Islamic University of Gaza
Deanship of Research and Graduate Studies
Faculty of Information Technology
Master of Information Technology

الجـامعـــــــة الإســـلامـــــية بغـزة
عمادة البحث العلمي والدراسات العليـا
كليـــــة تكنـولوجيـــا المعلومـــــات
ماجستير تكنولوجيـــا المعلومـــــات

An Ontology-Based Decision Support System for Judging the Social Cases

نظام دعم القرار بالاعتماد على الأنطولوجيا للحكم على الحالات الاجتماعية

By

Mohamad Suliman Salem Hamada

Supervised By

Dr. Rebhi S. Baraka

Associate Professor of Computer Science

A thesis submitted in partial fulfilment
Of the requirements for the degree of
Master of Information Technology

February/2018

إقـــــــــرار

أنا الموقع أدناه مقدم الرسالة التي تحمل العنوان:

An Ontology-Based Decision Support System for Judging the Social Cases

نظام دعم القرار بالاعتماد على الأنطولوجيا للحكم على الحالات الاجتماعية

أقر بأن ما اشتملت عليه هذه الرسالة إنما هو نتاج جهدي الخاص، باستثناء ما تمت الإشارة إليه حيثما ورد، وأن هذه الرسالة ككل أو أي جزء منها لم يقدم من قبل الآخرين لنيل درجة أو لقب علمي أو بحثي لدى أي مؤسسة تعليمية أو بحثية أخرى. وأن حقوق النشر محفوظة للجامعة الإسلامية – غزة.

Declaration

I hereby certify that this submission is the result of my own work, except where otherwise acknowledged, and that this thesis (or any part of it) has not been submitted for a higher degree or quantification to any other university or institution. All copyrights are reserves to IUG.

Student's name:		اسم الطالب:
Signature:		التوقيع:
Date:		التاريخ:

الرقم: Ref: ج س غ/35/ 2018/03/05م

التاريخ: Date:

نتيجة الحكم على أطروحة ماجستير

بناءً على موافقة عمادة البحث العلمي والدراسات العليا بالجامعة الإسلامية بغزة على تشكيل لجنة الحكم على أطروحة الباحث/ **محمد سليمان سالم حمادة** لنيل درجة الماجستير في كلية **تكنولوجيا المعلومات**/ قسم **تكنولوجيا المعلومات** وموضوعها:

نظام دعم القرار بالإعتماد على الأنتولوجيا للحكم على الحالات الاجتماعية

An Ontology-Based Decision Support System for Judging the Social Cases

وبعد المناقشة التي تمت اليوم **الاثنين 17 جمادي الثانية 1439**هـ الموافق **2018/03/05م** الساعة **الثانية ظهراً** ، في **قاعة اجتماعات الكلية** اجتمعت لجنة الحكم على الأطروحة والمكونة من:

د. ربحي سليمان بركة	مشرفاً ورئيساً	.....
د. إياد محمد الأغا	مناقشاً داخلياً	.....
د. ناجي شكري الظاظا	مناقشاً خارجياً	.....

وبعد المداولة أوصت اللجنة بمنح الباحث درجة الماجستير في **كلية تكنولوجيا المعلومات/قسم تكنولوجيا المعلومات.**

واللجنة إذ تمنحه هذه الدرجة فإنها توصيه بتقوى الله تعالى ولزوم طاعته وأن يسخر علمه في خدمة دينه ووطنه.

والله ولي التوفيق،،،

عميد البحث العلمي والدراسات العليا

أ. د. مازن إسماعيل هنية

نموذج رقم (4)

14

المواد المصاحبة : CD 3/0.6414

التاريخ: 16/5/2018 م

الموضوع/ مطابقة مواصفات النسخة الإلكترونية

بعد الإطلاع على الأسطوانات التي تحتوي على رسالة الطالب/ محمد سليمان عبد الله العجاجة

رقم جامعي: 120130561 كلية: الدراسات العليا قسم: .....

فإننا نحيطكم علماً بأنها مطابقة للمواصفات المطلوبة المبينة أدناه:

جميع فصول الرسالة في ملف (WORD) واحد وليست ملفات متفرقة.

تحتوي الأسطوانة على ملف (PDF + WORD).

مطابقة التنسيق في جميع الصفحات (نوع وحجم الخط) بين النسخة الورقية والإلكترونية.

مطابقة النص في الصفحة الورقية مع النص في الصفحة الإلكترونية لجميع صفحات الرسالة.

ملاحظة: ستقوم عمادة المكتبات بنشر الرسالة العلمية كاملة (PDF) على موقع المكتبة.

والله ولي التوفيق،

توقيع الطالب

.....

توقيع المكتبة المركزية

.....

Abstract

PNPSP is considered the biggest program in Gaza Strip for providing social cases with cash assistance. It supports 20% of the people in Gaza Strip. It depends on the criteria of the social protection with regard to poverty line in deciding who deserves cash assistance.

Decision making in such social domain using the conventional ways leads sometimes to unaccurate decisions. This is because, firstly, it is up to the researcher decision (estimation), which may differ from time to time, secondly, because poverty criteria needs to be updated according to changing life costs, and thirdly because the amount of cash available to PNPSP makes decisions variable and changeable on who deserves cash assistance and how much he deserves. Semantic web technologies and techniques using domain ontology for organizing and inferring knowledge can contribute in improving decision making in the social domain.

In this research, we build an ontology based system for helping social researchers to take the best and the accurate decisions in cash assistance. The system is meant to standardize decision making on social cases leading to trusty and fair decisions. The system concentrates on the criteria of the social protection for the social cases and the related programs of these cases, where the criteria is translated into SWRL rules to infer new knowledge from the knowledge base.

A system prototype is designed and implemented as a prove of concept including the domain ontology; the SWRL rules; the knowledge base; the reasoner and the user interface. We evaluated the proposed system using 30 cases as a sample set with the family data provided by a social domain expert. The results show that the system has correctly judged 26 out of the 30 case with accuracy 86.7%.

Keywords: Semantic Web, Ontology, Social Protection Decision, SWRL Rule, Semantic Querying, Poverty criteria.

الملخص

يعتبر البرنامج الوطني الفلسطيني للحماية الاجتماعية أكبر برنامج في قطاع غزة لتقديم المساعدة المالية للحالات الاجتماعية. حيث يدعم ما نسبته 20٪ من سكان قطاع غزة. حيث يعتمد على معايير الحماية الاجتماعية وخط الفقر.

اتخاذ القرارات في المجال الاجتماعي باستخدام الطرق العادية يؤدي أحيانا إلى قرارات غير دقيقة. ويرجع ذلك في المقام الأول إلى قرار الباحث (تقديره) الذي قد يختلف من وقت لآخر، وثانيا لأن معايير الفقر تحتاج إلى تحديث وفقا لتغير تكاليف الحياة، وثالثا لأن مقدار النقدية المتاحة للبرنامج الوطني للحماية الاجتماعية يجعل القرارات متغيرة وقابلة للتغيير على من يستحق المساعدة المالية وكم يستحق. يمكن للتكنولوجيا وتقنيات الويب الدلالية باستخدام علم الأنماط في مجال تنظيم المعرفة واستنتاجها أن تسهم في تحسين عملية صنع القرار في المجال الاجتماعي.

في هذا البحث قمنا ببناء نظام قائم على الأنطولوجيا لمساعدة الباحثين الاجتماعيين على اتخاذ أفضل وأعدل القرارات في المساعدة المالية. ويهدف هذا النظام إلى توحيد عملية اتخاذ القرارات بشأن القضايا الاجتماعية التي تؤدي إلى اتخاذ قرارات موثوقة ودقيقة. يركز النظام على معايير الحماية الاجتماعية للحالات الاجتماعية والبرامج ذات الصلة في هذه الحالات، حيث يتم ترجمة المعايير إلى قواعد الويب الدلالي لاستنتاج المعرفة الجديدة من قاعدة المعرفة.

تم تصميم نموذج أولي للنظام وتطبيقه كإثبات للمفهوم، بما في ذلك علم الأنماط، وقواعد الويب الدلالي، وقاعدة المعارف، والسبب، وواجهة المستخدم. قمنا بتقييم النظام المقترح باستخدام مجموعة نموذجية مع البيانات الأسرية المقدمة من خبير المجال الاجتماعي. أظهرت النتائج أن النظام حكم على 26 من أصل 30 حالة بشكل صحيح بدقة 86.7 ٪.

كلمات مفتاحية: الويب الدلالي، انطولوجيا، قرار الحماية الاجتماعية، لغة قواعد الويب الدلالي، الاستعلام الدلالي، معايير الفقر.

Dedication

To spirit of my Father

To my beloved Mother

To my dear Wife

To my sweet Children

To my Brothers and Sisters

To my Friends

Acknowledgments

I thank Allah for giving me the ability to complete this thesis.

I would like to thank my supervisor, **Dr. Rebhi Baraka**, who guided and advised me throughout my thesis work. I have been extremely lucky to have a supervisor who cared so much about my work and who responded to my questions and queries so promptly.

I should thank the domain experts and social researchers for helping me to obtain the necessary information in the social field and helped in giving expert comments and doing some evaluations.

My sincere thanks to my father, who taught me helping people and appreciate the value of education.

I would also like to thank so much my mother for encouragement to complete this thesis.

In addition, I thank my wife for her support and encouragement during the thesis, without her encouragement, I would not have completed this work.

I also thank my brothers and sisters, specially my older brother and sister; they are the first people who encouraged me to obtain a master degree.

Mohamad S. Hamada

Feb / 2018

Table of Contents

List of Tables

List of Figures

List of Abbreviations

KB	Knowledge Base
MOSD	Ministry of Social Development
OWL	Web Ontology Language
PNPSP	The Palestinian National Program of Social Protection
RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
SPARQL	SPARQL Protocol and RDF Query Language
SocCaseOnt	Social Cases Ontology
SW	Semantic Web
SWRL	Semantic Web Rule Language

Chapter One

Introduction

Chapter One
Introduction

Social domain is one of the most important and sensitive domains. It is complex and consists of a number of decision rules, variables and various cases.

Social researchers who receive and decide in the social cases sometimes face difficulties searching in the criteria and looking for similar cases to accept or reject the current case that regards finding multiple cases with different variables.

The manger holds committee to judge social cases, which wastes a lot of time for giving judgments on the various cases and comparing them to similar cases to support the decision on current cases.

Recent concepts and tools such as Semantic Web and ontology can deal with this domain. Ontologies are a concept of Semantic Web that can be used in many applications such as information retrieval and decision support systems (Kalfoglou, 2007).

There are many studies dealing with ontology as approach for giving decisions in various domains such as Legal domain (Taylor, 2013), Health Advice Derivation in medical (Izumi et al., 2007) (Satoru Izumi, 2007), Diagnosing Date Palm Diseases in agriculture (El-Askary, 2015), Arabic Question answering in NL (AbuTaha, 2015). A method for merging logic rules in ontologies represents humanitarian needs and recommends good responses among a crisis. The continues processing of crisis data can provide necessary information to the decision makers (Jihan & Segev, 2014).

This research proposes to:

- Develop ontology for social domain to construct a knowledge base with rules depending on the social protection criteria to decide on the social cases, which related to PNPSP.
- Develop ontology-based system for helping the social researchers and decision makers to give the best and the fair decision on various social cases.

The rest of this chapter presents the statement of the problems, the objectives of the research, the importance of the research, scope and limitations, methodology and finally thesis structure.

1.1 Statement of the problem

The social domain is one of the important and sensitive domains. The Palestinian National Program of Social Protection (PNPSP) includes approximately 73,000 social cases (families) with an average of five persons in every family in Gaza strip, which means that PNPSP serves approximately 400,000 persons. Searching and working in such domain must guarantee the correct knowledge retrieval and the establishment of new relations to be a bases for correct and convincing social financial assistance decisions.

There is a need for an ontology as well as rule -based knowledge system that represents the social domain in Palestine to help social researchers draw correct and justified judgments on social cases as whether a case deserves social financial assistance or not.

Using ontology as well as semantic rules to build a knowledge base would formally and completely capture this sensitive social domain and work as a common and sharable understanding for the domain among social experts to guarantee decision fairness and correctness. The ontology as a recent and rigorous notion of the semantic web and artificial intelligence has proved to be so effective in decision support systems and has shown to be interoperable with other dependent and related systems.

1.2 Objectives

1.2.2 Main objective

To build an accurate ontology-based system for helping social researchers to judge on the social cases as whether to receive social cash assistance or not.

1.2.3 Specific objectives

- To collect data about social domain by studying the standards, which deal with the living, learning and economic condition of the family based on appropriate forms to be filled by the social researchers.
- To build a specific domain ontology and knowledge base based on the collected data.

- To make social decisions for social cases by transforming used criteria to suitable reasoning rules to be used with ontology and the built knowledge base.
- To develop a system for helping the realizing the above objectives as a proof of concept to integrate the ontology, the knowledge base as well as the rules and with user interface.
- To evaluate the ontology and the system with a number of experimental results on social cases and to compare them with the human experts decisions.

1.3 Importance of the research

The social domain is one of the important domains and there is a need to focus on it due to its big effect on the people life. It is considered a rich and complex domain because it contains various cases and rules.

The Palestinian social situation is complex and is hard to measure because it changes rapidly from time to time. Therefore, it needs suitable and adaptive approach such as ontology-based one to deal with it.

In this research we developed ontology for social domain to construct a knowledge base with rules depending on the social protection criteria to decide on the social cases, which related to PNPSP. Also, we developed ontology-based system for helping the social researchers and decision makers to give the best and the fair decision on various social cases.

The proposed system makes selecting among the social cases more fair, trusty and justified. It will be one of the few studies looking at the possibility of creating a knowledge base for social domain based on conceptual ontology to judge on social cases. The previous works have used traditional programming ways that give less trusted justified decisions.

The system can be useful in the Government, UNRWA and the private institutions related to social domain. It will help the social researcher to provide the best and the fair social judging on the social cases.

1.4 Scope and limitations of the research

This research is limited to the social protection domain, which serves the social cases (families) in Gaza Strip within **PNPSP** and it covers the most common social cases based on the agreed upon and predefined social protection criteria. The system is not meant to replace the social researcher, rather it is expected to produce decisions as a support to the social researcher to accept or reject the case based on the social protection.

The ontology covers only the social cases in the cash assistance part found in Gaza Strip as a domain with a number of poverty cases as instances to initiate a knowledge base. The decision making will be based on SWRL rules reflecting the well-known and agreed upon social criteria.

A system prototype will be developed and evaluated through a number of experiments and comparing the resulting decisions to those of the experts for measuring the accuracy.

1.5 Methodology

To accomplish the objectives of the research, the following research methodology is followed:

Phase 1: Data collection and preparation: data and information are collected through Social Data and Social Protection criteria. This is performed through the data from and the practices of the PNPSP.

Phase 2: Domain Analysis and Modeling: model the social protection system using Unified Modeling Language (UML) to understand the overall view of social protection domain. The proposed ontology is expected to cover the main classes and properties of this domain (shown in Figure 1.1).

Phase 3: Ontology Development: model the social protection domain knowledge and represent it in a conceptual form, define the concepts and relationships between concepts. By using a tool named Protégé, we convert it into OWL language. Also, we follow the ontology development process as follows:

1. Determine the domain and scope of the ontology.

2. Consider reusing existing social domain ontology and extend it as an option.
3. Indicate the important terms in the ontology.
4. Define the classes and the properties of classes.
5. Define the facets (Cardinality restrictions, required values, Relational characteristics).
6. Create instances from (DB or text corpus).
7. Use Reasoner to get new knowledge for judging on the social cases.
8. Use some SPARQL to perform quires on the ontology that ensure the correct building of ontology and check whether it returns what we want.

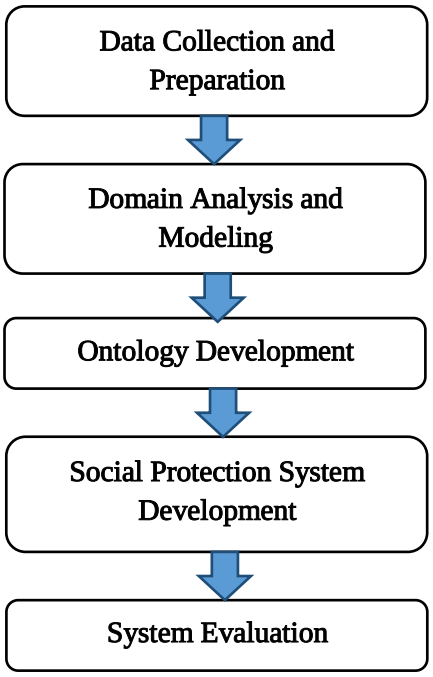


Figure (1.1): System Stages

Phase 4: System Development: we develop a social case system, which includes the social assistance knowledge base. This system will be ontology based with interface for selection of cases for humanitarian aid as in the following steps:

- Specify the requirements of system.
- Design the system.
- Implement the prototype.
- Test the prototype.

Phase 5: Evaluation of the system: we perform a set of experiments and evaluate the system by entering a number of social cases and comparing the results of the cases to those of a human expert.

1.6 Thesis Structure

This thesis consists mainly of seven chapters as follows:

- **Chapter 1 (Introduction):** introduces the social domain, Methodology and the thesis problem and objectives.
- **Chapter 2 (Theoretical and Technical Foundation):** describes the theoretical and technical background needed for the research, ontology concepts, RDF, RDFs, SWRL rules.
- **Chapter 3 (Related Work):** presents works related to judging the social cases, which use ontology to support the decisions making.
- **Chapter 4 (Cash Assistance Decision Support Ontology):** presents the development of the social domain ontology.
- **Chapter 5 (Cash Assistance Decision Support System):** describes the design and implementation of the system prototype which uses the Cash Assistance Decision Support Ontology for taking decision related to cash assistance.
- **Chapter 6 (Experimental Results and Evaluation):** presents an evaluation of system by performing a number of experiments on the system.
- **Chapter 7 (Conclusions and Future Work):** presents the conclusions and the Possible future works.

Chapter Two

State of the Art

Chapter Two
State of the Art

In this chapter, we present the state of the art for our research. It covers fundamental concepts and technical knowledge related to the notion of ontology including its development and evaluation process as well as rules and reasoning used to extract and query for knowledge from the ontology. Also, we present works in various domains that use Semantic Web and Ontology to build knowledge base to support decision making. We present and discuss works divided on the social domain, the health domain, the legal domain, and the business domain. First, we present an overview of the social domain and the cash assistance program in Palestine since it is the main focus of our study and is considered the domain of the ontology to be constructed later in the thesis.

2.1 Palestinian Cash Assistance Program

Palestinian society is suffering from a decline in living standards and the deterioration of livelihoods, the percentage of poverty among individuals in 2011 according to monthly consumption patterns was 25.7% (18.3% in the West Bank and 38.0% in the Gaza Strip). While 48.6% of Palestinian individuals have less monthly income than the poverty line (36.2% in the West Bank and 69.3% in the Gaza Strip).

It was also found that about 14.1% of Palestinian individuals suffer from extreme poverty according to monthly consumption patterns (8.8% in the West Bank and 23.0% in the Gaza Strip). On the income level, it was found that about 37.6% of Palestinian individuals have a monthly income below the extreme poverty line (24.6% in the West Bank and 59.2% in the Gaza Strip).

Social Protection domain has a large and fragmented range of governmental and non-governmental institutions and the private and international institutions alike. This was the first attempt of its kind and which culminated in the completion of the social protection domain strategy responsibility of the (MOSD), in partnership actors working in the social protection domain. It deals with similar issues in the broad sense, including services for the poor and the marginalized and vulnerable and is extended to include social insurance issues(MOSD, 2013).

The Palestinian Cash Assistance Program is mainly based on targeting the poor families. A family eligibility is determined by applying an equation after examining the

ways of living based on surveys of spending and consumption carried out by the Palestinian Central Bureau of Statistics. It is intended for the family and all individuals living in the house and sharing in the food, drink and spending regardless of family size.

The social researcher fills in a form of an initial interview with the head of the family. The interview relies on information provided by the family and an initial paperwork.

Family data is dumped on a computer program which in turn calculates (based on the equation) the means of subsistence for the family and then determines the amount without any human intervention, relying on the state of poverty of the family and their needs.

The real amount each beneficiary family gets of the program is ultimately determined according to the total budget available for each assistance program. It currently covers 50% of the family needs of cash assistance.

The minimum amount a family gets is NIS 750 every three months and the maximum amount is NIS 1800 every three months. The social researcher does not intervene in determining the amount of increase or decrease. The amount is determined by computer program according the poverty equation.

2.2 Theoretical and Technical Foundations

In this section, we present and discuss the following concepts and technologies: Ontology, Ontology Engineering, SWRL Rules, Ontology Evaluation and Tools

2.2.1 Ontology

The notion of the ontology is strongly related to the Semantic Web (SW), which is an extension of the current Web for sharing the content beyond the boundaries of applications and websites (Jain & Singh, 2013a). Adding logic to the current Web is the main goal of the Semantic Web, i.e. expressing the meaning of data, the properties of objects, and the complex relationships existing between them by a series of formal rules, which can make information more accessible to the machines. Machine accessibility should be understood as representing information in such a way that it is possible to make queries depending on the meaning (i.e., semantics) of the data,

separate of the form in which the information is presented (Robu, Robu, & Thirion, 2006). SW depends greatly on the notion of ontology to achieve much of its functionality.

The nearest Ontology definition is "*a formal explicit description of concepts in a domain of discourse (classes). Properties of each concept describe various features and attributes of the concept (slots), and restrictions on slots (facets) ontologies together with a set of individual instances of classes constitutes a knowledge base*" (V. Jain. et al., 2013).

Ontology is basic of the Semantic Web and allow agent applications to communicate effectively. The ontology can represent clearly the concepts and inter concept relationships and can be used as a framework to represent implied domain concepts in different languages. In our research, we have a relation between criteria and the social case, so we need the power ontology to get more accurate results (Abusalah, Tait, & Oakes, 2009).

Ontologies is useful in the medical field in the diagnosis of plant diseases and in judging on social cases. It can used as a shared a common understanding of the structure of information between users or agents. It allows the reuse of knowledge and the development of explicit assumptions on the ground, which is generally implied (Noy & McGuinness, 2001).

A common standard of ontology representation is the Web Ontology Language (OWL). OWL presents a knowledge domain through definition of the structure of classes, terms, properties, individual and restrictions. Individuals represent objects in the domain; can be members of one or more classes. Properties describe the relationships between individuals it may be object property to share two classes or data property to share the class with literal (as integer). Classes, also known as sets members of classes, share some properties or characteristics (W3C, 2012). OWL is designed to show rich and complex knowledge about things, groups of things, and relations between things (Jain & Singh, 2013b).

2.2.2 Ontology Engineering

There are various engineering approaches and methods for constructing an ontology. We consider the ontology engineering process due Grigoris Antoniou (2012) in building the social assistance ontology which consists of: determining the domain and the scope of the ontology, considering reuse of existing ontology, enumerating basic terms, defining the class hierarchy, defining properties and relations among the classes in the hierarchy, defining facets, defining instances to populate the ontology and create a knowledge base, creating semantic rules, and finally applying a reasoner to check the consistency of the ontology. Next, we elaborate in each step since the process is so essential in our ontology development.

1. Determining scope

In general, there is no correct ontology of a specific domain. An ontology is an abstraction of a specific domain. This abstraction includes alternatives that can be applied and used in the near future as it expected for it.

The basic questions to be answered at this stage are:

- What is the domain coverage of the ontology?
- Why we are going to use the ontology?
- What are questions should the ontology answers?
- Who will use and maintain the ontology?

2. Considering reuse

Ontologies will become more widely available with spreading deployment of the Semantic Web. We rarely have to start from scratch when defining an ontology. There is a third party almost can make it easy use an ontology. This provides at least a useful starting point for our own ontology.

3. Enumerating terms

Write down in an unstructured list all the relevant terms that are expected to appear in the ontology

- Nouns form the basis for class names
- Verbs (or verb phrases) form the basis for property names

Traditional knowledge engineering tools (e.g. laddering and grid analysis) can be used to obtain

- the set of terms

- an initial structure for these terms

4. Defining taxonomy

   Related terms must be organized in a taxonomic hierarchy

   - Opinions differ on whether it is more efficient/reliable to do this in a top-down or a bottom-up style (mentioned before)

   Ensure that hierarchy is indeed a taxonomy:

   - If X is a subclass of Y, then every instance of X must also be an instance of Y (compatible with semantics of rdfs:subClassOf

5. Defining properties

   Often interleaved with the previous step. The semantics of subClassOf demands that whenever A is a subclass of B, every property statement that holds for instances of B must also apply to instances of A. It makes sense to attach properties to the highest class in the hierarchy to which they apply.

   While attaching properties to classes, it makes sense to immediately provide statements about the domain and range of these properties. There is a methodological tension here between generality and specificity:

   - Flexibility (inheritance to subclasses)
   - Detection of inconsistencies and misconceptions

6. Defining facets

   In this step we refer to Cardinality restrictions, Required values (owl:hasValue, owl:allValuesFrom, owl:someValuesFrom), Relational characteristics (symmetry, transitivity, inverse properties, functional values).

7. Defining instances

   With a separate step, which is filling the ontologies with such instances

Number of instances >> number of classes

Individuals (instances) are the basic, "ground level" components of an ontology.
The instance in general is greater than the classes because they are example of these classes, the instances show meaning of the classes in practice way to understand the relation between them. The instance can be retrieved from legacy data sources (DBs) or extracted automatically from a text corpus.

SPARQL Protocol and RDF Query Language (SPARQL)

SPARQL is the standardized query language for RDF. SPARQL query construction has been described as “absurdly difficult”, and even experienced users may struggle with it. For this reason, various methods have been suggested for aiding in SPARQL query generation, including assisted query construction (McCarthy, Vandervalk, & Wilkinson, 2012).

SPARQL is the standardized query language for RDF, it is closed to SQL is the standardized query language for relational databases, and it is share several keywords such as SELECT, WHERE, etc.(S. wikipedia, 2013). Also has special keywords that not used in SQL such as OPTIONAL, FILTER and much more as shown in next example.

Data:

```
@prefix foaf:       <http://xmlns.com/foaf/0.1/> .
_:a  foaf:name       "Ali" .
_:a  foaf:homepage   <http://work.example.org/ali/> .
_:b  foaf:name       "Ahmed" .
_:b  foaf:mbox       <mailto:ahmed@work.example> .
```

Query:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?mbox ?hpage
WHERE  { ?x foaf:name  ?name .
       OPTIONAL { ?x foaf:mbox ?mbox } .
       OPTIONAL { ?x foaf:homepage ?hpage }}
```

Query result:

Name	Mbox	hpage
"Ali"		<http://work.example.org/ali/>
"Ahmed"	<mailto:ahmed@work.example>	

2.2.3 SWRL Rules

The Semantic Web Rule Language (**SWRL**) is a proposed language for the Semantic Web that can be used to express rules as well as logic (wikipedia, 2013).

Rules are the form of an implication between an antecedent (body) and consequent (head). We can be read as: if the conditions specified in the body hold, then the conditions specified in the head must also hold. As in the following example the rule states: if someone (w1) has parent (w2) and (w2) is brother of (w3) then we can deduce that (w3) is uncle of (w1).

$$hasParent(?w1,?w2) \wedge hasBrother(?w2,?w3) \Rightarrow hasUncle(?w1,?w3)$$

- Reasoning:

Reasoning is the process that infers logical consequences from a set of asserted facts/axioms. The strength of the reasoning depends on inference rules and the axioms that deduced which are related to a certain kind of logic.

Reasoning is the main and activity key for working with ontologies. All querying of an ontology should be done using a reasoner. This is because knowledge in an ontology might not be explicit and a reasoner is required to deduce implicit knowledge so that the correct query results are obtained.

Examples of the reasoner such as FaCT++, Pellet, HerMiT are required for executing SWRL rules to infer new ontology axioms. Pellet reasoner has best functionality for working with SWRL rules. The example below explains the parents' relation, which deduces the uncle relation.

$$hasParent(?w1,?w2) \wedge hasBrother(?w2,?w3) \Rightarrow hasUncle(?w1,?w3)$$

By running the Pellet reasoner on ontology with SWRL rules, the reasoner takes returns conclusions (inference) based on those rules.
The required system that supports the ontology. There are many tools, for example (Java Netbeans, JSP).

2.2.4 Ontology Evaluation

The evaluation of the quality of ontology is one of the important parts of ontology. There are many criteria for evaluating the ontology such as: the richness; the coverage of a particular domain, complexity and granularity of that coverage; the specific use cases, scenarios, requirements, applications, data sources it was developed to address, formal properties such as the completeness and consistency of the ontology and the representation language in which it is modeled (Obrst et al., 2007). In this

research, we evaluate the system by choosing 30 case of common and uncommon cases, the research relates between the social protection criteria with the cases data for deciding the social cases for making fair decision.

2.2.5 Tools

- **Protégé:** The Protégé-OWL API is an open-source Java library for the Web Ontology Language and RDF(S). It provides classes, properties, query and manipulate OWL data models and SWRL rules to perform reasoning. Moreover it improves the applying of graphical user interfaces (Stanford University, 2015).
- **Pellet reasoner:** it is an open-source Java that can be used in conjunction with both Jena and OWL API libraries. It merges optimizations for nominal, conjunctive query answering, and incremental reasoning we can download and include the reasoner in the applications.

2.3 Related Works

In this section, we present and discuss the following concepts and technologies: Ontology in Social Domain, Ontology in Health Domain, Ontology in Legal Domain and Ontology in Business Domain.

2.3.1 Ontology in Social Domain

In the last few years, there has been an increase for information stored in the knowledge bases and are semantically enriched and represented as ontologies. These improve the accuracy of the search results when queries are semantically formulated (Sitthisarn, Lau, & Dew, 2011). In this research, we focus on the power of using ontology to represent the information of social cases and to make related cash assistance support decision making. Di Maio and Paola (2007) addressed the open ontology methodology for open source emergency response systems. It allows users and developers to collaboratively and dynamically create and support knowledge and semantic consistency for emergency response systems (Di Maio, 2007). The proposed system addresses knowledge base in consistent ontology and retrieves the decision without need to expert consultation when it succeeds.

Fan and Zlatanova (2011) explored the semantic interoperability of the terms and spatial information to be used by different emergency response communities. These

works didn't aim to create a solution based on humanitarian standards to create a bridge between the emergency needs and the crisis response recommendations using an ontology-merging process to represent the knowledge and rule-based techniques to support the decision making process (Fan & Zlatanova, 2011). In our research, we merge between the cases data and social criteria set by PNPSP. (Li et al., 2008) proposed a practical emergency response workflow and emergency response ontology architecture. The proposed ontology's are to standardize semantic concepts that can be applied to various different emergency response systems and to define practical common vocabularies between emergency response personnel. By closely examining a generic emergency response workflow, this paper proposes a practical emergency response ontology. Using this ontology, the authors implement a prototype emergency evacuation planning system. This implementation demonstrates the feasibility and effectiveness of our proposed ontology and its potential to grow into a more completed ontology for collaborative Crisis Information Management Systems and use such ontology to process the social case judging.

(Biletskiy & Vorochek, 2004) described a model for building and integrating ontologies that aim at building a domain of common ontology and use it in the approach of context mediation for assisting decision-making. This happens when delivering data to decision support systems in an acceptable form. The use of ontologies in the approach of context mediation allows reducing a number of context conversions and, as a result, improving performance of the process of mediation. The process of building ontology figured out as follows. First, identification of ontology purpose needed to achieve semantic interoperability among data sources and DSS. Second, pre-processing, Conceptualizations, pre-integration and integrating are important to build the ontology. Third, evaluation and documentation applied.

(Castells, Foncillas, Lara, Rico, & Alonso, 2004) concerted on the field of economy and finance, which is complex and rich domain with information, huge in volume and a highly valuable business product by itself. Novel management techniques are required for economic and financial information in order to enable an efficient generation, management and consumption of complex and big information resources. The authors have developed and ontology-based platform that provides 1) the integration of contents and semantics in a knowledge base that provides a conceptual view on low-

level contents, 2) semantic search facilities and 3) an adaptive hypermedia-based knowledge visualization and navigation system. They developed, as the basis of this platform, an ontology for the domain of economic and financial information.

There are three ideal types that emphasize the ideational dimension of social assistance are analyzed, namely, the entitlement, workfare, and activation paradigms. The aim of the typology is lying in its utility for characterizing the ideational orientation of social assistance systems. The typology provides a criterion to measure the ideas of policy actors with respect to social assistance and can facilitate the conduct of case studies, comparative research, and causal analyses on this policy sector (Daigneault, 2014).

Ontology connects the different communities in the software development to overcome barriers created by disparate vocabularies, approaches, representations, and tools in their respective contexts (Fan & Zlatanova, 2011). The formalization activity transforms the conceptual model into a formal or semi-computable model. Formalization is not obligatory activity because using ontology tools the conceptualization model is usually automatically implemented with translators to ontology languages. The implementation activity builds computable models in an ontology language (Ontolingua), RDF Schema (Brickley & Guha, 2004), OWL (Aguado, Bernaras, Smithers, Pedrinaci, & Cendoya, 2003). Tools are implemented automatically on conceptual models have varieties of ontology languages.

3.2 Ontology in Health Domain

(Agha & Baraka, 2017) used ontology and Semantic Web Rule Language (SWRL). The aim of the research was to elaborate the diagnosis of heart diseases by using Semantic Web technologies. The built domain ontology was called (HeartOnt) that covers domain knowledge of heart diseases. The ontology contents are classes, relationship and properties which are used as a main in the approach. They mainly depend on SWRL rules which created from the relationships between ontology concepts to detect heart disease and estimate the risk of heart disease. In our research, we focus on deciding on social cases by using appropriate classification using the knowledge base and SWRL rules.

Health advice derivation system could take appropriate advice as to the purpose of user and user's health where ontology is introduced into the system to describe knowledge about exercise, meal and user's health. New relationships are on the knowledge found by the deduction system and are eventually provided with advice to the user (Satoru Izumi, 2007). In this research, we try to achieve fair advice and right decision for the social researcher.

(Kato, Maneerat, Varakulsiripunth, Kato, & Takahashi, 2009) developed an e-health system based on ontology and Semantic Web technologies in order to help find suitable Thai herbs to cure diseases in Thailand. Thai herbs are used instead of traditional medical treatment. While, there are various Thai herbs sorts, it is difficult to find a suitable one.

3.3 Ontology in Legal Domain

(Afifi & Baraka, 2017) built a semantically enriched approach for the derivation of legal advice in the labour domain depends on the Palestinian labour law. The built ontology called LabourLawOnt was built with a knowledge base for the labour law, which included terms, instances, relationships and data properties. To form the legal knowledge base, the set of parts, chapters and articles of the labour law with various legal cases were added as instances to the ontology. In addition, a set of semantic (SWRL) rules were used to infer new knowledge from the knowledge base. In our research, we define the instances of social cases and relate them with the social protection criteria which are represented as SWRL rules.

The Palestinian government ontology framework consists of agreed-upon vocabulary (naming), meaning, structure and business. The ontology is built as asset of subontologies (e.g., Legal Person, Address, Organization, Car, Land, etc.). The framework called Zinnar and handles the semantic interoperability issues in the government domain in Palestine.

The framework is composed of five components: Government Ontology, Entities, which contains agreed-upon national classifications and coding and naming schemes, Address (Geo-Entities), where unified addressing data in Palestine is kept, Service Repository, where all governmental services (i.e., business processes) are identified. The repository is also used to publish web services (Jarrar, 2011). In our search, we focus on derivation

a social deciding for users through showing the related programs, which support social case, in addition calculating cash assistance amount for each family.

(Taylor, 2013) made classification system for the group of entities that express the relationships between the classes of the legal cases as cases codes, related legal cases and the articles that explorer cross-references their differences and the like. In our research, we make good classifications based on the components of the social criteria and the provided social programs.

(Baraka & Dalloul, 2014) built an ontology-based Isnad Judgment System (IJS) that automatically generates a suggested judgment of Hadith Isnad. It is based on the rules that Hadith scholars follow to produce a judgment. A prototype of the approach is implemented to provide a proof of concept for the requirements and to verify its accuracy. The system was evaluated according to Al-Albani scholar and according to Hadith specialist.

3.4 Ontology in Business Domain

In the e-business domain, ontology is used to build the e-catalog system for e-business. An organization has a good command of the knowledge of its markets, customers, products and services, methods and processes, competitors and employee skills. Product information consists of many attributes and relationships between products. For this purpose, the ontology can play an important role in the standardization of product information to build e-catalogs. E-business is large domain and have a lot of attributes and relationships (Qian & Xu, 2010). In addition, the social domain is rich and complex domain; it has many terms and relations that can help in deriving judgment on the social case.

2.9 Summary

In this chapter, we have presented the basic theoretical and technical foundation of this research. We presented the Palestinian Cash Assistance Program and its relation to the social domain in Gaza Strip. We have presented the concepts related to Semantic Web and Ontology. In addition, we described the steps of ontology engineering, SWRL rules and reasoning as well as evaluation of the ontology and tools needed to develop and realize the ontology and the Cash Assistance Support System. Also, we have reviewed various works that use ontologies in social, health,

legal and business domains. We have focused on how to use ontology in the derivation of decisions related to these domains. Nowadays there are many systems in multiple languages and domains for decision making based on Semantic Web, ontology as well as reasoning. In our research, there is a need to build a system for the (MOSD) to solve part of the decision making on social cases to try to achieve social justice. Next chapter presents the development of the domain ontology to be used in cash assistance decision making system to be developed in Chapter 4.

Chapter Three

Cash Assistance Decision Support Ontology

Chapter Three
Cash Assistance Decision Support Ontology

In this chapter, we show the development steps of the Cash Assistance ontology that we will use for getting the decision to the social researchers from the knowledge base. The development steps of the ontology are implemented in the Protégé knowledge system, we called the ontology SocCaseOnt.

The ontology is important for giving social advice to the social researchers and workers in the social domain. The ontology content is collected form the difference papers and documents from the (MOSD) specific to the social work, which focus on the Cash program. The ontology is developed with the assistance of the social domain experts. They help to identify concepts, relationships, and definitions.

Many tools are available for developing ontologies such as Hozo, DOML, and Altova Semantic Works etc. Protégé is one of the most widely used ontology development editors and knowledge systems that define ontology concepts (classes), properties, cardinality, class instances and rules. It also supports several ontology representation languages, including OWL (V. Jain. et al., 2013).

OWL is used for modelling the ontology through expressing the aspects of the social cash assistance domain and its efficient reasoning and scalability. We follow the steps for engineering the ontology as presented in Section 2.5 that includes (Noy et al., 2001):

Step 1: Determining the domain of the ontology

Step 2: Reuse Existing Ontologies

Step 3: Enumerate the Important Terms in the ontology

Step 4: Define Classes and Class Hierarchy of the ontology.

Step 5: Define the Properties of classes (Slots).

Step 6: Define the Facets of the Slots

Step 7: Create instances of the ontology.

Step 8: Create Rules.

Step 9: Apply Ontology Reasoner on the ontology.

Next, we elaborate and apply each step

3.1 Determining the Domain and scope of the Ontology

To achieve this step we have to answer some questions:

1. What is the covered domain from the ontology?

The ontology covers Cash Assistance domain in the Palestinian National Program for the social protection.

2. What is the ontology usage?

The ontology provides a knowledge base for the social domain in the cash assistance sector. It will be used in the system for giving social advices for the researchers who judge social cases. It will work as a common and sharable vocabulary among these experts and other related Palestinian institutions.

3. What types of questions are answered by the information contained in the ontology?

The ontology would provide comprehensive answers to questions related to social cash assistance domain such as:

- What are the base information of a given case?
- What are the acceptance or rejection of the Social Cases?
- What are the reasons for the acceptance or rejection?
- What are the amount to be paid based on the ontology?
- What are the cases mentioned in the current case?
- What are the related program (another services) to the given cases?

4. Who will use the ontology?

The ontology will be available as a system to be used by the social researcher in the MOSD for giving them the social advice related with the social cases.

5. Why to develop such ontology?

We can share the common understanding of the structure of Social Cases information among the social workers or related users in the same domain.
Knowledge Base in such domain can be retrieved through the querying and reasoning. Many Cases will answered by the same answer (Accept or Reject), which is the big goal of the ontology (judging) in which will strength the usage of the ontology.

3.2 Reuse Existing Ontologies

With the enormous application of semantic web, ontologies are becoming more widely available. There is no single standard way to develop ontology. It is not necessary to start from scratch always. There are some existing cash assistance systems as TANF-2009 (Temporary Assistance for Needy Families) which deals with the children involved with Child Welfare. We have to take advantage of some of the classes and relations and use them to determine the method of building the ontology, as well as to identify some relationships and properties needed to give judging on the social cases (i.e. social advice)

3.3 Enumerate the Important Terms in the ontology

We define some common cases and rules that are needed in the process of judging the Social Cases in our system. The ontology is represented in OWL. It can be reused by other applications interested in the same domain. Figure 3.1 illustrates the core classes of the *ontology* as well as the relationships among them.

The ontology has 81 classes, 21 object properties, and 32 data properties.

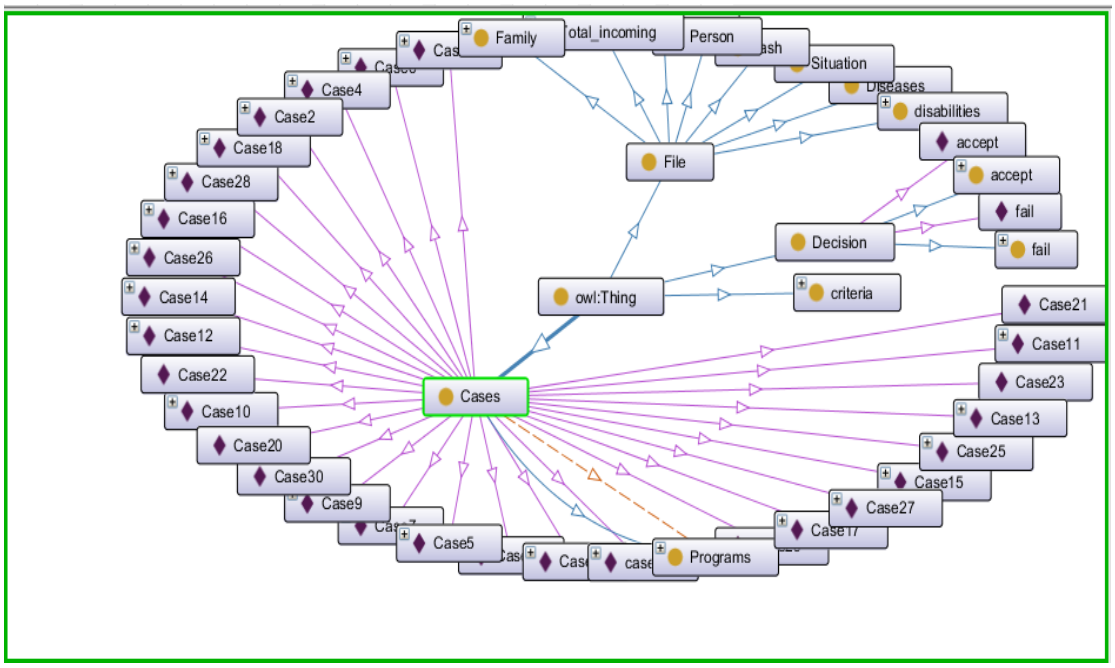


Figure (.1): Main classes in the Ontology

We add terms and properties for these terms by focusing and understanding the social protection criteria in MOSD and using some common cases. This is done by analyzing

the structure of cases and applying the criteria on them. To deduce the terms, the following questions guide us:

1. What are the main terms that we need to represent?

The main terms we talk about are Cases, Files and Decisions. Table 3.1 illustrates those terms and their importance within the ontology.

Table (3.1): Main Terms in the Cash Assistance Ontology

No	Full Name in English	Name in Arabic	Importance
1	Cases	الحالات	This class contains previous cases.
2	Criteria	معايير استهداف الحالة	This class is used to classify the criteria of the social protection for judging the social cases.
3	File	ملف الحالة	This class refers to the content of the social case types.
4	Decision	القرار	This class refers to the decision of the cases. (Accept or Reject).

3.4 Define Classes and Class Hierarchy of the ontology

We defined the classes from the terms; the terms in Table 3.2 are sub classes in the ontology and will become anchors in the class hierarchy. Classes also are organized into a hierarchical taxonomy.

Table (3.2): Ontology Sub-classes

No	Full Name in English	Name in Arabic	Description
1	Cases	الحالات	This class contains previous cases
2	Criteria	معايير استهداف الحالة	This class is used to classify the criteria of the social protection for judging the social cases.
3	C1	المعيار الأول	The first criteria of the social protection based on the adapted system in MOSD.

No	Full Name in English	Name in Arabic	Description
4	C2	المعيار الثاني	The second criteria of the social protection based on the adapted system in MOSD.
5	C3	المعيار الثالث	The third criteria of the social protection based on the adapted system in MOSD.
6	C4	المعيار الرابع	The fourth criteria of the social protection based on the adapted system in MOSD.
7	C5	المعيار الخامس	The fifth criteria of the social protection based on the adapted system in MOSD.
8	Decision	القرار	This class refers to the decision of the cases.
9	Accept	الموافقة على الحالة	The class refers to the cases acceptance.
10	Reject	رفض الحالة	The class refers to the cases rejection.
11	Disabilities	الاعاقات	There are 7 Types of standard disabilities
12	Diseases	الأمراض	The standard diseases types.
13	Special_Case	الحالة الخاصة	The class contains 13 types of Special Cases in the MOSD with its restrictions.
14	Family	العائلة المستهدفة	The family considered for the cash assistance program.
15	Files	الملفات المطلوبة	The files needed to complete the social case.
16	Person	الشخص داخل الأسرة	Person in the family.
17	Children	الأبناء داخل الاسرة	Children in the family.
18	Parents	الاباء داخل الأسرة	The parents in the family.
19	Single	الولد أعزب	The single person in the family.
20	Programs	البرامج الخدمية الأخرى	The related programs with Cash Assistance program
21	Total_income	مجموع مصادر الدخل	The sum of the incoming resources of the family.

No	Full Name in English	Name in Arabic	Description
23	Constant_income	الدخل الثابت	The types of the constant income for the family.
24	Lands	أراضي	The owned lands of the family.
25	Jobs	وظائف	The jobs of the family.
26	Real Estate	عقارات	The own properties of the family.
27	Facility_owner	مالك منشأة	The owned establishments of the family.
28	Rejected_case	الحالة غير مقبولة	The cases of the family is not accepted.
29	Trade	تجارة	The trade of the family.
30	Wealth	ثروة	The wealth of the family.

A **top-down** development process starts with the most general concepts "super class" in the domain and sub-class specialization of the concepts, and bottom-up development process starts with the definition of the most specific classes, the leaves of the hierarchy, with subsequent grouping of these classes into more general concepts. We used a top-down approach to identify the structure and we tried to cover the most common cases in Social Protection Domain (NOY, 2001).

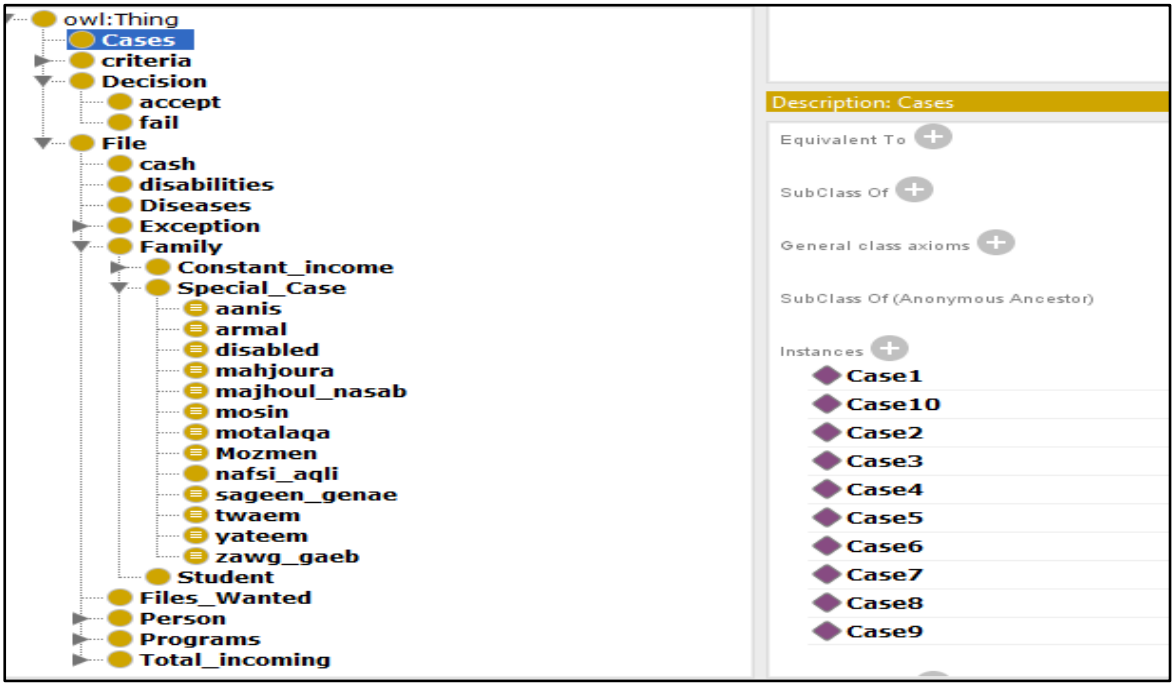


Figure (3.2): Top Level Class Hierarchy

3.5 Define the Properties of Classes (Slots)

Properties define the relationships between objects. There are two types of the properties. Object properties are used to relate objects with objects, and data properties, which are used to relate objects to literal data (integer, float, etc…).

Object Property: Table 3.3 presents object properties in the ontology; (has_CI) property relates the two classes (Family and Total_income) and (has_CI) is inverse property of it. (has_disability) property relates class (Disabled) and the classes (Disabilities). The classes (Family – Disabled) called domain of the property and the classes (Total_income – Disabilities) called range of the property. (has_Prog) property relates class (Family) and the classes (Programs).

Table (3.3): Ontology Object Properties

No	Object Property	Name in Arabic	Domain	Range	Characteristics
1	has_CI	لديه دخل ثابت	Family	Constant income	Complement of has_no_CI
2	has_disability	عنده اعاقات	Disabled	Disabilities	-
3	has_disease	عنده أمراض	Mozmen	Diseases	-
4	has_file	الحالة الخاصة لديها ملف	Nafsi, Mahjoura, Motalaqa, Zawg gaeb	Files	-
5	has_husband	لديها زوج	Sageen ,mahjoura, Zawg gaeb	Husband	complement of has_no_husband and symmetric
6	has_father	ليس لديه أب	Yateem, Majhoul nasab	Father	Complement of has_no_father and symmetric

No	Object Property	Name in Arabic	Domain	Range	Characteristics
7	has_no_parents	ليس له والدين	Yateem, Majhoul_nasab	Parents	-
8	has_Prog	لديه برنامج	Family	Programs	-
9	has_same_grade	لديه نفس المستوى	Not_accepted	Constant_income	Symmetric
10	Typef_of	من نوع	Special_Case	Mozmen, Person ,Maiden,Armal, Disabled ,mahjoura, Motalaqa Majhoul_nasab, Elderly , Nafsi_aqli, Sageen_genae,Twins, Yateem, Zawg_gaeb	Complement of 'has_type'

Data Property: Table 3.4 presents data properties in the ontology; (has_count_F) property relates the class (Family) as domain with (integer) as range. (has_count_Sp) property relates class (Family) as domain with (integer) as range. (Message) property represents the applied criteria used in the social case. (Message1) property represents the judging on the social case. (Message2) property represents the classification (programs) related with the social case.

Table (3.4): Ontology Data Properties

No	Data Property	Name in Arabic	Domain	Range
1	has_city	المدينة	Family	String
2	has_adddress	العنوان	Family	String

3	has_age	العمر	Student or Maiden or Elderly or Twins	Integer
4	has_cash	المبلغ	Family or cash	Integer
5	has_count_F	عدد أفراد العائلة	Family	Integer
6	has_count_Sp	عدد الحالات الخاصة	Special Case	Integer
7	has_count_St	عدد الطلاب	Student	Integer
8	has_count_T	عدد التوائم	Twins	Integer
9	has_duration	ملف الحالة	sageen_genae or Zawg_gaeb	Float
10	has_gender	الجنس	Zawg gaeb , motalaq, Armal ,Maiden ,Mahjoura	String
11	has_kader	الكادر	Family	Integer
12	has_text	نص المعايير	criteria	String
13	has_tincome	مجموع مصادر الدخل	Total_incoming	integer
14	has_total_cash	المبلغ الاجمالي	Family	Integer
15	has_value	لديها قيمة	Total_income or cash	Integer
16	ID	رقم الأسرة	Family	Integer
17	Phone	تلفون الاسرة	Family	integer
18	Message	عرض المعيار المطبق		Literal
19	Message1	عرض الحكم على الحالة		Literal
20	Message2	عرض التصنيف		Literal

3.6 Define the Facets of the Slots

We used some restriction on the classes to strength and enrich the ontology such as: has some values from and some relational characteristics such as: functional property and inverse property. Next, we give several examples for restrictions on Maiden*, Elderly and Twins* classes.

Example 1:

The class *Maiden* has some restrictions on the person as: has no husband and the condition: age >=40 to be accepted as Special Case. This is shown in the figure 3.3.

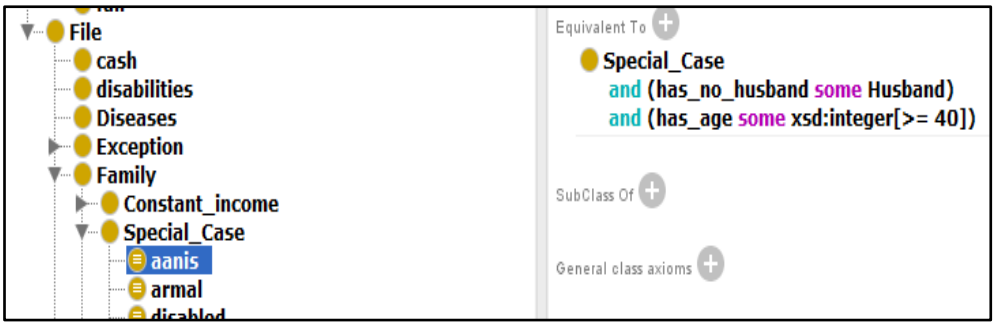


Figure (3.3): *Maiden* Class Restrictions

Example 2:

The class *Elderly* has restriction on the person such as has no husband and the condition: age >=60 to be accepted as Special Case. This is shown in the figure 3.4.

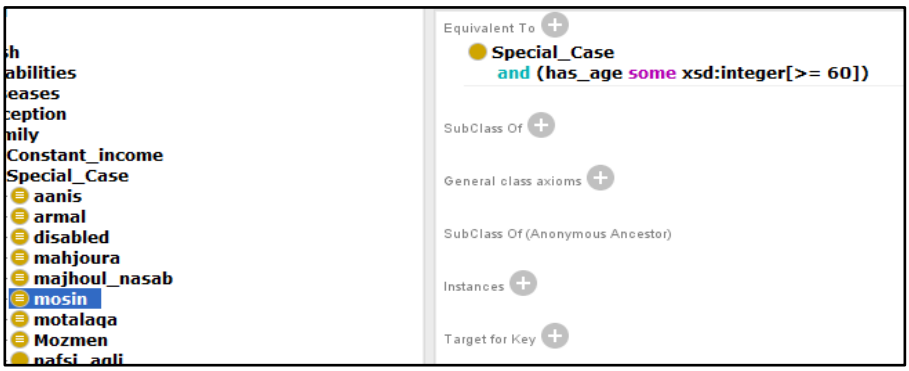


Figure (3.4): *Elderly* Class Restrictions

Example 3:

The class *Twins* has restriction on the family as: age <=3 years and count >=3 to be accepted as Special Case. This is shown in the figure 3.5.

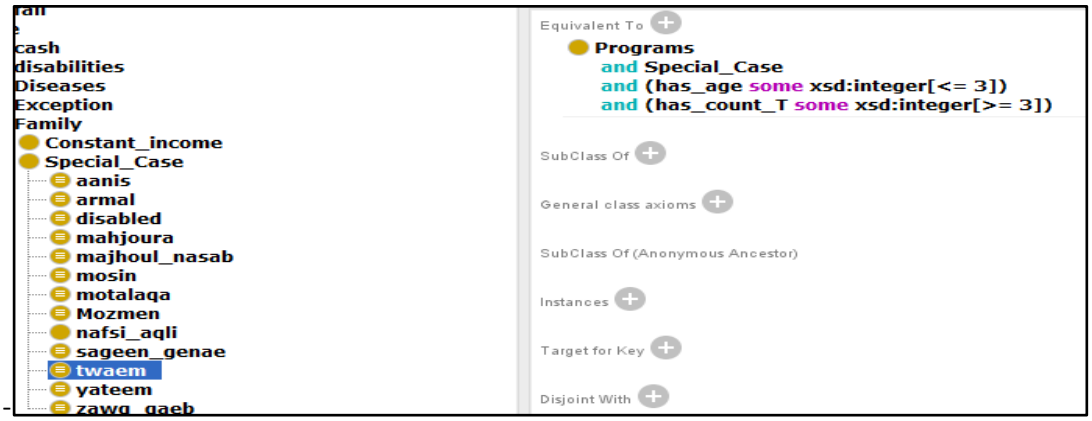
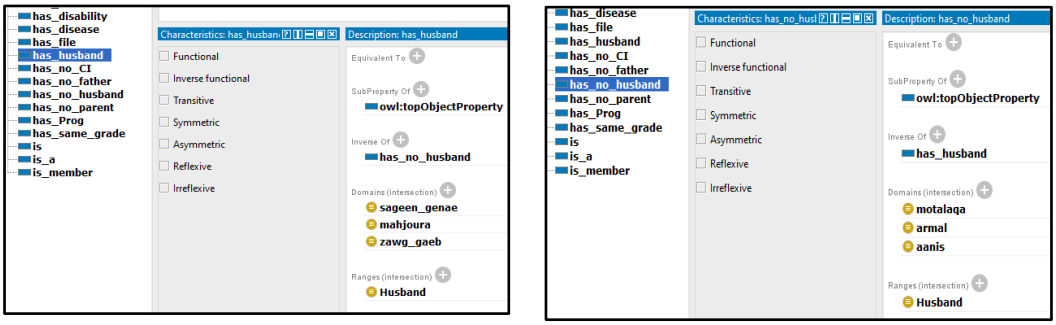


Figure (3.5): *Twins* Class Restrictions

Example 4:

The complement property is important for enriching the ontology. Figure4.6



has_husband Property *has_no_husband* Property

Figure (3.6): Complement Properties Class Restriction

3.7 Create Instances of the ontology

A knowledge Base for Cash Assistance is created by adding instances to the classes. The number of instances in ontology is larger than the number of classes. The creation of individuals allows for all the properties of the classes to be recorded through adding multiple types of previous Social Cases which enrich the ontology and are important for deriving the social advice for new cases.

We defined 168 instances in the ontology. An example of instances is *Cases,* which contains 30 case instances. One of these case instances is shown in Figure 3.7.

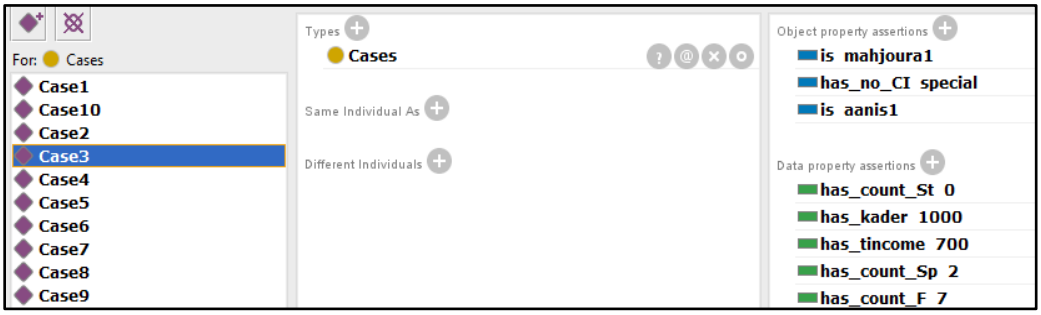


Figure (3.7): Instance Example of Social Case

Case (3) refers to the family data as follows:

1- The family has no students.
2- Kader = 1000 NIS
3- Total Income = 700 NIS
4- The count of special Cases = 2
5- The count of the family = 7
6- There is Special Case (Mahjoura = 1 , Maiden = 1)

7- The family has no constant income

3.8 Create Rules

The **Semantic Web Rule Language** (**SWRL**) is a proposed language for the Semantic Web that can be used to express rules as well as logic (see section 2.6)

We define and express the following rules shown in table 3.5.

Table (3.5): SWRL Rules used in the ontology

Rule No.	Explanation
1	**c1(?c), has_text(?c, ?g), Family(?f), has_count_Sp(?f, ?sc), xsd:integer[< 1](?sc), Special_Case(?f) -> fail(?f), message(?f, ?g)** Rule 1 refers to the rejection of the family if it does not contain Special Case (SP). The rule addresses case (1).
2	**c2(?c), has_text(?c, ?g), Family(?f), has_CI(?f, ?t) -> fail(?f), message(?f, ?g)** Rule 2 refers to the rejection of the family that has constant Income as (aqarat - Monshaa_owner -Tegara –Tharwa- aradi).The rule addresses case (2).
Rule No.	**Explanation**
3	**c3(?c), has_text(?c, ?g), Family(?f), has_count_F(?f, ?childNo), xsd:integer[>= 6](?childNo), has_tincome(?f, ?Total_incoming), xsd:integer[<= 750](?Total_incoming), has_count_Sp(?f, ?sc), xsd:integer[>= 1](?sc), has_count_St(?f, ?st), xsd:integer[>= 0](?st), has_kader(?f, ?b), swrlb:lessThan(?Total_incoming, ?b), has_no_CI(?f, ?gg) -> accept(?f), message(?f, ?g)** Rule 3 refers to the acceptance of the family if has count greater than 6 persons and the family has total income of less than 750 NIS and has special case and may have student and the **kader** is less than the total income and there is no constant income for the family. The rule addresses case (3). **Kader** = the calculated amount of the money by the social researcher.

4	**c4(?c), has_text(?c, ?g), Family(?f), has_tincome(?f, ?Total_incoming),has_kader(?f,?b), swrlb:greaterThan(?Total_incoming, ?b) -> fail(?f), message(?f, ?g)** Rule 4 refers to the rejection of the family if it has total income greater than kader, the result is presented by message. The rule addresses case (4).
5	**c5(?c), has_text(?c, ?g), Family(?f), has_count_Sp(?f, ?sc), xsd:integer[>= 1](?sc), has_tincome(?f, ?Total_incoming), has_kader(?f, ?b), swrlb:lessThan(?Total_incoming, ?b), has_no_CI(?f, ?gg) -> accept(?f), message(?f, ?g)** Rule 5 refers to the acceptance of the family if it has special case hreater than 1and the **kader** is less than the total income and there is no constant income for the family. The rule addresses case (5). **Kader** = the calculated amount of the money by the social researcher. .
Rule No.	**Explanation**
6	**c6(?c), big_family(?f), has_text(?c, ?g), accept(?b), has_dec(?b, ?gg) -> message(?f, ?g), message1(?f, ?gg)** Rule 6 refers to the acceptance of the family if it is big family. This happens when the economic situation for the family is very bad and the family.
7	**Emergency_Cash_Assistance(?p), has_text(?p, ?g), Special_Case(?f), disabled(?d), type_of(?f, ?d) -> message2(?f, ?g)** Rule 7 refers to the related program to the special case disabled, which is Emergency Cash Assistance

8	**Elderly_Care_Prog(?p), has_text(?p, ?g), Special_Case(?f), elderly(?m), type_of(?f, ?m) -> message2(?f, ?g)** Rule 17 refers to the related program to the special case of type elderly, which is Elderly Care program.
9	**Tech_and_Voc_Train_Prog_for_Women(?p), has_text(?p, ?g), Special_Case(?f), maiden(?aa), type_of(?f, ?aa) -> message2(?f, ?g)** Rule 9 refers to the related program to the special case of type maiden, which is Technical and Vocation Training program.
10	**Emergency_Cash_Assistance(?p), has_text(?p, ?g), Special_Case(?f), twins(?t), type_of(?f, ?t) -> message2(?f, ?g)** Rule 10 refers to the related program to the special case of type twins, which is Emergency Cash Assistance program.
11	**Twins_Care_Prog(?p), has_text(?p, ?g), Special_Case(?f), twins(?t), type_of(?f, ?t) -> message2(?f, ?g)** Rule 11 refers to the related program to the special case of type twins, which is Twins Care Program.
Rule No.	**Explanation**
12	**Small_Business_Support_Prog(?p), has_text(?p, ?g), Special_Case(?f), big_family(?b), type_of(?f, ?b) -> message2(?f, ?g)** Rule 12 refers to the related program to the special case of type big family, which is Small Business Support Program.
13	**Garmin_prog(?p), has_text(?p, ?g), Special_Case(?f), sageen_genae(?s), type_of(?f, ?s) -> message2(?f, ?g)** Rule 13 refers to the related program to the special case of type sageen genae, which is Garmin program.

14	**Emergency_Cash_Assistance(?p), has_text(?p, ?g), Special_Case(?f), sageen_genae(?s), type_of(?f, ?s) -> message2(?f, ?g)** Rule 14 refers to the related program to the special case of type sageen genae, which is Emergency Cash Assistance program.
15	**Sponsorship_Orphans_Prog(?p), has_text(?p, ?g), Special_Case(?f), yateem(?y), type_of(?f, ?y) -> message2(?f, ?g)** Rule 15 refers to the related program to the special case of type yateem, which is Sponsorship Orphans program.
16	**Reh_of_Disabilities_prog(?p), has_text(?p, ?g), Special_Case(?f), disabled(?d), type_of(?f, ?d) -> message2(?f, ?g)** Rule 16 refers to the related program to the special case of type disabled, which is rehabilitation of Disabilities program.

3.9 Apply Ontology Reasoner

An ontology reasoner can be applied after creating the ontology instances e.i, the knowledge base. The reasoner we used is Pellet. We downloaded the reasoner and added it to the plugin folder of Protégé to run. The reasoner identifies new inferences that helps us to give social advice to the social researchers for judging the cash assistance cases and its related cases. Figure 3.8 shows the new inference knowledge for example: accept, cases related programs to the case after applying. We present two examples to show the effect of reasoning on the Cash Assistance knowledge

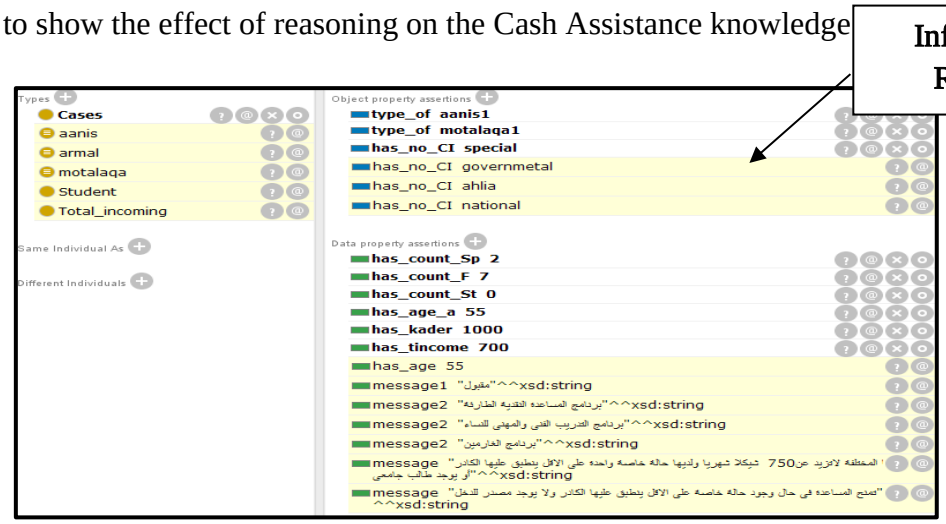


Figure (3.8): Reasoner Results (Inference) Case

Example (1):

The reasoner could infer new knowledge on case 1 which is the judging on the case 1 (**reject**) and all classes related to the case as (SP – Student – Total Incoming). Also, inferred the related individual for the object property (has_no_CI) as (national and ahlia). In addition, the reasoner inferred the social protection criteria that the case applied as shown in Figure 3.8.

Example (2):

The reasoner could infer new knowledge on case 3 as mentioned before in the creating instance. The judging on the case is (**Accept**). Figure 3.9 shows the criteria, which is applied and the related instance for the objet property (has_no_CI) which is (ahlia - national).

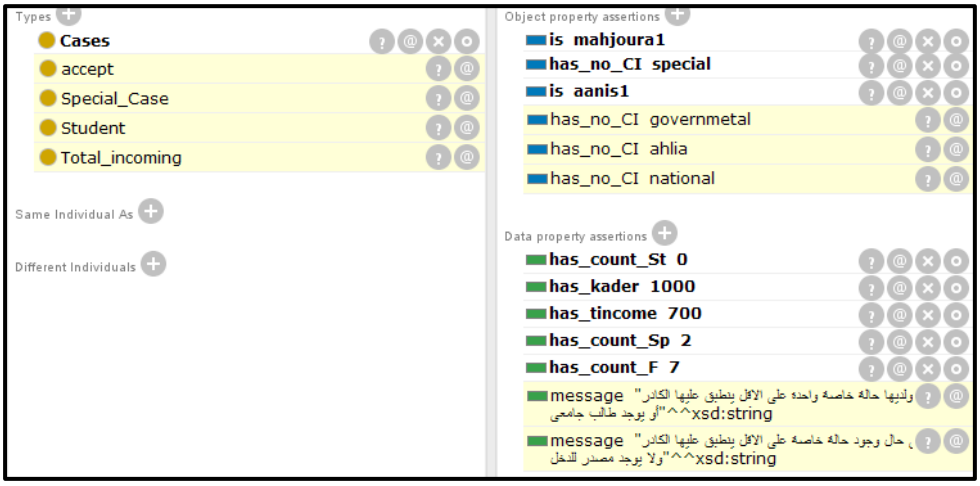


Figure (3.9): Reasoner Results (Inference)

3.10 Summary

In this chapter, we have presented and discussed the steps of building the *SocCaseOnt* ontology. We identified the domain and scope of the ontology together with defining the main terms and properties. Then we defined the facets and slots of the classes. We added a number of instances to the ontology to establish a knowledge base. Finally, we explored the used rules in the ontology and we used a reasoner to check the consistency of the ontology and identifying new relations based on existing ones. We have used Protégé OWL to implement and realize the ontology. In the next

chapter, we use the ontology as the essential component for decision making and reference in the proposed Cash Assistance Decision System.

Chapter Four

Cash Assistance Decision Support System

Chapter Four
Cash Assistance Decision Support System

In this chapter, we present the Soc*CaseOnt* system prototype to realize the proposed approach for judging on social cases in PNPSP. The development of the system prototype consists of two phases:

- Analysis of the system and its relation with the proposal ontology.
- Designing and implementing the system, which forms the execution of the proposed judging system. It completes the idea of the judging of the social cases as (SWRL Rules, the reasoner as well as the querying interface). In the following three sections, we describe each phase:

4.1 System Analysis

In this section, we show a complete description of the behavior of Soc*CaseOnt* system including a set of use cases that describe the interactions as well as the functionality with imposed constraints on the design and implementation of the system.

4.1.1 Overall Description

We develop a system prototype for the ontology-based judging approach on Social Cases which automatically derives the judging based on the rules (the criteria) of the social protection in the MOSD.

The system mainly consists of four components:

1-User Interface: allows users to interact with the system and view the saved social cases or enter new social cases data to get a new judgment.

2-*SocCaseOnt* Ontology and Knowledge Base: identifies social cases and the criteria, (As in chapter 4).

3-Social Cases Rules: SWRL rules designed based on the criteria in the Palestinian National Program of Social Protection.

4-Social Cases judging Engine: Derives a judgment through new knowledge known as inference based on SWRL rules using a reasoner. It includes different instances of the social cases.

4.1.2 System Functions

We present the system functions and requirements by drawing use case diagrams, which contain primarily actors and use cases. Actors are entities that interact with the *SocCaseOnt* system, while use cases are the system functions related to judging the social cases. The *SocCaseOnt* system needs the following use cases:

Use case	Actor	Figure No.
Display the social case by browsing	Social Researcher	4.1
Add new social case	Administrator	4.2

- **User Characteristics**

There are two types of users, Social Researcher and Administrator. Social Researcher must be familiar with the system and Administrator should be familiar with the criteria of the social protection.

- **Principal Actors**

The two principal actors in *SocCaseOnt* system are the users: Social Researcher and administrator.

4.1.3 Specific Requirements

The following are the specific requirements of *SocCaseOnt* system that covers the various functions:

- The system must enable Social Researcher to display social cases by browsing case category, as shown in use case 1 Figure 4.1.
- The system must enable Social Researcher to get the social cases judgment as shown in the use case 2 Figure 4.2.
- The system must enable Administrator to add new social cases as shown in the use case 3 Figure 4.3.

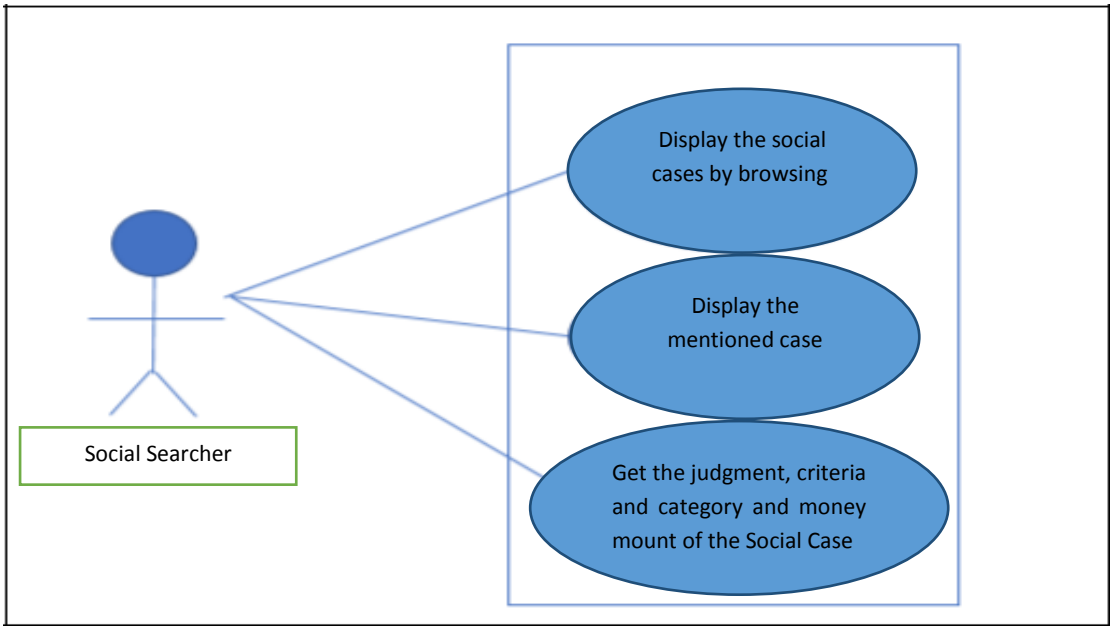


Figure (4.1): Social Researcher Reviews Use Case

Use case 1: Displays Social cases by browsing, as shown in Figure 4.1.

Primary Actor	**Social Researcher**
Main scenario	1- User selects one case from the case list.
	2- System displays the social case practiced criteria mentioned to the case and the judgment and the money mount and the category (Program category)

Use case 2: Add new Social Cases, as shown in Figure 4.2.

Primary Actor	**Administrator**
Main Scenario	1- Enter the family name 2- Enter the number of the persons of the family. 3- Enter the number of the university students in the family 4- Select yes/No to the income source. 5- Insert the mount of the income source 6- Enter Next.

	7- Select from the check box one or more from 13 type of the special cases then insert the completed data from each of those check box. 8- Select No of the special case.

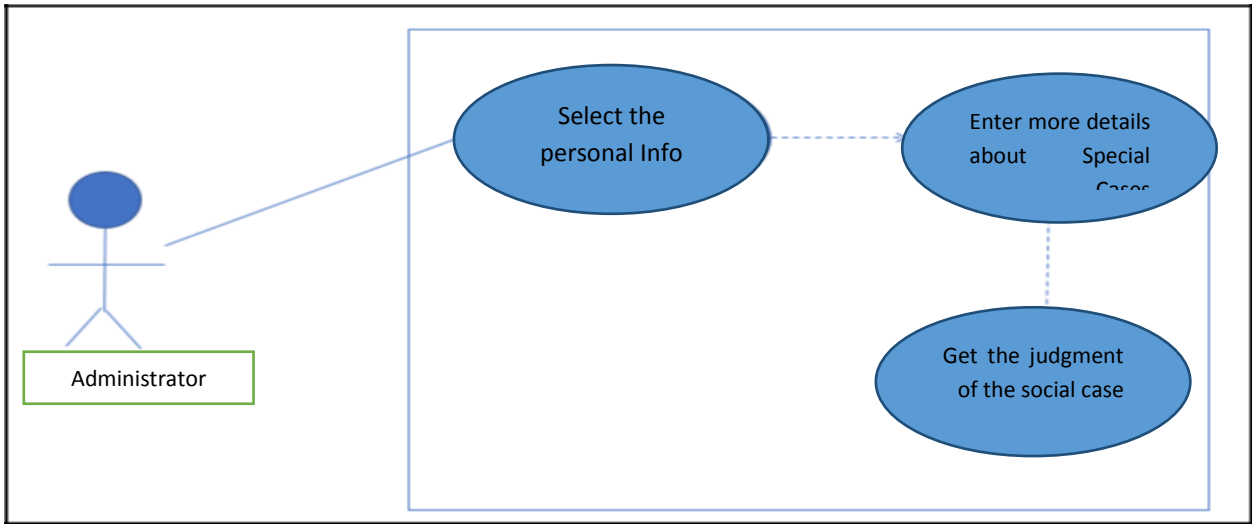


Figure (4.2): Administrator Use Case

4.2 Social Assistance System Design

The design of the system satisfies the requirements identified in the previous phase. The requirements, identified in the system analysis phase, are transformed into system design that accurately describes the approach and is used in the system implementation in the next phase. We start the design by the system architecture, which essentially reflects the proposed approach. We provide a detailed description of the Social Cases system and its abstract main components. The system architecture is shown in Figure 4.5 and presents the components, dependencies and interactions among them. It consists of four components (Social Case Knowledge Base, Semantic Component, Social Cases Rules and Social Case Deciding).

Next, we explain the design of each component:

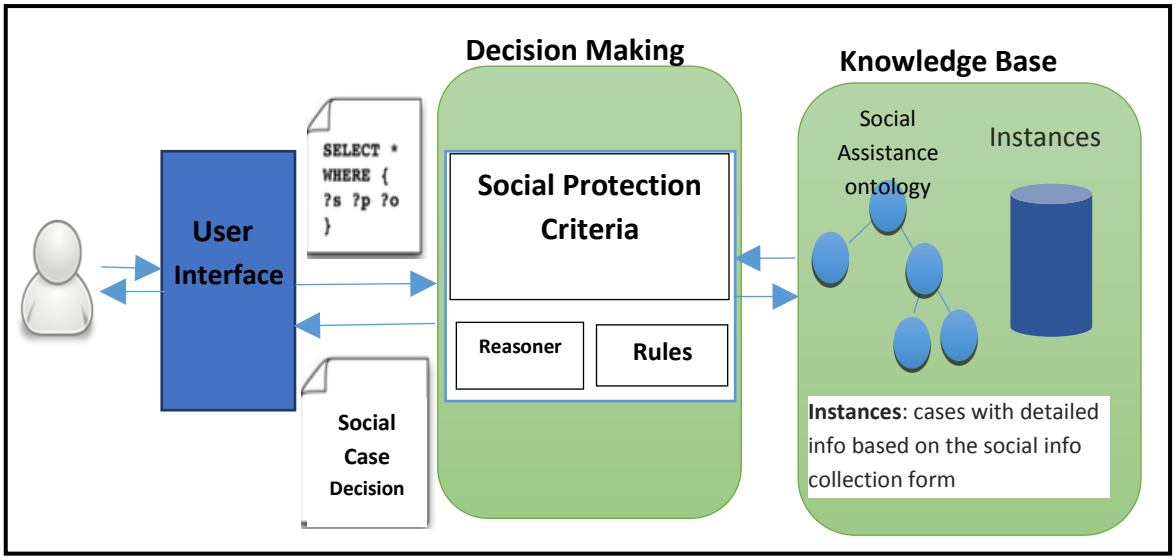


Figure (4.3): Social Assistance System Architecture

Social Assistance Knowledge Base: this component contains the common cases data in the social domain related to the financial assistance presenting the instances, which are collected from Social Researcher, the cases consist of 30 case, which are divided to two groups, 15 common case and 15 uncommon case.

Semantic Component: it contains the used rules in the system based on the social protection criteria; in addition the reasoner to infer new info about the acceptance or rejection for the case.

Social Case Decision: This component consists of new knowledge derived from the ontology, in addition to the programs related to the case and the reason of this decision.

User Interface: We have two interfaces. The first is the interface (social researchers) to help them in browsing Social Cases based and the second interface consists of the results of deciding on the social cases.

The system works as follows:

1- The social researcher retrieves the decision for the case by using the rules. Reasoner to give fair decision, which will be reasoned according the social criteria, infers it.

2- The social researcher retrieves the programs (classification), related to the tested case.

3- The social researcher retrieves the expected cash assistant amount according to kader criteria (mentioned in Section 3.7).

4.2.1 User Interface

Based on the Social Assistance ontology we present the social cases judgment system interface used by Social Researchers for judging the social cases and presenting the reasons for decisions and the dependent criteria.

Figure (4.4): Social Cases Interface by Browsing

The first interface shown in Figure 4.4 enables the social researchers to view the elements of the Social Assistance System, which contains:

1- The social Cases list from the ontology (الحالات الاجتماعية).
2- Number of the persons in the family (عدد أفراد الأسرة).
3- Number of the students in the family (عدد الطلبة الجامعيين).
4- Number of the special cases in the family (عدد الحالات الخاصة في الأسرة).
5- Constant Income (مجموع مصادر الدخل لدى الأسرة).
6- Details of the special cases (الحالات الخاصة), which contains of 13 types
   a- Maiden (نوع الحالة الخاصة عانس)
   b- Armal (الأرملة)
   c- Disabled (الشخص ذو الإعاقة)
   d- Mahjoura (المرأة المهجورة)
   e- Majhoul Nasab (مجهول النسب)
   f- Elderly (مسن)
   g- Divorced (Motalaqa مطلقة)

h- Mozmen (مريض مزمن)

i- Nafsi and Aqli patient (مريض عقلي أو نفسي)

j- Sageen gnae (زوجة السجين الجنائي)

k- Twins (التوائم)

l- Zawg gaeb (زوجة غائب)

m- University student (طالب جامعي)

Figure (4.5): Social Cases Judging System Results

The results of the system consists of the following parts:

1- Acceptance or rejection for Social Cases (الحكم على الحالة)

2- The reasons for the acceptance or rejection (المعيار المطبق)

3- The category of the needed programs for supporting the case (تصنيف البرنامج)

4- The expected amount must be paid based on social protection criteria (المبلغ المتوقع).

The second interface is the Data Entry interface for the elements of the social cases shown in Figure 4.6, which contains the parts:

Figure (4.6): Social Cases Interface by Entry

4.2.2 SWRL Rules

We designed this component based on needed SWRL rules for social cases judgment based on the criteria of the Social Assistance ontology in Chapter 4. The system depends heavily on SWRL rules to derive the reasoning needed for extracting and inferring knowledge from the knowledge base. They are used to relate the social cases entered by the user.

SWRL rule 5 is applied to Case 3 and the execution result of the rule is shown in Figure 4.7.

Rule 5:

```
c5(?c), has_text(?c, ?g), Family(?f), has_count_Sp(?f, ?sc), xsd:integer[>= 1](?sc), has_tincome(?f, ?Total_incoming), has_kader(?f, ?b), swrlb:lessThan(?Total_incoming, ?b), has_no_CI(?f, ?gg), accept(?a), has_dec(?a, ?ggg) -> message(?f, ?g), message1(?f, ?ggg)
```

the rule states that case is accepted when the family has Special Case and the total income is less than the kader of this family and there is no constant income, the rule will show two messages, message for presenting the criteria that applied and message for presenting the decision on the cases>

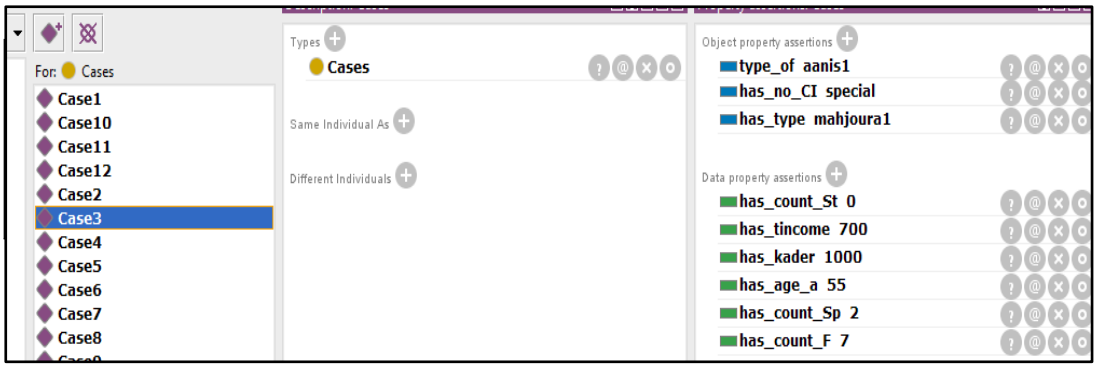


Figure (4.7): The Effect of Executing Rule 5 on Case 3

4.2.3 The Reasoner

We used a reasoner called Pellet to infer logical consequences from a set of asserted facts or axioms related to social cases and cases in the knowledge base. After Social Researcher completes entering the data in steps 1 to 2 (shown in Figure 4.8, the reasoner uses the ontology concepts and their relationships and the SWRL rules on the knowledge base to produce the social judgment for the specific case. After running the reasoner over the ontology and SWRL rules, all inferred knowledge is stored as values for inferred data properties inferred knowledge include case decision being accepted or not. For example, *has_kader* refer to data property used to calculate the kader (money amount) must be paid to the social case to be accepted or will be rejected; it depends on the family number of the family members.

4.2.4 Social Assistance Ontology

SocCaseOnt ontology contains the criteria of the Palestinian Program for social protection. Instances are used to identify the cases, similar cases and information that is needed in the process of derivation of judgment. The ontology presented in Chapter 4.

4.2.5 The Social Assistance Knowledge Base (KB)

The knowledge base is built using OWL ontology. It consists of the ontology and enriched with 30 individuals related to the social cases. More details on creating the knowledge base together with social cases individuals are found in Section 3.7.

The user, admin or social researcher interacts with engine to view social cases knowledge and to get the judgment.

4.3 Cash Assistance System Implementation

The implementation of the system is based on the previous design of the system components. The system is formed into code and functional interfaces as follows.

4.3.1 User Interface

Figure 4.8 shows the user interface, which contains three parts (list of cases (e.g., case3), Personal data (e.g., family count 2) and Special Cases (e.g., disabled).

First part is a list of cases. The second part displays family count (عدد أفراد الأسرة), student count (عدد الطلاب), special cases count (عدد الحالات الخاصة) and the total income (مجموع مصادر الدخل) related to the case. The third part displays details of the special cases.

Figure (4.8): Social Cases System Interface

The User Interface also contains the result of the judgment of the social cases as shown in Figure 4.9. It displays that the Researcher interface which contains four parts. The judgment result (مقبول/مرفوض), the program category (تصنيف الحالة) – the practiced criteria of the case (المعيار المطبق) and the total amount to be paid for the case (المبلغ المتوقع).

Figure (4.9): The Social Cases System Judgment Results

Figure 4.10 illustrates part of the code of these interfaces in Java; it presents URL of the ontology and defining static object of type File to get the ontology file which called 4.5.owl, in addition to define pm of type PrefixManager and selectedCase of type string

```
public final class Home extends javax.swing.JFrame {

private static final String BASE_URL =
"http://www.semanticweb.org/xx/ontologies/2016/3/untitled-ontology-49";
private static OWLObjectRenderer renderer = new DLSyntaxObjectRenderer();
static File file = new File("5.2.owl");

    OWLOntologyManager manager;
    OWLOntology ontology;
    OWLReasonerFactory reasonerFactory;
    OWLReasoner reasoner;
    OWLDataFactory factory;
    PrefixManager pm;
    Map<String, String> prefixMap;
    private static QueryEngine engine;
    String selectedCase;
    private DefaultListModel model;
```

Figure (4.10): Connection to the Ontology

Figure 4.11 define the data property *has_count_st* with the individual *family*. The reasoner returns the values of the selected properties for the individual. *has_count_Sp* data property for the count of the special cases in the family.

```
// ***************************** has_count_St  ********************
   OWLDataProperty has_count_St = factory.getOWLDataProperty(":has_count_St", pm);
for (OWLLiteral indv : reasoner.getDataPropertyValues(IndCase, has_count_St)) {
    jLStudentNum.setText(indv.getLiteral());
}
// ***************************** has_count_Sp  ********************
OWLDataProperty has_count_Sp = factory.getOWLDataProperty(":has_count_Sp", pm);
for (OWLLiteral indv : reasoner.getDataPropertyValues(IndCase, has_count_Sp)) {
    jLSpNum.setText(indv.getLiteral());
}
  // ***************************** has_age_s  ********************
OWLDataProperty has_age_s = factory.getOWLDataProperty(":has_age_s", pm);
   for (Iterator<OWLLiteral> it = reasoner.getDataPropertyValues(IndCase, has_age_s)
           .iterator(); it.hasNext();) {
       OWLLiteral indv = it.next();
       jTFUStudent.setText(indv.getLiteral());
       jChBUnivStudent.setSelected(true);
   }
     // ***************************** has_age_a  ********************
OWLDataProperty has_age_a = factory.getOWLDataProperty(":has_age_a", pm);
   for (Iterator<OWLLiteral> it = reasoner.getDataPropertyValues(IndCase, has_age_a)
           .iterator(); it.hasNext();) {
       OWLLiteral indv = it.next();
       jTFAnesAge.setText(indv.getLiteral());
       jChBAnes.setSelected(true);
      }
```

Figure (4.11): Data Property Code

4.3.2 SWRL Rules

We implemented the needed SWRL rules presented in Section 3.8 as shown on Figure 4.12. We use the getOWLDataPropertyValues () stored in data property *message 1* to show decision of the case Accept or Reject.

Datatype property is one two main categories of properties. It links individuals to data values, and is defined as an instance of the built-in OWL class owl:DatatypeProperty.

```
public void getResult(String theCase) {

OWLNamedIndividual IndCase = factory.getOWLNamedIndividual(":" + theCase
                                                           + "", pm);
 // ***************************** result  ********************
OWLDataProperty result = factory.getOWLDataProperty(":message1", pm);
  for (OWLLiteral indv : reasoner.getDataPropertyValues(IndCase, result)){
 jLResult.setText(indv.getLiteral());
    }

 // ***************************** Criterion  ********************
  OWLDataProperty Criterion = factory.getOWLDataProperty(":message", pm);
  for (OWLLiteral indv : reasoner.getDataPropertyValues(IndCase, Criterion)){
 MCriterion.addElement(indv.getLiteral());
    }

// ***************************** Program  ********************
 OWLDataProperty Program = factory.getOWLDataProperty(":message2", pm);
  for (OWLLiteral indv : reasoner.getDataPropertyValues(IndCase, Program)) {
 MProgram.addElement(indv.getLiteral());
    }
    }
```

Figure (4.12): Data Property Definition

Figure 4.13 shows a Java code to calculate the total amount of the money must paid for the family.

```
//**********************Calculation**********************//
int cnt ,cnt2, Tcnt;

    cnt = Integer.parseInt(jLFamelyNum.getText());
    if   (cnt == 1) {f_count.setText("300");}
      if   (cnt == 2) {f_count.setText("350");}
        if   (cnt == 3) {f_count.setText("400");}
          if   (cnt == 4) {f_count.setText("450");}
            if   (cnt == 5) {f_count.setText("500");}
              if   (cnt == 6) {f_count.setText("550");}
                if   (cnt == 7) {f_count.setText("600");}
                  if   (cnt == 8) {f_count.setText("650");}
                    if   (cnt == 9) {f_count.setText("700");}
                      if   (cnt == 10) {f_count.setText("750");}
                        if   (cnt == 11) {f_count.setText("800");}
                          if   (cnt == 12) {f_count.setText("850");}
                            if   (cnt == 13) {f_count.setText("900");}
                              if   (cnt == 14) {f_count.setText("950");}
                                if   (cnt == 15) {f_count.setText("1000");}
                                  if   (cnt >= 16) {f_count.setText("1050");}

     cnt2 = Integer.parseInt(jLStudentNum.getText());
    if   (cnt2 == 0) {s_count.setText("0");}
    if   (cnt2 == 1) {s_count.setText("200");}
    if   (cnt2 >= 2) {s_count.setText("400");}

    Tcnt = (Integer.parseInt(f_count.getText())) +
            Integer.parseInt(s_count.getText()) +
           (Integer.parseInt(jLSpNum.getText())*200) ;
      Tcount.setText(Tcnt + "");
```

Figure (4.13): Money Amount Calculation

4.3.3 Reasoner

By running the reasoner, new and hidden facts are inferred based on SWRL rules and relations among classes through object properties and/or data property as presented in Section 3.9.

Figure 4.14 presents inferred facts by the reasoner for the social Case 3. Messages are data properties presenting inferred results such as:

Message data property shows the applied criteria from the 6th social protection criteria.

Message1 data property shows judgment of the social case.

Message2 shows the category of the program which must be applied on social case.

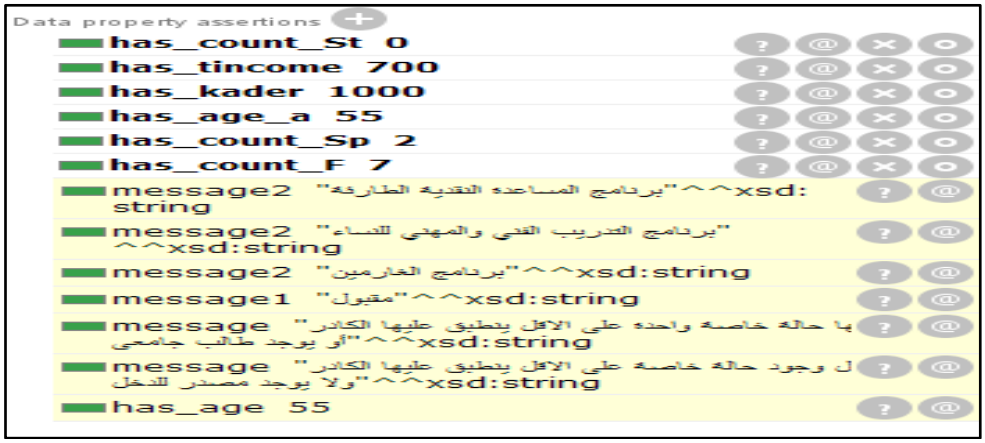


Figure (4.14): Reasoner Inferred Hidden Results

Also, there are inferred results for object properties such as *has_no_CI* as shown in Figure 4.15. It means that the case has no constant income, there are similar inference individual such as (governmental, ahlia and national) which means they are in the same grade of special individual. (By using functional object property *has_same_grade*).

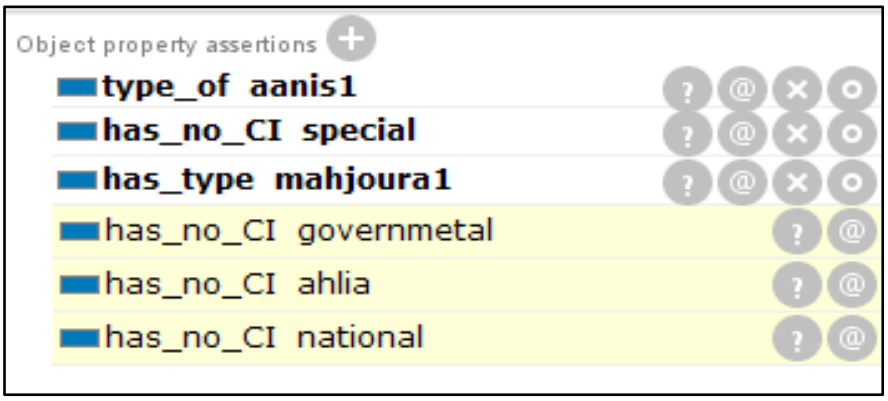


Figure (4.15): Object Property Inference

4.3.4 SocCaseOnt Ontology and Knowledge Base

We present the developments and implementation of *SocCaseOnt* in Chapter 4. The knowledge base is built using OWL ontology. It consists of the instances (individuals) related to social cases; the role of the ontology is to derive the judgment of the social cases. To create an instance needed to choose a class, we create an instance of that class and fill in the slot values. So, we define about 168 instances that are representing all ontology concepts.

4.4 Summary

In this chapter, we have presented the phases of building the prototype for Cash Assistance Decision Support System. In the analysis phase, we have analyzed and specified the requirements of the system and divided the system into four components: user interface, SocCaseOnt Ontology and Knowledge Base, SWRL rules and Social Cases judging Engine. We also described the functions of the system through a number of use cases.

In the Cash Assistance Decision Support System design phase, we have determined and designed the interaction and dependencies between these four components as a system architecture.

In the implementation phase, we have explained how the processes of the system operates to deduce the deciding on the social cases. Additionally, we have explained some implementation issues related to the components according to the design phase. The implementation of the user interface realized the items that have identified in the design phase using Java language in relation to SWRL rules, the reasoner, the ontology and the knowledge base.

Chapter Five

Experimental Results and Evaluation

Chapter Five
Experimental Results and Evaluation

In this chapter, we present the experiments performed for the evaluation of the proposed *SocCaseOnt* system. These experiments are used to evaluate the ability of the system in judging social cases that are offered by the system or those entered by Social Researcher.

The system is evaluated using a set of social cases provided by a social domain expert. We compare the results obtained by the system with those of the domain expert. In the following sections, we present details of the experimental settings including the Knowledge Base (KB) and SWRL rules used. Subsequently, the experimental results are presented and discussed.

5.1 Experimental Settings

For evaluating of the system, we have asked an expert in the social domain to select a sample set of cases with social criteria. The sample set contains some different social criteria for the cases for testing the system. Social data used for the evaluation are obtained from the (MOSD) in Gaza strip. The total of the cases dataset is 30 cases. The first 15 cases are Common Social Cases. The second 15 cases contain uncommon cases to test the effect of the criteria on the instances of the Social Cases in the *ontology*. The overall evaluation of the system is based on the two sets of cases.

Table 5.1 contains the Social Programs Classification, Special Social cases types and the number of the cases for every program within the 30 cases.

Table (5.1): Number of cases related to the Program

Programs Classification	Special Case Type	Number of Cases
Emergency Cash Assistance Program	Elderly	2
	Garmine	2
	Mozmen	3
	Twins	3
	Zawg sageen	3
	Disabled	2
Job Creation Program	Sageen genea	2
	Motalaqa	3
	Zawg sageen	3
Technical and Vocation Training Program	Motalaqa	1
	Zawg sageen	2
	Sageen genea	2
	Maiden	1
	Armal	1
Total: 30 cases		

Testing the Social Cases requires applied criteria such as type of the special case and its details. We asked the domain expert for the results of judging the cases to compare them with the results of the system. Figure 5.1 shows an example of case data to be judged by the Social Researcher (see section 2.7).

Figure (5.1): An Example of Case Data to be judged by Social Researcher.

The 30 Social cases have been selected such that they represent the SWRL rules used in judging the social cases defined by the ontology.

We defined 81 classes, 168 instances and 16 SWRL rules in the ontology. Table 5.2 represents the size of the ontology, the number of classes, the number of, instances, the number of object properties, the number of data properties and the number of SWRL rules in the ontology.

Table (5.2): The Size of the SocCaseOnt

Ontology Components	Component Number
Number of Classes	81
Number of Object Properties	21
Number of Data Properties	32
Number of Instances	168
Number of SWRL Rules	16

We have defined 168 instances (individuals) representing all ontology concepts. Table 5.3 shows the number of instances. The ontology together with these instances form the knowledge base, i.e., the core in component of the system.

Table (5.3): Number of Individuals per Ontology Class

Name of Classes	Number of individuals
Criteria	6
Decision	2
Cases	30
File	113
Programs	17

5.2 Evaluation of the Results

In this section, we present the evaluation to determine the behavior of the *SocCaseOnt* system if it works as expected. Namely, we want to assess the ability of the system in testing the judging on the social cases by presenting the results.

The evaluation of the system depends on the criteria presented as SWRL rules (see Section 3.8) which are used to find the correct / incorrect results of the 30 social cases, which judged by the system shown in Table 5.4.

Table (5.4): Number of Cases and Results

Total Cases	Correct Results	Incorrect Results
30	26	4

Results show that the system can correctly judge 26 out of the 30 cases (the percentage is 86.7%). Table 5.5 shows the 15 common cases that judged using the *SocCaseOnt*o system.

Table (5.5): The Results of Judging the Common Cases

Case ID	Expert Results	SocCaseOnt System Results	Final Results
C1	Accept	Accept	correct
C2	Reject	Reject	correct
C3	Accept	Accept	correct
C4	Accept	Accept	correct
C5	Accept	Accept	correct
C6	Reject	Reject	correct
C7	Accept	Accept	correct
C8	Reject	Reject	correct
C9	Accept	Accept	correct
C10	Reject	Reject	correct
C11	Accept	Accept	correct
C12	Accept	Accept	correct
C13	Accept	Accept	correct
C14	Accept	Accept	correct
C15	Reject	Reject	correct

Table 5.6 presents the other 15 uncommon cases, which means the Social Researcher does not reach a common judgment on them. There are 4 cases with decisions that are different from those of the domain expert. Next, we discuss the reasons of such decision disagreements.

Table (5.6): The Results of Judging the Uncommon Cases

Case ID	Expert Results	SocCaseOnt System Results	Final Results
C16	Accept	Accept	Correct
C17	Reject	Reject	Correct
C18	Reject	Accept	Incorrect

Case ID	Expert Results	SocCaseOnt System Results	Final Results
C19	Accept	Accept	Correct
C20	Accept	Accept	Correct
C21	Accept	Reject	Incorrect
C22	Accept	Accept	Correct
C23	Reject	Reject	Correct
C24	Accept	Accept	Correct
C25	Reject	Reject	Correct
C26	Reject	Accept	Incorrect
C27	Accept	Accept	Correct
C28	Reject	Accept	Incorrect
C29	Accept	Accept	Correct
C30	Reject	Reject	Correct

We present one of the common cases that is correctly judged. Then we present one uncommon case and explain why the system don't give the correct judgment.

Case 11 (Common Case)

Case 11 refers to family with the following data:

1- The case has ID 11.
2- The family has no constant income.
3- The family has situation status (الحالة صعبة).
4- The family has report from the researcher named Mohammad
5- The family members has count 7.

Step 1: The system generates a new instance of *Cases* class with Case ID and asserts case data shown in the data property.

Step 2: The system runs the Pellet inference engine over the ontology and SWRL rules. The SWRL Rule 6 is used. It refers to the acceptance of the family

if it is big family. This happens when the economic situation of the family is very bad.

Rule 6:

c6(?c), big_family(?f), has_text(?c, ?g), accept(?b),
has_dec(?b, ?gg) -> message(?f, ?g), message1(?f, ?gg)

Step 3: The result of the reasoning for the case is as follow:

1- The judging on the case is Message1: "مقبول".
2- The Social Support Program the case which is applied is Message: " برنامج دعم المشاريع الصغيرة".
3- The reason of accepting the case is Message2: " في حال وجدت أسرة بحاجة الى مساعدة وتعيش في ظروف صعبة وخاصة الأسرة كبيرة الحجم يتم اعتمادها بعد اعداد تقرير اجتماعي من واقع زيارة ميدانية ومن ثم تعرض على لجنة للدراسة".

According the system, the results includes the judgment on the case, the suggested Social Support Programs related to the case and the reason of this judgment. According to the domain expert (Researcher), the judgment result for this case is accept "مقبول" and the related Social Support Program is the same as the system has deduced as shown in Figure 5.2.

Figure (5.2): Judgment Results for Case11

Next, we present and explain the 4 uncommon cases which have no consent upon among the social experts and therefore it is natural that the system would return such decisions for them.

Case 18 (Uncommon Case)

Case 18 refers to family with the following data:

The case has ID 18.

The family has no student.

Kader = 1000 NIS

Total Income = 700 NIS

The count of special Cases = 2

The count of the family = 7

The Special Case (Motalaqa = 1, Maiden = 1)

The family has no constant income

Step 1: The system generates a new instance of *Cases* class with Case ID and asserts case data shown in the data property.

Step 2: The *SocCaseOnt* system run the Pellet inference engine over the ontology and SWRL rules. SWRL rule 5 is used. It refers to the acceptance of the family if it has special case hreater than 1and the kader is less than the total income and there is no constant income for the family. The rule addresses case (5).

Rule 5:

c5(?c), has_text(?c, ?g), Family(?f), has_count_Sp(?f, ?sc),xsd:integer[>= 1](?sc), has_tincome(?f,?Total_incoming),has_kader(?f, ?b), swrlb:lessThan(?Total_incoming, ?b),has_no_CI(?f, ?gg), accept(?a), has_dec(?a, ?ggg) -> message(?f, ?g), message1(?f, ?ggg)

Step 3: The result of the reasoning for the case is as follows:

1- The judgment on the case is Message1: "مقبول".

2- The Social Support Program related to the case is Message (برنامج المساعدة المالية), (برنامج الغارمين), (برنامج التدريب الفني والمهني للنساء).

3- The reason of accepting the case is Message2: " تمنح المساعدة في حال ان تكون الاسر ة كبيرة العدد مكونة من 6 أفراد فما فوق وليس لها مصدر دخل ثابت ومجموع مصادرها المختلفة لا تزيد عن 750 شيكلا شهريا ولديها حالة خاصة واحدة على الاقل ينطبق عليها الكادر أو يوجد طالب جامعي".

4- There is another reason of accepting the case is Message2 with title " تمنح المساعدة في حال وجود حالة خاصة على الاقل ينطبق عليها الكادر ولا يوجد مصدر للدخل".

According the system, the results include the judgment of the case, the suggested Social Programs related to the case and the reason of the returned judgment. According to the domain expert (researcher), the judging result for this case is accept "مرفوض" and the related Social Support Programs is the same as the system has deduced.

Figure (5.3): Judgment Results for Case 18

Based on the system, the judgment on the case is "مقبول" as shown in Figure 5.3. The reason for judgment is because there is constant income and the case has a special case.

According to the domain expert, the result for this case is "مرفوض". The reason for this judgment is because the Researcher considers the constant income for the special case (Motalaqa) is for all family (Extended Family contains all the family that eat and drink with others) not for the special case itself (Neutral Family contains father mother and children's) and so the income applies for the whole family. The related Social Support Programs for the case is

Emergency Cash Assistance Program, *Gharmine* Program and *Technical, Vocational Training* Program).

Case 21 (Uncommon Case)

Case 21 refers to family with the following data:

The case has ID 21.

The family has no student.

Kader = 1000 NIS

Total Income = 700 NIS

The count of special Cases = 2

The count of the family = 5

The Special Case (Mozmen = 1, Maiden = 1)

The family has a car.

The wife has temporary job for 3 months and has land.

Step 1: The system generates a new instance of *Cases* class with Case ID and asserts case data shown in the data property.

Step 2: The *SocCaseOnt* system run the Pellet inference engine over the ontology and SWRL rules. SWRL rule 4 is used. It refers to the acceptance of the family if it has special case greater than 1and the kader is less than the total income and there is no constant income for the family. The rule addresses case (21).

Rule 4:

Family(?f), c4(?c), fail(?a), has_dec(?a, ?gg), has_kader(?f, ?b), has_text(?c, ?g), has_tincome(?f, ?Total_incoming), swrlb:greaterThan(?Total_incoming, ?b) -> message(?f, ?g), message1(?f, ?gg)

Step 3: The result of the reasoning for the case is as follows:

1- The judgment on the case is Message1: "مرفوض".

2- The Social Support Program related to the case is Message (برنامج برنامج التدريب الفني والمهني), (برنامج الغارمين), (المساعدة النقدية الطارئة للنساء).

3- The reason of rejecting the case is Message2: " ترفض المساعدة في حال أن تكون مدخولات الأسرة أكبر من الكادر".

According the system, the results include the judgment of the case, the suggested Social Programs related to the case and the reason of the returned judgment. According to the domain expert (researcher), the judging result for this case is accept "مقبول" and the related Social Support Programs is the same as the system has deduced.

Figure (5.4): Judgment Results for Case 21

Based on the system, the judgment on the case is "مرفوض" as shown in Figure 5.4. The reason for judgment is that the kader is greater than the total income.

According to the domain expert, the result for this case is "مقبول". The reason for this judgment is that the researcher considers the special cases (maiden and mozmen) are agreed cases regardless the total income whether it is less or greater than kader. The related social support programs for the case is *Emergency Cash Assistance* Program, *Gharmine* program and *Technical, Vocational Training* Program).

Case 26 (Uncommon Case)

Case 26 refers to family with the following data:

The case has ID 26.

The family has university student =2.

Kader = 1500 NIS

The count of special Cases = (nafsi_aqli= 1)

The count of the family = 7

The family has constant income (Total Income = 850 NIS)

Step 1: The system generates a new instance of *Cases* class with Case ID and asserts case data shown in the data property.

Step 2: The *SocCaseOnt* system run the Pellet inference engine over the ontology and SWRL rules. SWRL rule 4 is used. It refers to the acceptance of the family if it has special case greater than 1and the kader is less than the total income and there is no constant income for the family. The rule addresses case (26).

Rule 4:

Family(?f), c4(?c), fail(?a), has_dec(?a, ?gg), has_kader(?f, ?b), has_text(?c, ?g), has_tincome(?f, ?Total_incoming), swrlb:greaterThan(?Total_incoming, ?b) -> message(?f, ?g), message1(?f, ?gg)

Step 3: The result of the reasoning for the case is as follows:

1- The judgment on the case is Message1: "مرفوض".
2- There is no related social support program to the case.
3- The reason of rejecting the case is Message2: " ترفض المساعدة في حال أن تكون مدخولات الأسرة أكبر من الكادر".

According to the domain expert (researcher), the judging result for this case is accept "مقبول" and there is no related social programs to the case.

Figure (5.5): Judgment Results for Case 26

Based on the system, the judgment on the case is "مرفوض" as shown in Figure 5.5. The reason for judgment is that the kader is greater than the total income.

According to the domain expert, the result for this case is "مقبول". The reason for this judgment is that the researcher considers the special cases (nafsi_aqli) is agreed case regardless the total income whether it is less or greater than kader. There are no related social support programs for the case.

Case 28 (Uncommon Case)

Case 28 refers to family with the following data:

The case has ID 28.

The family has no student.

Kader = 1000 NIS

Total Income = 550 NIS

The count of special Cases (disabled = 2)

The count of the family = 8

The family has land 250m.

Step 1: The system generates a new instance of *Cases* class with Case ID and asserts case data shown in the data property.

Step 2: The *SocCaseOnt* system run the Pellet inference engine over the ontology and SWRL rules. SWRL rule 4 is used. It refers to the acceptance of the family if it has special case greater than 1and the kader is less than the total income and there is no constant income for the family. The rule addresses case (28).

Rule 4:

Family(?f), c4(?c), fail(?a), has_dec(?a, ?gg), has_kader(?f, ?b), has_text(?c, ?g), has_tincome(?f, ?Total_incoming), swrlb:greaterThan(?Total_incoming, ?b) -> message(?f, ?g), message1(?f, ?gg)

Step 3: The result of the reasoning for the case is as follows:

1- The judgment on the case is Message1: "مرفوض".
2- There is no related social support program to the case.
3- The reason of rejecting the case is Message2: " ترفض المساعدة في حال أن تكون مدخولات الأسرة أكبر من الكادر".

According to the domain expert (researcher), the judging result for this case is accept "مقبول" and there is no related social programs to the case.

Figure (5.6): Judgment Results for Case 26

Based on the system, the judgment on the case is "مرفوض" as shown in Figure 5.6. The reason for judgment is that the kader is greater than the total income.

According to the domain expert, the result for this case is "مقبول". The reason for this judgment is that the researcher considers the land is not constant income because it is used for trading; it is used for the family itself. There are no related social support programs for the case.

5.3 Summary

In this chapter, the experimental results and the domain expert evaluation were performed and compared on the system using 30 test cases. The expert divided the test cases into two subsets: common cases and uncommon cases. The subset of common cases included agreed upon cases as whether they deserve social assistance or not among the social researchers. Uncommon cases subset included disagreed upon cases among the social researchers.

The results obtained from the system showed that the system has judged 26 cases out of 30 cases with same judgments as the expert with similarity ratio of 86.7%. The remaining 4 un-agreeable cases are uncommon cases that are not agreed up even among the experts in the MOSD. Nevertheless, the system is suggested to be used to support decision making in social assistance rather than to be used exclusively for making decisions.

Chapter Six
Conclusion and Future Work

Chapter Six
Conclusion and Future Work

The social domain is one of the important and sensitive domains. The Palestinian National Program of Social Protection (PNPSP) is the biggest social program in Palestine for helping the poor families; it serves approximately 400,000 persons.

This research has proposed to improve decision making related to the right of a citizen whether to receive social assistance from PNPSP or not. The improvement has focused on using semantic web approaches and tools in this respect. The main contribution towards this improvement is building a social domain ontology as well as a rule-based knowledge system that constitute the ontology and a number of social cases as a representation of the target social domain in Palestine. Using the knowledge base with the ontology within the system helps social researchers draw correct and justified decisions on social cases as whether a case deserves social financial assistance or not.

Towards solving the research problem and achieving its objectives, we were able to collect the required data related to the domain (the social cases) which deals with the living, learning and economic condition of the family. Based on this data, we were able to build a specialized domain ontology (SocCaseOnto) and populate it with the social cases as instances leading to the formation of the knowledge base.

An important part of the knowledge base system built here was the semantic rules. We have transformed the criteria used by the social researchers to make decisions on social cases to a set of suitable reasoning rules in SWRL format and connected them to the knowledge base (respectively the ontology) through the reasoner. To this end we have developed a system prototype with a user interface as a proof of concept to our approach. The system has been divided into four functions which are social cases judging, programs or classifications related to the family, results explanation from the criteria and determining the amount of cash the family deserves. Judging on the cases is performed with the help of the reasoner as well as the SWRL rules (as indicated above) and based on the input provided by the user directly through the user interface or based on the data retrieved from the saved cases from the knowledge base. Through

the user interface, the social researcher may enter the parameters related to a case applying for social assistant and then receives a decision with explanation about the decision. The decision is about accepting or rejecting the social case and in case of acceptance, it returns the appropriate social program(s) the case may receive.

We were able to evaluate the ontology as well as the prototype through a number of experiments base on a set of social cases collected form the PNSPS program. The results of decision making on social cases have been compared to those of a human expert. 30 pre-tested cases with family data have been used in the evaluation process. The results generated from the system showed that the system has correctly judged 26 out of the 30 cases indicating a success ratio of 86.7%. It can not be said that the 4 other cases (accounting to 13.3% of the cases) were wrongly judged but rather they are called uncommon cases meaning that they are not agreed upon among the social experts themselves. Therefore the system has also, based on the criteria given to it, taken its own decision.

Therefore, this contributes to the main goal of the system which is to achieve more correct and fair decisions related to the cases asking for Social Cash Assistance. This is exceptionally important for Gaza Strip on a time where poverty levels have increased and means of living have been decreased and at the same time the amount of cash for social cases is always limited.

Based on working toward achieving the research goals as indicated above, there are several improvements and future directions that can be tackled. They include the following:

- There is a need to use data mining techniques to improve classifying the reports and findings of social researchers made on social cases during emergencies. This will certainly affects the decisions made on social cases and ensures they are fair and correct.
- Concentrating on the overall budget available for PNPSP such that it will be a system parameter therefore affecting the overall decisions given by the system. This will lead to the available budget to be distributed evenly among all the social cases deserving assistance especially when this budget is usually limited and being decreased.

- The system can be extended to cover other types of programs related to social cases within PNPSP that are different from the cash assistance program. These include the emergency assistance program and food assistance program.
- The number of social cases involved in the experiments can be further increased to give more comprehensive and more accurate evaluation of the system. This is true when it comes to the portion of the uncommon cases.
- Finally, since the proposed system is a prototype, it is necessary to develop a complete and working web and mobile application versions and put into action for the citizens to enter their information and interact with the system and know in advance whether they may deserve social assistance or not.

References

Abusalah, M., Tait, J., & Oakes, M. (2009, 21-25 Sept. 2009). *Arabic-English Automatic Ontology Mapping Based on Machine Readable Dictionary.* Paper presented at the Computer Science (ENC), 2009 Mexican International Conference on.

AbuTaha, A. W. (2015). *An Ontology-Based Arabic Question Answering System(Unpublished Theseis).* (Master), Islamic University of Gaza.

Afifi, M. A., & Baraka, R. S. (2017). *Semantically Enriched Derivation of Legal Advice in the Palestinian Labour Domain (Unpublished Theseis).* (Master), Islamic University of Gaza, Palestine.

Agha, H. M., & Baraka, R. S. (2017). *Diagnosing Heart Diseases Using Ontology and SWRL Rules (Unpublished Theseis).* (Master), Islamic University of Gaza, Palestine.

Aguado, J., Bernaras, A., Smithers, T., Pedrinaci, C., & Cendoya, M. (2003). *Using ontology research in semantic web applications: a case study.* Paper presented at the Proceedings of the 10th Conference of the Spanish Association for Artificial Intelligence, CAEPIA.

Baraka, R., & Dalloul, Y. (2014). Building Hadith ontology to support the authenticity of Isnad. *International Journal on Islamic Applications in Computer Science And Technology, 2*(1), 25-39.

Biletskiy, Y., & Vorochek, O. (2004). *Building and integrating ontologies to assist decision making.* Paper presented at the Decision and Control, 2004. CDC. 3rd IEEE Conference on.

Brickley, D., & Guha, R. V. (2004). RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation, February 2004. *See http://www. w3. org/TR/rdf-schema*.

Castells, P., Foncillas, B., Lara, R., Rico, M., & Alonso, J. L. (2004). *Semantic web technologies for economic and financial information management.* Paper presented at the European Semantic Web Symposium.

Daigneault, P.-M. (2014). Three paradigms of social assistance. *Sage Open, 4*(4).

Di Maio, P. (2007). An open ontology for open source emergency response system. *Open Source Research Community*.

El-Askary, M. A. (2015). *An Ontology-Based Approach for Diagnosing Date Palm Diseases (Unpublished Theseis).* (Master), Islamic University of Gaza.

Fan, Z., & Zlatanova, S. (2011). Exploring ontologies for semantic interoperability of data in emergency response. *Applied Geomatics, 3*(2), 109-122.

Fensel, D., Van Harmelen, F., Horrocks, I., McGuinness, D. L., & Patel-Schneider, P. F. (2001). OIL: An ontology infrastructure for the semantic web. *IEEE Intelligent Systems, 16*(2), 38-45.

Grigoris Antoniou, P. G., Frank van Harmelen, Rinke Hoekstra. (2012). *A Semantic Web Primer* (3 ed.) MIT press, London, England .

Izumi, S., Yasuda, N., Itabashi, G., Kato, Y., Takahashi, K., Suganuma, T., & Shiratori, N. (2007). *A Health Advice Derivation System based on an Ontology.* Paper presented at the 6th International and 2nd Asian Semantic Web Conference (ISWC2007+ ASWC2007).

Jain, V., & Singh, M. (2013). Ontology Development and Query Retrieval using Protutf-8. *International Journal of Intelligent Systems and Applications, 5*(9), 67.

Jarrar, M. (2011). Ontology-based data and process governance framework: the case of e-government interoperability in Palestine.

Jihan, S. H., & Segev, A. (2014). Humanitarian Assistance Ontology for Emergency Disaster Response. *Intelligent Systems, IEEE, 29*(3), 6-13.

Kalfoglou, Y. (2007). Using ontologies to support and critique decisions. *Engineering Intelligent Systems, 15*(3), 33-40.

Kato, T., Maneerat, N., Varakulsiripunth, R., Kato, Y., & Takahashi, K. (2009). *Ontology-based E-health system with Thai herb recommendation.* Paper presented at the Proceeding of the 6th International Joint Conference on Computer Science and Software Engineering (JCSSE2009).

Li, X., Liu, G., Ling, A., Zhan, J., An, N., Li, L., & Sha, Y. (2008). *Building a practical ontology for emergency response systems.* Paper presented at the Computer Science and Software Engineering, 2008 International Conference on.

McCarthy, L., Vandervalk, B., & Wilkinson, M. (2012). SPARQL Assist language-neutral query composer. *BMC bioinformatics, 13*(1), 1.

MOSD. (2013). The Palestinian National Program of Social Protection. from http://www.mosa.gov.ps/showTopic.php?id=49

NOY, N. F. (2001). Ontology Development 101: A Guide to Creating Your First Ontology: Knowldege Systems Laboratory, Stanford University. *Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880*.

Noy, N. F., & McGuinness, D. L. (2001). *Ontology development 101: A guide to creating your first ontology* (Vol. 15): Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880.

Qawaqneh, Z., El-Qawasmeh, E., & Kayed, A. (2007, 28-31 Oct. 2007). *New method for ranking arabic web sites using ontology concepts.* Paper presented at the Digital Information Management, 2007. ICDIM '07. 2nd International Conference on.

Qian, F., & Xu, L. (2010). *Constructing Ontology-Based Product Catalogs System for E-business.* Paper presented at the Intelligent Computation Technology and Automation (ICICTA), 2010 International Conference on.

Robu, I., Robu, V., & Thirion, B. (2006). An introduction to the Semantic Web for health sciences librarians. *Journal of the Medical Library Association*, *94*(2), 198.

Satoru Izumi, N. Y., Goichi Itabashi, Yasushi Kato, Kaoru Takahashi, Takuo Suganuma, Norio Shiratori. (2007). *A Health Advice Derivation System based on an Ontology*. Paper presented at the The 6th International Semantic Web Conference and the 2nd Asian Semantic Web Conference, Busan, Korea. http://iswc2007.semanticweb.org/main/default.asp.html

Sitthisarn, S., Lau, L., & Dew, P. M. (2011, 28-29 June 2011). *Semantic keyword search for expert witness discovery.* Paper presented at the Semantic Technology and Information Retrieval (STAIR), 2011 International Conference on.

Taylor, A. (2013). *Developing an Ontology of Legal Research*. Paper presented at the Confrance for Low School Computing. http://conference.cali.org/2013/sessions/developing-ontology-legal-research

Stanford University, S. (2015). Protégé. Retrieved 10/8/2017, from http://protege.stanford.edu/

W3C. (2012). Web Ontology Language (OWL). Retrieved 5/9/2017, 2017, from https://www.w3.org/2001/sw/wiki/OWL
wikipedia. (2013). Semantic Web Rule Language. from https://en.wikipedia.org/wiki/Semantic_Web_Rule_Language
wikipedia. (2016). Expert system. from https://en.wikipedia.org/wiki/Expert_system
wikipedia, S. (2013). SPARQL. from https://en.wikipedia.org/wiki/SPARQL

APPENDIX A: SWRL RULES

c1(?c), has_text(?c, ?g), Family(?f), has_count_Sp(?f, ?sc), xsd:integer[< 1](?sc), Special_Case(?f) -> fail(?f), message(?f, ?g)
c2(?c), has_text(?c, ?g), Family(?f), has_CI(?f, ?t) -> fail(?f), message(?f, ?g)
c3(?c), has_text(?c, ?g), Family(?f), has_count_F(?f, ?childNo), xsd:integer[>= 6](?childNo), has_tincome(?f, ?Total_incoming), xsd:integer[<= 750](?Total_incoming), has_count_Sp(?f, ?sc), xsd:integer[>= 1](?sc), has_count_St(?f, ?st), xsd:integer[>= 0](?st), has_kader(?f, ?b), swrlb:lessThan(?Total_incoming, ?b), has_no_CI(?f, ?gg) -> accept(?f), message(?f, ?g)
c4(?c), has_text(?c, ?g), Family(?f), has_tincome(?f, ?Total_incoming),has_kader(?f,?b), swrlb:greaterThan(?Total_incoming, ?b) -> fail(?f), message(?f, ?g)
c5(?c), has_text(?c, ?g), Family(?f), has_count_Sp(?f, ?sc), xsd:integer[>= 1](?sc), has_tincome(?f, ?Total_incoming), has_kader(?f, ?b), swrlb:lessThan(?Total_incoming, ?b), has_no_CI(?f, ?gg) -> accept(?f), message(?f, ?g)
c6(?c), big_family(?f), has_text(?c, ?g), accept(?b), has_dec(?b, ?gg) -> message(?f, ?g), message1(?f, ?gg)
Emergency_Cash_Assistance(?p), has_text(?p, ?g), Special_Case(?f), disabled(?d), type_of(?f, ?d) -> message2(?f, ?g)
Elderly_Care_Prog(?p), has_text(?p, ?g), Special_Case(?f), elderly(?m), type_of(?f, ?m) -> message2(?f, ?g)
Tech_and_Voc_Train_Prog_for_Women(?p), has_text(?p, ?g), Special_Case(?f), maiden(?aa), type_of(?f, ?aa) -> message2(?f, ?g)

Emergency_Cash_Assistance(?p), has_text(?p, ?g), Special_Case(?f), twins(?t), type_of(?f, ?t) -> message2(?f, ?g)
Twins_Care_Prog(?p), has_text(?p, ?g), Special_Case(?f), twins(?t), type_of(?f, ?t) -> message2(?f, ?g)
Small_Business_Support_Prog(?p), has_text(?p, ?g), Special_Case(?f), big_family(?b), type_of(?f, ?b) -> message2(?f, ?g)
Garmin_prog(?p), has_text(?p, ?g), Special_Case(?f), sageen_genae(?s), type_of(?f, ?s) -> message2(?f, ?g)
Emergency_Cash_Assistance(?p), has_text(?p, ?g), Special_Case(?f), sageen_genae(?s), type_of(?f, ?s) -> message2(?f, ?g)
Sponsorship_Orphans_Prog(?p), has_text(?p, ?g), Special_Case(?f), yateem(?y), type_of(?f, ?y) -> message2(?f, ?g)
Reh_of_Disabilities_prog(?p), has_text(?p, ?g), Special_Case(?f), disabled(?d), type_of(?f, ?d) -> message2(?f, ?g)

APPENDIX B: PART OF OWL SOURCE CODE

```
<Ontology xmlns="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.semanticweb.org/xx/ontologies/2016/3/untitled-ontology-49"
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:xml="http://www.w3.org/XML/1998/namespace"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     ontologyIRI="http://www.semanticweb.org/xx/ontologies/2016/3/untitled-
ontology-49">
    <Prefix name="" IRI="http://www.semanticweb.org/xx/ontologies/2016/3/untitled-
ontology-49"/>
    <Prefix name="owl" IRI="http://www.w3.org/2002/07/owl#"/>
    <Prefix name="rdf" IRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#"/>
    <Prefix name="xml" IRI="http://www.w3.org/XML/1998/namespace"/>
    <Prefix name="xsd" IRI="http://www.w3.org/2001/XMLSchema#"/>
    <Prefix name="Mosa" IRI="http://www.w3.org/2002/07/owl#"/>
    <Prefix name="rdfs" IRI="http://www.w3.org/2000/01/rdf-schema#"/>
    <Declaration>
        <DataProperty IRI="#has_count_F"/>
    </Declaration>
    <Declaration>
        <Class IRI="#Garmin_prog"/>
    </Declaration>
    <Declaration>
        <NamedIndividual IRI="#so"/>
    </Declaration>
```

<Declaration>
<NamedIndividual IRI="#hi"/>
</Declaration>
<Declaration>
<Class IRI="#aqarat"/>
</Declaration>
<Declaration>
<Class IRI="#Constant_income"/>
</Declaration>
<Declaration>
<Class IRI="#Own_produced"/>
</Declaration>
<Declaration>
<DataProperty IRI="#has_total_cash"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#yateem1"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#student2"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#tav"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#Case18"/>

```
</Declaration>
<Declaration>
    <NamedIndividual IRI="#rabw"/>
</Declaration>
<Declaration>
    <Class IRI="#Osra"/>
</Declaration>
<Declaration>
    <Class IRI="#disabled"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#maiden2"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#majhoul_nasab2"/>
</Declaration>
<Declaration>
    <Class IRI="#sageen_genae"/>
</Declaration>
<Declaration>
    <Class IRI="#Reh_of_Disabilities_prog"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#Family10"/>
</Declaration>
<Declaration>
```

<NamedIndividual IRI="#cash_1050"/>
</Declaration>
<Declaration>
<DataProperty IRI="#hasCI"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#zawg_gaeb1"/>
</Declaration>
<Declaration>
<Class IRI="#zawg_gaeb"/>
</Declaration>
<Declaration>
<Class IRI="#Family"/>
</Declaration>
<Declaration>
<Class IRI="#Files"/>
</Declaration>
<Declaration>
<Class IRI="#Parents"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#nafsi_aqli2"/>
</Declaration>
<Declaration>
<DataProperty IRI="#has_adddress"/>
</Declaration>

<Declaration>
<NamedIndividual IRI="#cash_200"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#yateem2"/>
</Declaration>
<Declaration>
<Class IRI="#elderly"/>
</Declaration>
<Declaration>
<ObjectProperty IRI="#case_type"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#national"/>
</Declaration>
<Declaration>
<Class IRI="#Situation"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#ahlia"/>
</Declaration>
<Declaration>
<Class IRI="#Mother"/>
</Declaration>
<Declaration>
<ObjectProperty IRI="#is_member"/>

```
</Declaration>
<Declaration>
    <DataProperty IRI="#message4"/>
</Declaration>
<Declaration>
    <ObjectProperty IRI="#has_disability"/>
</Declaration>
<Declaration>
    <Class IRI="#Health_insurance_prog"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#Case1"/>
</Declaration>
<Declaration>
    <Class IRI="#disabilities"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#Family11"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#zawg_gaeb2"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#Case2"/>
</Declaration>
<Declaration>
```

<NamedIndividual IRI="#ahmed$ali_zakia"/>
</Declaration>
<Declaration>
<Class IRI="#twins"/>
</Declaration>
<Declaration>
<ObjectProperty IRI="#has_file"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#nafsi_aqli1"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#fail"/>
</Declaration>
<Declaration>
<DataProperty IRI="#has_count_Sp"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#Mozmen1"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#cash_300"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#abeer"/>
</Declaration>

<Declaration>
<NamedIndividual IRI="#motalaqa2"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#saw"/>
</Declaration>
<Declaration>
<Class IRI="#accept"/>
</Declaration>
<Declaration>
<ObjectProperty IRI="#has_situation"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#tc"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#o"/>
</Declaration>
<Declaration>
<Class IRI="#Summer_and_Winter_Camps_Prog"/>
</Declaration>
<Declaration>
<ObjectProperty IRI="#has_disease"/>
</Declaration>
<Declaration>
<NamedIndividual IRI="#special"/>

```
</Declaration>
<Declaration>
   <DataProperty IRI="#has_age"/>
</Declaration>
<Declaration>
   <Class IRI="#Person"/>
</Declaration>
<Declaration>
   <NamedIndividual IRI="#Family12"/>
</Declaration>
<Declaration>
   <NamedIndividual IRI="#uss"/>
</Declaration>
<Declaration>
   <NamedIndividual IRI="#cash_650"/>
</Declaration>
<Declaration>
   <DataProperty IRI="#has_text"/>
</Declaration>
<Declaration>
   <NamedIndividual IRI="#critrtia1"/>
</Declaration>
<Declaration>
   <NamedIndividual IRI="#Mozmen2"/>
</Declaration>
<Declaration>
```

```
    <ObjectProperty IRI="#has_no_CI"/>
</Declaration>
<Declaration>
    <Class IRI="#motalaqa"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#Ahmed"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#cancer"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#cash_800"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#motalaqa1"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#n"/>
</Declaration>
<Declaration>
    <NamedIndividual IRI="#sugar"/>
</Declaration>
<Declaration>
    <Class IRI="#Tech_and_Voc_Train_Prog_for_Women"/>
</Declaration>
```