## **Cenário – GoCare**

Através de estudos, no futuro a sociedade ficará em sua maioria com pessoas idosas devido as pessoas terem menos filhos. Com isso, a necessidade de consultas médicas, pronto socorro, realização e exames, terão um aumento significativo e com a tecnologia cada vez mais acessível a todos, juntamos isso a necessidade de deslocamento do usuário (Paciente) ao hospital, que melhor se adequa a sua necessidade naquele momento, bom como a integração do sistema na área da saúde. Explico abaixo:

Vimos à carência existente no deslocamento de pessoas, rapidez e assertividade nos diagnósticos médico, seja através de uma consulta marcada, passar em um pronto socorro ou até mesmo realizar exames.

A GoCare como é chamada une o Paciente ao Hospital que mais atende sua necessidade. No primeiro momento será somente para quem possui convênios médicos (Particulares). Assim que o usuário baixar o aplicativo, o mesmo inicia o cadastro informando dados pessoais, convenio medico, endereço, histórico de doenças, alergia a medicamentos entre outros.

No momento em que necessita de ir a um pronto socorro, o usuário abrirá o aplicativo e selecionará o serviço que necessita, a exemplo o PRONTO SOCORRO (PS), com isso através do BIGDATA com parcerias aos hospitais e convênios teremos o retorno no aplicativo (através de API) informando qual hospital mais próximo do local onde o paciente está (utilizando GPS) e o menor tempo de espera nos Pronto Socorro. Assim que o usuário aceitar, um carro credenciado à GoCare (Ex: UBER) irá buscar o paciente e leva-lo até o hospital escolhido pelo paciente.

Assim que o motorista iniciar a corrida, será enviado as informações ao hospital de que o paciente está a caminho, o tempo estimado de chegada e até mesmo informações sobre medicamentos como por exemplo a consulta em redes de farmácias.

O hospital por sua vez terá a informação que o paciente está a caminho do hospital e o tempo estimado de chegada do mesmo, assim com sistemas integrados poderá adiantar todo tramite de ficha na recepção com intuito de agilizar a triagem e até mesmo informação importante ao médico, bastando somente a conferencia dos dados e assinatura do paciente, que futuramente poderá ser realizada digitalmente.

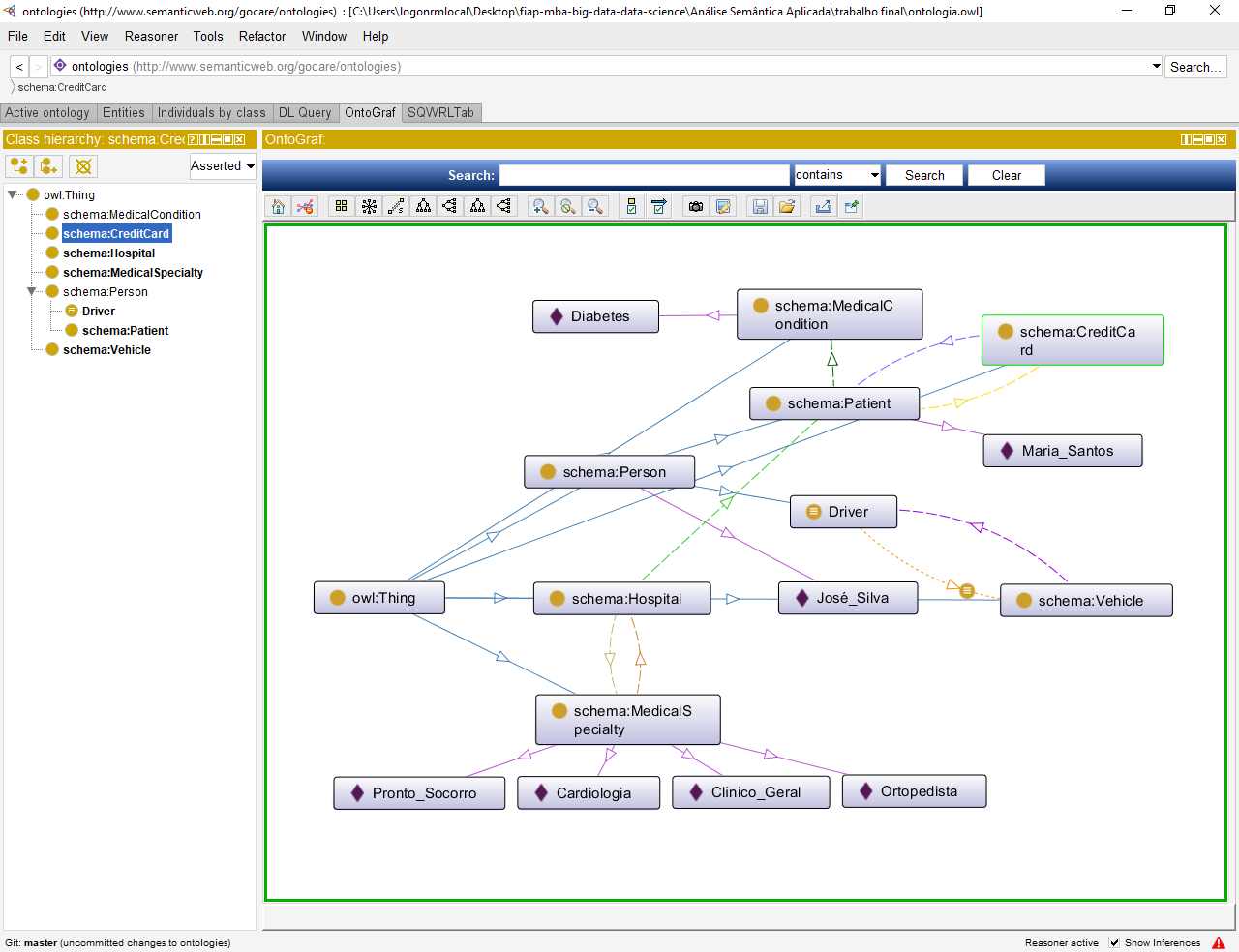
Após algum tempo de utilização e melhorias no processo, temos a intenção de adquirir parceria com as farmácias e laboratórios, pois através do CPF do paciente que é informado no momento da aquisição de remédios e a frequência com que isso ocorre, incluiremos essa informação também para que durante o trajeto do paciente até o hospital, o médico já recebe esses dados, ou seja, saberá qual remédio o paciente está ingerindo e com que frequência.

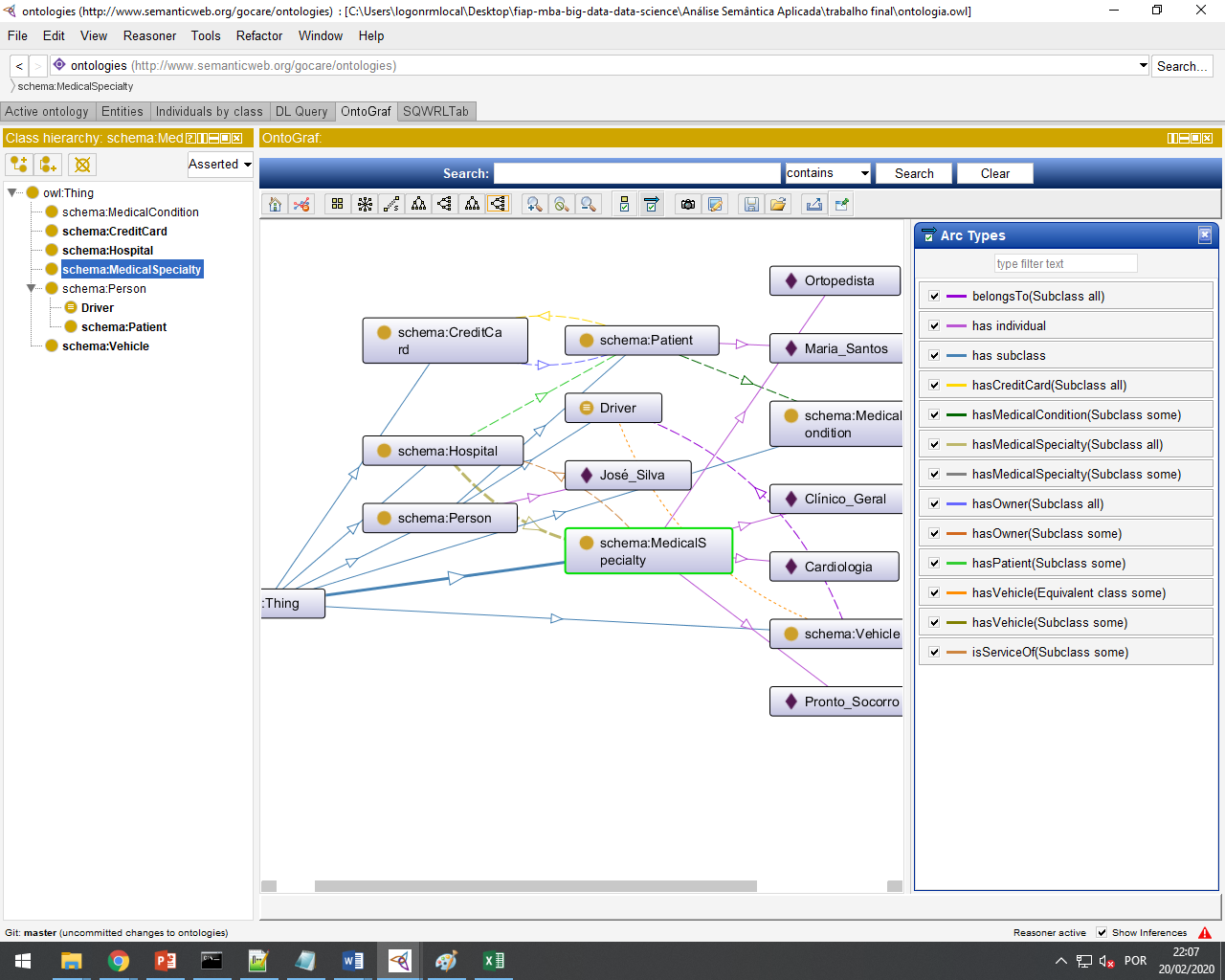
Sobre os laboratórios, o intuito é que assim que o médico der alta ao paciente e o mesmo estiver exames laboratoriais a serem feitos, a GoCare irá gerar um pou-up em seu aplicativo indicando os laboratórios mais próximos que poderá realizar os exames. Com isso a ideia é fazer parcerias com esses laboratórios a fim do mesmo ofertar ao paciente um desconto ou até mesmo a isenção do seu deslocamento até o laboratório.

## **Associar metadados aos conceitos do cenário escolhido**



## **Grafo do conceito da taxonomia**





## **Lista dos metadados relacionados no cenário proposto associados ao schema.org**



## **Linked Open Data que podem enriquecer o cenário**

Utilização de dados e imagens de hospitais do WikiData para enriquecimento da nossa base.

Exemplo de query SPARQL para coleta das informações de localização e imagens de todos os hospitais do Brasil:

SELECT \* WHERE {  
 ?item wdt:P31/wdt:P279\* wd:Q16917;

wdt:P625 ?geo.

?item wdt:P18 ?pic.

?item wdt:P17 wd:Q155.

}

Exemplo de código em Python para coleta das informações de localização e imagens de todos os hospitais do Brasil:

from SPARQLWrapper import SPARQLWrapper, JSON

endpoint\_url = "https://query.wikidata.org/sparql"

query = """SELECT \* WHERE {

?item wdt:P31/wdt:P279\* wd:Q16917;

wdt:P625 ?geo .

?item wdt:P18 ?pic.

?item wdt:P17 wd:Q155.

}"""

def get\_results(endpoint\_url, query):

sparql = SPARQLWrapper(endpoint\_url)

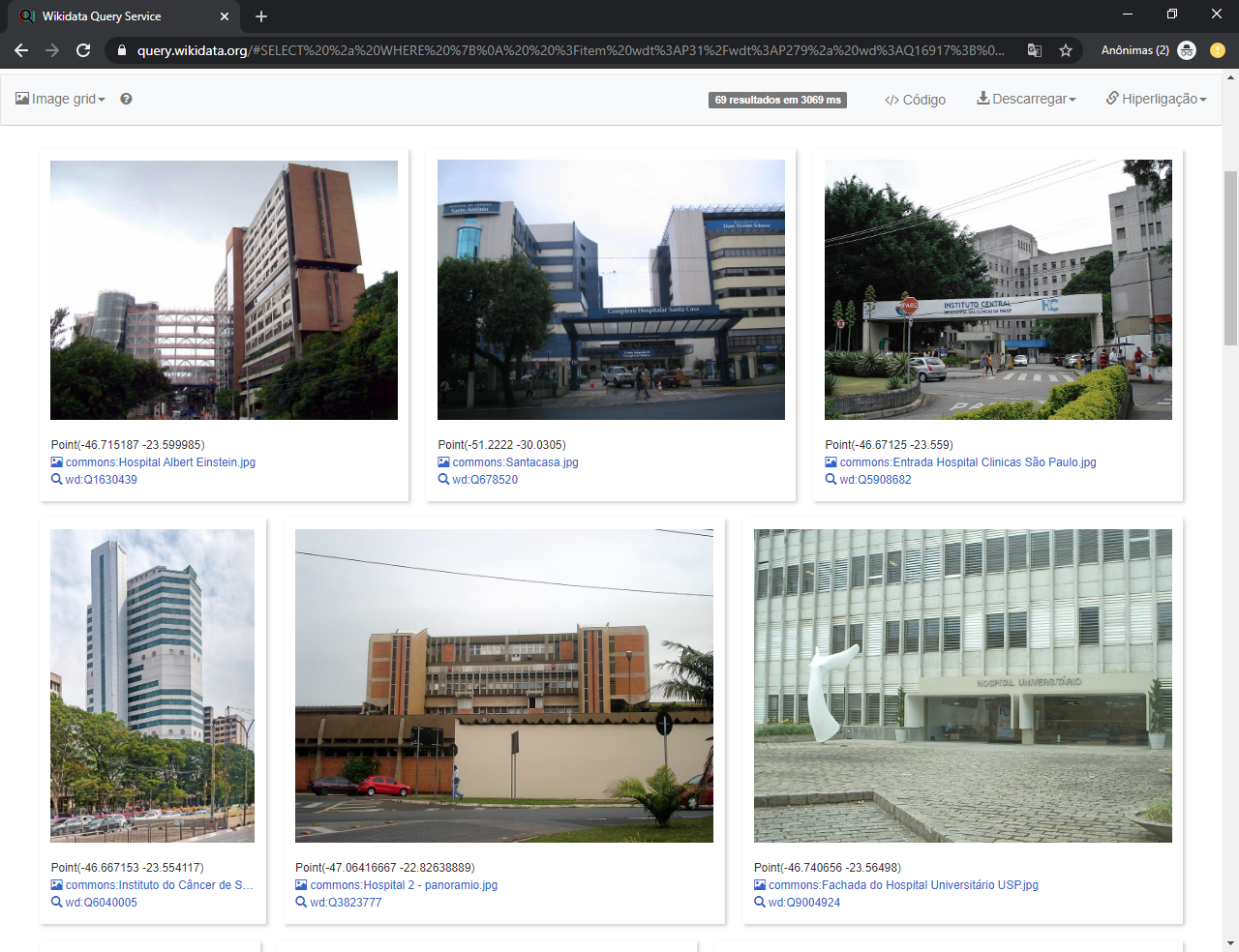
sparql.setQuery(query)

sparql.setReturnFormat(JSON)

return sparql.query().convert()

results = get\_results(endpoint\_url, query)

Retorno das informações do WikiData:



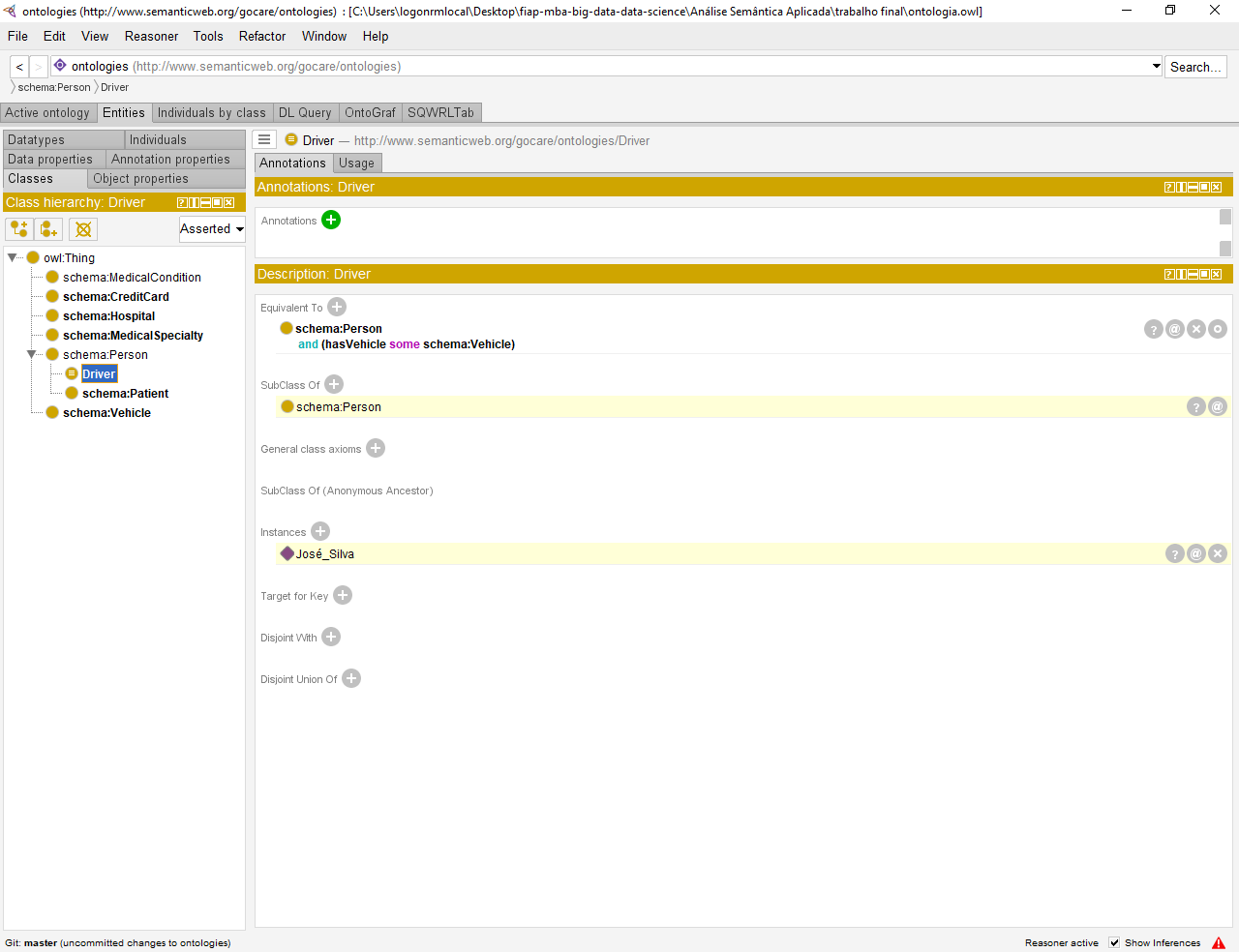
URL para consulta das informações utilizadas:  
  
[https://query.wikidata.org/sparql?query=SELECT%20\*%20WHERE%20%7B%0A%20%20%3Fitem%20wdt%3AP31%2Fwdt%3AP279\*%20wd%3AQ16917%3B%0A%20%20%20%20%20%20%20%20wdt%3AP625%20%3Fgeo%20.%0A%20%20%3Fitem%20wdt%3AP18%20%3Fpic.%0A%20%20%3Fitem%20wdt%3AP17%20wd%3AQ155.%0A%7D](https://query.wikidata.org/sparql?query=SELECT%20*%20WHERE%20%7B%0A%20%20%3Fitem%20wdt%3AP31%2Fwdt%3AP279*%20wd%3AQ16917%3B%0A%20%20%20%20%20%20%20%20wdt%3AP625%20%3Fgeo%20.%0A%20%20%3Fitem%20wdt%3AP18%20%3Fpic.%0A%20%20%3Fitem%20wdt%3AP17%20wd%3AQ155.%0A%7D)

## **Representação do cenário como uma ontologia**

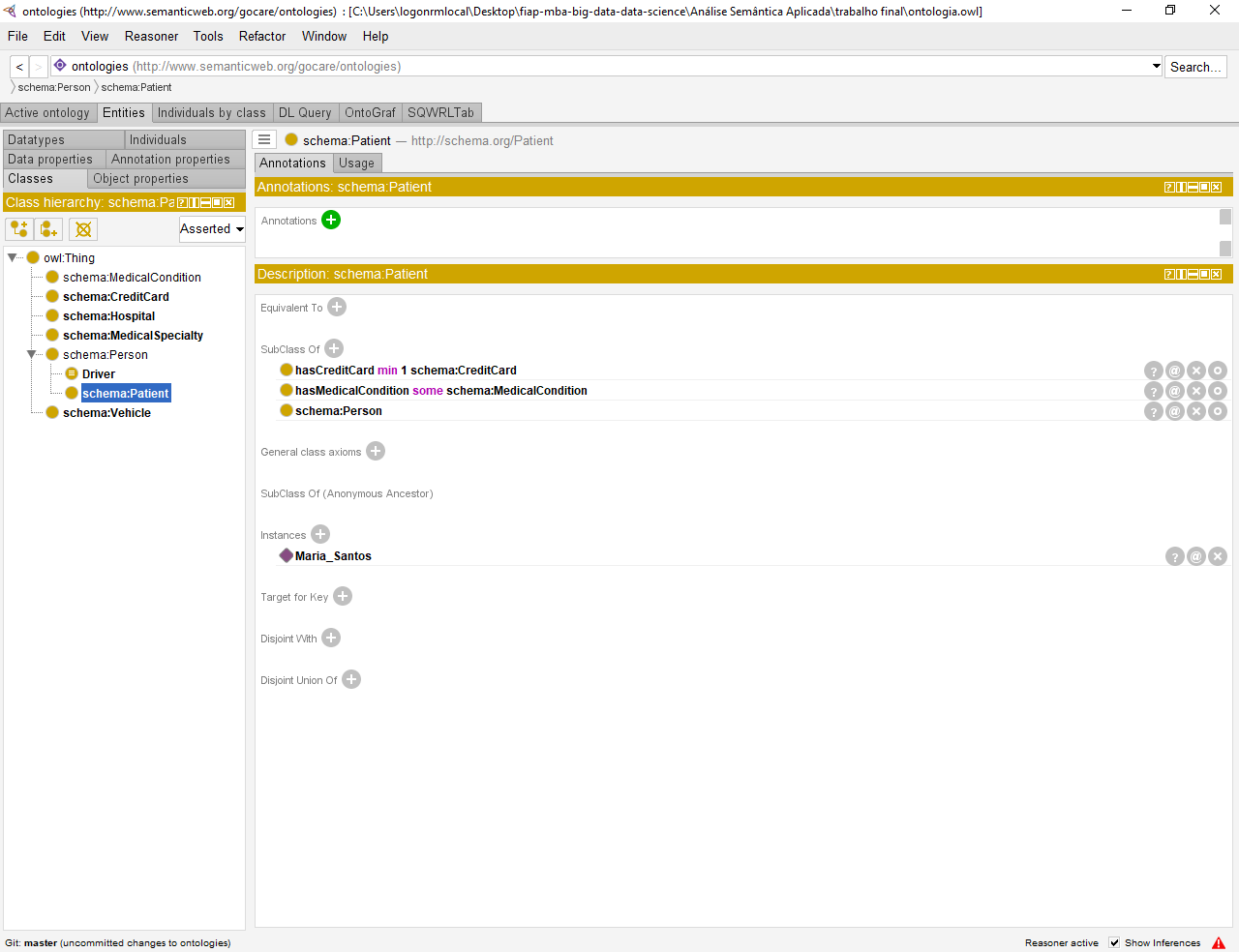
Utilizamos o Protégé para criar a ontologia, o arquivo completo está anexo (ontologia.owl), abaixo um trecho do código RDF/XML representando o paciente:  
  
<owl:Class rdf:about="http://schema.org/Patient">  
 <rdfs:subClassOf rdf:resource="http://schema.org/Person"/>  
 <rdfs:subClassOf>  
 <owl:Restriction>  
 <owl:onProperty rdf:resource="http://www.semanticweb.org/gocare/ontologies/hasMedicalSpecialty"/>  
 <owl:someValuesFrom rdf:resource="http://schema.org/MedicalSpecialty"/>  
 </owl:Restriction>  
 </rdfs:subClassOf>  
 <rdfs:subClassOf>  
 <owl:Restriction>  
 <owl:onProperty rdf:resource="http://www.semanticweb.org/gocare/ontologies/hasCreditCard"/>  
 <owl:minQualifiedCardinality rdf:datatype="http://www.w3.org/2001/XMLSchema#nonNegativeInteger">1</owl:minQualifiedCardinality>  
 <owl:onClass rdf:resource="http://schema.org/CreditCard"/>  
 </owl:Restriction>  
 </rdfs:subClassOf>  
</owl:Class>

## Listar três regras construídas a partir da ontologia

1) Motoristas (Driver) são subclasses de schema:Person, devem ter algum veículo (schema:Vehicle):



2) Pacientes (schema:Patient) devem ter um ou mais cartões de crédito (schema:CreditCard), podem ter alguma Declaração de Saúde (schema:Medicalcondition):



3) Hospitais (schema:Hospital) devem ter uma ou mais especialidades (schema: MedicalSpecialty), podem ter algum Paciente (schema:Patient):

