

Rapport de projet de fin de cycle

Maison Intelligente



Proposé par :

Mme Amel BOUZEGHOUB

Réalisé par :

ARAB Koussaila

BABOU Ghiles

GUENANE Toufik

RAHMANI Nadjib

M2 DataScale 2021-2022

SOMMAIRE

- 1) Introduction
- 2) Objectif du projet
- 3) Outils Utilisés
- 4) Dataset Utilisé
- 5) Conception de l'ontologie
- 6) Actions automatiques et règles SparQl
- 7) Détections d'actions des utilisateurs et Vérification d'inférences
- 8) Conclusion

1 - Introduction

La technologie a tout changé dans le monde et rien ne représente mieux notre progression incessante vers l'avenir que la domotique, qui devient chaque jour plus courante et plus puissante.

Qu'est-ce qu'une maison intelligente ? Quels avantages offre-t-elle ?

La Maison intelligente :

Une maison intelligente fait référence à une configuration domestique pratique où les appareils peuvent être automatiquement contrôlés à distance depuis n'importe quelle position avec une connexion internet à l'aide d'un appareil mobile ou d'un autre appareil connecté. Les appareils d'une maison intelligente sont interconnectés via une connexion internet, ce qui permet à l'utilisateur de contrôler à distance des fonctions telles que l'accès sécurisé à la maison, la température, l'éclairage et un home cinéma.

Dans une maison connectée, par exemple le radiateur de la chambre se coupe automatiquement lorsque on ouvre la fenêtre de cette chambre car tous les capteurs et tous les actionneurs sont reliés ensemble et répondent à de multiples règles pour améliorer notre confort, notre sécurité ou baisser nos consommations.

Dans cette configuration, la maison reste connectée pour notamment nous alerter à distance, mais automatise un certain nombre d'actions que nous devons jusqu'ici faire, et que nous pouvons oublier bien souvent.

Notre maison intelligente, c'est donc des appareils qui deviennent de plus en plus intelligents et qui apprennent même à nous connaître et à nous proposer des solutions en relation avec nos goûts et nos habitudes.

Les bienfaits d'une maison Intelligente :

Une protection contre des intrus :

Si une simple alarme sonore ne suffit pas parfois à décourager des cambrioleurs. Des solutions astucieuses existent et la domotique de sécurité permet de mettre en place un véritable plan de défense.

En fonction des besoins (départ en vacances, week-end, nuit), vous anticipez des scénarios : vol par effraction, dégradation du jardin, fêtards nocturnes, etc.




L'équipement nécessaire est installé pour répondre à ces scénarios : fermeture des volets, arrosage automatique, éclairage programmé, etc.

Cela peut simuler votre présence pour décourager les intrusions ou déclencher la télésurveillance quand vous le décidez.

La caméra détectrice de présence permet de générer une alerte sur un service de sécurité ou même sur votre portable, grâce à de nouvelles applications : c'est la vidéo surveillance. Vous êtes informé en temps réel et à distance si quelqu'un cherche à entrer dans votre domicile. Vous pouvez alors faire appel à une société de sécurité ou à la police.

Une basse consommation :

La technologie et la communication permettent la programmation des multiples appareils de votre logement :



-  **Le chauffage** : il se déclenche une heure avant votre arrivée et s'éteint quand vous partez.
-  **L'éclairage** : minuterie et détecteur de mouvement vous évitent les oublis.
-  **L'arrosage** : l'arrosage automatique intelligent est aussi le plus performant. Programmé la nuit, l'eau ne s'évapore pas, les plantes ne sont pas roussies par le soleil.

Cette programmation permet de réaliser des économies d'énergie et donc d'argent.

Une protection contre des accidents domestiques :

Personne n'est à l'abri d'une chute, d'une coupure, d'une intoxication ou d'une brûlure ou encore d'un incendie. Les enfants sont malheureusement les premières victimes des accidents domestiques.

La maison intelligente permet de limiter les risques en intégrant :

-  Des détecteurs de fumée ;
-  Des détecteurs de monoxyde de carbone ;

- ✚ Des alarmes à déclencher soi-même en cas de besoin (PMR, pour personne à mobilité réduite) ;
- ✚ Des détecteurs de mouvement avec alarme reliés au smartphone de vos proches ;
- ✚ Des caméras de surveillance à distance (pour les enfants et/ou les personnes dépendantes).

Équipée de nombreux capteurs et alarmes, la maison intelligente permet de repérer tout dysfonctionnement potentiellement dangereux.

2 - Objectif du projet

L'objectif de notre projet est de réaliser une maison intelligente capable de reconnaître les changements de l'environnement ambiant en traitant les données récupérer à partir de capteurs ainsi de proposer un ensemble de services et d'actions automatique ou des configuration adaptées à ces changements.

3 - Outils Utilisés

Dans cette section nous allons présenter les différents outils et technologies que nous avons utilisés pour la mise en place de notre maison intelligente.

Protégé : est un éditeur d'ontologies gratuit (open source) et qui est un système de gestion de connaissances. Il nous permettra de :

- Définir les classes de notre ontologie.
- Définir les propriétés de notre ontologie.
- Ajout d'individus



Jena : est un ensemble d'outils pour la mise en place d'applications orientées Web sémantique. Parmi elles, on trouve une API Java avec laquelle on peut manipuler de nombreux langages tels que :

OWL, SPARQL, RDF/RDFS et de faire un raisonnement sur des modèles ontologiques à l'aide de moteurs d'inférences inclus dans Jena ou externes.



SparQL : est un langage de requête et un protocole qui permet de rechercher, d'ajouter, de modifier ou de supprimer des données RDF disponibles à travers Internet.



4 - Dataset Utilisé

On a choisi le « Dataset aruba » car il nous a donné plus d'informations sur les différentes activités que la personne avait faites durant son séjour dans la maison intelligente. Par exemple, il y'a des moments où elle prépare son repas, des moments où elle s'est couchée et des moments où elle s'est réveillée pour aller à la douche...

La figure suivante nous permet de voir 3 activités différentes avant que la personne aille se coucher. On constate que la personne s'est déplacé de son lit aux toilettes à **5h 40mins 51.303s**, ce mouvement est capté par le capteur M004 qui se trouve dans la chambre à coucher (bedroom dans la carte de la maison).

```
2010-11-04 05:40:51.303739 M004 ON Bed_to_Toilet begin
2010-11-04 05:40:52.342105 M005 OFF
2010-11-04 05:40:57.176409 M007 OFF
2010-11-04 05:40:57.941486 M004 OFF
2010-11-04 05:43:24.021475 M004 ON
2010-11-04 05:43:26.273181 M004 OFF
2010-11-04 05:43:26.345503 M007 ON
2010-11-04 05:43:26.793102 M004 ON
2010-11-04 05:43:27.195347 M007 OFF
2010-11-04 05:43:27.787437 M007 ON
2010-11-04 05:43:29.711796 M005 ON
2010-11-04 05:43:30.279021 M004 OFF Bed_to_Toilet end
2010-11-04 05:43:34.261135 M005 OFF
2010-11-04 05:43:35.941892 M007 OFF
2010-11-04 05:43:40.821615 M007 ON
2010-11-04 05:43:45.619681 M007 OFF
2010-11-04 05:43:45.7324 M003 ON Sleeping begin
```

On peut également trouver d'autres informations à part celle de mouvement par exemple celle de la température qui est capté par des capteurs de température, ainsi que des capteurs sur les trois différentes portes de la maison.

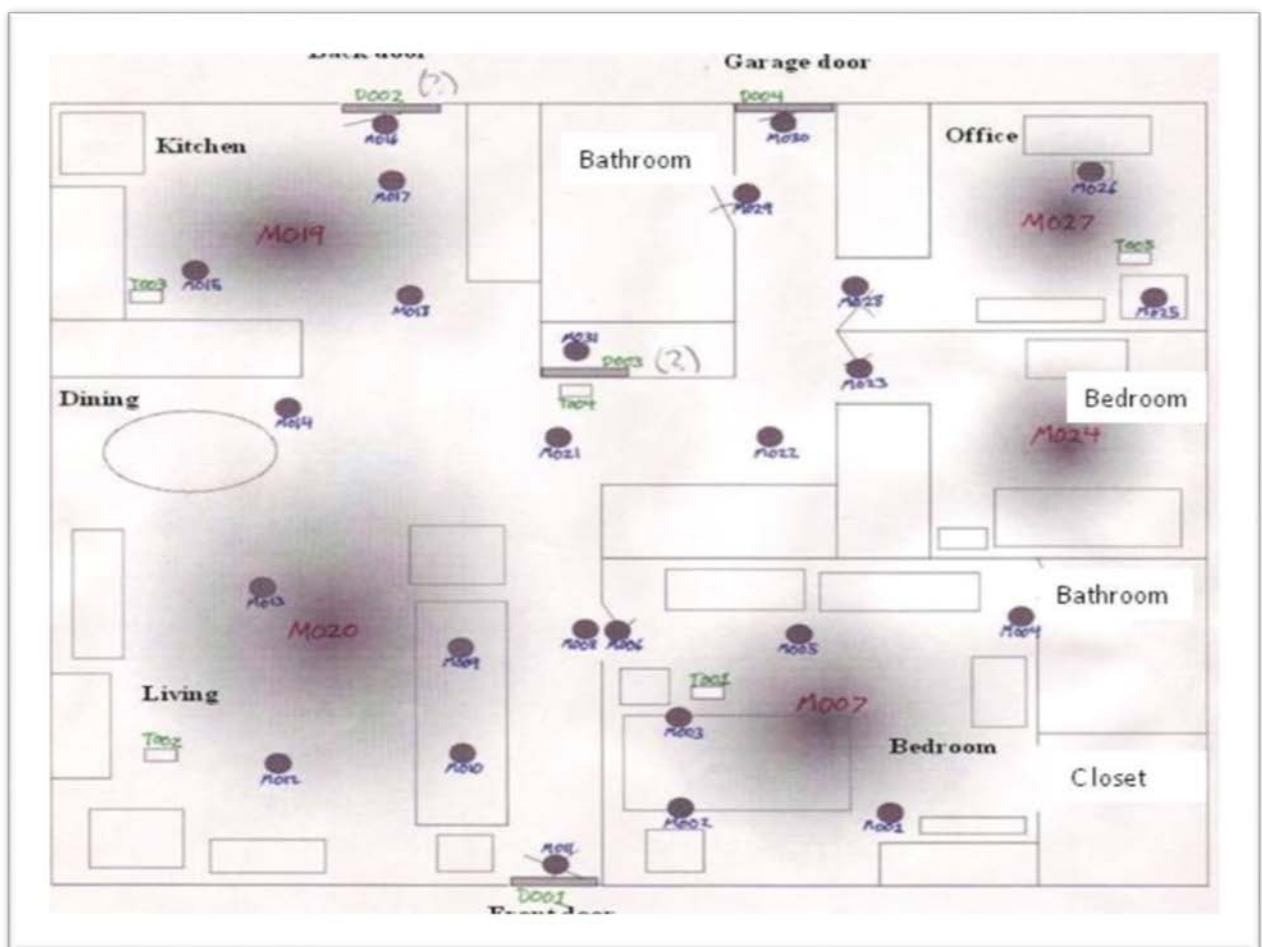
On a noté les différentes activités qu'on peut trouver dans le dataset et leurs différentes occurrences (notées entre parenthèse devant l'activité correspondante).

Le plan de la maison intelligente :

La figure suivante correspond au plan de la maison intelligente dont on a extrait les informations du jeu de données. On trouve plusieurs pièces telles que :

- ✚ Kitchen (Une cuisine) ;
- ✚ Dining (Une salle à manger) ;
- ✚ Living (Un salon) ;
- ✚ Bedrooms (Deux chambres à coucher) ;
- ✚ Bathrooms (Deux douches) ;
- ✚ Office (Un bureau).

Dans chaque pièce on trouve des capteurs de mouvement notés **M0XX**, des capteurs de température notés **T0XX** et on trouve également des capteurs sur les portes notés **D0XX**.



Voici le lien de téléchargement de Dataset Aruba :

<http://casas.wsu.edu/datasets/aruba.zip>

5 - Conception de l'ontologie





Pour modéliser les différentes informations sur la maison intelligente contenues dans le jeu de données et la figure qu'on a vu dans la partie précédente, on a conçu l'ontologie suivante :



On a d'abord pensé à créer la classe **SmartHome** et ses sous classes les plus pertinentes.









La sous-classe Action :

Cette sous-classe représente les différentes actions que les autres sous-classes peuvent être amené à faire. On trouve parmi ces actions :

-  **Allume** : qui permet d'allumer les lumières par exemple.
-  **Declanche** : qui déclenche une action ou une activité si une condition est vraie.
-  **Eteint** : qui permet d'éteindre les lumières s'il n'y a pas de mouvement ou permet d'éteindre le chauffage si un certain seuil de température est atteint.
-  **Ferme** et **ouvre** : qui permet de fermer et d'ouvrir les volets.






La sous-classe Objet :

C'est la classe des différents objets qu'on trouve dans la maison.

-  **Capteur** : est une sous-classe d'objet qui sert de super-classe aux sous-classes suivantes :
 - **Mouvement** : pour les capteurs de mouvements.
 - **Porte** : pour les capteurs de mouvements des portes.
 - **Temperature** : pour les capteurs de température.
-  **Climatiseur** : qui s'allume si une température est atteinte.
-  **Gaziniere** : pour représenter la gazinière ou la personne prépare ces repas.
-  **Door** : qui contiendra les instances des trois portes de la maison.
-  **Lampe** : pour les lampes des différentes pièces
-  **Lit** : qu'on trouve dans les chambres à coucher.
-  **TV** : qu'on trouve dans le salon.
-  **Volet** : qu'on a dans chaque pièce et qui s'ouvre ou se ferme selon le besoin.

La sous-classe Piece :

Elle a comme sous-classes les différentes pièces apparues sur le plan de la maison. On trouve :

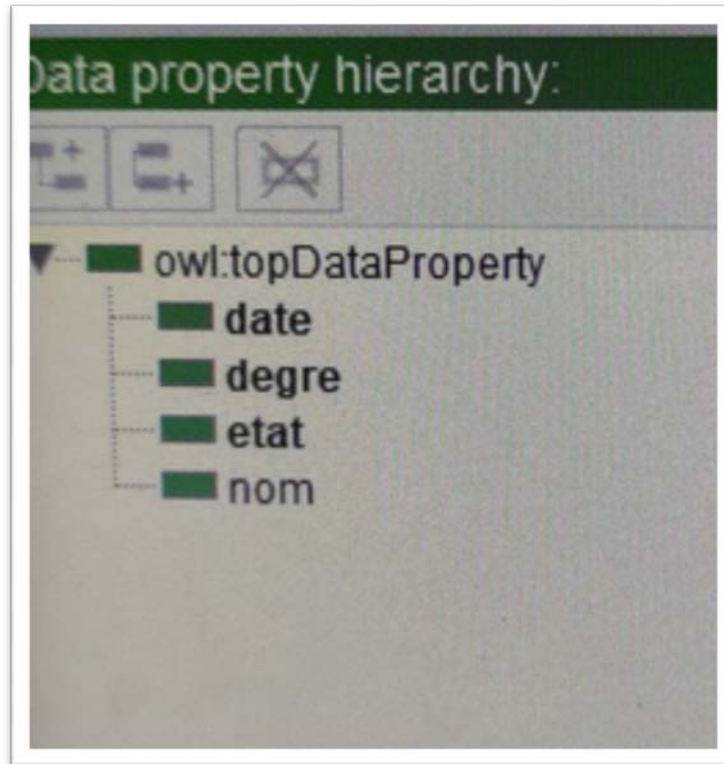
-  **Closet, Kitchen, Dining, Office, Living**
-  **Bathroom** : pour la salle de bain en bas à droite.
-  **Bathroom_big** : pour la salle de bain à gauche du bureau.
-  **Bedroom** : qui est la chambre à coucher au-dessous du bureau.
-  **Bedroom_big** : pour la chambre à coucher en bas à droite où on trouve une salle de bain.

La sous-classe Utilisateur :

Elle représente les différentes personnes pouvant habiter dans la maison.

Propriétés des données :

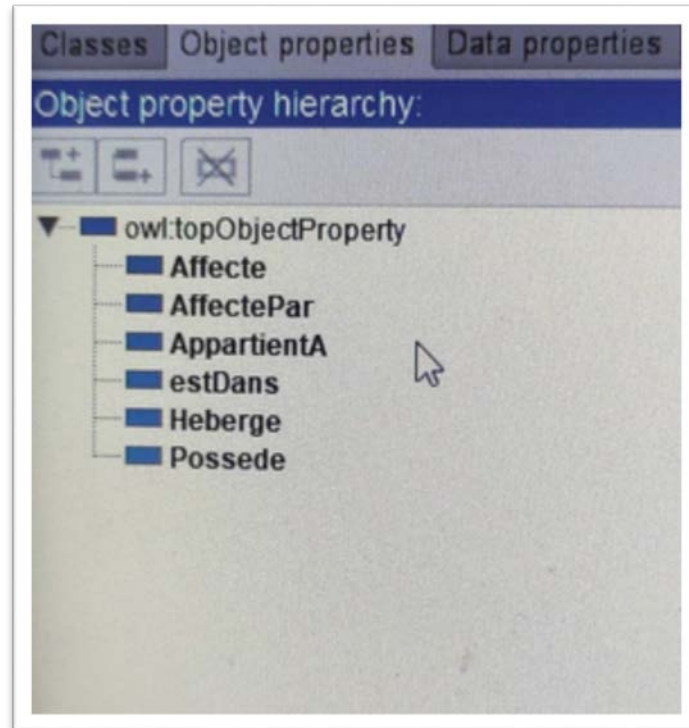
Pour les propriétés des données (Data Property) on a conclu qu'on aura besoin de quatre propriétés et selon la classe on attribue ce qui lui correspond.



- ✚ **Date** : c'est la date qui apparait dans le jeu de données et on la fait correspondre à l'objet auquel elle appartient dans le dataset.
- ✚ **Degre** : c'est les températures correspondantes au capteur de température dans le jeu de données.
- ✚ **Etat** : permet de décrire l'état d'un objet à un instant t et sa valeur diffère en fonction de l'objet :
 - **Volets et portes** : état = « Ouvert » ou « Fermée »
 - **Chauffage, TV, Climatiseur, Gaziniere** : état = « ON » ou « OFF »
 - **Capteurs de mouvements/portes** : état = « ON » ou « OFF »
 - **Lampe** : état = « allumée » ou « éteinte »
- ✚ **Nom** : c'est le nom qu'on donne à chaque pièce, objet...

Les Propriétés des objets (Object property) :

Pour mettre en place les équipements de la maison ou les objets dans des pièces convenables, on a créé des relations entre les classes comme suit :



- ✚ **estDans** : pour dire qu'un objet est dans une pièce.
- ✚ **Heberge** : l'inverse de estDans, une pièce héberge un objet.
- ✚ **Affecte** : Pour affecter une action à un objet
- ✚ **AffectePar** : l'inverse d'Affecte, une action est affectée par un objet
- ✚ **AppartientA** : Pour savoir quelle pièce appartient à un utilisateur.
- ✚ **Possede** : inverse de AppartientA.

6 - Actions automatiques et règles SparQl

Afin que notre application puisse prendre des décisions automatiques (ouvrir les volets, éteindre les lumières, fermé les portes ...) nous allons exécuter des règles SparQl sur notre model OWL.

Ici nous présentons les différentes règles qu'on a utilisé pour assurer ces décisions automatiques :

Ouvrir le Volet :

Cette règle permet d'ouvrir les volets d'une chambre si la température de cette chambre est comprise entre 19.5 et 27 degré.

```
[Ouvrir_Volet:(?capteur rdf:type ns:Temperature)
(?capteur ns:estDans ?piece) (?capteur ns:date ?date)
(?volet rdf:type ns:Volet)(?volet ns:estDans ?piece) (?capteur ns:degre ?value)
greaterThan(?value, "19.5"^^xsd:double) lessThan(?value, "27"^^xsd:double)
-> (?volet ns:etat "Ouvert"^^xsd:string) (?volet ns:date ?date) ]
```

Eteindre la télévision et la lampe :

Cette règle éteint la télévision et la lampe d'une pièce s'il ne y'a pas de mouvement dans cette pièce.

```
[Eteindre_tv_et_lampe: (?capteur_mv rdf:type ns:Mouvement) (?capteur_mv ns:estDans ?living)
(?capteur_mv ns:etat "OFF")(?lampe rdf:type ns:Lampe) (?lampe ns:estDans ?living)
(?tv rdf:type ns:TV) (?tv ns:estDans ?living)
-> (?tv ns:etat "eteinte"^^xsd:string) (?lampe ns:etat "eteinte"^^xsd:string)
(?capteur_mv ns:etat "OFF"^^xsd:string)]
```

Fermer le volet :

Cette règle permet de fermer les volets d'une pièce si sa température est inférieure à 20 degré.

```
[Fermer_Volet: (?capteur rdf:type ns:Temperature)
(?capteur ns:estDans ?bedroom) (?capteur ns:date ?date) (?volet rdf:type ns:Volet)
(?volet ns:estDans ?bedroom) (?capteur ns:degre ?value)
lessThan(?value, "20"^^xsd:double)
-> (?volet ns:etat "fermee"^^xsd:string) (?volet ns:date ?date) ]
```

Ouvrir la porte et le volet :

Cette règle permet d'ouvrir les portes et les voles si une alarme est déclenchée.

```
[Ouvrir_Porte_et_volet: (?alarme rdf:type ns:Alarme) (?alarme ns:etat "on")
(?door rdf:type ns:Door) (?door ns:etat "fermÃ©")(?volet rdf:type ns:Volet)
(?volet ns:etat "fermé")
-> (?door ns:etat "ouverte"^^xsd:string) (?volet ns:etat "ouvert"^^xsd:string)]
```

Allumer le climatiseur :

Cette règle permet d'allumer le climatiseur d'une pièce si la température détecter dans cette dernière est supérieur à 27 degré.

```
[Allume_Climatiseur:(?capteur rdf:type ns:Temperature)
(?capteur ns:estDans ?piece) (?capteur ns:date ?date)
(?Climatiseur rdf:type ns:Climatiseur)(?Climatiseur ns:estDans ?piece)
(?capteur ns:degre ?value)
greaterThan(?value, "27"^^xsd:double)
-> (?Climatiseur ns:etat "Allume"^^xsd:string) (?volet ns:date ?date) ]
```

Eteindre le climatiseur :

Cette règle permet d'éteindre le climatiseur d'une pièce si la température détectée dans cette dernière est supérieure à 27.5 degré.

```
[Eteindre_Climatiseur:(?capteur rdf:type ns:Temperature)
(?capteur ns:estDans ?piece) (?capteur ns:date ?date)
(?Climatiseur rdf:type ns:Climatiseur)(?Climatiseur ns:estDans ?piece)
(?capteur ns:degre ?value)
lessThan(?value, "27.5"^^xsd:double)
-> (?Climatiseur ns:etat "Eteint"^^xsd:string) (?volet ns:date ?date)]
```

Fermer la Gazinière et Ouvrir la Porte :

Cette règle permet d'éteindre la gazinière et d'ouvrir la porte de la cuisine s'il n'y a pas de mouvement dans cette dernière et la température détectée est supérieure à 40 degré.

```
[Fermer_Gaziniere_et_ouvrir_porte: (?capteur rdf:type ns:Temperature)
(?capteur2_mv rdf:type ns:Mouvement)
(?capteur ns:estDans ?piece) (?capteur ns:date ?date)
(?capteur2 ns:estDans ?piece) (?capteur2 ns:date ?date) (?capteur2 ns:etat "OFF")
(?gaziniere rdf:type ns:Gaziniere) (?gaziniere ns:etat "on")
(?door rdf:type ns:Door) (?door ns:etat "ferme")
(?gaziniere ns:estDans ?piece) (?door ns:estDans ?piece)(?capteur ns:degre ?value)
greaterThan(?value, "40"^^xsd:double)
-> (?door ns:etat "ouverte"^^xsd:string) (?gaziniere ns:etat "off"^^xsd:string)]
```

7 - Détections d'actions des utilisateurs et Vérification d'inférences

Comme notre dataset est trop volumineux (plus d'un million de lignes), l'exécution de notre application sur ce dataset prends énormément de temps. Cependant on a décidé d'extraire 5 sous-dataset pour 5 jours différents, par la suite on pourra choisir sur quel dataset auquel on appliquera nos règles SparQL qu'on a élaboré avant.

```
Veuillez choisir une date à évaluer de 1 à 5:  
1: pour la date 2010-11-04  
2: pour la date 2010-11-12  
3: pour la date 2010-12-25  
4: pour la date 2010-12-30  
5: pour la date 2011-02-01
```

L'un des premiers résultats qu'on peut voir après l'exécution de l'application sont les objets d'une des pièces de la maison :

Les objets de Kitchen:

```
| obj |  
-----  
| <http://www.semanticweb.org/win10/ontologies/2022/0/SmartHomeAruba#Chaise>  
| <http://www.semanticweb.org/win10/ontologies/2022/0/SmartHomeAruba#Chauffage_bed>  
| <http://www.semanticweb.org/win10/ontologies/2022/0/SmartHomeAruba#Table%20%C3%A0%20manger>  
| <http://www.semanticweb.org/win10/ontologies/2022/0/SmartHomeAruba#Bureau>  
| <http://www.semanticweb.org/win10/ontologies/2022/0/SmartHomeAruba#refrigerateur>  
| <http://www.semanticweb.org/win10/ontologies/2022/0/SmartHomeAruba#Micro-ondes>  
| <http://www.semanticweb.org/win10/ontologies/2022/0/SmartHomeAruba#table_dining>  
-----
```

De plus, on a pu aussi détecter les actions et les comportements établies par l'habitant en analysant le Dataset ligne par ligne, avec la fonction qui lit le type de l'action dans chaque pièce.

Exemple : Mouvement détecter dans la cuisine, et l'utilisateur prépare à manger dans la salle.

```
====> Quelqu'un prepare à manger à 21:14:57.877796  
  
====> Quelqu'un dans la cuisine, la lampe est allumé à 21:14:57.877796  
Update=====>M019  
|  
====> Quelqu'un prepare à manger à 21:14:59.481465  
Update=====>M017  
  
====> Quelqu'un prepare à manger à 21:15:00.676282  
Update=====>M015  
Update=====>M018  
Update=====>M017  
Update=====>M014  
  
-==> Quelqu'un mange dans la salle à 21:15:07.318885  
Update=====>M020
```

Exemple : Mouvement détecter dans la grande chambre et allumage d'une lampe du couloir qui mène à la douche.

```
====> Mouvement dans la grande chambre, Lampe allumé à 21:26:19.716641  
Update=====>M004  
  
====> EntRAIN de marcher de la chambre à la douche à 21:26:25.890022  
  
====> Mouvement dans la grande chambre, Lampe allumé à 21:26:25.890022  
Update=====>M005|
```

Exemple : Mouvement détecter dans la douche et puis la lampe est allumée au couloir.

```
-==> Quelqu'un dans la douche à 11:32:33.809867  
Update=====>M031  
Update=====>M021  
Update=====>M021  
Update=====>M021  
Update=====>M021  
Update=====>M021  
Update=====>M021  
Update=====>M020  
  
====> Mouvement dans le salon, lampe allumé à 11:33:00.276813  
Update=====>M020  
Update=====>M021  
Update=====>M021  
Update=====>M021  
Update=====>M031
```

Exemple : Mouvement détecté sur le canapé du salon indiquant que quelqu'un est en train de se détendre et l'allumage de la lampe.

```
Update=====>M010
=====> Quelqu'un se détend dans le salon à 13:30:43.623042
=====> Mouvement dans le salon, lampe allumé à 13:30:43.623042
Update=====>M012
```

- De plus, on a pu visualisé l'états des objets après l'application des règles SparQL qui nous permet de constater les résultats d'inférences.

Le dernier état du volet kitchen

etat	date	degre
"Ouvert"^^<http://www.w3.org/2001/XMLSchema#string>	"23:25:16.840313"^^<http://www.w3.org/2001/XMLSchema#string>	"22.0"^^<http://www.w3.org/2001/XMLSchema#double>

La règle utiliser pour ce résultat est celle d'Ouvrir Volet sur le volet kitchen avec une température de 22 degrés.

Le dernier état du volet bed

etat	date	degre
"Ouvert"^^<http://www.w3.org/2001/XMLSchema#string>	"23:40:27.418802"^^<http://www.w3.org/2001/XMLSchema#string>	"21.0"^^<http://www.w3.org/2001/XMLSchema#double>

La règle utiliser pour ce résultat est celle d'Ouvrir Volet sur le volet bed avec une température de 21 degrés.

Le dernier état du volet Living

etat	date	degre
"fermee"^^<http://www.w3.org/2001/XMLSchema#string>	"23:24:42.216445"^^<http://www.w3.org/2001/XMLSchema#string>	"19.0"^^<http://www.w3.org/2001/XMLSchema#double>

La règle utiliser pour ce résultat est celle de Fermer Volet sur le volet living avec une température de 19 degrés.

Le dernier état du volet office

etat	date	degre
"Ouvert"^^<http://www.w3.org/2001/XMLSchema#string>	"22:29:40.311758"^^<http://www.w3.org/2001/XMLSchema#string>	"21.5"^^<http://www.w3.org/2001/XMLSchema#double>

La règle utiliser pour ce résultat est celle d'Ouvrir Volet sur le volet office avec une température de 21.5 degrés.

8 - Conclusion

La domotique permet de contrôler, d'automatiser et de programmer votre habitat dans le but de vous simplifier et d'améliorer la vie au quotidien, de mieux gérer vos dépenses énergétiques et d'assurer votre sécurité.

Afin d'assurer toutes ces facilités automatiques, on a dû passer par plusieurs tâches. La première tâche était la construction d'une ontologie web où on représente les capteurs utilisés dans cette maison, ses équipements et les informations propres à chaque composant de notre environnement intelligent. Puis à l'aide de ces différents capteurs installés dans cette maison on a pu reconnaître et appliquer les bonnes actions nécessaires pour l'automatisation de cette environnement (Exemple : déclenchement d'alarme, fermeture de volets, éteindre la télévision, éteindre les lampes et réglage des thermostats ...).

Ce projet nous a permis la découverte des meilleures connaissances d'application de la domotique ce qui est fortement utile pour notre vie professionnelle future dans le domaine du Web Sémantique.

Notre contribution dans ce projet consiste à être un bon début pour la réalisation d'un environnement intelligent. Mais ce travail ne s'arrête pas là, il est toujours possible d'améliorer encore plusieurs tâches. En effet, beaucoup de possibilités s'offrent aux passionnés de domotique, tant sur le matériel disponible que sur les actions à réaliser. Nous pensons avoir entraperçu une partie de notre future vie active.