



FORMALISATION DE DONNEES TECHNOLOGIES XML

Rapport du projet Voyageur de Santé

Projet réalisé par :

KONE Zeïnabou Bissi

XU Sicong

Professeur chargé de l'UE : *GLADE Nicolas*

06 DECEMBRE 2022

UGA

Table des matières

I.INTRODUCTION	2
II.REALISATION DU PROJET	2
1.Problématique	2
2.CV – XHTML	2
3.Fichier XML de données	3
4.XML Schema et Namespace	4
5.La page de l'infirmière	5
6.Les visites d'un patient	5
a.Partie 1 : fichier XML patient	6
b.Partie 2 : fiche_patient	6
7.Clé d'unicité et d'existence	7
III.CONCLUSION	7

I. INTRODUCTION

Ce présent rapport est un récapitulatif de la réalisation du projet « **Voyageur De Santé** » qui a été fait de façon incrémentale et qui résume une grande partie des notions étudiées au cours de l'UE **FDD-Technologies XML**. Il a consisté en 7 exercices et l'ensemble des réalisations se trouvent dans le sous-répertoire **data** du répertoire **src** de telle sorte que :

- Les CV et les fichiers html se trouvent dans le sous-dossier **html** ;
- Les fichiers xml, xsd et xslt se trouvent respectivement dans le sous-répertoire **xml**, **xsd**, et **xslt**.
- Les feuilles de styles CSS utilisées pour nos mises en page et mise en forme se trouvent dans le dossier **css**.
- Un dernier dossier **logos** contient les images qui nous été nécessaires pour la mise en forme de nos pages html.

Tout au long du projet, nous avons utilisé l'environnement technique **NetBeans** pour tester la correction de nos réalisations. Nous sommes arrivées jusqu'à l'exercice de la Facture que nous avons commencé mais qui n'a pas été achevé (le script ne fonctionnait pas) !

Ci-dessous, nous allons détailler les différents exercices faits au fur et à mesure des TP.

II. REALISATION DU PROJET

1. Problématique

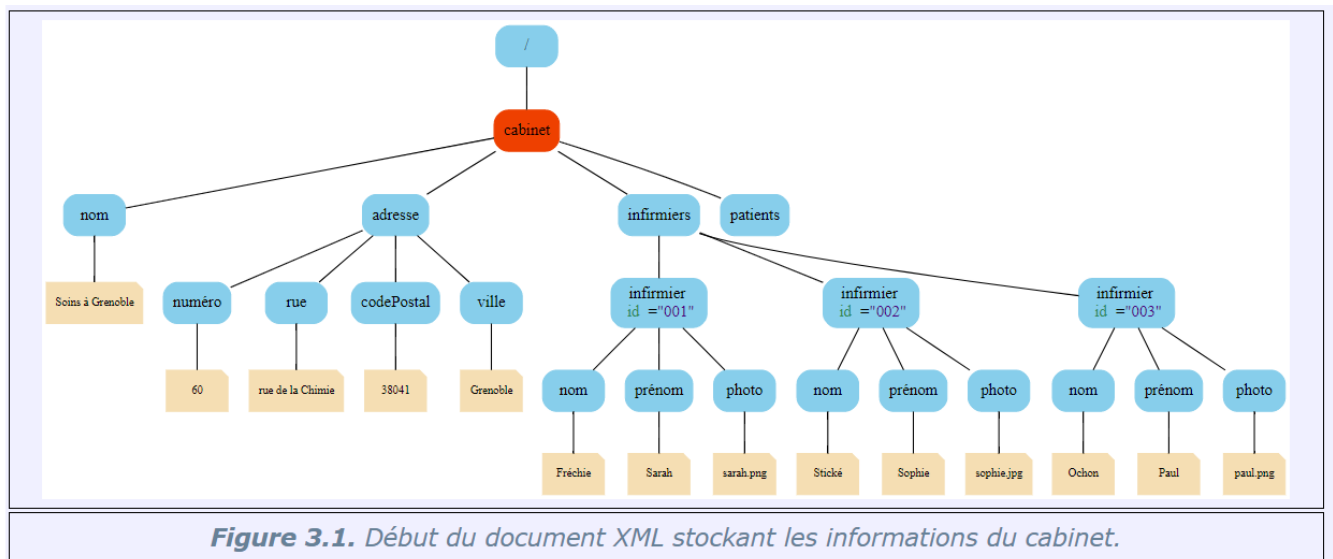
Cette première partie a fait l'objet d'une introduction globale du projet. Elle nous a permis de mieux nous imprégner de l'idée du projet. Elle nous aurait surtout servi pour la partie « **Sérialisation et Data Binding** » que nous avons commencé. Malheureusement nous avons rencontré des problèmes et avons jugé adéquat de nous concentrer sur ce qui était plus important.

2. CV – XHTML

Cette partie consistait à faire nos premiers pas avec le langage xml. Dans un premier temps, nous nous sommes familiarisées avec le langage xhtml en codant nos CV respectifs sous forme de page web. Nos différents CV se trouvent dans le dossier **html** (*CV_KONE* et *CV_XU*). Nous les avons rangés dans des dossiers différents pour avoir un rendu plus propre. Vous pourrez voir que nos pages CV contiennent des éléments que nous avons été et serons amenées à utiliser la plupart du temps (*tableau, liste, lien relatif vers le CV de l'autre, lien URL vers un site web extérieur, etc...*).

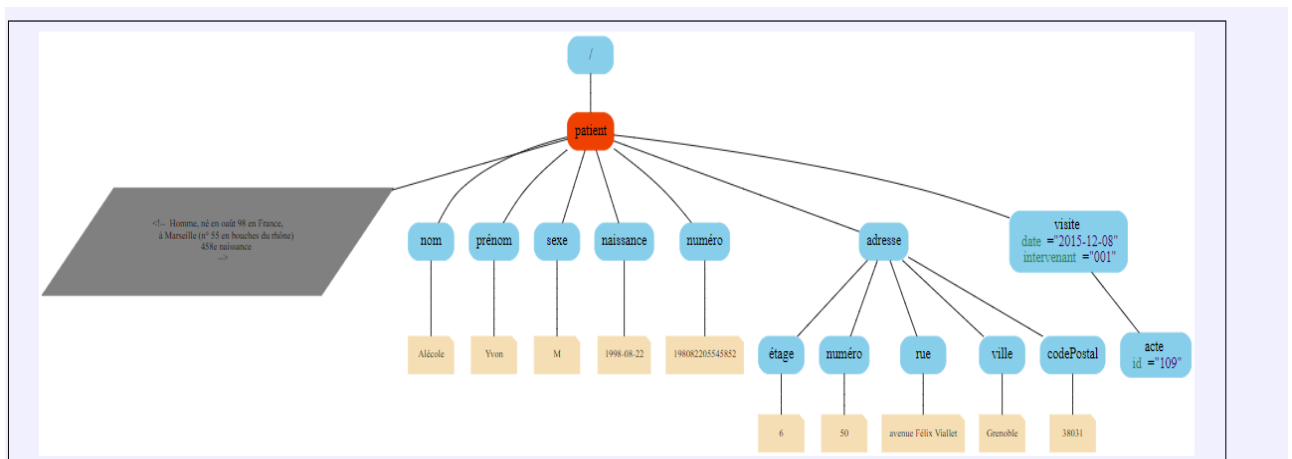
3. Fichier XML de données

Au cours de cet exercice, nous avons écrit notre tout premier document **xml** en respectant l'arborescence suivante (qui ne représente qu'une partie du document final) :



Ces informations sont stockées dans le fichier [cabinet.xml](#) (dans le sous-répertoire **xml**).

Par la suite, nous avons complété le nœud **patients** par les sous-éléments **patient** qui ont été donnés dans le sujet. Un nœud **patient** a également des sous-éléments comme l'indique la figure ci-après :



Dans cet exercice, nous nous sommes un peu familiarisées avec des données que nous ne manipulons pas forcément tous les jours : il s'agit du calcul des clés NIR des numéros de sécurité sociale. Nous avons acquis également des connaissances sur la nomenclature **NGAP** pour **déterminer** l'identifiant des actes des patients.

Nous avons ainsi calculé les clés NIR des numéros de sécurité sociale de tous les patients avec les données qui nous ont été fournies (grâce à l'outil web [Calcul de la clé NIR - Petits utilitaires \(marlot.org\)](#)) et avons exploré le fichier **actes.xml** pour déterminer l'identifiant de l'acte des soins des patient Alécole, Orouge, Pien et Kapoëtla.

SEXE (1 chiffre)	ANNEE DE NAISSANCE (2 chiffres)	MOIS DE NAISSANCE (2 chiffres)	DEPARTEMENT DE NAISSANCE (2 chiffres)	COMMUNE DE NAISSANCE (3 chiffres)	ORDRE DE NAISSANCE (3 chiffres)	CLE NIR (2 chiffres)
1 pour M et 2 pour F	Entre 00 et 99	Entre 01 (Janvier) et 12 (Décembre)	Deux premiers chiffres	3 derniers chiffres	De 001 à 999	Entre 01 et 97

Remarque :

Pour la patiente OROUGE Elvire, nous avons souligné une petite erreur concernant sa commune de naissance. Elle est née à Lyon qui correspond à la commune n°**123** dans le Rhône selon l'INSEE (comme vous pourriez le voir ici [Commune de Lyon \(69123\) – COG | Insee](#)). Mais les informations fournies dans le sujet du projet attestent que Lyon correspond à la commune n°**23** (auquel cas le numéro de sécurité sociale sera différent). Nous avons tout de même corrigé l'information et avons pris le **123** pour le calcul de notre clé NIR.

Également pour le patient Alécole Yvon (dont les informations nous ont été données), il y'a une erreur dans le numéro de sécurité sociale. Les informations du commentaire ne reflètent pas du tout le numéro donné. Nous avons donc recalculé ce numéro en fonction des informations données dans le commentaire.

4. [XML Schema et Namespace](#)

Cette partie a consisté à mettre en pratique nos connaissances acquis sur le **schéma** et les **espaces de nom**. Nous avons donc écrit le fichier *[cabinet.xsd](#)* qui nous a permis de pouvoir valider notre fichier *[cabinet.xml](#)*. Nous avons réalisé cette partie en prenant garde de bien contraindre notre fichier avec des types conformes aux données (de façon générale). Pour faire court :

- Le nom et le prénom d'un patient et d'un infirmier commencent par une lettre majuscule (nous pourrions éventuellement avoir des noms/prénoms composés, et donc chaque première lettre est en majuscule) : type simple **NomEtPrénom** ;
- Un étage est soit un rez-de-chaussée (RDC) ou un numéro : type simple **Etage** ;
- Un nom de ville commence toujours par une majuscule (similaire au cas des noms et prénoms) : type simple **Ville** ;
- Un code postal ne comprend que 5 chiffres exactement : type simple **CodePostal** ;
- L'identifiant d'une infirmière est composé d'exactly 3 chiffres : type simple **Id** ;
- La photo peut commencer soit par une lettre minuscule ou majuscule et se termine par des extensions du type .jpg, .png, .gif ou encore .jpeg : type simple **Photo** ;
- Le numéro de sécurité sociale respecte un certain nombre de contraintes : le premier chiffre est soit 1 (pour le sexe masculin) ou 2 (pour le sexe féminin) et seules ces deux valeurs peuvent être admises. Le deuxième et troisième chiffre représentent l'année de naissance du patient (de 00 à 99). Le troisième et le quatrième chiffre représentent le mois de naissance (01 à 12) et les 10 chiffres suivants n'ont pas vraiment de contraintes « spécifiques » : type simple **NuméroSécu** ;

5. La page de l'infirmière

Dans cet exercice, nous avons mis en pratique nos acquis théoriques concernant la transformation **xslt** de notre fichier *cabinet.xml*. Ici, nous avons généré une page **html** qui est une représentation de la page d'une infirmière (par défaut, il s'agit de **Sarah Fréchie** d'identifiant **001** car c'est la seule à avoir été attribué à un seul patient qui est Yvon Alécole). Nous nous sommes quand même amusées à attribuer des patients aux autres infirmiers/infirmières mais aussi à attribuer plusieurs patients à un(e) même infirmier/infirmière en modifiant le fichier *cabinet.xml* afin de tester notre feuille **xslt**. Bien évidemment après nos différents tests, nous avons **recorrigé** notre fichier *cabinet.xml* de façon qu'il respecte les spécifications données dans l'exercice 3. Ces différentes pages générées (à titre de test) se trouvent dans le dossier html (**page_fréchie.html**, **page_stické.html**, **page_ochon_1.html** et **page_ochon_2.html**).

La feuille CSS correspondante est quant à elle, dans le répertoire **css** (*ficheInfirmier.css*).

Exemple de page générée

The screenshot shows a web page titled "Cabinet médical de Grenoble". Below the title is a subtitle "Page de l'infirmier/infirmière". The main content area displays a greeting: "Bonjour Sarah" and "Aujourd'hui, vous avez 1 visite(s) !". Below this, it says "Patient(s) du jour :" followed by "PATIENT N°1/1". A table follows with the following data:

Nom	Prénom	Adresse	Soins(s)	Date
Alécole	Yvon	avenue Félix Viallet, 38031 Grenoble	Injection sous-cutanée.	2015-12-08

Below the table, there is a small text "Recette" in a light blue box.

6. Les visites d'un patient

Cet exercice comprend deux parties :

- ❖ Une première partie qui consistait à **générer un fichier xml à partir d'une feuille xslt** ;
- ❖ Une deuxième partie, analogue à l'exercice 5 qui consistait à **réaliser la page d'un patient (à partir du fichier xml généré dans la première partie)**.

a. Partie 1 : fichier XML patient

Ici, nous avons écrit la feuille **xslt** (*patient_xml.xsl*) du sous-répertoire **xslt** qui utilise le nom d'un patient, contenu dans le fichier *cabinet.xml* pour transformer ce dernier (*cabinet.xml*) en un **nouveau** fichier xml nommé *patient_NOMPATIENT.xml*. De la même façon que dans l'exercice 5, nous nous sommes amusées à générer le fichier xml de tous les patients de *cabinet.xml*. Ces différents fichiers se trouvent dans le sous-répertoire **xml** du projet (*patient_ALECOLE.xml*, *patient_OROUGE.xml*, *patient_PIEN.xml* et *patient_KAPOËTLA.xml*).

Afin d'avoir un travail plus propre et cohérent, nous avons à ce stade écrit le **Schéma XML** de ces *nouveaux fichiers xml*. Ce schéma se trouve dans le fichier *patient.xsd* du répertoire **xsd**. Pour que nos fichiers xml soient valides par rapport à ce schéma, nous avons rajouté dans chacun des fichiers xml des patients les namespaces qu'il faut (voir schéma ci-dessous).

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <patient xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
4         xmlns:act='http://www.ujf-grenoble.fr/l3miage/actes'
5         xmlns='http://www.ujf-grenoble.fr/l3miage/medical'
6         xsi:schemaLocation='http://www.ujf-grenoble.fr/l3miage/medical ../xsd/patient.xsd'>
```

b. Partie 2 : fiche_patient

Dans cette deuxième partie, nous avons écrit une feuille **xslt** (*patient_html.xsl*) qui nous permettrait d'avoir une page html qui renseigne les informations d'un patient. Cette page html est générée à partir du fichier xml patient_NOMPATIENT.xml (généré précédemment).

Les pages html générées (**pour tous les patients**) se trouvent dans le sous-répertoire **html** (la page de chaque patient est nommée avec son nom en majuscule : **patient_OROUGE**, **patient_ALECOLE**, **patient_KAPOËTLA** et **patient_PIEN**).

Exemple de page générée

Cabinet Infirmier de Grenoble

Fiche patient

ETAT CIVIL

Nom:
 Prénom:
 Sexe : ☒ M ☐ F
 Date de Naissance :
 Numéro de SS:


ADRESSE

Etage:
 Numéro:
 Rue:
 Ville:
 Code postal:

INTERVENANT

Nom :
 Prénom :

Nombre de visite(s) prévue(s) : 1
 Date : 2015-12-08
 • Injection sous-cutanée.



7. Clé d'unicité et d'existence

Cette dernière partie est la finalisation du projet. Nous avons modifié le schéma XML de notre cabinet (*cabinet.xsd*) de façon que les identifiants des médecins soient uniques et à ce que ceux auxquels se réfèrent les patients existent (voir cabinet.xsd).

On ne peut donc pas avoir deux infirmiers avec le même identifiant ni un infirmier avec deux identifiants différents. Aussi, il n'est pas possible d'attribuer à un patient, un intervenant qui n'existe pas (ici, on ne peut avoir que des intervenants d'identifiant 001 : Fréchie Sarah, 002 : Stické Sophie ou 003 : Ochon Paul).

III. CONCLUSION

Le projet était vraiment intéressant et nous a permis de mieux comprendre les notions vues en cours/TDs et de pouvoir avoir une connaissance un peu plus élargie concernant les technologies xml (**xmlns**, **espace de noms**, **clé d'existence**, **préfixes...**). La manipulation de données que nous ne maîtrisons pas forcément (**les nomenclatures NGAP**, **le calcul de la clé NIR...**) était également une facette assez intéressante du projet. Nous avons voulu aller un peu plus loin dans le data binding, mais avons été contraintes par le temps.