# INTRODUCTION

As the trend nowadays to try making machines more intelligent [1], sharing knowledge of information instead of sharing the raw data in the web is becoming more desirable. Thinking computers that are able to able to understand, sharing knowledge and simulate the reasoning process of human would seems an idea from Artificial Intelligent (AI) fiction movie. Semantic web is helping in converting the current web of information into a web of knowledge. It is all about sharing knowledge, which is understandable for machines, on the web. Knowledge of a concept domain is been captured and represented according to our understanding within a file called ontology. Ontologies are considered the main pillar of the semantic web. Computers do not understand information stored on the web such as Extensible Markup Language (XML) and Hypertext Transfer Protocol (HTML). They are just codes to the machines and they display it to users regardless of what knowledge needed. So, Sematic web came along for machines to make sense of retrieved information. Ontologies are used to represent knowledge and make inferences from that knowledge using machines computational capabilities and some reasoning techniques such as description logics.

An intelligent way of representing knowledge needs an intelligent way of browsing and retrieving it. There are a lot of intelligent browsers that is ontology driven user interfaces such as Transparent Access to Multiple Bioinformatics Information Sources (TAMBIS) [2], Semantic Webs and AgentS in Integrated Economies (SEWASIE) [3], and the Manchester Pizza Finder [4]. Browsing and constructing queries through such user interfaces would be easy and it will save time, due to the fact that the UI acts as an interactive manual. It eases the process of constructing the intended query since the process itself is guided be the UI. In addition, it saves the user time by displaying only what is the system intended to do. The user does not need to have previous knowledge about the domain, because the explicit display of the options of constructing a query. Ideas like manuals and the help menu in the menu bar of a UI would seem absolute comparing to the self-guided UI. Additional technique to make the UI smart is to user faceted browsing. The idea behind faceted browsing is to personalize the search and get more specific results by suggesting some filters. Faceted browsing is very related to ontology driven UI since both provide some information about the query while been constructed [5].

There are some systems, that are ontology driven UI, exist such as TAMBIS and SEWASIE. TAMBIS is a system that gather and analysis bioinformatics information from different sources through one interactive UI. While SEWASIE meant to access multiple sources of data and help user through out constructing the exact needed query.

The idea of ontology driven UI is not new. In this project, will try to build an application on the top of existing tool (the Manchester Pizza Finder) with new functionalities and enhancements. The new tool is called the Manchester Sushi Finder; since it is build mainly for sushi ontology that was previously developed by Ontology Engineering course unit. This does not mean the tool will run only sushi ontology, but it can run ontologies with similar structure, concept domain and have specific configurations.

## Aims:

The aim of this project is to investigate and demonstrate the benefits of using OWL ontologies and OWL API within ontology driven UI application as shown in the project page [6]. As well as, making the process of checking and testing ontology easier for students by uploading their ontologies, this will be shown by implementing a configurable and flexible UI. So that most of the configurations will lay in the ontology file, and the UI could browse other ontologies that contain some specific configurations as annotations. The application is called the Manchester Sushi Finder, where a user can construct queries to search for sushi based on included and excluded ingredients defined in conceptual model represented in ontology file.

## Objectives:

To achieve the aim of the project, the aim is divided into several of objectives. These objectives are:

* Gather project requirements.
* Increase the reusability of the UI by making it configurable to suite content of other conceptual models.
* Increase the usability of the UI by showing the languages available in the ontology with their percentage and switch between them.
* Increase the accessibility of the system by applying filters on the content of the conceptual model or/and on the result of the search query. By introducing the notion of filters and facets search to access more specific information.
* Increase the accuracy of the system, so users can only construct valid queries and they get the intended results. Making the UI driven by ontology and using the faceted browsing along with will increase the accuracy of what needed to be queried.
* Provide more flexible system by saving most of the configurations as annotations within the ontology itself.
* Represent the constant of the conceptual model with different views such as tree, and list. Users have more one option to view the content of the model.

## Limitations:

During the implementation of the tool, some limitations were faced. Some of these limitations are related to the user and some related to the OWL. These limitations are:

* Annotating the ontology limitation: since it is one of the requirements for an ontology to work to have certain annotations. It is our nature as human to make mistakes. User may accidently annotate the ontology with wrong information or misspell the annotations’ values.
* Annotation in OWL limitation: as a part of the project to annotate ingredient with filter annotation to make them appear in the filter panel. But a class expression (anonymous class) in OWL cannot be annotated and must be named class subclass of ingredients class.
* Facets limitation: clarity regarding facets is an issue. User could find it difficult to have different facets like hot, medium and mild in pizza ontology. So hot pizza has any hot topping, and medium pizza could has hot and mild topping. As a result medium pizza could be both hot and medium.

## Contributions of this Project:

This project was undertaken to

## Structure of the Dissertation:

1. Ding, L., et al., *Using Ontologies in the Semantic Web: A Survey*, in *Ontologies*. 2007, Springer US. p. 79-113.

2. Stevens, R., et al., *TAMBIS: Transparent Access to Multiple Bioinformatics Information Sources.* IBM System Journal. **40**(2): p. 532-551.

3. Catarci, T., et al., *An Ontology Based Visual Tool for Query Formulation Support.* ECAI, 2004: p. 308-312.

4. Horridge, M., *The Manchester Pizza Finder*. University of Manchester.

5. Bechhofer, S. and N.W. Paton, *Ontology Visual Querying*, in *Encyclopedia of Database Systems*. 2009, Springer.

6. Bechhofer, S. *The Manchester Sushi Finder - Project Page*. 2014 [cited 2014 March 5, 2014]; Available from: <http://studentnet.cs.manchester.ac.uk/pgt/2013/COMP60990/project/projectbookdetails.php?projectid=20889>.