

CIAM Extendido con Generación Automática de IU Frente a Metodologías no Guiadas: Evaluación de una Experiencia con COFARCIR

Maximiliano Paredes Velasco Grupo de Investigación LITE Universidad Rey Juan Carlos 28933 – Móstoles (Madrid) maximiliano.paredes@urjc.es	Ana I. Molina Grupo de Investigación CHICO Universidad de Castilla-La Mancha 13071 - Ciudad Real AnaIsabel.Molina@uclm.es	Miguel A. Redondo Grupo de Investigación CHICO Universidad de Castilla-La Mancha 13071 - Ciudad Real miguel.redondo@uclm.es	José Julio Gonzalez Departamento de Informática COFARCIR Ciudad Real	Laura Díaz Grupo de Investigación LITE Universidad Rey Juan Carlos 28933 – Móstoles (Madrid)	Khalid Ezzaid Grupo de Investigación LITE Universidad Rey Juan Carlos 28933 – Móstoles (Madrid)
--	---	---	--	---	--

Resumen

Las actuales propuestas para diseñar aplicaciones que soporten trabajo en grupo no asisten totalmente a los ingenieros del software. Se echan en falta notaciones y metodologías que faciliten este proceso. Con este objetivo surge CIAM (*Collaborative Interactive Applications Methodology*), que constituye una metodología y varias notaciones para el diseño y desarrollo de interfaces colaborativas. En este artículo nos proponemos validar si incorporando a CIAM procesos de generación automática de IU (*Interfaz de Usuario*) se experimenta mejora en el diseño de interfaces de usuario colaborativas. Para ello hemos aplicado, a un mismo caso de estudio, la metodología CIAM con generación automática de IU y una metodología de diseño no sistemática, evaluando las respectivas interfaces de usuario generadas. Como resultado de esta evaluación hemos obtenido que las interfaces generadas con CIAM soportan todos los requisitos de interacción colaborativa y que el grado de satisfacción del usuario es elevado, no siendo así en los interfaces generados con la otra metodología de diseño. De estos resultados se desprende que CIAM extendido con métodos de generación automática garantiza que los requisitos de interacción y colaboración son satisfechos en las interfaces obtenidas.

1. Introducción

El diseño de sistemas para el soporte del trabajo en grupo resulta incompleto [4], no existiendo propuestas asentadas que permitan asistir a los ingenieros del software en el desarrollo de este tipo de aplicaciones. Habitualmente, el desarrollo de este tipo de sistemas se suele realizar *ad-hoc*. Sin embargo, resulta deseable contar con notaciones y metodologías que ayuden en este proceso y permitan obtener productos de una mayor calidad. En particular, uno de los aspectos en los que centramos nuestro interés es el desarrollo de las Interfaces de Usuario (IU) para aplicaciones *groupware* (aplicaciones con soporte de trabajo en grupo), las cuales presentan ciertas peculiaridades que las diferencian de las IUs convencionales [1], como que deben implementar técnicas *awareness* o soportar discusión en grupo. Hay varios trabajos en el campo de desarrollo de este tipo de aplicaciones, como la notación Action Port Model (APM) [2] o la propuesta Proclerts [12]. Otros trabajos constituyen extensiones de la notación UML como son COMO-UML [3] and UML-G [11]. En cualquier caso se echan en falta propuestas que contemplen conjuntamente aspectos interactivos y de trabajo en grupo con un planteamiento con mayor grado metodológico. Con el objetivo de proponer una metodología, así como un conjunto de notaciones para especificar los aspectos relativos a la interacción y la colaboración en sistemas de soporte al trabajo en grupo surge CIAM [5,8].

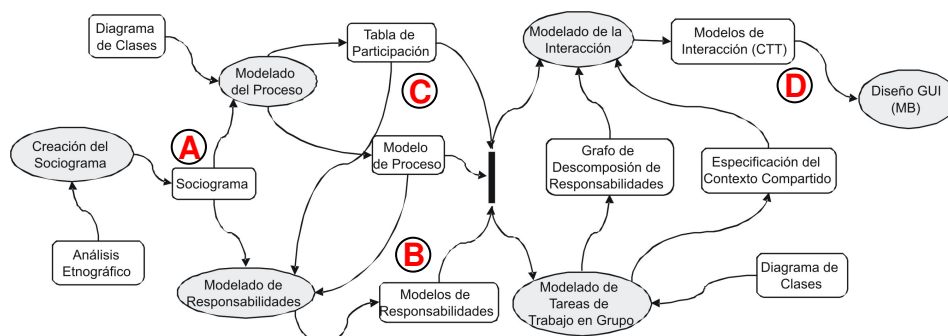


Figura 1. Etapas de CIAM y especificaciones que genera

Consta de una serie de fases (Figura 1), en cada una de las cuales se especifican distintos aspectos del sistema final. En primer lugar se modela, mediante el llamado *sociograma* (Figura 1, marca A), la organización (roles, actores, grupos,...) para la que se está diseñando el sistema. A continuación, se indican las responsabilidades de cada uno de sus miembros, mediante el llamado *Modelo de Responsabilidades* (Figura 1, marca B). Seguidamente se especifican aspectos relacionados con la estructura del trabajo a realizar en la organización antes descrita, el flujo de las tareas que lo forman y las restricciones (de orden, de datos, temporales, etc.) impuestas por el trabajo en grupo. Estos aspectos quedan expresados en forma de tablas y grafos (*Tabla de Participación* y *Modelo de Proceso*), ver Figura 1 marca C. Una vez identificadas las tareas del sistema, éstas se tipifican en individuales y grupales, pudiendo, éstas últimas ser a su vez clasificadas en cooperativas y colaborativas, permitiendo CIAM el modelado diferencial de cada uno de estos tipos. Como última etapa, y a partir de las especificaciones anteriores, se obtienen de forma automática modelos ConcurTaskTree (CTT) [6,9] del sistema (Figura 1, marca D), que pueden servir de entrada a procesos de generación de IU [7]. En trabajos posteriores hemos extendido CIAM con procesos computacionales para la generación automática de IUs a partir de los modelos CTT que se generan en la última etapa (para más información sobre el proceso de generación consultar [7]).

Nuestro objetivo en este artículo es comprobar si ampliando CIAM con procesos de generación automática se experimenta mejora significativa en el proceso de diseño de las interfaces de usuario que se obtienen como resultado de su aplicación, frente a métodos no sistemáticos de diseño y

desarrollo de interfaces de usuario. En particular, la hipótesis a contrastar en este trabajo es que: “*CIAM extendido con técnicas y procesos de generación automática de interfaces de usuario para aplicaciones colaborativas produce interfaces finales que satisfacen y soportan mejor los requisitos de colaboración entre usuarios*”. Con el objetivo de contrastar esta hipótesis se ha aplicado CIAM a un caso de estudio real en el contexto de las necesidades de una empresa específica. En las siguientes secciones se describirá dicho caso de estudio, se mostrará la aplicación de CIAM al mismo y se compararán las interfaces de usuario obtenidas como resultado de aplicar CIAM frente a un diseño de IU que no sigue ninguna metodología de diseño y desarrollo. Finalmente se indicarán las conclusiones que se extraen del presente trabajo, así como las líneas de continuación que se derivan del mismo.

2. Descripción de la experiencia

Para contrastar la hipótesis de trabajo hemos realizado una experiencia en la que se evalúan dos diseños diferentes (guiados con diferentes metodologías) de una aplicación de soporte a trabajo en grupo para el caso de estudio real con la empresa COFARCIR (Cooperativa Farmacéutica de Ciudad Real).

2.1. Caso de estudio: COFARCIR

La compañía COFARCIR gestiona la distribución de productos farmacéuticos procedentes de laboratorios y grandes proveedores a farmacias y pequeños comercios. Uno de los departamentos de la empresa se encarga de gestionar el *proceso de recepción y almacenamiento de productos*

Nombre Requisito	Descripción	Datos
Registro de entrada	El Operario de Control de Entrada asigna un número de entrada al producto recibido. Se genera una prioridad inicial	<u>RegistroEntrada</u> Numero_Entrada Proveedor
Generación de la orden de colocación	El Operario de Compras y el Jefe de Almacén, según el nivel de stock y las entradas pendientes, generan la orden de colocación	<u>Orden_Colocacion</u> stock Lista_Pedidos_urgentes Orden_colocacion
Colocación de mercancía	Se identifican cada uno de los Operadores de Colocación y empiezan a colocar la mercancía según la prioridad. Se actualiza el stock en cada artículo medido finalizado. Al finalizar se generan avisos/incidencias	<u>Identificacion</u> Login Password <u>Lista_Articulos_pendientes</u> Lista_articulos <u>Nuevo/Modificar_Articulo</u> Nº_Articulo Categoria Cantidad
Gestión de incidencias	El jefe de Compras consulta los avisos/incidencias y genera informes	<u>Lista_Incidencias</u> Avisos/Incidencias <u>Generar_Report</u> Nº_Report Text_Report

Tabla 1. Requisitos del proceso de recepción y colocación de mercancías en COFARCIR.

farmacéuticos, proceso que comienza cuando la agencia de transporte entrega un pedido de productos procedente de algún proveedor. En ese momento el operario de control de entrada registra la entrada de estos productos e informa al operario de compras. El sistema automáticamente marca una prioridad para esta recepción (principalmente dependiendo de aspectos como el nivel de stock de los artículos incluidos en el pedido): Urgente, Prioritario o Normal.

El operario de compras o el jefe de almacén pueden cambiar esta prioridad según su criterio. Una vez registrada la entrada de nuevos productos a almacenar, los operarios de colocación cogen un terminal móvil y permanecen disponibles para que se les asignen una orden de colocación (según la prioridad de las entradas pendientes). Cuando un operario de colocación recibe una orden de colocación, éste debe colocar la mercancía recibida en el almacén. Esto lo puede realizar de manera manual o automáticamente ayudado de sistemas de clasificación automático tipo Kardex¹ o Knapp² [10]. Una vez finalizada su tarea, el operario de colocación marca como finalizada su orden de colocación, generando automáticamente avisos/alertas al jefe de compras indicando que ha

terminado dicha colocación e indicando las posibles incidencias del pedido. La Tabla 1 resume los requisitos identificados en este proceso.

2.2. Planteamiento de la experiencia

En el marco de nuestra hipótesis de trabajo debemos comprobar que CIAM garantiza que los requisitos de interacción y colaboración son satisfechos en las interfaces de usuario obtenidas al aplicar dicho proceso de diseño. Esta hipótesis la vamos a validar mediante la aplicación y comparación de los resultados obtenidos al aplicar dos métodos de diseño diferentes: (a) el proceso sistemático propuesto por CIAM y (b) un método de diseño no sistemático. Por tanto, partiendo de los requisitos especificados en la Tabla 1 hemos abordado dos diseños: a) Diseño sin CIAM y b) Diseño con CIAM. El diseño a) consiste en no utilizar una metodología tipo sistemática como CIAM. En su lugar hemos utilizado un diseño intuitivo a partir del estudio y análisis de los requisitos. El diseño b) consiste en realizar todo el proceso de diseño guiado por la metodología CIAM. Hemos puesto a trabajar a dos diseñadores, uno en cada diseño, partiendo ambos de los mismos requisitos e información. El objetivo de cada uno de ellos es la creación de un prototipo de la Interfaz de Usuario final, constituyendo estas interfaces las variables a

Tabla 1.

¹ <http://www.kardex.es/es/almacenes-y-logistica.html>

² <http://www.logismarket.es/knapp-iberica/12035005-1124074-c.html>

evaluar en la experiencia. Estos diseñadores son ingenieros de software con experiencia en interfaces de usuario y sin conocimientos previos de CIAM. Al diseñador que utilizó CIAM se le proporcionó previamente una sesión explicativa de la metodología y documentación de la misma. Durante el proceso de ambos diseños hemos incorporado paralelamente métodos de seguimiento y estudio que nos permitiesen valorar cada uno de los diseños y productos obtenidos: *trazabilidad de requisitos* y *grado de satisfacción del usuario final de los prototipos*. La experiencia que hemos realizado constituye un estudio preliminar y exploratorio con un carácter comparativo de las interfaces obtenidas con los dos métodos de diseño.

2.3. Diseño sin aplicar CIAM

Estudiando los requisitos iniciales se identifican al menos tres procesos de trabajo: registrar los productos que llegan, establecer el orden en el que hay que colocarlos y realizar la colocación de los mismos en el almacén. Debido a que hay varios operarios y jefes de sección se deberán realizar autenticación de usuarios frente al sistema. El proceso más interesante para el objeto de este artículo es el de establecer el orden en el que hay que colocar los productos. Centrándonos en el diseño de la IU, en este proceso identificamos tres tareas:

1. Habría que visualizar en la misma IU el nivel de Stock y la lista de pedidos urgentes.
2. El operario de compras o el jefe de almacén cambian, según sus criterios, el orden de colocación.
3. Se genera una lista de prioridad de colocación.

Con estas premisas se ha diseñado una IU para establecer el orden de colocación. La Figura 2 muestra la interfaz resultante (el significado de las marcas que señalan algunas partes de la interfaz se detallan en la Tabla 7 y hacen referencia a diferentes requisitos especificados).

2.4. Diseño con CIAM

En la primera etapa de la metodología CIAM se aborda el modelado de la estructura social de la organización en la que el sistema groupware será implantado. Se identifican los roles y actores que intervienen así como las relaciones entre los mismos. Revisando la especificación de requisitos de la Tabla 1 identificamos los siguientes roles: *Operario de control de entrada*, *Operario de compras*, *Jefe de almacén*, *Jefe de compras* y *Operario de colocación*. Puede haber de uno a varios operarios de control de entrada y varios operarios de compras, mientras que sólo hay un jefe de compras y un jefe de almacén. También puede haber varios operarios de colocación. La Figura 3 muestra el *sociograma* que se genera en

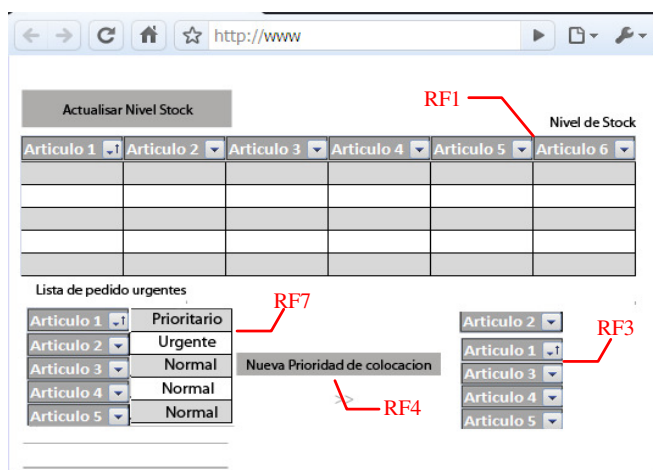


Figura 2. IU resultante sin aplicar CIAM

esta etapa de la metodología. Como podemos ver los roles *Operario de compras* y *Jefe de almacén* forman un equipo de trabajo, por lo que habrá tareas en las que tengan que participar conjuntamente. Hemos creado el rol *Usuario* del cual heredan todos los demás y asume las tareas comunes a todos ellos como por ejemplo validarse ante el sistema.

Siguiendo la metodología, en una segunda etapa abordamos el modelado del Proceso y creamos el Modelo de Responsabilidades. El objetivo de esta etapa es especificar las tareas a alto nivel, identificando los roles que las realizan, los datos con los que trabajan, así como las restricciones que puedan darse en su ejecución, etc. Para realizar este modelo de proceso creamos, por un lado, la Tabla de Participación y por otro los Modelos de Responsabilidades. Finalmente todo ello contribuye a la creación de una representación gráfica del proceso colaborativo a seguir durante el uso de la herramienta.

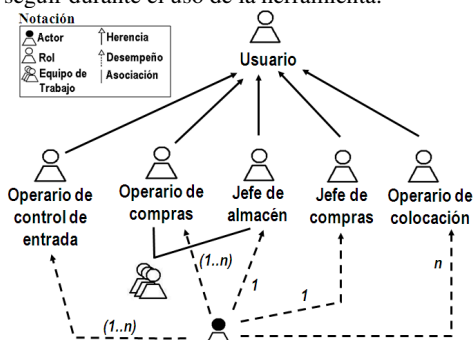


Figura 3. Sociograma que representa la estructura social de la organización

La Tabla de Participación recoge la colaboración e interacción que se produce entre

los miembros del grupo de trabajo, identificando para cada rol especificado en el sociograma las tareas que deben asumir. En algunos casos podremos identificar roles que participan en una misma tarea, y tareas en las que sólo participa un rol. Por tanto, en esta fase analizamos la participación de los roles en cada tarea, indicando el tipo de participación que se da, clasificando la tarea en tres posibles tipos: individual, cooperativa o colaborativa. Todo este análisis se sintetiza en la Tabla de Participación para nuestro caso de estudio. Como podemos observar en esta tabla la tarea *Generación de la orden de colocación* es una tarea colaborativa en la que participan los roles *Operario de compras* y *Jefe de almacén*. El resto de las tareas son tareas individuales. El tipo de la tarea queda expresado mediante el icono gráfico empleado en la última de las columnas de esta tabla.

A continuación nos centramos en las responsabilidades que debe asumir cada rol; para ello definimos el llamado Modelo de Responsabilidad de cada rol. Este modelo depura en detalle la información que se recoge para cada columna de la tabla de participación mostrada anteriormente, añadiendo a cada rol y para cada tarea, información de los objetos del dominio y de los prerequisites (tareas que deben haberse completado antes de ejecutar la actual tarea, y los datos necesarios para su correcta ejecución). Esta información, una vez más, se sintetiza principalmente de manera textual en forma de tabla. Volviendo a nuestro caso de estudio las Tablas de la 3 a la 6 muestran el Modelo de Responsabilidades de los diferentes roles.

Tareas \ Roles	Operario de control de entrada	Operario de compras	Jefe de almacén	Jefe de compras	Operario de colocación	Tipo
Registro de entrada	X					
Generación de la orden de colocación		X	X			
Colocación de mercancía					X	
Gestión de incidencias				X		

Tabla 2. Tabla de Participación

Nombre de la responsabilidad	Tipo de tarea	Objeto del modelo del dominio	Prerrequisitos	
			Tareas	Datos
Registro de entrada		C: Entrada	INI	

Tabla 3. Modelo de Responsabilidad del rol Operario de control de entrada

Nombre de la responsabilidad	Tipo de tarea	Objeto del modelo del dominio	Prerrequisitos	
			Tareas	Datos
Generación de la orden de colocación		L: Entrada C/E: Orden colocación L: Stock	Registros de entrada	Entrada Stock

Tabla 4. Modelo de Responsabilidad del rol Operario de compras y del rol Jefe de compras

Nombre de la responsabilidad	Tipo de tarea	Objeto del modelo del dominio	Prerrequisitos	
			Tareas	Datos
Colocación de mercancía		L: Orden colocación C: Incidencia/aviso E: Stock	Generación de la orden de colocación	Orden de colocación Stock

Tabla 5. Modelo de Responsabilidad del rol Operario de colocación

Nombre de la responsabilidad	Tipo de tarea	Objeto del modelo del dominio	Prerrequisitos	
			Tareas	Datos
Gestión de incidencias		L: Incidencia/aviso	Colocación de mercancía	Incidencia/aviso

Tabla 6. Modelo de Responsabilidad del rol Jefe de compras

Toda esta información, la de los Modelos de Responsabilidades y la Tabla de Participación, sirven para modelar finalmente la colaboración entre los miembros del grupo. En el Modelo de Proceso del sistema que estamos modelando (ver Figura 4) podemos ver que la tarea colaborativa 2, *Generación de la orden de colocación*, lee las entradas y el nivel de stock y crea la Orden de colocación. La tarea individual 3 (*Colocación de mercancías*) no comienza hasta que la tarea 2 no ha creado la orden de colocación. Como podemos ver en el Modelo de Proceso hay tareas individuales y colaborativas. Especial atención para nuestro trabajo merecen las tareas colaborativas. El sistema que estamos modelando tiene solo una tarea colaborativa, la tarea 2. Veamos el modelado de esta tarea. La Tabla 4 muestra esta tarea colaborativa con un nivel de detalle mayor. En la zona central de la Figura 5 B se muestra el contexto compartido y el tipo de visibilidad que hay para los datos que se manejan en el mismo. El objeto Orden_colocación

(contiene la información relacionada con el orden en el que se colocarán los artículos) debe ser tratado en un *área de edición exclusiva*. Esto significa que los dos roles que pueden operar con este objeto (Figura 5 A) pueden acceder de una manera restringida y los cambios que produzcan lo verán ambos roles.

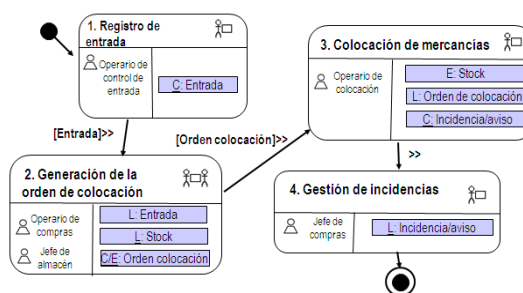


Figura 4. Modelo de Proceso del sistema COFARCIR

En esta etapa se genera, de forma automática, el árbol CTT (Figura 5 C) asociado a la tarea colaborativa especificada. Dicho CTT sirve de entrada a un proceso de generación de la interfaz final. El proceso de obtención de la IU final, a partir de los modelos de datos y tareas, se basa en la aplicación de un *árbol de decisión* [7], el cual, teniendo en cuenta el tipo de tarea y subtarea CTT, cardinalidad y tipo de datos propone el

control o *widget* más apropiado para soportar cada tarea del modelo CTT. Hemos desarrollado un prototipo de herramienta para soportar este proceso que obtienen interfaces de usuario para plataforma .NET. La IU resultante para la tarea *Generación de la orden de colocación* se muestra en la Figura 6 (las marcas sobre la interfaz se explicarán más adelante).

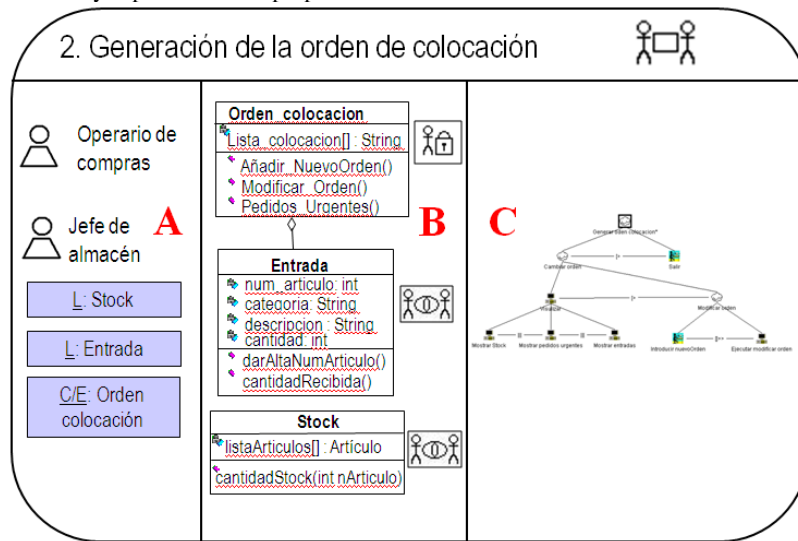


Figura 5. Tarea colaborativa *Generación de la orden de colocación* a un nivel detallado

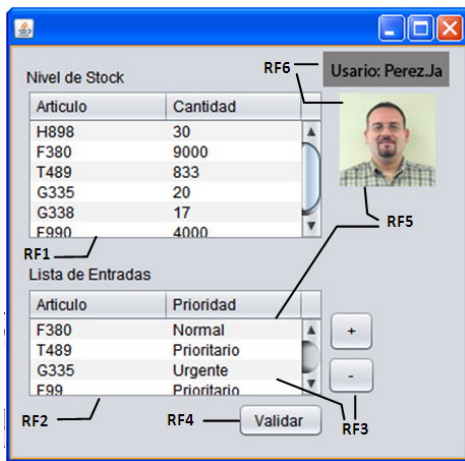


Figura 6. IU resultante de aplicar la metodología CIAM

3. Discusión de los resultados

Para contrastar la hipótesis de trabajo planteada vamos a evaluar las dos interfaces de usuario obtenidas en la experiencia realizada. Esta evaluación la hemos enfocado desde dos dimensiones: seguimiento y confirmación de los requisitos iniciales en el producto final y satisfacción de los usuarios finales. Para evaluar el seguimiento y confirmación de los requisitos hemos realizado una traza de los mismos. De los cuatro requisitos iniciales y genéricos identificados previamente (ver Tabla 1) es de interés, por su carácter colaborativo, el que denominamos *Generación de la orden de colocación*. Este requisito general se descompone en requisitos específicos y más detallados tal y como muestra la Tabla 7. Podemos ver que se identifican tres tipos de requisitos específicos:

- *Visualización colaborativa*: son los requisitos RF1, RF2 y RF7 los cuales exigen la visualización de datos a un grupo de usuarios de manera concurrente.
- *Sensibilidad al grupo*: son los requisitos RF5 y RF6, los cuales exigen de la implementación de mecanismos *awareness*.
- *Procesamiento*: son los requisitos RF3 y RF4, los cuales definen restricciones del dominio para la lógica de la aplicación.

Si estudiamos cómo estos requisitos han ido “evolucionando” hasta terminar generando una parte del IU final, y centrándonos en los requisitos del tipo de sensibilidad al grupo, podemos observar que con la metodología CIAM sí se garantizan la satisfacción de estos requisitos colaborativos y sin embargo con otra metodología no guiada no es así. Cojamos un requisito y hagamos una traza del mismo desde su especificación inicial hasta la generación de la IU final con CIAM. Por ejemplo, la Figura 7 muestra la traza del requisito RF1: *Visualización del stock de artículos al grupo de usuarios*. En esta figura podemos observar cómo los diferentes modelos generados a lo largo de CIAM aportan diferentes aspectos sobre este requisito para generar la IU. El Sociograma y la Tabla de Participación identifican que esta tarea está inmersa en un grupo de trabajo (ver Figura 7, paso 2), con lo cual se especifica el aspecto colaborativo de la misma. Posteriormente el Modelo de Responsabilidad identifica, entre otras que, de acuerdo al requisito, se debe acceder el objeto Stock para su lectura (ver Figura 7, paso 3). El Modelo de tareas colaborativa identifica además que la visualización del objeto Stock debe ser colaborativa (ver Figura 7, paso 4). Finalmente, con estas tres características identificadas: 1) sobre el objeto interactúan un grupo de usuarios, 2) la información a tratar es el stock de los artículos y 3) se debe visualizar esta información a ese grupo de usuarios, se genera en la IU final un *ListView* (definido así por el Árbol de Decisión que permite inferir la IU final) para mostrar el stock de los artículo (ver Figura 7, paso 5).

Como resultado de estas trazas hemos marcado en las interfaces de los diseños los requisitos específicos y los controles generados. En la Figura 6 podemos ver que el requisito RF1 (Visualización del stock de artículos) ha generando un control *ListView* en el IU final, el RF2 (visualización de las entradas) ha generado

un control *ListBox*, el requisito RF3 (modificar el orden de colocación y visualizar nuevo orden) ha generado que el control *ListBox* tenga la columna “Prioridad” editable mediante el control *Spick*, el requisito RF4 (confirmar nuevo orden) ha generado un control *Button*, el requisito RF5 (relacionar usuario/operación y ordenar las contribuciones) ha generado un control *Image* y, en el control *ListBox*, ha generado la columna “Prioridad”. Por último, el requisito RF6 (identificación de usuario) ha generado los controles *Image* y *Label*. De forma similar la Figura 2 muestra la IU del diseño sin CIAM indicando qué controles ha generado cada uno de los requisitos.

A modo resumen de todo este proceso, en la Tabla 7 hemos marcado los requisitos que soportan cada uno de los dos IUs (diseño guiado con CIAM y sin CIAM). Podemos observar que la IU generada con CIAM soporta todos los requisitos (a excepción del RF7 -Visualización de la lista de pedidos urgentes al grupo de usuarios- debido a que el diseñador ha decidido no contemplar este aspecto suprimiendo el mismo del Modelo de Responsabilidades). Hemos de destacar que los requisitos de tipo Sensibilidad al grupo no están soportados por el IU desarrollado sin metodología CIAM (columna *IU sin CIAM*), sin embargo, estos mismos sí están soportados en la IU resultante del diseño con CIAM (columna *IU con CIAM*).

La otra parte de la evaluación de las IUs generadas en la experiencia consiste en analizar la satisfacción del usuario final. Esta evaluación la han realizado dos usuarios familiarizados con la utilización de herramientas software y ajenos a la empresa COFARCIR por lo que en una primera etapa de la evaluación se les explicó la funcionalidad de las interfaces de usuario y el contexto de uso de las mismas. Posteriormente se les entregó la especificación de requisitos. Uno de los usuarios asumió el papel de operario de compras y el otro el de jefe de almacén. En una segunda etapa se le entregó a cada uno los dos prototipos que implementan las interfaces generadas y se les pidió que rellenasen un cuestionario en el que indicaban el grado con que pensaban que cada uno de los requisitos habían sido alcanzado por cada herramienta (valorándolo como bajo, medio o alto). La Tabla 8 sintetiza la comparativa entre las dos IU, indicando para cada control de la IU el grado de satisfacción del

Control IU	IU sin CIAM		IU con CIAM	
	Id. Req.	Grado Satis.	Id. Req.	Grado Satis.
<i>ListView</i>	RF1	*	RF1	***
<i>ArrayList</i>	RF7	**	-	-
<i>Button</i>	RF4	**	RF4	**
<i>ListBox</i>	RF3	*	RF2 RF3 RF5	***
<i>Image</i>	-	-	RF5 RF6	***

Tabla 8. Comparativa de los IU según el grado de satisfacción del usuario (*bajo,**medio,***alto)

4. Conclusiones finales y trabajo futuro

En este artículo hemos presentado una experiencia de uso para la creación de interfaces de usuario colaborativas en la que hemos aplicado la metodología CIAM extendida con métodos de generación automática y una metodología no sistemática. Hemos analizado los resultados de la experiencia evaluando las interfaces obtenidas mediante trazabilidad de requisitos y satisfacción del usuario final. Claramente se observa que con CIAM se satisfacen los requisitos en materia de soporte de colaboración, mientras que con el otro caso no es así. Este hecho viene a confirmar la hipótesis de trabajo propuesta en este artículo, CIAM extendido con generación automática de interfaces produce interfaces de usuario finales que satisfacen y soportan mejor los requisitos de colaboración entre usuarios.

Como futuro trabajo a realizar nos falta aun por valorar el esfuerzo y el coste que conlleva aplicar la metodología CIAM en una experiencia de mayor complejidad y alcance.

Referencias

- [1] Antunes, P. and N. Guimaraes. Multiuser Interface Design in CSCW systems, 1994.
- [2] Carlsen, S. (1998) Action port model: A mixed paradigm conceptual workflow modelling language. Third IFCIS International Conference on Cooperative Information Systems.
- [3] Garrido, J.L., Gea, M., and M.L. Rodríguez, (2005) Requirements Engineering in Cooperative Systems. Requirements Engineering for Sociotechnical Systems. IDEA GROUP, Inc.USA, 226–244.
- [4] Grudin, J. Groupware and social dynamics: eight challenges for developers. Comm. ACM 37(1), pp. 92-105, 1994.
- [5] Molina, A.I., Redondo, M.A., Ortega, M.: A methodological approach for user interface development of collaborative applications: A case study. Science of Computer Programming 74, pp.754-776, 2009.
- [6] Mori, G., Paternò, F., Santoro, C. Design and Development of Multidevice User Interfaces through Multiple Logica IDescriptions. IEEE Transactions on Software Engineering August 2004, pp. 507-520, 2004.
- [7] Paredes, M., Molina, A.I., Giraldo, J., Redondo, M.A., Ortega, M. La Generación Automática de Interfaces de Usuario en el contexto de CIAM. Aplicación en el caso de estudio AULA. Actas X Congreso Internacional de Interacción Persona-Ordenador (Interacción 2009), 2009.
- [8] Paredes, M., Molina, A.I., Redondo, M.A., Ortega, M. Designing Collaborative User Interfaces for Ubiquitous Applications Using CIAM: The AULA Case Study. Journal of Universal Computer Science, 14(16), pp. 2680-2698, 2008.
- [9] Paternò, F. ConcurTaskTrees: An Engineered Notation for Task Models. The Handbook of Task Analysis for HCI. D. Diaper and N. A. Stanton. LEA, Mahwah, NJ, pp 483-501, 2004.
- [10] Rihm, A., et. al., Intra-logistic, 2ª edición, CEMAT 08. Intra-Logistic de KARDEX y MEGAMAT, 2008.
- [11] Rubart, J. and P. Dawabi (2004) Shared Data modelling with UML-G. International Journal of Computer Applications in Technology, 19.
- [12] van der Aalst, W.M.P., et al. (2001) Proclats: a framework for lightweight interacting workflow processes. Journal of Cooperative Information Systems, 10 (4): 443–482.