

# PROJET INF403 - Conception d'une base de données SQLite et intégration avec Python

## 1 Position du problème

Un hameau de santé est une structure de santé ayant un pilier global (holistique), un démocratique, un social, et un écologique, qui permet aux malades, même ceux habitants dans les déserts médicaux français, de se soigner et se faire soigner, grâce aux nombreux services que le hameau propose (structures de consultations, jardins thérapeutiques, serre de phytothérapie, micro-ferme, etc).

Un hameau de santé a informatisé la gestion de ses consultations (patients, services, maladies, etc.).

La base de données représente, pour chaque **patient** identifié par son numéro, ses consultations, la ou les maladie(s) qu'il a, le personnel qui s'occupe de lui s'il y en a (cela dépendra du service dans lequel le patient se trouve), le(s) service(s) dans le(s)quel(s) il se trouve.

Une **consultation** est identifiée par un code, et on y note son type, le patient, le personnel qui s'occupe de lui s'il y en a, la date et les frais de consultation du patient.

L'hameau de santé possède plusieurs **services**, identifiés par leurs noms, et chacun gère un certain nombre de maladies différentes. Une maladie peut être gérée dans un ou plusieurs services.

Chaque patient a un numéro, un nom, un prénom, une date de naissance et une adresse.

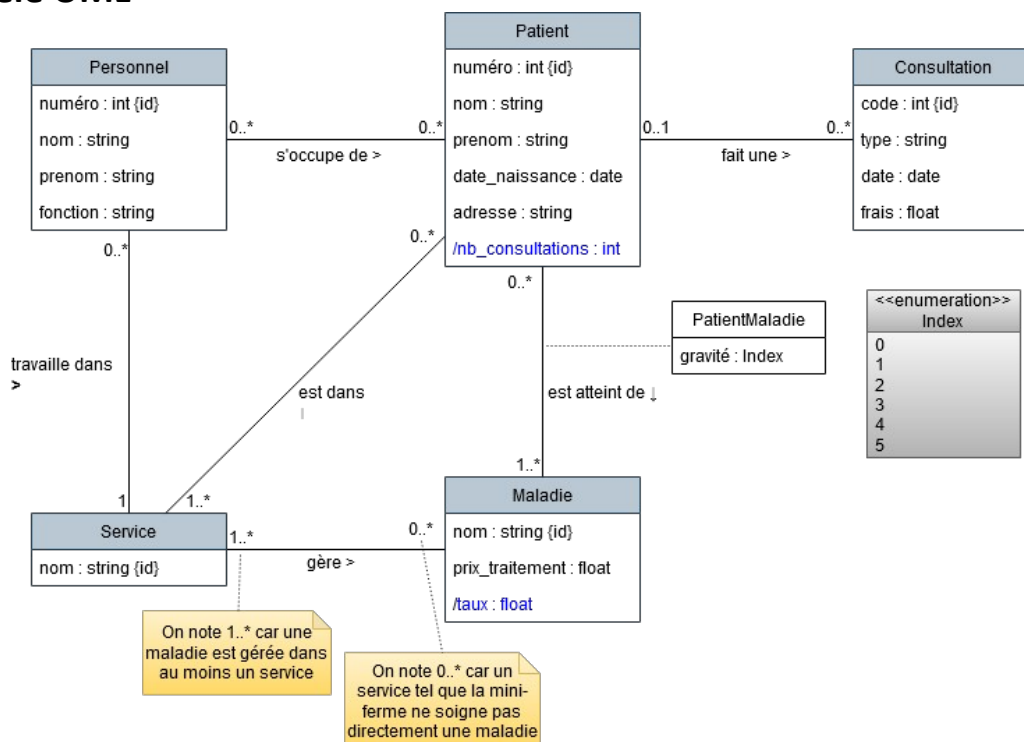
Chaque **membre du personnel** du hameau est identifié par un numéro, a un nom, un prénom, et une fonction. En effet, ce peut être un médecin, spécialiste (ophtalmologue, pédiatre, orthopédiste etc), aide-soignant, ou même un fermier (qui travaillerait dans le département de micro-ferme). Un employé ne peut travailler que dans un seul service. Un service peut ne pas comporter d'employés (par exemple dans les jardins thérapeutiques). Le personnel s'occupe des patients du service dans lequel il travaille. Le patient va dans le service qui gère sa maladie.

Un employé peut s'occuper de plusieurs patients et un patient peut voir plusieurs employés.

Une **maladie** est identifiée par son nom, a une gravité (indexée de 0 à 5) et un prix de traitement.

Un patient peut avoir plusieurs maladies, et une maladie peut atteindre plusieurs patients. Il se peut qu'une maladie ne soit atteinte par aucun patient.

## 2 Modèle UML



### 3 Modèle relationnel

#### Règles des noms :

Nom classe en singulier → Nom table en pluriel  
att → att\_nomclasse

#### Relations :

Personnel (numero\_personnel, nom\_personnel, prenom\_personnel, fonction\_personnel, nom\_service)  
/\* <u, n, p, f, s> ∈ Personnel ⇔ l'employé de numéro u a un nom n, un prénom p, une fonction f, et il travaille dans le service s \*/

Patients\_base (numero\_patient, nom\_patient, prenom\_patient, date\_naissance\_patient, adresse\_patient)  
/\* <u, n, p, d, a> ∈ Patients ⇔ le patient de numéro u, de nom n, prénom p et date de naissance d habite à l'adresse a \*/

PersonnelPatients (numero\_personnel, numero\_patient)  
/\* <e, a> ∈ PersonnelPatient ⇔ l'employé de numéro e s'occupe du patient numéro a \*/

Services (nom\_service)  
/\* <s> ∈ Service ⇔ le service est de nom s \*/

PatientsServices (numero\_patient, nom\_service)  
/\* <p, s> ∈ PatientService ⇔ Le patient de numéro p est dans le service s \*/

Maladies\_base (nom\_maladie, prix\_traitement\_maladie)  
/\* <m, g, p> ∈ Maladie ⇔ la maladie de nom m a un prix de traitement p \*/

PatientsMaladies (numero\_patient, nom\_maladie, gravite\_patientmaladie)  
/\* <p, m, g> ∈ PatientMaladie ⇔ le patient de numéro p a la maladie de nom m et de gravité g \*/

ServicesMaladies (nom\_service, nom\_maladie)  
/\* <s, m> ∈ ServiceMaladie ⇔ le service s s'occupe de la maladie m \*/

Consultations (code\_consultation, numero\_patient, numero\_personnel, type\_consultation, nom\_service, date\_consultation, frais\_consultation)  
/\* <c, n, p, t, s, d, f> ∈ Consultation ⇔ le patient n fait une consultation de code c, de type t, dans le service de nom s, à la date d, avec le personnel p, et paye les frais f \*/

(View) Patients (numero\_patient, nom\_patient, prenom\_patient, date\_naissance\_patient, adresse\_patient, nb\_consultations)

(View) Maladies (nom\_maladie, prix\_traitement\_maladie, taux\_maladie)

## Contraintes d'intégrité référentielles :

Personnel [nom\_service]  $\subseteq$  Services [nom\_service]

Personnel [nom\_service]  $\neq$  NULL

PersonnelPatients [numero\_personnel]  $\subseteq$  Personnel [numero\_personnel] /\* tout le personnel ne s'occupe pas des patients (ex : le fermier) \*/

PersonnelPatients [numero\_patient]  $\subseteq$  Patients\_base [numero\_patient] /\* tous les patients ne sont pas traités par les employés (ex : ceux qui vont aux jardins thérapeutiques) \*/

PatientsServices [numero\_patient] = Patients\_base [numero\_patient] /\* tous les patients sont dans au moins un service \*/

PatientsServices [nom\_service]  $\subseteq$  Services [nom\_service] /\* Il n'y a pas forcément de patients dans tous les services \*/

PatientsMaladies [numero\_patient] = Patients\_base [numero\_patient] /\* Tous les patients ont une maladie \*/

PatientsMaladies [nom\_maladie]  $\subseteq$  Maladies\_base [nom\_maladie] /\* Toutes les maladies ne sont pas atteintes par les patients \*/

ServicesMaladies [nom\_service]  $\subseteq$  Services [nom\_service]

ServicesMaladies [nom\_maladie]  $\subseteq$  Maladies\_base [nom\_maladie] /\* Une maladie peut-être gérée par plusieurs services \*/

gravite\_maladie  $\subseteq$  {0, 1, 2, 3, 4, 5}

Consultations [numero\_patient] = Patients\_base [numero\_patient]

Consultations [numero\_personnel]  $\subseteq$  Personnel [numero\_personnel]

Consultations [nom\_service]  $\subseteq$  Service [nom\_service]

## Domaines :

domaine (numero\_personnel) = domaine (numero\_patient) = domaine (code\_consultation) = entiers > 0

domaine (nom\_personnel) = domaine (prenom\_personnel) = domaine (fonction\_personnel) = domaine (nom\_patient) = domaine (prenom\_patient) = domaine (adresse\_patient) =  
domaine (nom\_maladie) = domaine (nom\_service) = domaine (type\_consultation) chaîne de caractères

domaine (date\_naissance\_patient) = domaine (date\_consultation) = date

domaine (prix\_traitement\_maladie) = domaine (prix\_consultation) = réel

domaine (gravite\_maladie) = énumération {0, 1, 2, 3, 4, 5}