

Data PipeLine Project

A22 Cohort, Applied MSc in Data Analytics

XML database modeling for storing the data of a Language Holiday Agency.

Performed by:

Mahdi Khalsi, Ainur Iagafarov, Cyrille Nwabo, Leonardo Bertocco

Instructed by:

Professor Catherine Faron

DataScienceTechInstitute, Paris 27.01.2023

**Table of Contents**

[**I. Introduction** 3](#_Toc125719744)

[**II. XML Database Modeling** 3](#_Toc125719745)

[**III. XML File Processing** 4](#_Toc125719746)

[**IV. Pros and cons of the selected model** 5](#_Toc125719747)

[**V. The use of XSL transformations** 5](#_Toc125719748)

[**IV. Conclusion** 8](#_Toc125719749)

# **I. Introduction**

The report represents the XML schema's modeling decisions along with the benefits and drawbacks of those decisions as they relate to the data processing. In additions to this, the report contains the XSLT transformations and the use cases they addressed to. The project involved following working environment, tools and services:

**-** **Visual Studio Code**;

**- Free Online XSLT Test Tool**: used to apply XSL transformation to the XML file and get the output in html format (<https://xslttest.appspot.com/>);

- **Fake Person ID creator:** used to create fake profile data randomly;

- **Creately** (SaaS visual collaboration tool with diagramming and design capabilities designed by Cinergix): used to create an XML Schema tree.

# **II. XML Database Modeling**

Figure 1: Graphical Representation of the XML Tree used in the modeling of the XML database.

# **III. XML File Processing**

The Schema, represented on Figure 1 above, has a ‘tree structure’ and consists of the following crucial elements: the main node Language\_Holiday\_Agency\_Data; Client\_Info is the sub node which is divided into sequence list of 4 parts containing details about location, holiday, booking information and, of course, personal information. We relied on the modeled database of 10 client profiles, which were created using our imagination and various generator tools, mentioned in the introduction of the report.

**Personal\_Info** element represents general information about a client. Such things as name, surname, age, nationality, city, contact details, etc. – are here for identification purposes.

**Booking\_Info** has two branches: **Holiday\_Type** and **Payment\_Info**. The first branch shows preferences of clients in type of holiday they prefer, such as family immersions, summer camps, 60+ groups, Christmas courses, etc. In addition, it represents the information of number of children and adults, as well as shows the period duration, which was chosen by a client to study a particular language. Payment\_Info is important – it shows the promos, information about person who makes the payment. Additionally, the contract is included here (with Signed or NotSigned options).

**Holiday\_Info** is divided in three branches.

**Certificates\_Info** is for identification of a client’s current level at a certain language. **Course\_Info** contains the information about a teacher chosen for learning and his experience level, also a learning type (offline or online) is represented here. **Offers** branch is considered for two groups only: Juniors (under 16) and those who are older than 60.

The **Location** node represents the information answering two main questions: where to stay during holiday and how to get there. **Accommodation** branch has the name and address of a place, type (hotel, appartement etc.) and number of rooms preferred by a client.

In the **Transport** branch, we decided to include all means of transport and put a *Ticket\_ID* and *Tranport\_Type* to clarify. Moreover, the branch contains information about client’s departure and destination cities and departure and arrival dates.

# **IV. Pros and cons of the selected model**

**Advantages:**

1) well-defined model and simplicity of implementation: The “Client Info” element in the travel agency database houses all the data in an orderly fashion. As a result, the "Client Info" element, which offers a simple overview of the clients, may access all the database information. It’s simple to implement the model, modify the model, and get data from the database using this particular grouping style.

2) simple and flexible interpreting: the tree structure makes analyzing over the tree simple and flexible, which simplifies maintenance of the XML database.

3) simple and understandable XQUERY for XSL: The XML Tree shown in Figure 1 makes it simpler to understand and construct XSL stylesheet queries as well as to retrieve data from the database.

**Disadvantages:**

1) complex XPATH: the query needed to access specific data, from the tree structure model, result to be composed by a long chain of queries in order to access specific data related to one another.

2) alternative strategy: perhaps, all the database could be reorganized under the element “Booking Info” or “Holiday Info” instead of “Client Info” to allow easier access to certain elements, in some cases. This would mean a change in the assumptions of the initial viewpoint regarding the structure of the database.

**XML Schema**

The XML file is referenced in the schema for the Language\_Holiday\_Agency\_Data model. The Schema aids in defining the various data types found in the XML file as well as elements.

**XML vs. JSON MODELING DIFFERENTIATION**

A second approach involved writing the XML database model in JSON. This method used suggested that modeling the database in JSON seems to be less complicated than modeling it in XML.

# **V. The use of XSL transformations**

There are 7 possible scenarios imagined for making 7 transformations from the Language\_Holiday\_Agency\_Data model presented above.

**TRANSFORMATION 1**

This scenario was written to extract all the Personal information of all Clients, in order to have an overview of the clients profile in the database.

**Clients Personal Information**

Graphical user interface, text, application

Description automatically generated

Figure 2: Transformation 1 output

**TRANSFORMATION 2**

This second scenario displays the clients in the database traveling with a school and some other details regarding in general duration and language to study.

In the column ‘Holiday Type’ if True then the student belongs to a school, else False. (There are 2 Clients traveling with School in the Database).

**Table

Description automatically generated**

Figure 3: Transformation 2 output

**TRANSFORMATION 3**

The third scenario output shows the certificate exam attempts passed by clients and the languages chosen. In addition it shows chosen support tools (Yes/No). The average entry test score for all clients is 52.2.

Table

Description automatically generated

Figure 4: Transformation 3 output

**TRANSFORMATION 4**

Payment information of clients is represented in this case. In addition, we counted percentage of clients choosing a particular language: English language: 30 %, French language: 30 %, Chinese language: 10 %, Spanish language: 20 %, German language: 10 %.

Graphical user interface, text, application

Description automatically generated

Figure 5: Transformation 4 output

**TRANSFORMATION 5**

5th scenario represents transportation and accommodation info of all clients (total number of clients is 10). Number of clients travelling with Plane: 9; number of clients preferred a cruise ship: 1.

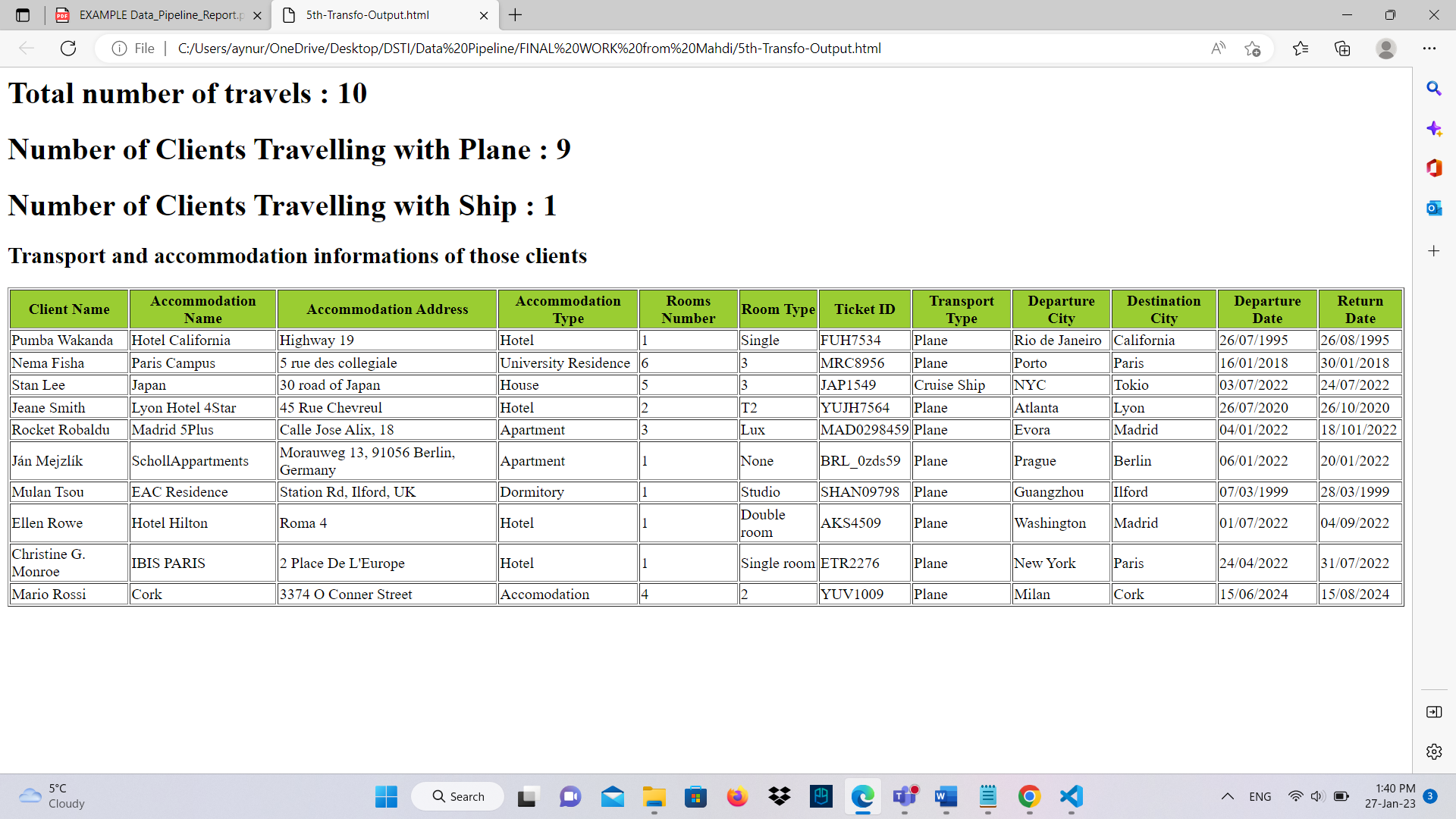


Figure 6: Transformation 5 output

**6th and 7th scenarios are in the zipped file, coming with the report.**

# **VI. Conclusion**

This report provides a thorough analysis of the Data Pipeline part 1 project by modeling an XML database for a language holiday agency in accordance with the assignment description and instructions. The report includes:

- modeling choices behind the XML schema, and the advantages and disadvantages of choices for the processing of the data;

- XSLT transformations and the use cases they answer.

Working environment, the tools and online services used also presented in the report.

Files included:

- source XML file

- XML Schema file

- XSLT files containing in comments their description in natural language

- the output of the XSLT transformations

- the JSON Schema