

# Una experiencia de Captura de Requerimientos en un Ambiente Jerárquico duro

H. Nelson Acosta & Nicolás A. Mosca

INTIA – INCA

Universidad Nacional del Centro de la Provincia de Buenos Aires

Tandil, Argentina, 7000

{nacosta | nmosca }@exa.unicen.edu.ar

II Workshop de Ingeniería de Software y Bases de Datos (WISBD)

## Resumen

El planeamiento de misiones aéreas puede involucrar escenarios tales como preparación, coordinación, definición de parámetros, simulación, etc. Los sistemas actualmente desarrollados cumplen con parte de las necesidades pero no se ajustan en forma completa a las exigencias militares. En pos de lograr una captura de requerimientos exitosa y teniendo en cuenta que el grupo de usuarios se organiza en un estructura jerárquica, se propone un ciclo de desarrollo a medida del caso, cuya descripción es el objeto de este artículo. Para finalizar, se describen conclusiones obtenidas a partir de la implementación de dicho ciclo.

**Palabras claves.** Ingeniería de Requerimientos, Captura de Requerimientos, Casos de Uso y Escenarios, Brainstorming, Prototipación Evolutiva.

## Abstract

The planning of air missions can involve scenes such as preparation, coordination, definition of parameters, simulation, etc. The systems at the moment developed fulfill part of the necessities but they do not adjust in complete form to the military exigencies. In order to obtain a successful capture of requirements and considering that the user group organizes itself in a hierarchic structure, a development cycle sets out in accordance with the case, whose description is the object of this article. To finalize, conclusions obtained from the implementation of this cycle are described.

## 1 INTRODUCCIÓN

El planeamiento de misiones aéreas puede involucrar alguno de los siguientes desarrollos de escenarios esperados a ser seguidos; preparación y generación de productos; coordinación previa de todos los elementos de la misión; consideración de información disponible de variados recursos; establecimiento de parámetros para la reunión de las metas presentadas; y preparación de los datos de inicialización tanto para el equipo a ser usado, como para la simulación de la misión.

La planificación y preparación para misiones aéreas requiere que información esencial sea ingresada al sistema, previamente al comienzo de la misión. Puede incluir planes de rutas, datos de comunicación, designación de puntos de referencia, situaciones amigables o de amenazas, información climática, entre otras. El planeamiento en un ambiente de simulación requiere la inicialización de datos del sistema, tales como incluir posicionamiento inicial de fuerzas (amigas y enemigas), condiciones de la misión, variables del entorno, y parámetros de comunicación.

Los sistemas actualmente desarrollados cumplen con parte de las necesidades pero no se ajustan en forma completa (son muy poco flexibles). Carecen de funcionalidad crucial y poseen ítems importantes rígidos ante cambios menores, como por ejemplo, sólo ofrecen un conjunto predefinido de aviones sin ofrecer la posibilidad del agregado de uno nuevo; ejercicios de misiones pueden ser creados pero sólo utilizando un tipo de avión determinado, no pudiendo hacer uso de una combinación de los existentes; no son multiusuarios, siendo que en la mayoría de los casos son utilizados por integrantes de grupos que realizan prácticas de ejercicios y configuran sus preferencias sobre el sistema. En su mayoría son de propósito general y de uso civil, razón por la cual no contienen funcionalidad que es meramente de interés militar, como son las zonas restringidas, puntos de notificación, datos de carga de armamento, etc. Todo ello da una idea difusa de lo que debe contemplar un sistema de este tipo.

Para el desarrollo del sistema deben obtenerse los requerimientos básicos y agregado a ellos, los específicos del contexto. En pos de lograr una captura exitosa, y por consiguiente, el sistema buscado, y teniendo en cuenta que el grupo de usuarios se organiza en una estructura jerárquica, se propone un ciclo de desarrollo a medida del caso.

Este artículo está organizado de la siguiente forma: en la sección 1 se presenta el objetivo del proyecto de desarrollo junto con la problemática la cual motiva este trabajo, en la sección 2 se describe el caso de aplicación y las fases del método aplicado y en la sección 3 se describe el estado actual del proyecto de desarrollo. Finalmente, en la sección 4 se plantean las conclusiones.

## 2 CASO DE APLICACIÓN

Un proyecto de ingeniería de software involucra un grupo diverso de diferentes stakeholders [2]. Específicamente, el grupo de stakeholders se compone del gerente de la empresa encargada del desarrollo del sistema, el director de proyecto, el equipo desarrollador, y el grupo de usuarios cuya organización se basa meramente en una estructura jerárquica dura, en donde cada uno posee basta experiencia en el dominio. cabe aclarar que se ha tomado la libertad de utilizar el adjetivo “dura” por tratarse de una estructura jerárquica militar.

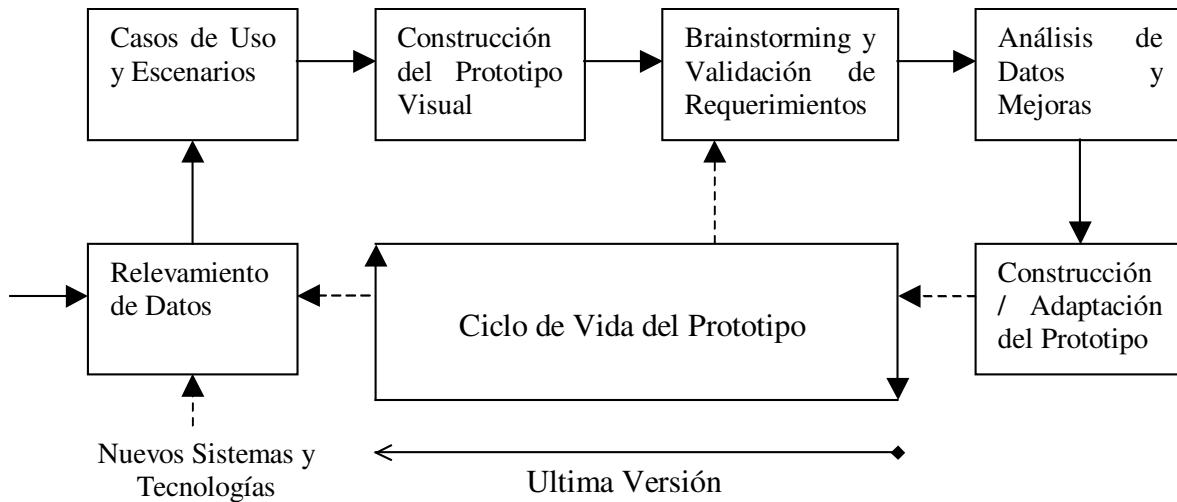


Figura 1: Fases de Captura de Requerimientos y Ciclo de Vida del Prototipo.

Uno de los problemas más difíciles en la determinación de requerimientos del sistema es la comunicación entre diferentes stakeholders. Los analistas de sistemas tienden a hablar con términos técnicos mientras que el cliente y los usuarios finales prefieren el lenguaje utilizado diariamente en su trabajo, y la jerga de los pilotos tiene una terminología basta y sumamente específica que dificulta el diálogo. Como resultado los analistas pueden no tener una clara idea de que tipo de sistema ellos están construyendo y el cliente y los usuarios finales son comúnmente sorprendidos cuando se ponen al tanto que el producto final no cumple con sus expectativas [2].

En la figura 1 se describen las etapas correspondientes al ciclo de captura de requerimientos. Se comienza el ciclo con un relevamiento de datos a partir sistemas existentes y nuevas tecnologías, contrastado con la versión actual del prototipo (si esta no es la primera iteración). Con la documentación de los resultados de la fase anterior se realiza una absorción de escenarios y generalización a casos de uso. Una vez definida la nueva funcionalidad, se representan los escenarios y casos de uso a través de un prototipo visual no ejecutable, el cual es construido para ser luego presentado en la sesión de brainstorming. Esta fase es la de mayor importancia, aquí se capturan nuevos requerimientos y se validan los ya encontrados. Los resultados obtenidos son analizados, dando lugar al diseño de extensiones y mejoras del prototipo ejecutable, que en la siguiente fase son implementadas, obteniéndose así una nueva versión ejecutable del sistema.

## 2.1 Relevamiento de Datos

Esta fase obtiene como entrada lineamientos generales que deben ser tenidos en cuenta a la hora del comienzo del desarrollo del sistema o ante una eminente actualización.

A través de entrevistas, cuestionarios e inserción de analistas dentro del grupo de trabajo de los usuarios por un corto periodo, se describen líneas generales que deben ser seguidas en pos de alcanzar un sistema con la funcionalidad deseada. A partir de estas reuniones se realiza una búsqueda exhaustiva de datos, incluyendo principalmente sistemas existentes y restricciones operacionales.

Los escenarios son definidos como descripciones narrativas [9] o como instancias específicas conteniendo descripciones del entorno, el contexto, los actores, y las acciones con punto de comienzo y fin determinado [8]. Por ejemplo “un piloto ingresa un nuevo avión con características específicas a la base de datos del sistema, para ser utilizado en la configuración de una nueva misión”, o “el sistema genera automáticamente, en menos de 15 segundos, una planilla de vuelo de una misión específica, solicitada por un piloto”.

Sistemas existentes, que en forma parcial o total contienen áreas funcionales de interés, son representados a través de escenarios de situaciones que el sistema debería soportar. Específicamente, el caso de herramientas de uso civil como Jeppessen [10], Garmin [11] y OziExplorer [12], son utilizadas en esta fase para extraer funcionalidad que debería contemplar el sistema final y determinar falencias que no debería contener.

Requerimientos no funcionales definen restricciones globales sobre el sistema o subsistema de software, sobre un requerimiento funcional o sobre el proceso de desarrollo [1]. Problemáticas referidas a restricciones operacionales, obtenidas y documentadas de experiencias previas durante el desarrollo de un sistema similar, son también importantes factores a la hora de realizar una descripción precisa y real de los escenarios. Se emplearon aproximadamente 84 horas hombre, de personal de sistemas.

## 2.2 Absorción de Escenarios y Generalización a Casos de Uso

Los escenarios han sido usados en Human Computer Interface e Ingeniería de Requerimientos, y su uso, como una forma de entender el dominio en el cual el software será usado [13], ha sido recomendado por muchos autores [14, 15, 16, 17].

Los escenarios también son presentados a los pilotos/usuarios como instancias específicas de casos de uso [2] donde un escenario describe el camino de acciones a través de cada caso de uso [3]. Se plantean escenarios posibles, que concuerden con las líneas generales propuestas y con las áreas específicas de interés, para su validación a partir de las restricciones operacionales.

Sommerville y Sawyer [5] y Wiegers [7] sugieren prototipos no ejecutables para ser usados junto con escenarios textuales para capturar requerimientos de los usuarios. A partir de los escenarios se generaliza con el objeto de obtener casos de uso, y de esta manera iniciar/extender la documentación funcional del sistema, dando lugar a la construcción del prototipo visual no ejecutable. Para esta etapa se requirieron aproximadamente 170 horas hombre de personal de sistemas.

## 2.3 Construcción del Prototipo

Asur y Hufnagel [4] sugieren prototipos para ser usados por ejemplo en búsqueda y especificación de requerimientos, estudio de factibilidad de estrategias de desarrollo, definición de interfaces usuario, ayuda en la comunicación entre stakeholders e incremento de su participación en el proceso de desarrollo del sistema, y visualización de aplicaciones futuras.

Normalmente se considera a los prototipos no ejecutables y visuales como técnicas de prototipación baratas, rápidas, y de bajo riesgo que deben ser usadas cuando el comportamiento hipotético del sistema será deshilado o el problema no es bien entendido [4, 5, 7].

Por lo tanto, se decide realizar la construcción de un prototipo visual no ejecutable para comunicar a los stakeholders los casos de uso encontrados. Se propone el uso del prototipo buscando eliminar, o al menos reducir, la diferencia de nomenclatura entre stakeholders. Cabe destacar que el lenguaje de los pilotos de aviones es una jerga muy particular, rica, y difícil de adquirir.

Las pantallas son diseñadas para cada escenario y el esqueleto del prototipo por descripción de que tipos de objetos multimedia va a contener cada una. Entre los objetos multimedia utilizados puede encontrarse imágenes, animaciones con audio, texto emergente, entre otros.

La herramienta utilizada para el diseño y creación de las pantallas es Corel Draw, y para la posterior presentación, Microsoft Power Point, en donde el tiempo total empleado para la construcción del prototipo visual no ejecutable es de 27 horas hombre.

#### 2.4 Sesión de Brainstorming para Captura y Validación de Requerimientos

Una vez obtenido el prototipo, se presenta a los stakeholders durante una sesión de Brainstorming, la cual tiene por objeto capturar nuevos requerimientos y validar los ya encontrados. La técnica de brainstorming es de gran utilidad cuando existe la necesidad de liberar la creatividad de los equipos, generar un número extenso de ideas, involucrar a todos en el proceso, identificar ítems a mejorar, entre otras.

Es sabido que en una sesión de Brainstorming se comienza dando ideas (con la participación de cada stakeholder desde su propia perspectiva) anotando cada una de ellas en un lugar donde todos puedan verlas, incentivando la generación de nuevas. Sin embargo, este mecanismo en un ambiente jerárquico duro puede producir que los miembros de nivel más bajo queden inhibidos e influenciados por ideas propuestas por miembros de nivel más alto, dando como resultado una perspectiva parcial del sistema a desarrollar de la perspectiva global que se hubiera podido obtener si cada uno de los participantes hubiera sentido que podía expresarse libremente.

Con el objeto de solucionar este inconveniente, el trabajo propone capturar las ideas generadas por cada participante a través de entrevistas individuales y anónimas (previas a la sesión), y presentarlas como base al inicio de la sesión de Brainstorming, introduciendo de esta manera las perspectivas de todos los participantes en la captura de requerimientos.

El audio de la sesión es almacenado para ser posteriormente analizado. Si bien esto es visible, no es informado explícitamente porque puede presentar algunos problemas, tales como distracción o retracción de los stakeholders [8].

Estratégicamente se decidió que el moderador sería el más joven de los analistas (24 años), y ocuparía la tarima; mientras que los otros dos analistas se ubicaron en medio de la sala (entre los pilotos y el personal jerárquico). Con esto se buscó acercar al analista a los pilotos, mientras que el personal de mayor edad se limitó a intervenir en caso de conflicto o para cerrar algún avance.

Al comienzo de la sesión el moderador presenta a los stakeholders cada una de las actividades que se van a realizar. Allí también se explica la importancia de la participación fluida de cada uno para lograr una sesión exitosa. Es destacable que se hizo especial énfasis en lo informal de la sesión, fundamentalmente debido a que los stakeholders están organizados en una estructura jerárquica rígida (militar) y se pretende obtener aportes de todos los estratos de la organización.

Seguidamente, se presentan cada una de las pantallas diseñadas para los escenarios encontrados, y se describe brevemente su contexto. Se da lugar a la discusión e inserción de nuevas ideas. Temas no contemplados y discusiones con posturas finales en desacuerdo, son registrados.

Una vez revisadas todas las pantallas, el moderador vuelve a incentivar la participación de los stakeholders para que cada uno comente que temas no tocados en esta sesión son de importancia, detalles que hayan quedado inconclusos y remarcar ítems ya nombrados de gran interés.

Se presentaron y discutieron un total de 23 pantallas en aproximadamente 3 horas, con aproximadamente 15 pilotos / usuarios, 5 personal jerárquico y 3 ingenieros de sistemas. Se utilizó una sala de conferencias con capacidad para 50 personas, provista con cañón sobre una gran pantalla, puntero láser y equipo de sonido. Cabe destacar que se requirieron aproximadamente 80 horas hombre para la preparación de esta etapa.

## 2.5 Análisis de Resultados y Diseño de Extensiones y Mejoras

Al final de la sesión de brainstorming se sigue con una fase de análisis, donde los datos almacenados son analizados. Los datos almacenados consisten en información textual, capturas de nuevas pantallas, otros documentos de requerimientos y el audio de la sesión.

Aquí se analizan los datos y se genera un curso de acción a seguir durante la construcción / adaptación del prototipo ejecutable, para que soporte el nuevo conjunto de requerimientos obtenido.

Se evalúa el impacto sobre el diseño y se proponen potenciales adaptaciones que minimicen la cantidad de cambios y maximicen la flexibilidad ante nuevos requerimientos. Esta etapa requirió aproximadamente 150 horas hombre.

## 2.6 Construcción / Adaptación del Prototipo Ejecutable a partir del Conjunto de Requerimientos

De acuerdo con Kotonya y Somerville [6], los prototipos descartables son desechados cuando el sistema final ha sido desarrollado, mientras que prototipos evolutivos son puestos a disponibilidad de los usuarios tempranamente en el proceso de desarrollo y luego extendidos hasta producir el sistema final.

El conjunto de requerimientos funcionales y restricciones operacionales analizado y validado, da lugar al diseño e implementación de extensiones y mejoras sobre el prototipo evolutivo.

El conjunto de requerimientos es dividido en tres categorías: requerimientos básicos, primarios y secundarios. Los requerimientos básicos representan la estructura básica del sistema; la cual soporta escenarios que cualquier sistema de este tipo debe soportar; los requerimientos primarios son un subconjunto de requerimientos básicos que representan funcionalidad que facilita las tareas

de los usuarios; y por último, los requerimientos secundarios son aquellos que mejoran y extienden la funcionalidad del sistema, pero que son prescindibles.

Una vez divididos, se comienza con el diseño y una posterior implementación de los requerimientos básicos; testeado el prototipo y validado con respecto a lo obtenido de la etapa anterior, se sigue con la adaptación del diseño para soporte de los requerimientos primarios y secundarios respectivamente, previa validación en cada caso.

Una vez realizada la construcción, adaptación y / o extensión el prototipo evolutivo se convierte en la última versión funcional del sistema, la cual será modificada al final de la ejecución del ciclo, durante otra iteración de este mismo método. Esta etapa requirió un total aproximado de 320 horas hombre.

### 3 ESTADO ACTUAL

El prototipo evolutivo se encuentra en la segunda iteración, dentro de la fase de construcción del prototipo visual.

La primera iteración determinó la gran mayoría de los requerimientos y restricciones operacionales. También se encontraron puntos de desacuerdo entre los stakeholders, los cuales fueron registrados y analizados para una posterior prototipación y discusión.

Actualmente la última versión funcional del sistema contempla: visualización y navegación sobre imágenes rasterizadas (Ej. satelitales y aeronáuticas) y vectorizadas (Ej. rutas y caminos, espejos de agua y ferrocarriles, entre otras), todas ellas separadas en capas visuales; también contiene flexibilidad en el diseño para soporte de nuevas imágenes con distintos formatos; agregado de puntos referenciales, creación de piernas, rutas aéreas y misiones; y un entorno personalizado para cada piloto, incluyendo configuración de colores, aviones, cargas, puntos, rutas, misiones, directorios, entre los ítems más importantes. Todo este desarrollo ha requerido aproximadamente 830 horas hombre, para un ciclo de la primera iteración del prototipo evolutivo.

### 4 CONCLUSIONES

Se ha presentado un caso específico de captura de requerimientos, aplicado en un ámbito particular como es el caso de los pilotos y personal jerárquico de la VI Brigada Aérea Tandil.

En un grupo de usuarios organizado jerárquicamente, un primer obstáculo a evadir para capturar requerimientos, a partir de sesiones de Brainstorming, es la estructura misma. Para obtener un buen resultado, se debe asegurar a cada uno completa libertad de expresión y no permitir que usuarios de menor rango se sientan intimidados o que usuarios de mayor rango anulen su participación fluida.

Dos hechos se destacan como altamente favorables a la apertura de ideas al proyecto: 1) que el moderador que ocupa la tarima sea el más joven (24 años), los otros se ubican mezclados entre los participantes. 2) el comenzar el brainstorming con la pizarra llena de las ideas propuestas anónimamente.

En esta etapa del proyecto se concluye que las estrategias descritas, principalmente en los puntos 2.3 y 2.4, han sido completamente exitosas en la definición de la funcionalidad del sistema.

Sumado a lo anterior, los problemas de comunicación con los pilotos, debido a la jerga que utilizan, dificulta aún más la captura de requerimientos válidos. Sin embargo, el resultado obtenido a través del prototipo visual construido excedió las expectativas, minimizando en gran medida el tiempo utilizado en la fase de brainstorming.

## AGRADECIMIENTOS

Este trabajo fue llevado a cabo en el INCA/INTIA, en la Universidad Nacional del Centro de la Provincia de Buenos Aires, en cooperación con Redimec S.R.L. Los autores agradecen a Redimec S.R.L. y a pilotos y personal jerárquico de la VI Brigada Aérea Tandil por hacer este trabajo posible.

## REFERENCIAS

- [1] Julio C. Sampaio, Jorge H. Doorn, *Perspectives on Software Requirements*, Kluwer Academic Publishers, 2004.
- [2] Markus Mannio, Uolevi Nikula, "Requirements Elicitation Using a Combination of Prototypes and Scenarios", *Proceedings from IV Workshop on Requirements Engineering*, Buenos Aires, Argentina, Noviembre 2001, pp. 283-296.
- [3] Kulak, D., Guiney E., *Use case – requirements in context*, ACM Press, 2000.
- [4] Asur, S., Hufnagel, S., "Taxonomy of rapid-prototyping methods and tools", *Proceedings from the IEEE Fourth International Workshop on Rapid System Prototyping*, June 1993, pp. 42-56.
- [5] Somerville, I., Sawyer, P., *Requirements engineering: A good practice guide*, John Wiley & Sons Inc., 1997.
- [6] Kotonya, G., Somerville, I., *Requirements engineering: processes and techniques*, John Wiley & Sons Inc., 1997.
- [7] Wigers, K. E., *Software requirements*, Microsoft Press, 1999.
- [8] McGraw, K., Harbison K., *User-centred requirements: The scenario-based engineering process*, Lawrence Erlbaum Associates, Inc., 1997.
- [9] Constantine, L.L., Lockwood, L.A.D., *Software for use: a practical guide to the models and methods of usage-centered design*, Addison Wesley Longman Inc., 1999.
- [10] Jeppessen sistema de uso civil para planificación. [www.jeppessen.com](http://www.jeppessen.com)
- [11] Garmin GPS (Global Positioning System). [www.garmin.de](http://www.garmin.de)
- [12] OziExplorer SIG (Sistema de información global). [www.ozieplorer.com](http://www.ozieplorer.com)
- [13] Ridao, M., Doorn, J., Leite, J., "Domain Independent Regularities in Scenarios", *Proceedings from the IEEE Fifth International Symposium on Requirements Engineering*, August 2001, pp. 120-127.
- [14] Booch, G., *Object Oriented Design with Applications*, The Benjamin Cumming Publishing Company, Inc. Redwood City, 1991.
- [15] Jacobson, I., Christerson, M., Jonsson, P., Overgaard, G., *Object Oriented Software Engineering- A Use Case Driven Approach*, Reading, MA: Addison Wesley, New York, ACM Press, 1992.



- [16] Potts, C., "Using Schematic Scenarios to Understand User Needs", *Proceedings of DIS'95, Symposium on Designing Interactive Systems: Processes, Practices and Techniques*, ACM Press, University of Michigan, 1995, pp. 247-256.
- [17] Zorman, L., *Requirements Envisaging by Utilizing Scenarios (Rebus)*, Ph.D. Dissertation, University of Southern California, 1995.