Module Code & Module Title

CS5004NI Emerging Programming Platforms and Technologies

Assessment Weightage & Type

30% Individual Coursework

Year and Semester

2020-21 Autumn

Student Name: Tenzing Tsering Lama

London Met ID: 20049082

College ID: NP01CP4S210057

Assignment Due Date: 5 May,2022

Word Count:1519

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

Table of Contents

Table of table

Table of figure

1.Introduction

This assignment of module Emerging programming platforms and technologies accounts for 30% of the module's credit. The assignment was to use XML, XML Schema, and CSS to create a static website for a gift shop. During the coursework, all the scenario's minimum prerequisites must have been met.

XML is a text-based mark-up language that is a derivative of SGML (Standard Generalized Markup Language). In XML, there are no pre-defined tags. Unlike HTML, which has predefined tags, these tags are used to store and manage information. It may be used in a variety of ways because it comprises self-descriptive tags. XML is a public standard developed by the World Wide Web Consortium (W3C) and is available as an open standard.

The XML Schema Definition (XSD) is a way of describing the structure of an XML document. It specifies the rules for all an XML document's attributes and elements. It examines the document's vocabulary. structure of the XML file is proper. A schema specifies the rules that a valid XML document must follow to ensure that the data it contains is in the correct format.

Figure 1 XML image

2.XML Document

2.1 List of data

2.1.1 List of Elements

List of Elements	Element Type	Element Data	Occurrence
GaloreGifts	Complex Type	Child Elements	Only Once
Home	Complex Type	Child Elements	Only Once
GiftCards	Complex Type	Child Elements	Only Once
Store	Complex Type	Child Elements	Only Once
Nav	Complex Type	Child Elements	Only Once
Body	Complex Type	Child Elements	Only Once
Galore_Logo	Simple Type	-	Only Once
Nav_Menu	Complex Type	Child Elements	Only Once
ShopBtn	Simple Type	String	Only Once
Shop	Simple Type	String	Only Once
Pricing	Simple Type	String	Only Once
Blog	Simple Type	String	Only Once
Contact	Simple Type	String	Only Once
Description	Complex Type	Child Elements	Only Once
Home_Photo	Simple Type	-	Only Once
Info	Simple Type	String	Only Once
Title	Simple Type	String	Only Once
Sub_Title	Simple Type	String	Only Once
CallToActionBtn	Simple Type	String	Only Once
GiftCards_G1	Complex Type	Child Elements	Only Once
GiftCards_G2	Complex Type	Child Elements	Only Once
GiftCard	Complex Type	Child Elements	One or More
Logo	Simple Type	-	Only Once
GiftCard_Details	Complex Type	Child Elements	One or More

Product_Name	Simple Type	String	Only Once
Label	Simple Type	String	Only Once
Company	Complex Type	Child Elements	Only Once
Company_Name	Simple Type	String	Only Once
Type	Complex Type	Child Elements	Only Once
Type_Name	Simple Type	String	Only Once
Validity	Complex Type	Child Elements	Only Once
Validity_Date	Simple Type	Date	Only Once
Cost	Complex Type	Child Elements	Only Once
Total	Simple Type	Float	Only Once
Card	Complex Type	Child Elements	Only Once
Card_Number	Simple Type	String	Only Once
Number_Of_Users	Complex Type	Child Elements	Only Once
Users	Simple Type	Integer	Only Once
Redeem_Group	Complex Type	Child Elements	Only Once
Redeem	Simple Type	String	Only Once
Discount_Group	Complex Type	Child Elements	Zero or one
Discount	Simple Type	Float	Zero or one
Cashback_Group	Complex Type	Child Elements	Zero or one
CashBack	Simple Type	String	Zero or one
Personalized_Group	Complex Type	Child Elements	Zero or one
Personalized	Simple Type	String	Zero or one
Delivery_Group	Complex Type	Child Elements	Zero or one
Delivery	Simple Type	String	Zero or one
Returm_Group	Complex Type	Child Elements	Zero or one
Return	Simple Type	String	Zero or one
Guide	Complex Type	Child Elements	Only Once
Guideline	Simple Type	String	Only Once
Terms_Condition	Simple Type	String	Only Once
Store_Details	Complex Type	Child Elements	Only Once

AboutImg	Simple Type	-	Only Once
Store_Logo	Simple Type	-	Only Once
Name	Simple Type	String	Only Once
Phone_Group	Complex Type	Child Elements	Only Once
Phone_Logo	Simple Type	-	Only Once
Phone_Number	Simple Type	Short	Only Once
Email_Group	Complex Type	Child Elements	Only Once
Email_Logo	Simple Type	-	Only Once
Email	Simple Type	String	Only Once
Address_Group	Complex Type	Child Elements	Only Once
Address_Logo	Simple Type	-	Only Once
Address	Simple Type	String	Only Once
Website_Group	Complex Type	Child Elements	Only Once
Website_Logo	Simple Type	-	Only Once
Website	Simple Type	String	Only Once

2.1.2 List of Attributes

List of Attributes	Data Type	Default	Fixed	Use
slogan	string	-	“Gifts for all”	Required
founded	date	-	“2021-09-24”	Required
founder	string	-	“Jack Donalds”	Required
id	id	-	-	Required
type	String	“Digital”	-	Required
sector	string	“Technology	-	Required
reload	String	“Non-Reloadable	-	Optional
country	String	“USA”	-	Required
currency	String	“USD”	-	Required
phoneType	String	“Landline”	-	Optional
street	string	-	“Oxford Street”	Required

2.2 Tree Diagram

Element GaloreGifts and its child elements and attributes.

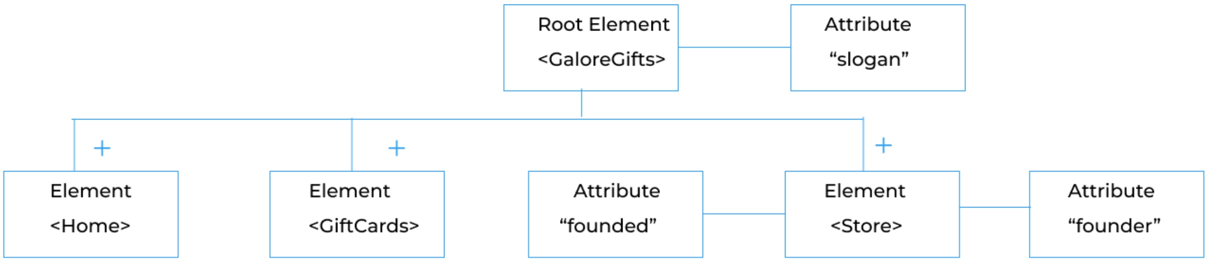


Figure 2 Galore Gifts tree diagram

Element Home and its child elements

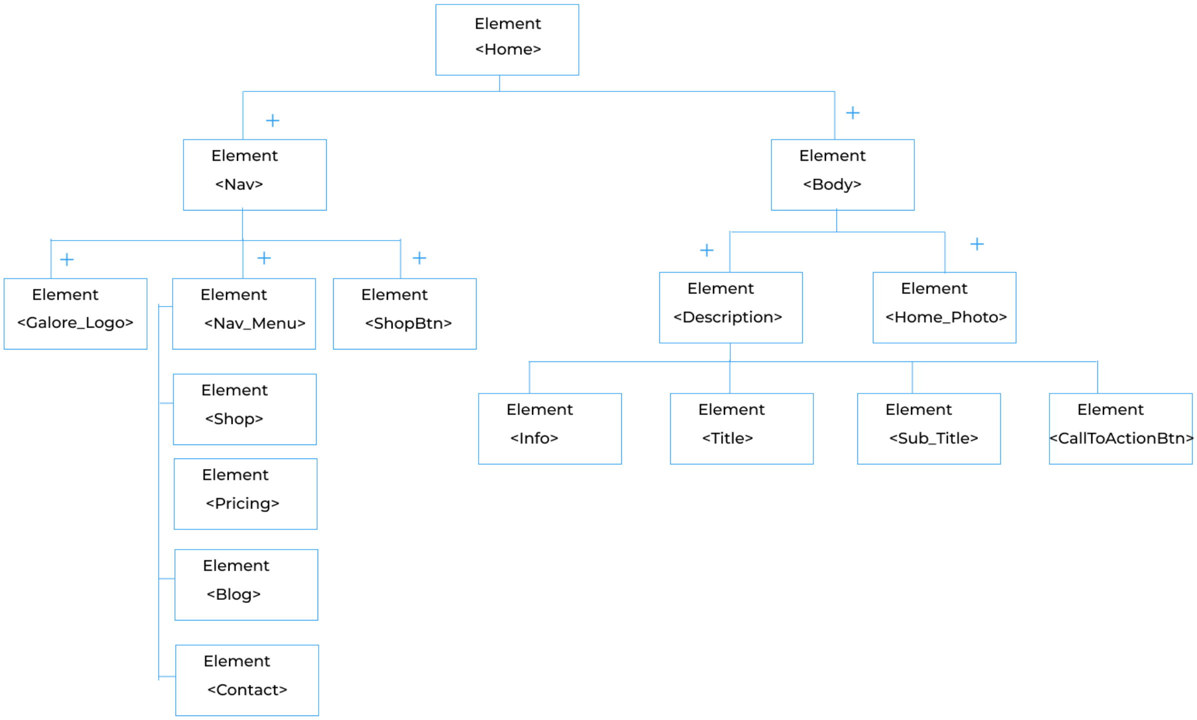


Figure 3 Home tree diagram

Element GiftCards with child elements and attributes

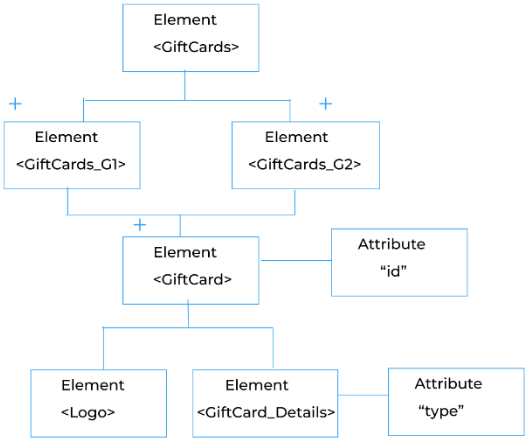


Figure 4 Gift Cards tree diagram

Element GiftCard_Details with child elements and attributes

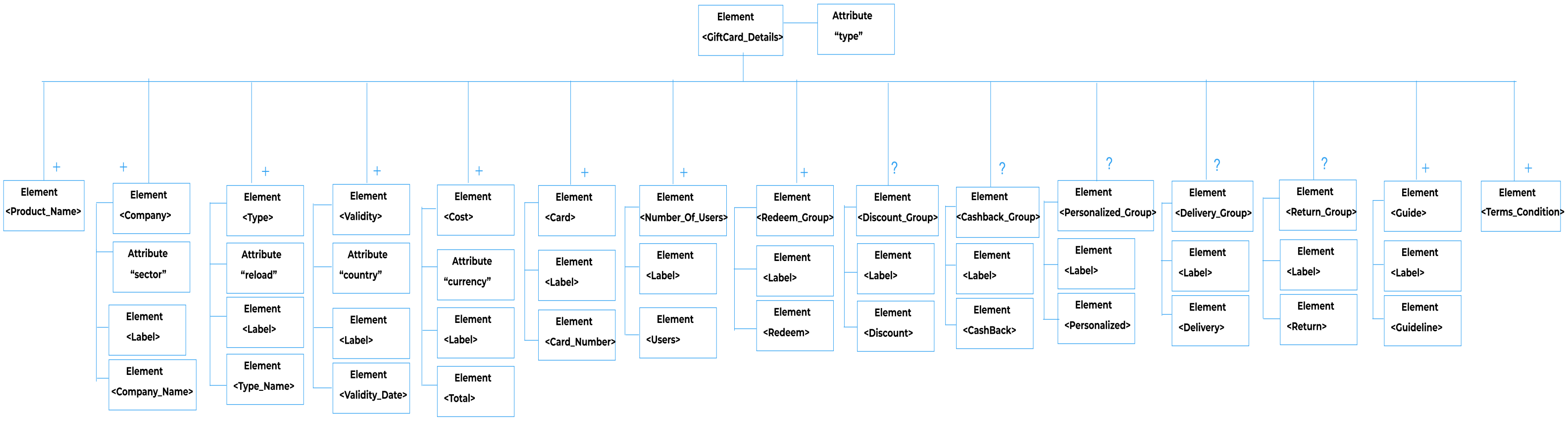


Figure 5 GiftCard_Details tree diagram

Element Store with child elements and attributes

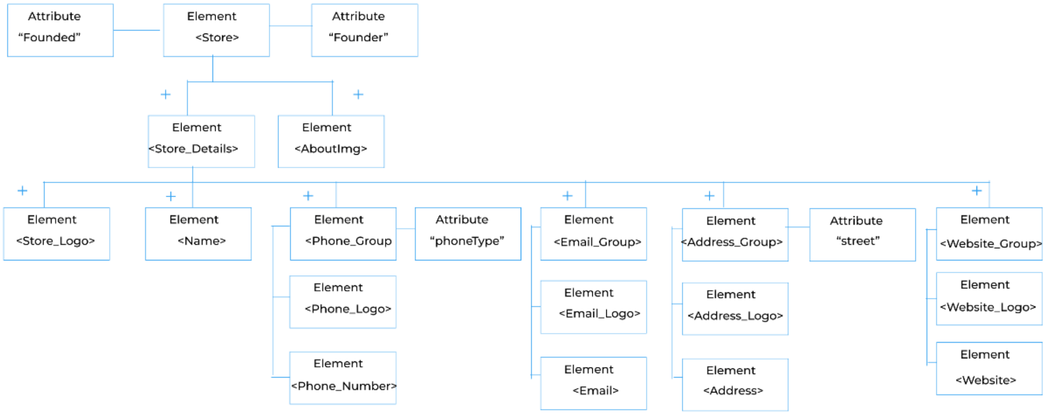
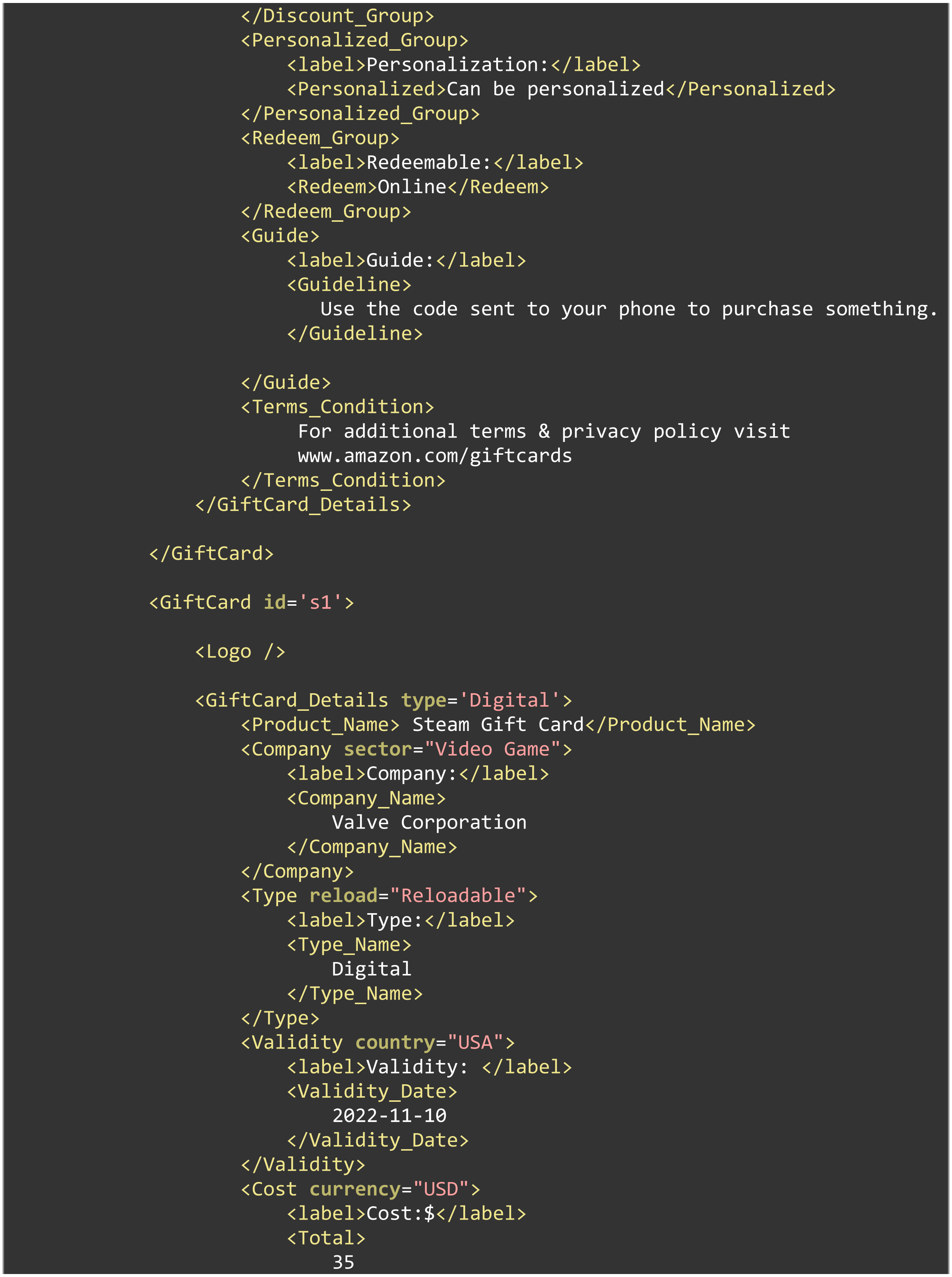


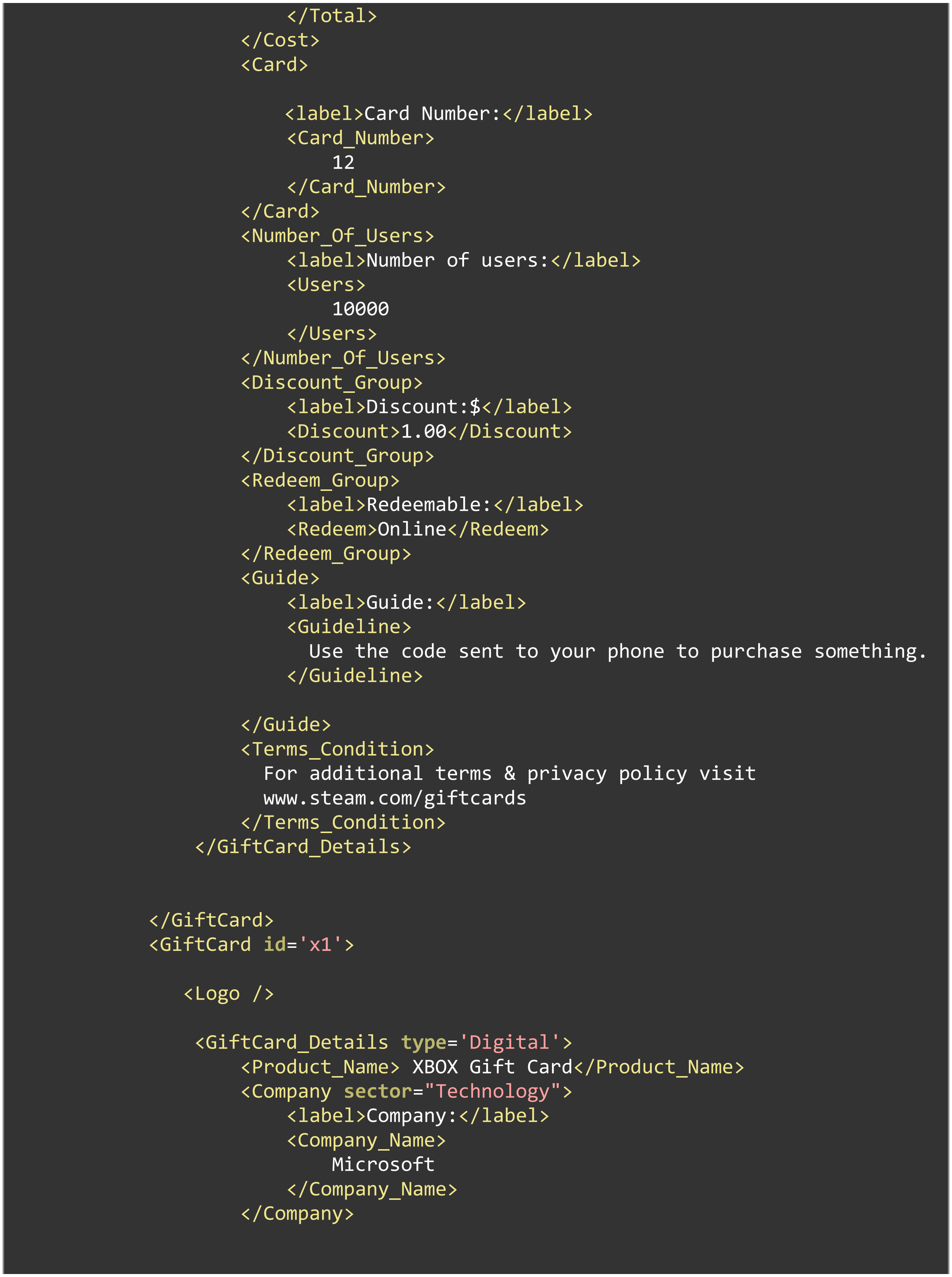
Figure 6 Store tree diagram

2.3 XML Code

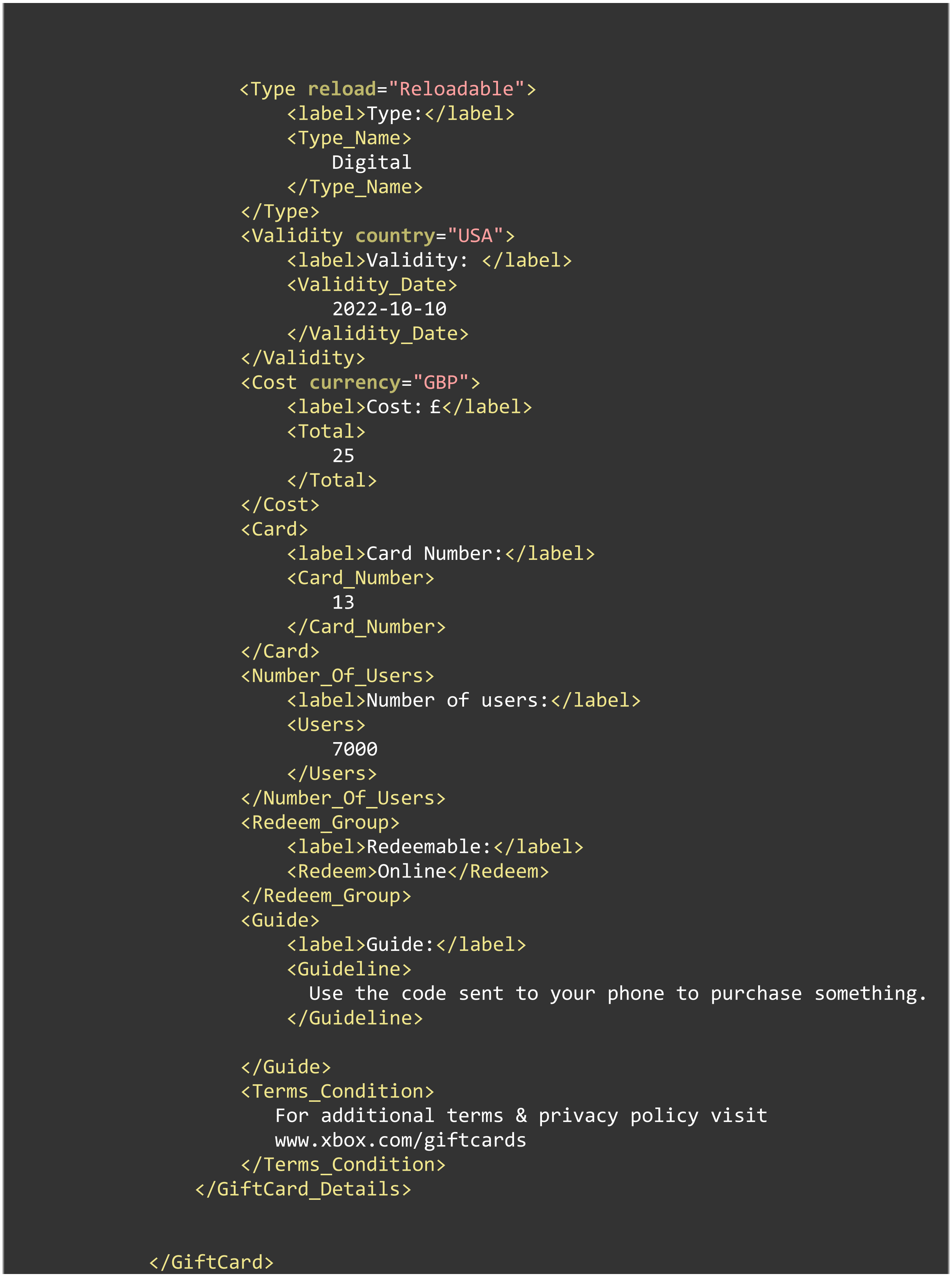
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<!-- css link -->
<?xml-stylesheet type="text/css" href="catalog_20049082.css"?>
<!-- schema link -->
<!-- <GaloreGifts xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="catalog_20049082.xsd"> -->
<GaloreGifts slogan="Gifts for all" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
xsi:noNamespaceSchemaLocation="catalog_20049082.xsd">


    <Home>
        <Nav>

            <Galore_Logo />
            <Nav_Menu>
                <Shop>Shop</Shop>
                <Pricing>Pricing</Pricing>
                <Blog>Blog</Blog>
                <Contact>Contact</Contact>
            </Nav_Menu>
            <ShopBtn>
                Shop now
            </ShopBtn>

        </Nav>
        <Body>
            <Description>
                <Title>Galore Gifts</Title>
                <Sub_Title>GIFTS FOR ALL</Sub_Title>
                <Info>
                   We sell and deliver digital & physical gift cards to your
doorstep. Make sure to remember us.
                </Info>

                <CallToActionBtn>
                    Get your giftcard
                </CallToActionBtn>


            </Description>

            <Photo />
        </Body>

    </Home>

    <GiftCards>
        <GiftCards_G1>
            <GiftCard id='a1'>

                <Logo />

                <GiftCard_Details type='Digital'>
                    <Product_Name> Amazon Gift Card</Product_Name>
                    <Company sector="Ecommerce">
                        <label>Company:</label>
                        <Company_Name>
                            Amazon
                        </Company_Name>
                    </Company>
                    <Type reload="Non-Reloadable">
                        <label>Type:</label>
                        <Type_Name>
                            Digital
                        </Type_Name>
                    </Type>
                    <Validity country="USA">
                        <label>Validity: </label>
                        <Validity_Date>
                            2022-10-10
                        </Validity_Date>
                    </Validity>
                    <Cost currency="USD">
                        <label>Cost:$</label>
                        <Total>
                            45
                        </Total>
                    </Cost>
                    <Card>
                        <label>Card Number:</label>
                        <Card_Number>
                            10
                        </Card_Number>
                    </Card>
                    <Number_Of_Users>
                        <label>Number of users:</label>
                        <Users>
                            200000
                        </Users>
                    </Number_Of_Users>
                    <Discount_Group>
                        <label>Discount:$</label>
                        <Discount>5.00 </Discount>
```



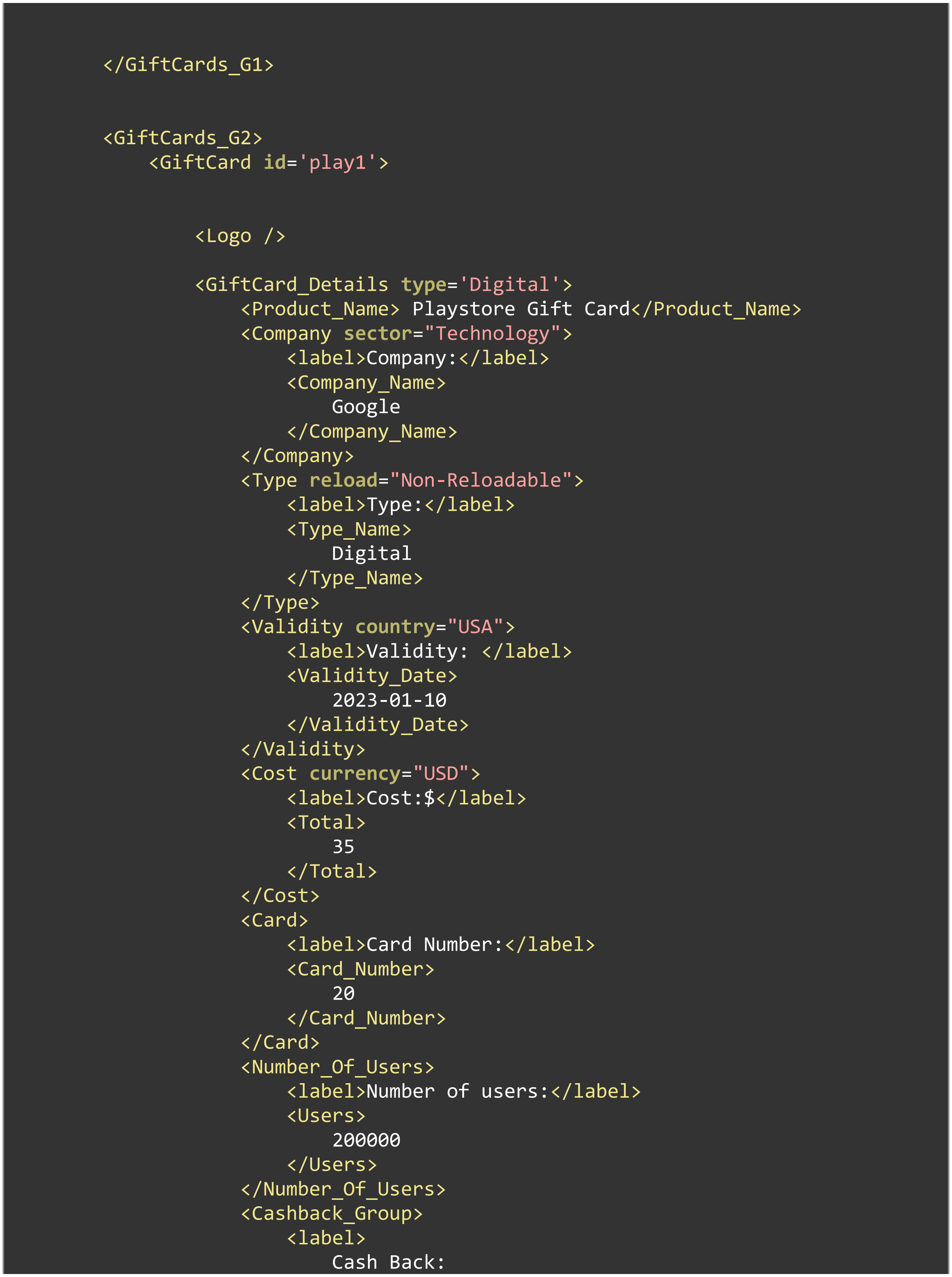
```
                </Discount_Group>
                <Personalized_Group>
                    <label>Personalization:</label>
                    <Personalized>Can be personalized</Personalized>
                </Personalized_Group>
                <Redeem_Group>
                    <label>Redeemable:</label>
                    <Redeem>Online</Redeem>
                </Redeem_Group>
                <Guide>
                    <label>Guide:</label>
                    <Guideline>
                       Use the code sent to your phone to purchase something.
                    </Guideline>

                </Guide>
                <Terms_Condition>
                     For additional terms & privacy policy visit
                     www.amazon.com/giftcards
                </Terms_Condition>
            </GiftCard_Details>

        </GiftCard>

        <GiftCard id='s1'>

            <Logo />

            <GiftCard_Details type='Digital'>
                <Product_Name> Steam Gift Card</Product_Name>
                <Company sector="Video Game">
                    <label>Company:</label>
                    <Company_Name>
                        Valve Corporation
                    </Company_Name>
                </Company>
                <Type reload="Reloadable">
                    <label>Type:</label>
                    <Type_Name>
                        Digital
                    </Type_Name>
                </Type>
                <Validity country="USA">
                    <label>Validity: </label>
                    <Validity_Date>
                        2022-11-10
                    </Validity_Date>
                </Validity>
                <Cost currency="USD">
                    <label>Cost:$</label>
                    <Total>
                        35
```



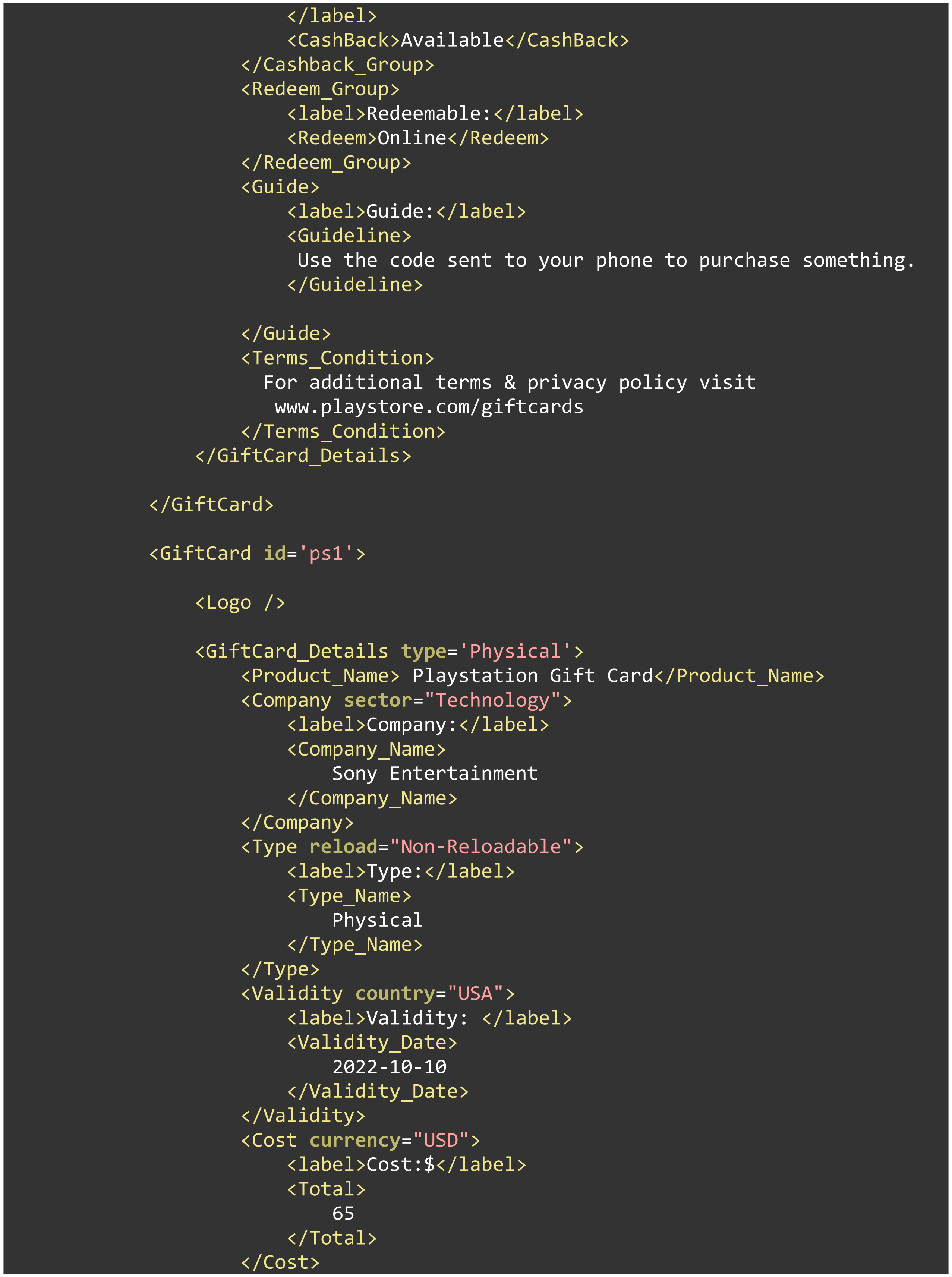
```
            </Total>
        </Cost>
        <Card>

            <label>Card Number:</label>
            <Card_Number>
                12
            </Card_Number>
        </Card>
        <Number_Of_Users>
            <label>Number of users:</label>
            <Users>
                10000
            </Users>
        </Number_Of_Users>
        <Discount_Group>
            <label>Discount:$</label>
            <Discount>1.00</Discount>
        </Discount_Group>
        <Redeem_Group>
            <label>Redeemable:</label>
            <Redeem>Online</Redeem>
        </Redeem_Group>
        <Guide>
            <label>Guide:</label>
            <Guideline>
              Use the code sent to your phone to purchase something.
            </Guideline>

        </Guide>
        <Terms_Condition>
          For additional terms & privacy policy visit
          www.steam.com/giftcards
        </Terms_Condition>
    </GiftCard_Details>


</GiftCard>
<GiftCard id='x1'>

   <Logo />

    <GiftCard_Details type='Digital'>
        <Product_Name> XBOX Gift Card</Product_Name>
        <Company sector="Technology">
            <label>Company:</label>
            <Company_Name>
                Microsoft
            </Company_Name>
        </Company>
```



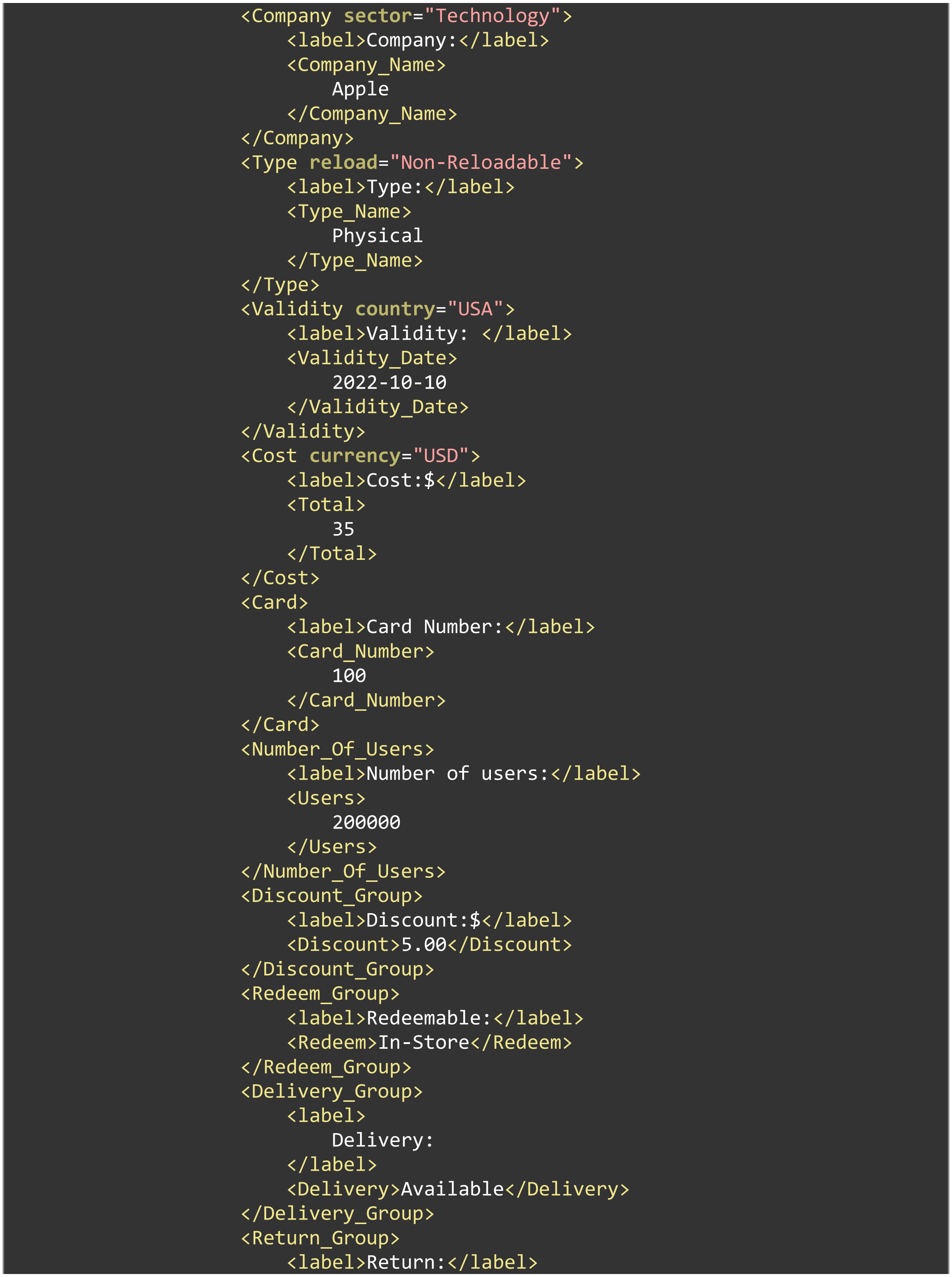
```
        <Type reload="Reloadable">
            <label>Type:</label>
            <Type_Name>
                Digital
            </Type_Name>
        </Type>
        <Validity country="USA">
            <label>Validity: </label>
            <Validity_Date>
                2022-10-10
            </Validity_Date>
        </Validity>
        <Cost currency="GBP">
            <label>Cost: £</label>
            <Total>
                25
            </Total>
        </Cost>
        <Card>
            <label>Card Number:</label>
            <Card_Number>
                13
            </Card_Number>
        </Card>
        <Number_Of_Users>
            <label>Number of users:</label>
            <Users>
                7000
            </Users>
        </Number_Of_Users>
        <Redeem_Group>
            <label>Redeemable:</label>
            <Redeem>Online</Redeem>
        </Redeem_Group>
        <Guide>
            <label>Guide:</label>
            <Guideline>
              Use the code sent to your phone to purchase something.
            </Guideline>

        </Guide>
        <Terms_Condition>
           For additional terms & privacy policy visit
           www.xbox.com/giftcards
        </Terms_Condition>
    </GiftCard_Details>


</GiftCard>
```



```
</GiftCards_G1>


<GiftCards_G2>
    <GiftCard id='play1'>


        <Logo />

        <GiftCard_Details type='Digital'>
            <Product_Name> Playstore Gift Card</Product_Name>
            <Company sector="Technology">
                <label>Company:</label>
                <Company_Name>
                    Google
                </Company_Name>
            </Company>
            <Type reload="Non-Reloadable">
                <label>Type:</label>
                <Type_Name>
                    Digital
                </Type_Name>
            </Type>
            <Validity country="USA">
                <label>Validity: </label>
                <Validity_Date>
                    2023-01-10
                </Validity_Date>
            </Validity>
            <Cost currency="USD">
                <label>Cost:$</label>
                <Total>
                    35
                </Total>
            </Cost>
            <Card>
                <label>Card Number:</label>
                <Card_Number>
                    20
                </Card_Number>
            </Card>
            <Number_Of_Users>
                <label>Number of users:</label>
                <Users>
                    200000
                </Users>
            </Number_Of_Users>
            <Cashback_Group>
                <label>
                    Cash Back:
```



```
                        </label>
                        <CashBack>Available</CashBack>
                    </Cashback_Group>
                    <Redeem_Group>
                        <label>Redeemable:</label>
                        <Redeem>Online</Redeem>
                    </Redeem_Group>
                    <Guide>
                        <label>Guide:</label>
                        <Guideline>
                         Use the code sent to your phone to purchase something.
                        </Guideline>

                    </Guide>
                    <Terms_Condition>
                      For additional terms & privacy policy visit
                       www.playstore.com/giftcards
                    </Terms_Condition>
                </GiftCard_Details>

            </GiftCard>

            <GiftCard id='ps1'>

                <Logo />

                <GiftCard_Details type='Physical'>
                    <Product_Name> Playstation Gift Card</Product_Name>
                    <Company sector="Technology">
                        <label>Company:</label>
                        <Company_Name>
                            Sony Entertainment
                        </Company_Name>
                    </Company>
                    <Type reload="Non-Reloadable">
                        <label>Type:</label>
                        <Type_Name>
                            Physical
                        </Type_Name>
                    </Type>
                    <Validity country="USA">
                        <label>Validity: </label>
                        <Validity_Date>
                            2022-10-10
                        </Validity_Date>
                    </Validity>
                    <Cost currency="USD">
                        <label>Cost:$</label>
                        <Total>
                            65
                        </Total>
                    </Cost>
                    <Card>
                        <label>Card Number:</label>
                        <Card_Number>
                            45
                        </Card_Number>
                    </Card>
                    <Number_Of_Users>
                        <label>Number of users:</label>
                        <Users>
                            9000
                        </Users>
                    </Number_Of_Users>
                    <Discount_Group>
                        <label>Discount:$</label>
                        <Discount>3.00</Discount>
                    </Discount_Group>
                    <Redeem_Group>
                        <label>Redeemable:</label>
                        <Redeem>In-Store</Redeem>
                    </Redeem_Group>
                    <Delivery_Group>
                        <label>
                            Delivery:
                        </label>
                        <Delivery>Not Available</Delivery>
                    </Delivery_Group>
                    <Return_Group>
                        <label>Return:</label>
                        <Return>Within 7 days</Return>
                    </Return_Group>
                    <Guide>
                        <label>Guide:</label>
                        <Guideline>
                            Scratch the label off the card and use the code.
                        </Guideline>

                    </Guide>
                    <Terms_Condition>
                        For additional terms & privacy policy visit
www.playstation.com/giftcards
                    </Terms_Condition>
                </GiftCard_Details>

            </GiftCard>

            <GiftCard id='i1'>


                <Logo />

                <GiftCard_Details type='Physical'>
                    <Product_Name> Itunes Gift Card</Product_Name>
```



```
<Company sector="Technology">
    <label>Company:</label>
    <Company_Name>
        Apple
    </Company_Name>
</Company>
<Type reload="Non-Reloadable">
    <label>Type:</label>
    <Type_Name>
        Physical
    </Type_Name>
</Type>
<Validity country="USA">
    <label>Validity: </label>
    <Validity_Date>
        2022-10-10
    </Validity_Date>
</Validity>
<Cost currency="USD">
    <label>Cost:$</label>
    <Total>
        35
    </Total>
</Cost>
<Card>
    <label>Card Number:</label>
    <Card_Number>
        100
    </Card_Number>
</Card>
<Number_Of_Users>
    <label>Number of users:</label>
    <Users>
        200000
    </Users>
</Number_Of_Users>
<Discount_Group>
    <label>Discount:$</label>
    <Discount>5.00</Discount>
</Discount_Group>
<Redeem_Group>
    <label>Redeemable:</label>
    <Redeem>In-Store</Redeem>
</Redeem_Group>
<Delivery_Group>
    <label>
        Delivery:
    </label>
    <Delivery>Available</Delivery>
</Delivery_Group>
<Return_Group>
    <label>Return:</label>
```

```
                        <Return>Not Available</Return>
                    </Return_Group>
                    <Guide>
                        <label>Guide:</label>
                        <Guideline>
                            Scratch the label off the card and use the code.
                        </Guideline>

                    </Guide>
                    <Terms_Condition>
                        For additional terms & privacy policy visit
www.itunes.com/giftcards
                    </Terms_Condition>
                </GiftCard_Details>

            </GiftCard>

        </GiftCards_G2>


    </GiftCards>


    <Store founder="Jack Donalds" founded="2021-09-24">

        <Store_Details>
            <Store_Logo />

            <Name>Galore Gifts: Gifts for all</Name>


            <Phone_Group>
                <Phone_Logo />
                <Phone_Number phoneType="Landline">99880900</Phone_Number>

            </Phone_Group>

            <Email_Group>
                <Email_Logo />
                <Email>galoregifts@gmail.com</Email>
            </Email_Group>
            <Address_Group>
                <Address_Logo />
                <Address street="Oxford Street">UK,London</Address>

            </Address_Group>

            <Website_Group>
                <Website_Logo />
                <Website>galoregifts.com</Website>

            </Website_Group>
        </Store_Details>

        <AboutImg />

    </Store>

</GaloreGifts>
```

3. Schema Code

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

<!-- schema declaration -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">


    <xs:element name="Contact" type="xs:string" />
    <xs:element name="Blog" type="xs:string" />
    <xs:element name="Pricing" type="xs:string" />
    <xs:element name="Shop" type="xs:string" />


    <xs:element name="ShopBtn" type="xs:string" />
    <xs:element name="Nav_Menu">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Shop" />
                <xs:element ref="Pricing" />
                <xs:element ref="Blog" />
                <xs:element ref="Contact" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="Galore_Logo" type="xs:string" />

    <xs:element name="Nav">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Galore_Logo" />
                <xs:element ref="Nav_Menu" />
                <xs:element ref="ShopBtn" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>


    <xs:element name="CallToActionBtn" type="xs:string" />
    <xs:element name="Info" type="xs:string" />

    <xs:element name="Sub_Title" type="xs:string" />
    <xs:element name="Title" type="xs:string" />

    <xs:element name="Description">
```

```
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Title" />
                <xs:element ref="Sub_Title" />
                <xs:element ref="Info" />
                <xs:element ref="CallToActionBtn" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>


    <xs:element name="Photo" type="xs:string" />


    <xs:element name="Body">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Description" />
                <xs:element ref="Photo" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>


    <xs:element name="Home">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Nav" />
                <xs:element ref="Body" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="label" type="xs:string" />


    <xs:element name="Terms_Condition" type="xs:string" />


    <xs:element name="Guideline" type="xs:string" />

    <xs:element name="Guide">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Guideline" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="Return">
        <xs:simpleType>
```

```
            <xs:restriction base="xs:string">
                <xs:enumeration value="Within 7 days" />
                <xs:enumeration value="Not Available" />
            </xs:restriction>
        </xs:simpleType>

    </xs:element>

    <xs:element name="Return_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label"/>
                <xs:element ref="Return"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="Delivery">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Available" />
                <xs:enumeration value="Not Available" />
            </xs:restriction>
        </xs:simpleType>
    </xs:element>

    <xs:element name="Delivery_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label"/>
                <xs:element ref="Delivery"/>
            </xs:sequence>
        </xs:complexType>

    </xs:element>

    <xs:element name="Personalized">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Can be personalized" />
                <xs:enumeration value="Cannot be personalized" />
            </xs:restriction>
        </xs:simpleType>
    </xs:element>

    <xs:element name="Personalized_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label"/>
                <xs:element ref="Personalized"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="CashBack">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Available" />
                <xs:enumeration value="Not Available" />
            </xs:restriction>
        </xs:simpleType>
    </xs:element>

    <xs:element name="Cashback_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label"/>
                <xs:element ref="CashBack"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="Discount" type="xs:float" />

    <xs:element name="Discount_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Discount" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="Redeem">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Online" />
                <xs:enumeration value="In-Store" />
            </xs:restriction>
        </xs:simpleType>
    </xs:element>

    <xs:element name="Redeem_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Redeem" />
            </xs:sequence>
        </xs:complexType>

    </xs:element>

    <xs:element name="Users" type="xs:int" />
    <xs:element name="Number_Of_Users">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Users" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>


    <xs:element name="Card_Number" type="xs:int"/>

    <xs:element name="Card">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Card_Number"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>


    <xs:attribute name="currency" default="USD">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="USD" />
                <xs:enumeration value="AUD" />
                <xs:enumeration value="GBP" />
                <xs:enumeration value="INR" />
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>

    <xs:element name="Total" type="xs:float" />

    <xs:element name="Cost">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Total" />
            </xs:sequence>
            <xs:attribute type="xs:string" name="currency" use="required" />
        </xs:complexType>
    </xs:element>


    <xs:attribute name="country" default="USA">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="USA" />
                <xs:enumeration value="Australia" />
                <xs:enumeration value="UK" />
                <xs:enumeration value="India" />
                <xs:enumeration value="Worldwide" />

            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>

    <xs:element name="Validity_Date" type="xs:date" />

    <xs:element name="Validity">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Validity_Date" />
            </xs:sequence>
            <xs:attribute ref="country" use="required" />
        </xs:complexType>
    </xs:element>


    <xs:attribute name="reload" default="Non-Reloadable">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Non-Reloadable" />
                <xs:enumeration value="Reloadable" />

            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>


    <xs:element name="Type_Name" type="xs:string" />
    <xs:element name="Type">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Type_Name" />
            </xs:sequence>
            <xs:attribute ref="reload" use="optional" />
        </xs:complexType>
    </xs:element>


    <xs:attribute name="sector">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Technology" />
                <xs:enumeration value="Video Game" />
                <xs:enumeration value="Ecommerce" />
                <xs:enumeration value="Marketing" />
                <xs:enumeration value="Others" />
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>

    <xs:element name="Company_Name" type="xs:string" />
    <xs:element name="Company">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="label" />
                <xs:element ref="Company_Name" />
            </xs:sequence>
            <xs:attribute ref="sector"/>
        </xs:complexType>
    </xs:element>

    <xs:element name="Product_Name" type="xs:string" />


    <xs:attribute name="type" default="Digital">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Physical" />
                <xs:enumeration value="Digital" />
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>


    <xs:element name="GiftCard_Details">
        <xs:complexType>
            <xs:all>
                <xs:element ref="Product_Name" />
                <xs:element ref="Company" />
                <xs:element ref="Type" />
                <xs:element ref="Validity" />
                <xs:element ref="Cost" />
                <xs:element ref="Card" />
                <xs:element ref="Number_Of_Users" />
                <xs:element ref="Redeem_Group"/>
                <xs:element ref="Discount_Group" minOccurs="0" maxOccurs="1"/>
                <xs:element ref="Cashback_Group" minOccurs="0" maxOccurs="1"/>
                <xs:element ref="Personalized_Group" minOccurs="0"
maxOccurs="1"/>
                <xs:element ref="Delivery_Group" minOccurs="0" maxOccurs="1"/>
                <xs:element ref="Return_Group" minOccurs="0" maxOccurs="1"/>
                <xs:element ref="Guide" />
                <xs:element ref="Terms_Condition" />
            </xs:all>
            <xs:attribute ref="type" use="required" />
        </xs:complexType>
    </xs:element>
    <xs:element name="Logo" type="xs:string" />

    <xs:attribute name="id" type="xs:ID"/>

    <xs:element name="GiftCard">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Logo" />
                <xs:element ref="GiftCard_Details" maxOccurs="unbounded"
minOccurs="1"/>
            </xs:sequence>
            <xs:attribute ref="id" use="required" />
        </xs:complexType>
    </xs:element>


    <xs:element name="GiftCards_G1">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="GiftCard" maxOccurs="unbounded" minOccurs="1" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="GiftCards_G2">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="GiftCard" maxOccurs="unbounded" minOccurs="1" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>


    <xs:element name="GiftCards">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="GiftCards_G1" />
                <xs:element ref="GiftCards_G2" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>


    <xs:element name="Website_Logo" type="xs:string" />
    <xs:element name="Website" type="xs:string" />


    <xs:element name="Website_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Website_Logo" />
                <xs:element ref="Website" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>


    <xs:element name="Address_Logo" type="xs:string" />

    <xs:attribute name="street" type="xs:string" fixed="Oxford Street"/>

    <xs:element name="Address">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:string">
              <xs:attribute ref="street" use="required"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>

    </xs:element>

    <xs:element name="Address_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Address_Logo" />
                <xs:element ref="Address" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="Email_Logo" />

    <xs:element name="Email">
        <xs:simpleType>
            <xs:restriction base="xs:token">
                <xs:maxLength value="254"/>
                <xs:pattern value="[_\-a-zA-Z0-9\.\+]+@[a-zA-Z0-9](\.?[\-a-zA-Z0-
9]*[a-zA-Z0-9])*"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:element>

    <xs:element name="Email_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Email_Logo" />
                <xs:element ref="Email" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="Phone_Logo"/>

    <xs:attribute name="phoneType"/>
    <xs:element name="Phone_Number">
        <xs:complexType>
          <xs:simpleContent>
            <xs:extension base="xs:int">
              <xs:attribute ref="phoneType" use="optional"/>
            </xs:extension>
          </xs:simpleContent>
        </xs:complexType>
      </xs:element>

    <xs:element name="Phone_Group">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Phone_Logo" />
                <xs:element ref="Phone_Number" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="Store_Logo" />
    <xs:element name="Name" type="xs:string" />

    <xs:element name="Store_Details">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Store_Logo" />
                <xs:element ref="Name" />
                <xs:element ref="Phone_Group" />
                <xs:element ref="Email_Group" />
                <xs:element ref="Address_Group" />
                <xs:element ref="Website_Group" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>

    <xs:element name="AboutImg" />

    <xs:attribute name="founder" type="xs:string" fixed="Jack Donalds" />
    <xs:attribute name="founded" type="xs:date" fixed="2021-09-24"/>

    <xs:element name="Store">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Store_Details" />
                <xs:element ref="AboutImg" />
            </xs:sequence>
            <xs:attribute ref="founder" use="required"/>
            <xs:attribute ref="founded" use="required"/>
        </xs:complexType>
    </xs:element>
    <xs:attribute name="slogan" type="xs:string" fixed="Gifts for all" />

    <xs:element name="GaloreGifts">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="Home" />
                <xs:element ref="GiftCards" />
                <xs:element ref="Store" />
            </xs:sequence>
            <xs:attribute ref="slogan" use="required"/>
        </xs:complexType>
    </xs:element>


</xs:schema>
```

4.Difference between Schema and DTD

Document Type Definition (DTD) :

Document Type Definition (DTD) is a document that specifies the structure of an XML document. It is used to precisely describe the characteristics of the XML language.

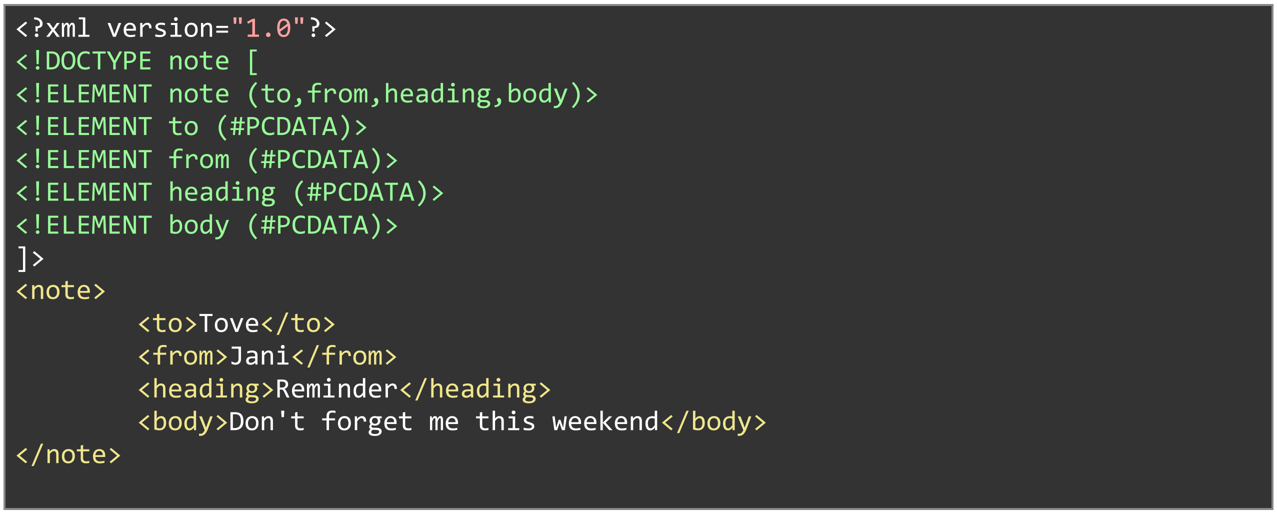
DTDs (document type definitions) serve two purposes. They define the syntax for specifying and restricting a document's logical structure, as well as creating a logical document out of physical entities. It can be specified both inside and outside of a document. The fundamental purpose of a DTD is to check the grammar and validity of an XML document. It determines whether an XML document has a valid structure.

It is divided into two types: internal DTD and external DTD.

1.Internal DTD

The DTD must be wrapped inside the !DOCTYPE> specification if it is declared inside the XML file:

XML document with an internal DTD

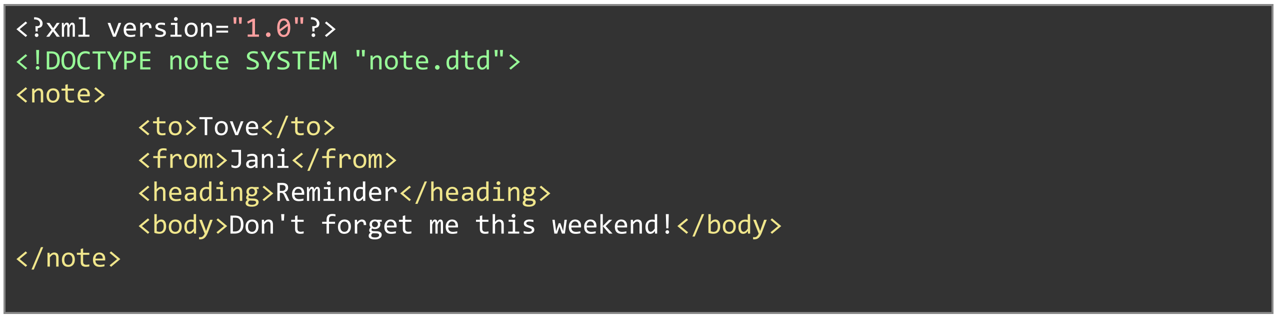


```
<?xml version="1.0"?>
<!DOCTYPE note [
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
]>
<note>
        <to>Tove</to>
        <from>Jani</from>
        <heading>Reminder</heading>
        <body>Don't forget me this weekend</body>
</note>
```

2.External DTD

The !DOCTYPE> definition must include a reference to the DTD file if the DTD is declared in an external file:

XML document with a reference to an external DTD



```
<?xml version="1.0"?>
<!DOCTYPE note SYSTEM "note.dtd">
<note>
        <to>Tove</to>
        <from>Jani</from>
        <heading>Reminder</heading>
        <body>Don't forget me this weekend!</body>
</note>
```

Here's the DTD, which is in the file "note.dtd":

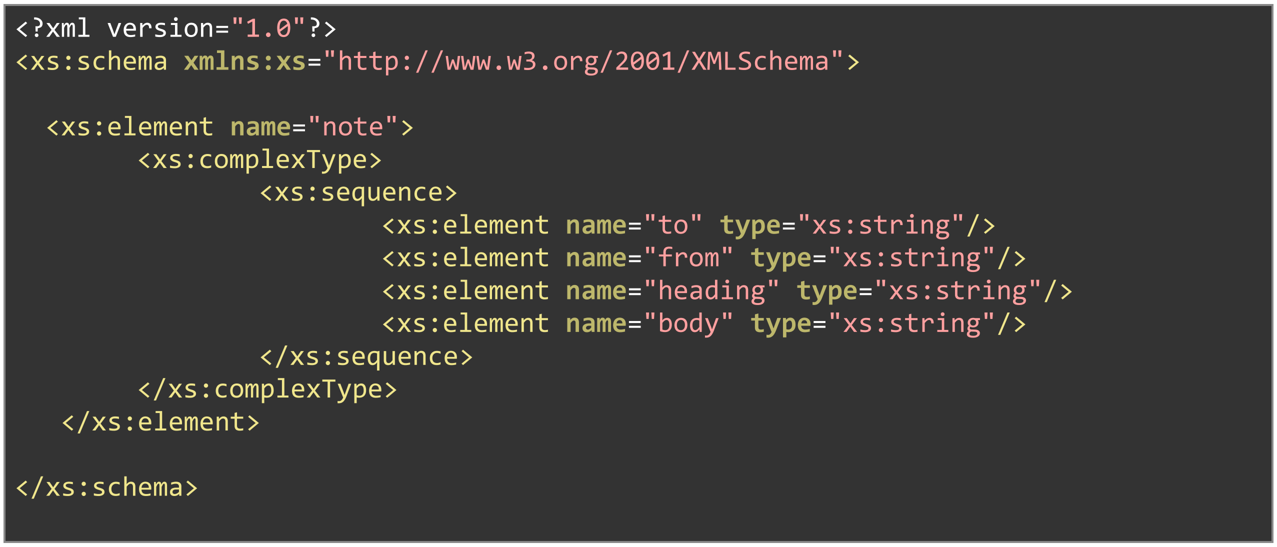
```
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
```

XML Schema Definition (XSD) :

The XML Schema Definition (XSD) is a way of describing the structure of an XML document. It specifies the rules for all an XML document's attributes and elements. It examines the document's vocabulary. structure of the XML file is proper.

A schema specifies the rules that a valid XML document must follow to ensure that the data it contains is in the correct format. In effect, the schema serves as a template, defining the format in which the XML document must be written. A program validates an XML document created against a schema when it processes it. The XML Schema definition is nothing more than a how-to guide for constructing an XML document that specifies the structure that other XML documents should adhere to. Despite their unique purpose, schemas are simply XML documents; we cannot combine a schema and its XML document into a single file.

XSD Example



```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="note">
        <xs:complexType>
                <xs:sequence>
                        <xs:element name="to" type="xs:string"/>
                        <xs:element name="from" type="xs:string"/>
                        <xs:element name="heading" type="xs:string"/>
                        <xs:element name="body" type="xs:string"/>
                </xs:sequence>
        </xs:complexType>
   </xs:element>

</xs:schema>
```

Difference between Document Type Definition (DTD) and XML Schema Definition (XSD) :

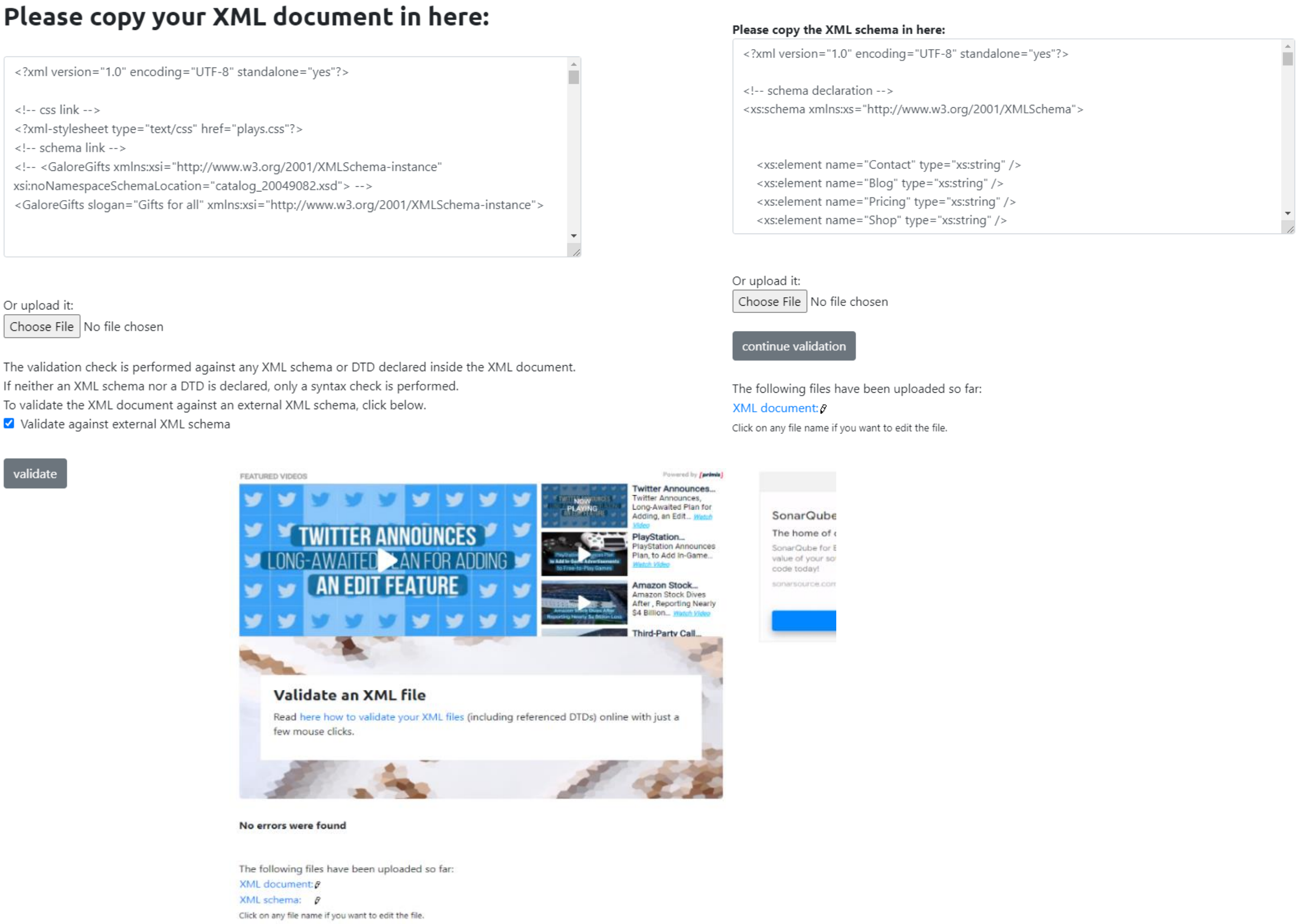
S.NO	DTD	XSD
1	DTD stands for Document Type Definition	XSD stands for XML Schema Definition.
2	DTD is derived from SGML (Standard Generalized Markup Language) syntax.	XSD is written in XML
3	DTD can be used to define the structure of an XML document.	XSD can be used to describe the structure and contents of an XML document.
4	DTD does not support Namespace and datatypes.	XSD supports Namespace and datatypes.
5	DTD has less control over XML.	XSD provides better control over XML.
6	DTD is not easy to learn.	XSD is easy to learn because here we do not need to learn a new language.
7	DTD is not extensible.	XSD is extensible.
8	DTD does not define order for child elements.	XSD defines order for child elements.

5. Testing

Test 1: Validating XML document with Schema

Test No.	1
Action	XML document and schema were inserted in xml validator
Expected Output	XML document should be properly validated
Actual Output	XML document validated without any errors
Test Result	The test was successful

Table 1Test 1



Please copy your XML document in here:

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

<!-- css link -->
<?xml-stylesheet type="text/css" href="plays.css"?>
<!-- schema link -->
<!-- <GaloreGifts xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="catalog_20049082.xsd"> -->
<GaloreGifts slogan="Gifts for all" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

Or upload it:

Choose File 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.
To validate the XML document against an external XML schema, click below.

☑ Validate against external XML schema

validate

Please copy the XML schema in here:

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

<!-- schema declaration -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">


    <xs:element name="Contact" type="xs:string" />
    <xs:element name="Blog" type="xs:string" />
    <xs:element name="Pricing" type="xs:string" />
    <xs:element name="Shop" type="xs:string" />
```

Or upload it:

Choose File No file chosen

continue validation

The following files have been uploaded so far:
XML document:
Click on any file name if you want to edit the file.

Validate an XML file

Read here how to validate your XML files (including referenced DTDs) online with just a few mouse clicks.

No errors were found

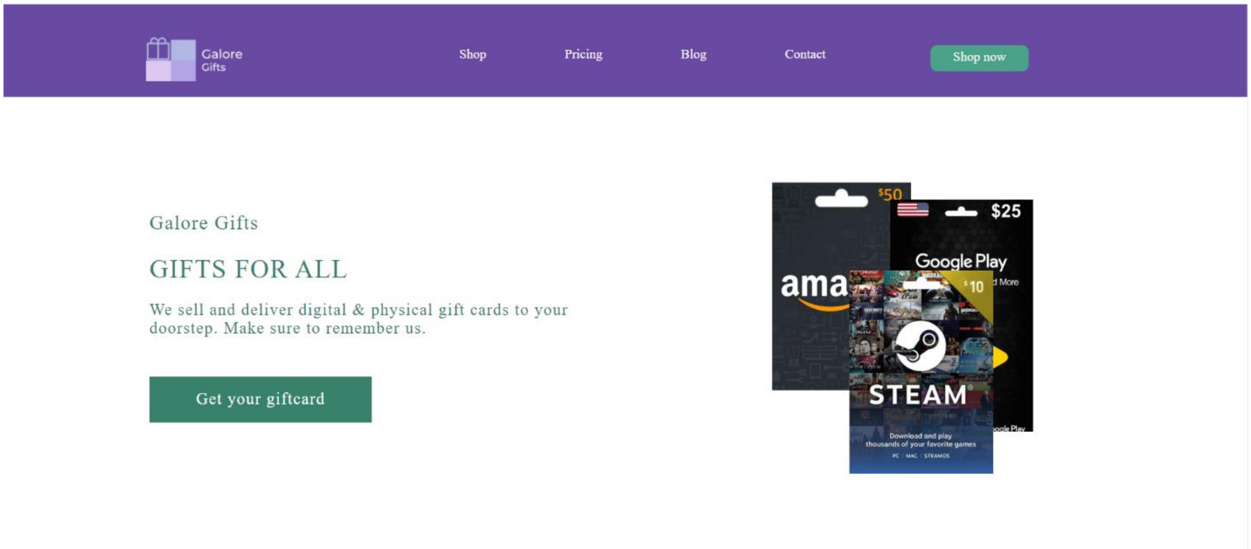
The following files have been uploaded so far:
XML document:
XML schema:
Click on any file name if you want to edit the file.

Figure 7 XML Validation

Test 2: Checking whether XML runs in the browser or not

Test No.	2
Action	Checking whether XML runs in the browser or not
Expected Output	XML document should run properly.
Actual Output	XML document ran successfully, and the website is show
Test Result	The test ran successfully

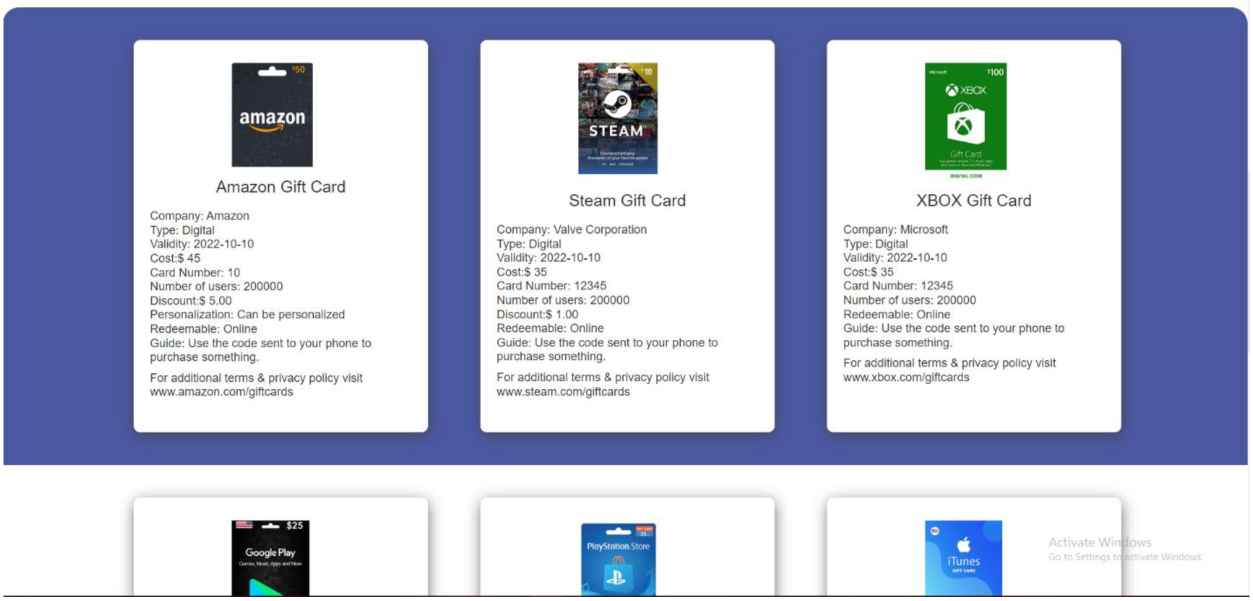
Table 2 Test 2



Galore Gifts

GIFTS FOR ALL

We sell and deliver digital & physical gift cards to your doorstep. Make sure to remember us.



Amazon Gift Card

Company: Amazon
Type: Digital
Validity: 2022-10-10
Cost:$ 45
Card Number: 10
Number of users: 200000
Discount:$ 5.00
Personalization: Can be personalized
Redeemable: Online
Guide: Use the code sent to your phone to purchase something.

For additional terms & privacy policy visit www.amazon.com/giftcards

Steam Gift Card

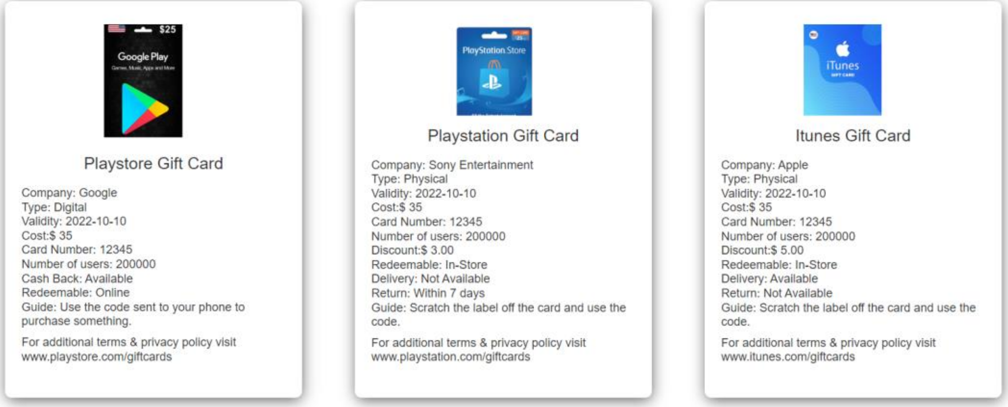
Company: Valve Corporation
Type: Digital
Validity: 2022-10-10
Cost:$ 35
Card Number: 12345
Number of users: 200000
Discount:$ 1.00
Redeemable: Online
Guide: Use the code sent to your phone to purchase something.

For additional terms & privacy policy visit www.steam.com/giftcards

XBOX Gift Card

Company: Microsoft
Type: Digital
Validity: 2022-10-10
Cost:$ 35
Card Number: 12345
Number of users: 200000
Redeemable: Online
Guide: Use the code sent to your phone to purchase something.

For additional terms & privacy policy visit www.xbox.com/giftcards



Playstore Gift Card

Company: Google
Type: Digital
Validity: 2022-10-10
Cost:$ 35
Card Number: 12345
Number of users: 200000
Cash Back: Available
Redeemable: Online
Guide: Use the code sent to your phone to purchase something.

For additional terms & privacy policy visit www.playstore.com/giftcards

Playstation Gift Card

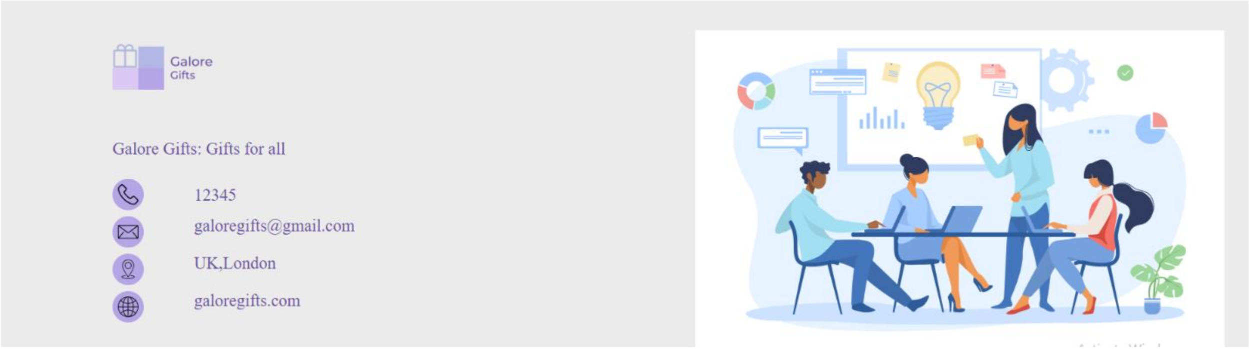
Company: Sony Entertainment
Type: Physical
Validity: 2022-10-10
Cost:$ 35
Card Number: 12345
Number of users: 200000
Discount:$ 3.00
Redeemable: In-Store
Delivery: Not Available
Return: Within 7 days
Guide: Scratch the label off the card and use the code.

For additional terms & privacy policy visit www.playstation.com/giftcards

Itunes Gift Card

Company: Apple
Type: Physical
Validity: 2022-10-10
Cost:$ 35
Card Number: 12345
Number of users: 200000
Discount:$ 5.00
Redeemable: In-Store
Delivery: Available
Return: Not Available
Guide: Scratch the label off the card and use the code.

For additional terms & privacy policy visit www.itunes.com/giftcards



Galore Gifts: Gifts for all

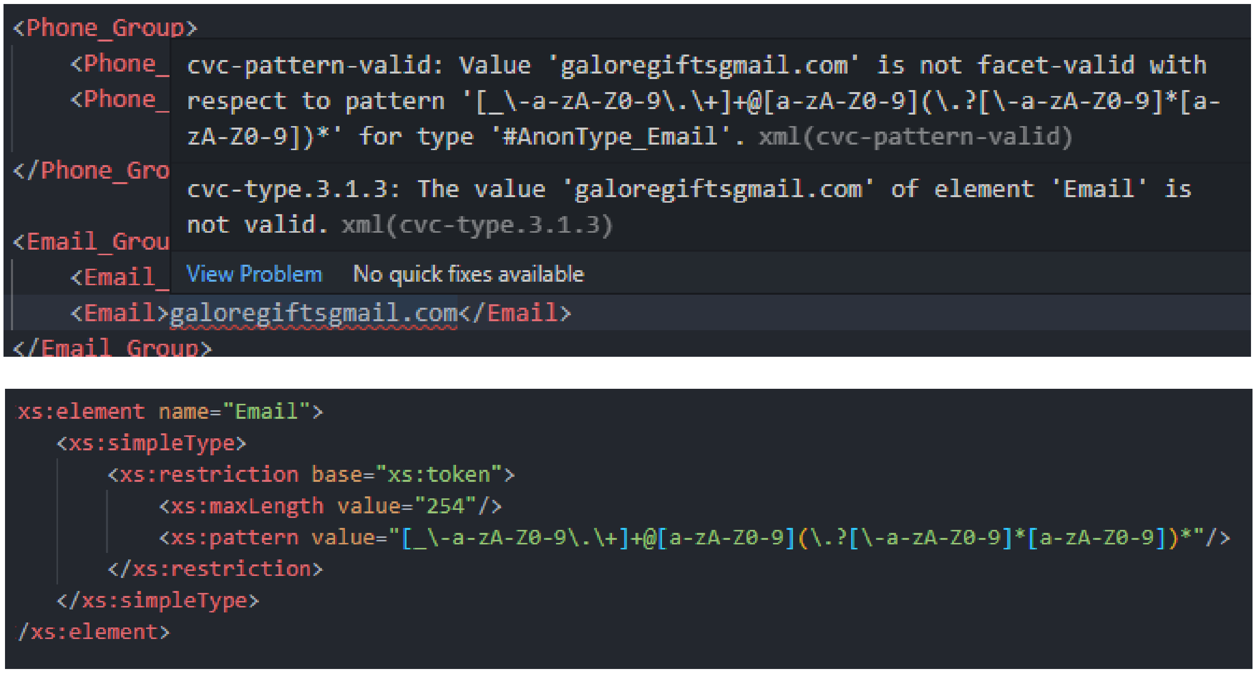
☎ 12345
✉ galoregifts@gmail.com
UK,London
galoregifts.com

Figure 8 Galore gifts website

Test 3: Validating email

Test No.	3
Action	Removing “@” sign from the email
Expected Output	An error message will be shown
Actual Output	An error message is shown “Value is not facet-valid with respect to pattern” while hovering it.
Test Result	The test was successful.

Table 3 Test 3



```
<Phone_Group>
    <Phone_
    <Phone_

</Phone_Gro

<Email_Grou
    <Email_
    <Email>galoregiftsgmail.com</Email>
</Email_Group>
```

```
cvc-pattern-valid: Value 'galoregiftsgmail.com' is not facet-valid with respect to pattern '[_\-a-zA-Z0-9\.\+]+@[a-zA-Z0-9](\.?[\-a-zA-Z0-9]*[a-zA-Z0-9])*' for type '#AnonType_Email'. xml(cvc-pattern-valid)

cvc-type.3.1.3: The value 'galoregiftsgmail.com' of element 'Email' is not valid. xml(cvc-type.3.1.3)
```

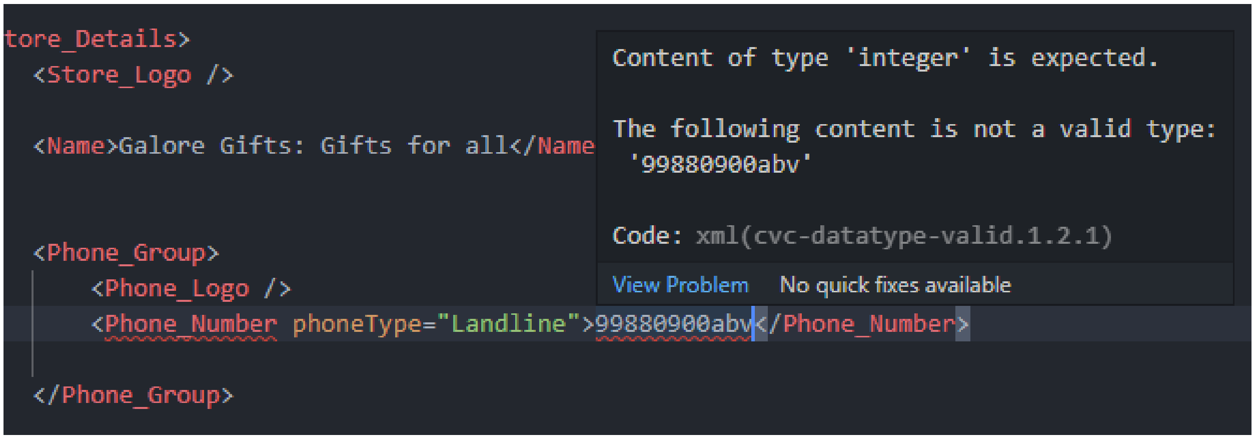
```
xs:element name="Email">
   <xs:simpleType>
       <xs:restriction base="xs:token">
           <xs:maxLength value="254"/>
           <xs:pattern value="[_\-a-zA-Z0-9\.\+]+@[a-zA-Z0-9](\.?[\-a-zA-Z0-9]*[a-zA-Z0-9])*"/>
       </xs:restriction>
   </xs:simpleType>
/xs:element>
```

Figure 9 XML and XSD for email

Test 4: Validating phone number

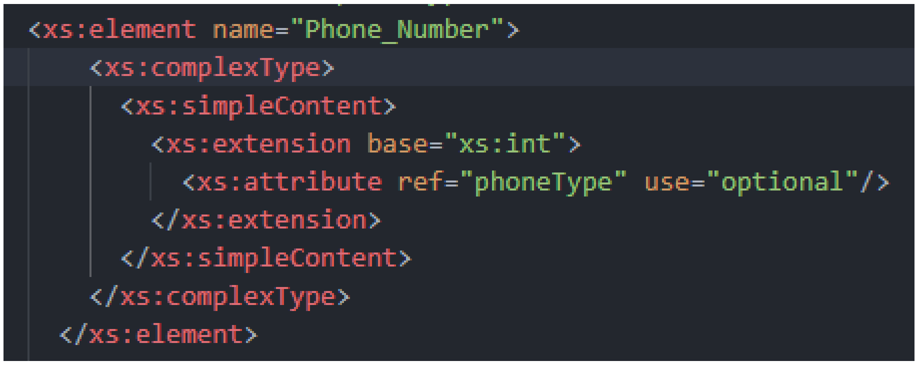
Test No.	4
Input	Adding alphabets in the phone number
Expected Output	An error message will be shown
Actual Output	An error message is shown “Content of type ‘integer’ is expected.” while hovering it
Test Result	The test was successful.

Table 4 Test 4



```
tore_Details>
  <Store_Logo />

  <Name>Galore Gifts: Gifts for all</Name


  <Phone_Group>
      <Phone_Logo />
      <Phone_Number phoneType="Landline">99880900abv</Phone_Number>

  </Phone_Group>
```

```
Content of type 'integer' is expected.

The following content is not a valid type:
 '99880900abv'

Code: xml(cvc-datatype-valid.1.2.1)
```



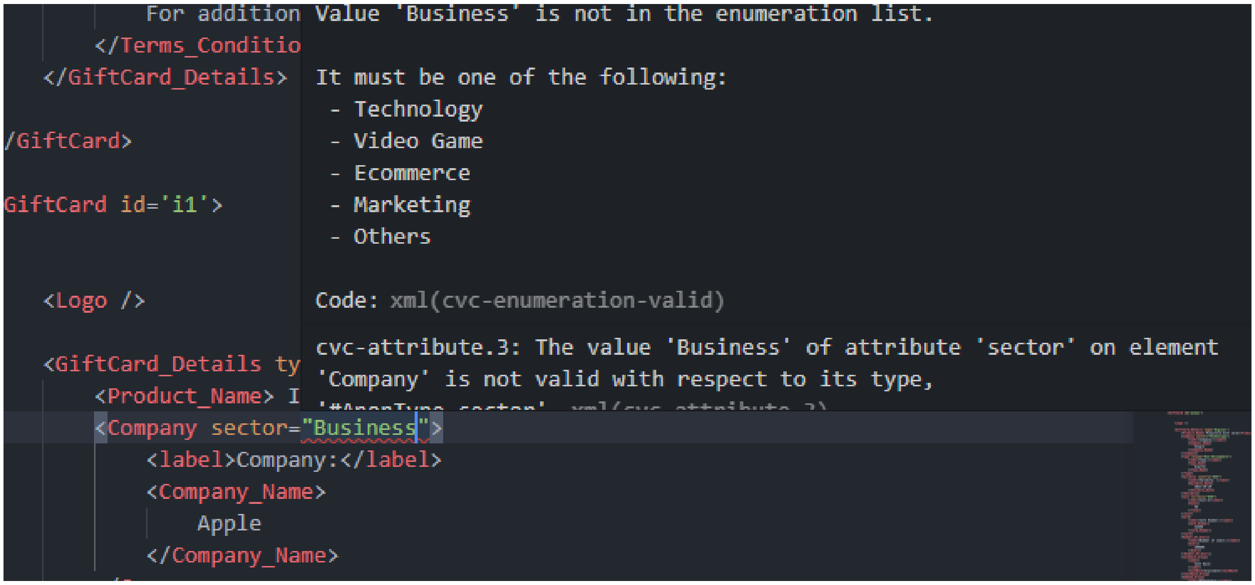
```
<xs:element name="Phone_Number">
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base="xs:int">
          <xs:attribute ref="phoneType" use="optional"/>
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>
```

Figure 10 XML and XSD for phone

Test 5: Validating enumeration

Test No.	5
Action	Adding “Business” as a sector value which is not in the enumeration
Expected Output	An error message must be shown
Actual Output	An error message is shown ‘Value “Business” is not in the enumeration list.’ while hovering it
Test Result	The test was successful.

Table 5 Test 5



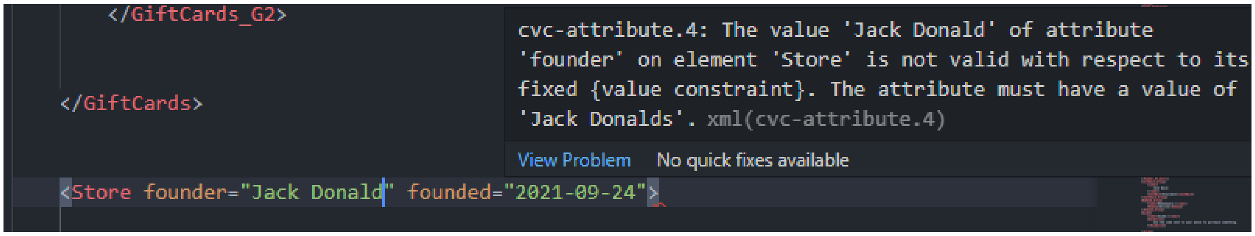
```
        For addition  Value 'Business' is not in the enumeration list.
      </Terms_Conditio
    </GiftCard_Details>  It must be one of the following:
                          - Technology
/GiftCard>                - Video Game
                          - Ecommerce
GiftCard id='i1'>         - Marketing
                          - Others

    <Logo />              Code: xml(cvc-enumeration-valid)

    <GiftCard_Details ty  cvc-attribute.3: The value 'Business' of attribute 'sector' on element
      <Product_Name> I    'Company' is not valid with respect to its type,
                          '#AnonType_sector'. xml(cvc-attribute.3)
      <Company sector="Business">
        <label>Company:</label>
        <Company_Name>
          Apple
        </Company_Name>
```

Figure 11 XML enumeration error

Test 6: Validating fixed attributes

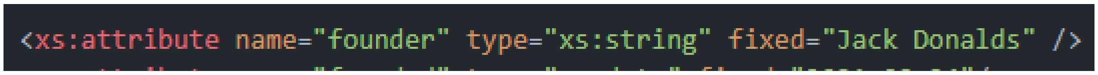
Test No.	6
Action	Entering incorrect founder value as “Jack Donald”
Expected Output	An error message must me shown.
Actual Output	An error message is shown “The value ‘Jack Donald’ of attribute ‘founder’ on element ‘Store’ is not valid to its fixed constraint”.
Test Result	The test was successful.

Table 6 Test 6



```
    </GiftCards_G2>

</GiftCards>

<Store founder="Jack Donald" founded="2021-09-24">
```

```
cvc-attribute.4: The value 'Jack Donald' of attribute 'founder' on element 'Store' is not valid with respect to its fixed {value constraint}. The attribute must have a value of 'Jack Donalds'. xml(cvc-attribute.4)
```



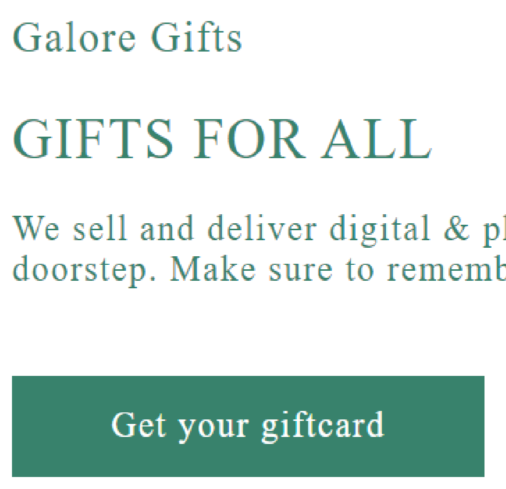
```
<xs:attribute name="founder" type="xs:string" fixed="Jack Donalds" />
```

Figure 12 XML and XSD for fixed attributes

Test 7: Testing CSS button hover effect

Test No.	7
Input	Hovering over the button
Expected Output	The button should pop up while hovering
Actual Output	The button pops up while hovering
Test Result	The test was successful

Table 7 Test 7

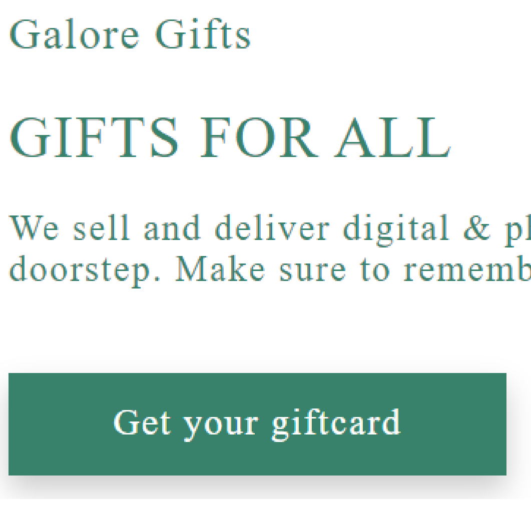


Galore Gifts

GIFTS FOR ALL

We sell and deliver digital & physical gift cards to your doorstep. Make sure to remember us.

Figure 13 Button before hover



Galore Gifts

GIFTS FOR ALL

We sell and deliver digital & physical gift cards to your doorstep. Make sure to remember us.

Figure 14 Button after hover

Test 8: Testing CSS hover effect on navigation item

Test No.	8
Action	Hovering over the navigation item “Shop”
Expected Output	A glowing effect should be seen
Actual Output	A glowing effect is seen when hovering the navigation item
Test Result	The test was successful

Table 8 Test 8

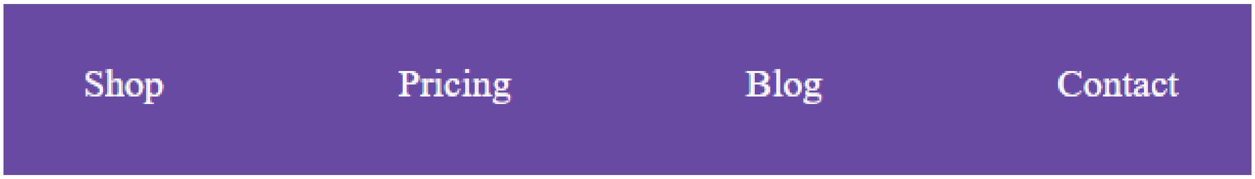


Figure 15 Navigation item before hover

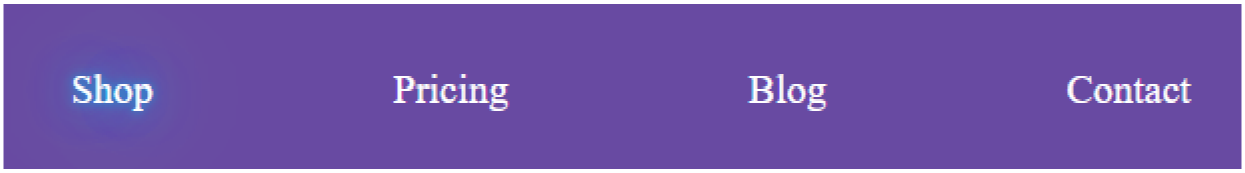
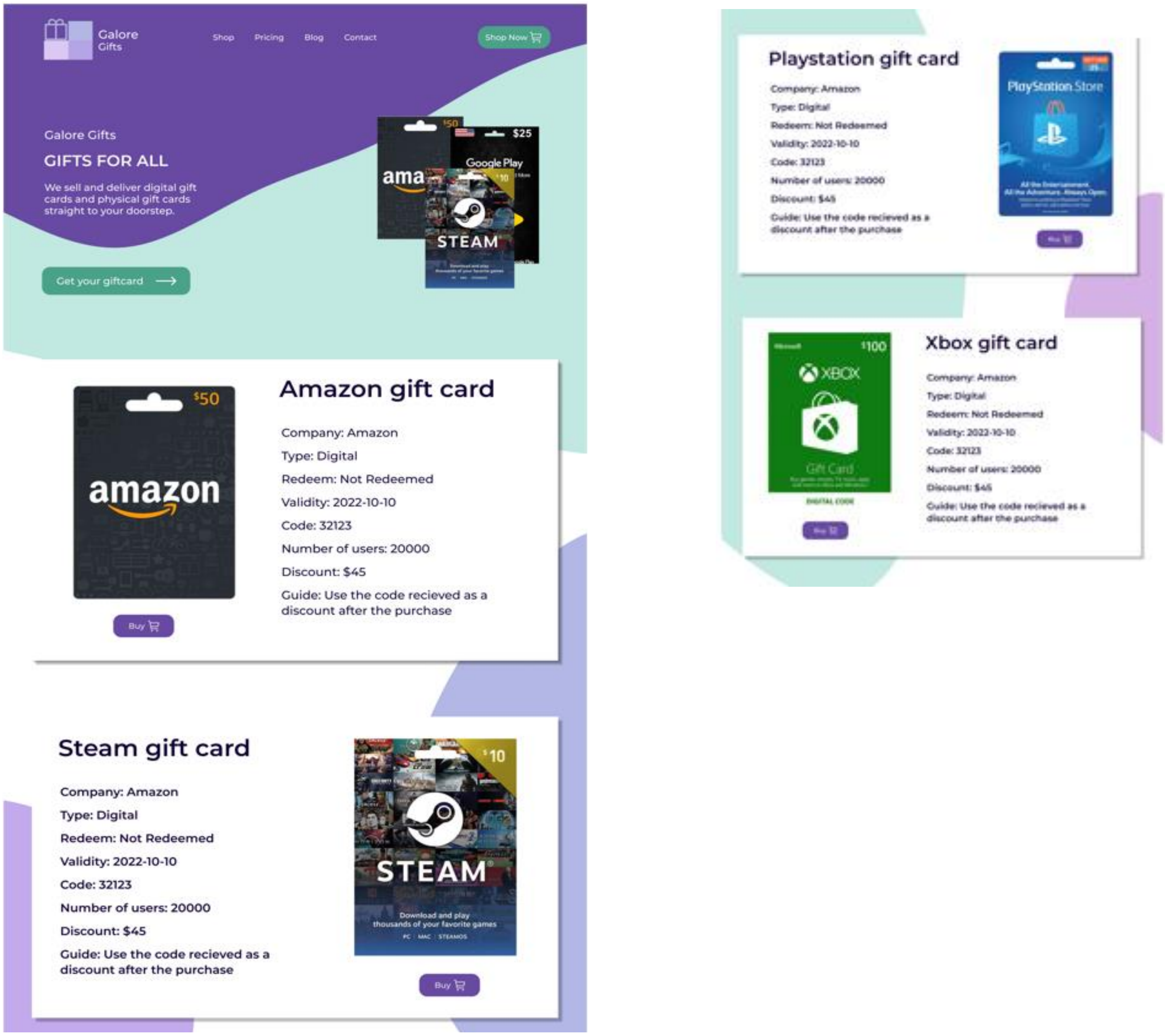


Figure 16 Navigation item after hover

6. Coursework Development

The coursework was started by carefully reading the questions and the requirements asked and evaluating the result according to it. Firstly, brainstorming was done to visualize the design of the coursework including the requirements in the design. Then, a design tool was selected to convert the visualized design into a real design. The most suitable and popular design tool, Figma, is the one I selected for this coursework.



Galore Gifts

GIFTS FOR ALL

We sell and deliver digital gift cards and physical gift cards straight to your doorstep.

Amazon gift card

Company: Amazon
Type: Digital
Redeem: Not Redeemed
Validity: 2022-10-10
Code: 32123
Number of users: 20000
Discount: $45
Guide: Use the code recieved as a discount after the purchase

Steam gift card

Company: Amazon
Type: Digital
Redeem: Not Redeemed
Validity: 2022-10-10
Code: 32123
Number of users: 20000
Discount: $45
Guide: Use the code recieved as a discount after the purchase

Playstation gift card

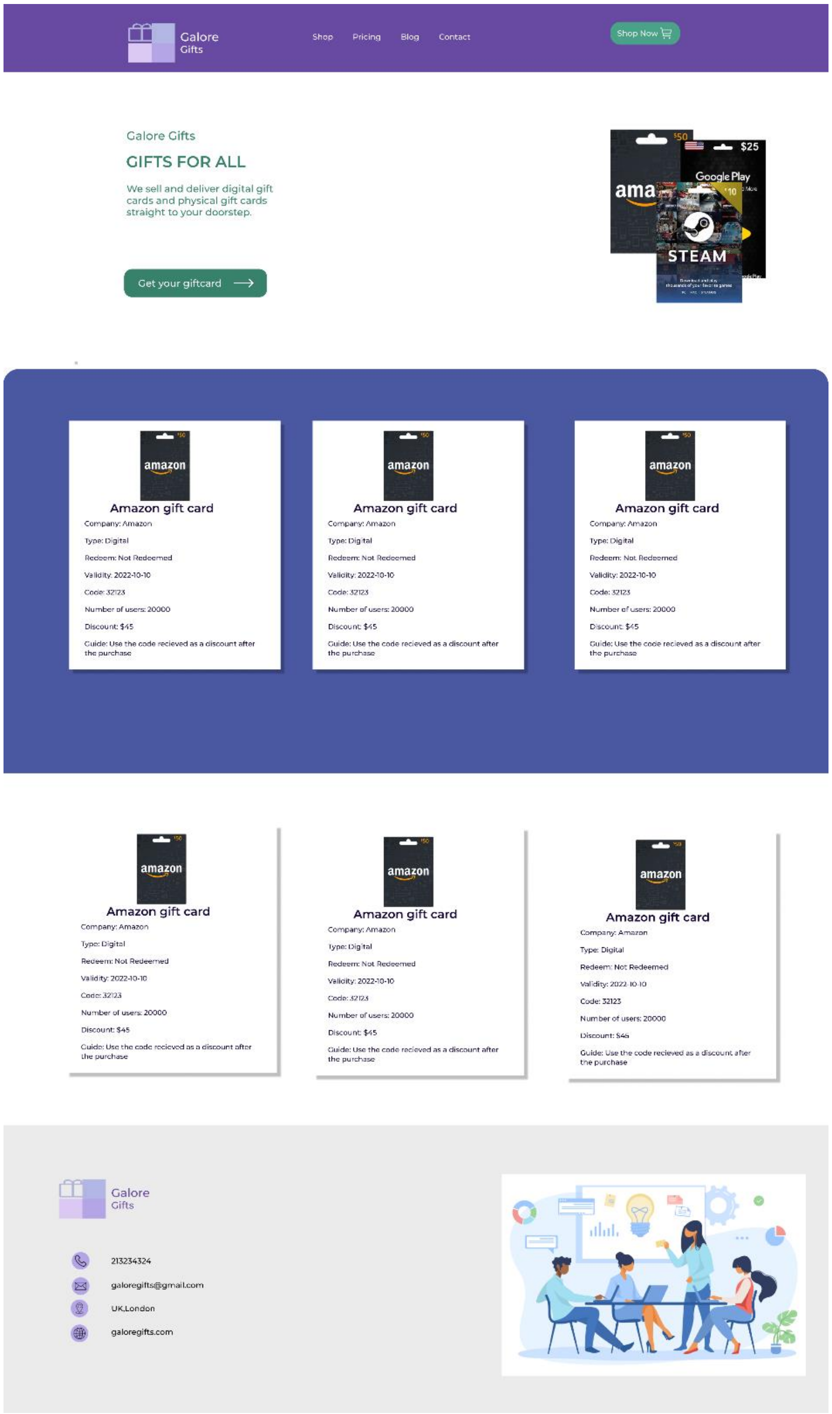
Company: Amazon
Type: Digital
Redeem: Not Redeemed
Validity: 2022-10-10
Code: 32123
Number of users: 20000
Discount: $45
Guide: Use the code recieved as a discount after the purchase

Xbox gift card

Company: Amazon
Type: Digital
Redeem: Not Redeemed
Validity: 2022-10-10
Code: 32123
Number of users: 20000
Discount: $45
Guide: Use the code recieved as a discount after the purchase

Figure 17 Initial design

Above is the initial design of the website. After trying to implement the design some of the things were quite difficult to implement like the waves and blobs, so the design had to be refactored into a simpler one.



Galore Gifts

GIFTS FOR ALL

We sell and deliver digital gift cards and physical gift cards straight to your doorstep.

Amazon gift card

Company: Amazon

Type: Digital

Redeem: Not Redeemed

Validity: 2022-10-10

Code: 32123

Number of users: 20000

Discount: $45

Guide: Use the code recieved as a discount after the purchase

Amazon gift card

Company: Amazon

Type: Digital

Redeem: Not Redeemed

Validity: 2022-10-10

Code: 32123

Number of users: 20000

Discount: $45

Guide: Use the code recieved as a discount after the purchase

Amazon gift card

Company: Amazon

Type: Digital

Redeem: Not Redeemed

Validity: 2022-10-10

Code: 32123

Number of users: 20000

Discount: $45

Guide: Use the code recieved as a discount after the purchase

Amazon gift card

Company: Amazon

Type: Digital

Redeem: Not Redeemed

Validity: 2022-10-10

Code: 32123

Number of users: 20000

Discount: $45

Guide: Use the code recieved as a discount after the purchase

Amazon gift card

Company: Amazon

Type: Digital

Redeem: Not Redeemed

Validity: 2022-10-10

Code: 32123

Number of users: 20000

Discount: $45

Guide: Use the code recieved as a discount after the purchase

Amazon gift card

Company: Amazon

Type: Digital

Redeem: Not Redeemed

Validity: 2022-10-10

Code: 32123

Number of users: 20000

Discount: $46

Guide: Use the code recieved as a discount after the purchase

Galore
Gifts

☎ 213234324
✉ galoregifts@gmail.com
UK,London
galoregifts.com

Figure 18 Final design

The above is the final refactored design created in Figma which was later implemented using XML and CSS.

During the designing process, several things that were required in the coursework were kept in mind. The details of the store including store name, address, telephone number, website address (URL), and logo (image) and gift card having properties like type, cost, validity, number of users, discount vouchers and so on were all included. Additionally, at least 15 unique data, 5 unique attributes, 5 optional data (xml elements) were also included in the XML and designed. In the XML part, restrictions, pattern, enumeration where also included in the schema file.

Furthermore, in the CSS part, borders, two different font sizes and font families, colors, logo, flexbox are used. Some animations were also done in CSS like hovering a button creates a 3D pop up effect, clicking the button creates a pressing effect, when hovering the navigation menu items, the navigation item glows.

7. Critical Analysis

I had to put in a lot of time researching the essential materials for this coursework. To complete this homework, I had to read through the lecture materials, do some googling, and watch some YouTube videos. This was a difficult coursework for me because I had no experience with XML before to this semester.

To begin, a significant amount of time was spent selecting the color scheme for this course. Dribble was one of the websites where I found inspiration. During this coursework, several problems arose. I had made several foolish blunders that prevented me from detecting the problem for a long time, such as forgetting to close the brackets and failing to close the self-closing elements with the "/." I experienced a lot of constant errors since the element and attribute names of XML sometimes differed from those of XSD. The elements of the gift card, as well as the attributes of the gift cards, were included after much thought. I had to research gift cards and their features on the internet. Furthermore, I had to update the CSS style such as margins, padding, width, and height on a constantly. It was difficult to get the image to the required area.

However, I was really determined to accomplish this coursework, and with the support of my lecturers and tutors, as well as my own hard work, I was able to complete it in the time frame allotted. I am grateful for the assistance I received from my instructors in both my tutorial classes and on social media platforms such as Gmail and hangouts.

8. Conclusion

After a long-dedicated effort, I have finally completed this project in which I generated an XML document, a Schema document, and a CSS document for Galore Gifts, an online gift card shop. This webpage is mostly used to view the store's available gift cards. In the store, there are many popular and best-selling gift cards, and this website has all the information about those gift cards. All the prerequisites were easily met, and I took a few extra steps to improve my grades and, most importantly, gain a better understanding of XML. By completing this coursework, I was able to better understand the concepts of XML, Schema, and DTD, as well as improve my CSS skills. I also learned how to make a tree diagram and how to test a program. Finally, I am grateful that I was given the opportunity to learn about XML and complete this amazing and challenging assignment.

9.References

askanydifference, 2022. *Difference Between XML Schema And DTD (With Table).* [Online]
Available at: https://askanydifference.com/difference-between-xml-schema-and-dtd-with-table/
[Accessed 28 April 2022].

Dykes, L. & Tittel, E., 2005. *XML for dummies.* 4th ed. s.l.:Wiley Publishing,Inc..

Geeks For Geeks, 2022. *Difference between Document Type Definition (DTD) and XML Schema Definition (XSD).* [Online]
Available at: https://www.geeksforgeeks.org/difference-between-document-type-definition-dtd-and-xml-schema-definition-xsd/
[Accessed 28 April 2022].

Skonnard, A. & Gudgin, M., 2002. *Essential XML quick reference.* Addison-Wesley ed. s.l.:s.n.

w3schools, 2020. *XML.* [Online]
Available at: https://www.w3schools.com/xml/xml_whatis.asp
[Accessed 28 April 2022].