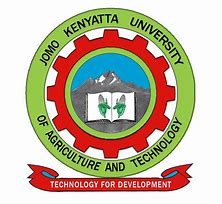
## 

****

**Foundations of Computer Logic and Symbolic Reasoning**

**ICS 3102: Project**

**Presented By:**

Margaret Mumbi Gichuhi - SCT313-2331/2023

**Date**: January 08, 2024

## 

Table of Contents

[**Question One (1)** 3](#_Toc155456558)

[**Introduction** 3](#_Toc155456559)

[**Inference Questions Analysis for Agriculture Dataset:** 4](#_Toc155456560)

[**Solution Overview** 6](#_Toc155456561)

[**Data Model** 9](#_Toc155456562)

[**Process Workflow** 10](#_Toc155456563)

# **Question One (1)**

Identify a scenario within a given domain (e.g. healthcare, Agriculture, Climate Change etc), identify data sources for your modeling (e.g. open government portals). The task is to perform knowledge engineering by creating a domain ontology from this data, which then you use to populate the KB by creating instances of entities and the relations. Subsequently, building a query interface to enable users access your KB through key words / Natural Language.

HINTS:

1. Ontology Modeling knowledge can be obtained through the video lecture materials I shared with you.
2. You may need modeling tools such as Protege, or WebVowl.
3. Ensure your KB file is dumped as a .ttl (turtle file)
4. Ensure your KB is expressive by implementing OWL elements and reasoning capabilities
5. Some of the questions your portal should answer must include inference questions.
6. Your interface should build a SPARQL query based on the keyword the user gives to search the KB.

# **Introduction**

Agriculture, a vital sector for global sustenance, faces challenges posed by climate change. The focus of this project is exploring the intricate relationship between crop production and climate change.

The aim of this project is to investigate how climate change affects the crop production using sample data from two African countries: Angola and Kenya. The project uses publicly available datasets from FAO (Food and Agriculture Organization) and World Data Bank to analyse the crop yield of maize and wheat from 2002 to 2013. The project also uses an ontology-based approach to model the domain knowledge and enable natural language queries and responses.  
The core objectives of this project is:

* To construct a comprehensive knowledge base (KB) that captures the relationship between crop yield and climate variables.
* To develop an intuitive query interface to facilitate seamless interaction for users with this knowledge repository.

The set of data used provides data on crop yields, location, year, harvest yield, average rainfall per year, pesticides used per year and average temperature per year.

**Sample Data**

A screenshot of a computer

Description automatically generated

# **Inference Questions Analysis for Agriculture Dataset:**

* **Crops in Kenya**:

This question aims to explore the diversity of crops cultivated in Kenya. By querying the ontology, we can provide a comprehensive list of crops grown in Kenya during the specified time frame (2002 to 2013). This information is crucial for understanding the agricultural landscape in Kenya and can be used to identify trends and patterns.

* **Which crop has the highest yield in Kenya in 2010?**

This question aims to analyse the crop yield in Kenya in the year 2010. By identifying the crop with the maximum yield (measured in HG/HA), insights about the most productive crop during that specific year can be drawn.

* **How does the average temperature affect the yield of Wheat in Angola?**

This question delves into the relationship between weather conditions and crop yield, specifically focusing on the impact of average temperature on Wheat production in Angola. By analysing the correlation between these variables over the years, we can provide insights into how temperature fluctuations influence Wheat yields. This information can be vital for developing strategies to mitigate the effects of climate change on crop production.

* **Which countries use more than 50 units of pesticide on Wheat in 2010?**

This question examines the Pesticide Usage data for the year 2010 and filter countries where the pesticide usage on Wheat exceeds 50 units. This analysis can reveal regions with intensive pesticide practices, raising awareness about potential environmental and health implications. Additionally, it provides valuable information for policymakers aiming to regulate pesticide usage.

* **What is the Maize production in Kenya in 2013?**

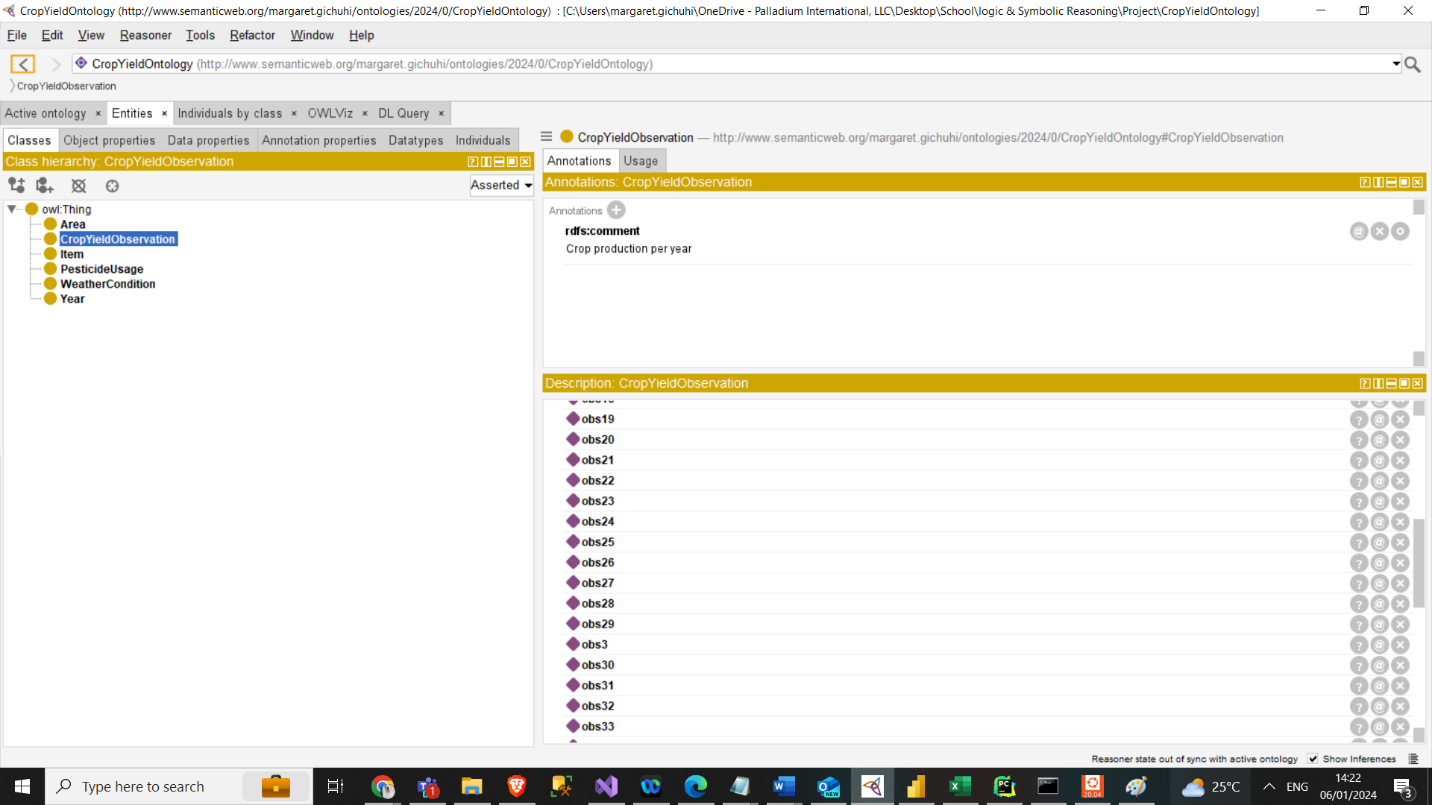
This question focuses on a specific crop (Maize) in a particular country (Kenya) and a specific year (2013). By querying the ontology for Maize production data in Kenya during 2013, we can provide accurate information on the quantity of Maize harvested. This data is essential for assessing food security, identifying surplus or deficit, and aiding in strategic planning for the agricultural sector.

# **Solution Overview**

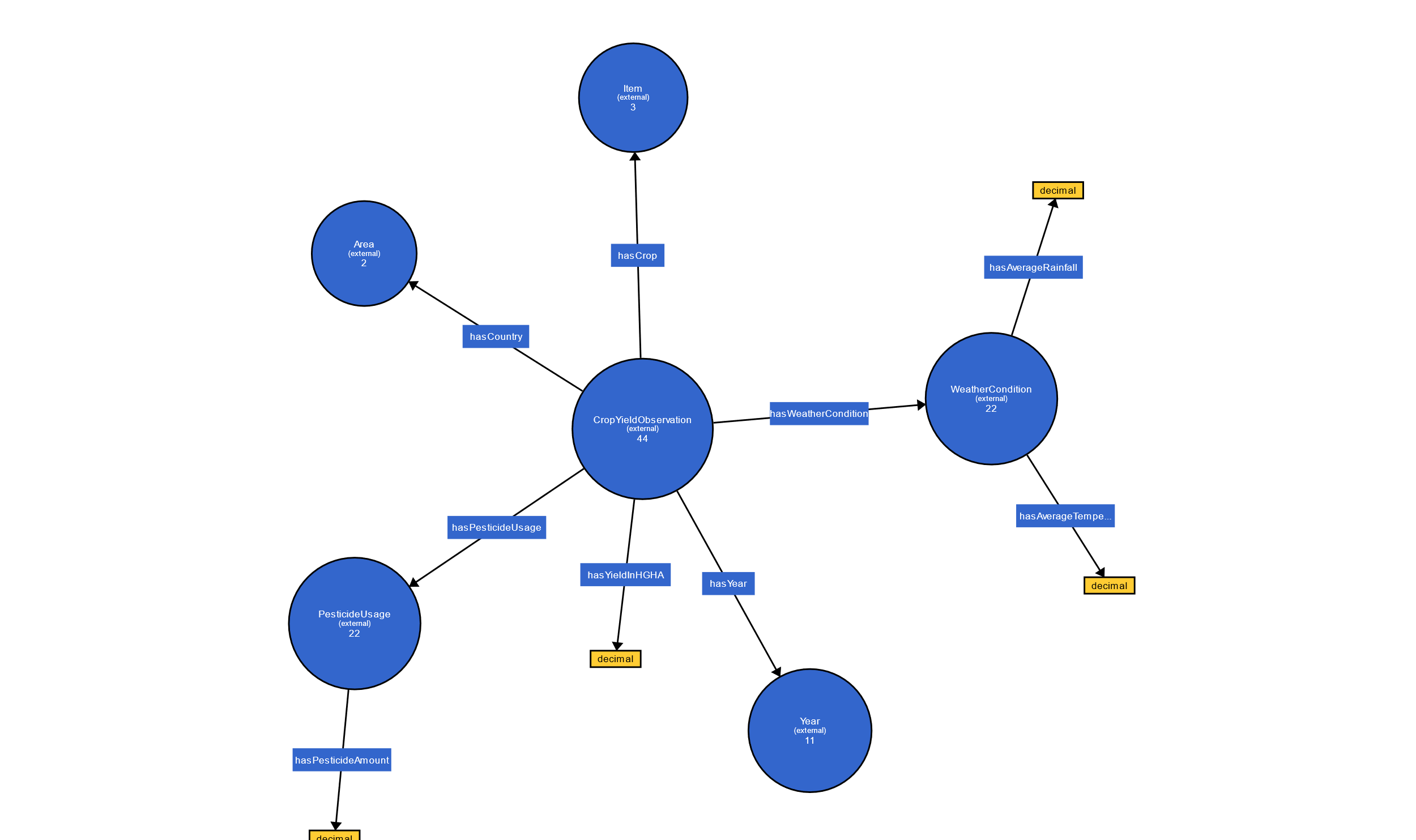
The solution proposed in this project is to use an ontology-based approach to model the domain knowledge and enable natural language queries and responses. An ontology is a formal representation of the concepts and relationships in a domain, which can be used to store, share, and reason about the domain knowledge. An ontology can also facilitate natural language processing and generation, by providing a semantic structure and vocabulary for the domain.

The project uses Protege, an open-source ontology editor, to create an ontology for the domain of crop production and climate change. The ontology defines the following classes, object properties which link the CropYieldObservation class to the other classes and also defines the data properties which assign numerical values to the instances of the classes.

|  |  |  |
| --- | --- | --- |
| Classes | Object Properties | Data Properties |
| Area | hasPesticideUsage | hasAverageRainfall |
| CropYieldObservation | hasCountry | hasAverageTemperature |
| Item | hasCrop | hasPesticide |
| PesticideUsage | hasWeatherCondition | hasYieldInHGHA |
| WeatherCondition | hasYear |  |
| Year |  |  |



The ontology is written in OWL (Web Ontology Language), a standard language for creating ontologies on the web. The ontology is saved as a Turtle file, a syntax for expressing RDF (Resource Description Framework) data. The following figure shows a visualization of the crop yield ontology using WebVOWL web application.



The project uses Blazegraph, an open-source graph database, to store and query the ontology and the data. Blazegraph supports SPARQL, a standard query language for RDF data. The project also uses the SPARQL query endpoint provided by Blazegraph, which allows users to execute SPARQL queries over the web using natural language.

The project uses Python Flask, a web framework, to build a query interface that enables users to access the knowledge base through keywords or natural language. The project also integrates a natural language processing (NLP) library, spaCy, to perform entity recognition and intent extraction. The project also implements a natural language query formalization (NLQF) system, which translates the user query into a SPARQL query, and a natural language generation (NLG) system, which generates a meaningful response from the query results.

A screenshot of a computer

Description automatically generated

A screenshot of a search results

Description automatically generated

# **Data Model**

The data model of the project consists of the ontology and the data. The ontology defines the schema of the domain, while the data provides the instances of the domain concepts and their properties. The data is extracted from two sources: FAO and World Data Bank. The FAO data provides the crop yield of maize and wheat for Angola and Kenya from 2002 to 2013, measured in hectograms per hectare (hg/ha). The World Data Bank data provides the average rainfall and temperature for Angola and Kenya from 2002 to 2013, measured in millimeters per year (mm/year) and degrees Celsius (°C), respectively. The data also provides the pesticide usage for Angola and Kenya from 2002 to 2013, measured in kilograms per hectare (kg/ha).

The data is transformed into RDF triples, which consist of a subject, a predicate, and an object. The subject and the object are the instances of the ontology classes, while the predicate is the instance of the ontology properties. For example, the following triple represents the crop yield observation for maize in Angola in 2002:

|  |
| --- |
| <http://www.semanticweb.org/margaret.gichuhi/ontologies/2024/0/CropYieldOntology/hasYear> <http://www.semanticweb.org/margaret.gichuhi/ontologies/2024/0/CropYieldOntology#2002> ;  <http://www.semanticweb.org/margaret.gichuhi/ontologies/2024/0/CropYieldOntology/hasYieldInHGHA> 15455 ; |

The RDF triples are stored in Blazegraph, which assigns a unique graph identifier (URI) to each triple. The URI can be used to access and query the triple over the web.

# **Process Workflow**

The process workflow of the project consists of the following steps:

* Data collection: The project uses the data from FAO and World Data Bank saved as CSV files.
* Ontology creation: The project uses Protege to create an ontology for the domain of crop production and climate change and saves it as a Turtle file.
* Data loading: The project uses Blazegraph to load the ontology and the data into the graph database and assigns a URI to each triple.
* Query interface development: The project uses Python Flask to develop a web-based query interface that allows users to enter keywords or natural language queries.
* Natural language processing: The project uses spaCy to perform entity recognition and intent extraction on the user query and identifies the relevant concepts and relationships in the ontology. Spacy can perform various NLP tasks, such as tokenization, lemmatization, parsing, tagging, named entity recognition.
* Natural language query formalization: The project uses Python to translate the user query into a SPARQL query, based on the ontology schema and the identified entities and intents.
* Query execution: The project uses Blazegraph to execute the SPARQL query over the graph database and returns the query results.
* Response display: The project uses Python Flask to display the response on the query interface.