



PETRÓLEOS DE VENEZUELA S.A.



MINISTERIO DEL PODER POPULAR  
PARA CIENCIA Y TECNOLOGÍA



Fundación Instituto de  
Ingeniería (FII-MCT)



Fondo Nacional de Ciencia,  
Tecnología e Innovación  
(FONACIT-MCT)



MINISTERIO DEL PODER POPULAR  
PARA EDUCACIÓN SUPERIOR



Fundación Gran Mariscal  
de Ayacucho (FGMA-MES)



GOBIERNO DE NUEVA ESPARTA



CÁMARA VENEZOLANA DE EMPRESAS DE  
TECNOLOGÍAS DE LA INFORMACIÓN

**Microsoft**® MICROSOFT

X WORKSHOP IBEROAMERICANO DE  
INGENIERÍA DE REQUISITOS Y AMBIENTES DE SOFTWARE

<http://kuainasi.ciens.ucv.ve/ideas07>

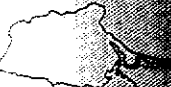
# X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software



Primer Encuentro Venezolano sobre  
Tecnologías de Información e  
Ingeniería de Software

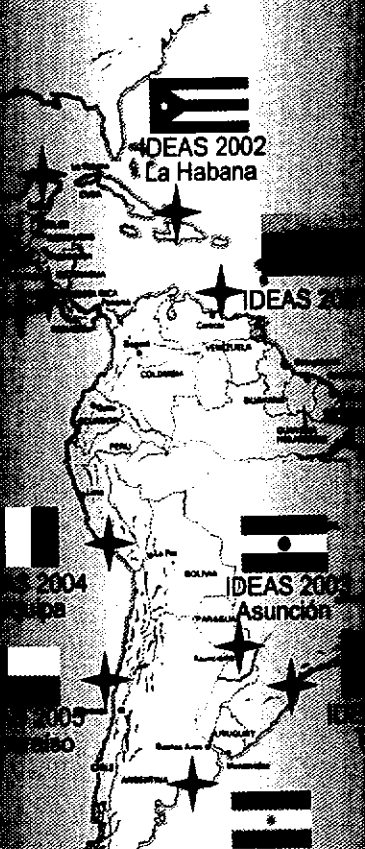
Del 7 al 11 de Mayo de 2007

Iberoamericano  
de Requisitos  
de Software



Isla  
Estado

Del 7 al 11 de



MEMORIAS

## X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software



Editores

Francisca Losavio  
Guilherme Horta Travassos  
Vicente Pelechano  
Isabel Díaz  
Alfredo Matteo

Isla de Margarita, Venezuela  
Del 7 al 11 de Mayo de 2007

MEMORIAS

---

**X Workshop Iberoamericano de  
Ingeniería de Requisitos  
y Ambientes de Software**

### Ficha Técnica

Memorias del X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software (IDEAS'07)  
Editores: F. Losavio, G. H. Travassos, V. Pelechano, I. Díaz, A. Matteo  
Mayo, 2007 – Caracas, Venezuela

Copyright © 2007 by IDEAS'07

All rights reserved

Prohibida la reproducción total o parcial de esta obra,  
por cualquier medio, sin la autorización de sus editores

ISBN:978-980-325-323-3

If58120076201398

### Agradecimiento

La publicación de estas memorias fue posible gracias al apoyo de las siguientes instituciones venezolanas:



Fondo Nacional de Ciencia Tecnología e Innovación (Fonacit)  
del Ministerio del Poder Popular para Ciencia y Tecnología



Consejo de Desarrollo Científico y Humanístico (CDCH)  
de la Universidad Central de Venezuela

---

### Presidencia IDEAS'07

---

#### Francisca Losavio

Universidad Central de Venezuela  
Facultad de Ciencias  
Escuela de Computación

---

### Comité Directivo de IDEAS

---

#### Ernesto Pimentel

Universidad de Málaga  
España

#### Jaelson Brelaz de Castro

Universidad Federal do Pernambuco  
Brasil

#### Luca Cemuzzi

Universidad Católica Nuestra Señora de la Asunción  
Paraguay

#### Luis Olsina

Universidad Nacional de La Pampa  
Argentina

#### Miguel Katrib

Universidad de La Habana  
Cuba

#### Oscar Pastor

Universidad Politécnica de Valencia  
España

#### Silvia Gordillo

Universidad Nacional de La Plata  
Argentina

---

### Comité Organizador IDEAS'07

---

#### Alfredo Matteo

Co-Presidente  
Universidad Central de Venezuela  
Facultad de Ciencias  
Escuela de Computación

#### Isabel Díaz

Co-Presidente  
Universidad Central de Venezuela  
Facultad de Ciencias Económicas y Sociales  
Escuela de Economía

---

### Comité de Programa

---

**Guilherme Horta Travassos**  
Co-Presidente  
Universidade Federal do Rio de Janeiro  
Brasil

**Vicente Pelechano**  
Co-Presidente  
Universidad Politécnica de Valencia  
España

---

### Miembros del Comité de Programa

---

- (1) **Alessandro Garcia**, Lancaster University – United Kingdom
- (2) **Alfredo Matteo**, Universidad Central de Venezuela – Venezuela
- (3) **Altigran Soares da Silva**, Universidade Federal do Amazonas – Brasil
- (4) **Álvaro Arenas**, CCLRC Rutherford Appleton Laboratory – Reino Unido
- (5) **Amador Durán**, Universidad de Sevilla – España
- (6) **Antonio Brogi**, Universidad de Pisa – Italia
- (7) **Antonio Ruiz**, Universidad de Sevilla – España
- (8) **Antonio Valleclillo**, Universidad de Málaga – España
- (9) **Carne Quer**, Universitat Politècnica de Catalunya – España
- (10) **Claudia Maria Lima Werner**, COPPE/Universidade Federal do Rio de Janeiro
- (11) **Emília Mendes**, University of Auckland – Nueva Zelanda
- (12) **Ernesto Pimentel**, Universidad de Málaga – España
- (13) **Ernest Teniente**, Universitat Politècnica de Catalunya – España
- (14) **Esperanza Marcos**, Universidad Rey Juan Carlos – España
- (15) **Fernanda Aletar**, Universidade de Pernambuco – Brasil
- (16) **Francisco Ruiz**, Universidad de Castilla-La Mancha – España
- (17) **Gustavo Rossi**, Universidad Nacional de La Plata – Argentina
- (18) **Itana Gimenes**, Universidade Estadual de Maringá – Brasil
- (19) **Jaelson Castro**, Universidad Federal de Pernambuco – Brasil
- (20) **João Araújo**, Universidade Nova de Lisboa – Portugal
- (21) **Joao Falcão e Cunha**, Universidade do Porto – Portugal
- (22) **Jonás Montilva**, Universidad de Los Andes – Venezuela
- (23) **José Carlos Maldonado**, Universidade de São Paulo – Brasil
- (24) **Juan Hernández**, Universidad de Extremadura – España
- (25) **Judith Barrios**, Universidad de Los Andes – Venezuela
- (26) **Júlio Leite**, Pontificia Universidade Católica do Rio de Janeiro – Brasil
- (27) **Kátia Marçal de Oliveira**, Universidade Católica de Brasília – Brasil
- (28) **Luca Cernuzzi**, Universidad Católica Nuestra Señora de la Asunción – Paraguay
- (29) **Luis Olsina**, Universidad Nacional de La Pampa – Argentina
- (30) **Maria Lancastre**, Universidade de Pernambuco – Brasil
- (31) **Manoel Mendonça**, Universidade Salvador UNIFACS – Brasil
- (32) **Marcello Visconti**, Universidad Técnica Federico Santa María – Chile
- (33) **Márcio Delamaro**, Centro Universitário Eurípides de Marília – Brasil
- (34) **Márcio de Oliveira Barro**, Universidade Federal do Estado do Rio de Janeiro – Brasil
- (35) **María Angélica Ovalles**, Universidad Simón Bolívar – Venezuela
- (36) **Mario Plattini**, Universidad de Castilla-La Mancha – España
- (37) **Miguel Katrib**, Universidad de La Habana – Cuba
- (38) **Nancy Zambrano**, Universidad Central de Venezuela – Venezuela
- (39) **Natalia Juristo**, Universidad Politécnica de Madrid – España
- (40) **Oscar Pastor**, Universidad Politécnica de Valencia – España
- (41) **Pere Botella**, Universitat Politècnica de Catalunya – España
- (42) **Regina Maria Maciel Braga**, Universidade Federal de Juiz de Fora – Brasil
- (43) **Ricardo de A. Falbo**, Universidade Federal do Espírito Santo – Brasil
- (44) **Sandra Fabbri**, Universidade Federal de Sao Carlos – Brasil
- (45) **Silvia Gordillo**, Universidad Nacional de La Plata – Argentina

---

## Revisores Colaboradores

---

- (1) **Alberto Abelló**, Universidad Politècnica de Catalunya – España
- (2) **Alicia Díaz**, Universidad Nacional de La Pampa – Argentina
- (3) **Antonia Reina Quintero**, Universidad de Sevilla – España
- (4) **Arló Claudio Dias Neto**, COPPE/Universidade Federal do Rio de Janeiro – Brasil
- (5) **Heman Melgratti**, Universidad de Pisa – Italia
- (6) **Isabel Díaz**, Universidad Central de Venezuela – Venezuela
- (7) **Isi Castillo**, Universidad Central de Venezuela – Venezuela
- (8) **Ismael Navas**, Universidad de Pisa – Italia
- (9) **Javier Bazzocco**, Universidad Nacional de La Pampa – Argentina
- (10) **Joaquín Peña Siles**, Universidad de Sevilla – España
- (11) **Jobson Luiz Massolar da Silva**, COPPE/Universidade Federal do Rio de Janeiro – Brasil
- (12) **José María Conejero**, Universidad de Extremadura – España
- (13) **Juan Manuel Murillo**, Universidad de Extremadura – España
- (14) **María Istela Cagnin**, Centro Universitário Eurípides de Marília – Brasil
- (15) **María José Escalona**, Universidad de Sevilla – España
- (16) **Pedro J. Clemente**, Universidad de Extremadura – España
- (17) **Pedro Valderas**, Universidad Politècnica de Valencia – España
- (18) **Razvan Popescu**, Universidad de Pisa – Italia
- (19) **Rodrigo de Oliveira Spinola**, COPPE/Universidade Federal do Rio de Janeiro – Brasil
- (20) **Thalzel Fuentes**, Universidad de Pisa – Italia
- (21) **Valter Vieira de Camargo**, Centro Universitário Eurípides de Marília – Brasil
- (22) **Victoria Torres**, Universidad Politècnica de Valencia – España

## PRÓLOGO

Este volumen contiene los trabajos aceptados y presentados en el *X Workshop Iberoamericano de Ingeniería de Requisitos y Ambientes de Software: IDEAS'07* celebrado en la Isla de Margarita, Venezuela, del 7 al 11 de mayo de 2007.

Muy brevemente mencionaremos los números y los pasos que dimos durante el proceso de evaluación. Inicialmente se enviaron 104 resúmenes que finalmente se concretaron en el envío de 82 artículos. Cada artículo se asignó a 3 revisores. En la difícil decisión de aceptar o rechazar los trabajos se discutieron las discrepancias con respecto a un mismo artículo para alcanzar, donde fuera posible, un consenso. Además de esto adoptamos el criterio de aceptar como artículos todos y sólo aquellos que obtuvieron un promedio igual o superior a 4 sobre 6. Así, el resultado final ha sido que de los 82 artículos enviados, 29 han sido aceptados para su presentación.

La cantidad de trabajos enviados se ha mantenido, lo que constituye un síntoma claro del interés que existe por IDEAS. Esto nos hace pensar que, pasados 10 años, IDEAS ha lograrse consolidarse como un foro de primer nivel en Ingeniería del Software para Iberoamérica. Al mismo tiempo, el porcentaje de aceptación indica un nivel de exigencia interesante para la realidad Iberoamericana que puede ayudar a situar a IDEAS a una altura que permita obtener un mayor reconocimiento de las calificaciones de los autores en sus respectivos países.

Evidentemente, todo esto no hubiera sido posible sin la valiosa colaboración de distintos actores a quienes se dirigen nuestros más sinceros agradecimientos. Entre ellos cabe mencionar a los *autores*, por su esfuerzo en investigación, los *revisores*, *miembros del Comité de Programa* y sus *colaboradores*, quienes han realizado un esfuerzo particularmente intenso por la cantidad de trabajos recibidos y el proceso adoptado, los *miembros del Comité Organizador*, los *ponentes* de los tutoriales y conferencias invitadas, el *Comité Directivo de IDEAS* y todas las demás *personas e instituciones patrocinantes* que de distintas formas han contribuido para que este evento se constituya en una ocasión importante y beneficiosa para la comunidad científica en nuestra región.

Esperamos disfruten de IDEAS'07 así como de la cálida acogida y exuberante belleza de la Isla de Margarita.

Francisca Losavio  
Presidenta IDEAS'07

Guilherme Horta Travassos  
Co-Presidente Comité de Programa  
Vicente Pelechano  
Co-Presidente Comité de Programa

## ÍNDICE DE CONTENIDOS

<b>SESIÓN 1: LENGUAJES, MÉTODOS, PROCESOS Y HERRAMIENTAS (Parte 1) .....</b>	<b>1</b>
<b>Extracting the Best Features of Two Tropos Approaches for the Efficient Design of MAS .....</b>	<b>3</b>
María Jocelia Silva, Paulo Roberto Maciel, Rosa C. Pinto, Fernanda Alencar, Patrícia Tedesco, Jaelson Castro Universidade Federal de Pernambuco, Brasil	
<b>Soporte Automatizado a la Ingeniería de Requisitos de Seguridad .....</b>	<b>17</b>
Daniel Mellado <sup>1</sup> , Moisés Rodríguez <sup>2</sup> , Eduardo Fernández-Medina <sup>2</sup> , Mario Piattini <sup>2</sup> <sup>1</sup> Ministerio de Trabajo y Asuntos Sociales, España <sup>2</sup> Universidad de Castilla-La Mancha, España	
<b>SESIÓN 2: MODELADO ORGANIZACIONAL .....</b>	<b>31</b>
<b>Construcción de Modelos de Requisitos a partir de Modelos de Procesos y de Metas .....</b>	<b>33</b>
José De la Vara, David Anes, Juan Sánchez Universidad Politécnica de Valencia, España	
<b>Modelagem de Requisitos Organizacionais, Não-Funcionais e Funcionais em Software Legado com Ênfase na Técnica I* .....</b>	<b>47</b>
Victor F. Araya S. <sup>1,2</sup> , André A. Vicente <sup>2</sup> , Fabio G. Köerich <sup>2</sup> , Jaelson Brelaz de Castro <sup>3</sup> <sup>1</sup> Universidad de Talca, Chile <sup>2</sup> Universidade Estadual do Oeste do Paraná, Brasil <sup>3</sup> Universidade Federal de Pernambuco, Brasil	
<b>SESIÓN 3: MEDICIÓN Y EXPERIMENTACIÓN .....</b>	<b>61</b>
<b>Validando la Usabilidad y Mantenibilidad de los Modelos de Procesos de Negocio: un Experimento y su Réplica .....</b>	<b>63</b>
Elvira Rolón <sup>1</sup> , Félix García <sup>2</sup> , Francisco Ruiz <sup>2</sup> , Mario Piattini <sup>2</sup> <sup>1</sup> Universidad Autónoma de Tamaulipas, México <sup>2</sup> Universidad de Castilla-La Mancha, España	
<b>Soporte de Información Contextual en un Marco de Medición y Evaluación .....</b>	<b>77</b>
Hernán Molina, Luis Olsina Universidad Nacional de La Pampa, Argentina	
<b>Propuesta de Marco para la Selección de Técnicas de Educación de Requisitos ...</b>	<b>91</b>
Dante Carrizo <sup>1</sup> , Oscar Dieste <sup>2</sup> <sup>1</sup> Universidad Complutense de Madrid, España <sup>2</sup> Universidad Politécnica de Madrid, España	

<b>SESIÓN 4: INGENIERÍA DE REQUISITOS .....</b>	<b>105</b>
<b>Discovering Group Communication Requirements .....</b>	<b>107</b>
Igor Miranda <sup>1</sup> , Renata Araujo <sup>2</sup> , Marcos Borges <sup>1</sup> <sup>1</sup> Universidade Federal do Rio de Janeiro, Brasil <sup>2</sup> Universidade Federal do Estado do Rio de Janeiro, Brasil	
<b>Comparaç�o do Impacto do Uso de um Processo de Engenharia de Requisitos entre Grupos de Desenvolvimento de Software - Um Estudo de Caso .....</b>	<b>121</b>
Elias Canhadas Genvigir <sup>1-2</sup> , Nilson Sant'Anna <sup>1</sup> <sup>1</sup> Universidade Tecnol�gica Federal do Paran�, Brasil <sup>2</sup> Instituto Nacional de Pesquisas Espaciais, Brasil	
<b>Extensi�n al Modelo de Separaci�n Multi-Dimensional de Concerns en Ingenier�a de Requisitos .....</b>	<b>135</b>
Carlos A. Ospina, Carlos A. Parra, Luis F. Londo�o, Raquel Anaya Universidad EAFIT, Colombia	
<b>SESI�N 5: PROCESOS DEL NEGOCIO .....</b>	<b>149</b>
<b>Una Propuesta Basada en Modelos para la Construcci�n de Sistemas Ubicuos que den Soporte a Procesos de Negocio .....</b>	<b>151</b>
Pau Giner, Victoria Torres Universidad Polit�cnica de Valencia, Espa�a	
<b>Experiencia en Transformaci�n de Modelos de Procesos de Negocios desde BPMN a XPDL .....</b>	<b>165</b>
Beatriz Mora, Francisco Ruiz, F�lix Garc�a, Mario Piattini Universidad de Castilla-La Mancha, Espa�a	
<b>Verification of Models in a MDA Approach for Collaborative Business Processes ...</b>	<b>179</b>
Pablo Villarreal <sup>1</sup> , Jorge Roa <sup>1</sup> , Enrique Salomone <sup>1-2</sup> , Omar Chlotti <sup>1-2</sup> <sup>1</sup> Universidad Tecnol�gica Nacional, Argentina <sup>2</sup> INGAR-CONICET, Argentina	
<b>SESI�N 6: LENGUAJES, M�TODOS, PROCESOS Y HERRAMIENTAS (Parte 2) .....</b>	<b>193</b>
<b>Towards a Standardized Description and a Systematic Use of Social Patterns .....</b>	<b>195</b>
Carla Silva <sup>1</sup> , Jo�o Ara�jo <sup>2</sup> , Ana Moreira <sup>2</sup> , Jaelson Castro <sup>1</sup> <sup>1</sup> Universidade Federal de Pernambuco, Brasil <sup>2</sup> Universidade Nova de Lisboa, Portugal	
<b>Aplicaci�n de QVT al Desarrollo de Almacenes de Datos Seguros: Un caso de Estudio .....</b>	<b>209</b>
Emilio Soler <sup>1</sup> , Juan Trujillo <sup>2</sup> , Eduardo Fern�ndez-Medina <sup>3</sup> , Mario Piattini <sup>3</sup> <sup>1</sup> Universidad de Matanzas, Cuba <sup>2</sup> Universidad de Alicante, Espa�a <sup>3</sup> Universidad de Castilla-La Mancha, Espa�a	

<b>SESI�N 7: MDA Y TRANSFORMACI�N DE MODELOS .....</b>	<b>223</b>
<b>Marco de Referencia para la Evaluaci�n de Herramientas basadas en MDA .....</b>	<b>225</b>
Juan Bernardo Quintero, Raquel Anaya de Paez Universidad EAFIT, Colombia	
<b>Composici�n de Transformaciones de Modelos en MDD basada en el �lgebra Relacional .....</b>	<b>239</b>
Roxana Giandini <sup>1</sup> , Gabriela P�rez <sup>1</sup> , Claudia Pons <sup>1-2</sup> <sup>1</sup> Universidad Nacional de La Plata, Argentina <sup>2</sup> Universidad Abierta Interamericana, Argentina	
<b>OOWS Suite: Un Entorno de Desarrollo para Aplicaciones Web basado en MDA ...</b>	<b>253</b>
Francisco Valverde, Pedro Valderas, Joan Fons Universidad Polit�cnica de Valencia, Espa�a	
<b>SESI�N 8: LENGUAJES, M�TODOS, PROCESOS Y HERRAMIENTAS (Parte 3) .....</b>	<b>267</b>
<b>Utilizando a T�cnica I* para Modelar a Concep�o de Vigotski Visando Auxiliar o Processo de Desenvolvimento de Software Educacional para Pessoas com Defici�ncia Visual .....</b>	<b>269</b>
Victor F. Araya S. <sup>1-2</sup> , Dorivaldo Rodrigues da Silva <sup>2</sup> , Andr� Abe Vicente <sup>2</sup> , Jaelson de Castro <sup>3</sup> <sup>1</sup> Universidad de Talca, Chile <sup>2</sup> Universidade Estadual do Oeste do Paran�, Brasil <sup>3</sup> Universidad Federal de Pernambuco, Brasil	
<b>Intercambio de Modelos UML y OO-Method .....</b>	<b>283</b>
Beatriz Mar�n, Giovanni Giachetti, Oscar Pastor Universidad Polit�cnica de Valencia, Espa�a	
<b>Apoio Automatizado � Ger�ncia de Riscos Cooperativa .....</b>	<b>297</b>
Victorio Carvalho, Alexandre Coelho, Ricardo Falbo Universidade Federal do Esp�rito Santo, Brasil	
<b>SESI�N 9: LENGUAJES, M�TODOS, PROCESOS Y HERRAMIENTAS (Parte 4) .....</b>	<b>311</b>
<b>Um Modelo Integrado de Requisitos com Casos de Uso .....</b>	<b>313</b>
Michel Fortuna <sup>1-2</sup> , Claudia Werner <sup>1</sup> , Marcos Borges <sup>1</sup> <sup>1</sup> Universidade Federal do Rio de Janeiro, Brasil <sup>2</sup> Universidade Federal do Julz de Fora, Brasil	
<b>Planejamento Integrado das Atividades de Codifica�o e Testes em Orienta�o a Objetos no N�vel de Granularidade dos M�todos.....</b>	<b>327</b>
Tatiane Lopes, Clovis Fernandes Instituto Tecnol�gico de Aeron�utica, Brasil	
<b>Desenvolvimento de Interface com Usu�rio Dirigida por Modelos com Gera�o Autom�tica de C�digo .....</b>	<b>341</b>
Lucas Issa <sup>1</sup> , Clarindo P�dua <sup>1</sup> , Rodolfo Resende <sup>1</sup> , Stenio Viveiros <sup>1</sup> , Pedro Neto <sup>2</sup> <sup>1</sup> Universidade Federal de Minas Gerais, Brasil <sup>2</sup> Universidade Federal do Piaul, Brasil	



<b>SESIÓN 10: REQUISITOS Y COLABORACIÓN .....</b>	<b>355</b>
<b>OO-Sketch: Una Herramienta para la Captura de Requisitos de Interacción .....</b>	<b>357</b>
José Ignacio Panach, Sergio España, Inés Pederiva, Oscar Pastor Universidad Politécnica de Valencia, España	
<b>Colaboração e Negociação na Elicitação de Requisitos .....</b>	<b>371</b>
Danilo Freitas, Marcos Borges, Renata Araujo Universidade Federal do Rio de Janeiro, Brasil	
<b>SESIÓN 11: SERVICIOS WEB Y COMPONENTES .....</b>	<b>385</b>
<b>Automated Generation of BPEL Adapters .....</b>	<b>387</b>
Antonio Brogi, Razvan Popescu Universidad de Pisa, Italia	
<b>Un Enfoque Dirigido por Modelos para el Desarrollo de Servicios Web Semánticos</b>	<b>401</b>
César J. Acuña, Esperanza Marcos, Mariano Minoli Universidad Rey Juan Carlos, España	
<b>Un Perfil UML para la definición de Componentes Inteligentes .....</b>	<b>415</b>
José Luis Pastrana <sup>1</sup> , Ernesto Pimentel <sup>1</sup> , Miguel Katrib <sup>2</sup> <sup>1</sup> Universidad de Málaga, España <sup>2</sup> Universidad de La Habana, Cuba	
<b>EVETIS'07: PRIMER ENCUENTRO VENEZOLANO SOBRE TECNOLOGÍAS DE LA INFORMACIÓN E INGENIERÍA DE SOFTWARE .....</b>	<b>427</b>

---

## Sesión 1

### Lenguajes, Métodos, Procesos y Herramientas (Parte 1)

4. Bresciani, P., Giorgini, P., Giunchiglia, F., Mylopoulos, J., Perini, A.: Towards an Agent Oriented approach to Software Engineering. In the Workshop Dagli oggetti agli agenti: tendenze evolutive dei sistemi software (2001).
5. Bresciani, P., Giorgini, P., Giunchiglia, F., Mylopoulos, J. and Perini, A.: Tropos: An Agent-Oriented Software Development Methodology. *Autonomous Agents and Multi-Agent Systems*, 8(3):203-236, (2004).
6. Castro, J., Kolp, M. and Mylopoulos, J.: Towards Requirements-Driven Information Systems Engineering: The Tropos Project. *Information Systems Journal*, Elsevier, Vol 27: 365-89 (2002).
7. Castro, J., Kolp, M., Mylopoulos, J.: UML for Agent-Oriented Software Development: the Tropos Proposal. In *International Conference on the Unified Modeling Language* (2001).
8. Cernuzzi, L. and Rossi, G. "On the Evaluation of Agent Oriented Methodologies", *Proceedings of the Workshop on Agent Oriented Methodology (OOPSLA'02)*. Pages: 21-32. COTAR, (2002).
9. Cernuzzi, L. and Zambonelli, F. "Experiencing AUML in the Gaia Methodology", in *5th International Conference on Enterprise Information Systems*. Angers - France April 2003.
10. Dam, K. H. and Winikoff, M. "Comparing Agent-oriented Methodologies", in Giorgini, P., Henderson-Sellers, B., and Winikoff, M., editors, *Agent-Oriented Information Systems: 5th International Bi-Conference Workshop*, volume 3030 of LNAI, pages 78-93. Springer-Verlag, Berlin, Germany. (2003).
11. Fuxman, A., Giorgini, P., Kolp, M. and Mylopoulos, J.: Information systems as social structures. In *Proc. of the 2nd Int. Conf. on Formal Ontologies for Information Systems, FOIS'01*, Ogunquit, USA (2001).
12. Hayden, S., Carrick, C. and Yang, Q.: Architectural design patterns for multiagent coordination. In *Proc. of the 3rd Int. Conf. on Autonomous Agents, Agents'99*, Seattle (1999).
13. Horn, P.: *Autonomic Computing: IBM's Perspective on The State of Information Technology* - IBM Corporation. <<http://www.research.ibm.com/autonomic>>, last access Jan 2007.
14. Juan, T., Sterling, L., Martelli, M., and Mascardi, V. "Customizing AOSE Methodologies by Reusing AOSE Features", in *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'03)*, pages 113-120, New York, USA. ACM Press. (2003).
15. Kolp, M., Castro, J. and Mylopoulos, J.: A social organization perspective on software architectures. In *Proc. of the 1st Int. Workshop From Software Requirements to Architectures, STRAW'01*, pages 5-12, Toronto, Canada (2001).
16. Rao, A., and Georgeff, M.: Decision procedures for BDIlogics. *Journal of Logic and Computation* 8(3):293-344 (1998).
17. Sturm A., Dori, D. & Shehory O.: Comparative Evaluation of Agent-Oriented Methodologies, in *Methodologies and Software Engineering for Agent Systems*, Kluwer, pp. 127-149 (2004).
18. Wautelet Y., Kolp M. and Achbany Y.: S-Tropos, An Iterative SPEM-Centric Software Project Management Process, Working Paper IAG (2005).
19. Wooldridge, M.: *An Introduction to MultiAgent Systems*. John Wiley and sons, LTD, Chichester, England (2002).
20. Yu, E.: *Modelling Strategic Relationships for Process Reengineering*. PhD thesis, University of Toronto, Department of Computer Science (1995).
21. Zambonelli, F., Jennings, N. R., Wooldridge, M.: Developing Multiagent Systems: The Gaia Methodology. *ACM Transactions on Software Engineering Methodology*, v. 12, n. 3, p. 317-370 (2003).

## Soporte Automatizado a la Ingeniería de Requisitos de Seguridad

Daniel Mellado<sup>1</sup>, Moisés Rodríguez<sup>2</sup>, Eduardo Fernández-Medina<sup>2</sup>, Mario Piattini<sup>2</sup>

<sup>1</sup> Ministerio de Trabajo y Asuntos Sociales; Gerencia de Informática de la Seguridad Social; Centro Informático del Instituto Nacional de la Seguridad Social; Madrid, España.  
Daniel.Mellado@alu.uclm.es

<sup>2</sup> Grupo ALARCOS, Dpto. de Tecnologías y Sistemas de Información, Centro Mixto de Investigación y Desarrollo de Software UCLM-Soluziona; Universidad de Castilla-La Mancha. Paseo de la Universidad 4, 13071 Ciudad Real, España.  
(Eduardo.Fdez-Medina, Mario.Piattini, Moises.Rodriguez)@uclm.es

**Resumen.** La Ingeniería de Requisitos de Seguridad es una disciplina que se está erigiendo como una importante rama de la Ingeniería del Software, debido a que se está comprendiendo cada vez más que la seguridad debe abordarse desde el inicio de la fase de requisitos. Pero sin una herramienta CARE (Computer-Aided Requirements Engineering), la aplicación de cualquier metodología o proceso de ingeniería de requisitos suele fracasar por tener que realizarse manualmente. Por tanto, en este artículo presentamos el prototipo de la herramienta CARE denominada SREPTOOL, que da soporte automatizado al proceso de ingeniería de requisitos de seguridad SREP (Security Requirements Engineering Process) para facilitar su aplicación. SREPTOOL simplifica la gestión de requisitos de seguridad proporcionando una forma guiada, sistemática e intuitiva de tratarlos desde las primeras fases del desarrollo software, simplificando la gestión del repositorio de recursos de seguridad y la integración de los Criterios Comunes en el proceso de desarrollo software tal y como propone SREP.

**Palabras clave:** Ingeniería de Requisitos; Requisitos de Seguridad; CARE; Herramienta de Gestión de Requisitos; Criterios Comunes; ISO 15408.

### 1 Introducción

Hoy en día es ampliamente aceptado el principio que establece que la construcción de la seguridad en las etapas tempranas del proceso de desarrollo es más eficaz respecto a los costes y tiene como resultado diseños más robustos [14]. Por tanto, la seguridad en el software está generando cada vez mayor interés entre los ingenieros del software [26] lo cual ha ocasionado que la disciplina de la Ingeniería de Requisitos de Seguridad sea altamente considerada como parte de la Ingeniería de la Seguridad aplicada a los procesos de desarrollo de sistemas software, lo cual hasta la fecha, ha carecido de la atención necesaria [16]. La denominada Ingeniería de Requisitos de Seguridad, proporciona técnicas, métodos y normas aplicables en el ciclo de desarrollo de los SI

y que implica el uso de procedimientos repetibles y sistemáticos para asegurar que el conjunto de requisitos obtenidos es completo, consistente y fácilmente comprensible y analizable por parte de los diferentes actores implicados en el desarrollo del sistema a fin de desarrollar Sistemas de Información seguros [15].

A pesar de todas estas consideraciones, todavía existen muchas organizaciones en la actualidad que tienden a prestar poca atención a los requisitos de seguridad. Uno de los motivos es la carencia de herramientas CARE (Computer-Aided Requirements Engineering) que soporten la aplicación de los métodos o metodologías o procesos de ingeniería de requisitos de seguridad, lo que suele implicar que, tal y como se describe en [2], la implantación de este tipo de procesos suele fracasar por tener que realizarse manualmente.

Después de haber realizado un análisis comparativo de las propuestas existentes sobre requisitos de seguridad y herramientas CARE que los soporten en [20], concluimos que aunque en los últimos años se ha planteado una gran cantidad de propuestas, ninguna de las identificadas alcanzaba un nivel deseado de integración en el ciclo de desarrollo software, ni facilitaban soporte metodológico intuitivo y sistemático para la gestión de requisitos de seguridad a fin de desarrollar sistemas de información seguros y conformes a los estándares de seguridad actualmente más relevantes (como ISO/IEC 15408 [9] principalmente así como ISO/IEC 27001 [11], ISO/IEC 17799 [10] o ISO/IEC 21827 [8]) en lo relativo a la gestión de requisitos de seguridad. Con este fin y partiendo del anteriormente definido concepto de Ingeniería de Requisitos de Seguridad, propusimos el proceso SREP (Security Requirements Engineering Process) [21].

En este artículo describimos el prototipo de una herramienta de gestión de requisitos de seguridad denominada SREPTOOL, que hemos desarrollado para dar soporte automatizado a la aplicación de SREP. SREPTOOL proporcionará una forma guiada, sistemática e intuitiva para la aplicación del proceso de ingeniería de requisitos de seguridad SREP, asimismo posibilita una sencilla integración con los demás requisitos y con las distintas fases del ciclo de desarrollo, así como facilita el cumplimiento del estándar IEEE 830:1998 [6], ayudándose para ello de las funcionalidades que ofrece 'IBM Rational RequisitePro' (herramienta CARE que extiende SREPTOOL). Además, este prototipo ayuda en que los sistemas de información desarrollados sean conformes a los estándares de seguridad anteriormente mencionados en lo relativo a la gestión de requisitos de seguridad, sin la necesidad de dominar dichos estándares y reduciendo la participación de expertos de seguridad para conseguirlo, es decir, mejora la eficiencia de SREP. Y adicionalmente, gracias al Repositorio de Recursos de Seguridad que integra SREPTOOL, se facilita la reutilización de artefactos, mejorándose por ende la calidad sucesivamente.

El resto del artículo está organizado de la siguiente forma: en la sección 2, resumimos algunas características básicas de SREP con el fin de comprender la exposición posterior de la herramienta. A continuación, en la sección 3, ofrecemos una comparación de las herramientas CARE y mostraremos el marco comparativo que hemos usado para realizar el análisis de cada una de estas propuestas, y las consecuentes decisiones adoptadas. Posteriormente, en la sección 4, se establecen las necesidades concretas relativas a la ingeniería de requisitos de seguridad que una herramienta CARE tendría que soportar, además describimos la funcionalidad de

SREPTOOL y cómo ha sido implementada. Finalmente en la sección 5 expondremos nuestras conclusiones, junto con las aportaciones de SREPTOOL y el trabajo futuro.

## 2 El Proceso de Ingeniería de Requisitos de Seguridad SREP

SREP [21] es un proceso basado en activos y dirigido por el riesgo para el establecimiento de requisitos de seguridad en el desarrollo de SI seguros. Básicamente este proceso describe cómo integrar los Criterios Comunes (CC) [9] en el ciclo de desarrollo junto con el uso de un repositorio de recursos de seguridad para facilitar la reutilización de requisitos, activos, amenazas, test y contramedidas. Asimismo, facilita los distintos tipos de trazabilidad (según los conceptos de trazabilidad en [22] que se basan en [4, 15]): pre-trazabilidad y post-trazabilidad; la trazabilidad hacia atrás y hacia delante; las relaciones de trazabilidad entre requisitos y las relaciones de los requisitos con otros artefactos.

Este proceso está centrado en la construcción de conceptos de seguridad en las primeras fases del desarrollo. De manera genérica se puede describir como un 'add-in' de actividades (que se descomponen en tareas, donde se generan artefactos de entrada y salida, y con la participación de distintos roles) que se integran sobre el modelo actual de cualquier Organización dándole un enfoque en ingeniería de requisitos de seguridad. Como se describe en [21], nosotros hemos descrito más detalladamente cómo SREP se integra en el ciclo de vida del Proceso Unificado [12], que como sabemos está dividido en una secuencia de fases y cada fase puede implicar varias iteraciones. De esta manera, el modelo elegido por SREP es un modelo de proceso en espiral y los requisitos de seguridad y sus artefactos asociados (amenazas, etc.) evolucionan a lo largo del ciclo de vida y se tratan a la vez que los otros requisitos funcionales y no funcionales y demás artefactos del proceso de desarrollo software. Al mismo tiempo, los Componentes de los CC se introducen en el ciclo de vida de desarrollo software, de manera que SREP usa los diferentes componentes según la fase del ciclo de vida en que se esté y la actividad de SREP correspondiente, aunque las tareas de aseguramiento de la calidad se realizan durante todas las fases y son en estas tareas donde la mayoría de los requisitos de aseguramiento de los CC se incorporan.

El Repositorio de Recursos de Seguridad (RRS) facilita el desarrollo con reutilización de requisitos, lo cual incrementa su calidad, ya que las inconsistencias, errores, ambigüedades y otros problemas se pueden detectar y corregir en proyectos sucesivos [25]. Un meta-modelo de repositorio, describiendo la organización del RRS se muestra en la Fig. 1. Se trata de un meta-modelo dirigido por activos así como por amenazas y objetivos, porque los requisitos pueden obtenerse a través de los objetivos de seguridad o de las amenazas partiendo de los activos.

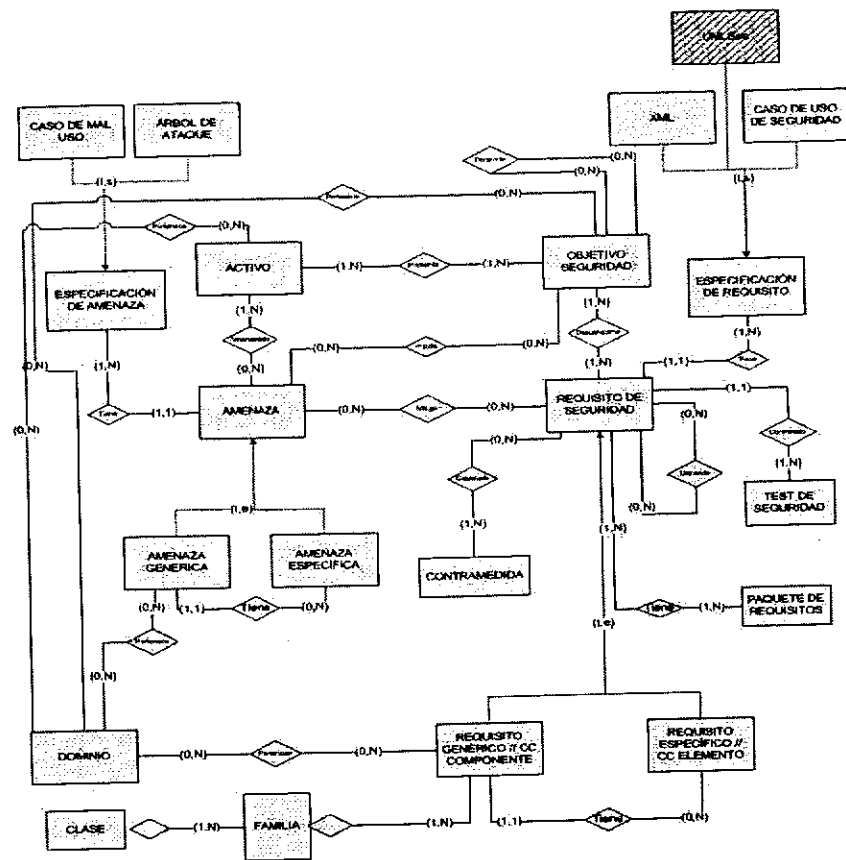


Fig. 1 Meta-modelo del Repositorio de Recursos de Seguridad

### 3. Comparativa de Herramientas CARE

En primer lugar, hubo que tomar la decisión inicial de desarrollar una herramienta nueva o extender una existente, y dadas las características del proceso SREP y los objetivos que pretende su aplicación, se consideró más adecuado extender una herramienta existente y centrarse en el campo de las herramientas CARE, descartándose herramientas de gestión documental o de contenidos (CMS, Content Management Systems), así como otro tipo de herramientas CASE (Computer-Aided Software Engineering) centradas en otras fases del ciclo de vida.

Por tanto, en esta sección, con el objetivo de obtener una visión general de las herramientas CARE, se expone un resumen del estado del arte de estas herramientas. Para lo cual, primeramente, se identifican las herramientas CARE existentes y se seleccionan las que se estudiarán en profundidad; a continuación se realizará una comparación de las seleccionadas anteriormente, para lo cual, se establece un marco de análisis. Para la definición del marco de análisis que nos facilitara la adecuada

selección de una herramienta CARE se tuvo en cuenta tanto las necesidades concretas que la herramienta CARE tendría que soportar para la correcta aplicación del proceso SREP, como los requisitos generales de una buena herramienta CARE. Para lo cual nos basamos en [5], al que consideramos como uno de los catálogos de requisitos de herramientas CARE más completos y concretos; así como en la encuesta de INCOSE (International Council on Systems Engineering) [7], considerada como uno de los análisis más importantes y completos de requisitos de herramientas CARE; y en [17], donde se realiza un análisis de herramientas CARE desde la perspectiva de los requisitos de seguridad.

Basándonos en la encuesta de INCOSE, realizamos una primera selección de herramientas que cumplieran con la mayoría de las funciones y que han demostrado con su grado de penetración en el mercado que proporcionan soluciones efectivas, tal y como se recoge en diversos estudios al respecto, como [1, 7, 24]. Nuestra lista seleccionada fue: RequisitePro, IRqA, DOORS y Caliber-RM.

En la Tabla 1 se resume el análisis comparativo realizado sobre las anteriores herramientas, utilizando como criterios de comparación aquellas características consideradas necesarias para la satisfactoria aplicación del proceso SREP.

Tabla 1 Resumen de la comparativa de herramientas CARE

	RequisitePro	IRqA	DOORS	Caliber-RM
Extensibilidad de la funcionalidad	Si, API basado en COM	Si, API basado en COM y JAVA	Si, API lenguaje DXL	Si, API basado en COM y JAVA
Trazabilidad	Si, entre los tipos de requisitos.	Si, entre tipos de requisitos, conceptos, UML, código, test.	Si, entre cualquier elemento del repositorio	Si, entre los tipos de requisitos y otros elementos.
Integración con otras herramientas del ciclo de vida	Si, con IBM Rational tools (Rational Rose, Rational TestManager, and Rational ClearQuest, MSProject, etc)	Si, con Office, IRqA-Rational Rose, Mercury TestDirector, CVS.	Si, mediante el lenguaje DXL.	Si, con Office, Project, Modeling, Testing e IDE tools. (Ej. Mercury TestDirector)
Soporte reutilización	No	No	No	No
Repositorio del proyecto	MS-Access, MS-SQL Server, Oracle	MS-SQL Server, Oracle, Informix, MySQL	Propietario	MS-Access y MS-SQL Server
Validación de la especificación	Si, con matriz de trazabilidad	Si, con matriz de trazabilidad	Si, con matriz de trazabilidad	Si, con matriz de trazabilidad
Estándares de especificación	Si, la salida puede adaptarse a las plantillas que se definan	Si, la salida puede adaptarse a diferentes formatos/plantillas	Si, la salida puede adaptarse a diferentes formatos	Si, la salida puede adaptarse a diferentes formatos
Experiencia previa. Facilidad uso e Interfaz de usuario	Si. Requisitos, vistas y documentos en una sola vista. Acceso web colaborativo	No. Organizada en vistas.	No. Difícil de seguir. Módulos y objetos	No. Práctica y amigable, orientada al entorno web
Importación de requisitos	Si, Word y CSV	Si, Word, CSV, Excel, XML	Si, Word y ficheros delimitados	Si, Word y ficheros delimitados

Control de versiones y líneas base	Si pero no comparables	Si y comparables	Si y comparables	Si y comparables
Control acceso por roles y usuarios	Si	Si	Si	Si
Requisitos parametrizados	No	No	No	No

Estas herramientas proporcionaban casi todas las necesidades exigibles para una herramienta CARE. Sin embargo observamos con el análisis realizado que ninguna de ellas cubría todas las necesidades para dar soporte automatizado a SREP. De hecho, alguna de las limitaciones eran críticas para una satisfactoria aplicación de SREP, por lo que la capacidad de extensibilidad de las herramientas se consideró un factor clave de éxito. Destacan entre otras debilidades encontradas, el hecho de que ninguna de las herramientas posibilitaba la parametrización. Al igual, que tampoco facilitaban técnicas adecuadas para la especificación de requisitos de seguridad (como UMLSec [13] o casos de uso de seguridad [3]), ni de las amenazas (casos de mal uso [23]); ni facilitaban soporte metodológico automatizado para la gestión de requisitos de seguridad, careciendo de soporte para actividades fundamentales, como la valoración del riesgo; ni facilitando la conformidad de los estándares de seguridad actualmente más relevantes relativos a requisitos de seguridad (como ISO/IEC 15408, ISO/IEC 27001, ISO/IEC 17799 o ISO/IEC 21827).

En definitiva, se decidió elegir extender RequisitePro como soporte para nuestro prototipo, debido fundamentalmente a los siguientes factores:

- Extensibilidad. RequisitePro facilita un API basada en COM que permite acceder a los datos almacenados en éste (proyectos, requisitos, atributos, etc.) tanto para consultarlos como para modificarlos. Así como controlar la interfaz de usuario de RequisitePro y también los documentos de Microsoft Word. Lo cual, a pesar de ser algo más limitada que en las otras herramientas, nos resultaba más claro y sencillo de adaptar a las necesidades de SREPTOOL.
- Integración automatizada con el resto de actividades del ciclo de vida. RequisitePro al estar integrada en el paquete "Rational Suite AnalystStudio" facilitaba un aspecto clave para SREP, la integración no solo con los otros requisitos, sino con otros artefactos del ciclo de vida (como su integración con elementos de modelado de Rational Rose).
- Experiencia previa. La herramienta RequisitePro ha sido ampliamente utilizada como herramienta de soporte en proyectos previos al desarrollo de SREPTOOL (utilizándose por ejemplo en [19]), al ser la herramienta corporativa de la Gerencia de Informática de la Seguridad Social (organismo al que pertenece el primer autor), lo que puede resultar muy interesante para la realización de futuros casos de estudio reales de SREPTOOL.
- Facilidad de uso y Multiusuario. Una de sus características más destacadas es su integración con el procesador de textos Microsoft Word, así como ver todas sus funciones en una sola vista. Además proporciona la posibilidad de acceso multiusuario al proyecto y una interfaz web colaborativa.

- Trazabilidad. RequisitePro permite la creación de relaciones de trazabilidad entre distintos tipos de requisitos, y se visualiza a través de una matriz de trazabilidad.
- Otros factores destacables. RequisitePro permite cierta reutilización utilizando plantillas de documentos. Asimismo, su repositorio está basado en una base de datos relacional comercial (MS-Access, Oracle, MS-SQLServer) y ofrece control de versiones de los requisitos.

#### 4. La Herramienta SREPTOOL

El prototipo que se presenta es una primera aproximación que servirá para obtener experiencia del problema mediante su aplicación en escenarios de uso y casos de estudio, para así refinarlo y obtener una versión definitiva de SREPTOOL.

##### 4.1. Tecnología utilizada

Para la creación del prototipo se ha utilizado el lenguaje de programación Visual Basic 6, produciendo como artefacto de salida una biblioteca *dll ActiveX*, que será enlazada con RequisitePro. De esta manera, los objetos de RequisitePro serán visibles desde SREPTOOL y por otro lado, los artefactos generados por el prototipo serán visibles desde RequisitePro. Así, la funcionalidad del prototipo estaría accesible desde la ventana principal de Rational RequisitePro, a través del menú *Tools* → *SREPTOOL*. Para la integración con RequisitePro, SREPTOOL se ha desarrollado como un add-in de dicha herramienta CARE, para lo cual se ha utilizado la interfaz de extensibilidad de RequisitePro, en concreto el *RequisitePro Extensibility Server (RPX)* que permite acceder a los datos almacenados en RequisitePro y el *RqGUILApp library* que controla la interfaz de usuario del RequisitePro y permite también controlar los documentos de Microsoft Word. Además, utilizará dos bases de datos (para este prototipo son MS-Access). La primera contendrá la información de los usuarios y estará cifrada. La segunda será el repositorio y su acceso estará controlado mediante contraseña.

##### 4.2. Funcionalidad del prototipo

El prototipo de SREPTOOL permite la aplicación del proceso SREP en el desarrollo de un proyecto dando soporte automatizado a sus nueve actividades. A continuación se explica el funcionamiento del prototipo y cómo SREPTOOL facilita la realización de cada una de estas actividades, ya que va guiando la actuación de los distintos roles y responsables, así como ayuda en la generación de la documentación necesaria. Para lo cual nos ayudaremos de escenarios de uso sencillos para ilustrarlas (extraídos de [19]), aunque debido a las restricciones de espacio no se explicarán en profundidad, ni se mostrarán todas las interfaces del prototipo.

Una vez arrancado RequisitePro, si se pulsa en el menú Tools, se puede observar la opción SREPTOOL. Seguidamente, y ya también arrancado el gestor de base de datos, el usuario deberá autenticarse y seleccionar un rol para poder arrancar el prototipo. Dependiendo del rol con el que el usuario se identifique ante el sistema, dicho usuario podrá realizar unas determinadas actividades tal y como se determina en los roles que se determinan en SREP [21], por ejemplo el responsable de la actividad (el ingeniero de requisitos, normalmente) junto con el asegurador de calidad validarán los artefactos en cada actividad y permitirán o no pasar a la siguiente actividad o volver a alguna anterior. Una vez autorizado, el usuario, y según sus permisos, podrá elegir entre crear un nuevo proyecto o abrir un proyecto ya existente. Esta segunda opción representará la situación en la que ya exista otra iteración de SREP guardada y el usuario desee realizar un refinamiento.

En la Fig. 2, se observa la interfaz principal de SREPTOOL. A través del menú Ver, se permite ver los distintos documentos generados. Hay una pestaña por cada actividad de SREP y tres botones en la parte inferior comunes para todas las pestañas, que permiten validar la actividad, generar el informe de la actividad o salir.

#### Actividad 1: Acuerdo de las definiciones

Tal y como se observa en Fig. 2, el usuario puede seleccionar los conceptos y definiciones de seguridad que desea tener en cuenta para su proyecto. Para ello tan solo tiene que seleccionar el estándar y dentro de él los conceptos que le resulten necesarios. Por otro lado, en esta actividad se seleccionan los stakeholders (partes interesadas o participantes) de entre el personal disponible, asignándole a cada uno de ellos el rol que va a desempeñar. Además en esta primera actividad el usuario puede definir el nivel de aseguramiento que desea aplicar al proyecto en desarrollo, así como recoger los artefactos de entrada de SREP en la Visión de Seguridad, como la Política de Seguridad Organizacional. Finalmente, al pulsar el botón "Generar Informe", se crea el "Documento de Visión de Seguridad" de manera automática con los datos que el usuario ha introducido en esta actividad.

#### Actividad 2: Identificación de activos

En esta actividad el usuario podrá seleccionar el Dominio de seguridad al cuál pertenece el proyecto en desarrollo. En función de dicho dominio, podrá seleccionar aquellos activos que considere relevantes para su proyecto. Por otro lado, el usuario podrá crear un nuevo Dominio y agregarle nuevos activos creados o activos pertenecientes a otros dominios relacionados.

#### Actividad 3: Identificación de objetivos de seguridad

En esta actividad el usuario puede elegir para cada uno de los activos seleccionados en la actividad anterior, uno/s de los objetivos de seguridad de los que ese activo tenga registrados en el repositorio. Además para cada uno de los objetivos de seguridad introducidos en el proyecto, el usuario puede establecer una valoración de dichos objetivos. Por otro lado, esta actividad facilita la creación de nuevos objetivos de seguridad por parte del usuario, así como la creación de dependencias entre dichos objetivos. El usuario puede crear un nuevo objetivo de seguridad y asociárselo tanto a alguno de los activos como al dominio de la aplicación. Finalmente, si se pulsa el

botón "Generar Informe", se plasmarán todos los datos en el "Documento de Objetivos de Seguridad".

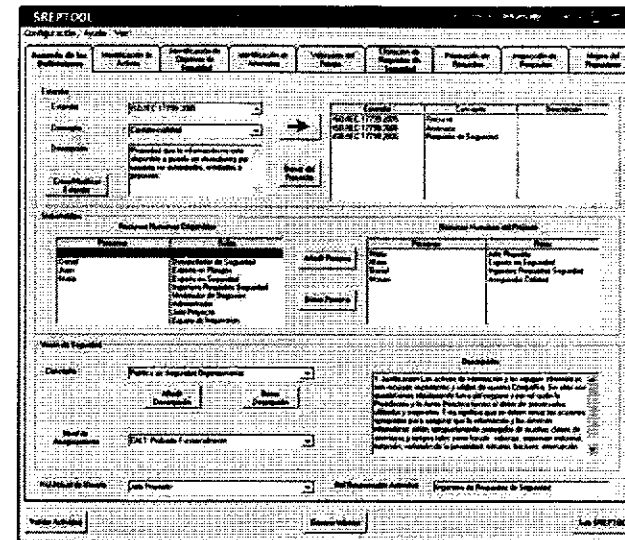


Fig. 2 Interfaz de la Actividad 1 de SREPTOOL

#### Actividad 4: Identificación de amenazas

En esta actividad el usuario podrá ver las amenazas que tienen asociadas los activos de su proyecto y/o las amenazas que tienen asociadas los objetivos de seguridad. De esta manera, podrá seleccionar para su proyecto aquellas amenazas que considere de mayor relevancia. Como se observa en la Fig. 3, el usuario habría seleccionado para el activo "Información personal de nivel alto" la amenaza "Alteración no autorizada de información".

Sin embargo, pudiera ser que las amenazas que el usuario quiera asociar a los activos u objetivos de su proyecto no se encontrasen en el Repositorio de Recursos de Seguridad. En este caso, el usuario podrá introducir una nueva amenaza mediante la instanciación de un "nuevo Caso de Mal Uso" o de un "nuevo Árbol de Ataque", al pinchar sobre dichos botones. De manera que rellenando una plantilla, podrá especificar: Nombre e Id del Caso de mal Uso; Casos de Uso Relacionados; Probabilidad de que se produzca la amenaza; Resumen, Precondiciones y Postcondiciones; Interacciones que se producen en el Caso de mal Uso; Tipo de amenaza (Genérica o Específica). Finalmente, si el usuario pulsa en esta actividad el botón de generación de informe, se creará el documento de "Definición del Problema de Seguridad".

#### Actividad 5: Valoración del riesgo

Una vez que se han identificado las amenazas en la actividad anterior, ahora el usuario puede estimar el impacto que produce la materialización de cada una de ellas. Una vez calculado el impacto que produce la amenaza, se puede estimar el riesgo aproximado de dicha amenaza teniendo en cuenta la frecuencia con que dicha amenaza se manifiesta. Para realizar el cálculo del impacto y el riesgo, el prototipo utiliza

una de las técnicas propuestas por MAGERIT v.2 [18] basada en el análisis mediante tablas. Por ejemplo, si el usuario ha calculado el impacto que tiene la amenaza "Alteración no autorizada de información". Dado que la degradación que produce la amenaza es "Alta" y la valoración del objetivo de seguridad también es "Alto", el resultado es un impacto "Muy Alto". A continuación si el usuario ha seleccionado una frecuencia para la amenaza de valor "Poco Frecuente". El resultado será un riesgo de valor "Alto". Por último, en esta actividad se genera el "Documento de Valoración del Riesgo" al pulsarse el botón "Generar Informe".

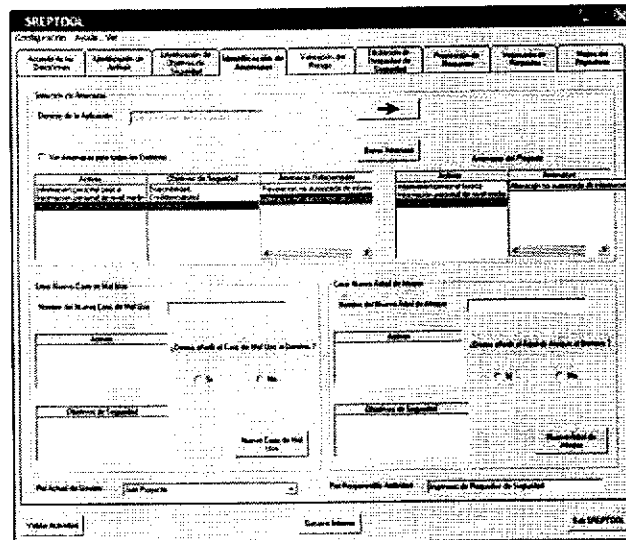


Fig. 3 Interfaz de la Actividad 4

#### Actividad 6: Elicitación de Requisitos de Seguridad

Esta actividad es una de las principales de SREPTOOL. Una vez se han seleccionado las amenazas relevantes para el proyecto, el usuario puede seleccionar aquellos requisitos de seguridad que quiere implantar. Para ello el usuario tiene tres opciones como se observa en la Fig. 4:

- Teniendo seleccionada una amenaza, el prototipo mostrará los requisitos que hay en el RRS para esa amenaza. El usuario solo tendrá que seleccionar aquellos requisitos que considere relevantes.
- Seleccionando una clase y una de sus familias, el prototipo mostrará los requisitos de seguridad asociados a dicha familia. El usuario podrá seleccionar y añadir a su proyecto los requisitos deseados.
- Seleccionando uno de los paquetes de requisitos y dentro del paquete aquellos requisitos deseados.

Por otro lado, el usuario puede introducir nuevos requisitos de seguridad que no se encuentren en el repositorio mediante una plantilla. En la que podrá introducir: El Nombre e Id del Caso de Uso Seguro; Aquellas amenazas y objetivos con las que está relacionado el requisito que instancia el caso de uso seguro; Los requisitos de seguridad que son excluyentes respecto al requisito de seguridad que se está instanciando;

El tipo de requisito de seguridad (genérico o específico) y en caso de que sea específico, a que requisito genérico pertenece; La clase, familia y paquete de requisitos a los que se quiere asociar el nuevo requisito de seguridad; Precondiciones, Postcondiciones e Interacciones del caso.

Además, el usuario puede seleccionar o crear Contramedidas o Test de Seguridad. Finalmente, con el botón "Generar Informe" se crea el "Documento de Especificación de Requisitos de Seguridad".

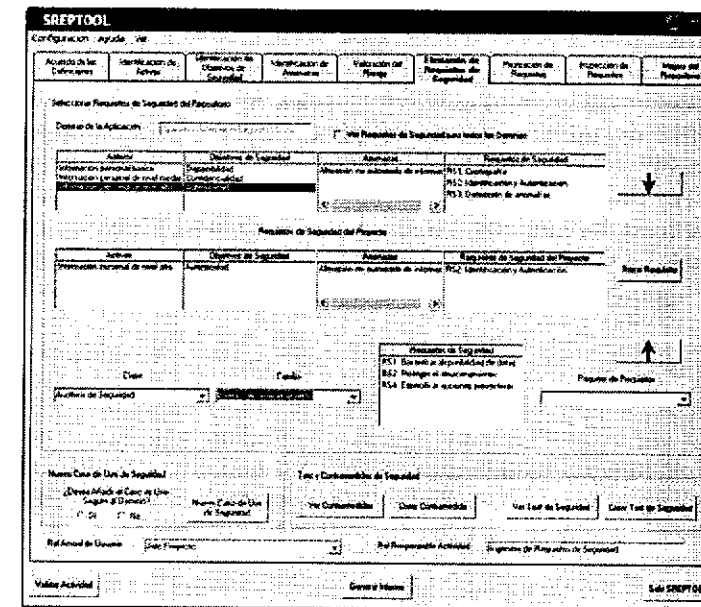


Fig. 4 Interfaz Actividad 6

#### Actividad 7: Priorización

Esta actividad tiene por objetivo automatizar la priorización de requisitos de seguridad en función del riesgo de las amenazas que mitigan. Para cada uno de los requisitos de seguridad establecidos en el proyecto, el usuario podrá seleccionar el valor de prioridad que desea asignarle (Crítico, Estándar, Óptimo). Una vez seleccionadas las prioridades de todos los requisitos, al pulsar el botón priorizar se ordenarán los requisitos de mayor a menor prioridad.

#### Actividad 8: Inspección de requisitos

En esta actividad, el prototipo facilita al usuario la verificación y validación de los requisitos de seguridad, mediante, la comprobación de aquellas amenazas para las cuales no se han especificado requisitos de seguridad en el proyecto, así como, los requisitos de aseguramiento que no han sido añadidos al proyecto de acuerdo con el nivel de aseguramiento que se definió en la actividad 1. En la Fig. 5 se aprecia cómo el prototipo nos indica que para la amenaza "Revelación no autorizada de información", el usuario no ha especificado ningún requisito de seguridad. Por otro lado, SREPTOOL muestra cuatro requisitos de aseguramiento pendientes, que de acuerdo

al nivel EAL1, el usuario no había añadido a su proyecto. Por último, el prototipo permite la generación automática del "Informe de Validación" y el "Documento de Fundamentación de los Requisitos de Seguridad".

#### Actividad 9: Mejora del repositorio.

En esta última actividad, el prototipo permite añadir al repositorio (RRS) todos aquellos recursos de seguridad que el usuario haya creado durante la iteración. SREPTOOL presenta en la parte superior y clasificados por categorías, todos aquellos recursos de seguridad que han sido creados nuevos. El usuario (en este caso el equipo de inspección o el asegurador de la calidad) ha de seleccionar aquellos que considere interesantes de ser introducidos en el RRS y pulsar el botón "Añadir al Repositorio". De esta manera, en la parte inferior se mostrará el número de elementos de cada categoría que se han creado en la iteración, y cuantos de ellos han sido introducidos en el repositorio. Por último, el prototipo generará el "Documento de Declaración de Seguridad" conforme a los Criterios Comunes (ISO/IEC 15408), que contendrá toda la información de los artefactos generados por SREPTOOL en anteriores actividades.

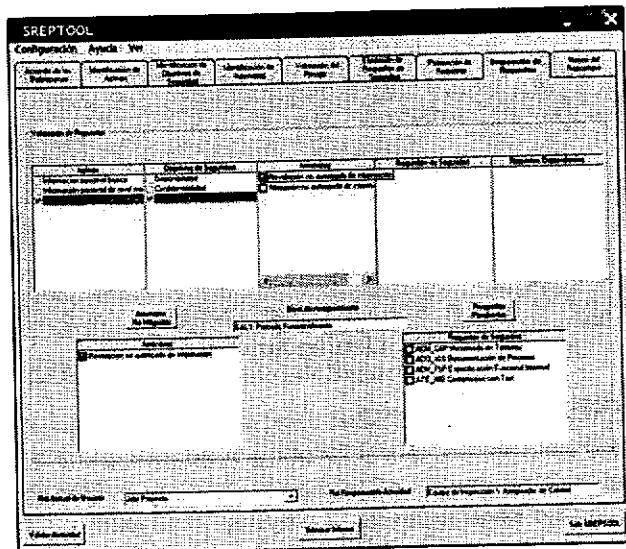


Fig. 5 Interfaz Actividad 8

## 5. Conclusiones y Trabajo Futuro

En nuestros días, la seguridad en el software está generando cada vez mayor interés. Existen diversas herramientas CARE interesantes, algunas de ellas han sido descritas y comparadas en este trabajo, aunque presentan algunas limitaciones en lo relativo a la gestión de requisitos de seguridad expuestas anteriormente.

Por ello, el prototipo que se presenta (SREPTOOL) proporciona una forma guiada, sistemática e intuitiva para la aplicación del proceso de ingeniería de requisitos de seguridad SREP, asimismo posibilita una sencilla integración con los demás requisi-

tos y con las distintas fases del ciclo de desarrollo, así como facilita el cumplimiento del estándar IEEE 830:1998, ayudándose para ello de las funcionalidades que ofrece 'IBM Rational RequisitePro'. Además, este prototipo facilita que los sistemas de información desarrollados sean conformes a los estándares de seguridad actualmente más relevantes relativos a la gestión de requisitos de seguridad (como ISO/IEC 15408, ISO/IEC 27001, ISO/IEC 17799 o ISO/IEC 21827), sin la necesidad de dominar dichos estándares y reduciendo la participación de expertos de seguridad para conseguirlo. Asimismo, gracias al Repositorio de Recursos de Seguridad que integra, se facilita la reutilización de artefactos, mejorándose por ende la calidad sucesivamente.

Además, existe un conjunto de aspectos proyectados para el futuro del prototipo que permitirán aumentar el nivel de automatización de la aplicación de SREP y por tanto una mejor eficacia en el proceso de ingeniería de requisitos, entre los cuales destacamos los siguientes: extender el tipo de especificaciones de requisitos soportadas, para que soporte UMLSec [13]; extender la herramienta para que pueda ser soportada en otras herramientas CARE; automatizar la creación de los casos de uso de seguridad utilizando los casos de mal uso creados en la actividad 4 de SREP.

## Agradecimientos

Este artículo es parte del proyecto ESFINGE (TIN2006-15175-C05-05) del Ministerio de Educación y Ciencia, y de los proyectos MISTICO (PBC-06-0082) y DIMENSIONS (PBC-05-012-2) de la Consejería de Ciencia y Tecnología de la Junta de Comunidades de Castilla-La Mancha y el FEDER.

## Referencias Bibliográficas

1. Atlantic\_Systems, *Requirements tools* (<http://www.volere.co.uk/tools.htm>). 2006.
2. Davis, A., *Tracing: A Simple Necessity Neglected*. IEEE Software, 12(5) (1995).
3. Firesmith, D.G., *Security Use Cases*. Journal of Object Technology, 2003: p. 53-64.
4. Gotel, O.C.Z. and Finkelstein, A.C.W. *An analysis of the requirements traceability problem*. in *First International Conference on Requirements Engineering (ICRE'94)*. 1994: IEEE CS Press.
5. Hoffmann, M., Kühn, N., and Bittner, M. *Requirements for Requirements Management Tools*. in *12th IEEE International Requirements Engineering Conference (RE'04)*. 2004. Kyoto, Japan.
6. IEEE, *IEEE 830: 1998 Recommended Practice for Software Requirements Specifications*. 1998.
7. INCOSE, *The International Council on Systems Engineering Requirements Management Tools Survey* (<http://www.incose.org>). 2006.
8. ISO/IEC, *ISO/IEC 21827:2002 Information technology -- Systems Security Engineering -- Capability Maturity Model (SSE-CMM)*. 2002.
9. ISO/IEC, *ISO/IEC 15408:2005 Information technology - Security techniques - Evaluation criteria for IT security, (Common Criteria v3.0)*. 2005.



10. ISO/IEC, *ISO/IEC 17799 Information technology - Security techniques - Code of practice for information security management*. 2005.
11. ISO/IEC, *ISO/IEC 27001:2005 Information technology -- Security techniques -- Information security management systems -- Requirements*. 2005.
12. Jacobson, I., Booch, G., and Rumbaugh, J., *The Unified Software Development Process*. 1999, Boston: Addison-Wesley Longman Publishing Co.
13. Jürjens, J., *UMLsec: extending UML for secure systems development*. UML 2002 - The Unified Modeling Language. Model Engineering, Languages, Concepts, and Tools. 5th International Conference., 2002. LNCS 2460: p. 412-425.
14. Kim., H.-K., *Automatic Translation Form Requirements Model into Use Cases Modeling on UML*. ICCSA 2005, LNCS, 2005: p. 769-777.
15. Kotonya, G. and Sommerville, I., *Requirements Engineering Process and Techniques*. Hardcover ed. 1998, UK: John Willey & Sons. 294.
16. Lamsweerde, A.v. *Elaborating Security Requirements by Construction of Intentional Anti-Models*. in *26th International Conference on Software Engineering*. 2004. Edinburgh: ACM-IEEE.
17. Lasheras, J., Toval, J.A., Nicolas, J., and Moros, B., *Soporte Automatizado a la reutilización de requisitos*. Jornadas de Ingeniería del Software y Bases de Datos (JISBD 2003), 2003: p. 335-346.
18. López, F., Amutio, M.A., Candau, J., and Mañas, J.A., *Methodology for Information Systems Risk Analysis and Management*. 2005: Ministry of Public Administration.
19. Mellado, D., Fernández-Medina, E., and Piattini, M., *Applying a Security Requirements Engineering Process*. 11th European Symposium on Research in Computer Security (ESORICS 2006), 2006. Springer LNCS 4189: p. 192-206.
20. Mellado, D., Fernández-Medina, E., and Piattini, M., *A Comparative Study of Proposals for Establishing Security Requirements for the Development of Secure Information Systems*. The 2006 International Conference on Computational Science and its Applications (ICCSA 2006), Springer LNCS 3982, 2006. 3: p. 1044-1053.
21. Mellado, D., Fernández-Medina, E., and Piattini, M., *A Common Criteria Based Security Requirements Engineering Process for the Development of Secure Information Systems*. Computer Standards and Interfaces, 2007. 29(2): p. 244 - 253.
22. Pinheiro, F.A., *Requirements Traceability*, in *Perspectives on software requirements*, Sampaio, J.C. and Horacio, J., Editors. 2004.
23. Sindre, G. and Opdahl, A.L., *Eliciting security requirements with misuse cases*. Requirements Engineering 10, 2005. 1: p. 34-44.
24. TCP, *Comparative Study Between Requirements Management and Engineering Tools* (<http://www.irqaonline.com/downloadarea/documents.htm>). 2004.
25. Toval, A., Nicolás, J., Moros, B., and García, F., *Requirements Reuse for Improving Information Systems Security: A Practitioner's Approach*. Requirements Engineering, 6(4) (2002). p. 205-219.
26. Viega, J. and McGraw, G., *Building Secure Software: How to Avoid Security Problems the Right Way*. 2002, Boston: Addison-Wesley.

## Sesión 2

### Modelado Organizacional