

Desarrollo basado en modelos de la interfaz de usuario de sistemas groupware

A model based proposal for user interface development in groupware systems

William J. Giraldo¹, MSc, Cesar A. Collazos², PhD y Faber D. Giraldo³, Ing.

1. Programa de Ingeniería de Sistemas y Computación, Universidad del Quindío, Colombia.

{wjgiraldo,fdgiraldo}@uniquindio.edu.co

2. Grupo IDIS, Universidad del Cauca, Colombia.

ccollazo@unicauca.edu.co

Recibido para revisión 30 de Abril de 2009, aceptado 25 de Agosto de 2009, versión final 11 de Septiembre de 2009

Resumen— En este artículo presentamos una propuesta para el desarrollo de la interfaz de usuario. Esta propuesta apoya el diseño de la interfaz de usuario de aplicaciones groupware que permite la integración con los procesos de software a través de la notación UML. Combinamos modelos casos de uso, modelado de tareas y modelado esencial para capturar lo que los usuarios están tratando de lograr y por qué lo están haciendo. Este enfoque promueve soluciones más sencillas para el diseño de la interfaz de usuario y la funcionalidad subyacente que soporta la interfaz de usuario. Se presenta un caso de estudio (el diseño de un sistema de gestión de Conferencias) para describir nuestra propuesta. El método propuesto se apoya en una herramienta de software llamada CIAT (Collaborative Interactive Applications Tool).

Palabras Clave— Interfaz de Usuario, Desarrollo Basado en Modelos, Groupware, Interacción Persona-Ordenador.

Abstract— In this paper we introduce a proposal for the development of user interface. This proposal supports the user interface design of groupware applications enabling integration with software processes through UML notation. We combine use cases models, task modelling and essential modelling to capture what it is that users are trying to accomplish and why they are doing it. This approach promotes simpler solutions for user interface design and the underlying functionality that supports the user interface. A case study (the design of a Conference Review System) is presented to describe our proposal. The method proposed is supported by a software tool called CIAT (Collaborative Interactive Applications Tool).

Keywords— User Interface, Model-Driven Development, Groupware, Human-Computer Interaction.

I. INTRODUCCIÓN

En este artículo se propone un método basado en modelos para el desarrollo de la interfaz de usuario de sistemas groupware interactivos. Nuestro objetivo es establecer una actitud más centrada en el usuario para el desarrollo de los sistemas groupware. Así, los individuos y los desarrolladores deben ser más precisos, tanto en sus objetivos personales como en las necesidades de los usuarios. Nuestra propuesta establece un puente que ayuda a los desarrolladores a relacionar directamente la estructura de la interfaz de usuario con la estructura de uso del sistema. La información que es esencial para el diseño de la interfaz de usuario se diferencia y separa de otra información que se considere útil para otros fines. Por ejemplo, los diseñadores de usabilidad no usan UML para especificar la información de interfaz de usuario, sino que ésta se recopila en otras notaciones más adecuadas mediante el uso de storyboards, modelos de tareas, prototipos, etc [1]. Por lo tanto, diversos enfoques deben ser integrados, tales como: orientación a objetos, diseño centrado en el usuario, Ingeniería de Software, y dominio de datos, etc

La Ingeniería de Software es una disciplina que se centra principalmente en la funcionalidad, la eficiencia y la fiabilidad del sistema en ejecución. Desde el punto de vista de la ingeniería de software, varios autores han propuesto modelos de proceso válidos para el diseño de interfaz de usuario [2]. Nuestra propuesta permite a los desarrolladores implementar sus aplicaciones, teniendo en cuenta parámetros de usabilidad. El diseño centrado en el usuario (DCU) se refiere al proceso que se centra durante todo el desarrollo del proyecto en la facilidad de uso del sistema. Sin embargo, estos métodos no garantizan productos usables [3]. El diseño de un sistema groupware interactivo involucra disciplinas como la Ingeniería de Software

(IS), CSCW, Ingeniería y Usabilidad (IU), por lo tanto, se requiere la interacción de múltiples actores con sus propios espacios de trabajo [4, 5]. Estos espacios de trabajo deben ofrecer apoyo para el modelado por medio de diferentes diagramas y notaciones.

Nuestro objetivo es desarrollar la interfaz de usuario de sistemas groupware considerando una integración de los distintos procesos involucrados por medio de una distinción del aporte que hace cada uno tanto al modelado de la actividad como a la interfaz misma. Este desarrollo se hace de tal forma que permita encontrar distintos puntos de separación y la posterior integración y que a su vez permita un desarrollo de una interfaz con ciertas características de usabilidad.

Este artículo está organizado de la siguiente forma: La sección 2 presenta algunos trabajos relacionados. La sección 3 muestra el desarrollo de la interfaz de usuario. Finalmente, se exponen las conclusiones extraídas del trabajo desarrollado así como el trabajo futuro que se desprende del mismo.

II. TRABAJOS RELACIONADOS

En torno al desarrollo de interfaces de usuario de sistemas colaborativos se han desarrollado pocos trabajos, uno de los más representativos es CIAM [6] el cual considera los aspectos de la colaboración y la cooperación de manera diferenciada y como es su relación en el desarrollo de la interfaz de usuario. IDEAS [7] se centra en el desarrollo con un enfoque integrador en la IS, y el modelado se hace con OASIS. IDEAS presenta un diseño de la interfaz abstracta de manera manual y más orientada a los datos que a las tareas. Limbourg [8] presenta un enfoque de diseño de la interfaz de usuario por medio de múltiples rutas. Es una de las propuestas más completas en la literatura sin embargo su generación de interfaces es muy centrada a las tareas lo cual ocasiona que se generen muchas interfaces sueltas y poco usables.

A diferencia de estas propuestas nuestro enfoque se centra en combinar el diseño de interfaces guiadas por los datos y por las tareas. Adicionalmente, el desarrollo de la interfaz parte de una selección de los lenguajes de modelado para ser usados de la manera más idónea según el aspecto que se esté modelando en un punto determinado.

III. PROPUESTA DE DESARROLLO DE LA INTERFAZ DE USUARIO

En esta sección se presenta una propuesta para el diseño de la interfaz de usuario de sistemas groupware interactivos. Para lo cual, se han tomado diferentes enfoques para abordar los distintos aspectos de un sistema, aunque nuestro interés se enmarca principalmente en la información relacionada con las

interfaces de usuario. El desarrollo se centra en la integración de procesos y notaciones para apoyar la colaboración y la usabilidad en el proceso de desarrollo de software. Este trabajo se enfoca de manera específica en el diseño de la interacción con el usuario y de manera más superficial en los aspectos de la usabilidad. Para la explicación de este método se ha utilizado un caso de estudio, el sistema de gestión de congresos, el cual se desarrolla a lo largo de la sección. Esto se hace debido a la naturaleza de sus necesidades de cooperación e interacción. Este problema se ha estudiado en la literatura mediante el uso de varios enfoques [9, 10]

Nuestro método está estructurado a partir de tres hilos conceptuales dentro del modelado de negocio y el análisis y diseño del sistema. El primer hilo se basa en el concepto de casos de uso, el segundo hilo en el modelado de tareas y el tercero en el modelado esencial [11]. Los casos de uso de sistema representan la funcionalidad que un sistema ofrece a sus usuarios. Los casos de uso de negocio representan los servicios o un resultado de valor que una empresa ofrece a sus clientes. El modelado esencial define los componentes y la organización de la interfaz de usuario que se ajuste más de cerca de la estructura esencial del trabajo que se está siendo soportado [11]. El modelado esencial se aplica tanto a los casos de uso de negocio como a los casos de uso del sistema. Esta característica nos ofrece dos niveles de casos de uso esencial, debido a la utilización de un mecanismo de abstracción, -el trabajador de negocio- que apoya los casos de uso de negocio con tecnología basado en múltiples modalidades. Esto se explica posteriormente con más detalle. Un modelo de tareas describe las diversas tareas que deben llevarse a cabo por un usuario en la interacción con un sistema interactivo [8]. El modelado de tareas promueve el diseño de la interfaz de usuario agregándole aspectos de usabilidad. Nosotros usamos modelos CTT, que es una notación ampliamente difundida en la comunidad de interacción persona-computador. En [1], Paternò presenta una propuesta para la integración de los modelos de tareas representados en CTT y los casos de uso en UML.

Cuando se combinan estos tres hilos, el resultado es un método que se conecta sin problemas al diseño de la arquitectura de la interfaz de usuario a cuerdo a los propósitos esenciales del sistema y la labor que apoya. Esta conexión se logra a través de cuatro modelos interdependientes: el modelo de dominio, el modelo de contexto, el modelo de trabajo, y el modelo de usuario.

A. Modelos de la Interfaz

La arquitectura de la interfaz de usuario se define teniendo en cuenta diferentes aspectos que introducen modelos para soportar el modelado de rol-usuario, el modelado de contexto, el modelado de tareas, el modelado de dominio y los modelos de presentación. Nuestro principal interés en el desarrollo de la interfaz de usuario se centra en los modelos de presentación de la interfaz de usuario como factor esencial incluso para

escenarios muy simples. Se definen cuatro etapas para la presentación de la interfaz de usuario: en la primera etapa, el modelo de contexto de la interfaz de usuario que define los componentes de la presentación de la interfaz de usuario en una forma menos detallada y de manera general. Cada etapa añade información más detallada hasta llegar a una interfaz de usuario final. Adicionalmente, se representan las relaciones y las operaciones que estos componentes de la interfaz de usuario deben tener.

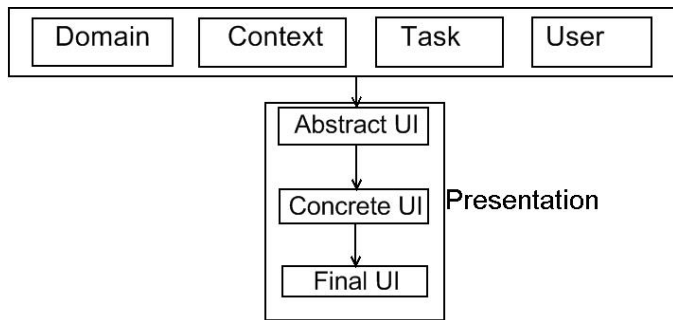


Figura 1. modelos de la interfaz de usuario

El objetivo es evitar la duplicación innecesaria y combinaciones inadecuadas de los elementos de interfaz de usuario, manteniendo siempre el modelo lo más simple posible. El diseño de interfaz de usuario debe captar las intenciones del usuario, describir y representar a su trabajo, y usar nuestra comprensión de su trabajo para el diseño. Los principales componentes de la interfaz de usuario se muestran en la Figura 1.

Un modelo de dominio especifica los datos que los usuarios manipulan acerca de las entidades del mundo real y sus interacciones tal como son entendidas por los usuarios en relación con las acciones que son posibles sobre estos conceptos. El análisis de dominio es parte de la mayoría de los enfoques de desarrollo, y no algo especial para el diseño de la interfaz de usuario. Los objetos del dominio son considerados como instancias de clases que representan los conceptos manipulados y que son totalmente independientes de la forma en que se mostrarán en la pantalla y cómo se almacenan en el computador. Nosotros usamos los diagramas de clases UML como la base para expresar nuestro modelo de dominio, ya que es el estándar de facto en modelado de diagramas de clases.

Un modelo de contexto representa los recursos y capacidades que la interfaz de usuario debe ofrecer al usuario. Ofrece un modelo abstracto de una interfaz de usuario, dejando abiertos los detalles acerca de la apariencia y la elección de los componentes de la interfaz.

Un modelo de contexto es un conjunto de elementos de la interfaz de usuario abstracta que representa los elementos necesarios o deseados y las áreas de trabajo que debe

proporcionar un sistema para apoyar a uno o más casos de uso esencial. Los modelos de contexto se construyen fácilmente por medio de transformaciones adecuadas a partir de los objetos del negocio que representan a cada contexto de uso.

Cada modelo de contexto de uso está en el contexto de un caso de uso de negocio concreto, y por lo tanto, dentro de un conjunto específico de casos de uso esenciales. Nuestro propósito es mantener una interfaz de usuario consistente que pueda ser compartida por diferentes casos de uso y evitar duplicaciones innecesarias.

Un modelo de usuario se compone de una lista de roles de usuario identificados. Un rol de usuario es un conjunto de usuarios que comparten características similares y que pueden ser dispuestos de forma jerárquica. La información del usuario representa un conjunto de intereses comunes, sus conductas y responsabilidades.

Un modelo de usuario se define por la comprensión del uso del sistema y son muy indicados para alcanzar fines de usabilidad. Ellos identifican y representan los aspectos esenciales de las necesidades y requerimientos de los usuarios.

B. Modelado Esencial

Los casos de uso representan los servicios o funciones que ofrece un sistema representados en una forma que pueda ser significativa para los usuarios, tanto a nivel de negocio como a nivel de sistema. Están escritos en términos de interacciones entre los elementos involucrados, a fin de satisfacer un conjunto de características iniciales propuestos por el sistema, tales los casos de uso no son muy útiles para el diseño de la interfaz de usuario; por esta razón, se tienen los casos de uso esencial.

Un caso de uso esencial es una forma simplificada de casos de uso, es un escenario abstracto de una interacción completa e intrínsecamente útil con un sistema tal como se entiende desde la perspectiva de los usuarios [11].

Como se muestra en la Figura 2.a es posible representar un primer flujo de caso de uso esencial con las actividades entre el actor de negocio y el cajero de una forma libre de tecnología libre, en una secuencia de ejecución independiente de la implementación de las actividades deseables y realistas que el cliente necesita para lograr sus objetivos de negocio. Mientras que el segundo caso de uso esencial, a nivel de sistema, a través de un diagrama de tareas puede capturar las tareas internas desde la perspectiva de la usabilidad, en otras palabras, la secuencia es conducida por las tareas del mundo real que proporcionan un valor real a los usuarios. Por otro lado, los casos de uso del sistema describen una unidad coherente de funcionalidad proporcionada por el sistema. Por lo tanto, este enfoque promueve soluciones más sencillas para el diseño de la interfaz de usuario y la funcionalidad subyacente que soporta la interfaz de usuario.

La Figura 2.a representa dos narrativas de casos de uso esenciales para el ejemplo de retiro de efectivo de un cajero

automático. La primera es entre el usuario (las intenciones del usuario) y el trabajador. La segunda es entre el trabajador y el sistema.

La primera descripción de caso de uso esencial está orientada al problema y deja abierta muchas posibilidades para

el diseño e implementación de la interfaz de usuario [12]. Por ejemplo, las acciones del usuario puede aplicarse de distintas maneras: interactuando con el cajero (Figura 2.b.1), por medio del teclado (Figura 2.b.2-3) o por medio de reconocimiento de voz (Figura 2.b.4).

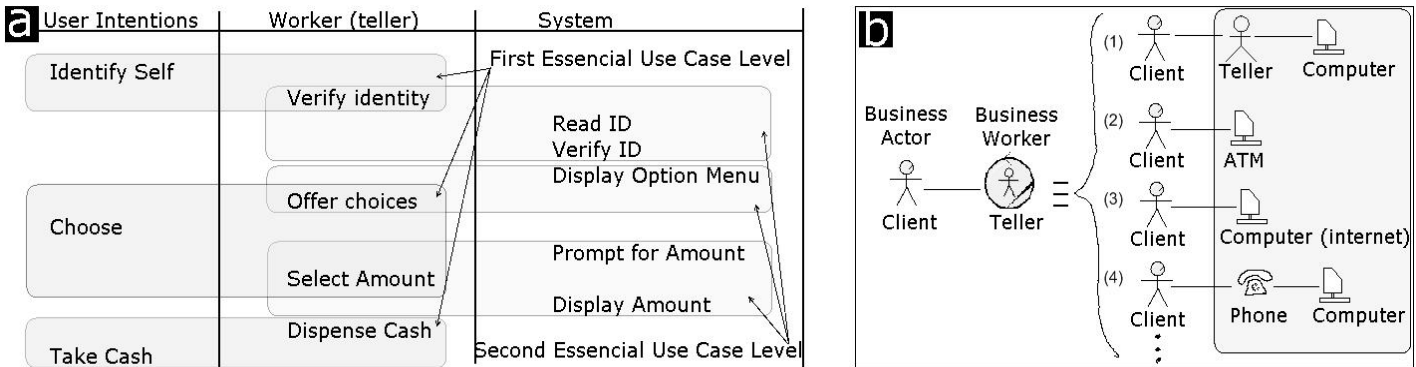


Figura 2. Casos de uso esenciales y realización de un trabajador de negocio. Para un ejemplo retiro de dinero en una máquina ATM.

En el modelo de negocio, todos los casos de uso de negocio son ejecutados por medio de los trabajadores de negocio. En este sentido, un modelo de negocio se define entre los actores de negocio (clientes) y los trabajadores de negocio (Figura 2.b). Esto implica que si un actor de negocio se convierte en un usuario del sistema interactivo es porque el trabajador ha sido automatizado o porque se ha omitido este patrón (Figura 2.b.2-4). Cuando un trabajador de negocio es automatizado, a continuación, sus responsabilidades son adoptadas por el sistema. Por lo tanto, el segundo caso de uso esencial (Figura 2.a), es realizado por el sistema. Utilizamos el concepto de trabajador de negocios para vincular los casos de uso desde el nivel de negocio hasta el nivel de sistema. Por lo tanto, se especifican dos niveles de casos de uso esencial (Figura 2.a). El primer nivel es un flujo de actividades en las que las intenciones del cliente son percibidas por el trabajador y en el segundo flujo las intenciones del trabajador son percibidas por el sistema. La Figura 2.b presenta las posibles formas en que un trabajador puede ser realizado. La realización de un trabajador implica que el usuario del sistema puede ser el cliente del negocio o el trabajador de la empresa. La realización número uno que se muestra en la Figura 2.b.1 es la que corresponde a dos niveles de casos de uso esencial que se muestra en la Figura 2.a. Mediante el uso de trabajadores de negocio, los modelos pueden ser construidos de forma que estén libres de limitaciones innecesarias o supuestos restrictivos. Con este patrón, actor de negocio-trabajador de negocio, el modelo de diseño resultante puede ser más flexible, dejando abiertas más opciones para acomodar los cambios en la tecnología más fácilmente. Por ejemplo, un caso de uso puede tener múltiples implementaciones en la interfaz de usuario. Esta aplicación podría llevarse a cabo con diferentes modalidades. Las interacciones pueden ser especificadas para ser independiente de la modalidad (Figura 2.b.4).

C. Modelado de la actividad

Como se mencionó anteriormente, esta propuesta combina varios enfoques haciendo un mayor énfasis en el análisis y descomposición de las tareas. La descomposición de tareas en sub tareas de menor dificultad es la forma habitual en que los seres humanos hacen el trabajo. Una vez que se define el modelo de tareas, se deben identificar los objetos manipulados por la interfaz de usuario.

Este método de diseño de la interfaz actúa como un flujo de actividades que integra otros procesos de desarrollo que requieren o proveen información útil para el modelado de la actividad o, directamente, para la interfaz de usuario. Este método está soportado mediante el uso de Eclipse Process Framework y de CIAT [13], los cuales permiten que las partes interesadas se centren en sus propias notaciones y procesos.

El primer objetivo de la clasificación de las tareas en torno al diseño de la interfaz de usuario es identificar las actividades del mayor nivel de granularidad, nosotros proponemos los procesos de negocio en UML. En la Fig 3 (arriba) se presenta el modelo de casos de uso de negocio del sistema de gestión de congresos.

Estos procesos están, en cierto modo, separando los usuarios del negocio de los trabajadores (workers) internos del sistema. Es en este primer punto en el que se empiezan a identificar las intenciones de los usuarios potenciales del futuro sistema. Estas intenciones se definen mediante una separación entre las actividades que son propiamente ejecutadas por el usuario en el desempeño de un proceso de negocio y las actividades que serán automatizadas.

Esta propuesta se centra inicialmente en un desarrollo centrado en la ingeniería de software en cuyo caso se pretende identificar primero los datos más relevantes en la operación de los procesos de negocio. Para efectos de la explicación se

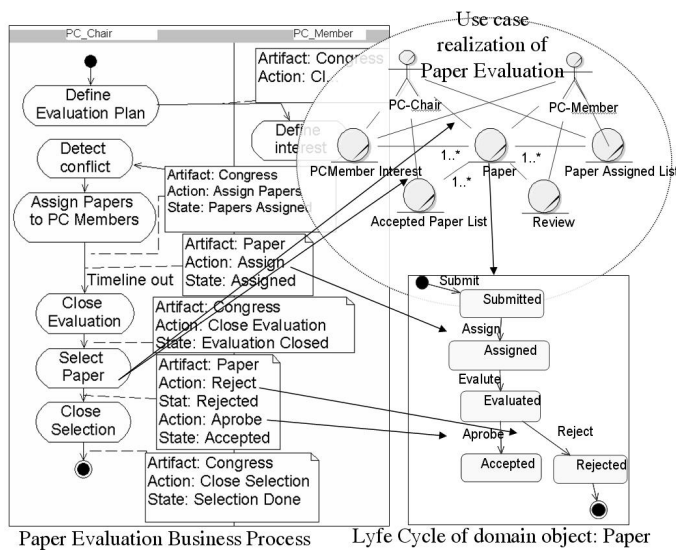
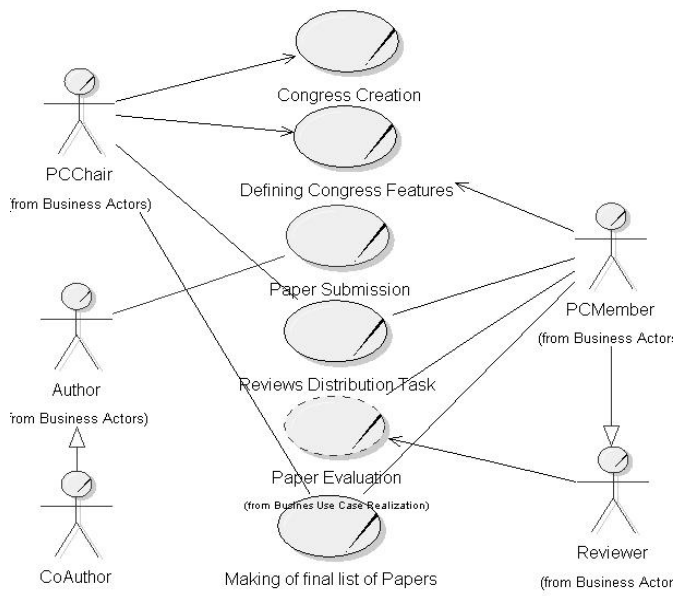


Figura 3. Modelo de casos de uso de negocio (derecha), Realización del Proceso de negocio "Paper Evaluation" (derecha)

desarrolla el caso de uso "Paper Evaluation" en el ejemplo. Cada caso de uso de negocio tiene asociado un diagrama de actividad y un diagrama de objetos de negocio. La Figura 3 presenta el diagrama de actividad y el diagrama de objetos de negocio de "Paper Evaluation".

La Ingeniería de Software estudia también el comportamiento dinámico de los objetos del negocio, esto se hace mediante los diagramas de estado, los cuales en el caso del desarrollo de la interfaz de usuario son útiles para la definición de las interfaces que están relacionadas con el cambio de estado de cada entidad [14].

Estos diagramas juegan un papel muy importante en nuestra propuesta porque permiten la generación de información

contextual asociada a roles, actividades, estado y datos. Esta información del estado es útil para el diseño de la interfaz de usuario.

Nuestro enfoque se basa en la combinación del diseño de la interfaz de usuario centrado en los datos y el diseño centrado en las tareas. Esta combinación se hace de tal forma que se obtiene un balance entre los beneficios de dichas propuestas mejorando así ciertos aspectos de la interfaz de usuario como la reutilización, facilidad de uso, recuerdo, consistencia, reducción del número de interfaces, etc.

En el inicio, se lleva a cabo el diseño de bosquejos de la interfaz de usuario centrado en los datos existentes dentro del contexto de un proceso de negocio (modelo de objetos de negocio). Ver Figura 3 (derecha). Se observa como en el diagrama de actividad "paper evaluation" intervienen dos roles, el "PC_Chair" y el "PC_Member". El PC_Chair tiene un conjunto de intenciones las cuales están asociadas a un conjunto de entidades del negocio. Es más cómodo para el "PC_Chair" interactuar con una única interfaz integrada y consistente que le permita llevar a cabo sus intenciones en el marco de este proceso, que un gran número de interfaces separadas. En propuestas de desarrollo de la interfaz de usuario centrado en los datos se tendría una interfaz por cada entidad de negocio y en las propuestas de desarrollo centrado en la actividad se tendría un conjunto de interfaces derivadas de todas las posibles tareas que surjan de este proceso.

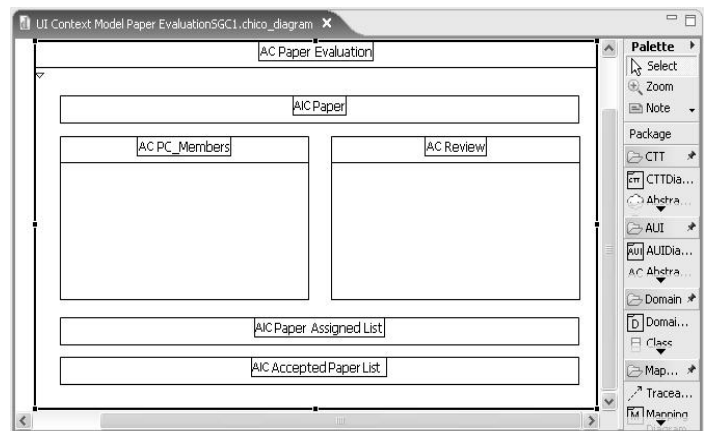


Figura 4. Modelo de contexto de la interfaz de usuario para el proceso "Paper Evaluation".

Nosotros proponemos iniciar el diseño de la interfaz de usuario con un modelo de contexto de la interfaz de usuario. Este modelo se origina a partir de la estructura de los datos que proveen un soporte para las actividades dentro del contexto de un proceso de negocio. Para este ejemplo, las actividades del "PC_Chair" están asociadas con las entidades {Accepted Paper List, Paper, Review, Paper Assigned List} y el "PC_Member" con {Paper, PCMember Interest}. Puede suponerse que todas las entidades intervienen en el proceso, sin embargo, es necesario analizar la relación existente entre las entidades y las actividades para

definir los elementos de la interfaz y su ubicación dentro del modelo de contexto de la interfaz de usuario.

La estructura de este modelo dependerá en gran medida del modelo mental que tenga el usuario acerca de la información que él manipula en las actividades. Es por eso que este modelo debe ser construido y evaluado en conjunto con los usuarios.

La Figura 4 presenta una primera versión del modelo de contexto de la interfaz de usuario. Se observa cómo se han ubicado estratégicamente los componentes que representan las entidades del sistema. Para el caso de la actividad "Select Paper" se encuentran ubicados los componentes {Paper, PC_Members y Review} los cuales intervienen en conjunto.

En este nivel, se posee un modelo de lo que los usuarios esperan esté detrás de las interfaces de usuario. Sin embargo, no se tiene un modelo completo de la interfaz y todos sus componentes deben ser refinados con la información que se obtiene de los diagramas de tareas y de los usuarios. Todo esto con la ayuda de herramientas de modelado y de transformación de modelos.

independencia con la tecnología que se utilice para implementar dichas actividades.

Las actividades del negocio se detallan mediante la especificación de una serie de pasos. Estos pasos son el menor nivel de granularidad al que se llega en el modelado de la actividad en el nivel de negocio. Al comienzo, en la especificación no se sabe si un paso se implementará con tecnología o será una actividad manual de usuario. La Figura 5 presenta a nivel de detalle los pasos que componen la actividad "Select Paper".

A partir de la identificación de los pasos se hace un cambio de notación para pasar a un modelado más idóneo de las actividades de los usuarios. Se utiliza para tal fin los diagramas de tareas CTT. Los modelos de tareas son más adecuadas para el diseño de interfaces usables [1]. Los diagramas CTT hacen la distinción entre las actividades que son propias de los usuarios, las actividades interactivas y las actividades propias del sistema. Los puntos de automatización se representan mediante tareas CTT y, por tanto, se cierra el ciclo del diseño de los casos de uso esenciales. La Figura 5 ilustra el diseño de los pasos desglosados para la actividad "Select Paper" dentro del diagrama de actividad. Las actividades encerradas en paréntesis son las actividades que serán soportadas por el sistema informático, las cuales se modelan a su vez en un conjunto de tareas en CTT.

Estas tareas CTT estarán asociados a casos de uso del sistema y es de esta forma como se integra de nuevo con el proceso de ingeniería de software para el desarrollo de la funcionalidad. Esta asociación permite reducir la brecha que existe entre el diseño de la interfaz de usuario y el desarrollo de la funcionalidad [1].

En la Figura 6 se ilustran los modelos de tareas para las actividades "Assign Papers to PC_Members" y "Select Paper".

E. Especificación de la interfaz de usuario a partir de las tareas del sistema.

Los diagramas de actividad proveen información que es útil tanto para determinar el nivel de agrupamiento de los elementos de la interfaz así como la secuencialidad del uso de los mismos. El nivel de agrupamiento está determinado de acuerdo a la estructura jerárquica que posee dicho modelo de tareas. Es posible mediante transformaciones determinar entonces como estarán agrupados los elementos de la interfaz de acuerdo a un conjunto de actividades. Similarmente, se obtiene es el nivel de adyacencia que poseen los distintos componentes de la interfaz de usuario.

Nuestra propuesta se centra en encontrar un balance entre la interfaz propuesta por el modelo de contexto y la interfaz propuesta por el modelo de actividades el cual se obtiene de manera automática.

Se propone para este fin una serie de modelos de mapping que asignan componentes del modelo de contexto a ciertas

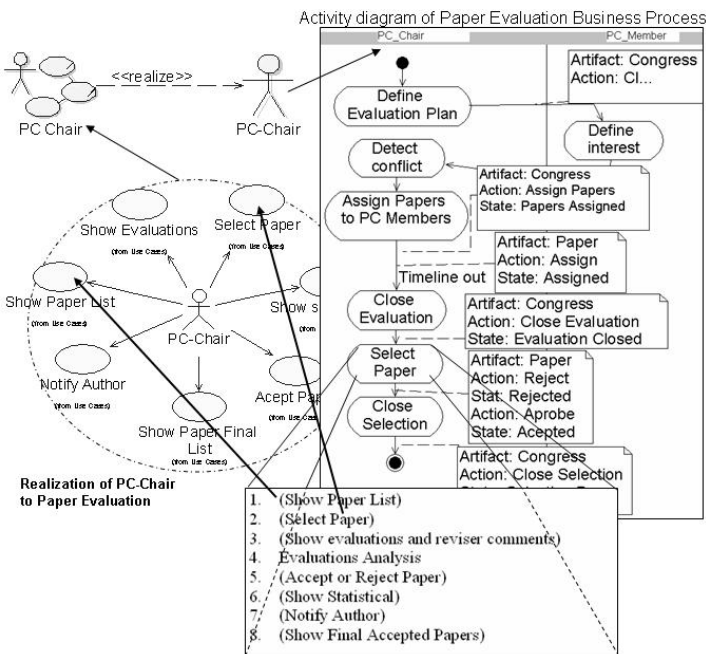


Figura 5. Relación entre las actividades de negocio los pasos y casos de uso de sistema.

D. Actividades orientadas a la interfaz de usuario

Los procesos de negocio son especificados en forma de diagramas de actividad. Dichos diagramas poseen un conjunto de actividades expresadas en términos de verbos o acciones en infinitivo indicando que no tienen un tiempo definido. Es decir, representan tanto al sistema actual como al futuro que se pretende diseñar. De esta forma están representando las intenciones de los usuarios conservando una alta

actividades para que se guíe el proceso de transformación y consecución de la interfaz final. Todo este proceso de diseño se enfoca en el desarrollo de una interfaz de usuario abstracta

que esté acorde al contexto de los datos del proceso, a la información capturada de las actividades y a la información obtenida de los modelos mentales de los usuarios.

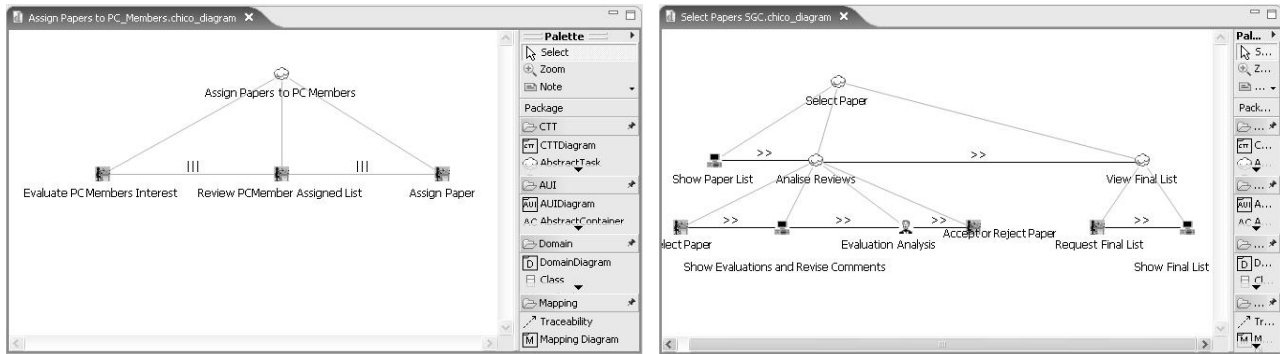


Figura 6. Diagramas de Tareas en CTT para el Proceso de negocio "Paper Evaluation".

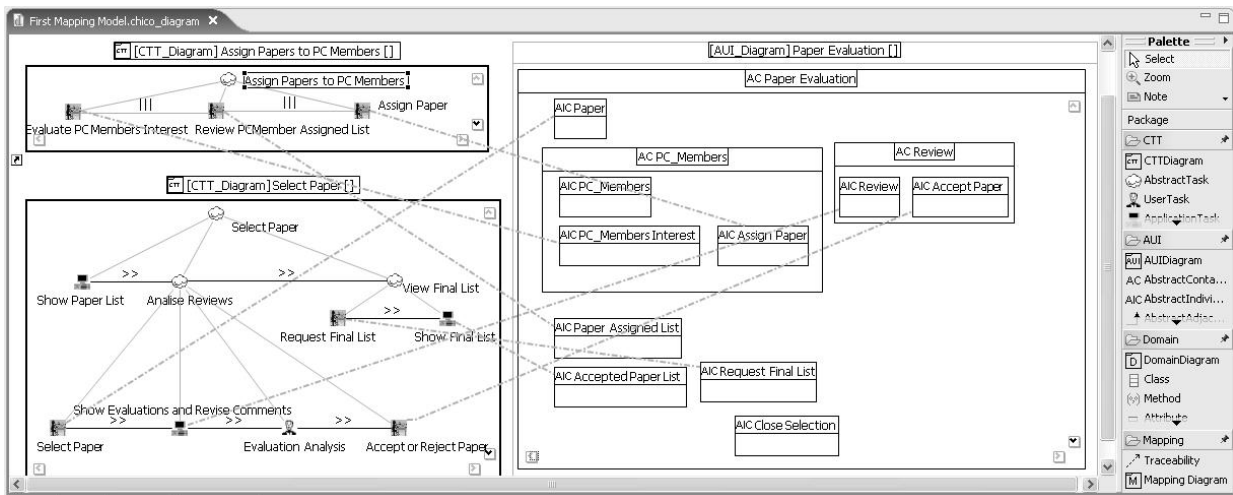


Figura 7. Modelo de mapping entre los diagramas de tareas y la interfaz abstracta.

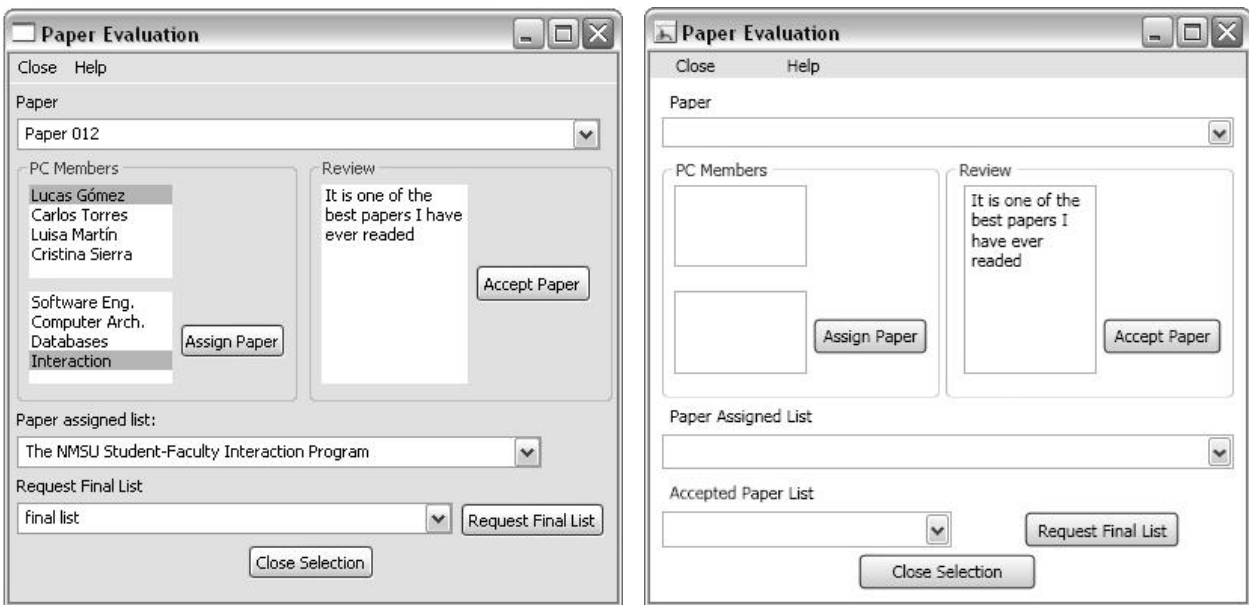


Figura 8. Interfaces finales para el caso de uso de negocio "Paper Evaluation"

La Figura 7 presenta un modelo de mapping entre los diagramas de tareas y el modelo de la interfaz de usuario abstracta para el proceso "Paper Evaluation". Se observa como surgen nuevos elementos de la interfaz de usuario a partir de las actividades. Este mapeo aporta información valiosa a los modelos de transformación permitiendo llevar a cabo un enlace entre los elementos de las entidades de datos que son compartidos entre actividades, así mismo como el enlace con las acciones que se llevan a cabo al interior de la capa de presentación que están enlazadas con la funcionalidad del sistema. La funcionalidad del sistema se modela de nuevo basada en el proceso de desarrollo de software.

F. Interfaz final

Las interfaces de usuario abstractas modelan información que es independiente de la modalidad en la cual será implementada la interfaz. Éstas corresponden a modelos que son independientes de la tecnología en la cual se decide realizar la interfaz de usuario final.

Una vez tomadas una serie de decisiones de diseño sobre la modalidad y la tecnología a utilizar se aborda la fase final del desarrollo. Como paso final de este proceso de desarrollo de la interfaz de usuario se llevan a cabo una serie de transformaciones las cuales obtienen información específica que se ha incorporado a lo largo del proceso tanto en las actividades, los datos y los elementos de la interfaz abstracta. Esta información tiene que ver con las distintas facetas que serán utilizadas para la descripción de los elementos de la interfaz así como los mecanismos asociados a la funcionalidad.

Por la extensión de este paper no se han incorporados detalles de las transformaciones ni de los metamodelos utilizados para llevar a cabo todo el proceso en sí. Estos metamodelos y transformaciones han sido incorporados y materializados una serie de herramientas que conforman el entorno de desarrollo CIAT el cual ha sido presentado en el desarrollo de este paper.

En la Figura 8 se presenta la interfaz final para Java SWT y para XAML.

Cabe anotar que en la versión actual del metamodelo no se han incorporado todas las características del metamodelo de XAML, sin embargo se han descrito las más relevantes para el diseño de las interfaces de usuario convencionales.

IV. CONCLUSIONES

El desarrollo de sistemas de soporte al trabajo en grupo resulta una tarea compleja, entre otros motivos, dada la naturaleza de los grupos involucrados en dicho proceso, cuyos integrantes suelen provenir de distintas áreas de conocimiento.

Es posible combinar distintos enfoques de diseño de la interfaz de usuario y al mismo tiempo seguir distintas consideraciones de usabilidad para mejorar la calidad de las interfaces generadas.

En particular, en este artículo se ha mostrado la aplicación de un método de diseño de la interfaz de usuario enmarcado dentro del desarrollo de un ejemplo típico pero que se observa que puede ser aplicado en una gran variedad de ejemplos, inclusive no solamente en sistemas groupware. Se ha desarrollado una herramienta software denominada CIAT, que implementa las ideas aquí presentadas. Dicha herramienta permite la edición de modelos en diversas notaciones hasta el desarrollo de la interfaz abstracta y concreta, así mismo como la transformación de los modelos necesarios en el proceso.

AGRADECIMIENTOS

Este trabajo ha sido financiado por la Universidad del Quindío (Colombia), y el proyecto Implementación de un Framework para la Evaluación de Usabilidad de Aplicaciones Software soportado en la Creación de un Colaboratorio de Usabilidad (No. 111345221103). Igualmente por el proyecto OMediaDis: Plataforma Abierta para la Gestión de la Distribución Multicanal de Contenidos. NTIN 2008, (TIN2008-06228/TSI),

REFERENCIAS

1. Paternò, F. Towards a UML for Interactive Systems. in 8th International Conference on Engineering for Human-Computer Interaction. 2001.
2. Puerta, A. and J. Eisenstein. MOBI-D: A Model Based Development Environment for User Centered Design. in ICHI'97. 1997. Atlanta (EE.UU.).
3. Granollers, T., et al. Integración de la IPO y la ingeniería del software: MPIu+a. in III Taller en Sistemas Hipermedia Colaborativos y Adaptativos. 2005. Granada España.
4. Molina, A.I., M.A. Redondo, and M. Ortega. A conceptual and methodological framework for modeling interactive groupware applications. in 12th International Workshop on Groupware (CRIWG 2006). 2006. Valladolid. Spain: Springer-Verlag (LNCS).
5. Gutwin, C. and S. Greenberg. Design for Individuals, Design for Groups: Tradeoffs between power and workspace awareness. in ACM CSCW'98. 1998. Seattle: ACM Press.
6. Molina, A.I., et al., CIAM: A methodology for the development of groupware user interfaces. Journal of Universal Computer Science(JUCS), 2007.
7. Lozano, M.D., Entorno Metodológico Orientado a Objetos para la Especificación y Desarrollo de Interfaces de Usuario. 2001.
8. Limbourg, Q., Multi-Path Development of User Interfaces. 2004.
9. Carlsen, S. Action Port Model: A Mixed Paradigm Conceptual Workflow Modeling Language. in Proceedings of the 3rd IFCIS International Conference on Cooperative Information Systems. 1998.
10. Trætteberg, H., Model-based User Interface Design, in Department of Computer and Information Sciences. 2002, Norwegian University of Science and Technology. p. 211.
11. Constantine, L.L., Essential modeling: use cases for user interfaces. interactions, 1995. 2(2): p. 34-46.
12. Nunes, D.N.J., Object Modeling for User-Centered Development and User Interface Design: The Wisdom Approach. 2001, Universidade da Madeira: Funchal.
13. Giraldo, W.J., et al. CIAT, A Model-Based Tool for designing Groupware User Interfaces using CIAM. in Computer-Aided Design of User Interfaces VI, CADUI 2008. 2008. Albacete España: Springer Verlag.
14. Giraldo, W.J., et al. A Model Based Approach for GUI development in groupware systems. in 14th International Workshop on Groupware (CRIWG'2008). 2008. Omaha (EEUU): Lecture Notes in Computer Science. Springer-Verlag.