



## **Gestión de Recursos Humanos por Competencias en los Proyectos de Software**

### *Human Resources Management based on Competences in Software Projects*

Margarita André Ampuero<sup>1\*</sup>, María G. Baldoquín de la Peña<sup>2</sup> y Jorge M. Soler McCook<sup>3</sup>

<sup>1</sup>Departamento de Ingeniería de Software, Instituto Superior Politécnico "José Antonio Echeverría", Marianao, CP. 19390, La Habana, Cuba

<sup>2</sup> Departamento de Matemática, Instituto Superior Politécnico "José Antonio Echeverría", La Habana, Cuba

<sup>3</sup> Empresa DESOFT, La habana, Cuba

\*Autor para la correspondencia: [mayi@ceis.cujae.edu.cu](mailto:mayi@ceis.cujae.edu.cu)

---

## Resumen

Este trabajo tiene como objetivo contribuir a la implementación, en la industria de software, de la gestión de recursos humanos basado en el enfoque por competencias. Para ello, se realiza y fundamenta una propuesta de roles invariantes que participan tanto en proyectos de desarrollo como de implantación de software, y de competencias tanto técnicas como genéricas requeridas para desempeñar los roles propuestos. En el trabajo se fundamenta el uso del método Delphi y las variaciones requeridas, como vía para validar la propuesta de roles y competencias, y para identificar la importancia de cada competencia en el desempeño de cada rol. Además, se propone un procedimiento para determinar el índice de competencia de una persona para desempeñar un rol asignado en el proyecto.

**Palabras clave:** Competencias, gestión de recursos humanos, método Delphi, proyectos de software, roles.

## Abstract

*The main objective of the paper is to contribute to the implementation, in the software industry, of human resources management based on competences. To do this, the use of the Delphi method and the required changes are discussed, as a way to validate the roles' and competences' proposal, and to identify the importance of each competence to carry out each role. Besides, it presents the procedure to calculate the competence's index of each person to perform the assigned role in the project.*

**Key words:** *Competences, Delphi method, human resources management, roles, software's projects.*

## Introducción

A pesar del vertiginoso desarrollo, los reconocidos resultados y el gran impacto que ha tenido la industria de software en prácticamente todas las ramas del desarrollo de la sociedad a nivel mundial, aún resulta significativo el número de proyectos de software que no culminan con éxito (Pressman, 2002). Entre las causas que provocan estos fracasos se encuentran las asociadas a factores humanos. Son varias las investigaciones que reconocen que los recursos humanos juegan un papel crítico en el éxito o fracaso de un proyecto de software (Acuña, 2002; DeMarco, 1999; Gorla, 2004; Pressman, 2004). Sin embargo, el personal continúa siendo el factor menos formalizado en los modelos de procesos de software actuales, en las metodologías de desarrollo y gestión de proyectos y en los estándares aplicables a esta industria; los cuales se centran más en aspectos técnicos que en los aspectos humanos (Acuña, 2002; André, 2008).

La industria de software ha experimentado un aumento significativo en el volumen y complejidad de los productos que desarrolla. En consecuencia, el desarrollo de software pasó de ser un trabajo esencialmente artesanal a un trabajo en equipo donde cada persona no solo debe desempeñar correctamente su rol sino que debe constituir un miembro efectivo del equipo mostrando disciplina personal y en equipo. La necesidad

que experimenta la industria del software se corresponde con la tendencia cada vez más fuerte, que existe en la actualidad, hacia la gestión de los recursos humanos, potenciando las características del personal que integra las organizaciones. La Gestión de Recursos Humanos (GRH) es mucho más eficiente si se basa en las competencias individuales de los trabajadores, teniendo en cuenta que para las organizaciones la competencia laboral apoya los procesos de selección, contratación, capacitación, evaluación y compensación del personal (Cuesta, 2005). Sin embargo, las investigaciones indican que esta disciplina está muy poco desarrollada aún (Urquiza, 2007). La industria de software no está exenta de esta realidad a pesar de que para ella, esta disciplina resulta muy importante. Ni los modelos de procesos, ni las metodologías de desarrollo ofrecen un marco de referencia bien formalizado, que contenga claramente definidas las competencias genéricas y técnicas requeridas para desempeñar cada rol establecido acorde a las características de los proyectos de software.

### **Gestión de Recursos Humanos por competencias en proyectos de software**

Los primeros aportes prácticos sobre el tema de competencias se deben a David McClelland, quien demostró que los resultados académicos y los tradicionales tests de inteligencia no predicen de forma fiable el éxito profesional (McClelland, 1973). Vargas (Vargas, 2001), a partir del análisis de varias definiciones de competencias y de los trabajos de un grupo de estudiosos del tema concluyó que las competencias: son características permanentes de la persona, se ponen de manifiesto cuando se ejecuta una tarea o se realiza un trabajo, están relacionadas con la ejecución exitosa en una actividad (sea laboral o de otra índole), tienen una relación causal con el rendimiento laboral y pueden ser generalizables a más de una actividad.

En este trabajo se utiliza el concepto de competencia laboral expuesto por Cuesta en (Cuesta, 2005) que plantea: "las competencias laborales son características subyacentes en las personas, asociadas a la experiencia, que como tendencia están causalmente relacionadas con actuaciones exitosas en un puesto de trabajo contextualizado en una determinada cultura organizacional". Además, se distinguen dos categorías de competencias: las genéricas (que definen características referidas al comportamiento general del empleado, independientes de los conocimientos técnicos concretos. Ejemplos: negociación, planificación y organización, liderazgo, capacidad de análisis, etc.) y las técnicas o específicas (asociadas a conocimientos y habilidades técnicas específicas de cada puesto de trabajo).

Para las organizaciones de software la GRH por competencias constituye una práctica muy necesaria ya que debido a las peculiaridades de esta industria resulta clave proteger y desarrollar a su personal. En el profesional del software confluyen condiciones ideales para que experimente gran movilidad: es especializado (por sus conocimientos y habilidades), pero a su vez generalista (por la gran cantidad y variedad de dominios de aplicación de su trabajo) y muy demandado, debido al impacto de la informatización en prácticamente todas las ramas de desarrollo de un país. Por otra parte, debido al desarrollo vertiginoso de las Tecnologías de la Información y las Comunicaciones, lo cual provoca cambios continuos en el estado del arte, las organizaciones de software necesitan contar con un personal técnicamente competente y en continua superación. Además, por ser la ingeniería de software un proceso social complejo en el cual la comunicación y la interacción cooperativa entre los miembros del equipo tiene su impacto en la calidad del producto desarrollado de forma colaborativa, es preciso que el personal también posea competencias de comportamiento como: la comunicación, la negociación y la capacidad de trabajar en equipo, entre otras.

Cabe señalar que, a pesar del avance en la comprensión y aplicación de esta disciplina en algunas de las organizaciones de software cubanas, no constituye una práctica generalizada. Prueba de ello es que en la actualidad, la asignación de personal a los roles de un proyecto de software y la formación del propio equipo se realiza basado, fundamentalmente, en criterios empíricos de los jefes de proyecto. En muchos casos no se contemplan factores tales como: competencias técnicas y genéricas, motivaciones, intereses, carga del personal, tamaño del equipo e incompatibilidades o conflictos entre miembros (Rodríguez, 2008).

Los autores de este trabajo participan en una investigación que tiene como objetivo principal proponer un modelo formal para asignar personal a proyectos de software. Como parte de la investigación se utilizó el método Delphi para identificar cuáles deben ser los factores fundamentales a tomar en cuenta en el proceso de asignación. Al culminar la segunda ronda en la aplicación del método, se pudo concluir que las competencias (tanto técnicas como genéricas), para desempeñar los distintos roles del proyecto, constituyen uno de los principales factores que los expertos consideran que se debe tomar en cuenta en la asignación (André, 2008; Rodríguez, 2008).

Sin embargo, para aplicar este modelo es preciso que las organizaciones tengan formalizado cuáles son los roles que intervienen en los proyectos y cuáles las competencias requeridas para desempeñar correctamente los roles. En este sentido, existen grandes dificultades ya que ni los modelos de procesos, ni las metodologías de desarrollo ofrecen un marco de referencia bien formalizado, que contenga claramente definidas las competencias genéricas y técnicas requeridas para desempeñar cada rol establecido acorde a las características del proyecto. Por otra parte, de manera general, las organizaciones de software tampoco cuentan con un marco de referencia bien formalizado que contenga los roles requeridos en función de los tipos de proyectos.

### **Propuestas de roles invariantes para proyectos de desarrollo e implantación de software**

Resulta difícil elaborar una propuesta que contenga todos los posibles roles ya que ello depende en gran medida del tipo y de la complejidad del proyecto a desarrollar. A continuación se realiza una propuesta de roles invariantes para proyectos de desarrollo e implantación de software y de las competencias requeridas para su adecuado desempeño. Esta propuesta puede servir de punto de partida y soporte en la implementación de la gestión de recursos humanos por competencias en las organizaciones de software cubanas. Para elaborar la propuesta se analizaron los roles definidos en reconocidas metodologías y modelos de proceso de desarrollo de software.

El análisis de los roles establecidos en el Proceso Unificado de Desarrollo de Rational (RUP) representa un buen punto de partida por su amplia aplicación a nivel internacional y por constituir la propuesta metodológica aceptada por *Object Management Group* (OMG). Según RUP, un rol es un puesto que puede ser asignado a una persona o conjunto de personas que trabajan juntas en un equipo, y que requiere habilidades y responsabilidades como: realizar determinadas actividades y desarrollar determinados artefactos. Los miembros de un equipo de proyecto generalmente cubren varios roles (Jacobson, 2002). Los roles de RUP se clasifican en cinco grandes grupos (André, 2005):

- Analistas: Analista del Proceso de Negocios, Diseñador de Negocios, Revisor del Modelo de Negocios, Analista de Sistema, Especificador de Requisitos, Revisor de Requisitos y Diseñador de la Interfaz Usuario.

- Desarrolladores: Arquitecto de Software, Revisor de la Arquitectura, Diseñador, Diseñador de Cápsula, Diseñador de Base de Datos, Revisor del Diseño, Programador, Revisor del Código, Integrador.
- Probadores: Diseñador de Prueba, Probador.
- Directivos: Director de Control de Cambio, Director de Configuración, Director de Implantación, Ingeniero de Proceso, Director del Proyecto, Revisor del Proyecto.
- Otros: Stakeholder, Desarrollador de Cursos, Artista Gráfico, Administrador de Sistema, Documentador Técnico, Especialista en Herramientas.

*Extreme Programming* (XP) es definida por su autor, Kent Beck, como una metodología ágil para el desarrollo de software destinada a ser utilizada por equipos de desarrollo pequeños y medianos (de 2 a 10 miembros) que enfrenten proyectos con requerimientos imprecisos o cambiantes, donde las relaciones desarrollador-desarrollador y desarrolladores-cliente, constituyen los elementos claves. XP propone siete roles: Programador, Cliente, Encargado de pruebas, Encargado de seguimiento, Entrenador, Consultor y Gestor (Beck, 1999). Crystal (familia de metodologías ágiles caracterizadas por estar centradas en las personas y en la reducción del número de artefactos producidos, propone ocho roles: Patrocinador Ejecutivo, Jefe de Proyecto, Experto en el Dominio, Experto de uso, Programador-Diseñador, Diseñador de Interfaz de Usuario, Probador, Programador Técnico (Cockburn, 2005).

El Proceso de Software en Equipo (TSP), modelo prescriptivo para equipos de desarrollo de software, propone que las responsabilidades deben distribuirse entre los miembros del equipo a través de ocho roles: Gerente de Interfaz-Usuario, Gerente de Diseño, Gerente de Implementación, Gerente de Planeación, Gerente de Procesos, Gerente de Calidad, Gerente de Soporte Técnico y Gerente de Pruebas. Los especialistas de calidad, procesos, administración de configuración, herramientas y bibliotecarios apoyan en cada una de las áreas correspondientes (Serrano, 2005).

A partir del estudio de las propuestas mencionadas anteriormente se puede concluir que es posible identificar un conjunto de roles comunes o invariantes que participan en proyectos de desarrollo de software. Estos son:

- Roles técnicos, responsables del desarrollo del sistema: Analista, Diseñador de Interfaz-Usuario, Arquitecto (Diseñador de alto nivel), Diseñador (de bajo nivel), Diseñador de Base de Datos (puede estar incluido en el rol anterior), Programador y Probador.
- Roles de gestión, responsables de la planificación y la ejecución del proyecto: Jefe de Proyecto (incluye funciones asociadas a la planificación y el control del proceso), Especialista de Calidad, Gestor de Configuración, Gestor de Cambios, Documentador y Especialista en Seguridad (Este rol se incorpora por la importancia que tiene tomar en cuenta los requisitos no funcionales asociados a la protección y seguridad del sistema).

Sin embargo, resulta importante señalar que no necesariamente son éstos los roles a establecer en cada proyecto. Es posible desglosar o integrar estos roles como ocurre en algunas de las propuestas estudiadas e incluso se pueden definir nuevos roles acorde

al tipo de proyecto en cuestión. Por ejemplo, en (Viera, 2007) aparece una propuesta de roles para enfrentar proyectos Multimedia. Por otra parte, existe un conjunto de empresas cuya misión no solo consiste en realizar proyectos de desarrollo sino que ejecutan proyectos de implantación. Para el caso de estos proyectos también resulta posible identificar tres roles invariantes: Jefe de Proyecto de Implantación, Especialista Operacional (con experiencia en el dominio del sistema) y Especialista Técnico (con experiencia en la actividad informática).

### **Propuesta de competencias técnicas y genéricas requeridas para el desempeño de los roles**

Al evaluar la propuesta de roles de cada uno de los procesos y metodologías de desarrollo de software analizadas se constató que en ningún caso se formalizan las competencias requeridas para su adecuado desempeño. En este sentido, se destaca el modelo de proceso de software propuesto en (Acuña, 2002) que incluye como elemento original, las capacidades de comportamiento o características de la conducta profesional. Acuña propone asignar las personas a roles del proyectos en función de las competencias genéricas requeridas para cada rol. Para ello las clasifica en cuatro grupos: habilidades intrapersonales, habilidades organizativas, habilidades interpersonales y habilidades directivas.

Tomando como base lo planteado por Acuña, encuestas a profesionales de la industria cubana de software, la propuesta de competencias de la empresa DESOFT y el análisis de los diccionarios de competencias, se decidió realizar la propuesta, que contempla 26 competencias (19 genéricas y 7 técnicas), que se muestran en la primera fila de la tabla que aparece en la Figura 1.

Una vez elaborada la propuesta de roles invariantes para enfrentar proyectos de desarrollo e implantación de software, y de las competencias requeridas para el desempeño de dichos roles, se decidió someterla a la consideración de un comité de expertos. El objetivo fundamental era obtener criterio de los expertos sobre: la propuesta de roles y competencias así como la importancia de cada competencia en el desempeño de los diferentes roles. Como paso previo a la conformación de dicho comité, los autores del trabajo decidieron aplicar el método Delphi para obtener los criterios de selección de los expertos. Después de dos rondas, se obtuvo como resultado que los expertos debían ser profesionales con conocimientos de ingeniería de software, que tuvieran 10 o más años de experiencia en el desarrollo y la dirección de proyectos de software, y que hubiesen desarrollado 2 o más proyectos de mediana o gran complejidad donde hubiesen dirigido a 3 o más personas. Posteriormente, se circuló entre los candidatos una plantilla para obtener su síntesis curricular y aplicando los criterios de selección, se creó el comité el cual está formado por 35 expertos provenientes de la industria y de la academia. A cada experto se le suministró una matriz en cuyas filas aparecían los 16 roles propuestos y cuyas columnas contenían las 26 competencias. Así, los expertos debían completar 416 celdas ( $16 \times 26 = 416$ ) con un valor entre 1 y 5 (ver en la Figura 1 la escala utilizada). Para facilitar el trabajo de los expertos, se asociaron comentarios donde se describían las principales responsabilidades de los roles y una breve definición de cada competencia. Adicionalmente, los expertos tenían la posibilidad de adicionar y/o eliminar tanto roles como competencias.

Una vez recogidas las 35 matrices se obtuvieron 14560 datos ( $416 \times 35$  expertos = 14560) los cuales fueron procesados con el paquete estadístico MINITAB, utilizando gráficos de boxplots. Estos gráficos muestran la mediana y el rango del intercuartil ya que los datos

son ordinales. Por un problema de espacio no resulta posible reflejar el procesamiento realizado a los 16 roles. A continuación, se explican los pasos ejecutados utilizando como ejemplo el procesamiento del rol Jefe de Proyecto:

Paso 1: Procesar, para cada rol, los datos emitidos por los expertos en las 26 competencias. La Figura 2a muestra el procesamiento de los datos para el rol Jefe de Proyecto.

Paso 2: Analizar los valores outliers (representados por \*). Tomando en cuenta la naturaleza de los datos procesados, la escala utilizada y el concepto de outlier que ofrece el MINITAB, los autores del trabajo definieron como outlier aquellos valores de la variable que diferían en al menos dos unidades de los extremos de los boxplots. En el caso en que el rango del intercuartil tomaba valor 1 y existían bigotes (ya fuera superior, inferior o ambos) se consideraba outlier el valor que difiriera en una unidad del extremo del bigote.

Paso 3: Eliminar los outliers. Una vez que se eliminaron los 22 outliers identificados para el rol Jefe de Proyecto, se procesaron nuevamente los datos y se obtuvo el gráfico que se muestra en la Figura 2b.

	Competencias genéricas													Competencias Técnicas													
	Trabajo en equipo y cooperación	Capacidad de análisis	Capacidad de planificar y organizar	Capacidad de Controlar	Capacidad de Negociación	Capacidad de Adaptación (Flexibilidad) búsqueda, indagación)	Capacidad de Estrategia (Búsqueda, analiza y define estrategias)	Pensamiento conceptual	Manejo de riesgos	Toma de decisiones	Tenacidad (Perseverancia)	Independencia	Orientación al cliente	Compromiso con la organización	Proactivo (Iniciativa)	Comunicación oral	Comunicación escrita	Aprendizaje continuo	Conocimiento del idioma	Conocimiento del lenguaje de programación	Conocimiento de la herramienta de modelación	Dominio de gestor de base de datos	Dominio de la metodología de desarrollo	Habilidades de Diseño Gráfico	Dominio del Producto		
Jefe de proyecto																											
Diseñador Gráfico																											
Gestor de Cambios																											
Gestor de Configuración																											
Arquitecto																											
Especialista de Seguridad																											
Analista																											
Diseñador																											
Diseñador de Base de Datos																											
Programador																											
Documentador																											
Probador																											
Especialista de Calidad																											
Jefe Proyecto																											
Especialista Operacional																											
Especialista Técnico																											
Legenda	Roles que intervienen en proyectos de desarrollo de software													Roles que intervienen en proyectos de implantación o implementación de software													

Fig. 1. Matriz suministrada a los expertos en la primera ronda del método Delphi.

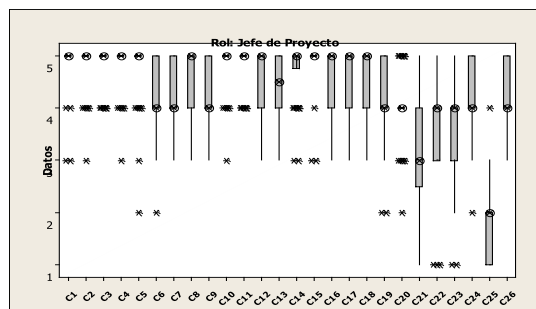


Fig. 2a. Análisis del rol Jefe de Proyecto.

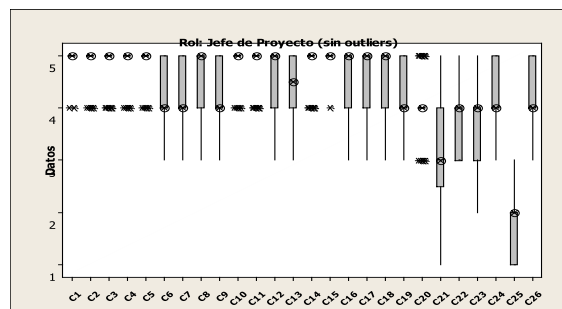


Fig. 2b. Análisis del rol Jefe de Proyecto (sin outliers).

Paso 4: Identificar las competencias requeridas para desempeñar cada rol. A partir del análisis del gráfico de la Figura 2b se pudo concluir que las competencias de la 1 a la 20, la 22, la 24 y la 26 eran necesarias para desempeñar el rol de Jefe de Proyecto ya que como se observa, todos los expertos concuerdan que cada una de estas competencias al menos resulta útil (valor  $\geq 3$ ) para el rol. Para llegar a conclusiones acerca de las competencias 21, 23 y 25, fue preciso analizar los valores de la mediana y la moda. Esto permitió concluir que tanto la competencia 21 como la 23 resultaban necesarias para el desempeño del rol y que en el caso de la competencia 25, debido a la gran dispersión en el criterio emitido por los expertos, se sugiere someterla a una segunda ronda de evaluación.

Paso5: Identificar el nivel de importancia de cada competencia para el desempeño de cada rol. Al culminar el paso anterior, para todos los roles se identificaron: las competencias requeridas, las no requeridas y las que debían ser sometidas a una segunda ronda. Sin embargo, entre las competencias requeridas no resultaba muy claro precisar, en una parte de los casos, el grado de importancia de cada una por la dispersión de los datos. Este hecho puede tener su explicación en la poca formalización que existen, en la mayor parte de las empresas, de los roles y las competencias necesarias para enfrentar los proyectos de software. Los autores del trabajo decidieron transformar los datos obtenidos con la escala de valores de 1 a 5, tomando en consideración una escala más simple, capaz de reflejar la información esencial obtenida bajo la escala anterior y que a su vez resultara más fácil de utilizar por los expertos al enfrentarse a la segunda ronda (este cambio de escala constituye una modificación en la aplicación del método Delphi). La nueva escala contempla tres valores:

- 2- la competencia es requerida en un alto por ciento.
- 1- la competencia es necesaria en alguna medida.
- 0- la competencia no es necesaria.

Al procesar todos los roles aplicando la nueva escala se obtuvo la matriz que se muestra en la Figura 3. Las celdas en blanco indican que esa competencia para ese rol debe ser sometida a la segunda ronda. Así, para el caso del rol jefe de proyecto es posible decir que las competencias 22 y 25 deben someterse a la segunda ronda, la competencia 21 es necesaria en alguna medida y el resto de las competencias son muy importantes para el desempeño del rol.



### Procedimiento para determinar un índice de competencia

Para utilizar un modelo formal en la asignación de personal a proyectos de software como el propuesto en (André, 2008; Rodríguez, 2008), y teniendo en cuenta que se desea maximizar las competencias requeridas por las personas que asumirán los diferentes roles necesarios en un proyecto, se propone un procedimiento para determinar un índice de competencia de cada persona en el desempeño de cada rol asignado al proyecto. Para ello se tiene en cuenta, para cada rol, tanto el nivel que tiene cada persona relacionado con las diferentes competencias necesarias para el rol, como el grado de importancia de dichas competencias para dicho rol.



	Competencias genéricas																	Competencias Técnicas								
	Trabajo en equipo y cooperación	Capacidad de análisis	Capacidad de planificar y organizar	Capacidad de Controlar	Capacidad de Negociación	Capacidad de Adaptación (Flexibilidad)	Capacidad de Investigar (exploración, búsqueda, indagación)	Capacidad de Estrategia (Busca, analiza y define estrategias)	Pensamiento conceptual	Manejo de riesgos	Toma de decisiones	Tenacidad (Perseverancia)	Independencia	Orientación al cliente	Compromiso con la organización	Proactivo (Iniciativa)	Comunicación oral	Comunicación escrita	Aprendizaje continuo	Dominio del idioma	Dominio del lenguaje de programación	Dominio de la herramienta de modelación	Dominio de gestor de base de datos	Dominio de la metodología de desarrollo	Habilidades de Diseño Gráfico	Dominio del Producto
<b>Jefe de proyecto</b>	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	2	2	2	2	2
<b>Diseñador Gráfico</b>	2	2	0	2	2	2	2	2	2	2	2	1	2	2	2	2	1	1	2	2	0	0	2	2	2	2
<b>Gestor de Cambios</b>	2	2	2	2	2	2	1	2	1	2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	2	2
<b>Gestor de Configuración</b>	2	1	1	2	1	2	1	1	2	2	2	2	2	2	2	2	1	1	2	1	2	2	1	1	0	2
<b>Arquitecto</b>	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
<b>Especialista de Seguridad</b>	2	2	2	2	1	2	2	2	1	2	2	2	2	1	2	2	2	2	2	2	2	2	2	1	0	2
<b>Analista</b>	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
<b>Diseñador</b>	2	2	2	2	2	2	2	2	2	2	2	2	2	1	2	2	2	2	2	2	2	2	2	2	2	2
<b>Diseñador de Base de Datos</b>	2	2	1	1	2	2	1	2	2	2	2	2	1	2	2	2	2	2	2	2	2	2	2	2	0	2
<b>Programador</b>	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	2	2	2	1	2	2	2	2	2
<b>Documentador</b>	2	1	1	0	1	2	2	1	2	2	1	1	2	2	1	2	2	0	2	0	0	0	2	2	2	2
<b>Probador</b>	2	1	0	1	2	2	2	2	2	2	2	1	2	2	1	2	2	2	1	2	1	0	0	0	2	2
<b>Especialista de Calidad</b>	2	2	2	2	2	2	1	2	1	2	1	2	2	2	2	2	2	2	2	2	0	2	2	2	0	2
<b>Jefe Proyecto</b>	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	1	1	0	2	2
<b>Especialista Operacional</b>	2	2	1	2	2	2	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	0	0	0	2	2
<b>Especialista Técnico</b>	2	2	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	2	2	2	2

Fig. 3. Matriz que contiene los resultados de la primera ronda.

Sean:

$R_j$ : el conjunto de competencias requeridas para desempeñar el rol  $j$

$p_{sh}$ : nivel que tiene la persona  $s$  en relación a la competencia  $h$ ,  $h \in R_j$

$i_{hj}$ : importancia de la competencia  $h$  para desempeñar el rol  $j$ ;  $h \in R_j$

$w_{hj}$ : peso que representa la importancia de la competencia  $h$  para desempeñar el rol  $j$ ;  $h \in R_j$

Se quiere determinar:

$c_{sj}$ : capacidad de la persona  $s$  para desempeñar el rol  $j$

Entonces:

$$w_{hj} = \frac{i_{hj}}{\sum_{h \in R_j} i_{hj}} \text{ tal que } \sum_{h \in R_j} w_{hj} = 1$$

$$c_{sj} = \sum_{h \in R_j} w_{hj} * p_{sh}$$

**Ejemplo de cómo determinar uno de dichos índices:**

En la organización para ejecutar un proyecto se precisan dos roles: A y B y una persona en cada rol.

Para el rol A se requieren las competencias:  $c_1$ ,  $c_3$  y  $c_5$ .

Entonces,  $RA = \{1, 3, 5\}$

Se utiliza una escala de cuatro valores (1, 2, 3 y 4) para evaluar la importancia de cada competencia en cada rol.

Para el rol A:  $i_{1A} = 1$ ;  $i_{3A} = 4$ ;  $i_{5A} = 2$

Para la persona X los niveles de cada competencia (en una escala de 1 al 5) son:

$p_{X1} = 5$ ,  $p_{X3} = 1$ ,  $p_{X5} = 2$

Entonces:

$$w_{1A} = 1/7; \quad w_{3A} = 4/7; \quad w_{5A} = 2/7;$$

Y la capacidad de la persona X para desempeñar el rol A se determina como:

$$c_{XA} = w_{1A} * p_{X1} + w_{3A} * p_{X3} + w_{5A} * p_{X5} = (1/7)*5 + (4/7)*1 + (2/7)*2 = 13/7$$

Si se requiere, es posible indicar un valor mínimo para cada competencia para el desempeño de cada rol.

Sea  $u_{hj}$  el valor mínimo requerido de la competencia  $h$  para desempeñar el rol  $j$ ;  $h \in R_j$

Entonces debe cumplirse que

$p_{ih} \geq u_{hj}$  cualesquiera sean los valores  $i, h, j$

## Conclusiones

Gestionar los recursos humanos basados en el enfoque por competencias resulta muy necesario para las organizaciones de software ya que permite preservar y desarrollar su personal en una industria donde la tecnología cambia de manera vertiginosa. Los recursos humanos constituyen el factor menos formalizado en los modelos y metodologías de desarrollo de software, las cuales no ofrecen un marco de referencia bien formalizado, que contenga claramente definidas las competencias genéricas y técnicas requeridas para desempeñar cada rol establecido acorde a las características del proyecto.

La propuesta de roles invariantes para enfrentar proyectos de desarrollo e implantación de software, y de competencias tanto técnicas como genéricas para el desempeño de los roles propuestos constituye un punto de partida para la implantación de esta práctica en las empresas de software cubanas. Asimismo los procedimientos descritos para identificar las competencias requeridas en el desempeño de cada rol y la importancia de cada competencia en el rol, así como la determinación del índice de competencia de cada persona para desempeñar cada rol asignado al proyecto, representa una guía para poder enfrentar el proceso de asignación de personal a proyectos de software de forma más objetiva.

## Referencias

- Acuña, S.T. Capabilities-Oriented Integral Software Process Model. Ph.D. Thesis. Universidad Politécnica de Madrid, Madrid, 2002.
- Beck, K. Extreme Programming Explained. Embrace Change. Addison-Wesley, 1999. 224 p.
- André, M. Roles definidos por el Proceso Unificado de Rational. Reporte de Investigaciones del CEIS, 2005. ISBN: 959-261-179-3.
- André, M., M.G. Baldoquín, S.T. Acuña, A. Rosete. A formalized model for the assignment of human resources to software projects. XIV Latin Ibero-American Congress on Operations Research (CLAIO 2008). Colombia: 2008, ISBN 978 958 825283-4.
- Cockburn, A. Crystal Clear: A Human-Powered Methodology for Small Teams. Addison-Wesley, 2005.
- Cuesta, A. Tecnología de Gestión de Recursos Humanos. Academia, 2005.
- DeMarco T., T. Lister. Peopleware: Productives Projects and Teams. Dorset House, 1999. 245 p.

- Gorla, N., Y. W. Wah. Who should work with whom?: building effective software project teams. *Communications of the ACM*, 2004. Vol. 47 (6): pp 79-82.
- Jacobson, I., G. Bosch and J. Rumbaugh. *El Proceso Unificado de Desarrollo de Software*. Madrid, Addison Wesley Iberoamericana, 2000. 464 p.
- McClelland, D.C. Testing for Competencies rather than intelligence. *American Psychologist*, 1973. 28: 1-14.
- Serrano, M.A. *La Calidad del Software: Introducción a PSP y TSP y su relación con CMMI*. XI Convención Informática, La Habana, 2005.
- Pressman, R. S. *Software Engineering: A Practitioner's Approach*. McGraw-Hill Science, 2004. 880 p.
- Rodríguez, J. L. *Modelación Formal del problema de asignación de recursos humanos a proyectos de software*. Tesis de Maestría, Instituto Superior Politécnico "José Antonio Echeverría", La Habana, 2008.
- Urquiza, A. Aplicación de modelos de competencias a la gestión de sistemas de información. *Revista de Innovación, Calidad e Ingeniería del Software*, 2007, Vol.3 (1): 23-37.
- Vargas, J. G. *Las reglas cambiantes de la competitividad global en el nuevo milenio. Las competencias en el nuevo paradigma de la globalización*. 2001. *Revista Iberoamericana de Educación*. [Disponible en: <http://www.oei.es>].
- Viera, K. *Una propuesta de Proceso de Desarrollo de Software Multimedia*. Tesis de Maestría, Instituto Superior Politécnico "José Antonio Echeverría", La Habana, 2007.