



Structures de données avancées - Mini Projet

Auteur: Robin GEORGI

Sujet : Modélisation d'une base de données XML pour stocker et manipuler les données d'un organisme de séjour linguistique.

Introduction

Le sujet de ce mini-projet est de modéliser une base de données XML pour stocker et manipuler les données d'un organisme de séjour linguistique selon les modalités données dans le sujet sur moodle. Afin de créer des scénarios variés, je me suis permis d'ajouter une caractéristique supplémentaire au sujet : la possibilité de gérer des séjours linguistiques accessibles aux personnes en situation de handicap. Ainsi chaque client peut être modélisé dans la base de données comme étant en situation de handicap ou non, et chaque séjour peut être modélisé dans la base de données comme étant accessible aux personnes en situation de handicap.

Modélisation de la base de données

Choix de modélisation

J'ai fait le choix de créer une base de données XML avec une structure hiérarchique dont la racine est le noeud correspondant aux établissements de séjours linguistiques car ce sont ces éléments qui relient l'ensemble des acteurs (clients, accompagnateurs et professeurs) évoqués dans le sujet.

Cette hiérarchie est modélisée dans le schéma XML qui met en place toutes les contraintes de la base de données.

Une fois le schéma XML créé, il est possible de créer une base de données XML valide en respectant les contraintes imposées par le schéma. C'est ce que j'ai fait dans le fichier LinguisticStay.xml.

Avantages de la modélisation

Le principal avantage de cette modélisation est sa construction hiérarchique qui permet de lier les différents acteurs de la base de données entre eux. Il est ainsi assez intuitif de rejouter un nouveau client dans la base de données, il suffit de le placer dans la liste des clients de l'établissement dans lequel il souhaite partir.

Inconvénients de la modélisation

Le principal inconvénient de cette modélisation est la duplication des données lorsqu'un même client souhaite participer à plusieurs voyages. En effet, il est nécessaire de créer un nouveau noeud client pour chaque voyage auquel il participe. Cela peut être problématique si l'on souhaite modifier les informations d'un client car il faudra modifier toutes les occurrences de ce client dans la base de données.

Scénarios de transformation XSLT

Scénario 1 : Affichage de tous les séjours linguistiques

Le premier scénario consiste à afficher tous les séjours linguistiques de la base de données. Pour cela, il suffit de parcourir la base de données et d'afficher les informations de chaque séjour linguistique.

Grâce au fichier [GetAllStays.xsl](#), on obtient ainsi le résultat suivant (le rapport ne prend pas en compte les feuilles de style CSS):

All Linguistic Stays

Stay ID	Linguistic Facility	Location	Type	Public	Destination	Start Date	End Date
1	ABC Language School	London, UK	familyTrip	adults	London, UK	2023-06-10	2023-06-15
2	ABC Language School	London, UK	schoolTrip	children/teenagers	New York, USA	2023-07-01	2023-07-05

Scénario 2 : Affichage de tous les séjours linguistiques dans une ville donnée

Le second scénario consiste à afficher tous les séjours linguistiques d'une ville donnée dans le cas où un client souhaite visiter un pays en particulier par exemple. Pour cela, il suffit de parcourir la base de données et d'afficher les informations de chaque séjour linguistique dont la ville correspond à la ville donnée en paramètre. Dans l'exemple ci-dessous, on affiche tous les séjours linguistiques de Londres.

Grâce au fichier [GetLondonStays.xsl](#), on obtient ainsi le résultat suivant (le rapport ne prend pas en compte les feuilles de style CSS):

Linguistic Stays in London

Stay ID	Linguistic Facility	Location	Type	Public	Destination	Start Date	End Date
1	ABC Language School	London, UK	familyTrip	adults	London, UK	2023-06-10	2023-06-15

Scénario 3 : Affichage de tous les séjours linguistiques accessibles aux personnes en situation de handicap

Le troisième scénario consiste à afficher tous les séjours linguistiques accessibles aux personnes en situation de handicap. Pour cela, il suffit de parcourir la base de données et d'afficher les informations de chaque séjour linguistique dont le champ "disabilitiesAdapted" est égal à true. Ainsi un client en situation de handicap pourra facilement trouver un séjour linguistique adapté à ses besoins.

Grâce au fichier [GetAccessibleStays.xsl](#), on obtient ainsi le résultat suivant (le rapport ne prend pas en compte les feuilles de style CSS):

Accessible Linguistic Stays

Stay ID	Linguistic Facility	Location	Type	Public	Destination	Start Date	End Date
2	ABC Language School	London, UK	schoolTrip	children/teenagers	New York, USA	2023-07-01	2023-07-05

Scénario 4 : Extraction des informations des clients et des professeurs

Le quatrième scénario consiste à extraire les informations des clients et des professeurs. Ces informations peuvent être utile pour automatiser un envoi de mail de demande confirmation de présence des clients et des professeurs durant le voyage linguistique, le tout via une API acceptant des fichiers XML ayant une syntaxe prédéfinie. Pour extraire ces données, il suffit de parcourir la base de données et d'afficher les informations de chaque client et de chaque professeur.

Grâce au fichier [GetTeachersAndClients.xsl](#), on obtient ainsi le résultat suivant ([cliquez ici pour voir le fichier XML complet](#)) :

```

<?xml version="1.0" encoding="UTF-8"?>
<participants xmlns:ls="http://www.example.org/LinguisticStay">
    <participant type="client">
        <firstName>Jane</firstName>
        <lastName>Doe</lastName>
        <dateOfBirth>1985-07-12</dateOfBirth>
        <email>jane.doe@example.com</email>
        <phone>+33123456789</phone>
        <isDisabled>true</isDisabled>
    </participant>
    <participant type="client">
        <firstName>John</firstName>
        <lastName>Doe</lastName>
        <dateOfBirth>1982-04-28</dateOfBirth>
        <email>john.doe@example.com</email>
        <phone>+33198765432</phone>
        <isDisabled>false</isDisabled>
    </participant>
    [...]
    <participant type="teacher">
        <firstName>Michael</firstName>
        <lastName>Brown</lastName>
        <dateOfBirth>1985-08-14</dateOfBirth>
        <email>michael.brown@example.com</email>
        <phone>+442012345679</phone>
        <isDisabled>false</isDisabled>
    </participant>
    <participant type="teacher">
        <firstName>William</firstName>
        <lastName>Taylor</lastName>
        <dateOfBirth>1991-01-22</dateOfBirth>
        <email>william.taylor@example.com</email>
        <phone>+442012345681</phone>
        <isDisabled>false</isDisabled>
    </participant>
</participants>

```

Scénario 5 : Extraction des informations des clients en situation de handicap

Le cinquième scénario consiste à extraire les informations des clients en situation de handicap. Ces informations peuvent être utile pour automatiser un envoi de mail de demande d'information supplémentaires sur le handicap des clients concernés dans le but de préparer du matériel spécifique ou éventuellement de former le personnel d'accueil pour faciliter le quotidien de leurs clients durant le voyage linguistique. Cet envoi automatique de mail peut être via une API acceptant des fichiers JSON ayant une syntaxe prédefinie. Pour extraire ces données, il suffit de parcourir la base de données et d'afficher les informations de chaque client en situation de handicap.

Grâce au fichier [GetDisabledClients.xsl](#), on obtient ainsi le résultat suivant :

```
[
  {
    "firstName": "Jane",
    "lastName": "Doe",
    "dateOfBirth": "1985-07-12",
    "email": "jane.doe@example.com",
    "phone": "+33123456789",
    "isDisabled": true
  },
  {
    "firstName": "Emily",
    "lastName": "Johnson",
    "dateOfBirth": "1987-11-23",
    "email": "emily.johnson@example.com",
    "phone": "+33198765433",
    "isDisabled": true
  },
  {
    "firstName": "Robert",
    "lastName": "Smith",
    "dateOfBirth": "2008-06-30",
    "email": "robert.smith@example.com",
    "phone": "+33123456790",
    "isDisabled": true
  }
]
```