



slington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CS5004NI Emerging Programming Platforms and Technologies

Assessment Weightage & Type

30% Individual Coursework

Year and Semester

2020-21 Spring

Student Name:Pratigya Rana

London Met ID:19033702

College ID:NP01CP4S200065

Assignment Due Date:7 May 2021

Assignment Submission Date:6 May 2021

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Table of content

1.Introduction	1
2.XML Content	2
2.1Tree Diagram.....	2
2.2XML Content.....	3
3.Schema Content	9
4.DTD Content	16
5.Testing	16
5.1Test 1: Validating XML file	17
5.2Test 2: Validating XML file with schema file.....	19
5.3Test 3: Validating XML file with DTD file.....	21
5.4Test 4:Checking if CSS file is rendered by browser	23
5.5Test 5: Checking hover in images.....	24
6.Development of Coursework.....	26
7.Difference between Schema and DTD	29
8.Critical Evaluation	30
9.Conclusion.....	31
10.Reference.....	32

List of Figures

Figure 1:Tree diagram	2
Figure 2:Validating XML file	17
Figure 3:Result of XML validation	18
Figure 4:Validating Xml with Schema file	19
Figure 5:Result of Validating XML with schema file	20
Figure 6:Validating XML with dtd file	21
Figure 7:Result of validating XML with dtd file.....	22
Figure 8:Result after rendering CSS by browser	23
Figure 9:Before hovering	24
Figure 10:After hovering.....	25
Figure 11:Logo of Notepad++ (Notepad++, 2021)	26
Figure 12:Logo of Draw.io (draw.io, 2021)	27
Figure 13:Logo of Visual Studio Code (Visual Studio Code-icon, 2021).	28

List of Tables

Table 1:Test 1	17
Table 2:Test 2	19
Table 3:Test 3	21
Table 4:Test 4	23
Table 5:Test 5	24

1.Introduction

This coursework is an individual coursework of “Emerging Programming Platforms and Technologies” module having 30% weightage of the total module grades. In this coursework, an online model of a system has to be developed for a music store using XML. So, an XML file has to be created which should be linked with schema file and (Data Type Definition) dtd file. A CSS file is also required to be created in order to render the XML file in a web browser.

XML Stands for Extensible Markup Language which is used to define documents in a conventional way which can be easily interpreted by any application compatible to XML. Though, XML is not a markup language but a metalanguage which can be used with HTML pages and is also used to create markup language for Particular applications (TechTerms, 2021).

XSD stands for XML Schema Definition which is used for describing the XML document structure. It was first published in 2001. A XML document contains different element and attributes and the XSD is used to define rules for them. It can be used as a method to determine the correctness of the XML file’s structure (Singh, 07 July 2020).

DTD stands for Data Type Documentation which is a set of markup declaration used to define document of any Standard Generalized Markup Language (SGML) like GML, HTML, XML.DTD in XML, can be declared internally or externally in the document and determines the occurrences of the parent and child element (GeeksforGeeks, 6 November 2020).

CSS stands for Cascading Style Sheet which is used for formatting the webpages layout. CSS is used in replace of webpage’s HTML to define text styles, table sizes and other details of the page. It is very useful to web developers as it makes easy to change the styles across various pages at once (TechTerms, 2021).

2.XML Content

2.1Tree Diagram

The tree diagram of my coursework is:

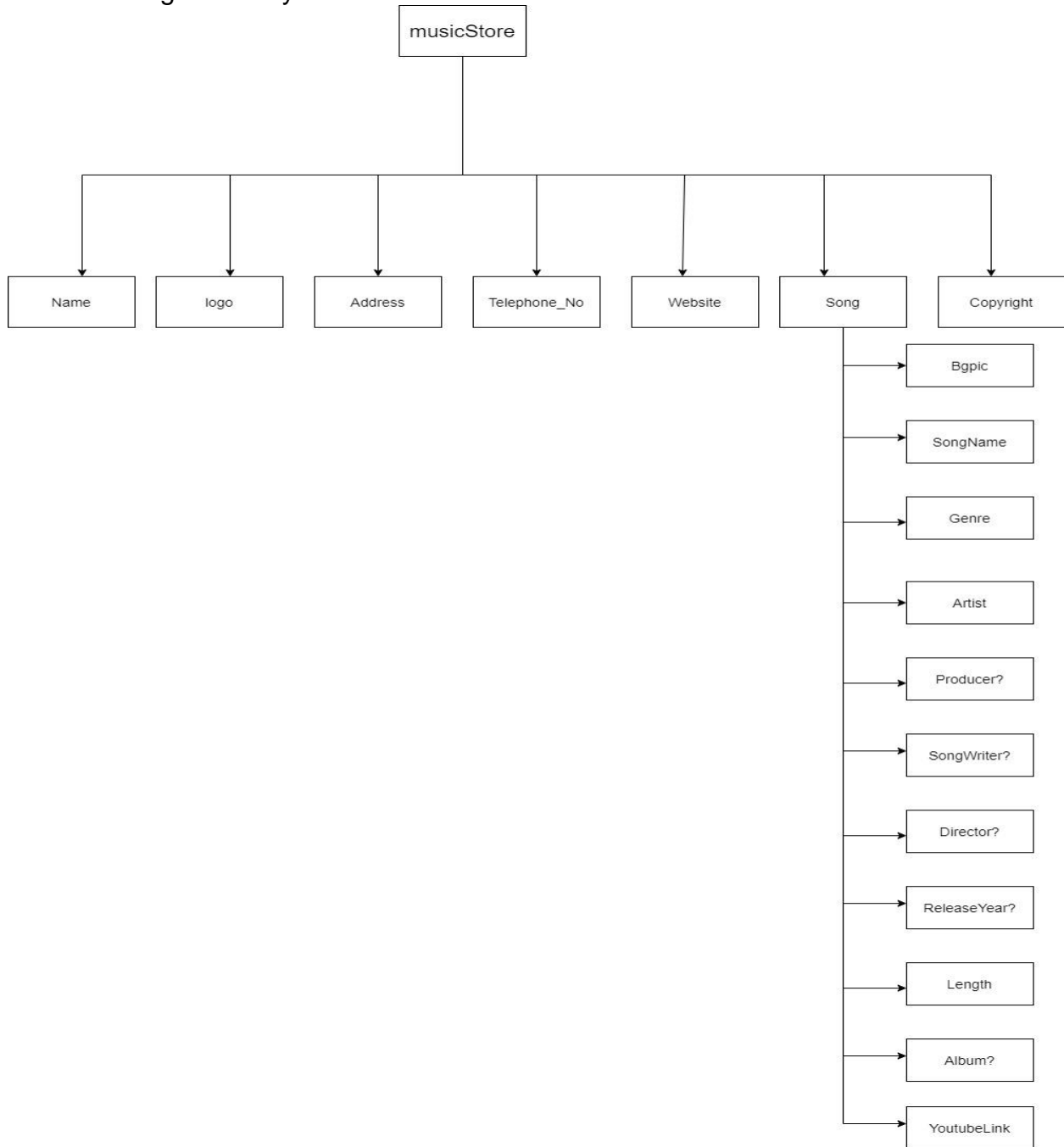


Figure 1:Tree diagram

2.2XML Content

The XML content of my coursework is:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="catalog_19033702.css"?>
<!DOCTYPE musicStore SYSTEM "catalog_19033702.dtd">
<!-- <musicStore xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="catalog_19033702.xsd"> -->
<musicStore>
  <header>

    <logo></logo>

    <Name>Mandu Music Store</Name>
    <Address>Thamel,kathmandu</Address>
    <Telephone_No>+937822729</Telephone_No>
    <Website>http://mandumusic.com</Website>

  </header>
  <Shelf>
    <first_row>
      <Song id="first_song">
        <bgPic id="one"></bgPic>
        <SongName>Name: Solo</SongName>
        <Genre>Genre: Kpop</Genre>
        <Artist type="solo">Artist: Jennie Kim</Artist>
        <Producer>Producer: Teddy park</Producer>
        <SongWriter>SongWriter: Teddy park</SongWriter>
        <Director>Director: Han sa-min</Director>
        <ReleaseYear>ReleaseYear: 12 November 2018</ReleaseYear>
        <Length songLength="short">Length: 02 mins 56 sec</Length>
        <YoutubeLink link_id="1">YoutubeLink:
https://youtu.be/b73BI9eUkjM</YoutubeLink>
      </Song>
      <Song id="second_song">
        <bgPic id="two"></bgPic>
        <SongName>Name: So What</SongName>
        <Genre>Genre: Jazz</Genre>
        <Artist type="solo">Artist: Miles Davis</Artist>
        <Producer>Producer: Teo Macreo</Producer>
        <SongWriter>SongWriter: Miles Davis</SongWriter>
        <ReleaseYear>ReleaseYear: 17 August 1959</ReleaseYear>
        <Length songLength="veryLong">Length: 09 mins 22 sec</Length>
        <Album>Album: Kind of Blue</Album>
        <YoutubeLink link_id="2">YoutubeLink:
https://youtu.be/yIXk1LBvlqU</YoutubeLink>
```

```

</Song>
<Song id="third_song">
  <bgPic id="three"></bgPic>
  <SongName>Name: We will rock you</SongName>
    <Genre>Genre: Rock</Genre>
  <Artist type="band">Artist: Queen</Artist>
  <Producer>Producer: Mike stone</Producer>
  <SongWriter>SongWriter: Brian May</SongWriter>
  <Director>Director: Christopher Renshaw</Director>
  <ReleaseYear>Releaseyear: 1977</ReleaseYear>
  <Length songLength="short">Length: 02 mins 2 sec</Length>
  <Album>Album: News of the World</Album>
  <YoutubeLink link_id="3">YoutubeLink: https://youtu.be/-tJYN-
eG1zk</YoutubeLink>
</Song>

</first_row>
<second_row>
<Song id="fourth_song">
  <bgPic id="four"></bgPic>
  <SongName>Name: How you like that</SongName>
    <Genre>Genre: Kpop</Genre>
  <Artist type="band">Artist: BlackPink</Artist>
  <Producer>Producer: Teddy Park</Producer>
  <SongWriter>Songwriter: Teddy Park, Kim jung-gu, Danny Chung</SongWriter>
  <Director>Director: Seo Hyun-seung</Director>
  <ReleaseYear>ReleaseYear: 26 June 2020</ReleaseYear>
  <Length songLength="average">Length: 03 mins 1 sec</Length>
  <Album>Album: The Album</Album>
  <YoutubeLink link_id="4">YoutubeLink:
https://youtu.be/ioNng23DkIM</YoutubeLink>
</Song>
<Song id="fifth_song">
  <bgPic id="five"></bgPic>
  <SongName>Name: Army</SongName>
    <Genre>Genre: Pop</Genre>
  <Artist type="solo">Artist: Ellie Goulding</Artist>
  <Producer>Producer: Max Martin, Ali Payami</Producer>
  <SongWriter>SongWriter: Ellie Goulding, Max Martin, Savan Kotecha, Ali
Payami</SongWriter>
  <Director>Director: Conor McDonnell</Director>
  <ReleaseYear>ReleaseYear: 9 January 2016</ReleaseYear>
  <Length songLength="average">Length: 03 mins 57 sec</Length>
  <Album>Album: Delirium</Album>
  <YoutubeLink link_id="5">YoutubueLink:
https://youtu.be/jTTNWpag6fl</YoutubeLink>

```



```

</Song>
<Song id="sixth_song">
  <bgPic id="six"></bgPic>
  <SongName>Name: Promises</SongName>
    <Genre>Genre: Dubstep</Genre>
  <Artist type="band">Artist: Nero</Artist>
  <Producer>Producer: Max Martin, Ali Payami</Producer>
  <SongWriter>SongWriter: Daniel Stephens, Joe Ray, Alana
Watson</SongWriter>
  <ReleaseYear>ReleaseYear: 5 August 2011</ReleaseYear>
  <Length songLength="long">Length: 04 mins 17 sec</Length>
  <Album>Album: Welcome Reality</Album>
  <YoutubeLink link_id="6">YoutubeLink:
https://youtu.be/CU6zwoWjWkg</YoutubeLink>
</Song>
</second_row>
<third_row>
<Song id="seventh_song">
  <bgPic id="seven"></bgPic>
  <SongName>Name: Friends in Low Places</SongName>
    <Genre>Genre: Country</Genre>
  <Artist type="solo">Artist: Garth Brooks</Artist>
  <Producer>Producer: Allen Reynolds</Producer>
  <SongWriter>SongWriter: Dewayne Blackwell, Earl Bud Lee</SongWriter>
  <ReleaseYear>ReleaseYear: 6 August 1960</ReleaseYear>
  <Length songLength="long">Length: 04 mins 18 sec</Length>
  <Album>Album: No Fences</Album>
  <YoutubeLink link_id="7">YoutubeLink:
https://youtu.be/p0_der_5hRM</YoutubeLink>
</Song>
<Song id="eighth_song">
  <bgPic id="eight"></bgPic>
  <SongName>Name: I Hope You Dance</SongName>
    <Genre>Genre: Country Pop</Genre>
  <Artist type="solo">Artist: Lee Ann Womack</Artist>
  <Producer>Producer: Mark Wright</Producer>
  <SongWriter>SongWriter: Mark D. Sanders, Tia Sillers</SongWriter>
  <Director>Director: Gerry Wenner</Director>
  <ReleaseYear>ReleaseYear: 9 January 2016</ReleaseYear>
  <Length songLength="long">Length: 04 mins 54 sec</Length>
  <Album>Album: I Hope You Dance</Album>
  <YoutubeLink link_id="8">YoutubeLink: https://youtu.be/RV-
Z1YwaOiw</YoutubeLink>
</Song>
<Song id="ninth_song">
  <bgPic id="nine"></bgPic>

```

```

    <SongName>Name: Sadda Haq</SongName>
      <Genre>Genre: Rock</Genre>
    <Artist type="solo">Artist: Mohit Chauhan</Artist>
    <Producer>Producer: A.R.Rahman</Producer>
    <SongWriter>SongWriter: A.R.Rahman, Irshad Kohli</SongWriter>
    <ReleaseYear>ReleaseYear: 30 September 2011</ReleaseYear>
    <Length songLength="veryLong">Length: 06 mins 5 sec</Length>
    <Album>Album: Rockstar</Album>
    <YoutubeLink link_id="9">YoutubeLink:
https://youtu.be/p9DQINKZxWE</YoutubeLink>
  </Song>
</third_row>
<fourth_row>
<Song id="tenth_song">
  <bgPic id="ten"></bgPic>
  <SongName>Name: Jo Tu Na Mila</SongName>
    <Genre>Genre: Pakistani Pop</Genre>
  <Artist type="solo">Artist: Asim Ajhar</Artist>
  <Producer>Producer: Waqas Hasan</Producer>
  <SongWriter>SongWriter: Asim Azhar, Kunaal Verma</SongWriter>
  <Director>Director: Yasir Jaswal</Director>
  <ReleaseYear>ReleaseYear: 9 January 2016</ReleaseYear>
  <Length songLength="long">Length: 04 mins 20 sec</Length>
  <YoutubeLink link_id="10">YoutubeLink:
https://youtu.be/6DtPF9W3ejl</YoutubeLink>
</Song>
<Song id="eleventh_song">
  <bgPic id="eleven"></bgPic>
  <SongName>Name: Dream on</SongName>
    <Genre>Genre: Rock</Genre>
  <Artist type="band">Artist: Aerosmith</Artist>
  <Producer>Producer: Adrian Barbe</Producer>
  <SongWriter>SongWriter: Steven tyler</SongWriter>
  <Director>Director: Marty Callner</Director>
  <ReleaseYear>ReleaseYear: 27 June 1973</ReleaseYear>
  <Length songLength="long">Length: 04 mins 26 sec</Length>
  <YoutubeLink link_id="11">YoutubeLink:
https://youtu.be/sZfZ8uWaOFI</YoutubeLink>
</Song>
<Song id="twelfth_song">
  <bgPic id="twelve"></bgPic>
  <SongName>Name: Call me maybe</SongName>
    <Genre>Genre: Pop</Genre>
  <Artist type="solo">Artist: Carly Rae Jepsen</Artist>
  <Producer>Producer: Josh Ramsay</Producer>

```

```

    <SongWriter>SongWriter: Carly Rae Jepsen, Josh Ramsay, Tavish
Crowe</SongWriter>
    <ReleaseYear>ReleaseYear: 20 September 2011</ReleaseYear>
    <Length songLength="average">Length: 03 mins 13 sec</Length>
    <Album>Album: Kiss and Curiosity</Album>
    <YoutubeLink link_id="12">YoutubeLink:
https://www.youtube.com/watch?v=fWNaR-rxAic</YoutubeLink>
</Song>
</fourth_row>
<fifth_row>
<Song id="thirteenth_song">
    <bgPic id="thirteen"></bgPic>
    <SongName>Name: Say you Won't Let Go</SongName>
        <Genre>Genre: Pop</Genre>
    <Artist type="solo">Artist: James Arthur</Artist>
    <Producer>Producer: Neil Ormandy, Bradley Spence</Producer>
    <SongWriter>SongWriter: James Arthur, Andrew Frampton, Daniel
O'Donoghue, Steve Kipner, Mark Sheehan, Neil Ormandy, Steve
Solomon</SongWriter>
    <Director>Director: Conor McDonnell</Director>
    <ReleaseYear>ReleaseYear: 9 September 2016</ReleaseYear>
    <Length songLength="average">Length: 03 mins 31 sec</Length>
    <Album>Album: Back from the edge</Album>
    <YoutubeLink link_id="13">YoutubeLink:
https://youtu.be/0yW7w8F2TVA</YoutubeLink>
</Song>
<Song id="fourteenth_sonngth">
    <bgPic id="fourteen"></bgPic>
    <SongName>Name: Kashmir</SongName>
        <Genre>Genre: Rock</Genre>
    <Artist type="band">Artist: Led Zepplin</Artist>
    <Producer>Producer: Jimmy Page</Producer>
    <SongWriter>SongWriter: John Bonham, Jimmy Page, Robert
Plant</SongWriter>
    <ReleaseYear>ReleaseYear: 24 February 1975</ReleaseYear>
    <Length songLength="veryLong">Length: 09 mins 41 sec</Length>
    <Album>Album: Physical Graffiti</Album>
    <YoutubeLink link_id="14">YoutubeLink:
https://youtu.be/HNERxbBtT5Q</YoutubeLink>
</Song>
<Song id="fifteenth_song">
    <bgPic id="fifteen"></bgPic>
    <SongName>Name: Stay</SongName>
        <Genre>Genre: Kpop</Genre>
    <Artist type="band">Artist: BlackPink</Artist>
    <Producer>Producer: Teddy Park</Producer>

```

```
<SongWriter>SongWriter: Teddy Park</SongWriter>
<Director>Director: Han Sa Min</Director>
<ReleaseYear>ReleaseYear: 1 November 2016</ReleaseYear>
<Length songLength="average">Length: 03 mins 50 sec</Length>
<Album>Album: Square Two</Album>
<YoutubeLink link_id="15">YoutubeLink:
https://youtu.be/FzVR_fymZw4</YoutubeLink>
</Song>
</fifth_row>
</Shelf>
<footer>
  <Copyright>© Copyright 2021 All Rights Reserved.</Copyright>
</footer>
</musicStore>
```

3.Schema Content

The Schema Content of my coursework is:

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="musicStore">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="header">
          <xs:complexType>
            <xs:sequence>
              <xs:element type="xs:string" name="logo"/>
              <xs:element type="xs:string" name="Name"/>
              <xs:element type="xs:string" name="Address"/>
              <xs:element type="xs:int" name="Telephone_No"/>
              <xs:element type="xs:anyURI" name="Website"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="Shelf">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="first_row">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="Song" maxOccurs="unbounded" minOccurs="0">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element name="bgPic">
                            <xs:complexType>
                              <xs:simpleContent>
                                <xs:extension base="xs:string">
                                  <xs:attribute type="xs:string" name="id" use="optional"/>
                                </xs:extension>
                              </xs:simpleContent>
                            </xs:complexType>
                          </xs:element>
                          <xs:element type="xs:string" name="SongName"/>
                          <xs:element type="xs:string" name="Genre"/>
                          <xs:element name="Artist">
                            <xs:complexType>
                              <xs:simpleContent>
                                <xs:extension base="xs:string">
                                  <xs:attribute type="xs:string" name="type" use="optional"/>
                                </xs:extension>
                              </xs:simpleContent>
                            </xs:complexType>
                          </xs:element>
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```

    </xs:complexType>
  </xs:element>
  <xs:element type="xs:string" name="Producer"/>
  <xs:element type="xs:string" name="SongWriter"/>
  <xs:element type="xs:string" name="Director" minOccurs="0"/>
  <xs:element type="xs:string" name="ReleaseYear"/>
  <xs:element name="Length">
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base="xs:string">
          <xs:attribute type="xs:string" name="songLength" use="optional"/>
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>
  <xs:element type="xs:string" name="Album" minOccurs="0"/>
  <xs:element name="YoutubeLink">
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base="xs:string">
          <xs:attribute type="xs:byte" name="link_id" use="optional"/>
          <xs:attribute type="xs:byte" name="link" use="optional"/>
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>
</xs:sequence>
<xs:attribute type="xs:string" name="id" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="second_row">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Song" maxOccurs="unbounded" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="bgPic">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="id" use="optional"/>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        </xs:complexType>
    </xs:element>
    <xs:element type="xs:string" name="SongName"/>
    <xs:element type="xs:string" name="Genre"/>
    <xs:element name="Artist">
        <xs:complexType>
            <xs:simpleContent>
                <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="type" use="optional"/>
                </xs:extension>
            </xs:simpleContent>
        </xs:complexType>
    </xs:element>
    <xs:element type="xs:string" name="Producer"/>
    <xs:element type="xs:string" name="SongWriter"/>
    <xs:element type="xs:string" name="Director" minOccurs="0"/>
    <xs:element type="xs:string" name="ReleaseYear"/>
    <xs:element name="Length">
        <xs:complexType>
            <xs:simpleContent>
                <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="songLength" use="optional"/>
                </xs:extension>
            </xs:simpleContent>
        </xs:complexType>
    </xs:element>
    <xs:element type="xs:string" name="Album"/>
    <xs:element name="YoutubeLink">
        <xs:complexType>
            <xs:simpleContent>
                <xs:extension base="xs:string">
                    <xs:attribute type="xs:byte" name="link_id" use="optional"/>
                </xs:extension>
            </xs:simpleContent>
        </xs:complexType>
    </xs:element>
</xs:sequence>
<xs:attribute type="xs:string" name="id" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="third_row">
    <xs:complexType>
        <xs:sequence>

```

```
<xs:element name="Song" maxOccurs="unbounded" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="bgPic">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute type="xs:string" name="id" use="optional"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
      <xs:element type="xs:string" name="SongName"/>
      <xs:element type="xs:string" name="Genre"/>
      <xs:element name="Artist">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute type="xs:string" name="type" use="optional"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
      <xs:element type="xs:string" name="Producer"/>
      <xs:element type="xs:string" name="SongWriter"/>
      <xs:element type="xs:string" name="Director" minOccurs="0"/>
      <xs:element type="xs:string" name="ReleaseYear"/>
      <xs:element name="Length">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute type="xs:string" name="songLength" use="optional"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
      <xs:element type="xs:string" name="Album"/>
      <xs:element name="YoutubeLink">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute type="xs:byte" name="link_id" use="optional"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```



```
</xs:sequence>
  <xs:attribute type="xs:string" name="id" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="fourth_row">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Song" maxOccurs="unbounded" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="bgPic">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="id" use="optional"/>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
            <xs:element type="xs:string" name="SongName"/>
            <xs:element type="xs:string" name="Genre"/>
            <xs:element name="Artist">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="type" use="optional"/>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
            <xs:element type="xs:string" name="Producer"/>
            <xs:element type="xs:string" name="SongWriter"/>
            <xs:element type="xs:string" name="Director" minOccurs="0"/>
            <xs:element type="xs:string" name="ReleaseYear"/>
            <xs:element name="Length">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="songLength" use="optional"/>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
```

```
<xs:element type="xs:string" name="Album" minOccurs="0"/>
<xs:element name="YoutubeLink">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute type="xs:byte" name="link_id" use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute type="xs:string" name="id" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="fifth_row">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Song" maxOccurs="unbounded" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="bgPic">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="id" use="optional"/>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
            <xs:element type="xs:string" name="SongName"/>
            <xs:element type="xs:string" name="Genre"/>
            <xs:element name="Artist">
              <xs:complexType>
                <xs:simpleContent>
                  <xs:extension base="xs:string">
                    <xs:attribute type="xs:string" name="type" use="optional"/>
                  </xs:extension>
                </xs:simpleContent>
              </xs:complexType>
            </xs:element>
            <xs:element type="xs:string" name="Producer"/>
            <xs:element type="xs:string" name="SongWriter"/>
            <xs:element type="xs:string" name="Director" minOccurs="0"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element type="xs:string" name="ReleaseYear"/>
<xs:element name="Length">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute type="xs:string" name="songLength" use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element type="xs:string" name="Album"/>
<xs:element name="YoutubeLink">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute type="xs:byte" name="link_id" use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute type="xs:string" name="id" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="footer">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Copyright"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```

4.DTD Content

The DTD content of my coursework is:

```
<?xml encoding="UTF-8"?>
<!ELEMENT musicStore (header,Shelf,footer)>

<!ELEMENT header (logo,Name,Address,Telephone_No,Website)>
<!ELEMENT logo (#PCDATA)>
<!ELEMENT Name (#PCDATA)>
<!ELEMENT Address (#PCDATA) >
<!ELEMENT Telephone_No (#PCDATA)>
<!ELEMENT Website (#PCDATA)>

<!ELEMENT Shelf (first_row, second_row, third_row, fourth_row, fifth_row)+>
<!ELEMENT first_row (Song+)>
<!ELEMENT second_row (Song+)>
<!ELEMENT third_row (Song+)>
<!ELEMENT fourth_row (Song+)>
<!ELEMENT fifth_row (Song+)>

<!ELEMENT Song (bgPic,SongName,Genre,Artist,Producer?,SongWriter?,Director?,
    ReleaseYear?,Length,Album?,YoutubeLink)>
<!ATTLIST Song id CDATA #REQUIRED>
<!ELEMENT bgPic (#PCDATA)>
<!ATTLIST bgPic id CDATA #REQUIRED>
<!ELEMENT SongName (#PCDATA)>
<!ELEMENT Genre (#PCDATA)>
<!ELEMENT Artist (#PCDATA)>
<!ATTLIST Artist type CDATA #REQUIRED>
<!ELEMENT Producer (#PCDATA)>
<!ELEMENT SongWriter (#PCDATA)>
<!ELEMENT Director (#PCDATA)>
<!ELEMENT ReleaseYear (#PCDATA)>
<!ELEMENT Length (#PCDATA)>
<!ATTLIST Length songLength CDATA #REQUIRED>
<!ELEMENT Album (#PCDATA)>
<!ELEMENT YoutubeLink (#PCDATA)>
<!ATTLIST YoutubeLink link_id CDATA #REQUIRED>

<!ELEMENT footer (Copyright)>
<!ELEMENT Copyright (#PCDATA)>
```

5.Testing

5.1 Test 1: Validating XML file

Test	1
Action	Validate XML file online
Expected Result	No error should appear
Actual Result	No error appeared
Conclusion	Test was successful

Table 1: Test 1

Please copy your XML document in here:

```

<ReleaseYear>ReleaseYear: 1 November 2016</ReleaseYear>
  <Length songLength="average">Length: 03 mins 50 sec</Length>
  <Album>Album: Square Two</Album>
  <YoutubeLink link_id="15">YoutubeLink: https://youtu.be/FzVR_fymZw4</YoutubeLink>
</Song>
</fifth_row>
</Shelf>
<footer>
  <Copyright>© Copyright 2021 All Rights Reserved.</Copyright>
</footer>
</musicStore>

```

Or upload it:

No file chosen

The validation check is performed against any XML schema or DTD declared inside the XML document. If neither an XML schema nor a DTD is declared, only a syntax check is performed.

Figure 2: Validating XML file

No errors were found

The following files have been uploaded so far:

[XML document:](#) 

Click on any file name if you want to edit the file.

Figure 3: Result of XML validation

5.2 Test 2: Validating XML file with schema file

Test	2
Action	Validate XML with schema file online
Expected Result	No error should appear
Actual Result	No error appeared
Conclusion	Test was successful

Table 2: Test 2

The file catalog_19033702.xsd is being referenced. Please copy it in here, so that the validation can continue:

```
<xs:element name="root" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Copyright"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```

Or upload it:

No file chosen

Figure 4: Validating Xml with Schema file

No errors were found

The following files have been uploaded so far:

[XML document:](#) 

[catalog_19033702.xsd](#) 

Click on any file name if you want to edit the file.

Figure 5: Result of Validating XML with schema file

5.3 Test 3: Validating XML file with DTD file

Test	3
Action	Validate XML and DTD file online
Expected Result	No error should appear
Actual Result	No error appeared
Conclusion	Test was successful

Table 3: Test 3

The file catalog_19033702.dtd is being referenced. Please copy it in here, so that the validation can continue:

```
<!ELEMENT Songwriter (#PCDATA)>
<!ELEMENT Director (#PCDATA)>
<!ELEMENT ReleaseYear (#PCDATA)>
<!ELEMENT Length (#PCDATA)>
<!ATTLIST Length songLength CDATA #REQUIRED>
<!ELEMENT Album (#PCDATA)>
<!ELEMENT YoutubeLink (#PCDATA)>
<!ATTLIST YoutubeLink link_id CDATA #REQUIRED>

<!ELEMENT footer (Copyright)>
<!ELEMENT Copyright (#PCDATA)>
```

Or upload it:

Choose File No file chosen

continue validation

Figure 6: Validating XML with dtd file

No errors were found

The following files have been uploaded so far:

XML document: 

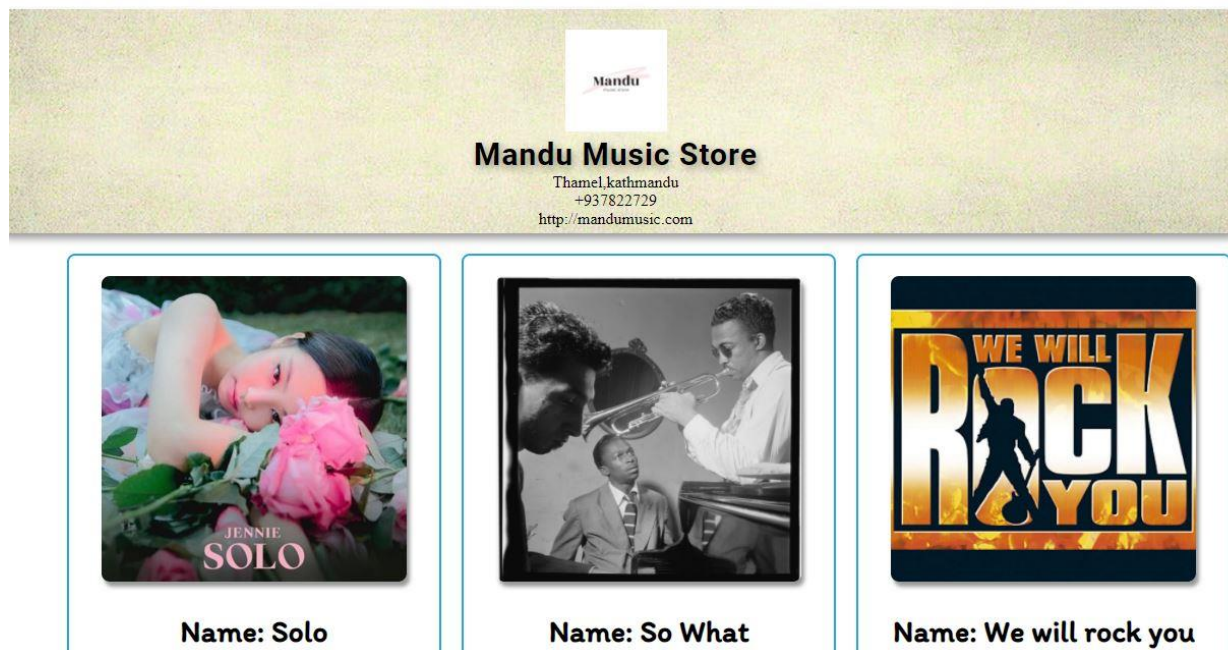
[catalog_19033702.dtd](#) 

Click on any file name if you want to edit the file.

Figure 7: Result of validating XML with dtd file

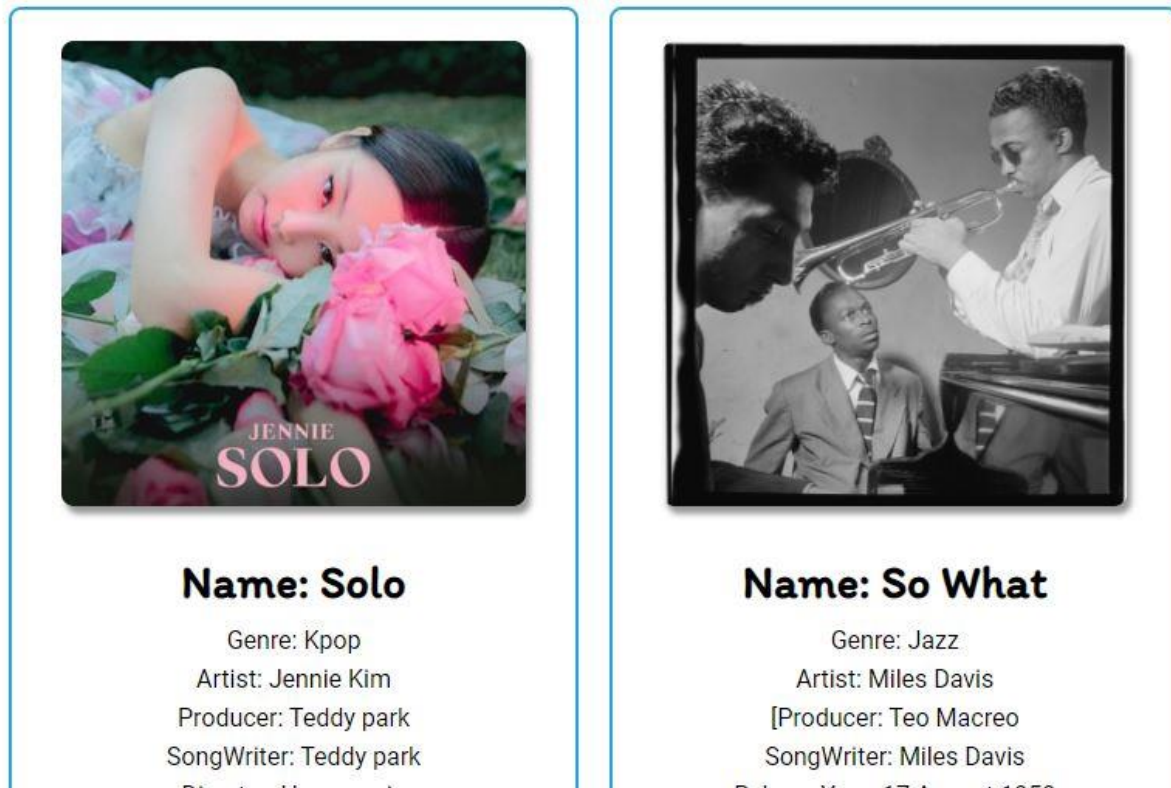
5.4 Test 4: Checking if CSS file is rendered by browser

Test	4
Action	Opening XML file in web browser
Expected Result	Design should be displayed
Actual Result	Design has been displayed
Conclusion	Test was successful

Table 4: Test 4*Figure 8: Result after rendering CSS by browser*

5.5 Test 5: Checking hover in images

Test	5
Action	Mouse is pointed in image
Expected Result	Image should be Zoomed
Actual Result	Zoomed image
Conclusion	Test was successful

Table 5: Test 5*Figure 9: Before hovering*

<http://mandumusic.com>




		
Name: Solo Genre: Kpop Artist: Jennie Kim Producer: Teddy park SongWriter: Teddy park	Name: So What Genre: Jazz Artist: Miles Davis Producer: Teo Macreo SongWriter: Miles Davis	N

Figure 10:After hovering

6. Development of Coursework

In this coursework, I have used Notepad++, Draw.io and Visual Studio code. For the coding part, I used Notepad++ and draw.io for drawing tree diagram. Also, for CSS I used Visual Studio code.

a) Notepad++

Notepad++ is developed by Dan Ho and was released on 23 November 2003. It is a text editor that provides additional features which are not found in the notepad. It is designed for editing source code and is very useful for making important changes in a text file. The ++ in the name signifies that it is used in various programming languages like C, C++, Java and JavaScript (Computer Hope, 6 February 2020).



Figure 11: Logo of Notepad++ (Notepad++, 2021)

b) Draw.io

Draw.io is a software used for making diagrams and charts. It was designed by Seibert Media. This software allows the user to choose or create a custom layout. It has a big option of shapes and elements which helps to make the drawing according to the user's need. The exporting of drawing is also very easy in draw.io as it provides options for saving it in the cloud, device or external media (Computer Hope, 6 February 2020).



Figure 12: Logo of Draw.io (draw.io, 2021)

C)Visual Studio Code

Visual Studio code is a source code editor which has a high ecosystem of extensions for languages like Java, Python, C++. It comes with built-in support for JavaScript, Node.js and is compatible with Windows, mac OS and Linux (Visual Studio Code, 2021).

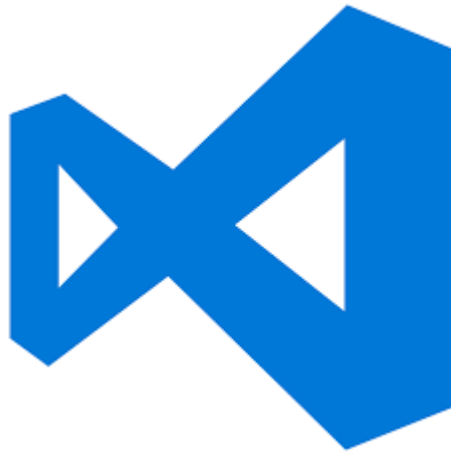


Figure 13:Logo of Visual Studio Code (Visual Studio Code-icon, 2021).

7. Difference between Schema and DTD

There are many differences between Schema and DTD which are mentioned below:

Schema	DTD
1. It defines elements and attributes of a XML file	It defines any SGML document
2. It checks the correctness of the structure of XML file	It checks the grammar and validity of a XML file
3. Namespace is supported in schema	Namespace is not supported in DTD
4. Datatypes are defined here	Datatypes are not defined here
5. Schema can be used for XML only	DTD can be used for any SGML like GML, HTML and XML
6. Schema can control the file's structure	It does not have authority over file's structure
7. Schemas are protractible	DTDs are not protractible

(Singh, 07 July 2020)

8.Critical Evaluation

In this coursework, a system has to be designed for a music store using XML. Though XML was a new language for us as we haven't read it before in any module, it was easy to understand because of its little resemblance with HTML. We were required to use XML with CSS for designing purpose. A tree diagram was also required to draw as drawing a tree diagram for XML was new for me, I had to go through some research and samples.

I started my work with XML by creating an XML file after that I created xsd and dtd file. XSD and DTD were used to check the structure of the file and validate it. I had some problem with my xsd and dtd files as they were new and kind of complex for me at the beginning. There were many small errors in my files and I had to tackle with them one by one. After going through different pages on internet and consulting with tutors I was finally able to create proper xsd and dtd files. After that I started doing my designing work using CSS. In CSS part, I had to use my creativity power the most. I had to insert logo of the music store which was not a quite easy job for me. I had to visit different websites so that I can get the appropriate logo and insert it using CSS. The CSS part was quite tough as I had to determine everything like inserting images, selecting proper font sizes and colours, assigning proper positions of different elements. At last, with proper mindset and consulting with tutors I was able to finish my work.

9. Conclusion

This main purpose of this coursework was to make us familiar with XML and CSS. The page was designed for Mandu music store which contains its details like telephone, address, website address and description of different songs of different genres. As the topic was new, I had to visit different pages on internet in order to get the concept. For the designing purpose, I did researches on the internet.

While doing the coursework, I had to tackle with many problems as I wasn't much familiar with XML and CSS. Due to the small errors in my files, the system was affected and I was getting nervous about the time management. To solve all those problems, I had to do researches and also, I consulted with my tutors and classmates.

With the completion of this coursework, I have gained confidence regarding XML, CSS and my research skills has also improved.

10.Reference

Computer Hope, 6 February 2020. *What is Draw.io_.html*. [Online]
Available at: <https://www.computerhope.com/jargon/d/drawio.htm>
[Accessed 1 5 2021].

draw.io, 2021. *draw.io – Diagrams for Confluence and Jira - draw.io.html*. [Online]
Available at: <https://drawio-app.com/>
[Accessed 1 5 2021].

GeeksforGeeks, 6 November 2020. *Document Type Definition - DTD - GeeksforGeeks.html*. [Online]
Available at: <https://www.geeksforgeeks.org/document-type-definition-dtd/>
[Accessed 01 05 2021].

Notepad++, 2021. *Notepad++ – Logos Download.html*. [Online]
Available at: <https://logos-download.com/41946-notepad-logo-download.html>
[Accessed 1 5 2021].

Singh, Y. P., 07 July 2020. *Difference between Document Type Definition (DTD) and XML Schema Definition (XSD) - GeeksforGeeks.html*. [Online]
Available at: <https://www.geeksforgeeks.org/difference-between-document-type-definition-dtd-and-xml-schema-definition-xsd/>
[Accessed 03 05 2021].

TechTerms, 2021. *CSS (Cascading Style Sheet) Definition.html*. [Online]
Available at: <https://techterms.com/definition/css>
[Accessed 1 5 2021].

TechTerms, 2021. *XML (Extensible Markup Language) Definition.html*. [Online]
Available at: <https://techterms.com/definition/xml>
[Accessed 01 05 2021].

Visual Studio Code, 2021. *Documentation for Visual Studio Code.html*. [Online]
Available at: <https://code.visualstudio.com/docs>
[Accessed 4 5 2021].

Visual Studio Code-icon, 2021. *Visual Studio Code-icon _ Brands VA - VZ*. [Online]
Available at: <https://vecta.io/symbols/100/brands-va-vz/49/visual-studio-code-icon>
[Accessed 1 5 2021].