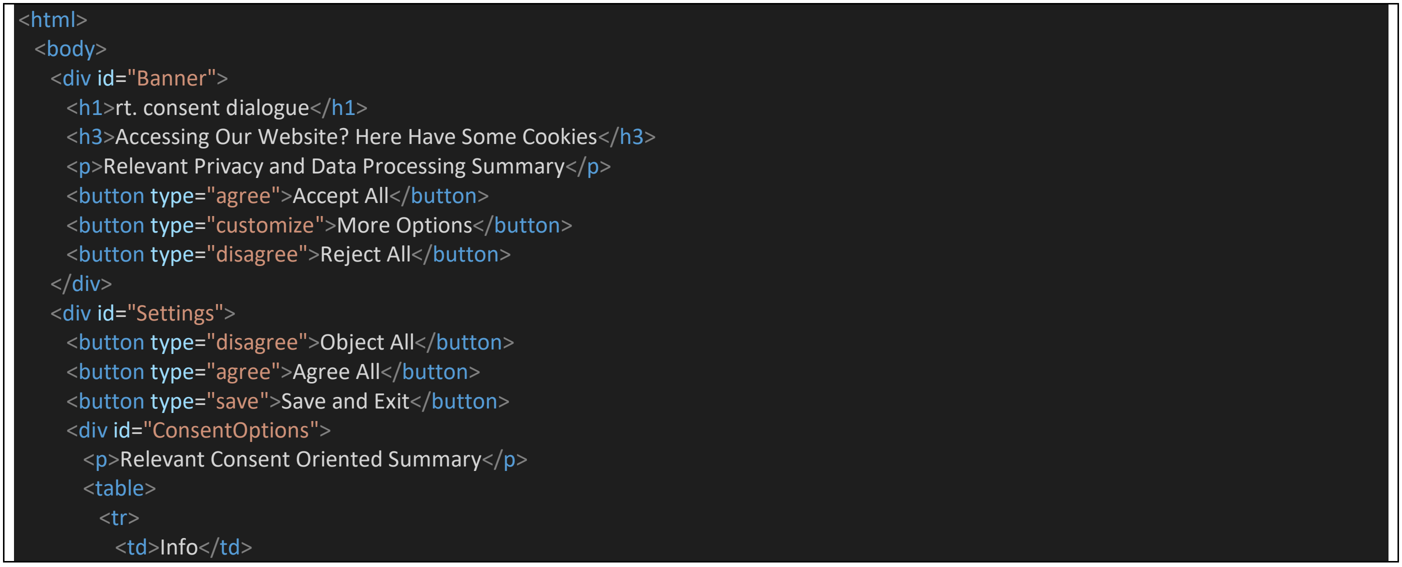


```
<?xml version="1.0" encoding="UTF-8"?>

<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="html"/>
  <xsl:template match="/">
  <html>
    <body>
    <div id="Banner">
    <xsl:for-each select="Dialogue/Banner">
      <h1><xsl:value-of select="Title"/></h1>
      <h3><xsl:value-of select="Heading"/></h3>
      <p><xsl:value-of select="Summary"/></p>
      <xsl:for-each select="Button">
      <button>
        <xsl:attribute name="type"><xsl:value-of select="@type"/></xsl:attribute>
        <xsl:value-of   select="."/>
      </button>
      </xsl:for-each>
    </xsl:for-each>
    </div>

    <div id="Settings">
      <xsl:for-each select="Dialogue/Settings">
      <xsl:for-each select="Button">
      <button>
        <xsl:attribute name="type"><xsl:value-of select="@type"/></xsl:attribute>
        <xsl:value-of   select="."/>
      </button>
      </xsl:for-each>
      <div id = 'ConsentOptions'>
        <p><xsl:value-of select="ConsentOptions/Summary"/></p>
        <table>
          <tr><td>Info</td><td>Toggle</td><td>Description</td><td>VendorList</td></tr>
          <xsl:for-each select="ConsentOptions/Purpose"><tr>
          <td><xsl:value-of   select="Info"/></td>
          <td><input type="checkbox">
            <label>
            <xsl:attribute name="id"><xsl:value-of select="@id"/></xsl:attribute>
            <xsl:attribute name="type"><xsl:value-of select="@type"/></xsl:attribute>
            <xsl:value-of   select="Toggle"/>
            </label></input>
          </td>
          <td><xsl:value-of   select="Description"/></td>
```



```
          <td><xsl:value-of  select="VendorList"/></td></tr>
        </xsl:for-each>
        </table>
      </div>
      <div id = 'LegitimateInterestsOptions'>
        <p><xsl:value-of select="LegitimateInterestsOptions/Summary"/></p>
        <table>
          <tr><td>Info</td><td>Toggle</td><td>Description</td><td>VendorList</td></tr>
          <xsl:for-each select="LegitimateInterestsOptions/Purpose">
          <tr><td><xsl:value-of  select="Info"/></td>
          <td><input type="checkbox">
            <label>
            <xsl:attribute name="id"><xsl:value-of select="@id"/></xsl:attribute>
            <xsl:attribute name="type"><xsl:value-of select="@type"/></xsl:attribute>
            <xsl:value-of  select="Toggle"/>
            </label>
          </input></td>
          <td><xsl:value-of  select="Description"/></td>
          <td><xsl:value-of  select="VendorList"/></td></tr>
        </xsl:for-each>
        </table>
      </div>
      <div id = 'VendorOptions'>
        <table>
          <tr><td>Vendor Name</td><td>Toggle</td><td>Description</td></tr>
          <xsl:for-each select="VendorOptions/Vendor">
          <tr>
          <td><xsl:value-of  select="VendorName"/></td>
          <td><input type="checkbox">
            <label>
            <xsl:attribute name="status"><xsl:value-of select="@status"/></xsl:attribute>
            <xsl:value-of  select="Toggle"/>
            </label>
          </input></td>
          <td><xsl:value-of  select="Description"/></td></tr>
        </xsl:for-each>
        </table>
      </div>
    </xsl:for-each>
    </div>
    </body>
  </html>
</xsl:template>
</xsl:stylesheet>
```

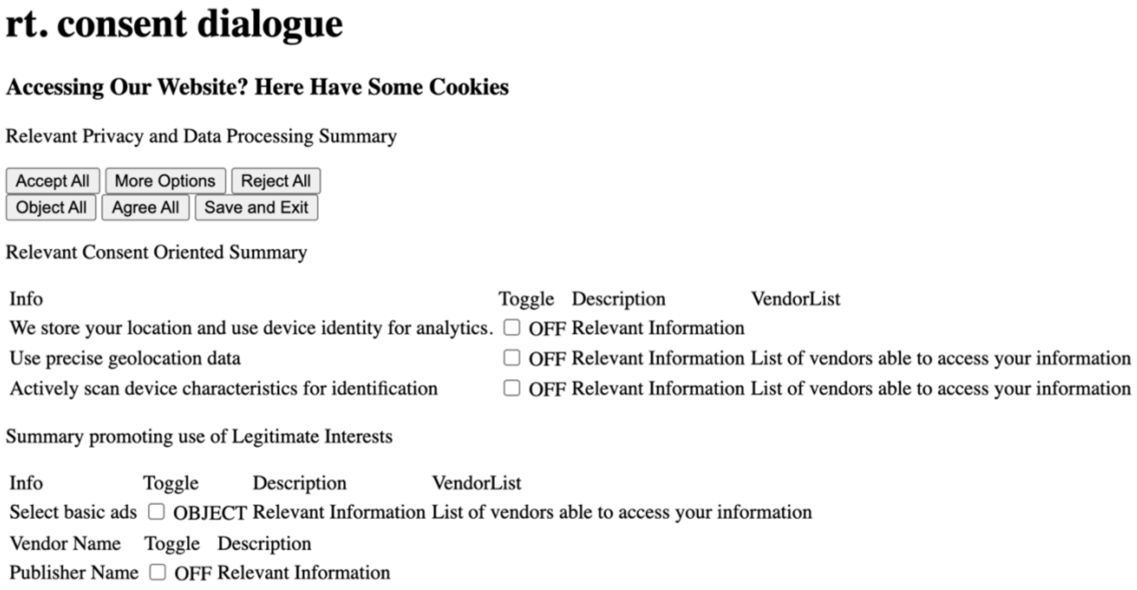
The generated HTML is as follows –



```
<html>
  <body>
    <div id="Banner">
      <h1>rt. consent dialogue</h1>
      <h3>Accessing Our Website? Here Have Some Cookies</h3>
      <p>Relevant Privacy and Data Processing Summary</p>
      <button type="agree">Accept All</button>
      <button type="customize">More Options</button>
      <button type="disagree">Reject All</button>
    </div>
    <div id="Settings">
      <button type="disagree">Object All</button>
      <button type="agree">Agree All</button>
      <button type="save">Save and Exit</button>
      <div id="ConsentOptions">
        <p>Relevant Consent Oriented Summary</p>
        <table>
          <tr>
            <td>Info</td>
          <td>Toggle</td>
          <td>Description</td>
          <td>VendorList</td>
        </tr>
        <tr>
          <td>We store your location and use device identity for analytics.</td>
          <td>
            <input type="checkbox">
              <label id="1010" type="parent">OFF</label>
            </input>
          </td>
          <td>Relevant Information</td>
        </tr>
        <tr>
          <td>Use precise geolocation data</td>
          <td>
            <input type="checkbox">
              <label id="1011" type="child">OFF</label>
            </input>
          </td>
          <td>Relevant Information</td>
          <td>List of vendors able to access your information</td>
        </tr>
        <tr>
          <td>Actively scan device characteristics for identification</td>
          <td>
            <input type="checkbox">
              <label id="1012" type="child">OFF</label>
            </input>
          </td>
          <td>Relevant Information</td>
          <td>List of vendors able to access your information</td>
        </tr>
      </table>
    </div>
    <div id="LegitimateInterestsOptions">
      <p>Summary promoting use of Legitimate Interests</p>
      <table>
        <tr>
          <td>Info</td>
          <td>Toggle</td>
          <td>Description</td>
          <td>VendorList</td>
        </tr>
        <tr>
          <td>Select basic ads</td>
          <td>
            <input type="checkbox">
              <label id="201" type="">OBJECT</label>
            </input>
          </td>
          <td>Relevant Information</td>
          <td>List of vendors able to access your information</td>
        </tr>
      </table>
    </div>
    <div id="VendorOptions">
      <table>
        <tr>
          <td>Vendor Name</td>
          <td>Toggle</td>
          <td>Description</td>
        </tr>
        <tr>
          <td>Publisher Name</td>
```

```
            <td>
              <input type="checkbox">
                <label status="">OFF</label>
              </input>
            </td>
            <td>Relevant Information</td>
          </tr>
        </table>
      </div>
    </div>
  </body>
</html>
```

The generated Consent Dialogue is as follows –



rt. consent dialogue

Accessing Our Website? Here Have Some Cookies

Relevant Privacy and Data Processing Summary

Relevant Consent Oriented Summary

Info	Toggle	Description	VendorList
We store your location and use device identity for analytics.	□ OFF	Relevant Information	
Use precise geolocation data	□ OFF	Relevant Information	List of vendors able to access your information
Actively scan device characteristics for identification	□ OFF	Relevant Information	List of vendors able to access your information

Summary promoting use of Legitimate Interests

Info	Toggle	Description	VendorList
Select basic ads	□ OBJECT	Relevant Information	List of vendors able to access your information

Vendor Name	Toggle	Description
Publisher Name	□ OFF	Relevant Information

f) XML to RDF Conversion using XSLT

The transformation from XML to RDF via XSLT can be performed by generating URIs from IDs or name attributes to map the XML to RDF vocabulary terms. A structured set of RDF triples expresses a directed graph labelling elements in different form in XML. Using XSLT, the elementary representation of attributes with its name and value pair is converted into a triple parameter, an example is shown as follows –
A sample button taken from the XML Document.

```
<Dialogue>
<Banner>
<Button type="agree">Accept All</Button>
</Banner>
</Dialogue>
```

This can be written in as follows (taking ancestor into account) –

RDF/XML	RDF Triple Language
<rdf:Description rdf:about="Dialogue/Banner/Button"> <cr:button>Accept All</cr:button> </rdf:Description> </rdf:RDF>	<Dialogue/Banner/Button> cr:type "agree" rdf:value "Accept All"

In RDF declaration on repetition of element addition _2 or _x (x – depicting a continual integer is added) to conserve the order of elements.

Hereby our question 1 is concluded.

Innovation in Web Search - Essay

With a revolution in the World Wide Web, the Internet has grown at an enormous scale. Traditional web technologies are replaced with an introduction of modern research based technology stack to overcome the prior limitations and improve efficiency of the interconnected network – the Web. Traditional web allows its user to access and retrieve information or services available on the internet by routing them directly based on location of the resources which is in form a hyperlink (generally known as URL) or its address. Retrieval of information on the internet to general public became a great deal and has been researched since the birth of the Internet.

Popular web search engines like Google, Bing or DuckDuckGo takes the approach to the next level in retrieval of Information on the Internet through algorithmic evolution in these service providers. The traditional approach of the search engine faces several challenges in organising the exponentially expanding database of the World Wide Web and maintaining an index of the growing data available online. This further results in depleting the quality and performance of these search engines to navigate through available internet services, published original content, or finding the right piece of information. In addition to this, many web service providers update information frequently or generate data dynamically on their webpages causing the search engines to actively update the information revisiting the webpages, or also resulting in these sites to not being indexed in the web search. Currently, the web search primarily works on the idea of retrieving information by matching phrases, keywords or citations. The search engines' approach is driven on the constantly growing vocabulary of the internet, marking certain limitations for these services to function effectively. The volatility of information on the web states a figure of 60% data is changed or updated per month, followed by a large volume of distributed data consisting of redundancy and unstructured information which is available to the user. The amount of information written in different languages, formatting and media type causes heterogeneity of data making it harder to generalise or adaptable for search engines to locate or navigate toward the correct resources.

As of January 2nd 2021, the web consists of over **1.826 bn websites** which are further structured to have webpages raising a thought of the amount of information available in these websites is immeasurable. With the amount of redundant information present, it's expected that the 30% of data is duplicate. This also raises a concern for unstructured data hard to be retrieved or monitored by the search engines ruling out the possibility of querying or logically fetching information from these websites. The anomalies present in these websites such as grammar, syntactical or human errors also reduces the quality of the information resulting in prevention of these resources to be tracked by the search engine providers. Presently, the information available on the internet is structurally designed for humans and barely understood by the machines. The semantic-based approach in the developing a future proof search engine plays a vital role in meaningfully structuring of the available information on the web to be understandable by both humans and machines effectively.

> *"The Semantic Web isn't just about putting data on the web. It is about making links, so that a person machine can explore the web of data. With linked data, when you have some of it, you can find other, related, data."*
>
> Tim Berners-Lee [1]

The W3 Consortium's vision of the web as a set of linked data exercise the idea for Semantic Web. The gigantic database of information on the internet can be addressed conveniently building vocabularies and defining rules for handling the data. Semantic Interpolation of data translates to connecting the information in a well-structured and definitive form expanding its reach and simplification in understanding the data. The semantic architecture further breaks down into 7 web layers as shown in the semantic web stack in the figure. The term Semantic Web, itself is used to identify the set of web technologies, standards and rules defined to formulate a basic structure of any system online.

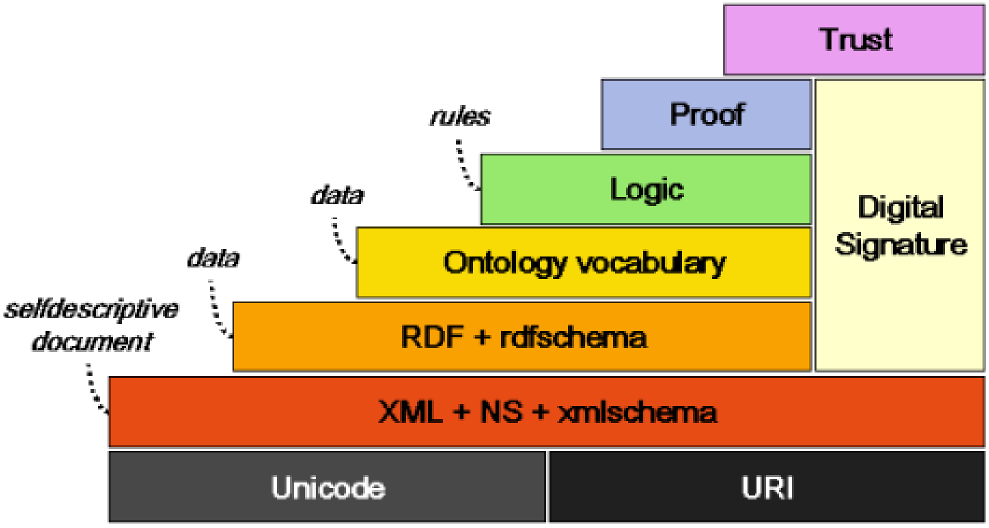


Figure 1. Semantic Web Stack Diagram

The hierarchal foundation of the system is further discussed. The layer at the bottom specifies the actual location of the resource present on the internet, identifying the point in a space of information – the Web. The schematic XML layer explores self-descriptive documentation, definition of the document tree, rules for vocabularies of the information and additionally, bringing a support for tools and technologies helping in declaring namespaces, restricting document structure, parsing correctly formatted data and an ability to navigate through the available database. Provided with the structure, the RDF or Resource Description Framework consists of describing information on the web present its exact meaning based on its semantics – a Resource, its property and a certain value assigned to it which can understood directly by computers resulting in increased precision when retrieving the information. RDF topped by another layer representing data, Ontology Vocabulary which outlines the concepts of interlinking the specified area of knowledge represented by relational link. The definition rules for logically reasoning a structured information on the web based upon the Logic layer following which the study on Proof and Trustable content is still continued today.

Information retrieval in a search engine is evaluated based on the following parameters – Recall & Precision. Presently, two main types of search engines are prevalent – **Web Crawler** based and **Human Powered Directory** based. The former is laid on the foundational idea of exhaustive automated web navigation and retrieval of relevant information stored into a centralised database ensuring high recall of the content as enormous database is created but resulting in lower precision of the obtained results. Whereas, the latter is developed upon manually reviewed database which leads to lower recall and a higher precision in appropriate content retrieval. The algorithmic approach has evolved based on scientific research and engineering but the large scale automated search engine still takes a trade off when dealing with immeasurable data set of information in context of their precision. Schematic structuring in XML represents a favourable condition for knowledge representation and the semantic approach incorporated in the search engines which results in an efficient query resolution using the systematic ontologies. With a higher recall, an increase in precision can be attained over traditional web search pursuing the use of ontologies.

The possibilities of an efficient and a better performing web search is endless taking the semantic-based approach but certain limitations arises when technically implemented. Semantic Web works on the development of descriptive vocabularies, conceptualized ontologies, and a rational management based on proof and trust model. The primary concern that is raised is constant changes in unique identifiers that is required to meet the ever evolving concepts of designed ontologies. The ontology design being publicly available and crowd sourced for consistency and scalable usage on the internet would require standardization at each level of definition and version tracked systematically. RDF can be taken into account for structural definition of metadata but still lacks formal semantics in its schematic extension – RDFS. In an open-source system of the semantic web, crowd contribution is welcome which raises concerns of credible sources and can result in possible conflicts. To further prevent any such happening, the contributor has to prove issuing a statement of trust, including a digital signature for insuring reliability of the source of information. Additionally, confidentiality can be achieved by restricting access control and use of encryption. The crucial concept modelling Trust cannot be standardized to apply on each developed semantic model. To build upon prevalent 'Web of Document' approach, a derived set of measurement of trust for each source a theory is proposed 'Web of Trust' and regarded as a linked source to a trust is automatically trusted with a lower extent leading to development of a hierarchal model to infer a knowledge of trust. Surrounding trust, privacy is another concern which has to be addressed to make GDPR compliant model of developed vocabularies. Presently, concept around Trust and Proof are not formalised and are in development for a commercial application leading to **technical limitation** of Semantic-based Web Search implementation.

The **future of Semantic Web** is unpredictably immense. The structural linked data approach in semantic web promotes large scale integration of information and its reasoning on the web. The revolutionized vision for Web 3.0 can be predicted to be more application based inheriting the advancements of Semantic Web than the current Web 2.0 focusing on document approach. The introduction of formalized Trust & Proof based scientific approach on top of the underlying structure of the Semantic Web Stack would encourage many opportunities specially in the domain of Personalization given the relational dependency of data on the web. Based on current progression, with available resources to accommodate Big Data and perform processing on a constantly increasing set of available data on the web. This could positively be impacted taking this vocabulary based approach and result in an efficient growth of information on internet. The scope of Internet of Things, highly depends upon the ever evolving methodologies to handle data in form of relation using definite framework such as RDF to innovate the existing digital libraries and optimising the growing web agents in automating menial tasks in the daily world of distributed computing. Semantic web enables its use case of effectively delivering web services to the end users by a specification of flow of actions reducing the workload of computing devices. Also, with introduction of user agents to assists with certain web activities would be able to provide internet users to optimise their work and focus on delivering real world business problems. The conventional client-service based web applications would be replaced by AI-based mobile agents. The semantic based web search is extensively discussed following which modern semantic based digital libraries which would play a vital role in delivering digital media content following which an integration of the modern web and software technologies would be accompanied like present day Amazon Alexa, Apple Siri and other popular personal assistants helping in delivering specific information just by the power of speech also preventing any manual navigation through the applications. Concluding the future of semantic web, which is known to grow and massively develop upon the vison of next-generation ease of access, understanding and visualization of information to both humans and the computers, an exponential increase in machine knowledge is foreseeable.

References

[1] "Linked Data - Design Issues." https://www.w3.org/DesignIssues/LinkedData.html (accessed May 22, 2021).

[2] O. Conlan, "4D2b – Navigating an XML Document," p. 28.

[3] O. Conlan, "4D2b – Transforming XML Documents," p. 31.

[4] "(PDF) A standard transformation from XML to RDF via XSLT," *ResearchGate*, doi: 10.1002/asna.200811233.

[5] H. J. Pandit, C. Debruyne, D. O'Sullivan, and D. Lewis, "GConsent - A Consent Ontology Based on the GDPR," in *The Semantic Web*, Cham, 2019, pp. 270–282. doi: 10.1007/978-3-030-21348-0_18.

[6] P. F. Patel-Schneider and D. Fensel, "Layering the Semantic Web: Problems and Directions," in *The Semantic Web — ISWC 2002*, vol. 2342, I. Horrocks and J. Hendler, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 16–29. doi: 10.1007/3-540-48005-6_4.

[7] "Semantic Web - extended.pdf." Accessed: May 26, 2021. [Online]. Available: https://www.scss.tcd.ie/Owen.Conlan/CS7063/Semantic%20Web%20-%20extended.pdf

[8] S. Lu, M. Dong, and F. Fotouhi, "The Semantic Web: Opportunities and Challenges for Next-Generation Web Applications," *Int. J. Inf. Res.*, vol. 7, p. 2002, 2002.