



Софийски университет „Св. Кл. Охридски“

Факултет по математика и информатика

*Специалност
„Софтуерно инженерство“*



Курсов проект

XML технологии за семантичен уеб

Зимен семестър, 2023/2024 год.

Тема № 01: „Каталог на хотели - 1“

Автори:

Мария Иванова, Ф№ 9MI0600045

Николай Костандиев, Ф№ 6MI0600046

Ръководители:

доц. д-р Александър Димов

гл. ас. д-р Явор Данков

януари, 2024
София

Съдържание

1	Въведение	3
2	Анализ на решението	4
2.1	Работен процес.....	4
2.2	Структура на съдържанието	4
2.3	Тип и представяне на съдържанието.....	7
3	Дизайн	8
4	Тестване	15
5	Заклучение и възможно бъдещо развитие.....	18
6	Разпределение на работата	19
7	Използвани литературни източници и Уеб сайтове	19

1 Въведение

Настоящият документ представлява документация на курсов проект за предмета „XML технологии за семантичен Уеб“. Проектът създава каталог на хотелите в България по региони. Каталогът е базиран на XML документи с текстово и графично съдържание, описващо възможните характеристики на хотели в страната.

Представения каталог съдържа актуална и лесно достъпна информация за хотелите в страната, което е от съществено значение за удобство на пътуващите, за развитието на хотелския бизнес, както и на туризма в страната.

Този проект решава проблема за систематизиране на информацията за хотелите в България. Този проблем е актуален за туристите, които търсят информация за хотели в конкретен район или с конкретни характеристики.

Решението на поставената задача се решава с различни XML технологии, които ни позволяват да създадем каталог с консистентна информация и стабилна структура, и да валидираме и трансформираме информацията. Основните технологии са:

- XML документ за структуриране на входните данни
- DTD документ за валидация на XML документа
- XSLT документ за трансформиране на XML документа в PDF файл

Проектът използва XML формат за структуриране на данните, като представя графичното съдържание чрез XML entities, описва връзките в документа чрез атрибути ID/IDREF. Валидацията на XML документа се извършва чрез DTD документ, а за генериране на PDF документ се използва XSLT технология. Трансформацията в PDF файл се извършва чрез Java код.

Този документ е структуриран по следния начин:

1. Въведение
2. Анализ на решението
 - 2.1. Работен процес
 - 2.2. Структура на съдържанието
 - 2.3. Тип и представяне на съдържанието
3. Дизайн
4. Тестване
5. Заключение и възможно бъдещо развитие
6. Разпределение на работата
7. Използване на литературни източници с Уеб сайтове

2 Анализ на решението

2.1 Работен процес

Работният процес за изработване на проекта се състои от няколко стъпки, през които системата преминава от приемането на входните данни до генерирането на крайния продукт. Тези стъпки включват създаването на няколко документа, които са нужни за последователна обработка, валидация и трансформиране на данните.

Входното съдържание представлява множество данни за хотели, вериги от хотели и региони в България. Тази информация се получава от различни източници, като уеб сайтове на веригите от хотели, резервационни платформи, уеб сайтове, съдържащи информация за регионите и други. Всичките тези данни са представени в XML файл, който описва избрани хотели (с обща информация, снимки и характеристики), вериги от хотели и региони (с обща информация, информация за забележителности и транспорт) в България. Във файлът са представени също връзки между някои елементи чрез XML entities и графични елементи чрез ID/IDREF атрибути.

Събраната информация се валидира чрез изработена DTD схема, която гарантира коректност и консистентност на структурирането на данните. В нея е описана структурата, която XML документа трябва да спазва за неговите елементи, поделемента, атрибути, ограничения и връзки.

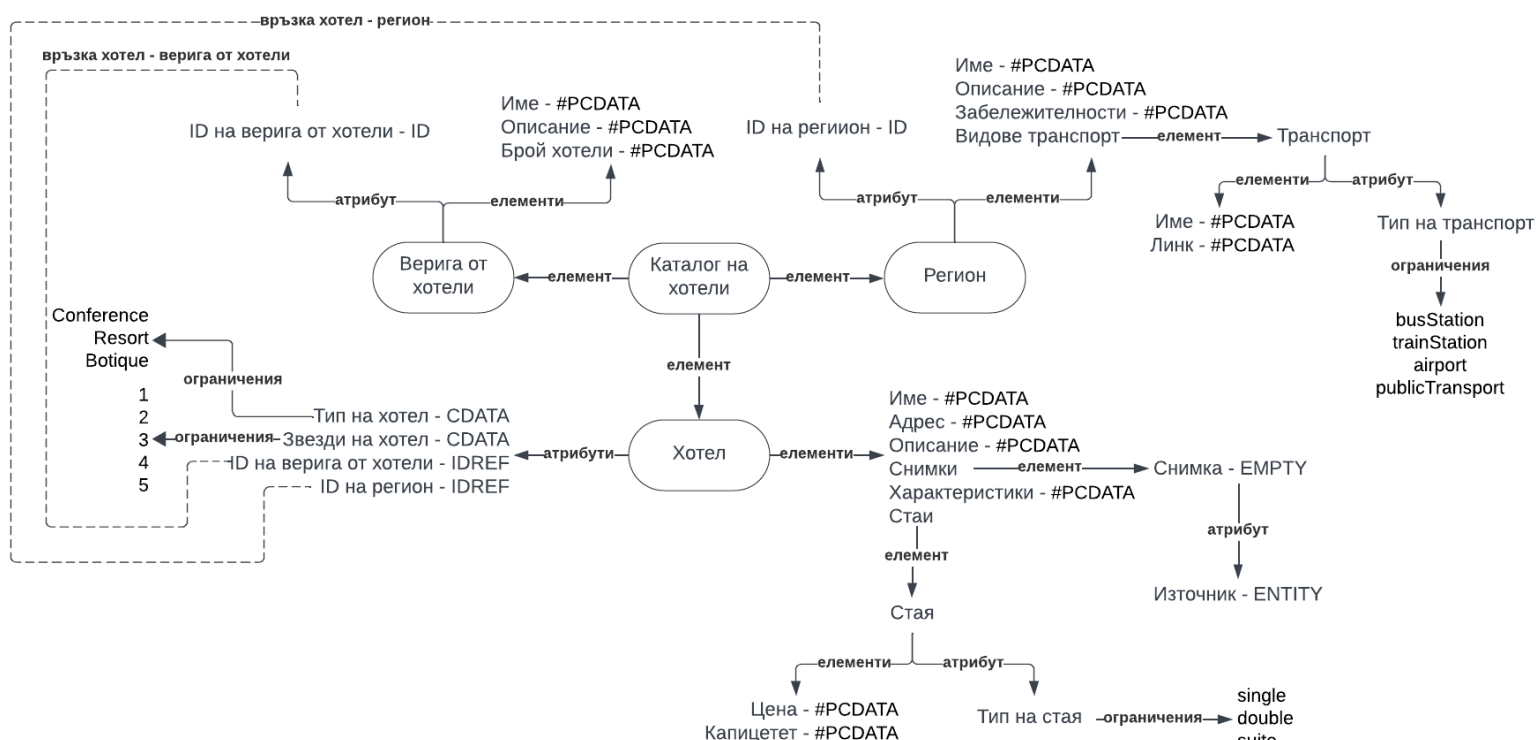
След като XML файлът е валидиран, той се трансформира в PDF файл чрез XSLT файл, който е изготвен за структуриране и свързване на информацията, и спазва някакъв формат. Самата трансформация е извършена чрез Java код.

Исходното съдържание е генериран PDF файл, представляващ каталог на хотели. Този каталог предоставя информация за хотели по региони, която може да бъде използвана от потребителите за лесен достъп до търсената от тях информация. PDF документа се предоставя на крайният потребител, който удобно може да разглежда добре структурираните от нас данни.

2.2 Структура на съдържанието

XML структурата е организирана в йерархична форма, така че лесно да се добавят, променят, премахват и достъпват данните, които са ни нужни. Съдържанието включва корен (каталог на хотели), който има три поделмента – главните елементи (хотел, верига от хотели, регион) и множество подчинени на тях други елементи (снимки, характеристики, забележителности и други). Използвани са атрибути на нужните места, както и са направени връзки между трите главни елемента. Добавено е и графично съдържание за по-пълно представяне на хотелите.

Графично представяне на структурата:



Таблично представяне на структурата на съдържанието:

Елемент	Характеристика	Честота на срещане
Каталог на хотели (hotelCatalog)	Корен на цялата структура, съдържащ в себе си един или повече хотели, вериги от хотели и региони	1
Хотел (hotel)	Един от трите главни елемента, съдържащ задължително в себе си множество поделемни: име, адрес, описание, снимки, характеристика и стаи	1 или повече
Име (name)	Представява текстово съдържание (#PCDATA) на името на съответния елемент	Веднъж за всеки хотел, всяка верига от хотели, всеки регион и всеки транспорт
Адрес (address)	Представява текстово съдържание (#PCDATA) на адреса на съответния елемент	Веднъж за всеки хотел
Описание (description)	Представява текстово съдържание (#PCDATA) на описанието на съответния елемент	Веднъж за всеки хотел, всяка верига от хотели и всеки регион

Тип на хотел (hotel type)	Атрибут (задължителен) на елемента хотел, които може да е един от следните изрази: Conference, Resort или Botique	Веднъж за всеки хотел
Звезди на хотел (hotel stars)	Атрибут (задължителен) на елемента хотел, които може да е един от следните изрази: 1, 2, 3, 4 или 5	Веднъж за всеки хотел
Снимки (photos)	Съдържа 0 или повече поделемента снимка	Веднъж за всеки хотел
Снимка (photo)	Представява елемент от празен тип	Произволен брой за всеки елемент снимки
Източник (src)	Атрибут (задължителен) на елемента снимка от тип ENTITY, който съдържа абсолютния път към снимката и така се представя графичното съдържание	Веднъж за всяка снимка
Характеристики (features)	Представява текстово съдържание (#PCDATA) на характеристиките на съответния елемент	Веднъж за всеки хотел
Стаи (rooms)	Съдържа 1 или повече поделемента стая	Веднъж за всеки хотел
Стая (room)	Съдържа задължително е себе си поделемента цена и капацитет	Произволен брой за всеки елемент стаи
Цена (price)	Представява текстово съдържание (#PCDATA) на цената на съответния елемент	Веднъж за всяка стая
Капацитет (capacity)	Представява текстово съдържание (#PCDATA) на капацитета на съответния елемент	Веднъж за всяка стая
Тип на стая (room type)	Атрибут (задължителен) на елемента стая, които може да е един от следните изрази: single, double или suite	Веднъж за всяка стая
Верига от хотели (hotelChain)	Един от трите главни елемента, съдържащ задължително в себе си поделемента име, описание и брой хотели	1 или повече
Брой хотели (numberOfHotels)	Представява текстово съдържание (#PCDATA) на броя на хотелите на съответния елемент	Веднъж за всяка верига от хотели
ID на верига хотели (hotelChain/hotel hotelChainID)	Атрибут от тип ID/IDREF, чрез който се прави връзка хотел – верига от хотели	Веднъж за всеки хотел (IDREF) и всяка верига от хотели (ID)
Регион (region)	Един от трите главни елемента, съдържащ задължително в себе си поделемента име,	1 или повече

	описание, забележителности и видове транспорт	
ID на регион (region/hotel regionID)	Атрибут от тип ID/IDREF, чрез който се прави връзка хотел – регион	Веднъж за всеки хотел (IDREF) и всеки регион (ID)
Забележителности (sights)	Представява текстово съдържание (#PCDATA) на забележителностите на съответния елемент	Веднъж за всеки регион
Видове транспорт (transports)	Съдържа 1 или повече поделементи транспорт	Веднъж за всеки регион
Транспорт (transport)	Съдържа задължително в себе си поделементите име и линк	Произволен брой за всеки елемент видове транспорт
Линк (link)	Представява текстово съдържание (#PCDATA) на линк към уеб сайта на съответния елемент	Веднъж за всеки транспорт
Тип на транспорт (transport type)	Атрибут (задължителен) на елемента транспорт, които може да е един от следните изрази: busStation, trainStation, airport или publicTransport	Веднъж за всеки транспорт

2.3 Тип и представяне на съдържанието

Съдържанието представено в проекта е основно текстово, като са включени и графични елементи. Текстовото съдържание е в различни типове файлове, изработени от нас. Графичните елементи предоставят по две снимки на включените хотели, както и крайния PDF файл. Снимките са взети от уеб сайт за резервации: <https://www.booking.com>

- Текстовите ресурси са 5. Това са XML файл с основната структура на данните, DTD файл със схема за валидиране, XSLT файл за трансформация, Java код, чрез който се извършва трансформацията
- Графичните файлове са 15, от които 13 JPEG и 1 PNG снимки на хотелите от XML документа и PDF файла, изработен за трансформацията на XML файла

Файл	Размер	Брой	Тип	Начин на кодиране
XML документ	21KB	1	XML document	UTF-8
DTD документ	1KB	1	XML document	UTF-8
XSLT документ	34 KB	1	XML document	UTF-8
Java код	2KB	1	Java source code	UTF-8
JPEG снимки	между 20 KB и 45 KB	14	JPG	бинарен код
PNG снимка	500 KB	1	PNG	бинарен код
PDF документ	1,1 MB	1	PDF document	UTF-8

3 Дизайн

Решението на поставената задача използва XML технологиите с различни функционалности. Технологиите включена за дизайна на конкретния проект са:

- XML документ, включващ входните данни за каталога на хотели в България:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <?xml-stylesheet type="text/xsl" href="hotelCatalog.xsl"?>
3 <!DOCTYPE hotelCatalog SYSTEM "hotelCatalog.dtd" [
4   <!NOTATION PNG SYSTEM "image/png">
5   <!NOTATION JPG SYSTEM "image/jpeg">
6   <!ENTITY festa_sofia_hotel SYSTEM "images/Festa_sofia_hotel.jpg" NDATA JPG>
7   <!ENTITY festa_sofia_room SYSTEM "images/Festa_sofia_room.jpg" NDATA JPG>
8   <!ENTITY festa_borovets_hotel SYSTEM "images/Festa_borovets_hotel.jpg" NDATA JPG>
9   <!ENTITY festa_borovets_room SYSTEM "images/Festa_borovets_room.jpg" NDATA JPG>
10  <!ENTITY royal_castle_hotel SYSTEM "images/Royal_castle_hotel.jpg" NDATA JPG>
11  <!ENTITY royal_castle_room SYSTEM "images/Royal_castle_room.jpg" NDATA JPG>
12  <!ENTITY marinela_hotel SYSTEM "images/Marinela_hotel.jpg" NDATA JPG>
13  <!ENTITY marinela_room SYSTEM "images/Marinela_room.jpg" NDATA JPG>
14  <!ENTITY chaika_hotel SYSTEM "images/Chaika_hotel.png" NDATA PNG>
15  <!ENTITY chaika_room SYSTEM "images/Chaika_room.jpg" NDATA JPG>
16  <!ENTITY rila_borovets_hotel SYSTEM "images/Rila_borovets_hotel.jpg" NDATA JPG>
17  <!ENTITY rila_borovets_room SYSTEM "images/Rila_borovets_room.jpg" NDATA JPG>
18  <!ENTITY perla_beach_hotel SYSTEM "images/Perla_beach_hotel.jpg" NDATA JPG>
19  <!ENTITY perla_beach_room SYSTEM "images/Perla_beach_room.jpg" NDATA JPG>
20 ]>
21
```

- Първият ред показва, че това е XML документ, като указва версията на XML (1.0) и начина на кодиране (UTF-8)
- На вторият ред е съобщено, че документът се стилизира с XSL и се посочва местоположението на XSL файлът
- След това имаме DOCTYPE обявление, което посочва външен DTD файл, както и вътрешна DTD схема, като се посочват декларации на entities, които са конкретни стойности атрибутите от тип ENTITY на елементите снимка в проекта. Така се реферират използваните изображения. Тъй като типът на снимките е JPEG и PNG, което не е XML съдържание, е необходимо да се използва NDATA секция със съответния тип на данните, представени като NOTATION

```
22 <hotelCatalog>
23
24 <hotel hotelChainID="Festa" regionID="SOF" type="Conference" stars="4">
25   <name>Festa Sofia</name>
26   <address>Bulgaria Blvd. 83, 1404 Manastirski Livadi, Sofia</address>
27   <description>
28     Located 200 meters from Mall Bulgaria and the tram stop, Festa Sofia hotel offers elegant rooms with free Wi-Fi and satellite TV.
29     It features a wellness center and beauty salon, as well as 2 restaurants, 3 bars, and 4 conference halls. The spa center includes a sauna and steam bath.
30     Guests can stay fit in the fitness center or enjoy massage and aromatherapy treatments. There is also a hair salon on site.
31     Restaurant Festa serves local and international cuisine in a secluded atmosphere. The Mezzanine bar offers sophisticated drinks and desserts.
32     The lobby bar serves a variety of coffee specialties. Sahara Hookah Bar offers its guests exquisite varieties of tobacco and a wonderful selection of drinks.
33     All rooms at Festa Sofia hotel have a panoramic view of the city or Vitosha. They are equipped with air conditioning and a safe.
34     The private bathrooms have a bathtub. The hotel is located in a modern business area, 18 km from Sofia International Airport and 5 km from the city center.
35   </description>
36   <photos>
37     <photo src="festa_sofia_hotel" />
38     <photo src="festa_sofia_room" />
39   </photos>
40   <features>
41     Private Parking, Free WiFi, Airport Shuttle, Fitness Center, Family Rooms, Non-Smoking Rooms, 2 Restaurants,
42     24-Hour Reception, Bar, Good Breakfast
43   </features>
44   <rooms>
45     <room type="single">
46       <price>120</price>
47       <capacity>1</capacity>
48     </room>
49     <room type="double">
50       <price>150</price>
51       <capacity>2</capacity>
52     </room>
53     <room type="suite">
54       <price>200</price>
55       <capacity>3</capacity>
56     </room>
57   </rooms>
58 </hotel>
59
60 <hotel hotelChainID="Festa" regionID="BOR" type="Botique" stars="5">--
61 </hotel>
62
63 <hotel hotelChainID="Vic" regionID="BRG" type="Resort" stars="5">--
64 <hotel hotelChainID="Vic" regionID="SOF" type="Conference" stars="5">--
65 </hotel>
66
67 <hotel hotelChainID="Vic" regionID="BRG" type="Resort" stars="3">--
68 </hotel>
69
70 <hotel hotelChainID="BT" regionID="BOR" type="Resort" stars="4">--
71 </hotel>
72
73 <hotel hotelChainID="BT" regionID="BRG" type="Resort" stars="4">--
74 </hotel>
75
```


- След това започва кореновият елемент
- Представени са 7 елемента хотел. Всеки от тях следва една и съща структура със задължителни атрибути и елементи
- Два атрибута ID на верига от хотели и ID на регион са от тип IDREF, като така се осъществява връзка с елементите верига от хотели и регион
- Имаме елементи снимка, които са от тип EMPTY и чрез атрибут от тип ENTITY се свързват с източника на снимката

```

248 <hotelChain hotelChainID="Vic">
249   <name>Victoria Group Hotels and Resorts</name>
250   <description>
251     Their portfolio consists of 11 hotels categorized as 3, 4, and 5 stars, over 500 holiday villas, and multifunctional halls located in various destinations.
252     Their hotels impress with unconventional and unique designs. Their advantage lies in their locations: by the sea, in the center of Sofia, or near some of the best ski slopes!
253     Their vacation hotels offer All-Inclusive accommodation with included meals, drinks, and services, ensuring a worry-free vacation! The Victoria Group Hotels and Resorts
254     portfolio includes some of the most elegant and luxurious wedding venues in Bulgaria from the heavenly beaches of Elenite to the city lights of Sofia.
255   </description>
256   <numberOfHotels>10</numberOfHotels>
257 </hotelChain>
258
259 <hotelChain hotelChainID="BT">--
260 </hotelChain>
261
262 <hotelChain hotelChainID="Festa">--
263 </hotelChain>

```

- Показани са 3 елемента верига от хотели, като всеки от тях има задължителни атрибути и елементи
- Атрибутът ID на верига от хотели е от тип ID и прави връзката с елемента хотел

```

282 <region regionID="SOF">
283   <name>Sofia</name>
284   <description>
285     Sofia is the capital and the largest city in Bulgaria. The population is around 1.3 million people. Sofia is located in the central part of Western Bulgaria.
286     It serves as the main administrative, industrial, transportation, cultural, and educational center of the country. The transportation system in the capital is highly developed.
287     In Sofia, you can find:
288     - Numerous educational institutions, including the oldest and largest higher education institution in Bulgaria – Sofia University "St. Kliment Ohridski,"
289     as well as the Bulgarian Academy of Sciences and the National Library "Sts. Cyril and Methodius."
290     - Temples of major world religions, with around 200 functioning Orthodox churches and chapels and over 40 monasteries, one of which is the memorial temple "Alexander Nevsky."
291     - Theatres and opera houses, including the Ivan Vazov National Theatre and the Sofia Opera and Ballet.
292     - Museums and galleries such as the National Historical Museum, the National Ethnographic Museum, and the National Art Gallery.
293     - Monuments from various historical periods.
294     - Concert halls, the largest of which is Arena Sofia with up to 15,000 seats.
295     - Vasil Levski Stadium, the largest stadium in the country with a capacity of 44,000 people.
296     - Other notable places include the National Palace of Culture, the two oldest and well-known bridges in Sofia – the Eagle Bridge and the Lion Bridge,
297     the largest park in Sofia – Borisova Gradina, and the Sofia Zoo.
298   </description>
299   <sights>
300     Sofia University "St. Kliment Ohridski", Bulgarian Academy of Sciences, National Library "Sts. Cyril and Methodius",
301     memorial temple "Alexander Nevsky", National Theatre "Ivan Vazov", Sofia Opera and Ballet, National Historical Museum,
302     National Ethnographic Museum, National Art Gallery, Arena Sofia, Vasil Levski Stadium, National Palace of Culture,
303     Eagle Bridge, Lion Bridge, Borisova Gradina, Sofia Zoo
304   </sights>
305   <transports>
306     <transport type="trainStation">
307       <name>Central Railway Station Sofia</name>
308       <link>https://www.centralnaavtogara.bg</link>
309     </transport>
310     <transport type="busStation">
311       <name>Central Bus Station Sofia</name>
312       <link>https://www.centralnaavtogara.bg</link>
313     </transport>
314     <transport type="airport">
315       <name>Sofia Airport</name>
316       <link>https://www.sofiaairport.bg</link>
317     </transport>
318     <transport type="publicTransport">
319       <name>Urban Mobility Center</name>
320       <link>https://www.sofiatraffic.bg</link>
321     </transport>
322   </transports>
323 </region>
324
325 <region regionID="BRG">--
326 </region>
327
328 <region regionID="BOR">--
329 </region>
330 </hotelCatalog>

```

- Представени са 3 елемента регион, като отново всеки от тях има задължителни атрибути и елементи
- Атрибутът ID на регион от тип ID прави връзката с елемента хотел

- DTD схема, валидираща структурата, правилата и ограниченията на XML документа, заедно с XML entities, позволяващи представянето на графичното съдържание, атрибути ID/IDREF, описващи връзките между елементи (хотел – верига от хотели, хотел – регион)

```

1  <!ELEMENT hotelCatalog (hotel+, hotelChain+, region+)>
2
3  <!ELEMENT hotel (name, address, description, photos, features, rooms)>
4  <!ELEMENT name (#PCDATA)>
5  <!ELEMENT address (#PCDATA)>
6  <!ELEMENT description (#PCDATA)>
7  <!ELEMENT features (#PCDATA)>
8  <!ATTLIST hotel hotelChainID IDREF #REQUIRED>
9  <!ATTLIST hotel regionID IDREF #REQUIRED>
10 <!ATTLIST hotel type (Conference|Resort|Botique) #REQUIRED>
11 <!ATTLIST hotel stars (1|2|3|4|5) #REQUIRED>
12
13 <!ELEMENT photos (photo*)>
14 <!ELEMENT photo EMPTY>
15 <!ATTLIST photo src ENTITY #REQUIRED>
16
17 <!ELEMENT rooms (room+)>
18 <!ELEMENT room (price, capacity)>
19 <!ELEMENT price (#PCDATA)>
20 <!ELEMENT capacity (#PCDATA)>
21 <!ATTLIST room type (single|double|suite) #REQUIRED>
22
23 <!ELEMENT hotelChain (name, description, numberOfHotels)>
24 <!ELEMENT numberOfHotels (#PCDATA)>
25 <!ATTLIST hotelChain hotelChainID ID #REQUIRED>
26
27 <!ELEMENT region (name, description, sights?, transports?)>
28 <!ATTLIST region regionID ID #REQUIRED>
29
30 <!ELEMENT sights (#PCDATA)>
31
32 <!ELEMENT transports (transport+)>
33 <!ELEMENT transport (name, link)>
34 <!ELEMENT link (#PCDATA)>
35 <!ATTLIST transport type (busStation|trainStation|airport|publicTransport) #REQUIRED>

```

Както можем да видим и в таблицата от 2.2 DTD схемата съдържа следните декларации:

- Кореновия елемент каталог на хотели със елементи хотел, верига от хотели и регион
- Елементите хотел, верига от хотели и регион имат атрибути и елементи, които ги описват
- Елементът снимка на хотел е деклариран като тип EMPTY с атрибут от тип ENTITY, сочещ източника на снимката
- Атрибутите ID на верига от хотели и ID на регион са от тип ID/IDREF, като при използване в елемента хотел са IDREF, а в елементите верига от хотели и регион са ID. Така се създава логическата връзка между елементите

- XSLT технология, осъществяваща трансформацията на XML документа

```

1 <?xml version="1.0"?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:fo="http://www.w3.org/1999/XSL/Format">
3   <xsl:output method="xml" version="1.0" indent="yes"/>
4   <xsl:template match="/">
5     <fo:root>
6
7       <fo:layout-master-set>
8         <fo:simple-page-master page-height="297mm" page-width="210mm" margin="5mm 25mm 5mm 25mm" master-name="template">
9           <fo:region-body margin="20mm 0mm 20mm 0mm"/>
10        </fo:simple-page-master>
11      </fo:layout-master-set>
12
13      <!--===== Title =====>
14
15      <fo:page-sequence master-reference="template">
16        <fo:flow flow-name="xsl-region-body">
17          <fo:block position="absolute" text-align="center" font-weight="bold" margin-bottom="50mm"
18            font-family="Calibri" font-size="48pt" padding-before="50mm" color="black">
19            Hotel catalog
20          </fo:block>
21        </fo:flow>
22      </fo:page-sequence>
23

```

- XSL файлът започва с оказване на използваната версия на XSL и добавяне на XML и XSL namespaces. Чрез xsl:template указваме кои са елементите, които ще бъдат обхванати от обработката. В конкретния случай това ще бъде целият входен XML документ, понеже match="/" указва root възела на XML. След това задаваме конкретна структура на всяка от страниците на изходния PDF документ, като указваме съответно ширина, височина на страницата, както и отстояние от всеки край.
- Чрез fo:page-sequence указваме нова страница в PDF документа. Първата страница представлява заглавието на каталога "Hotel Catalog", като в самия block форматираме визията на самия текст като: размер, шрифт, отстояние от краищата, цвят и др.

```

24 <!--===== Hotels =====>
25
26 <xsl:for-each select="//hotel">
27   <xsl:variable name="hotelChain_reference_id" select="@hotelChainID"/>
28   <xsl:variable name="region_reference_id" select="@regionID"/>
29   <fo:page-sequence master-reference="template">
30     <fo:flow flow-name="xsl-region-body">
31       <fo:block-container position="absolute" top="-2.5cm" left="-2.5cm">
32
33         <fo:block position="absolute" text-align="center" padding-before="2cm" margin-left="2.5cm"
34           font-family="Calibri" font-size="20pt" color="black">
35           <xsl:value-of select="name"/>
36         </fo:block>
37
38         <xsl:for-each select="photos/photo">
39           <fo:block position="absolute" text-align="center" padding-before="7mm" margin-left="2.5cm">
40             <fo:external-graphic src="{unparsed-entity-uri(@src)}"/>
41           </fo:block>
42         </xsl:for-each>
43
44         <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
45           padding-before="5mm" margin-left="2cm">
46           Description:
47           <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
48             <xsl:value-of select="description"/>
49           </fo:inline>
50         </fo:block>
51       </fo:block-container>
52     </fo:flow>
53   </fo:page-sequence>
54

```

- Следва трансформацията на самите хотели, като поради еднаквата им структура използваме xsl:for-each, за да обработим всички елементи hotel по един и същи начин. Чрез XPATH изразът "//hotel" match-ваме всички елементи с това име. Създаваме променливи, съдържащи стойностите на атрибутите hotelChainID и regionID чрез xsl:variable. За всеки един хотел, на нова страница, се извежда информация за неговото име, описание и снимки, където името и описанието отново са форматираны по: размер на текст, шрифт, цвят, отстояние от краищата, дебелина на шрифта и тн. Чрез xsl:value-of взимаме стойността на оказания елемент. Чрез fo:external-graphic добавяме снимките, тъй като те са в двоичен формат и не трябва да се обработват, подаваме път до тях чрез "unparsed-entity-uri".

```

56 <fo:page-sequence master-reference="template">
57 <fo:flow flow-name="xsl-region-body">
58 <fo:block-container position="absolute" top="-2.5cm" left="-2.5cm">
59
60 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
61 padding-before="20mm" margin-left="2cm">
62 Region:
63 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
64 <xsl:value-of select="//region[@regionID = $region_reference_id]/name" />
65 </fo:inline>
66 </fo:block>
67
68 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
69 padding-before="5mm" margin-left="2cm">
70 Address:
71 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
72 <xsl:value-of select="address"/>
73 </fo:inline>
74 </fo:block>
75
76 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
77 padding-before="5mm" margin-left="2cm">
78 Hotel Chain:
79 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
80 <xsl:value-of select="//hotelChain[@hotelChainID = $hotelChain_reference_id]/name" />
81 </fo:inline>
82 </fo:block>
83
84 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
85 padding-before="5mm" margin-left="2cm">
86 Hotel type:
87 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
88 <xsl:value-of select="@type"/>
89 </fo:inline>
90 </fo:block>
91
92 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
93 padding-before="5mm" margin-left="2cm">
94 Stars:
95 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
96 <xsl:value-of select="@stars"/>
97 </fo:inline>
98 </fo:block>
99
100 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
101 padding-before="5mm" margin-left="2cm">
102 Features:
103 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
104 <xsl:value-of select="features"/>
105 </fo:inline>
106 </fo:block>
107
108 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
109 padding-before="5mm" margin-left="2cm">
110 Rooms:
111 </fo:block>
112
113 <fo:table>
114 <fo:table-header>
115 <fo:table-row>
116 <fo:table-cell>
117 <fo:block position="relative" font-family="Calibri" font-weight="500" font-size="13pt" color="black"
118 padding-before="5mm" margin-left="2cm">
119 Type
120 </fo:block>
121 </fo:table-cell>
122 <fo:table-cell>
123 <fo:block position="relative" font-family="Calibri" font-weight="500" font-size="13pt" color="black"
124 padding-before="5mm" margin-left="2cm">
125 Capacity
126 </fo:block>
127 </fo:table-cell>
128 <fo:table-cell>
129 <fo:block position="relative" font-family="Calibri" font-weight="500" font-size="13pt" color="black"
130 padding-before="5mm" margin-left="2cm">
131 Price
132 </fo:block>
133 </fo:table-cell>
134 </fo:table-row>
135 </fo:table-header>
136
137 <xsl:for-each select="rooms/room">
138 <fo:table-body>
139 <fo:table-row>
140 <fo:table-cell>
141 <fo:block position="relative" padding-before="5mm" margin-left="2cm" font-weight="normal" font-size="12pt">
142 <xsl:value-of select="@type" />
143 </fo:block>
144 </fo:table-cell>
145 <fo:table-cell>
146 <fo:block position="relative" padding-before="5mm" margin-left="2cm" font-weight="normal" font-size="12pt">
147 <xsl:value-of select="capacity" />
148 </fo:block>
149 </fo:table-cell>
150 <fo:table-cell>
151 <fo:block position="relative" padding-before="5mm" margin-left="2cm" font-weight="normal" font-size="12pt">
152 <xsl:value-of select="price" />
153 </fo:block>
154 </fo:table-cell>
155 </fo:table-row>
156 </fo:table-body>
157 </xsl:for-each>
158
159 </fo:table>
160
161 </fo:block-container>
162 </fo:flow>
163 </fo:page-sequence>
164 </xsl:for-each>
165

```

- След това на нова страница, за всеки matched хотел извеждаме, форматирайки, неговия регион, адрес, верига хотели, тип, брой звезди и характеристики. За веригата хотели и регионът използваме ХРАТН израз, който match-ва съответната верига хотел/регион с оказаното ID, което извличаме от променливите, въведени по-горе. Информацията за стаите се представя във вид на таблица чрез fo:table , където стойностите на колоните са съответно: Тип, капацитет и цена. Всеки един ред се състои от съдържанието на съответните поделемени и атрибути на елемента room, който match-ваме чрез ХРАТН израза “rooms/room”, който е релативен на “//hotel”

```

166 <!--===== Hotel Chains =====>
167 <fo:page-sequence master-reference="template">
168 <fo:flow flow-name="xsl-region-body">
169 <fo:block-container position="absolute" top="-2.5cm" left="-2.5cm">
170
171 <fo:block position="absolute" text-align="center" padding-before="2cm" margin-left="2.5cm"
172 font-family="Calibri" font-size="20pt" color="black">
173 Hotel Chains
174 </fo:block>
175
176 <xsl:for-each select="//hotelChain">
177 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
178 padding-before="10mm" margin-left="2cm">
179
180 Name:
181 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
182 <xsl:value-of select="name"/>
183 </fo:inline>
184 </fo:block>
185
186 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
187 padding-before="5mm" margin-left="2cm">
188 Description:
189 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
190 <xsl:value-of select="description"/>
191 </fo:inline>
192 </fo:block>
193
194 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
195 padding-before="5mm" margin-left="2cm">
196 Number of hotels:
197 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
198 <xsl:value-of select="numberOfHotels"/>
199 </fo:inline>
200 </fo:block>
201
202 </xsl:for-each>
203 </fo:block-container>
204 </fo:flow>
205 </fo:page-sequence>
206

```

- Следва трансформацията на веригите хотели, където отново използваме xsl:for-each за да извършим трансформацията за всички елементи със името hotelChain, поради общата им структура. За всяка верига хотели извеждаме следната информация във форматиран вид – Име, описание и брой хотели на веригата.

```

207 <!--===== Regions =====>
208 <xsl:for-each select="//region">
209 <fo:page-sequence master-reference="template">
210 <fo:flow flow-name="xsl-region-body">
211 <fo:block-container position="absolute" top="-2.5cm" left="-2.5cm">
212
213 <xsl:if test="position() = 1">
214 <fo:block position="absolute" text-align="center" padding-before="2cm" margin-left="2.5cm"
215 font-family="Calibri" font-size="20pt" color="black">
216 Regions
217 </fo:block>
218 </xsl:if>
219
220 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
221 padding-before="20mm" margin-left="2cm">
222 Name:
223 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
224 <xsl:value-of select="name"/>
225 </fo:inline>
226 </fo:block>
227
228 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
229 padding-before="5mm" margin-left="2cm">
230 Description:
231 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
232 <xsl:value-of select="description"/>
233 </fo:inline>
234 </fo:block>
235
236 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
237 padding-before="5mm" margin-left="2cm">
238 Sights:
239 <fo:inline position="relative" padding-left="2mm" font-weight="normal" font-size="14pt">
240 <xsl:value-of select="sights"/>
241 </fo:inline>
242 </fo:block>
243
244 <fo:block position="relative" font-family="Calibri" font-weight="bold" font-size="14pt" color="black"
245 padding-before="5mm" margin-left="2cm">
246 Transport:
247 </fo:block>
248
249

```

```

250 </fo:table>
251 <fo:table-header>
252 <fo:table-row>
253 <fo:table-cell>
254 <fo:block position="relative" font-family="Calibri" font-weight="500" font-size="13pt" color="black"
255 padding-before="5mm" margin-left="2cm">
256 Type
257 </fo:block>
258 </fo:table-cell>
259 <fo:table-cell>
260 <fo:block position="relative" font-family="Calibri" font-weight="500" font-size="13pt" color="black"
261 padding-before="5mm" margin-left="2cm">
262 Name
263 </fo:block>
264 </fo:table-cell>
265 <fo:table-cell>
266 <fo:block position="relative" font-family="Calibri" font-weight="500" font-size="13pt" color="black"
267 padding-before="5mm" margin-left="2cm">
268 Hyperlink
269 </fo:block>
270 </fo:table-cell>
271 </fo:table-row>
272 </fo:table-header>
273
274 <xsl:for-each select="transports/transport">
275 <fo:table-body>
276 <fo:table-row>
277 <fo:table-cell>
278 <fo:block position="relative" padding-before="5mm" margin-left="2cm" font-weight="normal" font-size="12pt">
279 <xsl:value-of select="@type" />
280 </fo:block>
281 </fo:table-cell>
282 <fo:table-cell>
283 <fo:block position="relative" padding-before="5mm" margin-left="2cm" font-weight="normal" font-size="12pt">
284 <xsl:value-of select="name" />
285 </fo:block>
286 </fo:table-cell>
287 <fo:table-cell>
288 <fo:block position="relative" padding-before="5mm" margin-left="2cm" font-weight="normal" font-size="12pt">
289 <xsl:value-of select="link" />
290 </fo:block>
291 </fo:table-cell>
292 </fo:table-row>
293 </fo:table-body>
294 </xsl:for-each>
295 </fo:table>
296
297 </fo:block-container>
298 </fo:flow>
299 </fo:page-sequence>
300 </xsl:for-each>
301
302 </fo:root>
303
304 </xsl:template>
305
306 </xsl:stylesheet>

```

- Последната част от трансформацията е на регионите. Аналогично използваме `xsl:for-each` с XPath израз `"//region"`, за `match`-ването на всички елементи `region`. Информацията за всеки един регион се извежда на отделна страница в резултатния PDF файл. Тъй като искаме заглавния ред с името на ресурсите ("Regions") да се изведе само веднъж, използваме `<xsl:if>` с условие `position() = 1`, чрез което текста се извежда само при първия `match`. След това последователно извеждаме форматирана информация за поделементите на `region` – съответно име, описание, забележителности в региона, и транспорт. Транспортът се трансформира във вид на таблица, където колоните са съответно – тип, име и линк за връзка към сайта. Всеки един ред се състои от съдържанието на съответните поделементи и атрибути на елемента `transport`, което взимаме чрез `xsl:value-of`. Елемента `transport` `match`-ваме чрез XPath израза `"transports/transport"`, който е релативен на `"//region"`

4 Тестване

Тестването на проекта е извършено спрямо комплекс от тестове. Те включват валидация на входният XML документ чрез DTD схема, проверка на връзките между елементите, и също генериране на PDF файл чрез XSLT документ.

За изготвяне на XML, DTD и XSLT документите използвахме инструмента за разработка Visual Studio Code, като така потвърдихме създаването на правилни, валидни документи. Използваните браузъри за тестване са Chrome и Firefox, които предоставят лесно отваряне на XML, DTD и XSLT документи. Така подсигурихме още веднъж валидността на документите.

Тестовите за валидация потвърдиха, че системата е устойчива на грешки и успешно разпознава невалидни входни данни. Тестовите за генериране на PDF потвърдиха, че трансформацията се извършва правилно и създава документи по дадения формат. Обобщено, тестването потвърждава валидността на XML документа и коректността на връзките, осигурявайки стабилно функционираща система.

- Валидиране чрез DTD схема:

Валидността на XML документа, покриващ различните сценарии за хотели в някои региони на България, е проверена спрямо DTD схема. Тя проверява, дали XML документа спазва изградената стабилна схема на структурата, използваща различни комбинации от характеристики и свързаност между елементите. В тестването се извърши срещу коректни и некоректни елементи, за да подсигурим успешното разпознаване и обработване на грешки от системата. Връзките между елементите (хотел – верига от хотели, хотел – регион) също са тествани, като са проверени атрибутите ID/IDREF, за да се уверим, че системата правилно намира и свързва елементите в XML структурата. Този процес включва сценарии с различни стойности на атрибутите за пълно тестване срещу DTD схемата.

Изход на валидиране на крайният XML документ и на грешен XML документ чрез DTD схемата през XML валидаторът <https://www.xmlvalidation.com> :

No errors were found


The following files have been uploaded so far:

[XML document:](#) 

[hotelCatalog.dtd](#) 

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

4 errors have been found!

Click on  to jump to the error. In the document, you can point at  with your mouse to see the error message.

Errors in the XML document:

-  115: 11 The content of element type "hotel" must match "(name,address,description,photos,features,rooms)".
-  175: 10 The content of element type "hotel" must match "(name,address,description,photos,features,rooms)".
-  259: 15 The content of element type "hotelChain" is incomplete, it must match "(name,description,numberOfHotels)".
-  385: 11 The content of element type "region" must match "(name,description,sights,transports)".

- Генериране на PDF документ:

Изпълнени са и тестове за генериране на PDF документ, посредством XSLT трансформации и Java код, като така проверихме дали визуалните изходни данни отговарят на очаквания формат, както и дали изходната информация в PDF документа е консистентна с входните данни от XML документа.

- За преобразуването на документа в PDF формат използваме външната библиотека Apache FOP. За изтегляне на библиотеката използваме Maven, като информация за група и версия е налична във файла pom.xml, чрез който сваляме нужните зависимости.

```
<dependencies>
  <dependency>
    <groupId>org.apache.xmlgraphics</groupId>
    <artifactId>fop</artifactId>
    <version>2.7</version>
  </dependency>
</dependencies>
```

Java код за трансформация на XML файл чрез XSLT файл:

```
1 import org.apache.fop.apps.FOException;
2 import org.apache.fop.apps.FOUserAgent;
3 import org.apache.fop.apps.Fop;
4 import org.apache.fop.apps.FopFactory;
5 import org.apache.fop.apps.MimeConstants;
6
7 import java.io.*;
8 import java.nio.file.Files;
9 import java.nio.file.Paths;
10 import javax.xml.transform.*;
11 import javax.xml.transform.sax.SAXResult;
12 import javax.xml.transform.stream.StreamSource;
13
14 public class XmlToPdf {
15     public static void main(String... args) {
16         XmlToPdf fopPdfDemo = new XmlToPdf();
17         try {
18             fopPdfDemo.convertToPDF();
19         } catch (FOException | IOException | TransformerException e) {
20             e.printStackTrace();
21         }
22     }
23
24     public void convertToPDF() throws IOException, FOException, TransformerException {
25         // The XSL FO file
26         File xsltFile = new File("./src/main/resources/hotelCatalog.xsl");
27         // The XML file
28         StreamSource xmlSource = new StreamSource(new File("./src/main/resources/hotelCatalog.xml"));
29
30         FopFactory fopFactory = FopFactory.newInstance(new File(".").toURI());
31         FOUserAgent foUserAgent = fopFactory.newFOUserAgent();
32
33         // The output file
34         OutputStream out;
35         out = Files.newOutputStream(Paths.get("hotelCatalog.pdf"));
36
37         try {
38             Fop fop = fopFactory.newFop(MimeConstants.MIME_PDF, foUserAgent, out);
39
40             TransformerFactory factory = TransformerFactory.newInstance();
41             Transformer transformer = factory.newTransformer(new StreamSource(xsltFile));
42
43             Result res = new SAXResult(fop.getDefaultHandler());
44
45             transformer.transform(xmlSource, res);
46         } finally {
47             out.close();
48         }
49     }
50 }
51
```

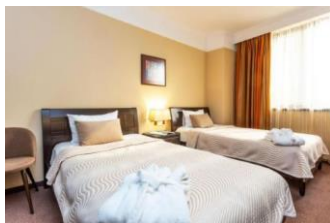
- В main функцията извикваме метода, чрез който се извършва трансформацията и обработваме евентуални грешки, свързани с несъществуващи входни файлове или неправилна конфигурация.
- В convertToPdf метода създаваме референции към входните файлове, които са необходими за трансформацията, съответно XML и XSL, като подаваме абсолютен път до тях. След това създаваме FopFactory, който представлява обект, съдържащ конфигурационни данни, необходими за трансформацията и FoUserAgent, чрез който се осъществява връзката с XSL-FO, необходима за форматирането на документа. Създаваме референция към изходния PDF файл, подавайки абсолютен път до него.
- Използвайки методите на FopFactory, създаваме инстанция на Fop, подавайки желаня формат (в случая PDF) и необходимите параметри. Създаваме инстанция на Transformer, която се грижи за процеса на трансформация, използвайки указанията в XSL файла. Трансформацията започва с извикването на метода transform, като се подава входния XML файл и изходния PDF файл, в който се записва форматирания резултат.

- След изпълнение на програмата, в конзолата се извежда текст, показващ успешното изобразяване на всяка една от страниците.

```
INFO: Rendered page #1.
дек 20, 2023 2:35:52 СЛ.0Б. org.apache.fop.events.LoggingEventListener processEvent
```

Така се създава крайния PDF документ, на посочения от нас път. Части от крайния PDF документ с показани страници за един хотел, веригите от хотели и един регион:

Festa Sofia



Description: Located 200 meters from Mall Bulgaria and the tram stop, Festa Sofia hotel offers elegant rooms with free Wi-Fi and satellite TV. It features a wellness center and beauty salon, as well as 2 restaurants, 3 bars, and 4 conference halls. The spa center includes a sauna and steam bath. Guests can stay fit in the fitness center or enjoy massage and aromatherapy treatments. There is also a hair salon on site. Restaurant Festa serves local and international cuisine in a secluded atmosphere. The Mezzanine bar offers sophisticated drinks and desserts. The lobby bar serves a variety of coffee specialties. Sahara Hookah Bar offers its guests exquisite varieties of tobacco and a wonderful selection of drinks. All rooms at Festa Sofia hotel have a panoramic view of the city or Vitosha. They are equipped with air conditioning and a safe. The private bathrooms have a bathtub. The hotel is located in a modern business area, 18 km from Sofia International Airport and 5 km from the city center.

Region: Sofia

Address: Bulgaria Blvd. 83, 1404 Manastirski Livadi, Sofia

Hotel Chain: Festa Hotels

Hotel type: Conference

Stars: 4

Features: Private Parking, Free WiFi, Airport Shuttle, Fitness Center, Family Rooms, Non-Smoking Rooms, 2 Restaurants, 24-Hour Reception, Bar, Good Breakfast

Rooms:

Type	Capacity	Price
single	1	120
double	2	150
suite	3	200

Hotel Chains

Name:

Victoria Group Hotels and Resorts

Description: Their portfolio consists of 11 hotels categorized as 3, 4, and 5 stars, over 500 holiday villas, and multifunctional halls located in various destinations. Their hotels impress with unconventional and unique designs. Their advantage lies in their locations—by the sea, in the center of Sofia, or near some of the best ski slopes! Their vacation hotels offer All-Inclusive accommodation with included meals, drinks, and services, ensuring a worry-free vacation! The Victoria Group Hotels and Resorts portfolio includes some of the most elegant and luxurious wedding venues in Bulgaria — from the heavenly beaches of Elenite to the city lights of Sofia.

Number of hotels: 10

Name:

BT Collection

Description: BT Collection is a collection of luxury hotels and complexes situated in some of the most beautiful places in Bulgaria—Black Sea Coast, Borovets, and Arbanasi. The hotels combine an elegant ambience, modern design, luxury, and high-quality service, symbolizing tourism in Bulgaria. The BT Collection offers SPA facilities, relaxation, delicious food, and entertainment, suitable for family vacations as well as business trips and conferences.

Number of hotels: 7

Name:

Festa Hotels

Description: Festa Hotels is a hotel chain established in 2003, known for its consistently high service quality and personalized approach to every guest. Thanks to their attractive locations, the chain's hotels offer various options for summer seaside relaxation, winter ski holidays, business travel, and a wide range of rejuvenating SPA and balneotherapy procedures. For sea vacation enthusiasts, Festa's collection includes three hotels in two of the most popular destinations on the Bulgarian Black Sea—Nessebar and Pomorie. Guests can also enjoy mountain ski vacations at their hotels in Borovets, as well as urban business trips with the offerings of Festa Sofia Hotel, located in a tranquil spot in the capital.

Number of hotels: 7

Regions

Name: Sofia

Description: Sofia is the capital and the largest city in Bulgaria. The population is around 1.3 million people. Sofia is located in the central part of Western Bulgaria. It serves as the main administrative, industrial, transportation, cultural, and educational center of the country. The transportation system in the capital is highly developed. In Sofia, you can find: - Numerous educational institutions, including the oldest and largest higher education institution in Bulgaria - Sofia University "St. Kliment Ohridski," as well as the Bulgarian Academy of Sciences and the National Library "Sts. Cyril and Methodius." - Temples of major world religions, with around 200 functioning Orthodox churches and chapels and over 40 monasteries, one of which is the memorial temple "Alexander Nevsky." - Theatres and opera houses, including the Ivan Vazov National Theatre and the Sofia Opera and Ballet. - Museums and galleries such as the National Historical Museum, the National Ethnographic Museum, and the National Art Gallery. - Monuments from various historical periods. - Concert halls, the largest of which is Arena Sofia with up to 15,000 seats. - Vasil Levski Stadium, the largest stadium in the country with a capacity of 44,000 people. - Other notable places include the National Palace of Culture, the two oldest and well-known bridges in Sofia - the Eagle Bridge and the Lion Bridge, the largest park in Sofia - Borisova Gradina, and the Sofia Zoo.

Sights: Sofia University "St. Kliment Ohridski", Bulgarian Academy of Sciences, National Library "Sts. Cyril and Methodius", memorial temple "Alexander Nevsky", National Theatre "Ivan Vazov", Sofia Opera and Ballet, National Historical Museum, National Ethnographic Museum, National Art Gallery, Arena Sofia, Vasil Levski Stadium, National Palace of Culture, Eagle Bridge, Lion Bridge, Borisova Gradina, Sofia Zoo

Transport:

Type	Name	Hyperlink
trainStation	Central Railway Station Sofia	https://www.centralnaavtogara.bg
busStation	Central Bus Station Sofia	https://www.centralnaavtogara.bg
airport	Sofia Airport	https://www.sofiaairport.bg
publicTransport	Urban Mobility Center	https://www.sofiatraffic.bg

5 Заключение и възможно бъдещо развитие

Представения проект успешно покрива задачата за изграждане на каталог на хотели. Данните, използвани в каталога са подбрани, така че да са актуални, полезни и привлекателни за потребителите. Подробно се представя информацията за хотелите, предоставяйки на потребителите да разгледат снимки, характеристики, цени и други ресурси за хотела.

XML структурата осигурява ясна и лесно разбираема представа за хотелите. Връзките между елементите (хотел – верига от хотели, хотел – регион) улесняват навигацията в каталога за по-добра възможност за ползване. Графичното съдържание прави каталога по-интересен и същевременно по-полезен за туристите. Използваните технологии правят добавянето на нови характеристики лесно, без промени в основната структура на системата.

Този проект е едно първоначално изграждане на каталог за хотели. Той има ясна структура и това позволява лесно разширение. Може да се подобри с нови функционалности, използвайки различни технологии, както и може да се добавят допълнителни слоеве от информация. Предвидени възможности за бъдещо развитие са:

- Интеграция към външни сайтове за създаване на резервации за хотелите и на забележителностите, като така туристите ще могат лесно да си запазват места в хотел, или да се възползват от информация за местата, от които се интересуват
- Допълнително обвързване на хотелите със забележителностите и с видовете транспорт в съответния регион, посочвайки по-близките локации до хотела, което ще предостави удобство при избора на хотел спрямо нуждите на потребителя
- Изработване на по-атрактивен интерфейс на каталога за по-добро усещане при употреба от потребителите
- Добавяне на повече графично и мултимедийно съдържание, като аудио и видео ресурси, за по-добро усещане при употреба от потребителите
- Редовно проверяване на това до колко актуални са данните в каталога, и съответно обновяване на тези данни при необходимост
- Оптимизация на управлението на данни чрез използването на бази данни за по-ефективно съхранение и извличане на информацията, което би помогнало при разрастване на броя на обектите в каталога

В заключение изготвения проект е успешна реализация на поставената задача и обхваща ключовите аспекти, предоставяйки удобен и лесен за употреба каталог на хотели в България. Решението на задача е стабилно и дава възможност за ефективно управление на информацията за хотелите в различните региони на България. Каталогът се справя успешно с текущите изисквания, но може да бъде подложен на бъдещо развитие и оптимизация.

6 Разпределение на работата

Работа	Мария Иванова	Николай Костандиев
Изработка на XML файл със съдържание на седем хотела, три вериги на хотели и три региона	Секции за: <ul style="list-style-type: none"> два хотела (от верига „Феста“) трите региона 	Секции за: <ul style="list-style-type: none"> пет хотела (от вериги „Виктория“ и „ВТ“) трите вериги на хотели
Изработка DTD файл	Да	Не
Изработка XSL файл	Не	Да
Изготвяне на Java код за преобразуване на XSL файла в PDF файл	Не	Да
Изготвяне на документацията	Да	Не

7 Използвани литературни източници и Уеб сайтове

1. Уеб сайт на верига от хотели „Феста“: <https://festahotels.com>
2. Уеб сайт на верига от хотели „Виктория“: <http://www.btcollection.com>
3. Уеб сайт на верига от хотели „ВТ“: <https://festahotels.com>
4. Уеб сайт за резервации „Буукинг“ за актуална информация за хотелите: <https://www.booking.com>
5. Уеб сайт на Уикипедия за информация за регион София: <https://bg.wikipedia.org/wiki/София>
6. Уеб сайт на Уикипедия за информация за регион Бургас: <https://bg.wikipedia.org/wiki/Бургас> и [https://bg.wikipedia.org/wiki/Бургас_\(община\)](https://bg.wikipedia.org/wiki/Бургас_(община))
7. Уеб сайт на Уикипедия за информация за регион Боровец: <https://bg.wikipedia.org/wiki/Боровец>
8. Уеб сайт на община Бургас: <https://www.burgas.bg>
9. Уеб сайт на град Боровец: <https://www.borovets-bg.com>
10. Уеб сайт на Централна гара София и Централна автогара София: <https://www.centralnaavtogara.bg>
11. Уеб сайт на Летище София: <https://www.sofiaairport.bg>
12. Уеб сайт на Център за градска мобилност на София: <https://www.sofiatraffic.bg>
13. Уеб сайт на БДЖ за информация за влаковия транспорт в Бургас: <https://www.bdz.bg>
14. Уеб сайт на Автогара Юг и БургасБус в Бургас: <https://burgasbus.info>
15. Уеб сайт на Летище Бургас: <https://burgas-airport.bg>
16. Уеб сайт на Автогара Самоков: <http://avtogara-samokov.eu>
17. xsl:fo documentation: <https://w3schools.sinsixx.com/xslfo/default.asp.htm>
18. Apache FOP documentation: <https://xmlgraphics.apache.org/fop>