

Competencias para desempeñar la labor de captura de requisitos en un entorno de desarrollo global del software

Miguel Romero¹, Aurora Vizcaíno², Mario Piattini²

¹Universidad del Bío-Bío,
Departamento de Ciencias de la Computación y Tecnologías de la Información,
Avenida Andrés Bello s/n 3780000 Chillán, Chile.
mromero@pehuen.chillan.ubiobio.cl

²Universidad de Castilla-La Mancha
Grupo Alarcos – Instituto de Tecnologías y Sistemas de Información
Dep. de Tecnologías y Sistemas de Información – Escuela Superior de Informática
{Aurora.Vizcaino, Mario.Piattini}@uclm.es

Resumen

La captura de requisitos de un software es un proceso eminentemente comunicativo, por consiguiente, una buena o mala comunicación influye en la calidad final del producto desarrollado. Los problemas típicos de esta etapa se ven incrementados cuando los participantes del proceso se encuentran distribuidos geográficamente bajo el paradigma del Desarrollo Global del Software (GSD en inglés). Cumplir el desafío de ejecutar dicha actividad de manera exitosa requiere de una adecuada preparación de los profesionales involucrados en la captura de requisitos. En este contexto hemos revisado la bibliografía en busca de las habilidades y conocimientos necesarios para desempeñar la labor de captura de requisitos. Luego de un proceso de análisis y síntesis, identificamos las competencias necesarias para la captura de requisitos presentando los resultados en tres grupos de interés: las competencias que son igualmente importantes en captura localizada y en GSD, las competencias que aumentan de importancia en entornos globales y las competencias que sólo son relevantes en escenarios GSD. Estas competencias pueden ser utilizadas tanto, para el desarrollo de curriculums adaptados a GSD, como en la industria, para el reclutamiento de personal.

1. Introducción

El Desarrollo Global del Software (GSD) [17], [18] es una de las tendencias actuales en el

ámbito de la ingeniería del software. El GSD ha tenido un crecimiento considerable en los últimos años debido a la cultura de la globalización [8], [11] y a otros factores tales como la deslocalización (*off-shoring*) [7], [10], y es de esperar que continúe creciendo ya que para las empresas es una manera de disminuir costos manteniendo el nivel de calidad. Como lo indican en [4]: “La globalización y deslocalización en el interior de la industria del software seguirá y, de hecho, aumentará. Este aumento será impulsado por la tecnología de la información, así como la acción de los gobiernos y de los factores económicos y se traducirá en una mayor competición mundial tanto en las habilidades de software de nivel más bajo, así como en los esfuerzos de nivel más alto tal como la investigación. Los datos actuales y la teoría económica sugieren que, a pesar de la deslocalización, las oportunidades de carrera en IT seguirán siendo fuertes en los países que ha sido fuerte en el pasado, incluso a medida que crece en los países que son blancos de la deslocalización. El futuro, sin embargo, es aquel en que el individuo se encontrará con una mayor competición mundial” [4] p.13.

En consecuencia, la enseñanza de la ingeniería del software debe ser ajustada con el fin de capacitar a los estudiantes para afrontar un escenario laboral global y más competitivo. Esta cuestión no es una tarea fácil, pues la deslocalización causa que cambie el conjunto de habilidades, conocimientos y competencias que requieren los ingenieros del software [4]. Además, estos cambios no han sido identificados

de manera completa y consensuada, según lo que hemos podido revisar en la literatura.

Este trabajo tiene el objetivo de proponer las competencias necesarias para que un ingeniero del software pueda realizar una captura de requisitos de alta calidad, incluso, cuando se enfrentan a las dificultades de entornos GSD.

Con el fin de alcanzar este objetivo hemos desarrollado una revisión y análisis de la literatura que nos ha permitido identificar cuatro conjuntos de competencias:

- Las competencias que no son importantes en entornos GSD. Para este conjunto no hemos identificado competencias.
- Las competencias que son de igual importancia tanto en ambientes locales como en GSD. En este conjunto identificamos 49 competencias.
- Las competencias que son importantes en entornos locales de desarrollo cuya importancia aumenta en entornos GSD. En este conjunto identificamos 17 competencias.
- Las competencias que únicamente son importantes en entornos GSD. En este conjunto identificamos 6 competencias.

Estas competencias, que se describen en el capítulo 4, pueden servir tanto a la industria del software como a las universidades. En la industria podría ayudar en los procesos de reclutamiento de personal para decidir qué persona es la más adecuada para llevar a cabo un proceso de captura de requisitos en GSD. Y a las universidades para el diseño de cursos o módulos donde se enseñe captura de requisitos en ambientes GSD.

El resto del artículo está organizado de la siguiente manera: la sección 2, presenta conceptos fundamentales para entender el artículo; en la sección 3 explicamos el proceso de investigación usado; en la sección 4 presentamos las competencias profesionales, tanto genéricas como específicas, identificadas en este estudio; por último, en la sección 5 presentamos nuestras conclusiones y trabajo futuro.

2. Conceptos básicos

Con el fin de aclarar el concepto de captura de requisitos, en la sección 2.1 describimos en que consiste y como forma parte de la ingeniería de requisitos. En la sección 2.2 describimos la

problemática del GSD a través de los factores claves reportados en la literatura, los cuales usaremos en la definición de las competencias que presentamos en la sección 4.

Esta sección finaliza con una breve descripción de los conceptos de conocimiento, habilidades y competencias (sección 2.3), pues consideramos oportuno hacer esta distinción para aquellos lectores no familiarizados con estos conceptos y en especial con el de competencias.

A continuación describimos el proceso de captura de requisitos.

2.1. Proceso de captura de requisitos

Independiente del enfoque metodológico utilizado, la captura de requisitos es el primer paso en el proceso de desarrollo de un producto software. En la literatura, este proceso ha sido llamado de diferentes maneras, por ejemplo: Captura de requisitos (*Requirements capture*), descubrimiento de requisitos (*requirements Discovery*), captura de requisitos (*requirements elicitation*) y adquisición de requisitos (*requirements acquisition*) [29]. Este proceso es parte de la ingeniería de requisitos junto con el análisis, modelado, especificación y revisión de requisitos.

El proceso de captura de requisitos tiene como finalidad la comprensión del problema para el cual el software es requerido con el fin de satisfacer las expectativas y necesidades de los clientes y usuarios [29]. Éste es un proceso fundamentalmente comunicacional entre el (o los) especialista de requisitos y los involucrados de la contraparte del cliente [29].

No es fácil delimitar la frontera de la captura de requisitos, pero en este trabajo consideramos que la captura está centrada en la obtención de las necesidades del cliente y que el modelado y la especificación de requisitos es una etapa diferente.

2.2. Problemática del GSD

La problemática del GSD afecta fuertemente la comunicación entre los participantes (*stakeholders*) del proceso de desarrollo y, por tanto, fuertemente a la ingeniería de requisitos y en particular al proceso de captura. Un estudio relevante sobre los factores críticos del GSD y como estos afectan a las etapas de la ingeniería de requisitos es presentado en [14] a partir del cual se

concluye que todos los factores críticos conocidos para GSD afectan a la captura de requisitos.

La siguiente lista resume los principales desafíos en GSD que son presentados en la literatura:

- **Diferencias Culturales.** Las diferencias culturales pueden afectar a los proyectos GSD de diferentes maneras incluyendo la efectividad de la comunicación y coordinación, la toma de decisiones de grupo y el desempeño de los equipos [18], [12], [14], [23], [28].
- **Comunicación inadecuada.** En un entorno global la comunicación es un desafío por diversos factores. Factores culturales, diferencias de lenguaje, la distancia, los diferentes husos horarios, dificultan la interacción entre las personas, por tanto los participantes deben realizar esfuerzos adicionales para lograr una comunicación efectiva. [6], [13], [18].
- **Diferentes husos horarios.** La distribución en puntos geográficos distantes hace que las diferencias de horario sea otro factor importante. Esto puede ser una ventaja, pues presenta la oportunidad de programar actividades de trabajo las 24 horas del día. Pero también puede ser una desventaja cuando las actividades necesitan una colaboración intensiva entre personas ubicadas en localidades distantes y, por consiguiente, la comunicación síncrona es difícil o imposible de establecer dentro de la jornada de trabajo normal [14], [23], [28].
- **Administración del Conocimiento.** Las personas que trabajan en proyectos GSD necesitan compartir mucha información acerca de los requisitos, los cuales provienen de diferentes fuentes en sitios distantes. Sin un efectivo mecanismo para compartir información y conocimiento no es posible explotar los beneficios del GSD [14], [18], [19].
- **Diferencias de lenguaje.** Las diferencias del lenguaje puede ser una fuente de malas interpretaciones especialmente cuando la lengua usado en la comunicación no es la lengua materna [19], [23], [28].
- **Confianza.** Mantener relaciones de confianza es especialmente difícil en ambientes GSD debido a la falta de una comunicación informal y espontánea [5], [7], [22].

2.3. Conocimientos, habilidades y competencias

En la literatura los conceptos de conocimientos, habilidades y competencias son usados para describir los atributos que debe tener una persona para desarrollar una profesión. En [27] estos conceptos son definidos así:

- **Conocimiento** es la información almacenada e interpretada en la mente humana.
- **Habilidades (skills)** son obtenidas a través de experiencias y están basadas en conocimiento.
- **Competencias** se entiende como el conocimiento y la habilidad desarrollada en la práctica.

Como mencionamos en la introducción, nuestro trabajo tiene como fin la definición de las competencias necesarias para la captura de requisitos. Decidimos usar el concepto de competencia porque este concepto es más general y en su definición incluye a los conceptos de conocimiento y habilidades. Además, porque es el usado en el proyecto Tuning [30]

3. Proceso de Investigación

Con el fin de obtener la lista de competencias que son necesarias para acometer el proceso de captura de requisitos hemos desarrollado el siguiente proceso que es descrito en la Fig. 1.



Figura 1. Proceso usado para la obtención de las competencias en este estudio.

En primer lugar, hemos llevado a cabo una revisión de la literatura en búsqueda de artículos relacionados con las competencias, conocimientos

y habilidades reportadas para ingenieros del software en general y aquellos artículos que hablan de las competencias necesarias para entornos GSD.

Basado en estos estudios, hemos desarrollado una lista inicial de 189 competencias para un ingeniero de requisitos, juntando todas las competencias encontradas no importando si estaban repetidas o contenidas unas en otras.

A partir de esta lista hemos seleccionado aquellas competencias que están relacionadas con el proceso de captura de requisitos reduciendo el conjunto de 189 a 90 competencias. A continuación, hemos eliminado las competencias repetidas y hemos agrupado aquellas que se refieren a lo mismo pero en otros términos o las que estaban contenidas en otras obteniendo un total final de 70 competencias.

Finalmente, hemos analizado cada una de estas competencias para estudiar el impacto de los factores críticos para GSD presentados en la sección 2.2. Como resultado de este análisis, hemos obtenido cuatro conjuntos de competencias dependiendo del grado de importancia de la competencia en un entorno GSD. Estos conjuntos corresponde a:

1. Las competencias que no son importantes en GSD.
2. Las competencias igualmente importantes en entornos locales y entornos GSD.
3. Las competencias que aumentan de importancia en entornos GSD.
4. Las competencias que solo son importantes en entornos GSD.

Buscando una clara presentación de los resultados hemos clasificado las competencias dentro de cada grupo en dos niveles. En un primer nivel las clasificamos como genéricas y específicas [30]. Son genéricas si estas pueden ser desarrolladas en otra titulación y son específicas aquellas competencias que están directamente relacionadas con la captura de requisitos. Para el segundo nivel de división hemos clasificado las competencias genéricas del siguiente modo [30]:

- *Instrumentales*, aquellas que tienen una función instrumental, como las habilidades cognitivas y capacidades metodológicas.
- *Interpersonales*, las que facilitan la integración social y la cooperación, como la

capacidad de crítica y autocrítica y de trabajo en grupo.

- *Sistémicas*, son aquellas que conciernen a los sistemas como totalidad. Estas competencias requieren la adquisición previa de competencias instrumentales e interpersonales.

Las competencias específicas las hemos clasificado de la siguiente manera:

- *Clasificación de requisitos*. El conjunto de competencias que permiten comprender como los requisitos son clasificados.
- *Captura de requisitos*. Las competencias que permiten acometer un proceso de captura de requisitos.
- *Fundamentos de Requisitos*. Los conocimientos fundamentales referentes a la captura de requisitos.
- *Proceso de requisitos*. Los conocimientos fundamentales relacionados con el proceso de ingeniería de requisitos y su administración que debe ser conocido por un experto de la captura de requisitos.

4. Resultados

Antes de detallar la lista de competencias que hemos identificado, vamos a comentar los principales trabajos que nos han permitido definir dicha lista.

4.1. Principales fuentes del estudio

Las principales fuentes de este estudio son las siguientes:

SWEBOK [29], tiene como filosofía definir el cuerpo de conocimientos generalmente aceptados para los ingenieros del software y por ende no se toman en cuenta aspectos de GSD en la definición de los conocimientos.

SE2004 [20], presenta el cuerpo de conocimiento para la ingeniería del software que se requiere enseñar en un programa de grado. SE2004 entrega información relevante sobre como enseñar estos conocimientos, indicando cuales son fundamentales y cuales son deseables, así como, la manera sugerida de estructurarlo dentro del curriculum académico. Al igual que en el SWEBOK cuenta con un amplio consenso pero tampoco hace referencia a los aspectos del GSD. Si bien ambos estudios son desarrollados por

ACM e IEEE, el cuerpo de conocimientos presentado en SWEBOK difiere en parte al del SE2004. Por ejemplo: “Capas/niveles de requisitos” aparece en SE2004 pero no en SWEBOK y “Propiedades emergentes” aparecen en SWEBOK pero no en SE2004.

Ambos estudios son los que aportaron una mayor cantidad de competencias específicas para la captura de requisitos de las listas 2 y 3.

El proyecto Tuning [30], presenta una visión clara de la importancia de la formación por competencias y el aporte que hace a la educación. Además presentan un estudio de las competencias genéricas e indica cuales son las más importantes y cuales son las menos importantes tanto para el mundo académico, profesores y estudiantes, como para la industria. Lamentablemente, este estudio no contempla la carrera de ingeniería informática. Esto hace que la jerarquía de competencias que han podido concluir no sea válida en nuestro campo. Por ejemplo las competencias de: entender las culturas y costumbres de otros países, apreciar la diversidad y la multiculturalidad, la habilidad de trabajar en un contexto internacional se encuentran entre las menos valoradas por los graduados y empleadores encuestados en el proyecto Tuning [30], esto se debe a que los grupos encuestados corresponden a campos del saber que no han sido afectados tan fuertemente por la globalización como lo es la informática. Por esta razón sería útil replicar el estudio en nuestro campo con el fin de jerarquizar adecuadamente las competencias genéricas según nuestras necesidades.

El estudio empírico de Damian [13], presenta las competencias que deben tener las personas que trabajen en ingeniería de requisitos en GSD. Sin embargo, este estudio no hace un análisis de las competencias que son necesarias en entornos tradicionales, simplemente, asume que todas esas características son necesarias en entornos globales. Como mostraremos a continuación esto no es del todo cierto pues existen competencias tradicionales que deben ser estudiadas más a fondo pues estas son afectadas por los factores críticos del GSD.

A continuación detallamos las competencias encontradas para cada grupo.

4.2. Grupo 1

Este grupo corresponde a las competencias que son importantes únicamente en entornos locales de captura y que no son necesarias para la captura en entornos GSD. No encontramos competencias para este grupo.

4.3. Grupo 2

Este grupo corresponde a las competencias con igual importancia tanto para entornos GSD como para entornos locales. Estas competencias no son afectadas por las características propias de entornos GSD y, por tanto, pueden ser enseñadas en cursos donde se trate el tema de la captura co-localizada. En este grupo hemos identificado un total de 49 competencias de las cuales 21 son específicas y 28 son genéricas. Estas competencias las mostramos en las Tablas 1 y 2.

| Competencias Genéricas | |
|---|--|
| Instrumentales | |
| Capacidad de análisis y síntesis [30], [21]. | |
| Desarrollo de contenido multimedia [16]. | |
| Habilidades básicas de manejo del ordenador [30]. | |
| Resolución de problemas (generales y de negocio) [3], [21], [30]. | |
| Habilidades de gestión de la información (habilidad para buscar y analizar información proveniente de fuentes diversas) [30], [21]. | |
| Comunicación oral y escrita en la propia lengua [30], [3], [20], [16]. | |
| Habilidades para presentaciones (dar presentaciones efectivas) [20], [16]. | |
| Identificación de problemas/oportunidades [21], [2]. | |
| Lectura, comprensión y resumen de lecturas (ejemplo: Código fuente, documentación) [20]. | |
| Interpersonales | |
| Capacidad para comunicarse con expertos de otras áreas [30]. | |
| Capacidad para trabajar en un equipo interdisciplinar [30], [31]. | |
| Atención a detalles [3]. | |
| Códigos de ética y conducta profesional [20], [3]. | |
| Iniciativa/motivación de trabajar [3], [16]. | |
| Habilidades interpersonales [30]. | |
| Habilidad para trabajar de forma autónoma (Autodirección, responsabilidad individual) [30], [21], [3], [2]. | |
| Reaccionar rápidamente a los cambios del proyecto [24]. | |
| Cuestiones y preocupaciones sociales, legales, históricas y profesionales [20]. | |
| Trabajar bajo presión [3]. | |
| Sistémicas | |
| Capacidad para adaptarse a nuevas situaciones [30]. | |
| Capacidad de aprender (habilidades para aprender, aprender rápidamente acerca del dominio o tecnología al comienzo de la planificación del proyecto) [30], [3], [25]. | |
| Preocupación por la calidad [30]. | |
| Liderazgo [3], [30]. | |
| Habilidades de investigación [30], [3]. | |
| Gestión del tiempo [3]. | |
| Motivación al logro [30]. | |

Tabla 1. Competencias genéricas para el grupo 2

| Competencias Específicas |
|---|
| Clasificación de requisitos |
| Propiedades emergentes [29]. |
| Requisitos funcionales y no funcionales [29]. |
| Capas/niveles de requisitos (Ej.: necesidades, metas, requisitos de usuario, etc.) [20]. |
| Requisitos del producto y del proceso [29]. |
| Capacidad para clasificación de requisitos [29], [32]. |
| Requisitos del sistema y del software [29]. |
| captura de requisitos |
| Fuentes de la captura (Ej.: <i>Clientes, usuarios, expertos del dominio, etc.</i>) [29], [20]. |
| Fundamentos de requisitos |
| Criterios de un buen requisito [32]. |
| Definición de requisitos (Ej.: producto, proyecto, restricciones, etc.) [29], [20]. |
| Requisitos cuantificables [29]. |
| Atributos de requisitos [32], [29]. |
| Característica de los requisitos (Ej.: verificable, no ambiguo, consistente, etc.) [20]. |
| Fundamentos de documentación de requisitos (Ej.: Tipo de audiencia, atributos, etc.) [20]. |
| Errores en requisitos (omitido, incorrecto, factible, fuera del ámbito) [32]. |
| Proceso de requisitos |
| Cliente/usuario involucrado con los requisitos (unido al equipo) [32]. |
| Naturaleza iterativa del proceso de requisitos [29]. |
| Actores del proceso [29]. |
| Modelos del proceso [29]. |
| Proceso de requisitos [20]. |

Tabla 2. Competencias específicas para el grupo 2

4.4. Grupo 3

Este grupo corresponde a las competencias importantes en entornos locales que aumenta su importancia en entornos GSD pues son afectadas en alguna medida por los desafíos del GSD presentados en la sección 2.2. Por lo anterior, estas competencias deben ser estudiadas con mayor profundidad para encontrar una pedagogía adecuada en entornos GSD.

En este grupo hemos identificado un total de 17 competencias de las cuales 7 son específicas y 10 son genéricas.

A continuación presentamos las competencias de este grupo. En la tabla 3 presentamos las competencias genéricas y en la tabla 4 las competencias específicas.

4.5. Grupo 4

Este grupo corresponde a las competencias que son importantes únicamente en GSD. En este grupo hemos identificado un total de 6 competencias, todas son genéricas. Las competencias para este grupo las presentamos en la tabla 5.

| Competencias Genéricas |
|--|
| Instrumentales |
| Comunicación mediante ordenador [26], [13]. La distancia geográfica entre los participantes del proceso hace que esta competencia sea más importante en GSD además por las dificultades de husos horarios. |
| Interpersonales |
| Protocolos de comunicación [26], [19]. Los protocolos de comunicación cambian entre culturas, así por ejemplo la manera de saludar a otro puede ser motivo de conflicto (por ejemplo con un beso). |
| Habilidades para comunicación, respuestas oportunas, velocidad, reconocer la brecha semántica [1], [2], [26], [21], [9]. En un entorno GSD es necesario un esfuerzo mayor para lograr una comunicación efectiva pues afectan los problemas culturales, la distancia, el lenguaje y las diferencias de horario. |
| Resolución de conflictos [3]. Esta competencia se ve afectada por las diferencias culturales y los problemas de comunicación en GSD. |
| Capacidad crítica y autocrítica [30], [3], [9]. Criticar el trabajo de otras personas cuando son de una cultura diferente es mucho más complicado que cuando la cultura es la misma. |
| Tratando con incertidumbre y ambigüedad (en equipos locales y remotos) [20], [13]. En equipos remotos la ambigüedad es mucho mayor por los problemas de gestión del conocimiento presentes en GSD. |
| Interacción con los participantes del proceso de desarrollo (a menudo de diferente cultura; relaciones interpersonales) [20], [15], [3]. La diferencia cultural, la comunicación inadecuada, la diferencia de lenguaje y una pobre confianza afectan la interacción entre usuarios y desarrolladores. |
| Comunicación de equipo y de grupo (oral y escrita, email, etc.) [20]. El desarrollo de esta competencia se ve particularmente afectada por todos los problemas en GSD descritos en la sección 2.2. |
| Trabajo en equipo (Dinámica del trabajo en equipo/grupo, habilidades de equipo) [30], [20], [21], [2], [3]. Al igual que la competencia anterior, esta competencia se ve afectada por todos los desafíos in GSD. |

Tabla 3. Competencias Genéricas para el grupo 3

| Competencias Específicas |
|---|
| captura de requisitos |
| Técnicas de captura (Ej.: entrevistas, cuestionarios, encuestas, prototipos, etc.) [20], [29]. Las técnicas para captura fueron desarrolladas para entornos locales, y por tanto es necesario revisar la manera de utilizarla en entornos globales para que sean efectivas. |
| Técnicas avanzadas de captura (Ej.: estudios etnográficos, captura de conocimiento, etc.) [20]. |
| Identificación de requisitos reales (desde los requisitos indicados) [32]. Los problemas de comunicación y las diferencias culturales afectan esta competencia. |
| Habilidad para captura de requisitos [21], [32], [9]. Esta competencia es afectada por todos los problemas del GSD. |
| Proceso de requisitos |
| Control y notificación de cambios en requisitos [32]. Esta competencia se ve particularmente afectada por los problemas de administración del conocimiento. |
| Trazabilidad de Requisitos [29]. Al igual que la anterior, esta competencia también se ve afectada por los problemas de administración del conocimiento in GSD. |
| Factores críticos [21]. El conocimiento de los factores críticos para proyectos de desarrollo debe incorporar los factores particulares del GSD. |

Tabla 4. Competencias específicas para el grupo 3

| |
|--|
| Competencias Genéricas |
| Competencias Instrumentales |
| Conocimiento de una segunda lengua (principalmente inglés) [30], [19], [26]. La necesidad de comunicarse con personas que hablan otro idioma hace que sea imprescindible el conocimiento de un segundo lenguaje, en especial el idioma inglés. |
| Competencias Interpersonales |
| Habilidad para trabajar en un contexto internacional (operar en un ambiente de equipo virtual, habilidades de equipo virtual) [30], [16], [31], [13]. |
| Apreciación de la diversidad y la multiculturalidad. [30]. |
| Competencias sistémicas |
| Tratar con ambientes multiculturales [20], [21], [26], [16]. |
| Conocimiento de culturas y costumbres de otros países [30]. |
| <i>Conciencia (Awareness)</i> global [1]. Lograr una conciencia del trabajo de los otros miembros del equipo virtual (<i>conciencia</i>) es particularmente difícil por los problemas en la gestión del conocimiento. |

Tabla 5. Competencias para el grupo 4

5. Conclusión y trabajo futuro

En este artículo presentamos el resultado de un estudio y análisis bibliográfico el cual nos permite proponer una lista de competencias, genéricas y específicas, que los profesionales necesitan para trabajar en la captura de requisitos, tanto en un entorno de desarrollo de software local, como en un entorno global. Hemos agrupado las competencias de manera que se pueda identificar aquellas que son más importantes o que no están presentes en una captura co-localizada y, por tanto, requieren de un modo de enseñanza orientado a GSD. Además, presentamos en otro grupo las competencias que son igualmente importantes en entornos locales como globales.

El objetivo de nuestro trabajo es el de aproximarnos a una lista de competencias, que tenga el consenso de la industria y el mundo académico para la captura de requisitos.

Esta especificación permitirá un diseño curricular que se acerca más a los desafíos actuales de la industria. Además ayudará a la industria del software a desarrollar procesos de reclutamiento de personal para decidir qué persona es la más adecuada para llevar a cabo un proceso de captura de requisitos en GSD.

Nuestro trabajo actual y futuro esta centrado en apoyar la docencia relacionada con estas competencias mediante el desarrollo de un entorno de simulación de manera que los estudiantes se puedan entrenar en el proceso de captura de requisitos en un entorno global.

Agradecimientos

Este trabajo es parcialmente financiado por los proyectos: MELISA (PAC08-0142-3315), Junta de Comunidades de Castilla-La Mancha, Consejería de Educación y Ciencia, en España; ESFINGE (TIN2006-15175-C05-05) Ministerio de Educación y Ciencia (Dirección General de Investigación)/Fondos Europeos de Desarrollo Regional (FEDER) en España; CompetiSoft (506AC0287, programa CYTED).

Referencias

- [1] Adya, M.P., Imparting Global Software Development Experience via an IT Project Management Course: Critical Success Factors. In 30th COMPSAC, (2006), 51-52.
- [2] Ahamed, S.I., Model for Global Software Engineering Project Life Cycle and How to Use it in Classroom for Preparing Our Students for the Globalization. in 30th COMPSAC, (2006), 59.
- [3] Aken, A. and Michalisin, M.D., The impact of the Skills Gap on the Recruitment of MIS Graduates. in SIGMIS-CPR, (2007), ACM.
- [4] Aspray, W., Mayadas, F. and Vardi, M. Globalization and Offshoring of Software, Association for Computing Machinery, Job Migration Task Force (ACM). 2006.
- [5] Babar, M.A., Verner, J.M. and Nguyen, P.T. Establishing and maintaining trust in software outsourcing relationships: An empirical investigation. The Journal of Systems and Software, 80 (9). 1438-1449.
- [6] Bellur, U., An Academic Perspective on Globalization in the Software Industry. in 30th COMPSAC, (2006), 53.
- [7] Bhat, J.M., Gupta, M. and Murthy, S.N. Overcoming Requirements Engineering Challenges: Lessons from Offshore Outsourcing. IEEE Software, 23 (5). 38-44.
- [8] Boehm, B. The Future of Software and Systems Engineering Processes, University of Southern California, Los Angeles, CA 90089-0781, 2005.
- [9] Callele, d. and Makaroff, D., Teaching Requirements Engineering to an Unsuspecting Audience. in 37th SIGCSE, (2006), ACM, 433 - 437.
- [10] Carmel, E. and Abbott, P., Configurations of global software development: offshore versus

- nearshore. in 2006 international workshop on Global software development for the practitioner, during ICSE'06, (Shanghai, China, 2006), ACM Press, 3-7.
- [11] Cheng, B.H.C. and Atlee, J.M., Research Directions in Requirements Engineering. in FOSE '07, (2007), IEEE Computer Society Washington, DC, USA, 285-303.
- [12] Damian, D. Stakeholders in Global Requirements Engineering: Lessons Learned from Practice. *IEEE Software*, 24 (2). 21–27.
- [13] Damian, D., Hadwin, A. and Al-Ani, B., Instructional design and assessment strategies for teaching global software development: a framework. in ICSE, (2006), ACM, 685-690.
- [14] Damian, D.E. and Zowghi, D., The Impact of Stakeholders' Geographical Distribution on Managing Requirements in a Multi-Site Organization. in RE'02, (2002), IEEE Computer Society, 319-328.
- [15] Ghezzi, C. and Mandrioli, D., The challenges of software engineering education. in 27th ICSE, (2005), 637-638.
- [16] Gorgone, J.T., Davis, G.B., Valacich, J.S., Topi, H., Feinstein, D.L. and Herbert E. Longenecker, J. IS 2002. Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems, Association for Computing Machinery (ACM), Association for Information Systems (AIS), Association of Information Technology Professionals (AITP), 2002.
- [17] Herbsleb, J.D., Global Software Engineering: The Future of Socio-technical Coordination. in FOSE'07 at ICSE'07, (2007), IEEE Computer Society, 188-198.
- [18] Herbsleb, J.D. and Moitra, D. Guest Editors' Introduction: Global Software Development. *IEEE Software*, 18 (2). 16-20.
- [19] Huang, H. and Trauth, E., Cultural influences and globally distributed information systems development: experiences from Chinese IT professionals. in SIGMIS-CPR '07, (2007), ACM Press New York, NY, USA, 36-45.
- [20] IEEE and ACM. Software Engineering 2004, Curriculum Guidelines for Undergraduate Degree Programs In Software engineering, IEEE computer Society Press and ACM Press, 2004.
- [21] Minor, O. and Armarego, J. Requirements Engineering: A Close Look At Industry Needs And Model Curricula. *Australian Journal of Information Systems (AJIS)*, 13 (1).
- [22] Nguyen, P., Babar, M. and Verner, J., Critical factors in establishing and maintaining trust in software outsourcing relationships. in ICSE'06, (Shanghai, China, 2006), ACM Press New York, NY, USA, 624-627.
- [23] Raffo, D. and Setamanit, S., A Simulation Model for Global Software Development Project. in The International Workshop on Software Process Simulation and Modeling, (St. Louis, MO, 2005).
- [24] Richardson, I., Casey, V., Zage, D. and Zage, W. Global Software Development – the Challenges, University of Limerick, Ball State University, SERC Technical Report 278, 2005, 10.
- [25] Richardson, I., Milewski, A., Mullick, N. and Keil, P., Distributed development: an education perspective on the global studio project. in ICSE'06, (2006), ACM Press New York, NY, USA, 679-684.
- [26] Richardson, I., Moore, S., Paulish, D., Casey, V. and Zage, D., Globalizing Software Development in the Local Classroom. in 20th CSEET, (2007), 64.
- [27] Rose, J., Pedersen, K., Hosbond, J.H. and Kraemmergaard, P. Management competences, not tools and techniques: Agrounded examination of software project management at WM-data. *Information and Software Technology*, 49 (6). 605-624.
- [28] Setamanit, S., Wakeland, W. and Raffo, D., Planning and improving global software development process using simulation. in International Workshop on Global Software Development for the Practitioner (GSD'06), (Shanghai, China, 2006), 8-14.
- [29] SWEBOK Guide to the Software Engineering Body of Knowledge, 2004.
- [30] Tuning. Tuning General Brochure. English version, 2007.
- [31] Vasudevan, V., Global Software Entrepreneurship. in 30th COMPSAC'06, (2006), 55.
- [32] Young, R. Twelve Requirements Basics for Project Success. *CROSSTALK The Journal of Defense Software Engineering*, 2006 (December).