|  | **Софийски университет „Св. Кл. Охридски”**  Факултет по математика и информатика  *Бакалавърска програма „Софтуерно инженерство”* |  |
| --- | --- | --- |

**Предмет: XML технологии за семантичен Уеб**

***Зимен семестър, 2021/2022 год.***

**Тема №44: Каталог на заложна къща**

**Курсов проект**

*Автори:*

*Айше Джинджи, фак. номер 62470*

*Ангел Марински, фак. номер 62469*

*Александър Стоилов, фак. номер 62382*

януари, 2022 г.

София

# Съдържание

[**Съдържание**](#_y8urslccgycz) **2**

[**Въведение**](#_v8ct7n5muc6a) **3**

[Анализ на решението](#_1fob9te) **3**

[Работен процес](#_rbfceqj4x4a4) 3

[Структура на съдържанието](#_mp1nqpooos2a) 4

[Тип и представяне на съдържанието](#_q3ijoxyghj7g) 8

[**Дизайн**](#_9uc5nh8pzxft) **9**

[**Тестване**](#_h5p198e71okp) **13**

[**Заключение и възможно бъдещо развитие**](#_u4cb4r2pk0me) **14**

[**Разпределение на работата**](#_spi2l79i4cur) **15**

[**Използвани литературни източници и Уеб сайтове**](#_mkppdjit2nl7) **15**

# **Въведение**

Целта на настоящето задание, разработено от горепосочения екип, бе следната:

● да бъде създаден качествено издържан каталог на заложна къща

● съответния да поддържа информация относно заложените ресурси в същата (впоследствие ще се запознаем със специфично установената от нас класификация на ресурсите, следствие на групирането им по общи признаци)

● да поддържа информация относно обстоятелствата, описващи въпросния заложен ресурс в къщата (приел залога, профил на залагащия ресурса, срок на залагане, времеви момент на създаване на залога в системата, време на последно модифициране и т.н.)

● да се предостави възможност за автоматично управление на обмяната на информационни ресурси в заложната къща

● да улесни контрола на бизнес процеса в съответната

Заданието е реализирано с помощта на XML документи с текстово и графично съдържание, описващо възможните характеристики на всеки един заложен ресурс. Графичното съдържание в каталога е представено чрез XML единици (Entities). Връзките между различните категории и ресурсите, както и между ресурсите и залозите, са осъществени чрез атрибутите ID/IDREF. В XML документа е представено описанието на 12 ресурса (по 2 от всяка от въпросните класификации, на които сме разбили типизирането на ресурсите). Съдържанието на документа е валидирано чрез обособена за целта DTD граматика, след което е представено в PDF документ посредством XSLT трансформации (и по специално xsl-fo). В текущия документ ще се запознаем подробно с цялостната картина, обграждаща процеса на изграждането на каталога - ще обсъдим първоначалните възгледи и идеи относно реализацията и функционалността, ще преминем през фазите на развитие и като за финал ще поговорим за бъдещия потенциал на платформата.

# **Анализ на решението**

## **Работен процес**

Входното съдържание, описващо проекта, е представено под формата на XML файл, съдържащ информация относно заложната къща. Информацията е структурирана по следния начин:   
Първо бива установена класификацията на категориите, по които ще разглеждаме ресурсите. След това, за всяка класификация биват упоменати произволен брой елементи, с прилежащите си IDREF номера. Въпросните елементи представляват референции към самите сектори с ресурси, принадлежащи към този момент на заложната къща. В ресурсите биват реферирани отделни елементи, описващи информацията по самото залагане на съответния в къщата, отново чрез ID - IDREF връзка.  
В допълнение, валидирането на структурата и съдържанието на XML документа е осъществено чрез специално създадена за целта DTD граматика. Целият процес по разработка на проекта е извършен в редактора VSCode, който предлага специални разширения, валидиращи съдържанието на документа по указаната граматика.   
Генерирането на PDF документа е осъществено с помощта на XSL документ, като за създаването на крайния документ е използван редактора Oxygen, а за улеснение на работния процес - онлайн трансформатора .NET XSLT Fiddle [1].

## **Структура на съдържанието**

XML документ */ pawnshop.xml /*

1. **pawnshop** - кореновият елемент. Той съдържа елементите **classification**, **resources** и **pawns**. Спрямо DTD схемата, тези елементи следва да бъдат принадлежащи в pawnshop точно веднъж. Асоциира се със цялостния каталог в къщата.
2. **classification** - пряк наследник на **pawnshop** елемента. Съдържа елементите **jewellery**, **electronics**, **antiques**, **art**, **precious\_metals** и **vehicles**. Спрямо DTD схемата, тези елементи следва да бъдат принадлежащи в classification точно веднъж. Представлява отделните категории/класификации, в които може да попада всеки от ресурсите.
   1. **jewellery** - пряк наследник на **classification** елемента. Съдържа елементи **resource\_ref**. Спрямо DTD схемата, въпросните поделементи **resource\_ref** следва да бъдат съдържани произволен брой пъти. Представлява класификацията “Бижута”, към която могат да се определят елементите.
      1. **resource\_ref** - пряк наследник на гореизброените елементи. Съдържа атрибут с име **ref**. Според DTD схемата би следвало да е налично следното типизиране на съдържанието на **resource\_ref**: той се състои от текстово съдържание, атрибута **ref** е от тип IDREF и е задължителен. Представлява връзката към всички ресурси в къщата, принадлежащи към съответната класификация (в този случай - “Бижута”)
   2. **electronics** - пряк наследник на **classification** елемента. Съдържа елементи **resource\_ref**. Спрямо DTD схемата, въпросните поделементи **resource\_ref** следва да бъдат съдържани произволен брой пъти. Представлява категорията “Електроника”.
   3. **antiques** - пряк наследник на **classification** елемента. Съдържа елементи **resource\_ref**. Спрямо DTD схемата, въпросните поделементи **resource\_ref** следва да бъдат съдържани произволен брой пъти. Представлява категорията “Антики”.
   4. **art** - пряк наследник на **classification** елемента. Съдържа елементи **resource\_ref**. Спрямо DTD схемата, въпросните поделементи **resource\_ref** следва да бъдат съдържани произволен брой пъти. Представлява категорията “Изкуство”.
   5. **precious\_metals** - пряк наследник на **classification** елемента. Съдържа елементи **resource\_ref**. Спрямо DTD схемата, въпросните поделементи **resource\_ref** следва да бъдат съдържани произволен брой пъти. Представлява категорията “Ценни метали”.
   6. **vehicles** - пряк наследник на **classification** елемента. Съдържа елементи **resource\_ref**. Спрямо DTD схемата, въпросните поделементи **resource\_ref** следва да бъдат съдържани произволен брой пъти. Представлява категорията “Превозни средства”.
3. **resources** - пряк наследник на **pawnshop** елемента. Съдържа елементи **resource**. Спрямо DTD схемата, въпросните поделементи **resource** следва да бъдат съдържани произволен брой пъти. Представлява всички ресурси, които се намират към текущия момент в заложната къща.
   1. **resource** - пряк наследник на **resources** елемента. Съдържа елементи **name**, **pawn\_ref**, **author**, **manufacturer**, **model**, **quantity**, **pawnshop\_evaluation**, **description** и **image**. Съдържа атрибут **id** и атрибут **classification**. Спрямо DTD схемата, въпросните поделементи следва да бъдат съдържани по следния начин: в реда, в който са изброени, name - точно веднъж, **pawn\_ref** - точно веднъж, **author** или **manufacturer** - точно едното от двете, незадължителен поделемент - може да се среща най-много веднъж, **model** - незадължителен поделемент - може да се среща най-много веднъж, **quantity** - незадължителен поделемент - може да се среща най-много веднъж, **pawnshop\_evaluation** - точно веднъж, **description** - незадължителен поделемент - може да се среща най-много веднъж, **image** - точно веднъж, атрибутът **id** е задължителен и е от тип ID, атрибутът **classification** e задължителен и е с една от следните изброени стойности: {Jewellery, Electronics, Antiques, Precious\_metals, Vehicles}. Представлява отделен ресурс, заложен в къщата.
      1. **name** - пряк наследник на **resource** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява името на заложения ресурс.
      2. **pawn\_ref** - пряк наследник на **resource** елемента. Спрямо DTD схемата е празен елемент с атрибут ref от тип IDREF, който представлява връзката към залога, асоцииран със съответния продукт (чрез ID - IDREF връзка).
      3. **author** - пряк наследник на **resource** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява автора на заложения ресурс, в случай че такъв съществува (напр. при ресурси от тип “Изкуство”).
      4. **manufacturer** - пряк наследник на **resource** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява производителя на заложения ресурс, в случай че такъв съществува (напр. при ресурси от тип “Електроника”).
      5. **model** - пряк наследник на **resource** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява модел на заложения ресурс, в случай че такъв съществува (напр. при ресурси от тип “Електроника”).
      6. **quantity** - пряк наследник на **resource** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява количество на заложения ресурс, в случай че такъв съществува (напр. при ресурси от тип “Ценни метали”).
      7. **pawnshop\_evaluation** - пряк наследник на **resource** елемента. Съдържа елементи **currency** и **value**. Спрямо DTD схемата, съответните следва да бъдат съдържани в този ред, всеки от тях - точно веднъж. Представлява данни за сумата, с която заложната къща оценява съответния ресурс.
         1. **currency** - пряк наследник на **pawnshop\_evaluation** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява валутата, като част от данни за цената
         2. **value** - пряк наследник на **pawnshop\_evaluation** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява стойността на сумата, като част от данни за цената.
      8. **description** - пряк наследник на **resource** елемента. Спрямо DTD схемата съдържа текстово съдържание. Асоциира се с описание на съответния продукт.
      9. **image** - пряк наследник на **resource** елемента, не съдържа поделементи. Съдържа атрибут href. Спрямо DTD схемата, въпросния атрибут следва да бъде от тип ENTITY, задължителен. Представлява изображение към съответния ресурс.
4. **pawns** - пряк наследник на **pawnshop** елемента. Съдържа елементи **pawn**. Спрямо DTD схемата, въпросните поделементи **pawn** следва да бъдат съдържани произволен брой пъти. Представлява данни за всички залози, направени в къщата.
   1. **pawn** - пряк наследник на **pawns** елемента. Съдържа елементи **amount\_given**, **recipient\_employee**, **submitter**, **term**, **created\_at**, **updated\_at**. Съдържа атрибут **id** и атрибут **status**. Спрямо DTD схемата, въпросните поделементи и атрибути следва да бъдат съдържани по следния начин: в реда, в който са изброени, всеки от тях - точно веднъж, атрибутът **id** е задължителен и е от тип ID, атрибутът **status** e задължителен и е с една от следните изброени стойности: {Active, Pending, Settled, EMNR */ Expired; money not returned /* }. Представлява данни за един конкретен залог в къщата.
      1. **amount\_given** - пряк наследник на **pawn** елемента. Съдържа елементи **currency** и **value**. Спрямо DTD схемата, съответните следва да бъдат съдържани в този ред, всеки от тях - точно по веднъж. Представлява сумата, която залагащия е получил за съответен ресурс.
      2. **recipient\_employee** - пряк наследник на **pawn** елемента. Съдържа елементи **name, surname, employee\_id**. Спрямо DTD схемата, въпросните поделементи следва да бъдат съдържани по следния начин: в този ред на изброяване, всеки от тях - точно веднъж. Представлява данни за служителя, приел съответния залог.
         1. **name** - пряк наследник на recipient\_employee елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява име на човека, приел залога.
         2. **surname** - пряк наследник на recipient\_employee елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява фамилия на човека, приел залога.
         3. **employee\_id** - пряк наследник на recipient\_employee елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява фирмен идентификатор за служителя, приел залога.
      3. **submitter** - пряк наследник на **pawn** елемента. Съдържа елементи **name, surname, ucn, permanent\_address, phone\_number, id\_card**. Спрямо DTD схемата, въпросните поделементи следва да бъдат съдържани по следния начин: в този ред на изброяване, всеки от тях - точно веднъж. Представлява данни за човека, направил залога.
         1. **ucn** - пряк наследник на **submitter** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява ЕГН на човека, направил залога.
         2. **permanent\_address** - пряк наследник на **submitter** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява постоянен адрес на човека, направил залога.
         3. **phone\_number** - пряк наследник на **submitter** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява телефонен номер на човека, направил залога.
         4. **id\_card** - пряк наследник на **submitter** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява номер на лична карта на човека, направил залога.
      4. **term** - пряк наследник на **pawn** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява срок на залога.
      5. **created\_at** - пряк наследник на **pawn** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява дата на поставяне на залога.
      6. **updated\_at** - пряк наследник на **pawn** елемента. Спрямо DTD схемата съдържа текстово съдържание (#PCDATA). Представлява дата на последна промяна на залога.

## **Тип и представяне на съдържанието**

Съдържанието на проекта ни е представено под фóрмата на файл с PDF разширение, визуализиращо установената структуризация на заложната къща и представящо концепцията, която екипът, разработил текущия проект, определи за целева. Състои се от съвкупността от текстови и графични елементи. Текстът, включен в документите, е създаден изцяло от екипа по проекта, и за надлежния не биха могли да бъдат посочени източници, различни от него. Графичното съдържание, използвано в проекта, представлява съчетанието от дванадесет (12) на брой изображения. За източник на въпросните се посочват Google Images. 11 от изображенията са във формат JPG, и едно от тях (това на заглавната страница) - във формат PNG. Горепосоченият PDF файл, визуализиращ изходните данни от процеса на оперативност, притежава следната структура:

* Заглавна страница
  + Използван е ресурса pawnshop.png в директория с абсолютен път /images/pawnshop.png.
* Категории
  + Създадени с идеята да помагат на служителите за по-лесното боравене със системата и намирането на ресурси и прилежащите им залози.
  + Биват разграфени в следната класификация: Бижута (**Jewellery**), Електроника (**Electronics**), Антики (**Antiques**), Изкуство (**Art**), Ценни метали (**Precious Metals**) и Превозни средства (**Vehicles**).

Следва подробно представена информацията за всеки от артикулите, принадлежащи към къщата в текущия времеви момент, разпределени по категории.

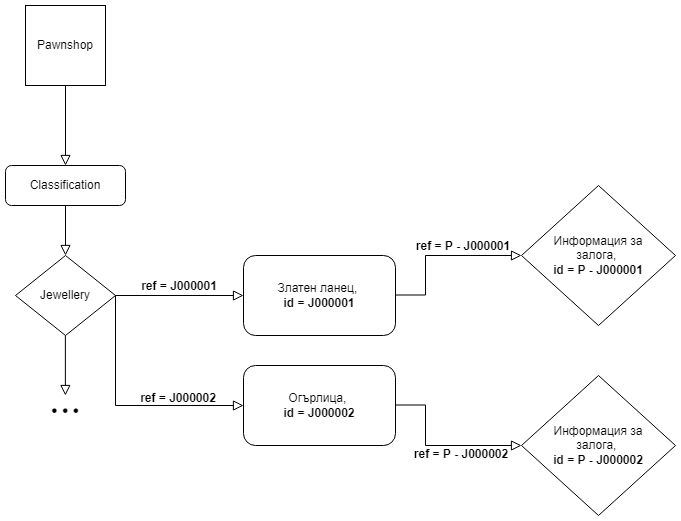
* **Бижута**
  + *Златен ланец*. Използван е ресурса gold\_chain.jpg в директория с абсолютен път /images/gold\_chain.jpg.
  + *Златна огърлица*. Използван е ресурса necklace.jpg в директория с абсолютен път /images/necklace.jpg.
* **Електроника**
  + *Смартфон iPhone XR*. Използван е ресурса iphone\_xr.jpg в директория с абсолютен път /images/iphone\_xr.jpg.
  + *Часовник Сейко Серия 5*. Използван е ресурса watch.jpg в директория с абсолютен път /images/watch.jpg.
* **Антики**
  + *Антична шотландска ваза*. Използван е ресурса antique\_vase.jpg в директория с абсолютен път /images/antique\_vase.jpg.
  + *Античен часовник*. Използван е ресурса antique\_watch.jpg в директория с абсолютен път /images/antique\_watch.jpg.
* **Изкуство**
  + *Картина American Gothic*. Използван е ресурса american\_gothic.jpg в директория с абсолютен път /images/american\_gothic.jpg.
  + *Картина Creation of Adam*. Използван е ресурса creation\_of\_adam.jpg в директория с абсолютен път /images/creation\_of\_adam.jpg.
* **Ценни метали**
  + *Златно кюлче 10 гр*. Използван е ресурса gold\_bar.jpg в директория с абсолютен път /images/gold\_bar.jpg.
  + *Платинено ключе 10 гр*. Използван е ресурса plat\_bar.jpg в директория с абсолютен път /images/plat\_bar.jpg.
* **Превозни средства**
  + *Фолксваген Голф 4*. Използван е ресурса vw\_golf.jpg в директория с абсолютен път /images/vw\_golf.jpg.
  + *Фиат Пунто*. Използван е ресурса fiat\_punto.jpg в директория с абсолютен път /images/fiat\_punto.jpg.

# **Дизайн**

Нека първо обобщим файловата структура на проекта, с цел по-лесното запознаване със системата. В директорията *images* съхраняваме снимковия материал, необходим за илюстрирането на системата. Файлът *pawnshop.dtd* представлява образуваната DTD граматика, стандартизирана с концепцията да отговаря на бъдещи разширения и актуализации по системата. В *pawnshop.xml* можем да открием данните, въведени от служителите в системата. Разбира се, при бъдещо развитие на системата, следва да се направи подобаващ TUI/GUI, чрез който в реалния свят въпросните да въвеждат информация, без да им се налага да разбират от XML структуризация и синаксис… Благодарение на XSL файла *pawnshop.xsl* разполагаме с възможността за окончателна визуализация на задачата, а именно информацията поместена в рамките на PDF файл, с добре онагледена композиция на съдържанието. Споменатата може да бъде открита във файла с наименование *pawnshop.pdf*.

Като една реална система на заложна къща, идеята на структурата е именно това да са материали, които служителите индивидуално въвеждат, и чрез които боравят със системата. За успешното визуализиране на проекта беше необходимо използването на съвкупност от графични изображения. Начинът, по който въпросните са интегрирани в каталога, е чрез т.нар. Entities - въпросните единици установяват връзка към изображенията на база именно абсолютния път, упоменат в *т. 2.3 - Тип и представяне на съдържанието*. По този начин, в самото съдържание на XML документа реферираме единиците, и така “прикрепяме” изображения към кореспондиращите им елементи image.

Нещо, на което със сигурност трябва да се акцентира, е как се осъществява логическата връзка в XML документа. Имаме класификация, имаме залог, принадлежащ на съответната, имаме и данни за залога. *Как ги свързваме?* Посредством именно връзката *ID - IDREF*, предоставена ни от XML езика, с цел конюгацията на елементи. В случая, като опростен модел на системата, бихме могли да обрисуваме следното:



Както виждаме на изображението, начинът, по който осъществяваме връзката между конкретната класификация и принадлежащ на нея обект, е чрез установяването на id на обекта и реферирането му от нея. По същия начин се композира и връзката между ресурса и неговата информация (причината да бъдат разделени от ресурсите изначално е, че предоставят по-добър поглед на системата, отговарят на *SOLID* принципите за структуризация и предоставят лесна възможност за модификация на специфични компоненти, без това да поставя цялостната база данни в риск) - данните за залога се обособяват със специфичен id, логически свързан с този на ресурса (id на ресурс е от вида: “*<Cl><ser. num.>”,* а на кореспондиращия му залог и информация, която го обгражда: *“*P - *<Cl><ser. num.>”*), и се достъпват от ресурс по същия начин, по който той бива достъпен от класификация - чрез рефериране. С цел непреднамерено усложнение на схемата, решихме да не изобразяваме връзката между ресурс и изображение - тя е абсолютно аналогична, с тази разлика, че абсолютният път на изображението в системата е фиксиран като единица, която бива достъпена в ресурса.

Начинът, по който се извършва валидацията, е на база изключително елементарни и еднообразни синтактични похвати в DTD граматиката - използването на ограничения за количество (\*, +, ?, *нищо*) и за или-възможност ( | ). Като примерна извадка от DTD файла можем да разгледаме следните:

| <!ELEMENT classification (jewellery, electronics, antiques, art, precious\_metals, vehicles)> |
| --- |

*Тук установяваме именно подредбата на елементите, принадлежащи в classification (описано в т. 2.3 - Тип и представяне на съдържанието). Нямаме специфични ограничения за количество.*

| *<!ELEMENT resource (name, pawn\_ref, (author | manufacturer)?, model?, quantity?, pawnshop\_evaluation, description?, image)>* |
| --- |

*Тук посочваме, че елементът resource се състои от следните поделементи: name, pawn\_ref, author, manufacturer, model, quantity, pawnshop\_evaluation, description, image. Тук обаче имаме и някои специфични ограничения, освен подредбата да е специфицираната:*

* *точно един от елементите измежду {author, manufacturer} може да се среща;*
* *въпросният елемент, може да съществува точно {0, 1} пъти; така, както и model, quantity, description. (т.е. или да не се среща, или да се среща точно веднъж).*

| <!ATTLIST pawn  id ID #REQUIRED  status (Active|Pending|Settled|EMNR) #REQUIRED> |
| --- |

*Тук указваме, че елементът pawn разполага с атрибут с име id, който е от тип ID и е задължителен, и атрибут status, който притежава една от стойностите измежду множеството {Active, Pending, Settled, EMNR} и отново е задължителен.*

Трансформацията на XML документа се случва чрез *pawnshop.xsl.* Нека разгледаме някои примерни откъси и от него:

| <fo:table-cell>  <fo:block-container position="relative">  <fo:block position="relative" margin-left="20pt">  <xsl:apply-templates select=" image" />  </fo:block>  </fo:block-container> </fo:table-cell> |
| --- |

*На примерът по-горе можем да видим част от сегмента, описващ как сме позиционирали изображението спрямо респективната страница от съдържанието на PDF файла. Указваме стойност relative на позиционния атрибут и извикваме темплейта за image, намиращ се в същия файл. (Илюстрацията на цялостния процес по трансформацията в PDF определено не е нещо постижимо в рамките на изображение от няколко реда, тъй като цялостният файл надхвърля 600 и прикачването му в този документ би било безсмислено)*

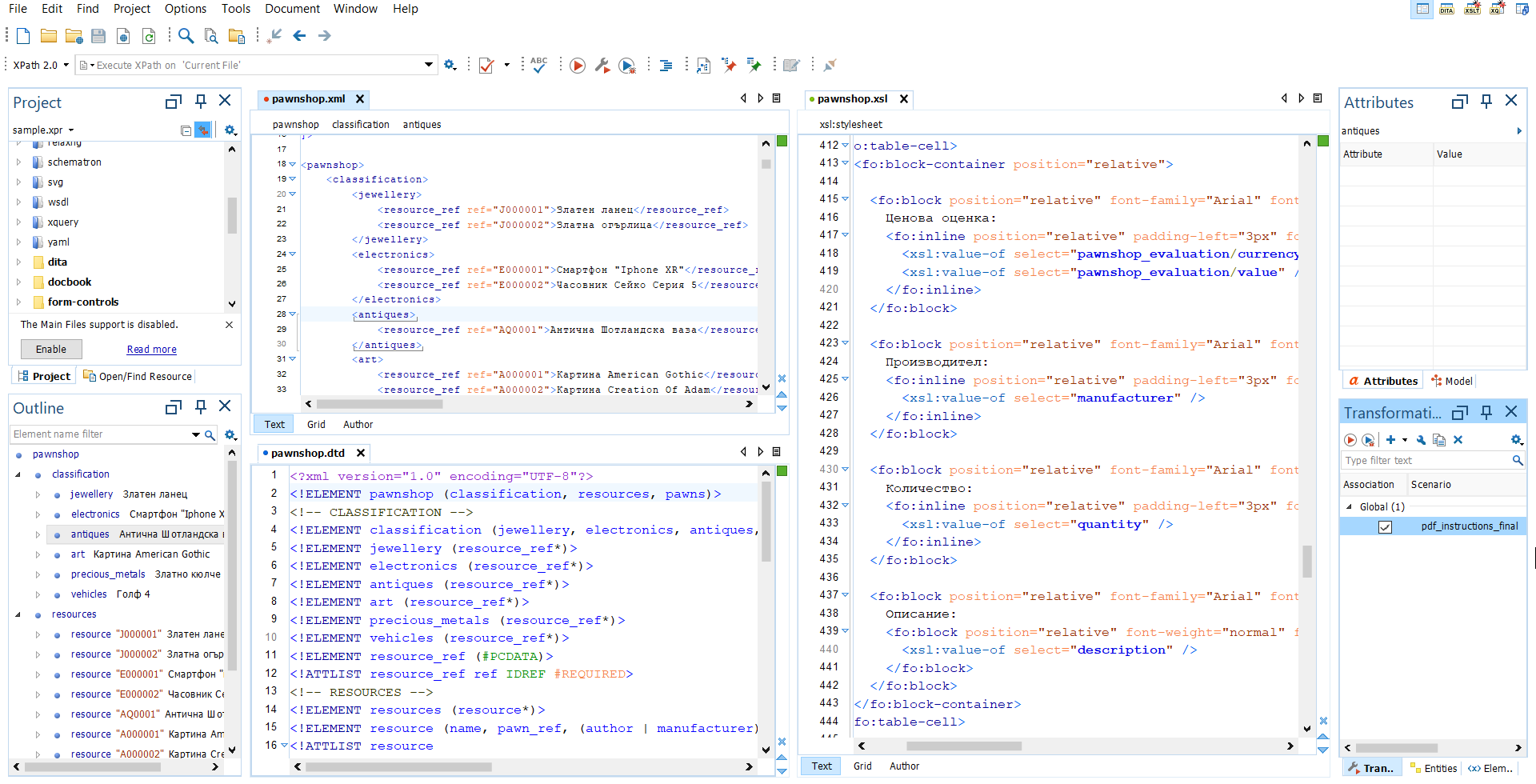
| <fo:page-sequence master-reference="page">  <fo:flow flow-name="xsl-region-body">  <fo:block-container position="absolute" top="-2.5cm" left="-2.5cm">  <fo:block position="absolute" text-align="center" font-weight="bold" margin-left="2cm" font-family="Monotype Corsiva" font-size="56pt" padding-before="50mm" color="black">  <fo:block margin-bottom="10mm" font-style="bold">  Каталог  </fo:block>  <fo:block>на заложна къща</fo:block>  <fo:block>ФМИ</fo:block>  <fo:external-graphic src="url('images/pawnshop.png')" content-height="500" content-width="360" margin-left="5cm" margin-bottom="15mm"></fo:external-graphic>  </fo:block>  </fo:block-container>  </fo:flow>  </fo:page-sequence> |
| --- |

*Тук създаваме нова страничка в PDF документа, която играе ролята на заглавна страница. Задаваме позиция, размер и отстояние на заглавието “Каталог на заложна къща ФМИ”. Освен това, добавяме графично изображение, което сме определили като външно (external) и му задаваме височина/ширина, както и отстояние.*

По този начин крайния резултат на проекта, а именно pawnshop.pdf, представлява цялото съдържание на *pawnshop.xml*, валидирано чрез *pawnshop.dtd*, и поместено в PDF формат посредством *pawnshop.xsl*.

# **Тестване**

След създаването на XML документа, той бе валидиран с подходящо създаден за целта DTD документ. За реализацията на самата валидация биха могли да се използват множество валидатори в Интернет пространството като тези, които използвахме за изпълнението на модулите (вкл. и средата Visual Studio Code, която валидира XML документите спрямо DTD граматиките *real time*, и която използвахме ние), а за графичното представяне на XML документа предпочетохме да използваме средата за XML обработка Oxygen XML Editor, версия 24.0, която при регистрация с валиден имейл и въвеждане на изпратения ключ за достъп предоставя безплатен тридесет дневен срок за ползване за академични цели. Програмата получава като вход XSL и XML документ, валидира данните с помощта на pawnshop.dtd (DTD граматиката на каталога) и генерира PDF документа, който визуализира данните.

*Примерна визуализация на средата за обработка, използвана като едно от средствата за съставянето на текущия проект, валидирането и тестването на компонентите, които го изграждат.*

Прикрепеното изображение предоставя информация относно крайния резултат на PDF документа - изискване, описано в шаблона на документацията на текущия проект.



*Примерни страници от каталога, визуализиращи обекти, притежавани към текущ момент от заложната къща, и наличната относно същността и залагането им информация:*

Повече информация относно дизайна на системата може да бъде намерена в точка 4 - Дизайн.

# **Заключение и възможно бъдещо развитие**

Като изходен резултат на изпълнението на проекта бихме могли да заключим, че разполагаме с качествено класифицирана и ясно структурирана система на ресурсите в една заложна къща. Въпросната система позволява добавянето и премахването на ресурси, допълването / модификацията / изтриването на данни, свързани със залога на кореспондиращия ресурс и т.н. От ключово значение за самата нея е DTD граматиката, поместена във файла *pawnshop.dtd*, която е съставена така, че да продължи да валидира успешно компонентите на XML файла при гореспоменатите трансформации и видоизменения в бъдеще. По този начин осигуряваме на проекта едно от най-необходимите качества за съществуването му и бъдещото му развитие - ***разширяемостта***. Благодарение на нея сме склонни да смятаме, че структурата на системата има далеч непренебрежим потенциал в сферата на информационното обслужване на подобни учреждения със стопанска цел.

# **Разпределение на работата**

Съставянето на текущия проект безусловно изискваше качествената оперативна синхронизация, която смятаме, че постигнахме като екип. Разбиването на работата по структурата, функционалността, дизайна и тестването на чисто индивидуални аспекти според нас е немислимо в конкретния сценарии, тъй като формираните компоненти са субективно зависими. Поради изброените фактори нашият екип предпочете да работи унитарно по отношение на прогреса, заедно да анализира и оценява възникващите възгледи и възможностите за имплементация, и да развива модулите на задачата в синхрон. За улеснение на работния процес при писането на кода беше използван GitHub, а за разработката на документа - GoogleDocs,

# **Използвани литературни източници и Уеб сайтове**

*Посочените източници на информация не се отнасят до конкретни извадки, цитирани из съдържанието на текущия документ, а представляват средства за научаване на операциите, необходими за изпълнението на проекта, и установяване на по-конкретни концепции за изходен резултат.*

1. [*https://xsltfiddle.liberty-development.net/bnnZWJ/1*](https://xsltfiddle.liberty-development.net/bnnZWJ/1)
2. [*https://w3schools.sinsixx.com/xslfo/default.asp.htm*](https://w3schools.sinsixx.com/xslfo/default.asp.htm)
3. [*https://www.oxygenxml.com/*](https://www.oxygenxml.com/)
4. [*http://lihvari.com/*](http://lihvari.com/)