

# Inspecciones para Mejorar la Calidad de Modelos en Lenguaje Natural

Graciela D. S. Hadad<sup>1,2</sup>, Alberto Sebastián<sup>1</sup>

<sup>1</sup> Facultad de Tecnología Informática, Universidad de Belgrano, Argentina

<sup>2</sup> Escuela de Informática, Universidad Nacional del Oeste, Argentina  
{graciela.hadad, alberto.sebastian}@comunidad.ub.edu.ar

**Resumen.** Las inspecciones han probado ser de gran eficacia en la detección de defectos en código fuente, por lo que esta técnica se ha adaptado para verificar modelos elaborados en el proceso de especificar requisitos. La literatura ha propuesto algunas variantes de inspección en el modo de detectar defectos en estos modelos, sin embargo, los estudios empíricos realizados no alcanzan para establecer qué variante puede ser más apropiada en determinadas situaciones, en función de la calidad exigida y tiempos disponibles. Dado el uso frecuente de modelos escritos en lenguaje natural en el proceso de requisitos, se ha llevado a cabo un experimento para comparar la eficiencia y eficacia de tres variantes de inspección aplicadas a 5 modelos léxico. Los resultados obtenidos permiten establecer una aproximación inicial en cuanto a que la variante basada en la lectura constructiva detecta un número superior de defectos, principalmente omisiones y ambigüedades, aunque insumiendo más tiempo.

**Palabras Clave:** Verificación, Inspecciones, Ingeniería de Requisitos, Modelos en Lenguaje Natural.

## 1 Introducción

Se han diseñado muy variados métodos para sistematizar el desarrollo de software y mejorar su nivel de calidad. Sin embargo, muchos proyectos de software han fracasado debido principalmente a una pobre especificación de requisitos. Una respuesta a este problema está dada en el marco de la Ingeniería de Requisitos (IR) mediante la verificación de sus modelos usando técnicas efectivas como las inspecciones [1-3], pues estos modelos son la base para construir los siguientes artefactos en el proceso de desarrollo de software.

En la IR han surgido diversas modalidades para detectar defectos mediante una inspección [1] y se han realizado estudios sobre la efectividad de la técnica [2-6], sin embargo, quedan aún cuestiones no resueltas sobre cual variante puede ser más conveniente según el caso. Es por ello que se presenta un estudio comparativo de tres variantes de inspección aplicadas sobre un modelo denominado Léxico Extendido del Lenguaje (LEL) [7]. Este modelo define la terminología utilizada en el contexto de aplicación bajo estudio, y se lo construye al iniciar un proceso de IR, por lo que su

calidad puede impactar en modelos derivados de él, escritos en lenguaje natural (LN), tales como escenarios, casos de uso e historias de usuario [7,8]. El propósito del estudio ha sido identificar la eficacia y eficiencia de las tres variantes, permitiendo colaborar en la toma de decisiones sobre la variante de inspección apropiada para la calidad esperada y los plazos previstos.

En la sección 2 se resume la técnica de inspección y sus variantes en la IR. La sección 3 presenta el modelo léxico y las variantes de inspección aplicadas al mismo. En la sección 4 se describe el experimento realizado y los datos obtenidos de cada variante de inspección. En la sección 5 se analizan los resultados, y finalmente en la sección 6 se exponen conclusiones y trabajos futuros.

## 2 Inspecciones en la Ingeniería de Requisitos

Las inspecciones fueron inicialmente diseñadas para detectar defectos en código fuente [9], y dada su alta eficacia [10,11] fueron adoptadas para verificar modelos de diseño y de requisitos. La técnica involucra la participación de inspectores, autores del material, un moderador y un escriba, siguiendo un proceso de 6 etapas: 1) Planificación de la inspección; 2) Visión General, donde los autores describen a los inspectores el material a inspeccionar; 3) Preparación o lectura del material por parte de los inspectores; 4) Reunión de Inspección, donde los inspectores revisan el material detectando defectos y los autores los confirman o rechazan, asistidos por el moderador mientras el escriba registra los hechos; 5) Corrección de los defectos a cargo de los autores; y 6) Seguimiento del estado del material.

En la IR se han propuesto diferentes variantes para identificar defectos en una inspección, centrando la detección en la etapa de Preparación, y ellas son [9]:

*Lectura ad-hoc*: el inspector lee el material e identifica defectos dependiendo exclusivamente de su experiencia, en un modo desestructurado.

*Lectura basada en lista de comprobación*: el inspector usa una lista de control que le sirve de ayuda-memoria para identificar defectos.

*Lectura basada en procedimientos*: el inspector sigue procedimientos detallados para capturar defectos, en un modo altamente estructurado.

*Lectura constructiva*: el inspector produce un artefacto intermedio a partir del material y sobre dicho artefacto detecta los defectos.

En el trabajo presentado en [6], se evaluó la influencia del nivel de experiencia del inspector, resultando que profesionales lograron mayor eficacia que estudiantes al detectar defectos en casos de uso, bajo la lectura ad-hoc. Resultados similares se habían presentado sobre especificaciones de requisitos [3]. En otro experimento [2], se evaluó la variante con procedimientos, observándose que la importancia de la definición del procedimiento decrecía a mayor experiencia del inspector. En [12] se recomienda usar variantes de lectura más sistematizadas para no depender de la experiencia del inspector. Por otro lado, en el trabajo expuesto en [5] se comprobó, usando inspectores estudiantes, que se detectaba en promedio más defectos usando la lectura basada en procedimientos que la lectura con lista de control, aunque insumía más tiempo. Son casi nulos los trabajos que analizan la efectividad de la lectura constructiva, siendo pocos los que comparan en un mismo experimento las variantes

existentes; esto permitiría tener mejor conocimiento de ellas al requerirse su uso.

### 3 Inspecciones del Modelo Léxico

Los modelos en LN, aunque son fáciles de construir y colaboran con la elicitación, validación y negociación, presentan habitualmente omisiones y ambigüedades [4,13]. En un estudio empírico [13] se observó que el 50% de los casos de uso contenía ambigüedades. En [14] se estimaron omisiones superiores al 57% en 9 conjuntos de escenarios. Valores similares de omisiones se estimaron sobre el modelo LEL [15].

#### 3.1 El Modelo Léxico Extendido del Lenguaje

El modelo LEL [7] tiene una estructura que aporta más información que un simple glosario, pues cada término o símbolo se describe con una *Noción* (la denotación del término) y un *Impacto* (su connotación). El modelo permite describir sinónimos, homónimos y jerarquías de términos. Se le aplican dos principios: el de circularidad y el de vocabulario mínimo, por los que la definición de cada símbolo se realiza procurando utilizar al máximo el propio vocabulario del léxico.

Los símbolos del LEL se clasifican en: *Sujetos* (entidades activas), *Objetos* (entidades pasivas), *Verbos* (actividades) o *Estados* (condiciones de sujetos, objetos o verbos). Según esta clasificación, sus nociones e impactos tienen una semántica diferente. La Fig. 1 muestra dos símbolos del LEL, cuyos términos subrayados (hipervínculos) identifican otros símbolos del LEL.

Símbolo: PRESTADOR	Tipo: Sujeto	Símbolo: EXPEDIENTE	Tipo: Objeto
<b>Noción:</b> <ul style="list-style-type: none"> <li>Es el médico de una <a href="#">prestadora</a> para controlar la <a href="#">facturación</a> proveniente del <a href="#">Instituto de Salud</a>.</li> </ul> <b>Impacto:</b> <ul style="list-style-type: none"> <li><a href="#">Audita expedientes</a> de los <a href="#">pacientes</a> para <a href="#">autorizar el pago</a> o <a href="#">rechazar el pago</a> de una <a href="#">prestación</a>.</li> </ul>		<b>Noción:</b> <ul style="list-style-type: none"> <li>Es el conjunto de información de un <a href="#">paciente</a> que genera el <a href="#">Instituto</a>.</li> <li>Contiene la <a href="#">historia clínica</a>, <a href="#">facturas</a>, <a href="#">estudios</a> e <a href="#">imágenes DICOM</a>.</li> </ul> <b>Impacto:</b> <ul style="list-style-type: none"> <li>Se envía al <a href="#">prestador</a> para <a href="#">auditar</a>.</li> </ul>	

Fig. 1. Ejemplos de símbolos del LEL

#### 3.2 Variantes de Inspección para el Modelo Léxico

En las variantes bajo estudio, se sigue el proceso de inspección propuesto por Fagan [9] con algunas adaptaciones para modelos en LN [4], como el LEL. No se realiza la etapa Visión General, dado que estos modelos deberían comprenderse por sí mismos sin requerir una descripción de los autores, con lo cual se evita la parcialidad de los autores, logrando mayor independencia entre inspectores y autores. La etapa Preparación se focaliza en detectar defectos, en lugar de solo una lectura del material. Esto lleva a que, en la Reunión de Inspección, los inspectores presenten los defectos hallados previamente y los autores los acepten o rechacen.

La categorización de defectos, aplicable a modelos en LN, es: *Discrepancias* (inconsistencias), *Errores* (elementos incorrectos), *Omisiones* (elementos faltantes) y

*Ambigüedades* (elementos entendidos de formas diferentes o no entendibles). Cada defecto es calificado por su grado de severidad en: *Alto, Medio* o *Bajo*.

Se utilizaron las siguientes variantes para inspeccionar el modelo LEL: i) Lectura con lista de comprobación: *Variante usando CheckList*; ii) Lectura con procedimientos: *Variante basada en Formularios* [16]; y iii) Lectura constructiva: *Variante basada en la construcción de Mapas Conceptuales* [17].

En las pautas de cada variante, se especifica el tipo de defecto a capturar y su nivel de severidad. En la Preparación, el inspector aplica las pautas especificadas y genera un reporte con el detalle de cada defecto encontrado con su tipo y severidad, midiendo el tiempo insumido. En la Reunión de Inspección, los defectos capturados son confirmados o rechazados, midiéndose el tiempo de la reunión y el número de defectos aceptados por tipo y severidad.

**Variante usando CheckList (VCH).** El inspector usa una lista con 44 ítems de control sobre el LEL durante la Preparación. Cada ítem tiene predefinido el tipo de defecto y su nivel de severidad (ver Fig. 2).

Ítem de Control	Tiempo hh:mm	Cantidad Defectos	Tipo de Defecto	Severi- dad
20. Verificar que toda mención a un símbolo en las descripciones de otros símbolos esté identificada (referencia no identificada)	00:15	4	Omisión	Media
24. Verificar que los símbolos estén descriptos con oraciones atómicas, concretas y directas (Un solo verbo en cada oración de la noción y del impacto)	00:08	3	Ambigüedad	Media
29. Verificar que todo símbolo Sujeto represente una entidad activa: rol, persona, área u otra que realice alguna acción	00:15	1	Error	Alta
36. Verificar que todo símbolo Objeto describa en su impacto las acciones que se le aplican o que se realizan con él	00:17	2	Omisión	Alta
43. Verificar que se mantenga la coherencia en las descripciones de un símbolo general respecto a sus símbolos especializados (jerarquía de símbolos)	00:20	0	Discrepancia	Media

Fig. 2. Ejemplo de una lista parcial de ítems de control

**Variante basada en Formularios (VFR).** El inspector completa 12 formularios siguiendo la guía de cada formulario. Las guías indican el objetivo del formulario, cómo llenarlo y cómo analizar los datos volcados para identificar defectos. Cada guía estipula el tipo y severidad de los defectos que permite detectar (ver Fig. 3).

FORMULARIO V: VERIFICACIÓN DE SÍMBOLOS TIPO OBJETO							
NOCIÓN:		definición del objeto: qué representa, sus características / otros objetos con los que se relaciona					
IMPACTO:		acciones que pueden ser aplicadas al objeto o realizadas con él					
Nº Símbolo	Tipo Erróneo	NOCIÓN			IMPACTO		
		SOBRANTE	¿Dónde va?	FALTANTE	SOBRANTE	¿Dónde va?	FALTANTE
4	NO	NO		NO	NO		NO
7	NO	NO		NO	"El prestador..., verifica la FACTURA EMITIDA y aprueba el pago."	No es del símbolo	NO
5	NO	NO		NO	"Contiene el detalle de cada FACTURA EMITIDA."	Va en Noción	NO
Tiempo insumido (hh:mm): 0:33							
GUÍA FORMULARIO V: VERIFICACIÓN DE SÍMBOLOS TIPO OBJETO							
Objetivo: detectar si el tipo Objeto asignado al símbolo es correcto y si su noción e impacto coincide con el contenido especificado para un símbolo del tipo Objeto.							
Pasos:							
<ul style="list-style-type: none"> <li>◆ Registrar en la columna SÍMBOLO el número de símbolo a analizar.</li> <li>◆ Registrar en la columna TIPO ERRÓNEO si el símbolo está bien o mal clasificado teniendo en cuenta la definición del tipo Objeto. En caso de mal clasificado, registrarlo en el Formulario XIII.</li> <li>◆ Determinar si existe alguna parte de la noción del símbolo que sobre o falte con respecto a la definición del tipo Objeto. Si se encontrara alguno, mencionar en la columna NOCIÓN en columna SOBRANTE o FALTANTE la porción de texto correspondiente.</li> <li>◆ .....</li> </ul>							
Análisis:							
En el caso de encontrar un Tipo Erróneo (Error / Alta) puede indicar un defecto en la clasificación del símbolo o que la definición (noción e impacto) no se corresponde con el tipo asignado. Cuando se detectan faltantes, es un indicativo de símbolos incompletos según la definición del tipo (Omisión / Alta). Cuando se detectan sobrantes que corresponden al otro componente del mismo símbolo, se trata de ...							

Fig. 3. Ejemplo de un formulario y su procedimiento

**Variante construyendo Mapas Conceptuales (VMC).** El inspector construye un mapa conceptual por cada símbolo del LEL, para luego analizarlos a través de sus conceptos y relaciones, siguiendo 13 pasos para detectar distintos defectos. Cada paso indica su objetivo, cómo identificar defectos en los mapas, el tipo de defecto y su severidad, y brinda recomendaciones para la corrección del LEL (ver Fig. 4).

PASO 1: Detectar sub-grafos inconexos en un mapa conceptual		
Acción de Detección	Tipo de Defecto	Severidad
Buscar en el mapa conceptual sub-grafos sin relación con el Concepto Raíz Símbolo.	Omisión	Baja
<b>Defecto en el LEL</b>	<b>Ejemplo:</b>	
Detectar frases gramaticales omitidas en la noción o impacto que conectan con el símbolo bajo estudio.		
<b>Corrección en el Mapa</b>		
1. Establecer una relación entre el Concepto Raíz Símbolo y el concepto desconectado, ya sea un Concepto Secundario Símbolo o un Concepto no Símbolo. 2. De corresponder, reescribir las relaciones derivadas del Concepto Secundario Símbolo o del Concepto no Símbolo.		
<b>Tratamiento sugerido en el LEL</b>		
Reescribir la oración en la noción o impacto del símbolo bajo estudio en función del mapa conceptual corregido.		

Fig. 4. Ejemplo de un paso de análisis y un mapa conceptual con el defecto a buscar

## 4 Diseño del Experimento

Se realizó un estudio empírico de las tres variantes de inspección antes descriptas, aplicadas sobre el modelo léxico, con el fin de poder establecer qué variantes pueden ser más eficaces para capturar determinados defectos y qué esfuerzo requieren.

En este estudio se utilizaron 5 glosarios distintos, los que habían sido construidos en contextos de sistemas de información en organizaciones argentinas, aunque desarrollados como parte de un curso de grado. Cada glosario fue inspeccionado usando cada una de las 3 variantes de inspección. A cada inspector se le asignó un glosario y una variante de inspección a aplicar. Los inspectores eran estudiantes de grado, independientes de los autores del LEL.

Tanto en la Preparación como en la Reunión de cada inspección, el inspector tomó datos detallados de tiempos, cantidad de defectos, tipos de defectos y severidad, por cada ítem de control, formulario o paso según la variante de inspección que se estaba utilizando. En cada Reunión, participaron el inspector, quien cumplía también el rol de escriba, los autores del LEL y el moderador, quien era un docente del curso. En la Tabla 1 se exponen los datos de cada inspección referidos a defectos detectados en la Preparación, confirmados en la Reunión, el grado de confirmación, defectos confirmados por tipo de defecto y severidad, y el tiempo total insumido.

Respecto a las amenazas a la validez interna del experimento, los inspectores seleccionados eran estudiantes de grado del curso Ingeniería de Requisitos de distintas universidades y ciclos lectivos. Todos los inspectores tenían conocimiento y práctica previo sobre las características y el proceso de construcción del modelo LEL. Los LELs utilizados se crearon con anterioridad al experimento, por otros estudiantes, siguiendo el mismo proceso, independiente del experimento. Los inspectores tuvieron una capacitación de 2 horas sobre la técnica de inspección y la variante a aplicar, y recibieron la documentación necesaria para aplicar la técnica y registrar los datos

necesarios. En las pautas de las 3 variantes, estaba estipulado el tipo y nivel de severidad para cada defecto a capturar, evitando la interpretación del inspector y, por ende, el desvío en las métricas a utilizar para el estudio comparativo. Las 3 variantes habían sido probadas previamente en otros LELs y calibradas para calificar de la misma manera cada defecto.

**Tabla 1.** Defectos capturados y tiempo por inspección

LEL #	Variante de Inspección	Cantidad Símbolos	Tiempo HH:MM	Defectos en Preparación	Defectos Confirmados	Confirmados / Preparación	TIPO DE DEFECTO				SEVERIDAD		
							Discrep.	Error	Omisión	Ambig.	Alta	Media	Baja
1	Checklist	22	09:29	48	22	45,8%	0	7	5	10	0	14	8
1	Formularios	22	10:39	27	17	63,0%	0	3	14	0	1	14	2
1	Mapas	22	06:55	68	50	73,5%	0	0	25	25	14	36	0
2	Checklist	27	04:01	50	49	98,0%	0	14	22	13	18	27	4
2	Formularios	27	07:12	57	52	91,2%	0	4	37	11	1	38	13
2	Mapas	27	09:37	91	70	76,9%	0	2	41	27	1	61	8
3	Checklist	43	03:36	127	94	74,0%	2	27	51	14	9	33	52
3	Formularios	43	06:14	115	92	80,0%	4	13	73	2	23	28	41
3	Mapas	43	15:45	90	81	90,0%	0	0	59	22	12	37	32
4	Checklist	27	03:40	66	48	72,7%	0	5	13	30	33	10	5
4	Formularios	27	10:35	24	25	104,2%	1	2	17	5	9	2	14
4	Mapas	27	05:47	57	62	108,8%	0	0	35	27	21	14	27
5	Checklist	30	05:39	44	31	70,5%	0	2	22	7	2	27	2
5	Formularios	30	05:14	45	42	93,3%	0	1	24	17	6	12	24
5	Mapas	30	10:56	54	48	88,9%	0	0	25	23	5	41	2

Respecto a las amenazas a la validez externa, los sujetos del experimento son solo representativos de inspectores sin experiencia en inspecciones y con escasa experiencia en la IR. Esto implica que los resultados obtenidos no son representativos de las inspecciones en la industria, pero pueden ser válidas si solo se dispone de inspectores novatos. El material inspeccionado eran LELs elaborados, a su vez, por estudiantes sin experiencia en su construcción, por lo que los resultados pueden no ser representativos de la cantidad de defectos reales que pueden aparecer en este tipo de modelo construido por profesionales con experiencia en IR. En cuanto al tamaño de los LELs (22 a 43 símbolos) corresponden a proyectos de software pequeños a medianos, típicos de pymes. El proceso de inspección utilizado fue propuesto y evaluado en la literatura [2,4,16]. El número de inspecciones analizadas (15) indica una muestra pequeña para poder generalizar algunos resultados. En base a estas consideraciones, se asume que este experimento podría ser replicado, obteniendo nuevos resultados que amplíen la muestra presentada en este trabajo, permitiendo alcanzar resultados más precisos o ajustados a diferentes realidades de la industria.

## 5 Análisis de Resultados

En un análisis descriptivo de los resultados generales obtenidos de las inspecciones, se observa que la VMC aunque requiere más tiempo por símbolo (19 min) es la que detecta en promedio más defectos por símbolo (2,13), mientras que por el contrario la VFR, que lleva un esfuerzo en tiempo un poco menor (17 min) que la VMC, detecta a su vez un número mucho menor de defectos por símbolo (1,43) (ver Fig. 5-A y B). Esto hace evidente que en promedio el tiempo para detectar un defecto en la VFR es 1,7 veces superior que en las otras variantes (ver Fig. 5-C), aunque entre esas 2 variantes la VMC detecta más defectos por símbolo que la VCH (ver Fig. 5-A). En

promedio, la VMC insume 1,6 veces más tiempo por símbolo que la VCH y apenas 1,1 más que la VFR (ver Fig. 5-B), lo que debería tenerse en cuenta al establecer tiempos a dedicar a una inspección.

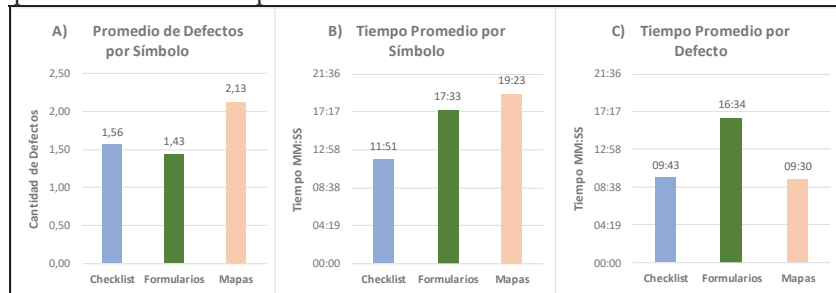


Fig. 5. Valores promedio de defectos y tiempos por variante de inspección

En particular, en 3 inspecciones la VMC llevó más tiempo de inspección que las otras 2 variantes, mientras que la VCH insumió menos tiempo en 3 inspecciones (ver Fig. 6-A), e insumió también menos tiempo por defecto en las mismas 3 inspecciones (ver Fig. 6-B). Sin embargo, debe notarse que tanto la VCH como la VFR presentan una dispersión excesivamente alta en la tasa promedio de tiempo requerido por símbolo (coeficientes de variación: 0,98 y 0,87 respectivamente), y también alta dispersión en la tasa promedio de tiempo por defecto (coeficientes de variación: 0,69 y 0,49 respectivamente). Esto se visualiza claramente en la Fig. 6, donde se muestran dichas tasas para cada inspección. Estas dispersiones, en cierta medida, debilitan las observaciones previas. Cabe mencionar que la VMC, por el contrario, tuvo para ambos tipos de tasas coeficientes de variación bajos; en la Fig. 6 se observan tasas de tiempo relativamente similares en 4 de las 5 inspecciones de VMC. Esto permitiría realizar proyecciones de tiempo según el tamaño del LEL.

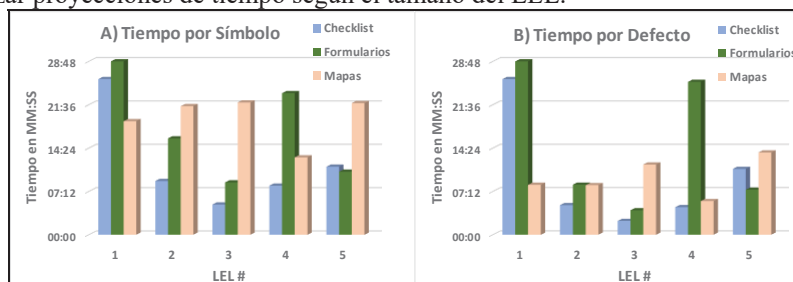


Fig. 6. Tasas de tiempos por LEL inspeccionado

En la Fig. 7-A se exponen los defectos confirmados en la Reunión de Inspección, donde se observa que en 4 de los 5 LELs la VMC tuvo las mayores tasas de detección por símbolo, llegando a duplicar dicha tasa en 2 LELs. Todas las inspecciones de la VMC superaron la tasa de detección del 1.60 defectos por símbolo, mientras que la VFR por el contrario tuvo 3 inspecciones por debajo de esa tasa y la VCH tuvo 2 inspecciones por debajo.

Por otro lado, las 3 variantes presentan una buena tasa promedio de defectos confirmados respecto a defectos capturados en la Preparación (con bajos coeficientes de variación): en la VFR y la VMC los autores confirmaron en promedio más del 86%

de los defectos, y en la VCH confirmaron en promedio el 72%. Analizando cada inspección, en 4 LELs el porcentaje de defectos confirmados respecto a capturados en a Preparación fue superior para VFR y VMC (ver Fig.7-B). Esto indica que VCH podría ser proclive a levantar defectos que no son tales, pudiendo considerarse que las otras 2 variantes son más precisas en guiar hacia la detección de defectos reales.

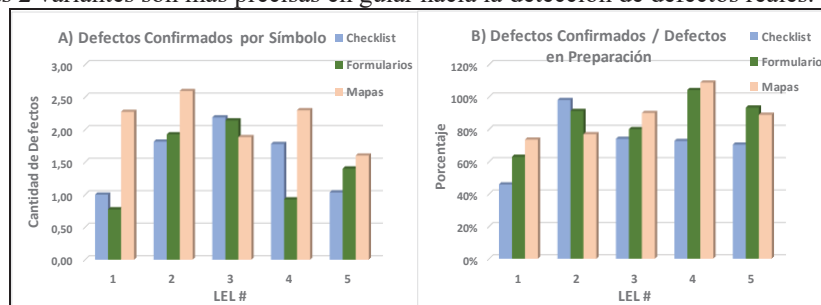


Fig. 7. Tasas de defectos por LEL inspeccionado

Otra métrica utilizada es la tasa de defectos detectados por símbolo según su nivel de severidad. Se observa una gran variación en el número de defectos por severidad entre los LELs y respecto de las variantes (ver Fig. 8), principalmente para defectos de alta y baja severidad. Para severidad media (ver Fig. 8-B), la VMC superó a las otras variantes en los 5 LELs, duplicando la tasa de detección en 2 de ellos y detectando 1,5 veces más defectos en otros 2 LELs.

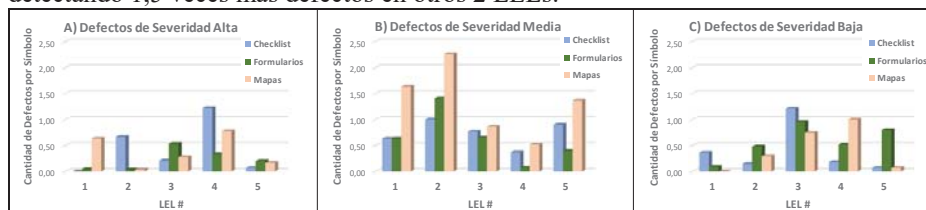


Fig. 8. Defectos confirmados por símbolo según nivel de severidad

Las 3 variantes están diseñadas para detectar omisiones y ambigüedades principalmente, en menor medida errores y solo algunos tipos de discrepancias, debido a las características del LEL y a qué defectos pueden capturarse en una verificación. Es por ello, que en los 5 LELs, las tasas de discrepancias y de errores detectados son muy bajas. Solo 2 inspecciones de la VFR y 1 inspección de la VCH detectaron Discrepancias (tasas  $\leq 0.09$  defectos por símbolo). La VCH fue la variante que logró tasas más altas de detección de Errores en los 5 LELs, duplicando (4 LELs) y hasta triplicando (1 LEL) el número de errores respecto a la VFR, siendo que la VMC solo detectó errores en un LEL y con una tasa muy baja (ver Fig. 9-A). La VMC detectó más omisiones que las otras variantes en 4 LELs, capturando en promedio 1,23 omisiones por símbolo. La VCH y la VFR tuvieron una variación significativa en sus tasas de detección respecto a los 5 LELs, siendo la VCH la que detectó la menor cantidad de omisiones en los 5 LELs (ver Fig. 9-B). Respecto a ambigüedades, al igual que en omisiones, la VMC detectó más cantidad de ambigüedades que las otras variantes en 4 LELs, capturando en promedio 0,88 por símbolo, duplicando la tasa en 3 LELs respecto a la VCH, y siendo la VFR la que detectó menos ambigüedades en los 5 LELs (ver Fig. 9-C). También en este tipo de



defecto la VCH y la VFR tuvieron una variación significativa en su tasa de detección. En resumen, se observa mejor posicionada a la VMC para la detección de omisiones y ambigüedades, prestando menos atención en capturar discrepancias y errores. Las otras variantes están más preparadas para identificar estos dos tipos de defectos, en desmedro de omisiones y ambigüedades.

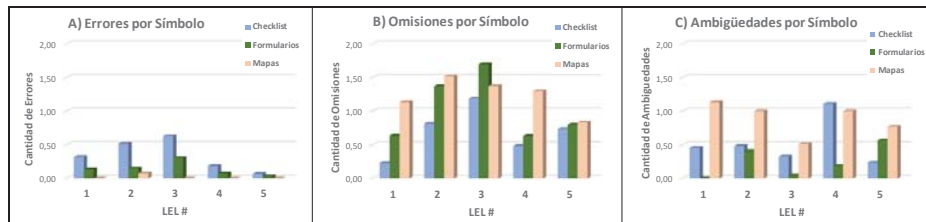


Fig. 9. Tipos de defectos confirmados por símbolo

## 6 Conclusiones

Es conocido que las inspecciones de software detectan mayor cantidad de defectos que las pruebas, aunque insumen más tiempo [10,11], también se ha probado su eficacia en modelos de la IR [1,3,5,6], proponiéndose distintas maneras de captura defectos. Es por ello que se ha llevado a cabo un experimento para establecer la efectividad de 3 variantes de inspección utilizando 5 léxicos.

En base al análisis de los resultados se observa que la variante basada en lectura constructiva (creando mapas conceptuales) logra tasas de detección de defectos muy superiores a las otras dos variantes (lectura con checklist y lectura basada en procedimientos), pero insumiendo más tiempo, mientras que la lectura con checklist lleva en promedio menos tiempo, pero captura menos defectos. Esto corrobora los resultados presentados en [2,5] sobre VCH y VFR. Por otro lado, debe notarse que en el caso de la lectura con checklist y la basada en procedimientos, los tiempos de inspección tuvieron alta dispersión respecto de los 5 LELs en este estudio empírico acotado, dado lo cual resulta difícil poder estimar tiempos a insumir en una inspección de este modelo para cualquiera de estas 2 variantes. Las 3 variantes tuvieron buenas tasas de confirmación de defectos en la reunión de inspección, siendo la variante con checklist la que en 4 LELs tuvo más rechazos que las otras 2, lo cual estaría indicando que brinda menos apoyo para identificar defectos. Respecto a la severidad de defectos, se observa una alta disparidad entre los 5 LELs y las 3 variantes utilizadas; solo se puede señalar que la variante constructiva tuvo tasas muy superiores de detección de defectos de mediana severidad frente a las otras. Se observa que la variante con checklist estaría más preparada para detectar errores, mientras que la variante constructiva estaría más enfocada en detectar omisiones y ambigüedades frente a las otras variantes. Estas observaciones están basadas en evidencia empírica proveniente de una muestra pequeña de 5 inspecciones por variante, realizadas por inspectores novatos sobre los mismos 5 LELs, creados también por novatos, por lo que estos resultados solo pueden dar indicios de cómo podrían capturarse ciertos defectos de una manera más eficaz, teniendo inspectores inexpertos y considerando los tiempos disponibles en un dado proyecto de software.

Para precisar los resultados obtenidos, se prevé replicar este experimento sobre LELs de mayor tamaño, y extender el experimento al modelo de escenarios o casos de uso con las variantes estudiadas en este trabajo. Aunque existen estudios sobre el impacto de la experiencia del inspector en la efectividad de algunas variantes [2,3,6], casi no se ha estudiado para la lectura constructiva, lo cual es un camino a seguir.

## Referencias

1. Regnell, B., Runeson, P., Thelin, T.: Are the perspectives really different? Further experimentation on scenario-based reading of requirements. *Empirical Software Engineering*, 5(4), 331-356 (2000).
2. Porter, A.A., Votta, L.G.: Comparing Detection Methods for Software Requirements Inspections: A Replication Using Professional Subjects. *Empirical Software Engineering*, 3(4), 355-380 (1998).
3. Albayrak, Ö., Carver, J.C.: Investigation of individual factors impacting the effectiveness of requirements inspections: a replicated experiment. *Empirical Software Engineering*, 19(1), 241-266 (2014).
4. Leite, J.C.S.P., Doorn, J.H., Hadad, G.D.S., Kaplan, G.N.: Scenario Inspections, *Requirements Engineering Journal*, 10(1), 1-21. Springer-Verlag, Londres (2005).
5. Paech, B., Denger, C., Kerkow, D., Von Knethen, A.: Requirements Engineering for Technical Products: Integrating Specification, Validation and Change Management. En: Maté Silva (eds) *Requirements Engineering for Sociotechnical Systems*, IGI-Global (2005).
6. Armijos, A.F., Monsalve, C., Ullón, R.H., Maya, R.D., Romero, J.A.: Calidad en la inspección de requerimientos de software: una propuesta de inspección. En: XIX Ibero-American Conference on Software Engineering, Quito, Ecuador (2016).
7. Leite, J.C.S.P., Doorn, J.H., Kaplan, G.N., Hadad, G.D.S., Ridaio, M.N.: Defining System Context using Scenarios. En: J.C.S.P. Leite, J.H. Doorn (eds.) *Perspectives on Software Requirements*, pp. 169-199. Kluwer Academic Publishers, Springer US (2004).
8. Antonelli, L., Rossi, G., Leite, J.C.S.P., Oliveros, A.: Deriving requirements specifications from the application domain language captured by Language Extended Lexicon. En: 15th Workshop on Requirements Engineering, Buenos Aires (2012).
9. Fagan, M.E.: Design and Code Inspections to reduce Errors in Program Development, *IBM Systems Journal*, 15(3), 182-211 (1976).
10. Laitenberger, O., Debaud, J.M.: An Encompassing Life-Cycle Centric Survey of Software Inspection. *Journal of Systems and Software*, 50(1), 5-31 (2000).
11. Jones, C.: Measuring defect potentials and defect removal efficiency. *CrossTalk, The Journal of Defense Software Engineering*, 21(6), 11-13 (2008).
12. Ciolkowski C., Laitenberger, O., Biffel, S.: Software reviews: The state of the practice, *IEEE software*, 20(6), 46-51 (2003).
13. Ben Achour, C., Rolland, C., Maiden, N.A.M., Souveyet, C.: Guiding Use Case Authoring: Results of an Empirical Study. En: *International Symposium on Requirements Engineering*, Limerick, Irlanda, IEEE Computer Society Press, pp. 36-43 (1999).
14. Ridaio, M., Doorn, J.H.: Estimación de Completitud en Modelos de Requisitos Basados en Lenguaje Natural. En: 9th Workshop on Requirements Engineering, pp.151-158 (2006).
15. Hadad, G.D., Litvak, C., Doorn, J.: Problemas y Soluciones en la Completitud de Modelos en Lenguaje Natural. En: II Congreso Argentino de Ingeniería, Tucumán, T366 (2014).
16. Kaplan, G., Hadad, G., Doorn, J., Leite, J.: Inspección del Léxico Extendido del Lenguaje. En: 3rd Workshop on Requirements Engineering, Río de Janeiro, Brazil, pp.70-91 (2000).
17. Sebastián, A., Hadad, G.: Mejoras a un Modelo Léxico mediante Mapas Conceptuales. En: XXI Congreso Argentino de Ciencia de la Computación, Buenos Aires, pp.526-535 (2015).