

# Rapport Projet Web Of Linked Data

EL KATEB Sami, PAUL Thomas

## Choix de modélisation

Lors de l'élaboration de notre modèle RDFS destiné à décrire les rapports de cas médicaux, nous avons consulté diverses sources. En particulier, nous nous sommes référés aux pages Wikipedia traitant de l'observation médicale, de l'examen clinique et de l'examen complémentaire. De plus, nous avons choisi de nous baser autant que possible sur des ontologies existantes, en les étendant si nécessaire.

Ainsi, nous avons défini une classe intitulée “rapport de cas” (CaseReport) qui nous permet de typer nos ressources. Nous lui associons un patient, un médecin, un examen clinique et une thérapie. Pour ce faire, nous avons introduit les propriétés `hasClinicalExamination` (a un examen clinique), `hasDoctor`, `hasPatient` et `hasMedicalTherapy` (a un traitement médical). Nous avons choisi de spécifier uniquement l'étendu (le range) de ces propriétés, car elles peuvent s'appliquer à de nombreux domaines. Par exemple, nous pouvons lier un patient à un examen clinique dont il est le sujet ou à un médecin qui le prend en charge.

Dans notre modèle, nous avons intégré les classes “patient” et “traitement médical” (Medical Therapy) de l'ontologie de `schema.org`. Toutefois, nous avons créé une classe spécifique pour l'examen clinique. Pour le médecin, nous avons créé une classe distincte que nous avons étendu de la classe `Person` de `schema.org`.

Un examen clinique a pour objectif d'identifier des signes et symptômes cliniques afin de poser un diagnostic. Par conséquent, nous avons ajouté à notre ontologie les propriétés `hasSignOrSymptom`, `hasMedicalTest` (a un examen complémentaire) et `hasDiagnosis`. La propriété `hasSignOrSymptom` sert à relier notre examen clinique aux signes et symptômes de l'ontologie. La propriété `hasMedicalTest` permet de le relier à des examens complémentaires, tandis que la propriété `hasDiagnosis` permet de lui associer une pathologie (diseases). De plus, nous associons un examen clinique à un médecin et à un patient à l'aide des propriétés `hasPatient` et `hasDoctor`. Toutes ces classes excepté la classe `Doctor` proviennent de l'ontologie `schema.org`.

## Création de notre jeu de données

Pour élaborer nos données, nous avons choisi d'utiliser des maladies présentant des symptômes similaires (tels que la Grippe, le Covid-19, la Pneumonie, la Bronchite) ainsi que des maladies chroniques courantes (comme le Diabète et l'Hypertension). Nous avons fait ce choix pour pouvoir créer des requêtes SPARQL pertinentes basées sur les symptômes des maladies.

Nous avons parfois décidé d'attribuer un médecin différent pour l'examen clinique et pour le rapport de cas. Ce choix illustre la possibilité qu'un patient soit examiné par un médecin, puis soit suivi par un autre, que ce soit son médecin traitant ou un spécialiste de la maladie concernée.

Nous avons opté pour l'utilisation de ressources anonymes comme sujets des propriétés `doseSchedule` des thérapies, ainsi que des `weight` et `height` de chaque patient. Ce choix a été motivé par le fait que ces entités sont définies en tant que `QuantitativeValue` dans l'ontologie de `schema.org`. Cette définition offre notamment la possibilité de distinguer la valeur de l'unité de mesure. Ceci s'avère pratique pour des requêtes nécessitant des calculs sur ces données ou des conversions d'unités. Toutefois, l'utilisation d'une URI ne semble pas pertinente dans ce contexte. En effet, les URIs servent à identifier de manière unique des entités distinctes susceptibles d'être réutilisées, ce qui n'est pas le cas ici.

De plus, pour chaque maladie, nous avons opté pour associer la ressource correspondante sur DBPedia en utilisant la propriété `owl:sameAs`.

## Représentation RDFa et JSON-LD

Nous avons choisi de réaliser notre représentation RDFa à partir du cas d'usage métier d'un professionnel de santé cherchant à consulter le rapport de cas d'un patient.

Pour la version JSON-LD de notre graphe RDF, nous avons eu recours à l'outil de conversion [easyrdf](#).

## Choix de requêtes SPARQL

Lors de la conception de nos requêtes SPARQL, nous avons envisagé divers cas d'utilisation auxquels pourrait répondre notre modélisation. Par exemple, nous avons cherché à identifier les trois symptômes les plus courants ou à repérer les maladies présentant comme symptômes la fièvre et le mal de tête.

De plus, nous avons sélectionné des requêtes nous permettant de mettre en pratique les différents concepts abordés en cours, notamment les unions, les optionals, les constructs et les Paths.