Received: June 24, 2021. Revised: July 24, 2021.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Development of Ontology for Semantic Structure of Strategic Alignment Framework for IT Projects Combining PMBOK, PMI and VAL IT

Lamia Moudoubah[1]* **Abir El yamami**[1] **Khalifa Mansouri**[1] **Mohammed Qbadou**[1]

[1]*Signals, distributed systems and Artificial Intelligence Laboratory, ENSET Mohammedia,*
University Hassan II, Casablanca, Morocco
* Corresponding author's Email: lamiae.modobah@gmail.com

Abstract: The objective of this paper is to provide an ontological overview of the semantically rich structure of best practice repositories for IT (Information Technology) project alignment. It emphasizes the importance of integrating IT management repositories with IT project management repositories to achieve high performance. It aims to present a semantic conceptual framework for IT project alignment that combines Project Management Body of Knowledge (PMBOK) , Project Management Institute (PMI's) risk management frameworks for improving project performance, and value from IT investment (VAL IT), which is an overarching framework for optimizing the business value of a project. We develop an ontology integrating these frameworks using the Ontology definition metamodel (ODM) approach. We use Unified Modeling Language (UML) for Resource Description Framework Schema (RDFS) representation to model the knowledge related to the principles of strategic alignment of IT projects and their interactions to visualize this field and give a relevant study, presented by an ontological model, that will serve the professionals of the domain as well as the researchers interested in the governance of information systems and IT project management. We continued the process of building our ontology from business objects based on our UML business model in order to populate a structured knowledge base that will be validated and shared in servers. We have also validated the consistency of the proposed ontology with the HermiT 1.4.3. 456 reasoning engine integrated in PROTÉGÉ 5 and we have also validated the structure of the proposed ontology with the OOPS! tool. The resulting ontology of this contribution is therefore validated by solid tools that allowed us to obtain a confirmed ontology. We confirm that the results of this contribution appear in the metrics of the resulting ontology which are summarized in the following data: 202 axioms (101: logical axiom count & 101: lexical axiom count), 64 Class count, 37 object property count, 26 subClassOf, 37 ObjectPropertyDomain and 38 ObjectPropertyRank. These results are inferred from the reasoning applied to the proposed ontology using the HermiT reasoner.

Keywords: PMBOK, PMI, VAL IT, OMD, UML, Ontology, RDFS.

1. Introduction

Matching a company's project portfolio to ensure full involvement in the desired outcomes requires a great deal of attention to detail. In particular, managers must ensure that projects do not deviate from established guidelines. Companies use a multitude of IT management methods and maturation models to achieve their goals. However, it is our understanding that project success rates in the technology sector are not high enough. Therefore, IT metrics need to be balanced, integrated, and implemented as a method of assessing IT project alignment.

Project management repositories have become a necessity to cope with the constraints of the environment and to achieve the strategic objectives defined by the organization [1]. indicates that the state of the art and the review of experimental data characterize and describe project management in the same way, the time factor, the budget factor and the technical characteristics are placed as the main constraints imposed on the actors and organizations operating in this field.

According to [2], project management is the business process of creating a unique product, service or outcome. A project is a completed effort, with a specific start and end time, to create a measurable deliverable. Projects are incrementally developed in stages and predictable increments that are linked to milestones and achievement dates. The main challenge of project management is to achieve all of the objectives in the project charter while meeting three of the four classic project constraints, sometimes called "triple constraints." The four constraints are defined as coverage, schedule, budget and performance [2].

Researchers and practitioners have created a range of project management tools, but have paid less attention to project risk assessment. International standards, on the other hand, define the principles of risk management but do not specify how the process should be conducted. Therefore, [3] have implemented a number of the most effective risk management strategies. They provide an application to assess both risks and their interactions to determine priorities for further decision making. [4, 5] Use the value measurement technique to model the dynamic domain of risk management; they provide an application to assess both risks and their interactions in order to prioritize subsequent decision-making. They propose a two-stage approach focused on dynamic risk characteristics and multiple attributes, with the first phase focused on Markov risk assessment and the second on TOPSIS risk management strategy collection.

According to [6], President and CEO (Chief Executive Officer) of PMI (Project Management Institute) many organizations do not have productive benefit realization management processes in place, and these organizations are wasting an opportunity to ensure that their programs are having the intended strategic impact. Numerous empirical studies have shown that investments in IT projects can deliver significant business value.

Organizations that view project management as a strategic capability to drive change are now outperforming their competitors [6] Therefore, effective IT project management should enable business and IT leaders to understand how IT has contributed to the achievement of business goals in the past and in the future.

Business case is one of the most useful methods available to management to guide business value development. The effectiveness of the business case has a significant effect on value development [7]. It provides forecasts of potential events to help decide whether to proceed with a project, and it should be reviewed regularly during the economic life cycle of the program.

Limitations of current approaches are primarily due to the ad hoc nature of the structures, each of which responds to a specific strategic project coordination process, as outlined in the previous three paragraphs. Therefore, an integrated IT strategic alignment structure is needed.

The objective of our contribution is to improve the conceptual model of the PMBOK project management system by integrating powerful IT investment management techniques defined from the VAL IT framework. In our case, we use UML meta-modeling and RDFS representation to model the strategic alignment principles of IT projects and their interactions, with the UML model translated into RDFS ontology. We continued the process of building our ontology from business objects based on our UML business model to populate a knowledge base structured by the created ontology.

According to [8], the OMG maintains the UML standard. Originally, this software design language allowed users to define the objects that would be manipulated in a program (class diagram). The UML has been used in different research domains: it can be used to structure meta-models in the context of a Model Driven Architecture (MDA) approach [9], to model an ontology [10] and to define domain models for business in an architecture system.

RDFS (RDF-Schema), a Resource Description Framework (RDF) extension, allows to develop lightweight ontologies based on RDF. Since 2004, it is a W3C recommendation [11]. Allows the development of RDF vocabularies, including the naming of classes, subclass relations and subproperty relations, as well as the typing of predicates (domain, co-domain, etc.). It allows to define a hierarchical organization of classes and properties. RDFS is thus a first ontology definition language [11]. RDFS has a reduced expressiveness, allows simple inferences (compared to other ontology definition languages like Web Ontology Language (OWL)), but too much expressiveness is not necessarily a quality [11]. In the next section, we explain in more detail the approach taken for the construction of our ontology. Here's the truth about investment risk based on the analysis of the work cited above: all investments carry some risk. IT managers can't completely eradicate investment risk, but they can have more influence on what happens to the money they invest by recognizing the different types of risk and employing unique investment techniques to better manage those risks and prevent problems.

Received: June 24, 2021. Revised: July 24, 2021.

To address these needs, we propose an ontology for IT projects that will enable organizations to achieve maximum value with reasonable risk and low cost. This paper examines how IT project management systems, IT risk models, and IT governance frameworks interact. Our contribution hopes to present the most successful risk management approaches reported in the research literature.

The rest of this paper is organized as follows: we describe the approach adopted for the construction of our ontology in Section 2. Conceptual model is presented in Section 3. RDFS ontological representation of proposed ontology is established in Section 4. We validate our ontology in Section 5. Finally, we discuss the main results of our work and conclude with a conclusion.

2. Adopted approach

Ontology definition metamodel (ODM) is applicable to knowledge representation, conceptual modeling, formal taxonomy development and ontology definition, and allows the use of a variety of business models as starting points for ontology development through mappings to UML and MOF. ODM-based ontologies can be used to support: -knowledge exchange between heterogeneous computer systems, -knowledge representation in ontologies and knowledge bases, -specification of expressions that are the input or output of inference engines [12].

Our choice of UML as the basis for describing our ontology might seem a simple alternative. A quick look at ontology description languages and the UML reveals the existence of a number of common notions: classes, relations, properties, inheritance, etc. However, there are several significant differences between the two languages. Nevertheless, there are several significant differences between these concepts [11]. The most important difference concerns the notion of property - in UML an attribute has a scope linked to the class, contrary to ontologies in which a property represents a first level concept that can exist independently of a class [11].

ODM proposes three meta-models (RDFS, OWL and Topic Maps) for the most used formalisms currently in the Semantic Web community according to the work of [11]. The UML profiles existing in ODM provide a bridge between UML and the different knowledge representation languages [11].

In this paper, we adopt the approach proposed by [11], as it is the most adapted to our need. It consists in transforming our business model expressed in UML into an ontology expressed in RDFS on the basis of the transformations and correspondences described in Table 1.

Table 1. Transformations performed between UML and RDFS

[UML]	[RDFS]
[Class]	[rdfs:Class]
[Generalization]	[rdfs:subClassOf]
[Association]	[rdf:Property]
[Attribute]	[rdf:Property]
[InstanceOf]	[rdf:type]
[Attribute type : String]	[rdfs:Literal]
[Attribute]	[rdf:value]

Source: [11]

Based on the ODM-based mappings in Table 1 between the UML elements and the elements of our RDFS ontology, we derive our pivotal model to build our RDFS trees. And since we have a UML model representing the ontology and ODM provides the transformations from UML to RDFS and OWL, we can generate an ontology expressed in OWL under the PROTÉGÉ 5 tool.

3. Proposed work

3.1 Design of proposed model

Project Management Framework: According to [13], project management is an iterative process that involves the application of expertise, resources, experience, and strategies to meet the needs and expectations of project stakeholders.

PMBOK (Project Management Body of Knowledge): guide is a collection of project management terms and information. It covers the project life cycle and stages, project stakeholders, organizational and socio-economic factors, and general management skills required by the project manager. It is a mix of 9 areas of expertise and 5 process groups with 49 key process areas (KPA).

Prince2: Since its creation in 1996, PRINCE2 has become the most widely used project management methodology in the world. The most recent version of the book was published in 2010. It is commonly used in the private sector, as well as by governments and public institutions. PRojects IN Controlled Environments (PRINCE2) is no longer just for IT projects; it is also used for R&D, product development and marketing.

Comparison and analysis of PMBOK and PRINCE2: The characteristics and final objective of the two methodologies discussed are similar. We propose a simple comparison of the structure of

Table 2. PMBOK and Prince2 comparison

[PRINCE2]	[PMBOK Guide]
[7 principles]	[No principles]
[7 thems]	[10 knowledge Areas]
[7 process]	[5 process groups]
[41 activities]	[49 processes]
[2 techniques described in detail and 40 techniques referenced]	[119 tools and technique described]

PMBOK and PRINCE2 (table 2).

The PMBOK guide includes many examples of methods, strategies, and procedures; it is described as descriptive, whereas Prince2 is often described as perspective. The PMBOK describes the outcomes of each process but does not specify what information should be documented in those outcomes.

Unlike the PMBOK, the Prince2 guide explains what information should be documented in the outputs of its processes, as well as who is responsible for it. The main strength of the PMBOK guide is that it includes a variety of useful tools and techniques. (There are 119 tools, compared to only 40 in the PRINCE2 guide).

The most important feature of PRINCE2 is that the majority of decisions must be made on the basis of a business case. Prince2 provides a clearer picture of the benefits versus the costs. In addition, Prince2 describes in detail the tasks of 9 separate positions on the project management team. Prince2's biggest flaw is its lack of resources and techniques; it only covers two.

Both Prince2 and PMBOK are designed to set out the process, not to tell managers how to use any of the strategies or resources mentioned.

Project Risk Management Models: In the literature, there are many risk maturity models, including the PMI risk management model, the M O R guideline, and the ISO31000 standard.

PMI Risk Management Model: Risk management is one of the ten areas of expertise in which a project manager must be proficient, according to the PMBOK Project Management Institute [14]. Risk management is characterized as an organizational strategy to maximize returns while minimizing the risk of failure. The PMBOK guide is used by PMI to establish a standard of practice for project risk management. It describes the function of risk management and provides a list of key success factors for implementing risk management [14]. Most organizations and programs, regardless of scale, will benefit from these considerations.

M_O_R guide: M o R is a risk management guide provided by the UK government's Office of Fair Trading. The M o R guide provides a risk management maturity model, or "health check," and a structure that consists of four key steps: Identify, Assess, Plan, Implement.

ISO 31000 STANDARD: The ISO Technical Management Board working group has developed ISO 31000: 2009, an international standard focusing on risk principles and guidelines. It serves as a roadmap for defining and managing risk.

Comparison of Project Risk Management Standards: unlike ISO, which is a stand-alone risk management standard, the PMI implementation is deeply integrated into the PMBOK framework. The main advantage of the PMI risk management standard is that it provides much more detailed information on how to apply the method and distinguishes between qualitative and quantitative risk analysis. The main drawbacks of the PMI risk management standard are that it excludes the articulation of organizational objectives and the risk assessment methodology.

ISO 31000, like the PMI system, emphasizes the importance of considering the context in which risk management is used. In addition, the risk management processes are equivalent to those proposed in the PMI management system and do not conflict. ISO 31000 deals with risk in general, while the PMI system deals with project management risk. It divides risks into three categories: qualitative, semi-quantitative and quantitative risks. M_O_R is based on ISO 31000, but it emphasizes practical implementation and offers more knowledge about risk management.

For a variety reasons, we chose PMI guides, PMBOK, and the risk management standard for our contribution:

- PMI standards have been widely disseminated and recognized worldwide [13].

- PMI frameworks provide additional project management information, tools, and techniques.

- PMI's Project Risk Management Best Practice Standard is standardized and can be used by any company.

- Finally, PMI is a member of the ISO/PC236 project committee as well as the ISO/TC258 committee, which ensures that the ISO and PMI risk management standards are closely related [13].

Nevertheless, based on the PMBOK guide and the PMI risk management standard, we will propose a project management meta-model.

IT Governance Frameworks: Val IT, along with all the documents in this archive (see Enterprise Value: IT Investment Governance), creates a new benchmark for IT investment governance [7]. It is a systematic and rigorous approach to value assessment and creation that helps organizations make better IT investment decisions and increase return on investment by having consistent policies. Value governance, portfolio management and investment management are the three main areas covered by VAL IT. Each consists of a set of procedures, key management strategies and activities that must be followed in order to select the investments with the greatest potential for value creation. VAL IT is intended to be used in conjunction with COBIT, but it can also be used independently of COBIT [7].

Proposed IT Project Governance Framework: The objective of this section is to present the principles used in our proposed framework for strategic project alignment. We present our model in figure 1 capable of representing the conceptual structure and fundamental semantics of IT system models, which can be used to analyze, integrate and adapt these best practices. We will also attempt to link the components of these models in order to extract the conceptual structures of their relationships. We propose the integrated model in Figure 1, which consists of three packages: a first one for the model derived from PMBOK, a second one for the model derived from the PMI risk management standard, and a third one for the model derived from VAL IT. To explain our meta-model, we will first describe the PMBOK, PMI risk management standard, and VAL IT architectures and then present the relationship between the 3 original packages.

3.2 RDFS ontological representation of proposed model

RDFS ontology representation of the proposed model: In this section, we present our ontology expressed in RDFS (language already quoted and used in our work of [18]) in Figure 2, 3, and 4 which was created through a UML model transformation process. In many domains where information systems are used, the knowledge associated with a career is frequently described by UML models, in particular class diagrams that model the unique entities of the domain [11]. These "business models" abstract a collection of information articulated in a commonly used and standardized language [11].

This formalism has the advantage of being well understood by a wide audience, which makes it a valuable means of communication between software designers [11].

To build our ontology, we used the recent OMG Ontology Definition Meta-model (ODM) proposal cited in section 2 of this paper. Using the UML formalism, this proposal defines meta-models for various information representation languages such as RDFS. Our methodology consists in using the UML profile for RDFS to define our knowledge under RDFS trees. These allow, from the business model expressed in UML in figure 1, to build our ontology expressed with RDFS in figures 2, 3 and 4, respecting the principles of the latter and the transformation rules mentioned in table 1 of section 2 of this paper.

relation 1 between PMBOK and VAL IT: There is a "requires" relation between the concept "program" of the PMBOK ontology and the concept "maturityLevel" of the VAL IT ontology, of which "program" is the domain and "maturityLevel" is the rang. There is a "requires" relation between the concept "program" of the PMBOK ontology and the concepts "result" and "Goal" of the VAL IT ontology, of which "program" is the domain, "result" and "Goal" are the ranges.

relation 2 between Val IT and PMI: The concept "FinancialRisk" of the PMBOK ontology is a subclass of the concept "Risk" of the PMI Risk Management ontology.

Relation 3 between PMI and PMBOK: There is a relatingTo relation between the concept "stakeholders" of the PMBOK ontology and the concept "RiskTolerance" of the PMI Risk Management ontology, of which "stakeholders" is the domain and "RiskTolerance" the rang. There is a "relatingTo" relation between the "Riskmanager" concept of the PMI Risk Management ontology and the "TeamMember" concept of the PMBOK ontology, where "Riskmanager" is the domain and "TeamMember" is the rang. There is an Includes relation between the PMI Risk Management ontology concept "Riskoccurence" and the PMBOK ontology concept "Activity", where "Activity" is the domain and "Riskoccurence" is the rang.

4. Validation of proposed ontology model

As we have already mentioned in section 2 of this paper, we have a UML model representing the ontology and since the ODM provides the transformations from UML to RDFS and OWL, we can therefore generate from the ontologies represented in the previous section an ontology expressed in OWL under the PROTÉGÉ 5 tool.

4.1 Validation of consistency of proposed ontology

The choice of the HERMIT 1.4.3. 456 reasoner to validate the consistency of our ontology was a natural one, since we use the PROTÉGÉ 5 ontology

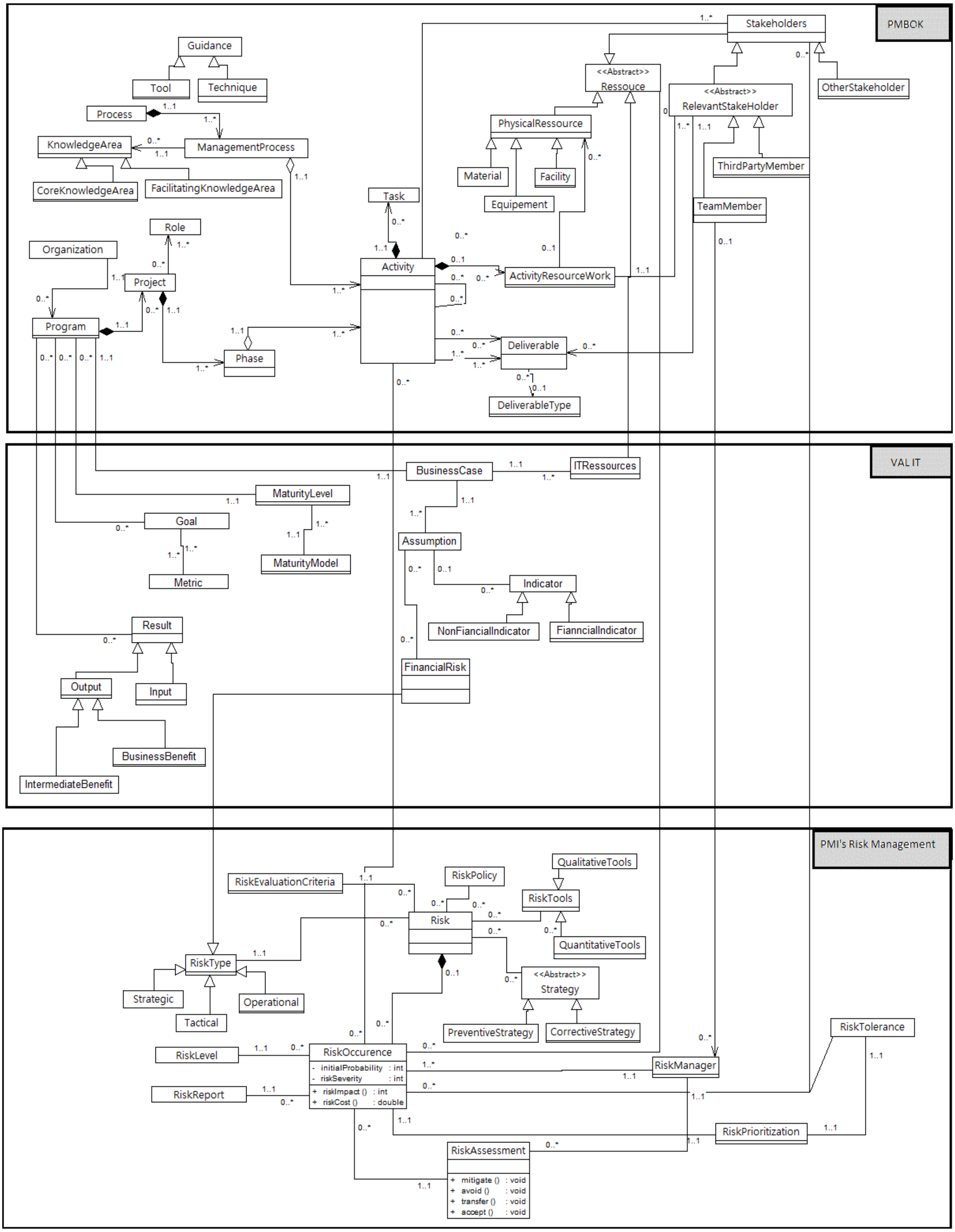


Figure. 1 Integrated model of PMBOK, PMI and VAL IT

editor in which it is integrated. HERMIT 1.4.3. 456 allowed us to verify that our ontology did not contain contradictory classes as it was being developed. Fig. 5, 6 and 7 gives an overview of the inferred classes and object properties generated after applying the reasoning on the concepts of the proposed ontology:

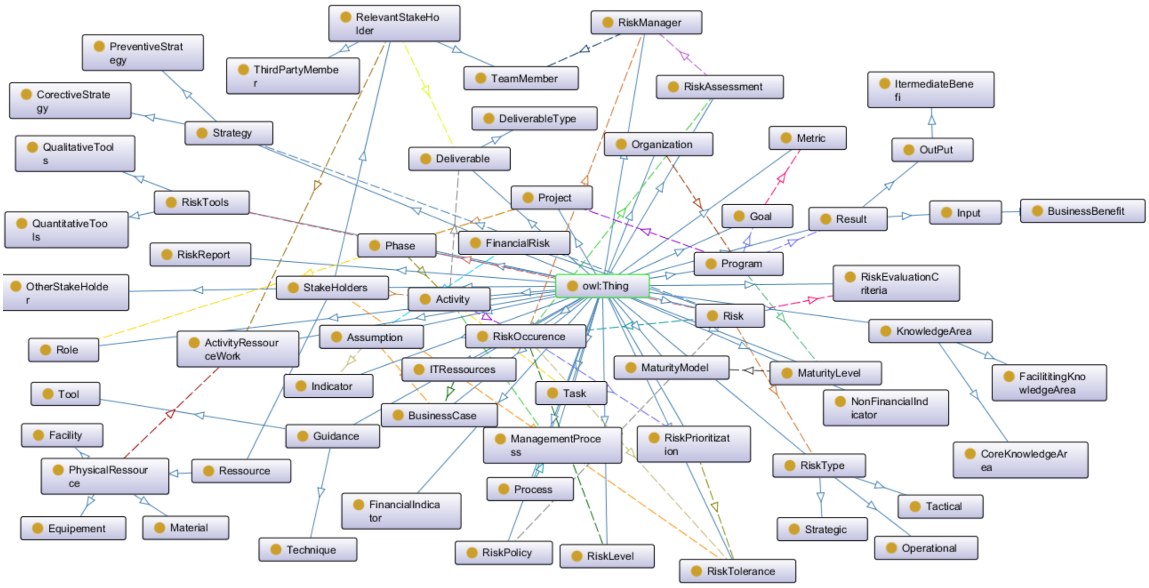
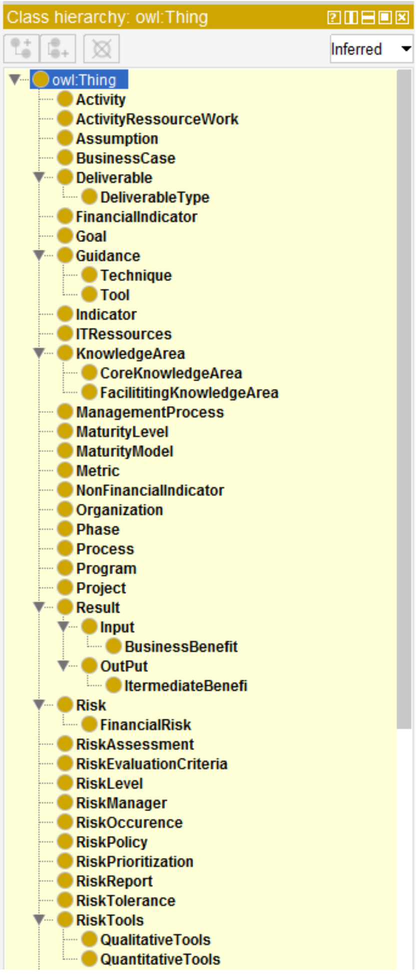


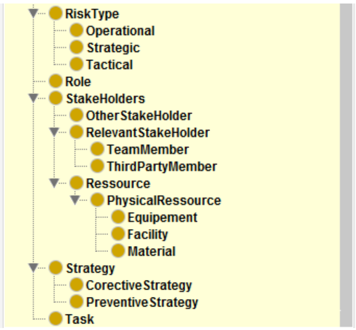
Figure. 2 Generated OntoGraf from RDFS representation of proposed ontology



- owl:Thing
  - Activity
  - ActivityRessourceWork
  - Assumption
  - BusinessCase
  - Deliverable
    - DeliverableType
  - FinancialIndicator
  - Goal
  - Guidance
    - Technique
    - Tool
  - Indicator
  - ITRessources
  - KnowledgeArea
    - CoreKnowledgeArea
    - FacilititingKnowledgeArea
  - ManagementProcess
  - MaturityLevel
  - MaturityModel
  - Metric
  - NonFinancialIndicator
  - Organization
  - Phase
  - Process
  - Program
  - Project
  - Result
    - Input
      - BusinessBenefit
    - OutPut
      - ItermediateBenefi
  - Risk
    - FinancialRisk
  - RiskAssessment
  - RiskEvaluationCriteria
  - RiskLevel
  - RiskManager
  - RiskOccurence
  - RiskPolicy
  - RiskPrioritization
  - RiskReport
  - RiskTolerance
  - RiskTools
    - QualitativeTools
    - QuantitativeTools

Figure. 3 Classes inferred from the reasoning of proposed ontology

4.2 Validation of structure of proposed ontology



- RiskType
  - Operational
  - Strategic
  - Tactical
- Role
- StakeHolders
  - OtherStakeHolder
  - RelevantStakeHolder
    - TeamMember
    - ThirdPartyMember
  - Ressource
    - PhysicalRessource
      - Equipement
      - Facility
      - Material
- Strategy
  - CorectiveStrategy
  - PreventiveStrategy
- Task

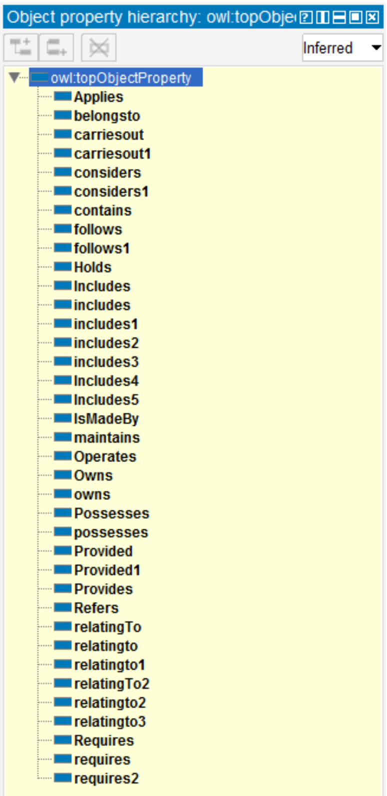
Figure. 4 Classes inferred from the reasoning of proposed ontology

Following our study made of the state of the art concerning the validation of the structure of an ontology, we were able to choose the OOPS! tool for several reasons: the availability of the tool, its regular update, the criteria used which were defined following a study of the domain, the possibility of keeping its source code private, the free nature, the independence of the module, the use under any web browser [15].

OntOlogy Pitfall Scanner (OOPS!) [16] is a tool that is independent of any ontology editor [16]. The objective of OOPS! is the identification of anomalies or bad practices in an ontology [16]. For this purpose, the authors have defined a number of pitfalls listed in natural language in a catalog.

There are currently 40 of them, 32 of which are implemented as java classes and added to the module. As input, the application takes the URI of an ontology or the source code in RDF format. The ontology is loaded via the Jena API before being parsed for potential errors. The result is a web page on which the pitfalls (identified errors) are listed as

Received: June 24, 2021. Revised: July 24, 2021.



Object property hierarchy: owl:topObje

Inferred

- owl:topObjectProperty
  - Applies
  - belongsto
  - carriesout
  - carriesout1
  - considers
  - considers1
  - contains
  - follows
  - follows1
  - Holds
  - Includes
  - includes
  - includes1
  - includes2
  - includes3
  - Includes4
  - Includes5
  - IsMadeBy
  - maintains
  - Operates
  - Owns
  - owns
  - Possesses
  - possesses
  - Provided
  - Provided1
  - Provides
  - Refers
  - relatingTo
  - relatingto
  - relatingto1
  - relatingTo2
  - relatingto2
  - relatingto3
  - Requires
  - requires
  - requires2

Figure. 5 Object Properties inferred from the reasoning of proposed ontology

well as a resolution proposal. The pitfalls may concern individual elements, several elements, or the entire ontology [16].

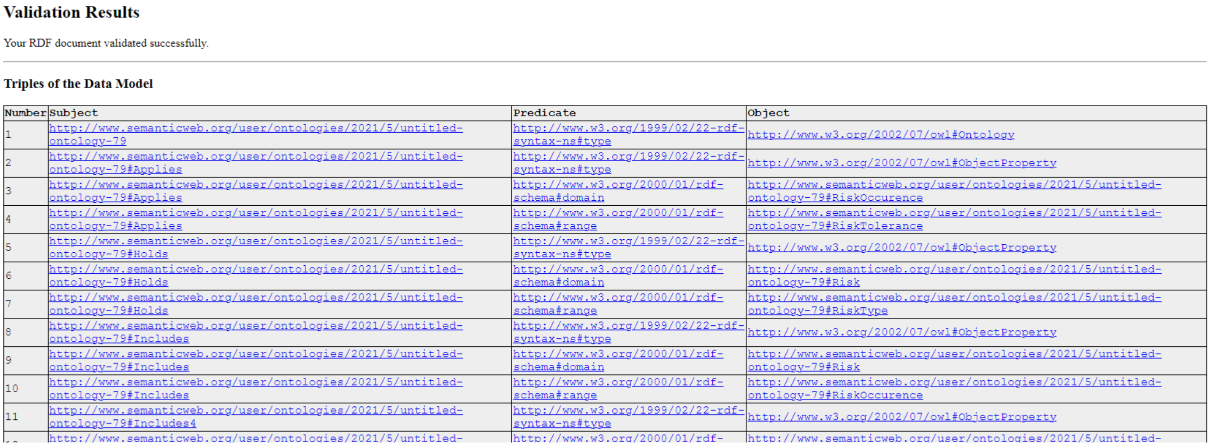
Before we use OOPS! we ensure that the RDF code of our ontology is correct using the W3 RDF validator [17], Fig. 8 gives an overview of the verification result of the RDF code of our ontology

To evaluate the structure of our ontology, we use OntOlogy Pitfall Scanner (OOPS!) by copying the RDF code (already validated in the previous step) of our PROTÉGÉ 5 edited ontology into the OOPS! (as mentioned in Fig. 9) to extract the results shown in Figure 10.

5. Results

Meaning of the pitfalls that appear in our evaluation results (according to the OOPS! tool catalog), these usually appear during ontology catalog), these usually appear during ontology development. Some of them are very common and do not cause any problems:

P04. Creation of unconnected ontology elements: ontology elements (classes, object properties and data type properties) are created in isolation, without any relation to the rest of the



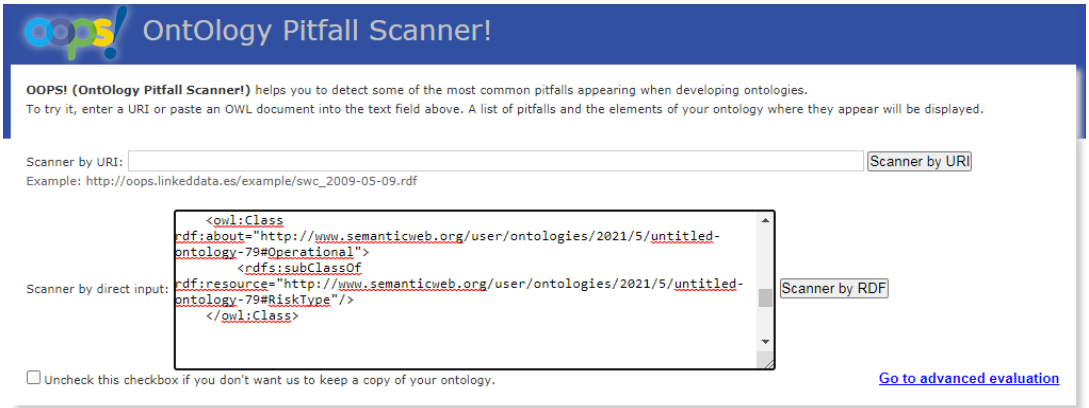
Validation Results

Your RDF document validated successfully.

Triples of the Data Model

Number	Subject	Predicate	Object
1	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Ontology
2	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Applies	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#ObjectProperty
3	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Applies	http://www.w3.org/2000/01/rdf-schema#domain	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#RiskOccurence
4	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Applies	http://www.w3.org/2000/01/rdf-schema#range	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#RiskTolerance
5	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Holds	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#ObjectProperty
6	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Holds	http://www.w3.org/2000/01/rdf-schema#domain	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Risk
7	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Holds	http://www.w3.org/2000/01/rdf-schema#range	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#RiskType
8	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Includes	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#ObjectProperty
9	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Includes	http://www.w3.org/2000/01/rdf-schema#domain	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Risk
10	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Includes	http://www.w3.org/2000/01/rdf-schema#range	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#RiskOccurence
11	http://www.semanticweb.org/user/ontologies/2021/5/untitled-ontology-79#Includes4	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#ObjectProperty
	http://www.semanticweb.org/user/ontologies/2021/5/untitled-	http://www.w3.org/2000/01/rdf-	http://www.semanticweb.org/user/ontologies/2021/5/untitled-

Figure. 6 Overview of the RDF code validation of the proposed ontology



OOPS! (OntOlogy Pitfall Scanner!) helps you to detect some of the most common pitfalls appearing when developing ontologies.
To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI: [Scanner by URI]
Example: http://oops.linkeddata.es/example/swc_2009-05-09.rdf

Scanner by direct input:

```
    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2021/5/untitled-
ontology-79#Operational">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2021/5/untitled-
ontology-79#RiskType"/>
    </owl:Class>
```

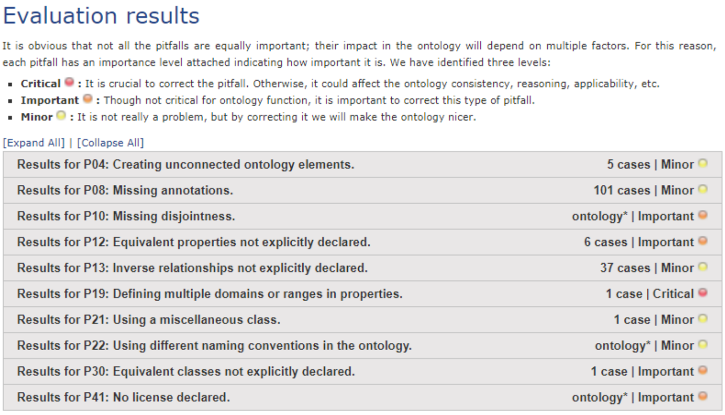
[Scanner by RDF]

□ Uncheck this checkbox if you don't want us to keep a copy of your ontology.

Go to advanced evaluation

Figure. 7 Overview of the use of the OOPS! Tool

Received: June 24, 2021. Revised: July 24, 2021.



Evaluation results

It is obvious that not all the pitfalls are equally important; their impact in the ontology will depend on multiple factors. For this reason, each pitfall has an importance level attached indicating how important it is. We have identified three levels:

- **Critical** ● **:** It is crucial to correct the pitfall. Otherwise, it could affect the ontology consistency, reasoning, applicability, etc.
- **Important** ● **:** Though not critical for ontology function, it is important to correct this type of pitfall.
- **Minor** ○ **:** It is not really a problem, but by correcting it we will make the ontology nicer.

[Expand All] | [Collapse All]

Results for P04: Creating unconnected ontology elements.	5 cases | Minor ○
Results for P08: Missing annotations.	101 cases | Minor ○
Results for P10: Missing disjointness.	ontology* | Important ●
Results for P12: Equivalent properties not explicitly declared.	6 cases | Important ●
Results for P13: Inverse relationships not explicitly declared.	37 cases | Minor ○
Results for P19: Defining multiple domains or ranges in properties.	1 case | Critical ●
Results for P21: Using a miscellaneous class.	1 case | Minor ○
Results for P22: Using different naming conventions in the ontology.	ontology* | Minor ○
Results for P30: Equivalent classes not explicitly declared.	1 case | Important ●
Results for P41: No license declared.	ontology* | Important ●

Figure. 8 Overview of the results of the evaluation of the proposed ontology under OOPS! Tool

ontology. This trap is mentioned (MINOR), so it is not a problem and it is justified by the fact that we originally have three distinct packages and we connect some elements of them according to the relation 1, 2 and 3 mentioned in previous section.

P08. Missing Annotations: This pitfall consists of creating an ontology element and not providing human-readable annotations attached to it. As a result, ontology elements lack annotation properties that label them. This pitfall is mentioned (MINOR), and justified by the fact that annotations are not mentioned.

P10. Missing disjunction: The ontology lacks disjunction axioms between classes or between properties that should be defined as disjoint. This is indeed a non-critical trap for the ontological function of our ontology.

P12. Equivalent properties not explicitly declared: The ontology lacks information about equivalent properties (owl:equivalentProperty) in cases of duplicate relations and/or attributes. Although not critical to the ontological function of our ontology.

P13. Inverse relations not explicitly declared: This trap appears when a relation (except those defined as symmetric properties using owl:SymmetricProperty) does not have an inverse relation (owl:inverseOf) defined in the ontology. This trap is mentioned (MINOR), so it does not pose a problem for the functioning of our ontology.

P19. Definition of multiple domains or ranges in properties: The domain or range (or both) of a property (relations and attributes) is defined by stating more than one rdfs:domain or rdfs:range statement. In OWL, multiple rdfs:domain or rdfs:range axioms are allowed, but they are interpreted as a conjunction and are therefore equivalent to the owl:intersectionOf construct. This trap is in '1case' and is not a problem because we are allowed to assign more than one "domain" or "range" for a relation. And this is the case for the relation "requires2" in our ontology.

P21. This trap refers to the creation of a class for the sole purpose of classifying instances that do not belong to any of its sister classes (classes with which the diverse problematic class shares a common direct ancestor). This trap is mentioned (MINOR), and appears in only one case which is: "OtherStakeHolder". This means that it does not pose a problem in the functioning of our ontology.

P22. Use of different naming conventions in the ontology. This trap is mentioned (MINOR), so it does not cause any problem in the functioning of our ontology.

P30. This trap consists in not defining equivalent classes (owl:equivalentClass) in the case of duplicated concepts. When an ontology reuses terms from other ontologies, classes that have the same meaning must be defined as equivalent in order to promote interoperability between the two ontologies. This trap appears only in one case which is: "Task" which we rectify by defining the equivalent class.

P41. No license declared.

After having evaluated our ontology, we expose through the graphical representation of Fig. 9, the set of metrics resulting from the reasoning of the latter.

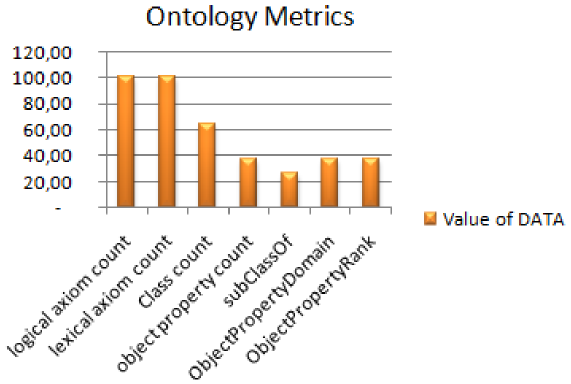


Figure. 9 Metrics of the proposed ontology after evaluation

6. Conclusion

The main objective of this paper was to better understand the concept of alignment in order to create a more effective strategic alignment framework for IT projects complementing our previous work [19, 20]. Practitioners and academics have proposed a variety of frameworks and methodologies. However, the adhoc structure of the presented frameworks, each catering to a specific strategic project coordination process, is the source of the shortcomings of the current methods. Therefore, an integrated model for strategic alignment of IT projects is needed.

We have compared the most used frameworks and methodologies within organizations and have chosen the PMI standards and the VAL IT system for this purpose. As a result, our conceptual model is based on a combination of the PMI standards and the VAL IT information domains, which are extracted from their manuals.

The PMI guidelines allow project risk management to begin early in the project life cycle by including key stakeholders, assessing project risks on a regular basis during the project life cycle, and developing risk mitigation strategies, thereby better aligning projects with the business strategy. In addition, we use VAL IT best practices to federate the financial aspects of IT projects, integrate the business side of IT decision making, and provide investment areas. It also provides guidelines for tracking the true value of a project, even as the business need for that project changes.

Incorporating these best practices helps identify and measure business priorities, assess project benefits, evaluate project costs, and allocate resources to projects, resulting in a reliable IT project alignment process. Our contribution highlights the increased need to design this knowledge into an ontology that integrates IT governance best practices to produce more effective results.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Document conceptualization, methodology, software and validation were performed by the first author. Formal analysis, investigation, resources, data curation, drafting, editing, and visualization were performed by the 1st and 2nd authors. Supervision and administration of the project was performed by the 3rd and 4th authors.

References

[1] E. Brams, *La gestion de l'autonomie en contexte de gestion de projet*, 2018.

[2] A. Abyad, “Project management, motivation theories and process management”, *Middle East Journal of Business*, Vol. 1, No. 4, pp. 18-22, 2018.

[3] U. I. Janjua, J. Jaafar, and I. B. A. Aziz, “Integration of Supportive processes with elementary Processes for making current practices of software Project Risk Management more effective”, In: *Proc. of International Symposium on Mathematical Sciences and Computing Research (iSMSC)*, pp. 292-297, 2015.

[4] C. Fang, F. Marle, and M. Xie, “Applying importance measures to risk analysis in engineering project using a risk network model”, *IEEE Systems Journal*, Vol. 11, No. 3, pp.1548-1556, 2016.

[5] G. Jiang and Y. Fu, “A two-phase method based on Markov and TOPSIS for evaluating project risk management strategies”, In: *Proc. of the 27th Chinese Control and Decision Conference (CCDC)*, pp. 1994-1998, 2015.

[6] M. A. Langley, “The strategic impact of projects identify benefits to derive business results”, *Project Management Institute*, 2016.

[7] IT Governance Institute, *Enterprise Value: Governance of IT Investments, the Val IT Framework*, Version 2. 0. ISACA, 2008.

[8] R. Platt and N. Thompson, “The Past, Present, and Future of UML”, In: *Proc. of Advanced Methodologies and Technologies in Network Architecture, Mobile Computing, and Data Analytics*, pp. 1452-1460, 2019.

[9] S. J. Mellor, K. Scott, A. Uhl, and D. Weise, “Model-driven architecture”, In: *Proc. of International Conference on Object-Oriented Information Systems*, pp. 290-297, 2002.

[10] S. Cranefield, “Networked knowledge representation and exchange using UML and RDF”, *Journal of Digital Information*, Vol. 1, No. 8, 2001.

[11] C. Faucher, F. Bertrand, and J. Y. Lafaye, “Ontology generation from an annotated UML business model”, *Journal of New Information Technologies*, Vol. 12, pp. 65-84, 2008.

[12] OMG, “FOUNDATION FOR APPLYING BUSINESS SEMANTICS”, 2021. [Online]. Available: https://www.omg.org/odm/.

[13] E. S. Dunne, “Project risk management: Developing a risk framework for translation

projects”, *Doctoral Dissertation, Kent State University*, 2013.

[14] PMI Standards Committee, and Project Management Institute, *A guide to the project management body of knowledge*, 1996.

[15] M. Richard, X. Aimé, M. O. Krebs, and J. Charlet, “LOVMI: Towards an interactive method for ontology validation”, In: *Proc. of 26th French-speaking days of Knowledge Engineering (KE)*, 2015.

[16] M. P. Villalón, M. C. S. Figueroa, and A. G. Pérez, “ Validating ontologies with oops! ”, In: *Proc. of International Conference on Knowledge Engineering and Knowledge Management*, pp. 267-281, 2012.

[17] D. Brickley, “RDF vocabulary description language 1.0: RDF schema”, http://www. w3. org/TR/rdf-schema/, 2004.

[18] L. Moudoubah, K. E. Guemmat, K. Mansouri, and M. Qbadou, “Elaboration of Psychological Ontology that Integrates the Problem of Human Frailty in COVID-19 Context”, *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 3, pp. 388-403, 2021.

[19] L. Moudoubah, K. Mansouri, and M. Qbadou, “Towards an ontological analysis of the alignment problems of fields in the architecture of an information system”, In: *Proc. of Embedded Systems and Artificial Intelligence*, pp. 785-794, 2020.

[20] L. Moudoubah, A. E. Yamami, K. Mansouri, and M. Qbadou, “Towards the implementation of an ontology based on COBIT framework (CobitOnyology)”, In: *Proc. of 1st International Conference on Smart Systems and Data Science (ICSSD)*, pp. 1-6, 2019.