

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



Departamento:
Departamento de Ingeniería e Investigaciones Tecnológicas

Programa de acreditación:
PROINCE

Programa de Investigación¹:

Código del Proyecto: C206

Título del proyecto

“Gestión de requisitos generados colaborativamente”

Director:

Mg. Santa María Cristóbal Raúl

Codirector:

Dra. Litvak Claudia Silvia

Integrantes:

Mg. Gigante Nora Cristina

Investigador Externo, Asesor- Especialista, Graduado UNLaM:

Dr. Rossi Gustavo (UNLP), Dr. Antonelli Leandro (UNLP), Dr. Gambo Ishaya Peni (OAU)

Alumnos de grado:

Lo Giudice Diego Andrés, Velazquez Santillán Facundo Darío, Milla Ximena Anabela

Fecha de inicio: 1/1/2018

Fecha de finalización: 31/12/2019

¹ Los Programas de Investigación de la UNLaM están acreditados con resolución rectoral, según lo indica la Resolución HCS N° 014/15 sobre **Lineamientos generales para el establecimiento, desarrollo y gestión de Programas de Investigación a desarrollarse en la Universidad Nacional de La Matanza**. Consultar en el departamento académico correspondiente la inscripción del proyecto en un Programa acreditado.

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



A. Desarrollo del proyecto (adjuntar el protocolo)

A.1. Grado de ejecución de los objetivos inicialmente planteados, modificaciones o ampliaciones u obstáculos encontrados para su realización (desarrolle en no más de dos (2) páginas)

La Ingeniería de Requisitos, una de las etapas iniciales en la Ingeniería de Software, tiene como objetivo describir lo más fielmente posible, las necesidades y expectativas de los clientes y usuarios. A tal fin, los ingenieros de requisitos desarrollan modelos, intentando obtener modelos de alta calidad.

El trabajo de varios ingenieros de requisitos de modo colaborativo, ha surgido como una buena posibilidad para mejorar la calidad de los modelos. Aunque como en todo trabajo colaborativo surgen conflictos, inherentes a toda interacción humana. Dichos conflictos deben ser resueltos a fin de mejorar la calidad de los modelos.

En este proyecto de investigación, se realizó inicialmente una revisión bibliográfica del tema de conflictos, estudiando los conflictos existentes en la generación de requisitos. Luego se investigaron posibles propuestas en la generación colaborativa de requisitos.

También se realizó el estudio comparativo de los conflictos en ontologías en general. Se estudió específicamente la construcción colaborativa del LEL, a través de casos de estudio realizados a tal fin. Se definió un catálogo de conflictos, definiendo cada situación y las soluciones de cada conflicto.

Se definió un proceso definido a fin de resolver los conflictos de modo colaborativo, y una clasificación de dichos conflictos. Se presentaron tres casos de estudio que muestran la efectividad del proceso, realizados en empresas reales de Argentina. Dichas empresas fueron Megaflex S.A, una fábrica de productos asfálticos para impermeabilización, Assistcard Argentina, compañía dedicada a la asistencia al viajero, y Textil Iberá (parte del grupo Norfabril), empresa productora de hilados.

De este modo se logró brindar un marco de análisis para resolver los conflictos que surgen al desarrollar modelos de requisitos de modo colaborativo.

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



Para validar el catálogo de conflictos, se diseñó y realizó un experimento en el que intervinieron 29 personas, que permitieron validar las características de los conflictos hallados y las soluciones propuestas. Estas personas fueron profesionales del área de sistemas dedicados a diferentes tareas en empresas de Nigeria, siendo todos ellos estudiantes o graduados de posgrado. Para este experimento se trabajó con el caso de estudio IP Etiquetas S.A., compañía que produce etiquetas autoadhesivas, sita en el Parque Industrial de Burzaco, Provincia de Bs. As.

Dicho experimento fue definido en Argentina, pero realizado en Nigeria, donde se presentó a los participantes del experimento una situación real, con cada conflicto y su solución. La realización del experimento en forma totalmente externa garantiza la calidad de los resultados obtenidos.

Además se aplicó el System Usability Scale, para evaluar la usabilidad de las soluciones planteadas, para lo cual se definió un formulario acorde que fue completado por cada uno de los participantes del experimento. Este formulario es una adaptación del formulario original planteado por Brooke en el año 1996.

Como conclusión se estableció que existen dichos conflictos y que su resolución colabora a la calidad del modelo colaborativo, evitando superposiciones o inconsistencias entre diferentes ingenieros de requisitos intervinientes.

Todos los objetivos planteados al inicio del proyecto se han cumplido, habiéndose estudiado en profundidad los conflictos en el desarrollo colaborativo de los modelos, habiendo definido un catálogo de conflictos y habiendo validado los conflictos y sus soluciones mediante experimentación.

Se propone a futuro trabajar en la validación de estos conflictos en casos reales, es decir aplicar el proceso desde adentro de empresas reales, y evaluar nuevamente los resultados obtenidos.

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



B. Principales resultados de la investigación

B.1. Publicaciones en revistas (informar cada producción por separado)

Artículo 1:	
Autores	
Título del artículo	
N° de fascículo	
N° de Volumen	
Revista	
Año	
Institución editora de la revista	
País de procedencia de institución editora	
Arbitraje	Elija un elemento.
ISSN:	
URL de descarga del artículo	
N° DOI	

B.2. Libros

Libro 1	
Autores	
Título del Libro	
Año	
Editorial	
Lugar de impresión	
Arbitraje	Elija un elemento.
ISBN:	
URL de descarga del libro	
N° DOI	

B.3. Capítulos de libros

Autores	
Título del Capítulo	
Título del Libro	
Año	
Editores del libro/Compiladores	
Lugar de impresión	
Arbitraje	Elija un elemento.
ISBN:	

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



URL de descarga del capítulo	
N° DOI	

B.4. Trabajos presentados a congresos y/o seminarios

Autores	<i>Claudia Litvak, Gustavo Rossi, Leandro Antonelli</i>
Título	<i>Conflict Management in the Collaborative Description of a Domain Language</i>
Año	<i>2018</i>
Evento	<i>SEKE 2018 - Software Engineering and Knowledge Engineering 2018</i>
Lugar de realización	<i>San Francisco- California- EEUU</i>
Fecha de presentación de la ponencia	<i>Julio 2018</i>
Entidad que organiza	<i>SEKE</i>
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	

Autores	<i>Claudia Litvak, Gustavo Rossi, Leandro Antonelli Nora Gigante</i>
Título	<i>Improving the identification of conflicts in collaborative requirements engineering</i>
Año	<i>2018</i>
Evento	<i>CSCI 2018</i>
Lugar de realización	<i>Las Vegas- Nevada-</i>



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

	<i>EEUU</i>
Fecha de presentación de la ponencia	<i>Diciembre 2018</i>
Entidad que organiza	<i>CSCI</i>
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	

Autores	<i>Claudia Litvak, Leandro Antonelli, Gustavo Rossi, Ishaya Peni Gambo, Facundo Velazquez Santillán, Diego Lo Giudice, Ximena Milla, Nora Gigante</i>
Título	<i>Resolución de conflictos en modelos de dominio en lenguaje natural: casos de estudio</i>
Año	<i>2019</i>
Evento	<i>CONAISI 2019</i>
Lugar de realización	<i>UNLaM- San Justo- Prov. de Bs. As.</i>
Fecha de presentación de la ponencia	<i>Noviembre 2019</i>
Entidad que organiza	<i>CONAISI</i>
URL de descarga del trabajo (especificar solo si es la descarga del trabajo; formatos pdf, e-pub, etc.)	

B.5. Otras publicaciones

Autores	
Año	
Título	

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



Medio de Publicación	
-------------------------	--

C. Otros resultados. Indicar aquellos resultados pasibles de ser protegidos a través de instrumentos de propiedad intelectual, como patentes, derechos de autor, derechos de obtentor, etc. y desarrollos que no pueden ser protegidos por instrumentos de propiedad intelectual, como las tecnologías organizacionales y otros. Complete un cuadro por cada uno de estos dos tipos de productos.

C.1. Títulos de propiedad intelectual. Indicar: Tipo (marcas, patentes, modelos y diseños, la transferencia tecnológica) de desarrollo o producto, Titular, Fecha de solicitud, Fecha de otorgamiento

Tipo	Titular	Fecha de Solicitud	Fecha de Emisión

C.2. Otros desarrollos no pasibles de ser protegidos por títulos de propiedad intelectual. Indicar: Producto y Descripción.

Producto	Descripción

D. Formación de recursos humanos. Trabajos finales de graduación, tesis de grado y posgrado. Completar un cuadro por cada uno de los trabajos generados en el marco del proyecto.

D.1. Tesis de grado

Director (apellido y nombre)	y Autor (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis

D.2 Trabajo Final de Especialización

Director (apellido y nombre)	y Autor (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título del Trabajo Final

D.2. Tesis de posgrado: Maestría

Director (apellido y nombre)	y Tesista (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019



D.3. Tesis de posgrado: Doctorado

Director (apellido y nombre)	Tesista (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título de la tesis

D.4. Trabajos de Posdoctorado

Director (apellido y nombre)	Posdoctorando (apellido y nombre)	Institución	Calificación	Fecha /En curso	Título del trabajo	Publicación

E. Otros recursos humanos en formación: estudiantes/ investigadores (grado/posgrado/ posdoctorado)

Apellido y nombre del Recurso Humano	Tipo	Institución	Período (desde/hasta)	Actividad asignada ²
Milla, Ximena	Estudiante	UNLAM	01-01-18/ 31-12-19	Revisión bibliográfica. Estudio de los conflictos existentes en la generación de requisitos. Estudio de la generación colaborativa de requisitos. Investigación de los conflictos que pudieran surgir como consecuencia del trabajo colaborativo al elaborar modelos

² Descripción de la/s actividad/es a cargo (máximo 30 palabras)

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019



				de requisitos. Realización de casos de estudio.
Lo Giudice, Diego	Estudiante	UNLAM	01-01-18/ 31-12-19	Revisión bibliográfica. Estudio de la generación colaborativa de requisitos. Investigación de los conflictos que pudieran surgir como consecuencia del trabajo colaborativo al elaborar modelos de requisitos. Realización de casos de estudio.
Velazquez Santillan, Facundo	Estudiante	UNLAM	01-01-18/ 31-12-19	Estudio de los conflictos existentes en la generación de requisitos Estudio comparativo de los conflictos en ontologías en general. Búsqueda de las posibles soluciones a los conflictos en el trabajo colaborativo al elaborar modelos

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



				de requisitos. Realización de casos de estudio.
--	--	--	--	--

F. Vinculación³: Indicar conformación de redes, intercambio científico, etc. con otros grupos de investigación; con el ámbito productivo o con entidades públicas. Desarrolle en no más de dos (2) páginas.

Entre los organismos externos se realizó un trabajo coordinado con cursos de grado y posgrado de “Ingeniería de Requerimientos” en UNLP (Universidad Nacional de La Plata, con el Dr. Leandro Antonelli y el Dr. Gustavo Rossi) mediante la colaboración e intercambio de conocimientos.

Además, se estableció relación e intercambio de conocimientos con proyectos de investigación del LIFIA (UNLP, dirigidos por el Dr. Gustavo Rossi) y con proyectos del CONICET.

Con el LIFIA - UNLP se trabajó además en los conceptos de definición de experimentos, dada la experiencia en dicho departamento en la realización de otros experimentos de ingeniería de Software. Se trabajó con el Dr. Matías Urbietta del LIFIA (UNLP) y CONICET, quien posee amplia experiencia en la definición y realización de experimentos en aplicaciones web. Esta cooperación facilitó la definición del experimento realizado en Nigeria.

Este experimento fue específicamente realizado con el Dr. Ishaya Gambo, del departamento de Computer Science and Engineering, Universidad Obafemi Awolowo University, Nigeria. Los participantes del experimento fueron 29 personas, todas ellas estudiantes de posgrado de dicha universidad.

En relación al ámbito productivo, los casos de estudio fueron realizados en empresas de diferentes áreas, donde se tuvo una interacción con personas de diferentes sectores de cada empresa, incluyendo gerentes, directivos, secretarías, directores de producción, y otros (dependiendo de la estructura de cada empresa).

³ Entendemos por acciones de “vinculación” aquellas que tienen por objetivo dar respuesta a problemas, generando la creación de productos o servicios innovadores y confeccionados “a medida” de sus contrapartes.

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



Se trabajó con la empresa Megaflex, una fábrica de productos asfálticos y protección contra la humedad, sita en el Parque Industrial de Burzaco, provincia de Bs. As.

Luego se trabajó con la empresa Assistcard Argentina, empresa de asistencia al viajero, y con la empresa Textil Iberá, parte del grupo Norfabril, productora de hilados que luego son empleados en la producción de pullovers, ponchos y otros productos.

Además se trabajó con la empresa IP Etiquetas, que fabrica todo tipo de etiquetas autoadhesivas, sita en el Parque Industrial de Burzaco, provincia de Bs. As.

G. Otra información. Incluir toda otra información que se considere pertinente.

H. Cuerpo de anexos

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



Anexo I:

Copia de cada uno de los trabajos mencionados en los puntos B, C y D, y certificaciones cuando corresponda.⁴

⁴ En caso de libros, podrá presentarse una fotocopia de la primera hoja significativa o su equivalente y el índice.

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019



1) SEKE 2018

Conflict Management in the Collaborative Description of a Domain Language

Claudia Litvak

DIT

Universidad Nacional de La Matanza
La Matanza, Bt. As., Argentina
clitvak@unlam.edu.ar

Gustavo Rossi¹, Leandro Antonelli

Lifa, Facultad de Informática

Universidad Nacional de La Plata

La Plata, Argentina

{gustavo, lanto}@lifa.info.unlp.edu.ar

¹also CONICET

Abstract—The identification and specification of the requirements of a software system is a difficult task that has the goal of obtaining requirements as correct and complete as possible. It is extremely important that Requirements Engineers understand a domain language in order to write high-quality requirements. Moreover, they must describe (and discuss) the language in a collaborative way in order to consider the different points of view of all stakeholders to assure that the resulting requirements will have more chances to meet their needs. However, collaborative construction implies the occurrence of conflicts that are unavoidable because of ambiguity, overlapping and misunderstanding natural language descriptions. This article relies on the Language Extended Lexicon in order to describe the application domain. Although it is a semi-structured glossary and this characteristic helps to reduce the conflicts, our experience shows that conflicts arise anyway. Thus, in order to mitigate this problem, this article presents a catalogue with a set of conflicts that could appear during a collaborative construction of the Language Extended Lexicon and proposes alternatives for their resolution.

Keywords—requirements engineering, collaboration, conflicts, natural language models

1. INTRODUCTION

Requirements Engineering is one of the initial stages of the Software Development Life Cycle. The goal of this stage is to acquire the knowledge and the requirements needed for the system to be built. Errors made in requirements specifications have a great impact towards the end of software development, since the cost of error correction increases as each stage progresses [1].

Several authors argue that the interaction of different stakeholders working collaboratively on the same problem improves the quality of the system requirements [2] [3]. Since different stakeholders have different concerns and different point of view, all of them working together will produce a richer model.

However, generating models collaboratively implies the emergence of conflicts that must be solved in order to build a consistent high quality model. The existence of a conflict is not a negative situation, in fact it might be positive since it

provides the possibility of improving the models, analyzing and discussing the different ideas observed and manifested by the conflict.

In this context, it is even more important, to define a basic language in order to interact and describe the needed models. There are two main kinds of languages: formal and natural language. Despite the introduction of ambiguity, the natural language has the advantage to be understood by all the stakeholders (technical and non technical).

Ambiguity means having two interpretations for the same word. For example, let's consider that the word "label" has two different meanings: (i) "It is the action of putting the brand of the product on the boxes of finished product"; and (ii) "It is the action of marking the price of each finished box of finished product". Imagine a situation where two stakeholders use the same word with different meaning: they would think they understand each other, but in fact, they want to transmit a different idea. An opposite situation could be the use of two different words, which in fact are synonyms and represent the same idea. In this case, both stakeholders can not know that they are talking about the same thing.

Our research is framed by the Language Extended Lexicon (LEL). The LEL is a model that uses Natural Language [4] to describe the vocabulary of the application domain. The LEL is a very convenient tool for stakeholders with no technical skills, although people with such skills will profit more from its use [5]. In particular, the convenience of the LEL as a tool arises from three significant characteristics: it is easy to learn, it is easy to use and it has good expressiveness. Goel [6] states that the LEL is widely used to capture the language to describe requirements. Moreover, it is a useful technique because can be understood by the stakeholders, and this characteristic encourage their active participation which is crucial in first steps of software development.

The LEL captures the terms (they are called symbols) and describes them with the name, the notion, and the behavioral responses. The name identifies the symbol; all synonyms that exist in the domain must be defined in this attribute. The notion describes the meaning (denotation) and the behavioral responses describe the relation of the symbol with other

DOI reference number: 10.18293/SEKE2018-106



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

symbols (connotation). Every LEL symbol belongs to one of four categories: Subject, Object, Verb, and State.

Antonelli [7] outlines a strategy to describe the LEL in a collaborative way. However, it is very difficult to produce a domain language specification when there are many actors involved [8]. In a collaborative context, all participants build a joint model, and as previously explained conflicts might emerge between the different viewpoints.

This paper presents an approach for the identification and resolution of conflicts that emerge when the LEL is developed collaboratively. The collaborative construction of the LEL means that different stakeholders propose symbols and provides definitions in an iterative way. This means that different people collaborate by making specific contributions: identifying the symbol that must be defined, or adding a definition. Nevertheless, in this context, it is necessary to have a full understanding of all the definitions. Our proposed approach consists in analyzing the whole glossary looking for conflicts and providing a solution for each conflict.

The paper is organized as follows: Section II provides the related work; Section III presents the conflicts, the proposed solutions, and a preliminary evaluation; finally, Section IV sets out the conclusions and future work.

II. RELATED WORK

Different authors have studied the existence of conflicts in Requirements Engineering [9]. Literature covers a wide range of conflict types and stages of the requirement phase where conflicts can appear [10]. Bendjenna [11] states the importance of dealing with conflictive situations during Requirements Engineering, considering the variety of stakeholders with the common objective of obtaining a unique system. Aldekhail [12] presents a literature review related to requirements conflicts. Some publications have presented requirements conflict management in a web-based collaborative environment. The SOP project [13] has developed a wiki using the Volere Requirements Specification Template [14], seeking to pinpoint inconsistencies in requirements documents created with their tool. WikiWinWin [15] is a wiki front-end to the WinWin tool. Urbietta [16] presents an approach for detecting and solving inconsistencies and conflicts in web software requirements and shows a taxonomy for conflicts in Web applications requirements. Lutz [2] developed CREW-Space, a tool to support the co-located collaboration of several users to simultaneously interact through Android-enabled mobile devices. They use role playing to involve different stakeholders in a use case analysis. Azadegan [3] proposes two steps: (i) identifying relevant user requirements and (ii) voting for user requirements.

The problem of conflicts also appears when building domain ontologies collaboratively. Lexons with properties, restrictions and relationships are defined in ontologies. In the LEL, there are symbols with two specific attributes (notion and behavioral responses), and relationships between the symbols are hyperlinks to other symbols used to make the description. Also each symbol has a type. The most important difference between ontologies and our approach is that we analyze these definitions, while approaches with ontologies mainly analyze

the relationship between the elements. It was analyzed if there is overlapping in definition of the notion or the behavioral responses, or even if they are similar. If definitions are similar it could imply that synonyms were found. It is important to pay attention to homonyms, which are the same symbol referring to different things. Symbols (concepts) are naturally organized in a hierarchy way. This approach also analyzes how definitions are organized or repeated in such structure. In collaborative ontology engineering there is a great variety of methodologies [17], nevertheless, they do not analyze the definitions. Chen [18] proposes an approach that deals with classes and relations. They detect three kinds of conflicts: hard, soft and latent conflicts between the classes. On the subject of building ontologies collaboratively some studies apply the consensus method [19] [20]. It has been proved to be useful in conflict solution between objects. The most important problem in consensus-based collaboration, is defining when they get an agreement. Consensus quality concept [21] is defined to show, how they get a consensus, in the construction of the Vietnamese language dictionary with WordNet.

III. CONFLICTS IN THE COLLABORATIVELY DEVELOPED LEL MODEL

This section presents the proposed conflict resolution approach and a preliminary validation. Section A describes the process to identify conflicts during the collaborative description of the LEL and presents a set of the conflicts that could arise. It is important to mention that these conflicts were identified from several real-life software systems descriptions. Section B shows each conflict and the actions to solve them. Finally, section C presents a preliminary evaluation.

A. Our Approach in a Nutshell

The LEL is built in an iterative and incremental way, where different Requirements Engineers contribute to its description. With different points of view a conflict may arise. Thus, it must be identified and solved as soon as possible in order to obtain a consistent LEL (see Figure 1).



Figure 1. Process for conflicts resolution (Req Eng.: Requirements Engineer.)

The first step represents the action that every Requirements Engineer performs: identifying a symbol or contributing with the description of notion or behavioral response. Every action



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

can give origin to a conflict. For example, two different Requirements Engineers could define independently the previous Label symbols of Table I and 2. Thus, the whole glossary must be analyzed in order to identify conflicts. If a conflict is identified, it should be solved in order to assure the consistency of the LEL.

The list of conflicts was defined from the analysis of several real projects. The conflicts are grouped in categories in order to make the description clearer.

The first category is *Semantic conflicts*: These are conflicts that arise when there are differences in the meaning of the symbols. For example, Label refers to two different actions: the action of putting the brand (Table I) and the action of marking the price (Table II). Subcategories of Semantic conflicts are: (i) the same identification for elements with different meaning and the same syntactic classification; (ii) different identification for elements that refers to the same concept in the same way; (iii) different identification for elements that refers to the same concept in different way; (iv) different identification for elements that refers to the same concept with complementary information.

The second category is *Structural conflicts*: Structural conflicts arise when there is complete or partial repetition in the definitions, considering the description of the behavioral responses or the organization of the description in hierarchies. For example, let's consider that one symbol is a generic concept, and there is a specific term that specializes the previous one, and the last symbol repeats information described in the first one. Subcategories are: (i) different level of detail; (ii) descriptions duplicated in hierarchies.

The last category is *Syntactic conflicts*: These conflicts appear when the same symbol has different syntactic classifications. For example, Label can be an Object or a Verb. There is no subcategory.

B. Catalogue of Conflicts and their Solutions

This section describes the conflicts with more detail together with their proposed resolution. In order to illustrate the proposed approach, we chose "IP Etiquetas S.A.", a company that produces some kinds of sticky labels, either with barcodes, with specific brands or white ones. Underlined words are other LEL symbols.

The study was developed by means of a series of interviews carried out by different Requirements Engineers with several people in the company. A series of conflicts arose during the attempt to define the LEL model collaboratively. The total number of conflicts found was 17. For space reasons we show some of them in detail. The other conflicts refer to behavioral response conflicts and also conflicts generated when part of a description of notion or behavioral response defined by a requirement engineer is contained on the defined by other requirement engineer. Some examples include the Label symbol, which was considered by a Requirements Engineer as the verb meaning "attach a label," whereas another Requirements Engineer considers that Label is the produced label. A third engineer thinks the Label symbol means "attach the price tag," this being also a verb.

1) The same identification for elements with different meaning and the same syntactic classification (Homonymy).

This conflict arises when there are two different entries that are identified with the same symbol, but they represent different things. For example, let's consider two different definitions of the symbol "Label" as described in Table I and Table II. The identification of both symbols is the same, since it is "Label". Nevertheless, both LEL entries refer to different things; one represents the action of putting the brand, while the other represents the action of marking the price.

TABLE I. LABEL SYMBOL

Symbol #: 10	Author: Req. Eng. 3	Type ¹ : Verb
Names ²	Label	
Notion	- It is the action of putting the brand of the <u>product</u> on the boxes of <u>finished product</u> .	
Behavioral Response	-The <u>logo of the brand</u> is defined with the client and is previously established.	

TABLE II. LABEL SYMBOL

Symbol #: 10	Author: Req. Eng. 1	Type ¹ : Verb
Names ²	Label	
Notion	- It is the action of marking the price of each finished box of <u>finished product</u> .	
Behavioral Response	-The <u>price per box</u> is previously established according to the total number required.	

Heuristic to detect the conflict: review all the LEL entries, identifying two or more entries with the same identification. Check the notion, in order to determine whether the entry is duplicated or they are different entries.

Solution: If the entry is duplicated merge both definitions. If the entries are different, specialize the identification in order to make clear that there are different entries: Label(1) and Label(2).

2) The same identification for elements with different syntactic classification (Homonymy).

This conflict is similar to the previous one, but the difference relies on the type of the entries. For example, let's consider a new symbol "Label" with Verb classification (Table III), while the other "Label" symbols refers to Objects (Table I). The "Label" of object category refers to the end product manufactured by the company.

TABLE III. LABEL SYMBOL

Symbol #: 11	Author: Req. Eng. 2	Type ¹ : Object
Names ²	Label	
Notion	- <u>Product</u> manufactured by the company	
Behavioral Response	-...	

Heuristic to detect the conflict: review all the LEL entries, identifying two or more entries with the same identification and different category.

Solution: Rename the symbols as Label(1) and Label(3).



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

3) *Different identification for elements that refer to the same concept in the same way (Synonym).*

This conflict arises when there are two different entries that are identified with different symbols, but they are described in the same way. For example, let's consider two different entries "missing stock" and "insufficient raw material" as described in Table IV and Table V. Both refer to the same situation described identically. That is, "State of raw material stock when it is lower than the minimum stock level."

TABLE IV. MISSING STOCK SYMBOL

Symbol #: 17	Author: Req. Eng. 2	Type ¹ : State
Names	Missing stock	
Notion	-State of <u>raw material</u> stock when it is lower than the <u>minimum stock level</u> .	
Behavioral Response	---	

TABLE V. INSUFFICIENT RAW MATERIAL SYMBOL

Symbol #: 9	Author: Req. Eng. 3	Type ¹ : State
Names	Insufficient raw material	
Notion	-State of <u>raw material</u> stock when it is lower than the <u>minimum stock level</u> .	
Behavioral Response	---	

Heuristic to detect the conflict: Compare all the notions of the different symbols checking for coincidences.

Solution: Define the elements as synonyms. In the example, "Missing Stock / Insufficient Raw Material element" must be defined as synonyms of the same entry.

4) *Different identification for elements that refer to the same concept in different way (Overlapping).*

This conflict arises when there are two different entries that are identified with different symbols, but they are described in different way. For example, let's consider two different entries "insufficient raw material" as described in Table V and Table VI. Both refer to the same situation described similarly. One symbol is described as "State of raw material stock when it is lower than the minimum stock level." while the other is described as "State of the stock of supplies when it must be changed to replenishment." Both symbols refer to the same concept, and both descriptions are similar.

TABLE VI. INSUFFICIENT RAW MATERIAL SYMBOL

Symbol #: 8	Author: Req. Eng. 1	Type ¹ : State
Names	Insufficient raw material	
Notion	-State of the stock of <u>supplies</u> when it must be changed to <u>replenishment</u> .	
Behavioral Response	---	

Heuristic to detect the conflict: Compare all the notions of the different symbols checking for similarities.

Solution: Since both descriptions are similar, it must be agreed only one description. The other entry must be removed. In Tables V and VI, the same symbol with a different Notion is shown.

5) *Different level of detail.*

This conflict arises when there are different symbols overlapping concepts in a hierarchy structure not well defined. Let's consider the situation of two different operators: (i) Rewinder Operator and (ii) Flexographic Printing Press Operator. One Requirements Engineer defines only one symbol named "Operator" with a general description considering both roles (i) and (ii). While other Requirements Engineer defines the two specific symbols (i) and (ii). In this situation, there are common characteristics to both roles; it should be described in a generic "operator" symbol, and then, the specific characteristics of both roles (i) and (ii) should be described in them.

TABLE VII. OPERATOR SYMBOL

Symbol #: 22	Author: Req. Eng. 3	Type ¹ : Subject
Names	Operator	
Notion	-It is the technician in charge of operating the <u>production machines</u> .	
Behavioral Response	---	

TABLE VIII. FLEXOGRAPHIC PRINTING PRESS OPERATOR SYMBOL

Symbol #: 9	Author: Req. Eng. 2	Type ¹ : Subject
Names	Flexographic printing press operator	
Notion	-It is the technician in charge of operating the <u>flexographic printing press</u> .	
Behavioral Response	---	

TABLE IX. REWINDER OPERATOR SYMBOL

Symbol #: 20	Author: Req. Eng. 2	Type ¹ : Subject
Names	Rewinder operator	
Notion	-It is the person in charge of rewinding the <u>label rolls</u> . -It is the technician in charge of operating the <u>rewinding machine</u> .	
Behavioral Response	---	

Heuristic to detect the conflict: Compare all the notions of the different symbols looking for possible hierarchy structures.

Solution: Identify the generic and specific terms of the hierarchy structure, and describe the specifics mentioning the generic. For example, in specializes symbols, refer to "Operator", saying that "He is an Operator that ..."

6) *Different identification for elements that refer to the same concept with complementary information (Synonym with complementary information).*

This conflict arises when there are two different entries that are identified with different symbols, and they are described

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



with complementary information. For example, let's consider two different entries "Cash Flow" and "Monetary Flow" as described in Table X and Table XI. Both refer to the same situation. In this case "Cash Flow" describes more details in Notion, defining it as "the amount of cash inflows and outflows" and that "it is originated by payments issued or received" while "Monetary Flow" is defined by "the amount of cash inflows and outflows". Moreover, this situation could be observed in Behavioral Response.

TABLE X. CASH FLOW SYMBOL

Symbol #: 3	Author: Req. Eng. 5	Type ¹ : Object
Names	Cash Flow	
Notion	-It is the amount of cash inflows and outflows. -It is originated by payments issued or received.	
Behavioral Response	-It is daily prepared by the Treasurer.	

TABLE XI. MONETARY FLOW SYMBOL

Symbol #: 13	Author: Req. Eng. 1	Type ¹ : Object
Names	Monetary flow	
Notion	-It is the amount of cash inflows and outflows. -It is approved and registered by Treasurer.	
Behavioral Response	-It is used as a source of information when preparing the Sales Forecast.	

Heuristic to detect the conflict: Compare all the notions and Behavioral Response looking for common descriptions in different symbols checking for coincidences and differences.

Solution: Define the elements as synonyms; merging all the descriptions, that is, the whole description must be used: the common part, and the particularities of each symbol. In the example, "Cash Flow / Monetary flow" must be defined as synonyms of the same entry with the richer description in each case.

7) Descriptions duplicated in in hierarchies

This conflict arises when descriptions are duplicated in specific elements of the hierarchy instead of putting them in the generic element. For example, two specific elements have the same description in the behavioral responses. Thus, the objective of the hierarchy is to put the common descriptions in the generic element. The same problem could arise in the notion.

TABLE XII. OPERATOR SYMBOL

Symbol #: 22	Author: Req. Eng. 1	Type ¹ : Subject
Names	Operator	
Notion	-It is the technician in charge of operating the <u>production machines</u> .	
Behavioral Response	- Send the finished order to the Plant Manager	

Let's consider the situation of two different operators: (i) Rewinder Operator and (ii) Flexographic Printing Press Operator. A requirements engineer has placed the same behavioral response on each specialized symbol and another

requirements engineer has defined a generic symbol, but the former did not realize that the generic symbol was the right place to put the description. The corresponding behavioral responses "Send the finished order to the Plant Manager" must be eliminated from each specialized, leaving this description only in the generic.

TABLE XIII. FLEXOGRAPHIC PRINTING PRESS OPERATOR SYMBOL

Symbol #: 9	Author: Req. Eng. 5	Type ¹ : Subject
Names	Flexographic printing press operator	
Notion	-It is the technician in charge of operating the <u>flexographic printing press</u> .	
Behavioral Response	- Send the finished order to the Plant Manager	

TABLE XIV. REWINDER OPERATOR SYMBOL

Symbol #: 21	Author: Req. Eng. 5	Type ¹ : Subject
Names	Rewinder operator	
Notion	-It is the person in charge of rewinding the <u>label roll</u> . -It is the technician in charge of operating the <u>rewinding machine</u> .	
Behavioral Response	- Send the finished order to the Plant Manager	

Heuristic to detect the conflict: Compare all the notions and Behavioral Response of the different symbols looking for repetitions in the specific elements.

Solution: Move the repeated description from the specific elements to the generic one.

C. Preliminary Evaluation

In order to validate the conflicts proposed in this paper, we analyzed a LEL built collaboratively by 5 Requirements Engineers. We analyze the resulting LEL looking for the conflicts we proposed. Then, we present every report to Requirements Engineers who participated in the construction of the LEL to check whether they agree with the conflicts reported. Requirements Engineers have agreed in almost all the conflict reported. The following Table XV presents some figures for the 5 different participants.

TABLE XV. TOTAL OF CONFLICTS FOUND IN IP ETIQUETAS

Req. Eng.	Total of symbols described	Symbols with conflicts	Percentage
Req. Eng. 1	42	31	74
Req. Eng. 2	35	28	80
Req. Eng. 3	28	21	75
Req. Eng. 4	31	27	87
Req. Eng. 5	47	38	80

Table XV presents for each Requirements Engineers the number of symbols in which he participated in their description, the symbol with conflict identified by our approach and the percentage that it represents. This table shows that conflicts are very common.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019

IV. CONCLUSIONS AND FURTHER WORK

Requirements definition is one of the initial stages in the software development process and their products are the groundwork for subsequent stages. Thus, errors made in requirements stage will be replicated and deepened in subsequent stages. For this reason, it is extremely important to develop requirements models of the highest quality as possible. When requirements models are developed collaboratively, conflicts unavoidable will arise. Moreover, natural language descriptions are more plausible to give origin to conflicts.

A vast experience in working with a structured glossary, the Language Extended Lexicon (LEL), proves that such structure reduces the occurrence of conflicts. However engineers have observed that while building the LEL collaboratively produces a richer model, it also introduces conflicts. In our research, and by analyzing several application domains of real projects, a classification of conflicts was devised. A process and guides for their resolution has been described in this paper. Our approach with some examples of a real project was also illustrated.

A preliminary evaluation was presented; it showed the importance of identifying conflicts and the solutions for the conflicts proposed. The percentage of conflicts was between 74% and 87%, in the five groups that have been evaluated. It shows the importance of solving those conflicts for arriving to better quality models.

An experiment to validate the conflicts and their resolutions is being designed. This experiment will be conducted in a different country to validate in another context the findings presented in this paper.

A process to identify the conflicts and an automated suggestion of solutions is planned. This implementation will be based on two important modules: (i) a module of natural language processing and (ii) a module of machine learning.

REFERENCES

- [1] B.W. Boehm, *Software Engineering Economics*. Prentice Hall, 1981.
- [2] R. Lutz, S. Schüler, and S. Diehl, "Using mobile devices for collaborative requirements engineering", 27th IEEE/ACM International Conference on Automated Software Engineering, pp. 298–301. ACM, 2012.
- [3] A. Azadegan, X. Cheng, F. Niederman, and G. Yin, "Collaborative requirements elicitation in facilitated collaboration: report from a case study", 40th Hawaii International Conference on System Sciences, pp. 569–578, ISSN 15301605, IEEE, 2013.
- [4] J. C. S. D. P. Leite, and A. P. M. Franco, "A strategy for conceptual model acquisition", *Requirements Engineering*, IEEE International Symposium on, pp. 243–246. IEEE, 1993.

- [5] A. d. P. A. Oliveira, J. C. S. d. P. Leite, L. M. Cysneiros and C. Cappelli, "Eliciting Multi-Agent Systems Intentionality: from Language Extended Lexicon to P Models", *Chilean Society of Computer Science, 2007. SCCC '07. XXVI International Conference of the, Iquique, 2007*, pp. 40–49, doi: 10.1109/SCCC.2007.20
- [6] S. Gost, "Transformation from LEL to UML", *International Journal of Computer Applications*, vol. 48, no. 12, 2012.
- [7] L. Antonelli, G. Rossi, and Oliviero A., "A collaborative approach to describe the domain language through the Language Extended Lexicon", *Journal of Object Technology*, vol. 15, no. 3, pp. 1–27, 2016.
- [8] N. Mulla, S. Girase S, "A new approach to requirement elicitation based on stakeholder recommendation and collaborative filtering", *International Journal of Software Engineering and Applications*, vol. 3(3), pp. 51–60, 2012, doi:10.5121/ijsea.2012.3305.
- [9] S. Easterbrook, "Resolving requirements conflicts with computer-supported negotiation", *Requirements engineering: social and technical issues*, vol. 1, pp. 41–65, 1994.
- [10] W. N. Robinson, S. D. Paulowski, and V. Volkov, *Requirements interaction management*. ACM Computer Survey, vol. 35(2), pp. 152–190, 2003.
- [11] H. Benajoua, P. J. Charrel, and N. E. Zassou, "Using AHP Method to Resolve Conflicts Between Non-Functional Concerns", *International Conference on Education, Applied Sciences and Management (ICEASIM'2012)*, Dubai, UAE, pp. 26–27, 2012.
- [12] M. Aldekhail, A. Chikh, and D. Ziari, "Software Requirements Conflict Identification: Review and Recommendations", *International Journal of advanced computer science and applications*, vol. 7, no. 10, pp. 326–335, 2016.
- [13] B. Decker, E.Ras, J. Koch, P. Jambert, and M.Bieth, "Wiki-based stakeholder participation in requirements engineering". *IEEE Software*, vol. 24(2), pp. 28–35, 2007.
- [14] J. Robertson, and S. Robertson, *Volere Requirements Specification Template*. The Atlantic Systems Guild, 2012.
- [15] D. Yang, D. Wu, S. Koolmanojwong, Brown, A. W., and B. W. Boehm, "Wikirequire: A wiki based system for collaborative requirements negotiation", *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, pp. 24–24. IEEE, 2008.
- [16] M. Urbina, M. J. Escalona, E. R. Luna, and G. Rossi, G., "Detecting conflicts and inconsistencies in web application requirements", *International Conference on Web Engineering*, pp. 278–288. Springer, Berlin, Heidelberg, 2011.
- [17] E. Simperl, and M. Luczak-Rösch, "Collaborative ontology engineering: a survey", *The Knowledge Engineering Review*, vol. 29, no. 1, pp. 101–131, 2014.
- [18] Y. Chen, X. Peng, and W. Zhao, "An approach to detect collaborative conflicts for ontology development". *Advances in Data and Web Management*, pp. 442–454. Springer, Berlin, Heidelberg, 2009.
- [19] S. Karapiperis, and D. Apostolou, D. "Consensus building in collaborative ontology engineering processes". *Journal of Universal Knowledge Management*, vol 1(3), pp. 199–216, 2006.
- [20] N. T. Nguyen, "Advanced methods for inconsistent knowledge management". Springer, London (2008)
- [21] T. H. Duong, M. Q. Tran, and T.P.T. Nguyen, "Collaborative Vietnamese WordNet building using consensus quality", *Vietnam J ComputSci 2017*, vol 4-85, Springer Berlin Heidelberg, Print ISSN: 2196-8888, Online ISSN: 2196-8896, 2017.

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



Knowledge Systems Institute

156 Park Square, Pittsburgh, PA 15238, USA
 Tel: 412-606-5022 Email: seke@ksiresearch.org

SEKE2018 Registration Receipt

2018/05/02

Claudia Litvak
 Universidad Nacional de La Matanza C.U.I.T: 30-64622868-5 Florencio Varela 1903 -San Justo-
 Bs.As.Argentina
 Argentina
claudialitvak@gmail.com

Dear Claudia Litvak:

Your paper, "Conflict Management in the Collaborative Description of a Domain Language " (ID 106), has been accepted by SEKE2018.

We have received your payment of the SEKE2018 Registration Fee. The total amount is \$945.

Your registration is now complete.

Sincerely yours,

Shi-Kuo Chang, Steering Committee Chair
 Software Engineering and Knowledge Engineering
 Conference Series



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

2) CSCI2018

Improving the identification of conflicts in collaborative requirements engineering

Claudia Litvak
DIIT
Universidad Nacional de La
Matanza
La Matanza, Bs. As., Argentina
clitvak@unlam.edu.ar

Leandro Antonelli
Lifla, Facultad de Informatica
Universidad Nacional de La
Plata
La Plata, Argentina
lanto@lifla.info.unlp.edu.ar

Gustavo Rossi
Lifla, Facultad de Informatica
Universidad Nacional de La
Plata
La Plata, Argentina
Also CONICET
gustavo@lifla.info.unlp.edu.ar

Nora Gigante
DIIT
Universidad Nacional de La
Matanza
La Matanza, Bs. As., Argentina
ngigante@unlam.edu.ar

Abstract—Requirements engineering has the aim of describing as accurately as possible the needs and expectations of all the stakeholders involved in the software development. The collaborative work of the stakeholders in this process allows them to improve the quality of the requirements. Nevertheless, collaborative work involves the raising of conflicts, and they must be solved in order to achieve the desired quality. This paper presents the evolution in our understanding of the process to identify and solve conflicts during the collaborative construction of the Language Extended Lexicon that captures the domain language. This process was validated with three different case studies and the usabilitySUS questionnaire.

Keywords— requirements engineering, collaboration, conflicts, natural language models

I. INTRODUCTION

Requirements Engineering involves the construction of models to reveal the knowledge obtained during this step. Those models are subsequently applied in oncoming stages of the software construction process.

There are standards, like IEEE 29148 [1], that deal with the products involved in Requirements Engineering throughout the life cycle of systems and software, seeking to improve the quality of requirements. However, these standards do not explicitly indicate how to do this.

Collaborative work results in richer, more complete and more consistent requirements [2]. Furthermore, the interaction of different stakeholders on the same problem improves the quality of system requirements [3].

The collaborative construction of domain language implies the existence of better quality descriptions. But, as in any collaborative work, appear conflicts that need to be solved in order to clarify those descriptions with fewer contradictions.

The model used in this study is the Language Extended Lexicon (LEL). The LEL is a model that uses natural language [4] to describe the vocabulary of the application domain. The LEL captures the terms and describes them with the symbol name, the notion, and the behavioral responses. The name identifies the symbol; all synonyms that exist in the domain must be defined accordingly. The notion describes the meaning (denotation) and the behavioral responses describe the relation of the symbol with other symbols (connotation). Every LEL

symbol falls into one of four categories: Subject, Object, Verb, or State.

Goel[5] states that “natural language (Language Extended Lexicon) is the only notation that can be read and understood by the stakeholders, hence encouraging their active participation which is crucial in first steps of software development” and also that “natural language (Language Extended Lexicon) is still widely used to model requirements information.” The LEL is a very convenient tool for stakeholders with no technical skills, although people with such skills will profit more from its use [6]. In particular, the convenience of using the LEL as a tool can be described as follows: it is easy to learn, it is easy to use and it has good expressiveness. There are several publications using the LEL in complex domains which validate these claims. Gil et al. [7] state that “building a LEL in an application completely unknown to the requirements engineer and with highly complex language can be considered a successful experience, since users stated that requirements engineers have developed a great knowledge about the application.” Cysneiros et al. [8] state that “the use of the LEL was very well accepted and understood by the stakeholders. As these stakeholders were non-technical experts from a specific and complex domain, the authors believe that the LEL can be suitable to be applied in many other domains.”

In this paper we present a process for finding and solving conflicts in the collaborative construction of a domain model. The process reflects the evolution of our research on this topic presented on [9].

Furthermore, we present the validation of that process, applied to three case studies. To do so, three different kinds of companies were selected, applying the process to the collaborative LEL construction. The first selected company was a factory that manufactures different kinds of roof membrane or other asphaltic products for roof protection. The second company is a travel assistance agency. The third one produces fabrics for the textile industry. In each case study three requirement engineers worked in the collaborative construction of the LEL for each domain. There were three different groups of three requirement engineers each one. There was a coordinator who also looked for conflicts at the end of each period. While defining those LEL collaboratively many conflicts appeared, which were solved before



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

continuing. Consequently, the requirement engineers remained working in the next period from a LEL with almost all the conflicts solved. This process improved the quality of the obtained model and it was verified in the definition of the collaborative LEL construction in the three cases. The requirements engineers acquired the knowledge of the possible conflicts in the LEL construction and this implied a reduction in the amount of conflicts during the following periods.

The paper is organized as follows: Section II provides the related work; Section III presents our approach; Section IV sets out the conclusions and future work.

II. RELATED WORK

Requirement Engineering is the first step in Software Development, in which a series of models based on previous ones is defined. Thus, the importance of the requirement models, that need to be delineated with the best quality possible. Defining the domain language before specifying the requirements is a way of coping with these problems [10].

Different stakeholders working collaboratively on the same problem would define better quality system requirement [3]. In Antonelli et al. [10] a strategy was outlined in order to capture the domain language in a collaborative way by means of LEL. In a collaborative context, all participants build a joint model, and as a result conflicts emerge from the different viewpoints. Different authors observed that it is very difficult to produce a domain language specification when there are many actors involved [11]. In any collaborative work, in the construction of the LEL some conflicts may arise, which need to be solved in order to have a LEL without contradictions.

Conflicts also appear in Requirements Engineering [12]. Literature covers a wide range of conflict types and stages of the requirements interaction lifecycle [13]. Some publications present the management of the requirements conflict in a web-based collaborative environment. The SOP project [14] has developed a wiki using the Volere Requirements Specification Template [15], seeking to pinpoint inconsistencies in requirements documents created with their tool. WikiWinWin [16] is a wiki front-end to the WinWintool. Urbieta et al. [17] presents an approach for detecting and solving inconsistencies and conflicts in web software requirements and shows a taxonomy for conflicts in Web applications requirements. Bendjenna [18] gives an explanation of the need to deal with conflictive situations during Requirements Engineering as a very important step, considering the variety of stakeholders and the common purpose of obtaining a seamless end system. Azadegan et al. [3] propose two steps: (i) identifying relevant user requirements and (ii) voting for user requirements. Aldekhail et al. presents a review of the literature related to requirements conflicts [19].

There are also conflicts in building domain ontologies collaboratively. In ontologies the lexons are defined together with their properties, restrictions and relationships. In LEL there is a definition of symbols, which includes the concepts of notion and behavioral responses, and also the symbol type (verb, subject, object or state). Fundamentally the contrast between our approach and ontologies is that we evaluate the definitions, while the research with ontologies mainly studies

the relationship between the elements. We analyzed if there is overlapping in the definition of the notion or in the behavioral responses, or even if they are similar. If the definitions are similar, it could suggest that synonyms were found. Furthermore, symbols are naturally organized in a hierarchy way. This approach also analyzes how definitions are organized or repeated in the LEL. In collaborative ontology engineering there is a great variety of methodologies [20], nevertheless, they do not analyze the definitions. DILIGENT [21] proposes a methodology for collaborative ontology engineering based on the IBIS argumentation model [22]. In [23] they describe the DOGMAMESS methodology and system for scalable, community-grounded ontology engineering. Scalable ontology engineering is hard to do in inter-organizational settings where there are many pre-existing organizational ontologies and rapidly changing collaborative requirements. A complex socio-technical process of ontology alignment and meaning negotiation is therefore required. The Human-Centered Ontology Engineering Methodology (HCOME) [24] is, like DILIGENT, a methodology that explicitly focuses on the distributed creation of ontologies in knowledge-intensive organizations. RapidOWL applies the agile engineering paradigm to ontology engineering [25]. The approach does not commit to a specific process model or an ontology life cycle, but aims at providing a number of guidelines to be taken into account by the engineering team. Chen in 2009 [26] proposed an approach to classify and detect three kinds of collaborative conflicts according to some mechanisms for large-scale collaborative ontology development. On the subject of building ontologies collaboratively some studies apply the consensus [27]. The consensus quality concepts and the consensus-building mechanism [28] allow a large group of people to contribute or annotate a common ontology in a collaborative way to reach consensus among individuals. They investigate a methodology for collaborative ontology building in which consensus quality and susceptible to consensus ideas were applied to realize when consensus was reached in the final version of the collaborative ontology building. In that paper the concepts of consensus quality and susceptible to consensus were defined specifically for the collaborative Vietnamese WordNet building.

III. OUR WORK

This section presents the evolution of our understanding of the process defined in [9] for finding and solving conflicts in the collaborative LEL construction. We validate the process usability with SUS in three case studies, performed by the requirements engineers involved in the LEL construction. It was done with workers of the software industry.

A. Process

This process consists of different stages. In the three initial stages, each requirement engineer identifies symbols with their descriptions, determines the synonyms, looks for conflicts and solves them. This is simultaneously carried out by different requirements engineers. Then, there are two stages in which a team work looks for conflicts and solutions, rewriting the symbols involved in those conflicts. There is also a coordinator



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

included as part of the team work, who helps to find conflicts and solutions. This process is delineated in an iterative and incremental way and continues until the quality is reached or the time is over. The main idea is obtaining a consistent LEL, involving different points of view of different stakeholders.

Loucopoulos [29] states that "The success of the requirements engineering process often depends on the ability to proceed from informal, fuzzy individual statements of requirements to a formal specification that is understood and agreed by all stakeholders." In our approach, we look for the agreement with all the stakeholders in our approach that relies on collaborative work.

The process is depicted in Fig. 1. The process presents two different parts: First, the parallel work where each requirement engineer identifies and describes the symbols, identifies conflicts and solves them, while others work on the same task. In the description of symbols they define all the synonyms, the notion and the behavioral response. Each requirement engineer analyzes and tries to understand the specific domain. They find conflicts with other symbols defined beforehand, when they were writing those symbols. As conflicts appear with both, the definitions made by other requirements engineers and/or with their own definitions they try to solve them. In second place, there is a moment for working as Team Work. A coordinator collaborates in looking for conflicts and all of them share this part of the process. They rewrite all the symbols that have been part of each conflict in order to have the minimum conflicts possible.

Loucopoulos [29] proposes a framework for the Requirements Engineering Process. In that process he defines three different stages: Elicitation, Specification and Validation. Elicitation and Specification are defined in our process in the parallel work, because the description of the LEL symbols is based on the Requirements Engineers' knowledge. Specification and Validation appear in our model in the Team Work, because the Requirements Engineers examine the LEL and adjust the symbols.

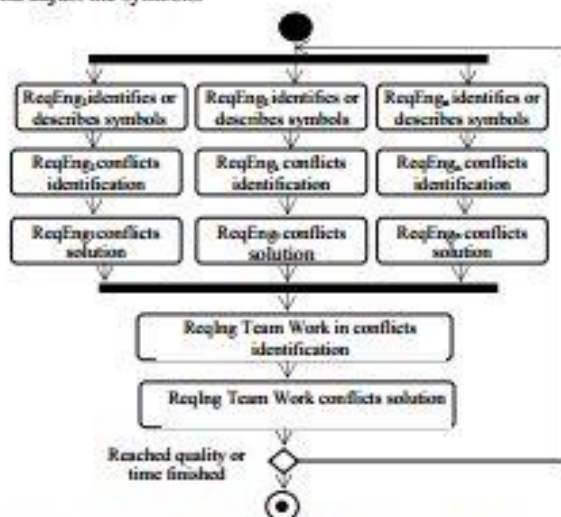


Fig. 1. Process for conflicts resolution (Req Eng: Requirements Engineer i)

TABLE I. TRAVELER SYMBOL

Symbol #: 6	Author: Req-Eng. 2	Type: Subject
Name/s	Traveler	
Notion	- Person who hires a product or package. - Individual to whom Assistcard provides traveler assistance services.	
Behavioral Response	- The traveler contracts a package. - The traveler purchases a product.	

TABLE II. TRAVELER SYMBOL

Symbol #: 3	Author: Req-Eng. 1	Type: Subject
Name/s	Passenger	
Notion	- Person who hires a product or package. - Individual to whom Assistcard provides traveler assistance services.	
Behavioral Response	- Contracts the package. - Purchases the product.	

TABLE III. TRAVELER/PASSENGER SYMBOL.

Symbol #: 6	Author: Req-Eng. 2, 1	Type: Subject
Name/s	Traveler/ Passenger	
Notion	- Person who hires a product or package. - Individual to whom Assistcard provides traveler assistance services.	
Behavioral Response	- Contracts the package. - Purchases the product.	

Each stage of Figure 1 will be described in detail showing specific examples from the LEL of the company Assistcard, a travel assistance company.

The stages are:

- Symbols identification and/or description: Requirements Engineers detect the words or expressions that are proper to that domain and thus deserve to be described in order to capture the specific language. An example is given in Table I: Requirement Engineer 2 defines the Subject Traveler, writing down its Notion and its Behavioral Response.
- Requirements Engineers conflicts identification: They review the catalog of conflicts presented in [9] and, keeping in mind that catalog, they evaluate whether these conflicts occur in the LEL being described. An example is given in Table II: Requirement Engineer 2 observes that there is another Subject called Passenger with the same Notion and Behavioral Response. Traveler and Passenger are synonyms.
- Requirements Engineers conflicts solution: They solve the conflicts observed in the previous stage, based on the catalog proposed in [9]. In Table III we give an example in which Traveler and Passenger are merged in only one symbol with both names.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

TABLE IV. INDIVIDUAL/INSURED SYMBOL.

Symbol #: 6	Author: Req-Eng. 3	Type: Subject
Names	Individual/ Insured	
Notion	- Person who hires a product or package. - Individual to whom Assisctard provides traveler assistance services.	
Behavioral Response	- Contracts the package. - Purchases the product. - Purchases the product through a seller by phone. - Purchases the product through a seller at an airport stall. - Purchases the product through a seller in a boutique. - Purchases the product through the Web Page. - Registers personal data in the Web Page if not registered yet.	

TABLE V. TRAVELER/PASSENGER/INDIVIDUAL/INSURED SYMBOL.

Symbol #: 6	Author: Req-Eng. 2,1,3	Type: Subject
Names	Traveler/ Passenger/ Individual/ Insured	
Notion	- Person who hires a product or package. - Individual to whom Assisctard provides traveler assistance services.	
Behavioral Response	- Contracts the package. - Purchases the product. - Purchases the product through a seller by phone. - Purchases the product through a seller at an airport stall. - Purchases the product through a seller in a boutique. - Purchases the product through the Web Page. - Registers personal data in the Web Page if not registered yet.	

- Requirements Engineers Team Work in conflict identification: In this stage the coordinator helps to identify conflicts that were not previously observed by all Requirements Engineers. The Team Work observes that there is another symbol defined by Requirement Engineer 3, called Individual/ Insured with the same Notion, and a broader description in the Behavioral Response. It can be seen in Table IV.
- Requirements Engineers Team Work in conflicts solution: They solve the conflicts observed in the previous stage, based on the catalog proposed in [9]. In this example the Team merges the symbols Traveler/ Passenger/ Individual/ Insured as a unique symbol considering the broader descriptions. This can be clearly seen in Table V.

B. Validation

The process defined for generating the LEL collaboratively was applied in three different case studies. That process was validated with the System Usability Scale (SUS) [30], which questionnaire was answered by each requirement engineer that created the LEL collaboratively in each case study.

1) Case studies

We selected three companies to which the participants had access in order to make some interviews with the stakeholders. The participants also got the information of the company from its web page. There were three requirements engineers assigned for each company. The main idea was to define the LEL collaboratively and solve the conflicts they found while working. To do so, they had to identify symbols, describe them (with notion and behavioral response) and look for conflicts. A coordinator was included in the Team Work in order to assist in the search and solution of conflicts before each day finished, before the process continued. This process was done for a week. Some days the requirement engineers had different times for working on the LEL definition, and sometimes they worked at the same time. At the end of each day the coordinator made a revision of the LEL: LEL symbols found, kind of symbols found, total of symbols defined until that day, conflicts solved each day. The revision also checked if there were symbols of the four categories.

2) Participants

All the participants involved in the case studies had an experience of more than 3 years in software development industries. Most of them were students at La Matanza University, in Buenos Aires, Argentina. Some participants were assistant researchers. In all, there was 1 woman and 8 men in the three groups; the ages varied from 22 to 30 years old; the coordinator was also a woman.

3) Real companies involved in this work

Three real companies involving different activities were selected to define the LEL collaboratively in those domains. All the companies were located in Argentina; two of them in Buenos Aires and the other in Corrientes. The first company is called Megaflex, a factory that manufactures different kinds of roof membrane and other asphaltic products for roof protection. They also offer complete systems for the solution of humidity roof problems. For this case study there was a continuous contact with the former owner of Megaflex, who was interviewed by the Requirement Engineers for defining the LEL. Employees in other areas were also interviewed: secretaries and workers in the production line. The second company selected was Assisctard Argentina. The participants had contacts with different stakeholders of this travel assistance company. They sell any kind of travel assistance, individual and group ones, and different kinds of insurance, the ordinary ones and the high risk insurance. They give travel assistance to their customers through travel agencies, at airports or on the Internet. The third one is a company that manufactures fabrics for the textile industry. This company is called Iberá and is part of a bigger group called Norfabril. Iberá sells the fabrics to Norfabril at special prices and also sell their product to other clients.

4) System Usability Scale (SUS)

The process was validated with the System Usability Scale (SUS) [30]. The System Usability Scale (SUS) provides a reliable tool for measuring the usability. It consists of a 10-item questionnaire with five response options where "1" is "Strongly Disagree" and "5" is "Strongly Agree". Originally created by John Brooke in 1986 [31], it allows you to evaluate



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

a wide variety of products and services, including hardware, software, mobile devices, websites and applications.

In our case, when a SUS is used, participants are asked to score 10 items, adapted from the original, (as shown here) for evaluating the usability of the process defined in Fig. 1.

1. I think that I would like to use the LEL frequently.
2. I found the LEL unnecessarily complex.
3. I thought the LEL was easy to use.
4. I think that I would need the support of a technician to be able to use the LEL.
5. I found the LEL is a highly cohesive tool.
6. I thought there was too much inconsistency in the LEL.
7. I can imagine that most people would learn to use the LEL very quickly.
8. I found the LEL very cumbersome to use.
9. I felt very confident using the LEL.
10. I needed to learn a lot of things before I could get going with the LEL.

Brooke states in [31]: "To calculate the SUS score, first sum the score contributions from each item. Each item's score contribution will range from 0 to 4. For items 1, 3, 5, 7, and 9 the score contribution is the scale position minus 1. For items 2, 4, 6, 8 and 10, the contribution is 5 minus the scale position." Interpreting scoring can be complex. The participant's scores for each question are converted to a new number, added together and then multiplied by 2.5 to convert the original scores of 0-40 to 0-100. Though the scores are 0-100, these are not percentages and should be considered only in terms of their percentile ranking. Based on research, a SUS score above 68 would be considered above average and anything below 68 is below average.

In 2008 Bangor [32] introduced an average of the SUS scores: by quartile ranges, by adjective ratings, and by acceptability of the overall SUS score. The SUS questions were complemented with one additional question asking about user-friendliness. This one was rated using a 6-point scale of adjective ratings: 1) Worst imaginable; 2) Poor; 3) OK; 4) Good; 5) Excellent; 6) Best imaginable.

In 2012 McLellan et al. [33] presented the SUS scores in other values: "Non acceptable", 0-64; "Acceptable", 65-84 and "Excellent", 85-100.

In 2013 Brooke [34] arrived to new conclusions about SUS, 25 years later from his original definition, in which he adapted Bangor's ideas. Brooke [34] also stated that the use of SUS had increased exponentially by 2013, being cited in more than 1200 publications.

5) Results

As usual, both, the mean and standard deviations of the users' responses are considered in the evaluation of the SUS ranking. Table VI shows the answers. The SUS average score was 91. This value is above both limits defined in literature 68 [31] and 64 [33]. So, the usability of our process can be considered as "Excellent" [33].

TABLE VI. SUS SCORING TEMPLATE.

	RE 1	RE 2	RE 3	RE 4	RE 5	RE 6	RE 7	RE 8	RE 9
Q1	5	4	5	4	5	5	4	5	4
Q2	2	1	1	1	1	1	1	1	1
Q3	4	5	5	5	5	5	5	5	5
Q4	1	1	1	2	2	1	1	1	1
Q5	5	3	4	5	4	4	3	4	4
Q6	2	2	2	2	2	2	2	1	2
Q7	4	5	5	4	4	5	5	5	4
Q8	1	1	1	1	1	1	1	1	1
Q9	5	4	5	5	5	5	5	5	5
Q10	2	1	1	1	1	2	1	1	1
Scores	88	88	95	90	90	93	90	98	90
Average	91								
Std. Dev.	3.4								

* RE is Requirement Engineer

It is also important to note that the standard deviation is 3.4. This low value indicates that the different measures are close to the mean and reinforces the good results.

The question of user-friendliness was answered with the options "Excellent" or "Best imaginable." These answers confirm the good result of the process usability.

IV. CONCLUSIONS AND FURTHER WORK

Requirement Engineering is one of the preliminary steps in the Software development life cycle in which all stakeholders are naturally involved. Natural language appears as an excellent option to share the knowledge with and between the stakeholders. Thus, the Language Extended Lexicon (LEL), written in natural language, is a good help to define the vocabulary of the application domain. The collaborative construction of the LEL makes it possible to define a richer model but implies the appearance of conflicts.

This paper presents the evolution of our understanding of the process defined in [9] for finding and solving conflicts in the collaborative LEL construction. The process was applied to three different case studies, with different domains and teams. After performing the case study where the LEL was collaboratively built, the participants were asked to answer a SUS questionnaire. The SUS score was 91, higher than the limits defined by different authors when considering the usability of the process as "acceptable" (64 and 68). That rank was reinforced by the very small standard deviation of 3.4. According to McLellan [33], this score ranks the process as "Excellent."

In order to continue with this research, an experiment is being done to validate the catalog of conflicts and their solutions. This experiment is being done in Nigeria with professionals with many years of experience in the computer science field. Furthermore, an application is being developed to help automate the process. This application relies on two technologies: natural language processing and machine learning.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

REFERENCES

- [1] IEEE 29148- 2011. *Systems and software engineering - Life cycle processes - Requirements engineering*. IEEE, Nueva York. (2011).
- [2] J. Konate, A. Sahraoui, and G. Kollfichoten, "Collaborative requirements elicitation: A processcentred approach. Group Decision and Negotiation", 23(4):847-877, 2014. URL: <https://doi.org/10.1007/s10726-013-9350-x>, doi:10.1007/s10726-013-9350-x.
- [3] A. Anadegan, X. Cheng, F. Naderman, and G. Yin, "Collaborative requirements elicitation in facilitated collaboration: report from a case study", 46th Hawaii International Conference on System Sciences, pp. 569-578, ISSN 15301605, IEEE, 2013.
- [4] J. C. S. D. P. Leite, and A. P. M. Franco, "A strategy for conceptual model acquisition", *Requirements Engineering, IEEE International Symposium on*, pp. 243-246. IEEE, 1993.
- [5] S. Goel, "Transformation from LEL to UML", *International Journal of Computer Applications*, vol. 48, no. 12, 2012.
- [6] A. d. P. A. Oliveira, J. C. S. d. P. Leite, L. M. Cysneiros and C. Cappelli, "Eliciting Multi-Agent Systems Intentionality: from Language Extended Lexicon to i* Models", *Chilean Society of Computer Science, 2007. SCCS '07. XXVI International Conference of the*, Iquique, 2007, pp. 40-49, doi: 10.1109/SCCS.2007.20
- [7] G. D. Gil, D. Arias Figueras, and A. Oliveros, "Producción del lel en un dominio técnico. informe de un caso". In WER, pages 53-69, 2000. URL: <http://dblp.uni-trier.de/db/conf/wer/wer2000.html#GilFO00>.
- [8] L. Cysneiros and J. Leite, "Using the language extended lexicon to support non-functional requirements elicitation", in WER, pages 139-153, 2001. URL: <http://dblp.uni-trier.de/db/conf/wer/wer2001.html#CysneirosL01>.
- [9] C. Litvak, L. Antonelli and G. Rossi, "Conflict Management in the Collaborative Description of a Domain Language", *Thirtieth International Conference on Software Engineering and Knowledge Engineering (SEKE 2018)*, San Francisco Bay, USA, doi: 10.18293/SEKE2018-106, 2018.
- [10] L. Antonelli, G. Rossi, and Oliveros A., "A collaborative approach to describe the domain language through the Language Extended Lexicon", *Journal of Object Technology*, vol. 15, no. 3, pp. 1-27, 2016.
- [11] S. L. Lim and A. Finkelstein. Stakecare, "Using social networks and collaborative filtering for large-scale requirements elicitation", *IEEE Trans. Software Eng.*, 38(3):707-735, 2012. URL: <http://dblp.uni-trier.de/db/journals/tse/tse38.html#LimF12>, doi:10.1109/TSE.2011.36.
- [12] S. Easterbrook, "Resolving requirements conflicts with computer-supported negotiation", *Requirements engineering: social and technical issues*, vol. 1, pp. 41-65, 1994.
- [13] W. N. Robinson, S. D. Pawlowski, and V. Volkov, *Requirements interaction management*. ACM Computer Survey, vol. 35(2), pp. 132-190, 2003.
- [14] B. Decker, E. Ras, J. Rech, P. Jaubert, and M. Rieth, "Wiki-based stakeholder participation in requirements engineering". *IEEE Software*, vol. 24(2), pp. 28-35, 2007.
- [15] J. Robertson, and S. Robertson, "Volere Requirements Specification Template". *The Atlantic Systems Guild*, 2012.
- [16] D. Yang, D. Wu, S. Koohmansojwong, Brown, A. W., and B. W. Boehm, "Wikiwinwin: A wiki based system for collaborative requirements negotiation", *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, pp. 24-24. IEEE, 2008.
- [17] M. Urbica, M. J. Escalona, E. R. Lara, and G. Rossi, G., "Detecting conflicts and inconsistencies in web application requirements", *International Conference on Web Engineering*, pp. 278-288. Springer, Berlin, Heidelberg, 2011.
- [18] H. Bendjennu, P. J. Charrel, and N. E. Zarour, "Using AHP Method to Resolve Conflicts Between Non-Functional Concerns", *International Conference on Education, Applied Sciences and Management (ICEASM'2012)*, Dubai, UAE, pp. 26-27, 2012.
- [19] M. Aldekhail, A. Chikh, and D. Zairi, "Software Requirements Conflict Identification: Review and Recommendations", *International Journal of advanced computer science and applications*, vol. 7, no. 10, pp. 326-335, 2016.
- [20] E. Simperl, and M. Luczak-Rösch, "Collaborative ontology engineering: a survey", *The Knowledge Engineering Review*, vol. 29, no. 1, pp. 101-131, 2014.
- [21] D. Vradovic, "The DILIGENT knowledge processes", *Journal of Knowledge Management* 2005, vol. 9, no 5, p. 85-96, 2005
- [22] Kunz, W., Rittel, H. W., "Issues as elements of information systems" (Vol. 131). Berkeley, California: Institute of Urban and Regional Development, University of California, 1970.
- [23] A. De Moor, P. De Leenheer, and R. Meersman, "DOGMA-MESS: A meaning evolution support system for interorganizational ontology engineering." In *International Conference on Conceptual Structures* (pp. 189-202). Springer, Berlin, Heidelberg, 2006. July.
- [24] K. Kotis, G. Vouros, "Human-centered ontology engineering: The HCOME methodology", *Knowledge and Information Systems*, vol. 10, no 1, p. 109-131, 2006
- [25] S. Auer, and H. Herr, "RapidOWL—An agile knowledge engineering methodology". In *International Andrei Ershov Memorial Conference on Perspectives of System Informatics*. Springer, Berlin, Heidelberg, p. 424-430, 2006.
- [26] Y. Chen, X. Peng, and W. Zhao, "An approach to detect collaborative conflicts for ontology development", *Advances in Data and Web Management*, pp. 442-454. Springer, Berlin, Heidelberg, 2009.
- [27] N. T. Nguyen, "Advanced methods for inconsistent knowledge management". Springer, London (2008).
- [28] T. H. Duong, M. Q. Tran, and T.P.T. Nguyen, "Collaborative Vietnamese WordNet building using consensus quality", *Vietnam J ComputSci* 2017, vol 4:85, Springer Berlin Heidelberg, Print ISSN: 2196-8888, Online ISSN: 2196-8896, 2017.
- [29] P. Loucopoulos, and V. Karakostas, "System requirements engineering", McGraw-Hill, Inc., 1995.
- [30] *System Usability Scale*, <https://www.usability.gov/over-to-and-from/methods/system-usability-scale.html>, revised on 10/10/18.
- [31] J. Brooke, "SUS-A quick and dirty usability scale. Usability evaluation in industry", 189(194), 4-7, 1996.
- [32] A. Bangor, P. T. Kortum, and J. T. Miller, "An empirical evaluation of the system usability scale." *Intl. Journal of Human-Computer Interaction* 24.6, pp. 574-594, 2008.
- [33] S. McLellan, A. Muddimer, and S. C. Peres, "The effect of experience on System Usability Scale ratings." *Journal of usability studies* 7.2, pp. 56-67, 2012.
- [34] J. Brooke, "SUS: a retrospective." *Journal of usability studies* 8.2, pp.29-40, 2013.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

Administered by:

UCMSS

14215 Via Michelangelo.

San Diego, CA 92129

Tel. 858-602-8266

CSCI
2018
Dec 13-15
Las Vegas, Nevada, USA

5th Annual Conference on
Computational Science &
Computational Intelligence

November 18, 2018

Prof. Claudia Litvak
Bogota 4139
-1407
CABA, Argentina

Dear Prof. Claudia Litvak,

This notification letter is to inform you that the paper entitled "Improving the Identification of Conflicts in Collaborative Requirements Engineering" which was submitted to the 5th Annual Conference on Computational Science and Computational Intelligence (CSCI'18) has been accepted for publication.

You have been invited by the members of the Organizing Committee of the CSCI'18 to attend its 5th annual conference to be held in Las Vegas, Nevada, USA (December 13-15, 2018). Universal Conference Management Systems & Support (UCMSS), who is managing the operation of these conferences, wishes to congratulate you on behalf of the Conference Committee and extend this invitation to you for oral presentation of the above paper at this event.

Enclosed, please find a letter addressed to the United States Consulate General for US VISA purposes. We hope that you will take advantage of this exceptional opportunity to join us. We are confident that you will enjoy the scientific program that is being offered at this year's event.

Congratulations once again and thank you for your contribution to the conference. We look forward to welcoming you at the conference in Las Vegas.

Sincerely,

*Universal Conference Management
Systems & Support*

Official Seal

Kaveh D. Arbtan
UCMSS
14214 Via Michelangelo
San Diego, CA 92129
U.S.A.

Official Contact: cs@american-cse.org | Web Address: americancse.org/events/csci2018



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019

3) CONAISI 2019

Resolución de conflictos en modelos de dominio en lenguaje natural: casos de estudio

Claudia Litvak¹, Leandro Antonelli², Gustavo Rossi^{2,3}, Ishaya Peni Gambo⁴, Facundo Velazquez Santillán¹, Diego Lo Giudice¹, Ximena Milla¹, Nora Gigante¹

¹ DIIT, Universidad Nacional de La Matanza, Argentina

² Lifa, Facultad de Informática, Universidad Nacional de La Plata, La Plata, Argentina

³ CONICET

⁴ Computer Science and Engineering, Faculty of Technology, Obafemi Awolowo University, Ile-Ife, Nigeria

clitvak@unlam.edu.ar, (gustavo.rossi)@lifa.info.unlp.edu.ar, ipgambo@gmail.com
(facundovelsantillan, dlogudice, ximilla, ngigante)@unlam.edu.ar

Resumen

La Ingeniería de Requisitos, una de las etapas iniciales en la Ingeniería de Software, tiene como objetivo describir lo más fielmente posible, las necesidades y expectativas de los clientes y usuarios. A tal fin, los ingenieros de requisitos desarrollan modelos, intentando obtener modelos de alta calidad. El trabajo de varios ingenieros de requisitos de modo colaborativo, ha surgido como una buena posibilidad para mejorar la calidad de los modelos. Aunque como en todo trabajo colaborativo surgen conflictos, inherentes a toda interacción humana. Dichos conflictos deben ser resueltos a fin de mejorar la calidad de los modelos. Los autores de este artículo han presentado un trabajo previo que propone un proceso para resolver los conflictos, una clasificación de conflictos y un catálogo. En este artículo se presentan tres casos de estudio que muestran la efectividad del proceso. Al resolver los conflictos, se eliminan ambigüedades e inconsistencias y se obtiene una disminución del tamaño del modelo.

1. Introducción

En la Ingeniería de Requisitos, una de las etapas iniciales del desarrollo de un sistema de software, los ingenieros de requisitos desarrollan modelos, que plasman el conocimiento adquirido de los diferentes stakeholders del software a construir.

Si bien existen estándares que presentan la propuesta de obtener modelos de alta calidad, como el estándar

IEEE 29148 [1], dichos estándares no aplican, como lograrlo. Por ello la propuesta es generar modelos colaborativamente, con distintos ingenieros de requisitos trabajando sobre el mismo dominio. Pero todo trabajo colaborativo implica la aparición de conflictos, que deben ser hallados y resueltos para obtener modelos de calidad. En el artículo Litvak, de 2018 [2], se ha presentado un proceso para resolución de conflictos, un catálogo de conflictos y las soluciones para resolverlos. Litvak se presentó una clasificación de los conflictos hallados. El estudio desarrollado se ha enfocado en el modelo Léxico Extendido del Lenguaje (LEL), modelo desarrollado por Litvak et al. [3] [4] [5].

El modelo LEL es un glosario, que describe el vocabulario del dominio de la aplicación, el cual es un medio para facilitar la comunicación entre todos los stakeholders. Este glosario es utilizado no sólo como modelo inicial en el proceso de Ingeniería de Requisitos, sino que resulta ser un complemento muy útil para etapas posteriores del desarrollo de software. En el LEL se definen todos los términos del dominio con todos los sinónimos utilizados por los stakeholders. Si hay varios sinónimos, se debe colocar en primera instancia el término más utilizado, y luego los demás sinónimos hasta llegar al menos utilizado. Para cada entrada del LEL se definen además dos conceptos, su noción y su impacto. La noción es la denotación del símbolo, es decir que explica qué es el símbolo en el dominio de la aplicación. Cuando hablamos de impacto, nos referimos a las respuestas de comportamiento planteadas por Litvak, en [3]. Por lo tanto, el impacto describe la connotación, es decir cómo el símbolo actúa en el dominio de la aplicación. Cada símbolo del glosario es clasificado como sujeto, objeto, verbo o estado.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019

Se presenta a continuación un ejemplo de un símbolo denominado Producto, de la empresa *Megaflex*. (Ver Tabla 1). En todos los símbolos del L.EI. las palabras o frases subrayadas son otros símbolos del L.EI.

Tabla 1. Símbolo Producto

Símbolo # 7	Autor: Ing. <i>Rog. 3</i>	Tipo: Objeto
Nombre/s	Producto	
NoCIÓN	-Es todo producto de impermeabilización fabricado por <i>Megaflex</i> .	
Impacto	-Pueden ser <i>membranas, transómbles y no transómbles, productos de terminación, membranas líquidas, y sistemas completos.</i>	

Como el trabajo se propone colaborativo, con la participación de varios ingenieros de requisitos trabajando sobre el mismo dominio, se ha agregado en la definición de cada símbolo el número de ingeniero de requisitos que ha definido dicho símbolo.

Se han desarrollado tres nuevos casos de estudio, en diferentes tipos de empresas, en distintos dominios y con grupos de ingenieros de requisitos diferentes, aplicando el proceso definido [2]. Los ingenieros de requisitos de cada caso de estudio han desarrollado el modelo L.EI. colaborativamente y estudiado luego la cantidad de símbolos involucrados en conflictos y la cantidad de símbolos finales luego de resolver los conflictos.

En la sección 2 se presentan los trabajos relacionados, en la sección 3 se presenta la planificación de los casos de estudio, en la sección 4 se presenta la ejecución de los casos de estudio, en la sección 5 se presenta el análisis de resultados y en la sección 6 se presentan conclusiones y trabajo futuro.

2. Trabajos Relacionados

Cuando un grupo de personas trabaja colaborativamente surgen conflictos, cualquiera sea el área bajo estudio. En particular en el área de la Ingeniería de Requisitos también aparecen conflictos al realizar trabajo colaborativo. Las personas suelen estar en desacuerdo sobre cómo interpretar las características de un dominio de aplicación, averiguar cuáles serán los requisitos para un nuevo sistema, y cómo cumplir con esos requisitos [6]. La literatura cubre una amplia gama de tipos de conflictos y etapas en fase de requisitos, donde pueden aparecer los conflictos [7].

Careron [8] definió el conflicto de requisitos como "la interacción contradictoria o inesperada entre los requisitos que tiene un efecto negativo sobre los resultados". Estos conflictos surgen cuando hay inconsistencia en los requisitos de dos o más interesados. En el trabajo de Robinson et al. [9], se observó que

cualquier situación en la que dos partes de una especificación de requisitos no obedecen a alguna relación que debe mantenerse entre ellos, implica que surgirá el conflicto. Por lo tanto, los conflictos de requisitos son interacciones y dependencias entre requisitos que pueden dar lugar a un funcionamiento no deseado del sistema.

En el trabajo presentado en 2012, *Benedictina* [10] señala la importancia de enfrentar situaciones conflictivas durante la ingeniería de requisitos, considerando la variedad de partes interesadas que tienen como objetivo común obtener un sistema único. En la mayoría de la literatura existente, los conflictos suelen resolverse por medio de la negociación, que es una actividad humana peculiar [11] [12].

En 2016 *Abdelhalim* [13] presenta una revisión de la literatura relacionada con los conflictos de requisitos. Algunas publicaciones han presentado la gestión de conflictos de requisitos en un entorno de colaboración basado en web. En 2007 el proyecto SOP [14] ha desarrollado una wiki utilizando la Plantilla de especificación de requisitos de *Yakov* [15], buscando identificar incoherencias en los documentos de requisitos creados con su herramienta. *WikiWinWin* [16] es una wiki *front-end* para la herramienta *WinWin Urbana* [17] presentó en 2011 un enfoque para detectar y resolver inconsistencias y conflictos en los requisitos de software de aplicaciones web y muestra una taxonomía para los conflictos en los requisitos de las aplicaciones web. En 2012 *Lee* [18] presentó *CREW-Space*, una herramienta para respaldar la colaboración conjunta de varios usuarios, para interactuar simultáneamente a través de dispositivos móviles con Android. Utilizan el juego de roles para involucrar a diferentes partes interesadas en un análisis de casos de uso. En 2013 *Abdelaziz* [19] propone dos pasos: (i) identificar los requisitos relevantes del usuario y (ii) votar por los requisitos del usuario.

El problema de los conflictos también aparece cuando se crean ontologías de dominios en forma colaborativa. En *la ontología* se definen los lexemas con sus propiedades, restricciones y relaciones. En el L.EI., hay símbolos con dos atributos específicos (noción y respuestas de comportamiento), y las relaciones entre los símbolos son hipervínculos a otros símbolos utilizados para hacer la descripción. *Además* cada símbolo tiene un tipo. La diferencia más importante entre las ontologías y nuestro enfoque es que analizamos esas definiciones, mientras que los enfoques con ontologías analizan principalmente la relación entre los elementos. *Además* se analizó si existe una superposición en la definición de la noción o en los impactos, o incluso si son similares. Si las definiciones son similares, podría implicar que se encontraron sinónimos. Es importante prestar atención a los homónimos, que son el mismo símbolo refiriéndose a diferentes cosas. Los símbolos (conceptos) se organizan



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

naturalmente de forma jerárquica. Este enfoque también analiza cómo se organizan o repiten las definiciones en jerarquías. En la ingeniería ontológica colaborativa existe una gran variedad de metodologías [20], sin embargo, no analizan las definiciones. En 2009 [Chen](#) [21] propone un enfoque que trata con las clases y las relaciones. Detectan tres tipos de conflictos: conflictos duros, suaves y latentes entre las clases. Sobre el tema de la construcción colaborativa de ontologías, algunos estudios aplican el método del consenso [22] [23]. Se ha demostrado que es útil en la solución de conflictos entre objetos. El problema más importante en la colaboración basada en el consenso es definir cuándo llegan a un acuerdo. El concepto de calidad de consenso se presentó en 2017 [24] para mostrar, cómo obtienen un consenso, en la construcción del diccionario de idioma vietnamita con [WordNet](#).

3. Planificación de los casos de estudio

El estudio realizado en este trabajo consistió de aplicar el proceso presentado por [Lizak](#), en 2018 [2]. Se utilizó el catálogo de conflictos definido y las soluciones a dichos conflictos, aplicando la clasificación de los conflictos establecida [2].

Se planificó realizar tres nuevos casos de estudio. En cada caso de estudio se planteó crear un LEL de modo colaborativo, aplicando el proceso definido. Se seleccionaron empresas diferentes, con ingenieros de requisitos diferentes, a fin de no sesgar la aplicación del proceso. Se planeó aplicar dicho proceso durante 7 días, considerando asignar 3 ingenieros de requisitos a cada caso de estudio. Se planificó que al finalizar cada día se incorpore un coordinador, que además de buscar nuevos conflictos en colaboración con los ingenieros de requisitos participantes, realizara estudios estadísticos, de la cantidad de símbolos obtenidos, tipos de símbolos, porcentaje del total, y la cantidad de símbolos totales al resolver los conflictos.

4. Ejecución de los casos de estudio

Esta sección presenta la ejecución realizada, con la descripción de las empresas involucradas, ejemplos de conflictos hallados de los tres casos de estudio que fueron realizados en cada una de las categorías de conflictos y características de los participantes de cada caso de estudio.

En cada caso de estudio se definió un LEL de modo colaborativo durante 7 días, realizado por tres ingenieros de requisitos, que fueron diferentes en cada caso de estudio. Tres compañías reales que realizan distintas

actividades fueron seleccionadas para definir el LEL colaborativamente en esos dominios. Todas las compañías están ubicadas en Argentina. En las tres compañías los participantes tuvieron acceso para hacer entrevistas con los [stakeholders](#). Los participantes además obtuvieron información de las compañías mediante sus respectivas páginas web.

En cada caso de estudio se generó el LEL colaborativamente y se resolvieron los conflictos que hallaron mientras trabajaban. Para ello, tuvieron que identificar los símbolos, describirlos (con noción e impacto), y buscar si existían conflictos. Se incluyó un coordinador al equipo de trabajo para ayudarlos en la búsqueda y resolución de conflictos al final de cada día, antes de continuar el proceso. Al final de cada día el coordinador hizo una revisión del LEL: símbolos hallados, tipos de símbolos hallados, total de símbolos definidos hasta ese día, conflictos resueltos cada día. La revisión verificó también si había símbolos de los cuatro tipos (Sujeto, Verbo, Objeto, Estado).

Todos los participantes involucrados en los casos de estudio tenían una experiencia de más de 2 años en la industria de desarrollo de software. Fue realizado por profesionales de áreas relacionadas, en su mayoría estudiantes o egresados de la Universidad Nacional de La Matanza, en Buenos Aires, Argentina. Algunos participantes eran investigadores asistentes. Entre los tres grupos, había un total de una mujer y 8 hombres; sus edades entre 22 y 40 años de edad; el coordinador fue también una mujer, de 56 años de edad.

La empresa [Megaflex](#) fue la primera compañía relevada. Esta empresa se encuentra en el Parque Industrial de [Buenos Aires](#), en la Provincia de Buenos Aires. [Megaflex](#) es una fábrica que manufactura diferentes tipos de membrana u otros productos asfálticos para protección de techos. También ofrecen sistemas completos de solución de problemas de humedad en techos. Para la realización del LEL de este caso de estudio se [consiguió](#) contacto continuo con el primer dueño y fundador de [Megaflex](#), quien fue entrevistado en varios encuentros, por los ingenieros de requisitos intervinientes. [Algunos](#) fueron entrevistados empleados de otras áreas: trabajadores de la línea de producción y administrativos. Se consultó además la página web de la empresa: <https://www.megaflex.com.ar/>.

La segunda compañía seleccionada fue la empresa [Assistard](#), Argentina. Es una compañía de asistencia al viajero. Los ingenieros de requisitos tuvieron contacto con personas en diferentes roles de la compañía. [Assistard](#) provee a los viajeros varios tipos posibles de asistencia al viajero, tanto individual como grupal, y con diferentes tipos de seguros, desde los seguros ordinarios hasta los seguros de alto riesgo. Los clientes pueden contratar la asistencia al viajero mediante agencias de viajes, en aeropuertos o por Internet. Los ingenieros de



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019

requisitos tuvieron acceso a personal de distintos niveles, gerentes, administrativos y vendedores, consultando también la página web de la empresa: <https://www.assistcard.com.ar/>.

La tercera compañía seleccionada es una compañía que fabrica tejidos de punto para la industria textil. Se trabajó con la empresa *libra*, parte del grupo *Nordabril*. Textil *libra* es una empresa que tiene su administración general en Capital Federal, pero tiene sucursales en Capital Federal, Resistencia, Corrientes y Villa Mercedes. Dado que pertenece al grupo *Nordabril*, cerca de la mitad de la producción de Textil *libra* se destina a la producción de prendas que el grupo comercializa. La parte restante de la producción se destina a empresas de primera línea que comercializan prendas de prestigio internacional y nacional. Los ingenieros de requisitos entrevistaron a varios involucrados de diferentes niveles, realizando algunas entrevistas online a la provincia de Corrientes, y otras presenciales en la sede administrativa de Capital Federal. Se consultó asimismo la página web de la empresa: <https://www.textilibera.com.ar/compania.html>.

La clasificación de conflictos aplicada en los tres casos de estudio fue la propuesta en [2]. Los conflictos fueron clasificados en:

- Conflictos semánticos:
 - La misma identificación para elementos con diferente significado y la misma clasificación sintáctica (homónimos).
 - Diferente identificación para elementos que se refieren al mismo concepto de la misma manera (sinónimos).
 - Diferente identificación para elementos que se refieren al mismo concepto con información complementaria.
- Conflictos estructurales:
 - Diferente nivel de detalle.
 - Descripciones duplicadas en jerarquías.
- Conflictos sintácticos:
 - La misma identificación para elementos con diferente clasificación sintáctica (homónimos).

A continuación se presentará un ejemplo de conflicto que se observó de cada tipo, semántico, estructural y sintáctico, para cada caso de estudio.

Es importante observar que no todos los conflictos han surgido en todos los dominios. En particular el caso de los Conflictos sintácticos, no surgió en el dominio de la empresa *Assistcard*.

Debe observarse además que la existencia de conflictos depende del idioma en que se trabaje. Por ejemplo en inglés *label* quiere decir Etiquetar y además

quiere decir Etiqueta, lo cual sería un conflicto de tipo sintáctico en inglés, siendo palabras distintas en castellano.

El primer conflicto presentado entre los conflictos semánticos es "La misma identificación para elementos con diferente significado y la misma clasificación sintáctica (homónimos)".

En el caso de estudio *Megaflex*, tres ingenieros de requisitos diferentes han denominado Etiquetar a tres acciones distintas: Etiquetar definido por el ingeniero de requisitos 3 implica la colocación del precio, Etiquetar para el ingeniero de requisitos 1 implica la colocación del código de barras, y Etiquetar para el ingeniero de requisitos 2 es la colocación del logo de *Megaflex*. (Ver Tablas 2, 3 y 4).

Tabla 2. Símbolo Etiquetar del Ingeniero de Requisitos 3

Símbolo # 66	Autor: Ing. Req. 3	Tipo: Verbo
Nombre/s	Etiquetar	
Noción	- Es la acción de colocar el <u>precio</u> en el <u>producto terminado</u>	
Impacto	-Se realiza cuando el <u>producto fue finalizado</u> .	

Tabla 3. Símbolo Etiquetar del Ingeniero de Requisitos 1

Símbolo # 67	Autor: Ing. Req. 1	Tipo: Verbo
Nombre/s	Etiquetar	
Noción	- Es la acción de colocar el <u>código de barras</u> en el <u>producto terminado</u> .	
Impacto	-Se realiza cuando el <u>producto fue finalizado</u> .	

Tabla 4. Símbolo Etiquetar del Ingeniero de Requisitos 2

Símbolo # 68	Autor: Ing. Req. 2	Tipo: Verbo
Nombre/s	Etiquetar	
Noción	- Es la acción de colocar el <u>logo de Megaflex</u> en el <u>producto terminado</u> .	
Impacto	-Se realiza cuando el <u>producto fue finalizado</u> .	

Como resultan homónimos la solución propuesta es denominarlos Etiquetar (1), Etiquetar (2), y Etiquetar (3), manteniendo los 3 símbolos definidos.

En los conflictos estructurales, en *Megaflex* surge el conflicto "Diferente nivel de detalle" en los símbolo Membrana, definido por el ingeniero de requisitos 1, en contraposición con el ingeniero de requisitos 2.

Este ingeniero de requisitos no define el símbolo Membrana, sino los símbolos Membrana transitable y Membrana no transitable únicamente, mientras el ingeniero de requisitos 1 define sólo Membrana (Ver Tablas 5, 6 y 7).



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

Tabla 5. Símbolo Membrana del Ingeniero de Requisitos 1

Símbolo # 2	Autor: Ing. Rcg. 1	Tipo: Objeto
Nombre/s	Membrana	
Noción	- Es el producto asfáltico que se coloca en el techo de las viviendas para prevenir filtraciones.	
Impacto	- Puede ser transitable o no transitable	

Tabla 6. Símbolo Membrana transitable del Ingeniero de Requisitos 2

Símbolo # 5	Autor: Ing. Rcg. 2	Tipo: Objeto
Nombre/s	Membrana transitable	
Noción	- Es un producto asfáltico para protección de techos.	
Impacto	- Permite el paso sobre la superficie.	

Tabla 7. Símbolo Membrana no transitable del Ingeniero de Requisitos 2

Símbolo # 4	Autor: Ing. Rcg. 2	Tipo: Objeto
Nombre/s	Membrana no transitable	
Noción	- Es un producto asfáltico para protección de techos. - Contiene poliéster en su capa superior.	
Impacto	- No permite el paso sobre la superficie.	

La solución propuesta es definir el símbolo Membrana como genérico de Membrana no transitable y Membrana transitable. En las definiciones de los especializados referenciar a Membrana, diciendo "Es una Membrana tal que...".

En los Conflictos sintácticos en la empresa **MIRALIX**, el conflicto "La misma identificación para elementos con diferente clasificación sintáctica (homónimos)" se presenta en el símbolo Venta registrada, cuando al mismo nombre de símbolo el ingeniero de requisitos 2 lo definió como un **Objeto**, y el ingeniero de requisitos 3 lo definió como un Estado (Ver Tablas 8 y 9).

Tabla 8. Símbolo Venta registrada del Ingeniero de Requisitos 2

Símbolo # 48	Autor: Ing. Rcg. 2	Tipo: Objeto
Nombre/s	Venta registrada	
Noción	- Es un producto asfáltico para protección de techos.	
Impacto	- Permite el paso sobre la superficie.	

Tabla 9. Símbolo Venta registrada del Ingeniero de Requisitos 3

Símbolo # 50	Autor: Ing. Rcg. 3	Tipo: Estado
Nombre/s	Venta registrada	
Noción	- Es un producto asfáltico para protección de techos. - Contiene poliéster en su capa superior.	
Impacto	- Permite el paso sobre la superficie.	

La solución propuesta es denominar a los símbolos Venta registrada (1) y Venta registrada (2).

En la empresa **ASSICORAL**, entre los Conflictos semánticos, el conflicto "Diferente identificación para elementos que se refieren al mismo concepto con información complementaria" surge con el símbolo de tipo Estado denominado **Voucher validado** del ingeniero de requisitos 3, con el símbolo **Validado** del ingeniero de requisitos 1. El impacto "El **voucher** llega a ese estado luego de generar la nota de crédito." fue definido por el ingeniero de requisitos 1 pero no por el ingeniero de requisitos 3. Se agrega a la definición del símbolo único con ambos nombres esta la información complementaria, colocando en primer lugar el nombre más usado en el dominio (Ver Tabla 10).

Tabla 10. Símbolo Voucher validado / Validado

Símbolo # 17	Autor: Ing. Rcg. 3,1	Tipo: Estado
Nombre/s	- Voucher validado / Validado	
Noción	- Estado que toma un voucher luego de la acción validar voucher .	
Impacto	- Se genera un voucher exactamente igual al original pero con una fecha de baja. - Se genera una nota de crédito al tratarse de un voucher facturado . - El voucher llega a este estado luego de generar la nota de crédito.	

En **ASSICORAL** surge un conflicto estructural de "Diferente nivel de detalle" cuando el ingeniero de requisitos 2 ha definido el símbolo Ingresar seguro a cliente, mientras que el ingeniero de requisitos 1 ha definido Registrar seguro de vida y Registrar seguro de deportes extremos. Estos últimos dos símbolos tienen mayor nivel de detalle que el símbolo Ingresar seguro a cliente (Ver Tablas 11, 12 y 13).

Tabla 11. Símbolo Ingresar seguro a cliente del Ingeniero de Requisitos 2

Símbolo # 57	Autor: Ing. Rcg. 2	Tipo: Verbo
Nombre/s	Ingresar seguro a cliente	
Noción	- Es la acción de registrar un seguro para un cliente dado.	
Impacto	- Luego de ingresado se deriva al departamento de asistencia.	

Tabla 12. Símbolo Registrar seguro de vida del Ingeniero de Requisitos 1

Símbolo # 58	Autor: Ing. Rcg. 1	Tipo: Verbo
Nombre/s	Registrar seguro de vida	
Noción	- Es la acción de registrar un seguro de vida para un cliente dado.	
Impacto	- Es derivado al departamento de asistencia - Se registran datos de beneficiarios para el seguro de vida.	



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019

Tabla 13. Símbolo Registrar seguro para deportes extremos del Ingeniero de Requisitos 1

Símbolo # 59	Autor: Ing. Req. 1	Tipo: Verbo
Nombre/s	Registrar seguro para deportes extremos	
Noción	-Es la acción de registrar un seguro para deportes extremos para un cliente dado.	
Impacto	-Es derivado al departamento de asistencia.	

Este conflicto tiene como solución en el LEL definir el símbolo Ingresar seguro a cliente como genérico de los símbolos Registrar seguro de vida y Registrar seguro para deportes extremos. En las definiciones de los especializados referenciar a Ingresar seguro a cliente, diciendo "Es un ingreso de seguro de vida tal que...".

Como mencionamos con anterioridad no hay conflictos sintácticos en *Assisgard*.

Si consideramos el caso de estudio Textil *Ibera*, se presenta el conflicto semántico "Diferente identificación para elementos que se refieren al mismo concepto de la misma manera (sinónimos)" en el símbolo definido por el ingeniero de requisitos 3 llamado Pago consolidado, y el ingeniero de requisitos 2 ha definido el símbolo Pago aceptado (Ver Tablas 14 y 15).

Tabla 14. Símbolo Pago consolidado del Ingeniero de Requisitos 3

Símbolo # 44	Autor: Ing. Req. 3	Tipo: Estado
Nombre/s	Pago consolidado	
Noción	-Condición por la cual un Pago llegó a concretarse totalmente.	
Impacto	-Genera Actualizar Cuenta Corriente del cliente o del proveedor. -Se marcan las facturas canceladas como pagadas.	

Tabla 15. Símbolo Pago aceptado del Ingeniero de Requisitos 2

Símbolo # 44	Autor: Ing. Req. 2	Tipo: Estado
Nombre/s	Pago aceptado	
Noción	-Condición por la cual un Pago llegó a concretarse en su totalidad.	
Impacto	-Desencadena Actualizar Cuenta Corriente del cliente o del proveedor. -Se marcan las facturas canceladas como pagadas.	

Como resultado se define un único símbolo con ambos sinónimos Pago *aceptado* / Pago consolidado, colocando Pago aceptado en primer lugar por ser el nombre más utilizado en el dominio. El símbolo final puede verse en la Tabla 16.

Tabla 16. Símbolo Pago aceptado / Pago consolidado del Ingeniero de Requisitos 1 y 3

Símbolo # 44	Autor: Ing. Req. 1 y 3	Tipo: Estado
Nombre/s	Pago aceptado / Pago consolidado	
Noción	-Condición por la cual un Pago llegó a concretarse en su totalidad.	
Impacto	-Desencadena Actualizar Cuenta Corriente del cliente o del proveedor. -Se marcan las facturas canceladas como pagadas.	

Entre los conflictos estructurales el conflicto "Diferente nivel de detalle" surge en el dominio de la empresa Textil *Ibera*. El símbolo "Operador" es definido por el ingeniero de requisitos 1, mientras que el ingeniero de requisitos 2 ha definido los *especializados* pero no el genérico. Los especializados que ha definido son "Operario de facturación", "Operario de tesorería" y "Operario de ventas" (Ver Tablas 17, 18, 19 y 20). En consecuencia, como solución se define la jerarquía y en el caso de cada operador se define su noción como "Es el operador que...", refiriendo al genérico Operador.

Tabla 17. Símbolo Operador del Ingeniero de Requisitos 1

Símbolo # 15	Autor: Ing. Req. 1	Tipo: Sujeto
Nombre/s	Operario	
Noción	-Es la persona responsable de las actividades secundarias realizadas dentro del sector de Facturación, Tesorería o Ventas. -Ocupa un rol de menor jerarquía que el Jefe de Sector.	
Impacto	-Realizan las tareas encargadas por el Jefe de sector. -Se encarga de la facturación, de la tesorería o de ventas.	

Tabla 18. Símbolo Operario de facturación del Ingeniero de Requisitos 2

Símbolo # 16	Autor: Ing. Req. 2	Tipo: Sujeto
Nombre/s	Operario de facturación	
Noción	-Es la persona responsable de las actividades secundarias realizadas dentro del sector de Facturación. -Ocupa un rol de menor jerarquía que el Jefe de Sector.	
Impacto	-Es el encargado de emitir facturas y remitos de venta. -Es el encargado de enviar facturas y remitos de venta. -Es el encargado de confirmar recepción de facturas y remitos. -Es el encargado de recibir facturas y remitos de compras.	



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019

Tabla 19. Símbolo Operario de tesorería del Ingeniero de Requisitos 2

Símbolo # 17	Autor: Ing. Req. 2	Tipo: Sujeto
Nombre/s	Operario de tesorería	
Nación	-Es la persona responsable de las actividades secundarias realizadas dentro del sector de Tesorería. -Ocupa un rol de menor jerarquía que el Jefe de Sector.	
Impacto	-Lleva a cabo los tareas de controlar caja. -Se encarga de informar los pagos emitidos.	

Tabla 20. Símbolo Operario de ventas del Ingeniero de Requisitos 3

Símbolo # 18	Autor: Ing. Req. 2	Tipo: Sujeto
Nombre/s	Operario de ventas	
Nación	-Es la persona responsable de las actividades secundarias realizadas dentro del sector de ventas. -Ocupa un rol de menor jerarquía que el Jefe de Sector.	
Impacto	-Puede Registrar Pedido. -Puede Modificar Pedido. -Puede Cancelar Pedido. -Puede consultar el estado de un pedido.	

En Textil (1994) se presenta el conflicto sintáctico al surgir homónimos entre el verbo Gestión de caja del ingeniero de requisitos 3 y Gestión de caja del ingeniero de requisitos 2 como objeto (Ver Tablas 21 y 22).

Tabla 21. Símbolo Gestión de caja del Ingeniero de Requisitos 2

Símbolo # 8	Autor: Ing. Req. 2	Tipo: Objeto
Nombre/s	Gestión de caja	
Nación	-Es el monto de entradas y salidas de dinero en efectivo originado por pagos emitidos o recibidos.	
Impacto	-Es generado duramente por el Operario de Tesorería. -Es utilizado como fuente de información al elaborar el Pronóstico de Ventas.	

Tabla 22. Símbolo Gestión de caja del Ingeniero de Requisitos 3

Símbolo # 9	Autor: Ing. Req. 3	Tipo: Verbo
Nombre/s	Gestión de caja	
Nación	-Es la acción diaria por la cual el Operario de Tesorería hace el ingreso o egreso de efectivo de la caja.	
Impacto	-Se verifica la cantidad de dinero en efectivo con la factura a abonar o a cobrar. -Se entrega el dinero o se guarda separando los distintos billetes. -Se asienta el movimiento en un registro contador.	

La solución de este conflicto es reemplazar ~~operarios~~ de símbolos denominados anteriormente Gestión de caja, de ambos ingenieros de requisitos llamando el objeto como Gestión de caja (1) y al verbo como Gestión de caja (2)

5. Análisis de resultados

Los casos de estudio fueron diseñados y ejecutados tratando de evitar efectos adversos y amenazas a la validez de los mismos.

Para obtener validez externa los casos de estudio se realizaron con profesionales del área de ~~desarrollo~~ con títulos de grado y/o posgrado, pero con un grupo heterogéneo, por tener trabajos diferentes (analistas, desarrolladores, ~~gestores~~ líderes de proyecto o con otras tareas), mejorando así la validez externa. Fueron personas de ambos sexos, con diferentes edades, entre 22 y 56 años. Todos tenían más de 2 años de trabajo en su rol, en general de 2 a 5 años en esa tarea.

Para obtener validez interna los casos de estudio se realizaron en diferentes horarios acorde a los tiempos disponibles para ello de cada profesional interviniente, pero siempre con el mismo contexto de trabajo.

Aplicando el proceso de búsqueda y solución de conflictos durante 7 días, en los 3 casos de estudio, surgieron gran cantidad de conflictos. Estos resultados se muestran en la Tabla 23. Algunos símbolos están involucrados en más de un conflicto.

La Tabla 23 muestra además el porcentaje de símbolos con conflictos, si se considera el total de símbolos obtenidos una vez resueltos los conflictos, en cada caso de estudio. Dicho porcentaje resultó del 75 %, 92 % y 83 %. De esta manera esta tabla muestra que la presencia de conflictos es un hecho muy común, evidenciando la importancia de resolver los conflictos.

Tabla 23. Total de conflictos hallados en cada caso de estudio

Casos de estudio	Total de símbolos (Conflictos resueltos)	Símbolos involucrados en conflictos	Porcentaje de símbolos involucrados en conflictos
Megafona	69	52	75
Autocarro	62	57	92
Isota	59	49	83

Otro resultado obtenido es la observación de la gran cantidad de símbolos que habrían sido definidos erróneamente, con problemas entre ellos, al no resolver los conflictos existentes.

La Tabla 24 muestra la Disminución de símbolos al identificar y resolver conflictos, mediante el total de



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLAM
Versión	5
Vigencia	03/9/2019

símbolos si no se resuelven los conflictos, el total de símbolos con conflictos resueltos, y la diferencia entre el total de símbolos sin conflictos resueltos y con conflictos resueltos (exceso de información).

Tabla 24. Disminución de símbolos al identificar y resolver conflictos

Casos de estudio	Total de símbolos si no se resuelven los conflictos	Total de símbolos con conflictos resueltos	Diferencia entre el total de símbolos sin conflictos resueltos y con conflictos resueltos (exceso de información)
Megaflex	113	69	44
Assistcard	110	62	48
Iberrá	88	59	29

El exceso de información corresponde a la cantidad de símbolos que habrían sido definidos erróneamente en cada caso de estudio (44, 48 y 29 símbolos respectivamente), con problemas entre ellos, al no resolver los conflictos existentes entre los distintos involucrados.

Se evalúa en la Tabla 25 la incidencia de la resolución de conflictos sobre el tamaño del modelo LEL, en cada uno de los tres casos de estudio realizados. Es decir que el tamaño del modelo LEL de cada caso de estudio habría sido mucho mayor si no se hubieran resuelto los conflictos entre los símbolos definidos por los distintos ingenieros de requisitos intervinientes en cada caso de estudio.

Tabla 25. Porcentaje de aumento del tamaño del modelo (exceso de información)

Casos de estudio	Total de símbolos si no se resuelven los conflictos	Diferencia entre el total de símbolos sin conflictos resueltos y con conflictos resueltos (exceso de información)	Porcentaje de aumento del tamaño del modelo LEL (exceso de información)
Megaflex	113	44	39%
Assistcard	110	48	44%
Iberrá	88	29	33%

Entonces en el caso de estudio Megaflex el exceso de información habría significado un aumento del tamaño

del modelo del 39%, en el caso de estudio Assistcard el exceso de información habría significado un aumento del tamaño del modelo del 44%, y en el caso de estudio Textil Iberrá sería un exceso de información que habría significado un aumento del tamaño del modelo del 33%.

Los resultados obtenidos muestran que el proceso definido en [2] es aplicable. En los tres casos de estudio se observó entonces la disminución de la cantidad de símbolos de cada modelo al resolver los conflictos y también la gran cantidad de símbolos involucrados en conflictos.

6. Conclusiones y trabajo futuro

En la Ingeniería de Requisitos, una de las etapas preliminares del ciclo de vida del desarrollo de software, los stakeholders están naturalmente involucrados.

El Lenguaje Natural aparece como una excelente opción para compartir el conocimiento entre los diferentes stakeholders. Por ello, el Léxico Extendido del Lenguaje (LEL) escrito en Lenguaje Natural, es una buena ayuda para definir el vocabulario del dominio de aplicación. La construcción colaborativa del LEL, posibilita obtener un modelo más rico, pero implica la aparición de conflictos.

Este trabajo se ha basado en el proceso definido para hallar y resolver los conflictos que surgen en la construcción colaborativa del LEL presentado por Litvak, en [2]. Se ha utilizado dicho proceso, el catálogo de conflictos y soluciones, y la categorización de conflictos presentados en [2].

Se aplicó dicho proceso y el catálogo de conflictos definido, en tres nuevos casos de estudio, con distintos dominios y equipos de trabajo diferentes, definiendo colaborativamente el LEL. Los casos de estudio fueron empresas reales de Argentina, donde se realizaron entrevistas a diferentes stakeholders, utilizando además las páginas web de las empresas. Las empresas seleccionadas fueron Megaflex, Assistcard, Argentina y Textil Iberrá.

En todos los casos de estudio se observó la gran cantidad de símbolos que habrían sido definidos erróneamente en cada caso de estudio (44, 48 y 29 símbolos respectivamente), con problemas entre ellos, al no resolver los conflictos existentes entre los distintos involucrados.

Se evaluó también el porcentaje de aumento del tamaño LEL si no se hubieran resuelto los conflictos, obteniendo en los tres casos de estudio un exceso de información del 39%, 44% y 33% respectivamente.

Megaflex se observó el alto porcentaje de símbolos involucrados en conflictos, siendo en los tres casos de



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

estudio respectivamente el 75 %, 92% y 83 %. Esto muestra la importancia de resolver los conflictos.

Se observa entonces la aplicabilidad del proceso definido y los beneficios de aplicarlo. Un LEL sin conflictos resulta más pequeño y más fácil de comprender, evitando el exceso de información.

Como trabajo futuro se propone realizar la validación de las soluciones de conflictos mediante un experimento en Nigeria, con la colaboración de Dr. Uwaia, Uwaia Gambo, de la Faculty of Technology, Obafemi Awolowo University, Ile-Ife, Nigeria. Dicho experimento ha sido ya diseñado y posteriormente testado en la Universidad Nacional de La Plata, y está siendo llevado a cabo actualmente con profesionales del área de sistemas en Nigeria. Luego se realizarán estudios estadísticos sobre los resultados.

7. Referencias

- [1] IEEE Systems and software engineering, Life cycle processes, Requirements engineering 2011, IEEE, New York.
- [2] Litvak, C., Antonelli, L., and Rossi G., "Conflict Management in the Collaborative Description of a Domain Language", *IEEE*, de 2018, Thirtieth International Conference on Software Engineering and Knowledge Engineering (SEKE 2018), San Francisco Bay, USA, doi: 10.18293/SEKE2018-106.
- [3] Leite, J. C. S. D. P., and Franco, A. P. M., "A strategy for conceptual model acquisition", Requirements Engineering, 1993, Proceedings of IEEE International Symposium on, IEEE, January 1993, pp. 243-246.
- [4] Leite, J.C.S.P., and Oliveira, A.P.A., "A Client Oriented Requirements Baseline", Second IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, 1995, pp. 108-115.
- [5] Leite, J.C.S.P., Rossi, G., Bolauer, F., Maiorana, V., Kaplan, G., Hadad, G., and Olivares, A., "Enhancing a Requirements Baseline with Scenarios", Requirements Engineering Journal, Springer-Verlag London Ltd., Vol. 2, N° 4, 1997, pp. 184-198.
- [6] Easterbrook S., "Resolving requirements conflicts with computer-supported negotiation", Requirements engineering: social and technical issues, vol. 1, 1994, pp. 41-65.
- [7] Robinson W. N., Robinson S. D., and Sobay V., "Requirements interaction management. ACM Computer Survey, vol. 35(2), 2003, pp. 132-190.
- [8] Cameron, E. J., and Verhulisen, H., "Feature interactions in telecommunications systems", IEEE Communications Magazine, 31(8), 1993, pp. 18-23.
- [9] Robinson, W. N., and Sobay, V., "Requirement Conflict Restructuring, Conflict-Oriented Requirement Restructuring", GSU CIS Working Paper 99-5, 1999, pp. 1-47.
- [10] Bendisana, H., Charrel P. J., and Zarruk, N. E., "Using AHP Method to Resolve Conflicts Between Non-Functional Concerns", International Conference on Education, Applied Sciences and Management (ICEASM2012), Dubai, UAE, 2012, pp. 26-27.
- [11] Bendisana, M. A., Costa, R., and Zarruk, M., "Enhancing conflict resolution through an AHP-based methodology", International Journal of Management and Decision Making, 13(1), 2014, pp. 17-41.
- [12] Nareedou, M., and Jureddou, K., "Requirement Negotiation Methods in Requirements Engineering", International Journal of Advanced Research in Computer Science, 8(3), 2017.
- [13] Abdelkhalil M., Chikh A., and Ziani, D., "Software Requirements Conflict Identification: Review and Recommendations", International Journal of advanced computer science and applications, vol. 7, no. 10, 2016, pp. 326-335.
- [14] Decker B., Ras E., Rech J., Jaubert P., and Rieth M., "Wiki-based stakeholder participation in requirements engineering", IEEE Software, vol. 24(2), 2007, pp. 28-35.
- [15] Robertson J., and Robertson S., "Value Requirements Specification Template", The Atlantic Systems Guild, 2012.
- [16] Yang D., Wu D., Koolmanojwong S., Brown, A. W., and Boelun B. W., "Wikiwinwin: A wiki based system for collaborative requirements negotiation", Hawaii International Conference on System Sciences, Proceedings of the 41st Annual, pp. 24-24. IEEE, 2008.
- [17] Urbieto M., Escalona M. J., Luna E. R., and Rossi, G., "Detecting conflicts and inconsistencies in web application requirements", International Conference on Web Engineering, Springer, Berlin, Heidelberg, 2011, pp. 278-288.
- [18] Lutz R., Schäfer S., and Dielz S., "Using mobile devices for collaborative requirements engineering", Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering (ASE 2012), ACM, 2012, pp. 298-301, ISBN 9781450312042.



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019

[19] Azadegan A., Cheng X., Niederman F., and Yin G. "Collaborative requirements elicitation in facilitated collaboration: report from a case study", 46th Hawaii International Conference on System Sciences, 2013, pp. 569-578, IEEE. ISSN 15301605.

[20] ~~Simon~~ Simon, E., and ~~Luczak-Rosch~~ Luczak-Rosch, M., "Collaborative ontology engineering: a survey", The Knowledge Engineering Review, 29(1), 2014, pp. 101-131.

[21] Chen Y., Peng X., and Zhao W., "An approach to detect collaborative conflicts for ontology development", Advances in Data and Web Management, Springer, Berlin, Heidelberg, 2009, pp. 442-454.

[22] Karapiperis S., and Apostolou D., "Consensus building in collaborative ontology engineering processes", Journal of Universal Knowledge Management, vol 1(3), 2006, pp. 199-216.

[23] Nguyen N. T., "Advanced methods for inconsistent knowledge management". Springer, London, 2008.

[24] Duong T. H., Tran M. Q., and Nguyen T.P.T., "Collaborative Vietnamese WordNet building using consensus quality", Vietnam J ComputSci 2017, vol 4:85, Springer Berlin Heidelberg, Print ISSN: 2196-8888, Online ISSN: 2196-8896, 2017.

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



Se certifica que Claudia Litvak ha participado del VII Congreso Nacional de Ingeniería Informática – Sistemas de Información, CONAISI 2019, realizado los días 14 y 15 de noviembre en la Universidad Nacional de La Matanza.


Ing. Claudio D'Amico
Coord. Gral. CONAISI


Dr. Carlos Neil
Coordinador RIISIC


Mg. Jorge Eterovic
Decano DIIT



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



CONAISI

VII Congreso Nacional de Ingeniería
Informática - Sistemas de Información

2019

San Justo, 5 de diciembre de 2019

Se certifica que **Claudia Litvak, Leandro Antonelli, Gustavo Rossi, Ishaya Peni Gambo, Facundo Velazquez Santillán, Diego Lo Giudice, Ximena Milla, Nora Gigante** han participado como Autores del artículo 51 *"Resolución de conflictos en modelos de dominio en lenguaje natural: casos de estudio"*, aceptado en el VII Congreso Nacional de Ingeniería Informática – Sistemas de Información, CONAISI 2019, realizado los días 14 y 15 de noviembre en la Universidad Nacional de La Matanza.


Ing. Claudio D'Amico
Coord. Gral. CONAISI


Dr. Carlos Neil
Coordinador RIISIC


Mg. Jorge Eterovic
Decano DIIT



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



- **Anexo II:**
- FPI-013: Evaluación de alumnos integrantes.



UNLaM - SECyT

Programa PROINCE

FI-013

FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACIÓN

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas.

Código: C-206

Título del Proyecto: "Gestión de requisitos generados colaborativamente"

Director del Proyecto: Mg. Cristóbal Raúl Santa María

Programa de investigación: PROINCE X CyTMA2: ...

Fecha de inicio: 01/01/2018

Fecha de finalización: 31/12/2019

1. Datos del alumno

Apellido y Nombre del Alumno: Valazquez Santillán, Facundo Darío

DNI: 38126754

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Carrera que cursa: Ingeniería en Informática

Período evaluado: 01/01/2019 al 31/12/2019

2. Dictamen de evaluación de desempeño del alumno:

2.1 Satisfactorio: X

2.1 No satisfactorio:

Fundamentos del dictamen:

El alumno desarrolló las tareas programadas para el segundo año del proyecto de investigación. Las mismas consistieron en el estudio comparativo de los conflictos en ontologías en general, la búsqueda de las posibles soluciones a los conflictos en el trabajo colaborativo al elaborar modelos de requisitos y la realización de casos de estudio.

La interacción con los demás integrantes del equipo de investigación redundó en un interesante trabajo colaborativo. Su trabajo en general fue muy bueno.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)

Colocar una cruz donde corresponda

3.1 Continuar en el presente proyecto: X

3.2 No continuar en el presente proyecto:

Fundamentos del dictamen:

Su trabajo fue excelente. La interacción con los demás integrantes del equipo de investigación redundó en un fructífero trabajo colaborativo y en el avance de la investigación.

San Justo, 30 de diciembre de 2019

Mg. Cristóbal Raúl Santa María

Lugar y fecha Firma del Director

Aclaración de firma



Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



UNLaM - SECyT

Programa PROINCE

FI-013

FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACION

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas.

Código: C-206

Título del Proyecto: "Gestión de requisitos generados colaborativamente"

Director del Proyecto: Mg. Cristóbal Raúl Santa María

Programa de investigación: ~~PROINCE-X~~ CyTMA2: ...

Fecha de inicio: 01/01/2018

Fecha de finalización: 31/12/2019

Apellido y Nombre del Alumno: ~~Lo Giudice~~ Diego Andrés

DNI: 34738835

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas

Carrera que cursa: Ingeniería en Informática

Periodo evaluado: 01/01/2019 al 31/12/2019

2. Dictamen de evaluación de desempeño del alumno:

2.1 Satisfactorio: X

2.1 No satisfactorio:

Fundamentos del dictamen:

El alumno Diego Lo Giudice desarrolló las tareas programadas para el segundo año del proyecto de investigación. Trabajó en la Participó en la investigación de los conflictos que pudieran surgir como consecuencia del trabajo colaborativo al elaborar modelos de requisitos y en la realización de casos de estudio.

Colaboró en el estudio estadístico, de la cantidad de símbolos involucrados en conflictos en los casos de los distintos ingenieros de requisitos intervinientes.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)

Colocar una cruz donde corresponda

3.1 Continuar en el presente proyecto: X

3.2 No continuar en el presente proyecto:

Fundamentos del dictamen:

Su trabajo fue muy bueno. La interacción con los demás integrantes del equipo de investigación redundó en un fructífero trabajo colaborativo y en el avance de la investigación.

San Justo, 30 de diciembre de 2019

Lugar y fecha

Mg. Cristóbal Raúl Santa María

Firma del Director

Aclaración de firma

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



UNLaM - SECyT

Programa PROINCE

FI-013

FORMULARIO DE EVALUACIÓN DE ALUMNOS INTEGRANTES DE EQUIPOS DE INVESTIGACION

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas.
 Código: C-206
 Título del Proyecto: "Gestión de requisitos generados colaborativamente"
 Director del Proyecto: Mg. Cristóbal Raúl Santa María
 Programa de investigación: PROINCE-X, CyTMA2: ...
 Fecha de inicio: 01/01/2018
Fecha de finalización: 31/12/2019

.....
Apellido y Nombre del Alumno: Ximena Anabela Milla
DNI: 32723071

Unidad Académica: Departamento de Ingeniería e Investigaciones Tecnológicas
Carrera que cursa: Ingeniería en Informática
 Periodo evaluado: 01/01/2019 al 31/12/2019

2. Dictamen de evaluación de desempeño del alumno:

- 2.1 Satisfactorio: X
 2.1 No satisfactorio:

Fundamentos del dictamen:

La alumna Ximena Milla desarrolló las tareas programadas para el segundo año del proyecto de investigación. Participó activamente en el Estudio de la generación colaborativa de requisitos, en la investigación de los conflictos que pudieran surgir como consecuencia del trabajo colaborativo al elaborar modelos de requisitos y en la realización de casos de estudio.

Trabajó en la realización de una categorización de conflictos, evaluando diferentes categorizaciones de conflictos en general de diferentes autores y también la categorización de conflictos en requisitos en particular.

3. Propuesta de continuidad en el proyecto (si corresponde según duración estimada)

Colocar una cruz donde corresponda

- 3.1 Continuar en el presente proyecto: X
 3.2 No continuar en el presente proyecto:

Fundamentos del dictamen:

Su trabajo fue muy destacado. La interacción con los demás integrantes del equipo de investigación redundó en un fructífero trabajo colaborativo y en el avance de la investigación.

San Justo, 30 de diciembre de 2019

Mg. Cristóbal Raúl Santa María

Lugar y fecha Firma del Director

.....
 Aclaración de firma

Código	FPI-009
Objeto	Guía de elaboración de Informe final de proyecto
Usuario	Director de proyecto de investigación
Autor	Secretaría de Ciencia y Tecnología de la UNLaM
Versión	5
Vigencia	03/9/2019



Firma y aclaración
del director del proyecto.

Lugar y fecha :.....