

# UN SISTEMA DE SOPORTE A LA DECISIÓN PARA LA ASIGNACIÓN DE RECURSOS HUMANOS A EQUIPOS DE PROYECTOS DE SOFTWARE

Margarita André Ampuero \* y María Gulnara Baldoquín de la Peña\*\*<sup>2</sup>

\* Centro de Estudios de Ingeniería de Sistemas (CEIS), Facultad de Ingeniería Informática

\*\* Departamento de Matemática, Facultad de Ingeniería Industrial

Instituto Superior Politécnico “José Antonio Echeverría” (CUJAE), Ciudad de La Habana, Cuba

## RESUMEN

En este artículo se describen las principales características de un sistema de soporte a la decisión diseñado con el propósito de apoyar el proceso de asignación de personal en las organizaciones de software. La herramienta sustenta un modelo formal de asignación que integra factores que contribuyen a la asignación individual a los roles del proyecto y a la formación del equipo como un todo, así como algunos de los métodos y algoritmos de solución del modelo posibles a implementar. La herramienta desarrollada en Java, posee una interfaz amigable y flexible, capaz de tratar los factores ya sea como objetivos o como restricciones, y de ajustar algunos parámetros del modelo.

## ABSTRACT

In this article the main characteristics are described from a support system to the decision which is designed with the purpose of supporting the assignment process of personal in the software organizations. The tool sustains a formal model of assignment that integrates factors that contribute to the individual assignment to the roles of the project and the formation of the team as a whole, as well as some of the methods and solution algorithms possible to implement. The tool developed in Java, possesses a friendly and flexible interface, able to treat the factors as objectives as well as restrictions, and for adjusting parameters of the model.

**KEY WORDS:** assignment of personal, decision support system, formal model, software project team

**MSC:**

## 1. INTRODUCCIÓN

A pesar de los resultados y el 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 (2004). Los problemas de liderazgo, la asignación de personal inadecuado y los problemas entre los miembros del equipo de proyecto se detectan como tres de las principales dificultades asociadas con factores humanos que afectan el éxito de los proyectos de software, Ryan (2007). Esta situación pone en evidencia una insuficiente gestión de los recursos humanos.

Varias investigaciones reconocen que el personal juega un papel crítico en el éxito o fracaso de un proyecto de software, De Marco et al. (1999), Acuña et al. (2006); sin embargo, la mayor parte de las propuestas metodológicas que guían el desarrollo de software, se centran en aspectos técnicos. De hecho, al evaluar un conjunto de reconocidos modelos de procesos como: People-CMM, Curtis et al. (2001), el Proceso de Software Personal (PSP), Humphrey (1995), el Proceso de Software en Equipo (TSP), Humphrey (1998) y el Proceso Unificado de *Rational* (RUP), Jacobson et al. (2000), es posible identificar que el personal constituye el factor menos formalizado. Por lo tanto, las gerencias de desarrollo de las organizaciones de software deben implementar una adecuada gestión de sus recursos humanos, para lo cual le deben prestar especial atención al proceso de asignación de personal a proyectos.

En el proceso de asignación de personal se deben considerar múltiples factores. Por lo tanto, el proceso se torna más complejo en organizaciones medianas y grandes, debido a la gran cantidad de combinaciones de asignaciones posibles, en función de los roles y empleados disponibles. Esto hace que esta etapa sea prácticamente imposible de abordar de manera eficiente, sin la ayuda de sistemas automatizados de soporte a la decisión que se basen en algoritmos de solución de modelos matemáticos que representen el problema a resolver lo más objetivamente posible. Sin embargo, en la literatura resultan escasos los trabajos donde se modele este proceso y la mayor parte de las herramientas de gestión de proyecto se centran en la gestión de tiempo, resultando la gestión de recursos humanos una de las áreas menos cubierta junto a la gestión de calidad y de riesgos.

En este trabajo se describen las principales características de un sistema de soporte a la decisión (TEAMSOF<sup>T</sup>) que apoya a directivos de las organizaciones de software a enfrentar la asignación de personal basado en el enfoque por competencias. Para lograr este objetivo, la herramienta sustenta un modelo formal (desarrollado por las autoras) para la asignación de recursos humanos a equipos de proyectos de software (MARHEPS) el cual logra integrar factores que contribuyen tanto a la asignación individual a los roles del proyecto como a la formación del equipo como un todo.

TEAMSOF<sup>T</sup> posee una interfaz flexible y amigable, capaz de tratar los factores considerados en la asignación ya sea como objetivos o como restricciones, y de ajustar algunos parámetros del modelo. Además, implementa algunos de los métodos y algoritmos de solución del modelo propuesto, lo que posibilita realizar estudios experimentales que permitan evaluar el desempeño de diferentes metaheurísticas en la solución de esta propuesta de modelo de asignación. Como parte de la fase de implantación en una organización de software, se ha aplicado la herramienta en la solución de casos de prueba, tarea desarrollada de conjunto con especialista de recursos humanos de la empresa DESOFT. Los resultados obtenidos muestran que su aplicación contribuye a que el proceso de asignación, en general, se torne más objetivo y transparente.

El artículo está estructurado como sigue. En la Sección 2 se realiza una breve descripción del modelo formal que sustenta la herramienta. En la Sección 3 se caracteriza la herramienta a partir de su vista funcional y se describen las pruebas realizadas como parte del proceso de implantación en una empresa cubana de software. Finalmente, en la Sección 4, se plantean las conclusiones del trabajo.

## 2. MODELO FORMAL PARA LA ASIGNACIÓN DE RECURSOS HUMANOS A EQUIPOS DE PROYECTOS DE SOFTWARE

Dadas las dificultades que enfrentan las organizaciones de software, resulta vital que las gerencias de desarrollo logren implementar los diferentes procesos de Gestión de Recursos Humanos (GRH) basado en el enfoque por competencias: estructuración del proyecto (definición de roles y competencias), asignación de personal y desarrollo del equipo (capacitación, evaluación del desempeño y el desarrollo de plan de carrera, entre otros). Un interés especial merece el proceso de asignación de personal a proyectos, el cual, de manera general, se lleva a cabo de forma empírica. La necesidad de contar con una herramienta que apoye a los directivos a enfrentar este proceso se hace más evidente a medida que aumenta la cantidad de roles y empleados a considerar. Así, en organizaciones medianas y grandes se precisa contar con un sistema de soporte a la decisión que se base en algoritmos de solución de modelos matemáticos que representen el problema a resolver lo más objetivamente posible.

Sin embargo, aunque el problema de asignación ha sido objeto de estudio desde hace varias décadas, resultan escasos los trabajos donde se modele la asignación de personal en el ámbito del software y desde la perspectiva del equipo. Por lo tanto, al elaborar el Modelo de Asignación de Recursos Humanos a Equipos de Proyecto de Software (MARHEPS) que se inserta en la herramienta que se describe en este trabajo fue preciso tomar en cuenta:

- Los factores considerados en un conjunto de propuestas de modelos y procesos que abordan la asignación de personal a proyectos de software, De Carvalho (2003), Barreto (2003), Ngo-The & Ruhe (2008), Acuña et al. (2006).
- Las consideraciones enunciadas en dos reconocidos estándares de gestión de proyectos: la Guía de Fundamentos para la Gestión de Proyectos (PMBOK), PMI (2004) y la norma ISO 10006:2003, ISO (2003).
- Los principales resultados de un proceso de gestión del conocimiento donde se utilizó el método Delphi como método de consulta a expertos en la identificación de factores a considerar en el modelo de asignación, y donde se aplicaron *tests* psicológicos y herramientas de minería en la identificación de patrones que contribuyen a la formación más adecuada de los equipos de proyectos de software.

El MARHEPS, cuya primera versión se presenta en André et al. (2008), logra incorporar la mayor cantidad de elementos de los considerados en las propuestas evaluadas (competencias tanto técnicas como genéricas, características psicológicas, disponibilidad del personal y experiencia), y como elemento original, toma en cuenta el costo de realizar desarrollos a distancia e integra factores que contribuyen tanto a la asignación individual a los roles del proyecto como a la formación del equipo como un todo (incompatibilidad del equipo y balance entre las categorías de roles). A continuación se presenta el modelo y se describen brevemente sus principales elementos.

Sean:

$m$ : Cantidad de roles necesarios para desarrollar un proyecto

$n$ : Cantidad de empleados disponibles

$L$ : Cantidad máxima de roles que puede asumir un empleado en un proyecto dado

$R$ : Cantidad de conjuntos de roles incompatibles contemplados

$J_r$ : Conjuntos de roles incompatibles,  $r = 1..R$

$Z_j$ : Conjunto de competencias necesarias para desempeñar el rol  $j$   
 $A_j$ : Cantidad de empleados necesarios en el rol  $j$ ,  $j = 1..m$   
 $c_{ij}$ : Competencia neta del empleado  $i$  para desempeñar el rol  $j$ ;  $i = 1..n$ ,  $j = 1..m$   
 $v_{ih}$ : Nivel que tiene el empleado  $i$  en la competencia  $h$   
 $s_{hi}$ : Incompatibilidad existente entre los empleados  $h$  e  $i$ ;  $h, i = 1..n$ . Este coeficiente toma valor 1 si los empleados  $h$  e  $i$  mantienen relaciones recíprocamente negativas, y 0 en caso contrario.  
 $g_i$ : Carga total de trabajo del empleado disponible  $i$  según los proyectos que desarrolla actualmente;  $i = 1..n$   
 $b_j$ : Carga de trabajo que implica asumir el rol  $j$  en el proyecto de análisis,  $j = 1..m$   
 $U$ : valor máximo de carga que puede asumir un empleado  
 $l_{ij}$ : Costo del empleado  $i$  según la lejanía que tenga del proyecto y el rol  $j$  que va a desempeñar;  $i=1..n$ ,  $j=1..m$   
 $k_{hj}$ : Valor mínimo requerido de la competencia  $h$  para desempeñar el rol  $j$ ,  $j = 1..m$ ,  $h \in Z_j$   
 $ME$ : valor medio de la carga total de trabajo que incluye la de los proyectos actuales en ejecución más la que generará los roles en el proyecto de análisis.

$$ME = \frac{\sum_{i=1}^n g_i + \sum_{j=1}^m b_j}{n}$$

$B = (b_{ij})$  es una matriz de valores 0 ó 1, tomando el valor 1 si el individuo  $i$  tiene preferencia por la primera letra de cada dimensión del *MBTI* (*E/I*, *S/N*, *T/F*, *J/P*), y 0 por la segunda;  $1 \leq i \leq n$ ;  $1 \leq j \leq 4$

$D = (d_{ij})$  es una matriz de valores 0 ó 1, tomando el valor 1 si el empleado  $i$  tiene el rol de Belbin  $j$  como preferido de acuerdo a su personalidad;  $1 \leq i \leq n$ ;  $1 \leq j \leq 9$

Las variables definidas son:

$$x_{ij} = \begin{cases} 1 & \text{si el empleado } i \text{ es asignado al rol } j \\ 0 & \text{en caso contrario} \end{cases} \quad i = 1..n, j = 1..m$$

$$u_i = \begin{cases} 1 & \text{si el empleado } i \text{ es asignado al menos a un rol} \\ 0 & \text{en caso contrario} \end{cases} \quad i = 1..n$$

Las funciones a optimizar que resultan son:

$$\max \sum_{i=1}^n \sum_{j=1}^m c_{ij} x_{ij} \quad (I)$$

$$\min \sum_{h=1}^{n-1} \sum_{i>h}^n s_{hi} u_h u_i \quad (II)$$

$$\min \sum_{i=1}^n \left[ (g_i + \sum_{j=1}^m b_j x_{ij}) - ME \right]^2 \quad (III)$$

$$\min \sum_{i=1}^n \sum_{j=1}^m l_{ij} x_{ij} \quad (IV)$$

Las restricciones que resultan son:

$$\sum_{i=1}^n x_{ij} = A_j, \quad j = 1, \dots, m \quad (1)$$

$$\sum_{j \in J_r} x_{ij} \leq 1, \quad i = 1, \dots, n, \quad r = 1, \dots, R \quad (2)$$

$$\sum_{j=1}^m x_{ij} \leq L \quad \forall i = 1, \dots, n \quad (3)$$

$$(1 - u_i) \sum_{j=1}^m x_{ij} = 0, \quad (1 - u_i) + \sum_{j=1}^m x_{ij} > 0 \quad i = 1, \dots, n \quad (4)$$

$$u_i (k_{hj} - v_{ih}) \leq 0, \quad \forall i = 1, \dots, n; \forall j = 1, \dots, m, \quad \forall h \in Z_j \quad (5)$$

$$g_i + \sum_{j=1}^m b_j x_{ij} \leq U \quad \forall i \quad (6)$$

$$\sum_{k=1}^3 \left( \sum_{i=1}^n d_{ik} u_i \right) > 0 \quad \sum_{k=4}^6 \left( \sum_{i=1}^n d_{ik} u_i \right) > 0 \quad \sum_{k=7}^9 \left( \sum_{i=1}^n d_{ik} u_i \right) > 0 \quad (7)$$

$$\sum_{k=1}^3 \left( \sum_{i=1}^n d_{ik} u_i \right) > \sum_{k=4}^6 \left( \sum_{i=1}^n d_{ik} u_i \right) \quad (8)$$

$$\sum_{k=4}^6 \left( \sum_{i=1}^n d_{ik} u_i \right) > \sum_{k=7}^9 \left( \sum_{i=1}^n d_{ik} u_i \right) \quad (9)$$

$$x_{i1} \leq d_{i1} + d_{i7} \quad \forall i = 1, \dots, n \quad (10)$$

$$\sum_{i=1}^n d_{i4} u_i \geq 1 \quad \forall i = 1, \dots, n \quad (11)$$

$$x_{i1} \leq \frac{(b_{i1} + b_{i4})}{2} \quad \forall i = 1, \dots, n \quad (12)$$

Las variables de decisión  $x_{ij}$  son binarias, tomando valores 1 ó 0, dependiendo de si un empleado es o no asignado a un rol determinado, respectivamente. Las variables de decisión  $u_i$  son también binarias y toman valor 1 solo en el caso que a la persona se le asigna al menos un rol en el proyecto.

La función a optimizar (I) maximiza el factor competencia de los empleados. El dato  $c_{ij}$  indica la capacidad de los empleados para desempeñarse en los roles y toma valor como resultado de la aplicación de un modelo de evaluación de dicha capacidad, basado en las competencias demostradas por el empleado y las competencias necesarias para desempeñar cada rol. El dato  $c_{ij}$  también involucra la experiencia del empleado  $i$  en el desempeño del rol  $j$ , la cual depende, de la cantidad de veces que el empleado  $i$  ha desempeñado el rol  $j$  y de la evaluación de desempeño recibida al concluir cada proyecto.

La función (II) minimiza la incompatibilidad del equipo tomando en cuenta lo propuesto en Yang & Tang (2004) acerca de determinar el índice de conflicto entre los miembros a partir de evaluar las relaciones recíprocamente negativas existentes entre ellos.

La función (III) tiene en cuenta el factor Disponibilidad de la persona y trata de balancear la carga de trabajo del personal. Cada término de la suma en  $i$ , es un número no negativo que representa la diferencia que hay entre la carga total de trabajo que un individuo tendría, en proyectos actuales más el nuevo a considerar, y la media total de carga de trabajo considerando el proyecto nuevo y los actuales. Se eleva al cuadrado dicha diferencia para que penalice de igual forma al que sobrepase la media (sobrecargado) que el que esté por debajo de la media (desahogado).

La función (IV) optimiza el factor Costo por lejanía. Para considerar este factor es necesario contar con un dato que refleje cuán costoso es cada empleado a considerar de acuerdo a su localización con respecto a la localización del proyecto y al impacto que tiene ejecutar a distancia el rol que desempeña en el proyecto.

La restricción (1) garantiza que los roles sean cubiertos exactamente en su demanda. La restricción (2) modela la existencia de roles incompatibles. En (3) se restringe el número máximo de roles que puede asumir cualquier empleado en el proyecto. La restricción (4) garantiza que las variables  $u_i$  tomen valor 1 si el empleado  $i$  es asignado al menos a un rol en el proyecto, y valor 0 si no se le asigna ningún rol. La restricción (5) impone el cumplimiento de condiciones mínimas en cuanto a las competencias necesarias en una persona para que se le asigne un rol dado. La restricción (6) establece que la carga de trabajo total asignada a un empleado no debe sobrepasar un valor máximo.

Las restricciones (7)-(12) resultan de la aplicación de un conjunto de *tests* psicológicos a 336 personas vinculadas al desarrollo de software, experimento desarrollado de conjunto con miembros del Grupo de Diagnóstico (formado esencialmente por psicólogos) del Ministerio de la Informática y las Comunicaciones. Las restricciones (7)-(11) tienen en cuenta los llamados roles de equipo tomando como base la metodología propuesta por Meredith Belbin; quien identifica y caracteriza nueve roles agrupados en tres categorías: mentales (Cerebro, Especialista y Monitor-Evaluador), de acción (Impulsor, Implementador y Finalizador) y sociales (Coordinador, Investigador de Recursos y Cohesionador), Belbin (2004). Si se asume que en la matriz D, donde las columnas representan los roles de Belbin, las tres primeras

columnas están asociadas con los roles de acción, las columnas 4 a la 6 con los roles mentales y las tres últimas con los sociales, entonces la restricción (7) impone que en el equipo seleccionado deben estar representadas las tres categorías de roles de Belbin. La restricción (8) impone que en el equipo la preferencia por desempeñar roles de acción debe sobrepasar la preferencia por desempeñar roles mentales. En (9) se impone que la preferencia por desempeñar roles mentales debe sobrepasar la preferencia por los sociales. La restricción (10) impone que si la persona *i* desarrolla el rol Jefe de Proyecto, debe tener como preferido al menos uno de los siguientes roles de Belbin: Impulsor o Coordinador, asumiendo que en la matriz D, la preferencia por desempeñar estos roles se registra en las columnas 1 y 7, respectivamente. En (11) se impone que en el equipo al menos una persona tenga como preferido el rol Cerebro, asumiendo que en la columna 4 de la matriz D se registra la preferencia o no por este rol.

La restricción (12) se basa en la identificación de la preferencia de la persona en función del tipo psicológico a partir de uso del *test* de Myers-Briggs, Briggs (2004); uno de los tests de personalidad más utilizados en las investigaciones donde se aborda la temática de formación de equipos de software, McDonald & Edwards (2007). Esta restricción impone que si la persona *i* desarrolla el rol Jefe de Proyecto, su preferencia se debe corresponder con el subtipo EJ (E- Extroversión, J- Juicio), asumiendo que en la matriz B, la dimensión E/I se registra en la columna 1 y la J/P en la columna 4.

### 3. CARACTERIZACIÓN DE LA HERRAMIENTA TEAMSOF<sup>T+</sup>

Además de la ausencia de marcos de referencia (donde aparezcan claramente definidos los roles y las competencias requeridas para su desempeño), una de las causas que provoca que la Gestión de los Recursos Humanos basado en el enfoque por competencias no constituya aún una práctica generalizada en las organizaciones de software, es la ausencia de métodos y herramientas de software que soporten su aplicación, Urquiza (2007).

En este sentido, al analizar cómo soportan los procesos de GRH, en especial la asignación de personal, un conjunto de reconocidas herramientas orientadas a la gestión de proyectos entre las que se encuentran: Microsoft Office Project Professional 2003, Microsoft Visual Studio 2005 Team System, Teamwork, ToDoList, DotProject, RHPlan, GanttProject, Planner y Atlassian Jira, se puede concluir que la mayor parte de ellas sólo registran el personal y/o lo asignan a las tareas sin tomar en cuenta factores como: las competencias y la disponibilidad del personal. El resto de los procesos como: capacitación, evaluación del desempeño y desarrollo de planes de carrera, prácticamente, no son cubiertos. La herramienta RHPlan, De Carvalho (2003), Montoni, et al. (2005), es la única que considera la gestión por competencias y que a su vez brinda soporte a la mayor parte de los procesos; sin embargo, constituye una solución propietaria.

En este entorno se desarrolló la herramienta TEAMSOF<sup>T+</sup>, la cual constituye un sistema de soporte a la decisión, en tanto apoya a los directivos de las organizaciones de software a enfrentar la gestión de recursos humanos basado en el enfoque por competencias. El hecho de sustentar el modelo descrito anteriormente lo convierte en un instrumento especialmente útil durante el proceso de asignación de personal a los equipos de proyectos de software, aunque cabe señalar que incorpora un conjunto de funcionalidades que apoyan los procesos de capacitación, desarrollo de planes de carrera y evaluación del desempeño.

Para la elaboración de TEAMSOF<sup>T+</sup> se empleó RUP como proceso de desarrollo y UML como lenguaje de modelado. La herramienta fue desarrollada en Java, bajo el ambiente de desarrollo Eclipse 3.3, y se implementó en base a una arquitectura guiada por modelos, lo que contribuye a mejorar la productividad, la interoperabilidad, la reutilización y la calidad del código generado, facilitando el desarrollo de versiones futuras.

#### 3.1 Descripción de la herramienta a partir de la vista funcional

La herramienta se estructura en cuatro grandes módulos: Roles, Competencias, Trabajadores y Proyectos. Las funcionalidades comprendidas en cada uno de los módulos las activan cuatro actores fundamentales (en función de los perfiles de usuario identificados). Estos actores al comportarse como *Usuarios* del sistema tienen la posibilidad de Autenticarse y Cambiar la contraseña. A continuación se describen brevemente las principales funcionalidades asociadas a cada uno de los actores fundamentales, tomando en cuenta el Diagrama de Casos de Uso del Sistema que se muestra en la Figura 1.

El *Especialista de Recursos Humanos* es el encargado de Gestionar los Trabajadores (entiéndase los procesos de alta, baja y modificación del trabajador) lo que incluye tareas claves como: el registro y actualización de las competencias (tanto técnicas como genéricas), de las características psicológicas y de los intereses personales y organizacionales (en cuanto al desempeño de roles se refiere). Adicionalmente, es el encargado de Gestionar las Competencias que resultan significativas para la organización así como los indicadores de los trabajadores, como son: provincia, carrera, centro de formación, entidad laboral, entre otros. Como *Gestor de Recursos Humanos* al igual que el *Jefe de Desarrollo de la*

*Organización y el Jefe de Proyecto* (que actúan como *Decisores*) puede solicitar reportes que permitan valorar y tomar decisiones en cuanto a la Capacitación del Personal en función de los niveles reales de competencias existentes en la organización.

El *Trabajador del Proyecto* es la persona que se encuentra vinculada a algún proyecto, ya sea en ejecución o concluido. Por lo tanto, tiene derecho a consultar el avance del proyecto, los estándares definidos, su agenda de trabajo, su evaluación de desempeño y su plan de carrera.

El *Jefe de Desarrollo de la Organización* es el encargado de Gestionar los Roles, Gestionar los estándares del proyecto, Gestionar el Proyecto (que implica registro, modificación y eliminación), Cerrar el proyecto (que implica la evaluación del Jefe de Proyecto), Gestionar los indicadores asociados al proyecto, tales como: tipo de cliente, categoría, tipo, dominio, complejidad e importancia del proyecto, entre otros. Además, es el responsable de Asignar y Desasignar a los Jefes de Proyecto.

El *Jefe de Proyecto* es el encargado de Gestionar las tareas y Finalizar el proyecto, lo que incluye: registro de métricas y evaluación de los miembros del equipo para lo cual, al igual que en el caso de la evaluación del Jefe de Proyecto, es preciso: registrar las incompatibilidades con el resto del equipo, evaluar el desempeño en el rol y actualizar el nivel en cada una de las competencias que posee.

Resulta importante señalar que la herramienta estructura el proceso de asignación de personal en dos etapas, como establece PMBOK, PMI (2004): asignación del Jefe de proyecto y asignación del equipo. Así, como se explicó anteriormente el *Jefe de Desarrollo de la Organización* puede asignar al Jefe de proyecto, y ambos *Decisores* (o sea, *Jefe de Desarrollo y Jefe de Proyecto*) pueden Asignar y Reasignar el equipo de proyecto. Previo a la reasignación es preciso desasignar los miembros requeridos, indicando debidamente las causas, con vistas a evitar que el personal vuelva a ser considerado para formar parte del nuevo equipo.

En la asignación como en la reasignación, antes de establecer el equipo de proyecto definitivo, los *Decisores* pueden evaluar tantas propuestas de equipos como deseen, considerado fijo al Jefe de proyecto, quien fue previamente asignado. Durante estas evaluaciones es posible, de forma interactiva, fijar personas en roles, variar los niveles mínimos requeridos para una competencia, variar los niveles de importancia de las competencias para el desempeño de un rol, así como actualizar la incompatibilidad entre los roles que se deben cubrir en el proyecto. Adicionalmente, resulta posible ajustar algunos parámetros del modelo, seleccionar los objetivos y las restricciones a tomar en cuenta, así como seleccionar el método y el algoritmo de solución. A pesar que estas últimas facilidades están asociadas a cuestiones más técnicas, el hecho de que la herramienta posea una interfaz flexible y amigable permite que resulte fácil para el decisor interactuar con el sistema, aún sin ser conocedores de la temática de investigación de operaciones y de no dominar el modelo, y los métodos y algoritmos de solución que subyacen.

Adicionalmente, TEAMSOF+ tiene la posibilidad de integrarse con la herramienta Process Dashboard, la cual automatiza las buenas prácticas que establecen cada uno de los niveles del Proceso de Software Personal. La facilidad de integración con esta herramienta brinda la posibilidad de alcanzar un mejor control del proyecto y del desempeño del personal, en tanto facilita el registro de métricas de tiempo, tamaño y defectos. Estas métricas contribuyen a que los jefes de proyectos estén en condiciones de realizar evaluaciones de desempeño más objetivas.

### **3.2 Implementación del modelo de asignación en TEAMSOF+**

La herramienta considera como objetivos seleccionados por defecto para enfrentar el proceso de asignación de personal: maximizar las competencias del personal, balancear la carga de trabajo y minimizar las incompatibilidades del equipo. Como vía para la solución de problemas multiobjetivo, en la versión actual de la herramienta sólo se encuentra implementado el método de factores ponderados. La configuración por defecto (pesos y objetivos seleccionados) se corresponde con la votación de los expertos consultados durante el proceso de gestión del conocimiento desarrollado para elaborar el MARHEPS. No obstante, la herramienta permite modificar tanto los objetivos seleccionados con los valores de los pesos.

En la determinación del nivel de competencia que posee cada trabajador para desempeñar cada uno de los roles establecidos en el proyecto se toman en cuenta tanto el nivel de las competencias (genéricas y técnicas) como la experiencia en el desempeño del rol (la cual considera la cantidad de veces que el trabajador ha desempeñado el rol y la evaluación de desempeño recibida en cada ejecución). La herramienta permite variar el peso asignado a cada uno de estos elementos.

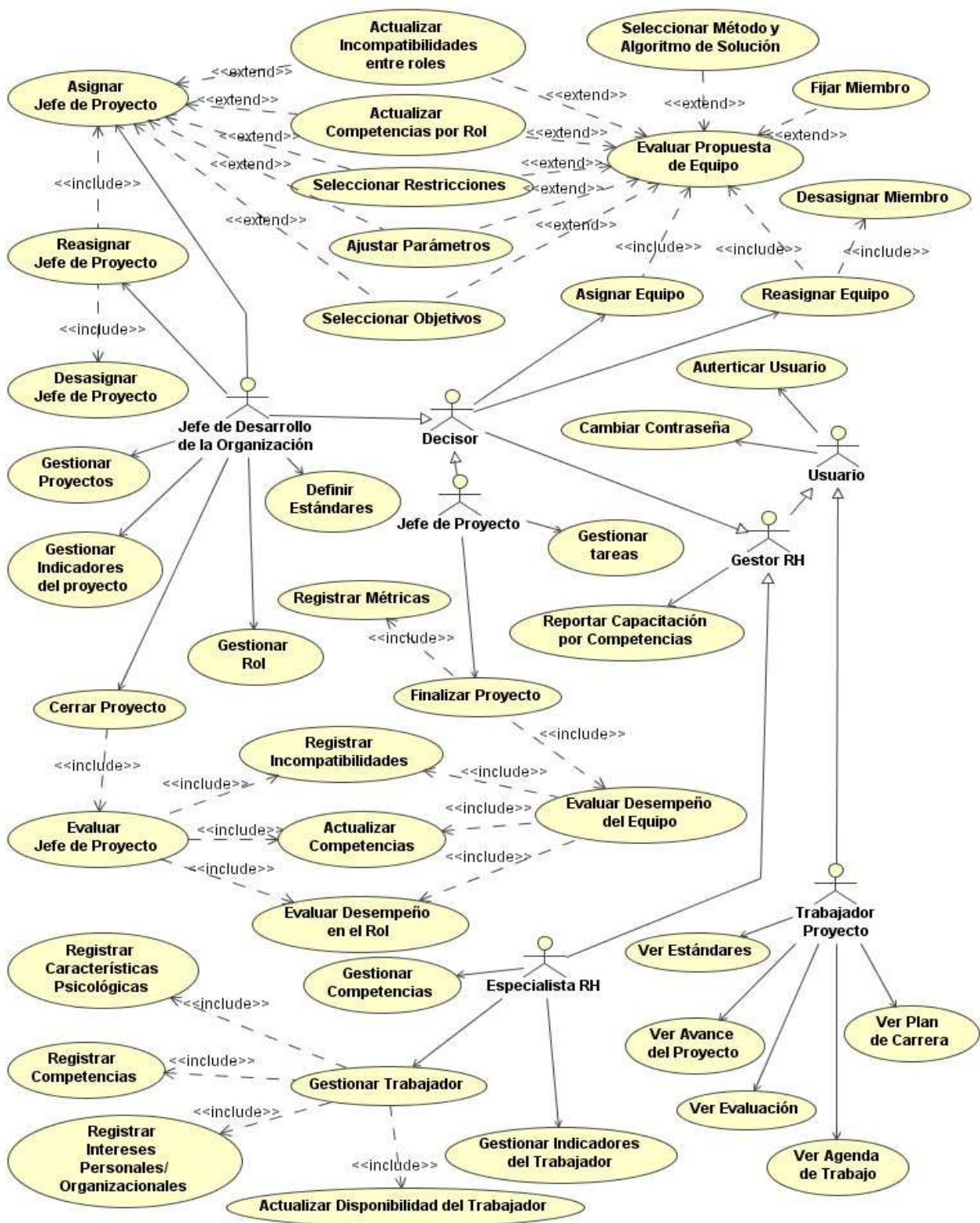


Figura 1: Diagrama de Casos de Uso del Sistema

Los algoritmos de solución del modelo implementados en la versión actual de la herramienta son: Escalador de Colinas con Reiniciación, Recocido Simulado, Búsqueda Tabú e híbridos como: GRASP y Escalador de Colinas, GRASP y Recocido Simulado, y GRASP con Tabú. La herramienta, por lo tanto, brinda la posibilidad de realizar estudios experimentales que permitan evaluar el desempeño de las diferentes metaheurísticas en la solución de esta variante del

problema de asignación formulado, en función de diferentes escenarios (diferentes tamaños de organizaciones y diferentes conformaciones de equipos).

Actualmente, la herramienta se encuentra en fase de implantación. Como parte de esta fase, TEAMSOF<sup>T</sup> ha sido aplicada en la solución de casos de prueba. Esta tarea se realizó de conjunto con especialistas de recursos humanos de la empresa DESOFT (empresa nacional encargada de la informatización de la sociedad cubana).

Los casos de prueba se generaron en función de dos escenarios:

- Escenario 1: organización formada por 60 trabajadores donde el equipo de proyecto debe cubrir 6 roles.
- Escenario 2: organización formada por 500 trabajadores donde el equipo de proyecto debe cubrir 10 roles.

Para cada escenario se generó un juego de datos que incluyó toda la información relacionada con la cantidad de trabajadores establecida y la de los proyectos en ejecución y concluidos. Las pruebas realizadas en ambos escenarios permitieron evaluar el desempeño de la herramienta en organizaciones medianas y grandes, tomando en cuenta uno de los criterios más utilizados al clasificar las organizaciones, Hunter (2004). Este criterio considera como mediana a aquella organización que tiene entre 51 y 250 empleados, y como grande, a las que poseen más de 250 empleados.

En función de esta clasificación el segundo escenario permitió evaluar el desempeño de la herramienta en organizaciones con características similares a la empresa DESOFT y la Universidad de Ciencias Informáticas (UCI), las cuales cuentan con aproximadamente la misma cantidad de trabajadores vinculados directamente a la producción (500) si se toman en cuenta: las dependencias territoriales, en el caso de DESOFT, y cada una de las facultades, en el caso de la UCI.

Las pruebas realizadas permitieron comprobar que la herramienta contribuye a que el proceso de asignación se torne más objetivo y transparente, en tanto los decisores están en condiciones de evaluar diferentes conformaciones de equipos tomando en cuenta un mayor número de factores de una manera rápida y flexible. Además, la herramienta permite realizar consultas que apoyan las decisiones que deben tomar los directivos de recursos humanos en función de la capacitación, la evaluación y la promoción del personal basado en el enfoque por competencias.

## 5. CONCLUSIONES Y TRABAJOS FUTUROS

La herramienta TEAMSOF<sup>T</sup> sustenta un modelo formal para la asignación de personal a equipos de proyectos de software (MARHEPS), así como algunos de sus métodos y algoritmos de solución. Este hecho añade valor ya que garantiza que la herramienta contribuya tanto a la asignación individual a los roles del proyecto como a la formación del equipo como un todo y contribuye a que el proceso de asignación, en general, se torne más objetivo y transparente, en tanto toma en consideración mayor cantidad de factores. El uso de la herramienta en la solución de los casos de prueba permite concluir que resulta flexible, que posee una interfaz amigable, y que constituye un instrumento útil para la gestión de recursos humanos basado en el enfoque por competencias, en especial como soporte al proceso de asignación de personal en organizaciones de software medianas y grandes.

Debido al alcance y los objetivos propuestos en este trabajo no se ha profundizado en aspectos tales como:

- La fase de deducción del MARHEPS
- El valor que toman los parámetros en cada uno de los algoritmos de solución implementados

Como trabajos futuros se proyecta desarrollar una nueva versión de la herramienta que incorpore otros métodos y algoritmos de solución del modelo.

RECEIVED OCTOBER 2008  
REVISED JULY 2009

## REFERENCIAS

- [1] ACUÑA, S.T., N. JURISTO and A. M. MORENO (2006): Emphasizing Human Capabilities in Software Development. **IEEE Software**. 23, 94-101.
- [2] ANDRÉ, M., M. G. BALDOQUÍN, S. T. ACUÑA, and A. ROSETE (2008): A formalized model for the assignment of human resources to software projects. **XIV Congreso Latino Ibero Americano de Investigación de Operaciones (CLAIO 2008)**, Eds. J. Amador, C. Paternita, J. Velázquez, . 115, Cartagena de Indias, Colombia.
- [3] BARRETO, A. S. (2003): **Apoio à Decisão Gerencial na Alocação de Recursos Humanos em Projetos de Software**. Tesis de maestría. COPPE/Universidade Federal do Rio de Janeiro.
- [4] BELBIN, R. M. (2004): **Management Teams: Why they Succeed or Fail**. 2<sup>nd</sup> ed., Butterworth



Heinemann, Oxford.

- [5] BRIGGS, I., L. KIRBY and K. D. MYERS (2004): **Introducción al Type (MBTI):. Una guía para entender los resultados de su evaluación Myers-Briggs Type Indicador**. Sexta ed. Consulting Psychologists Press, California.
- [6] CURTIS, B., W. E. HEFLEY, AND S. A. MILLER (2001): Capability Maturity Model (P-CMM): Version 2.0. **Technical Report CMU/SEI-2001-MM-01**, Software Engineering Institute.
- [6] DE CARVALHO, L. R. (2003): **Planejamento da alocação de recursos humanos em Ambientes de desenvolvimento de software orientados à Organização**. Tese para a obtenção do grau de mestre em ciências em engenharia de sistemas e computação. COPPE/Universidade Federal do Rio de Janeiro.
- [7] DE MARCO, T. and T. LISTER (1999): **Peopleware: Productive Projects and Teams**. Dorset House, New York.
- [8] HUMPHREY, W. S. (1995): **A Discipline for Software Engineering**. Addison-Wesley Longman, Boston.
- [9] HUMPHREY, W. S. (1998): **Managing Technical People: Innovation, Teamwork and the Software Process**. Addison-Wesley Longman, Boston.
- [10] JACOBSON, I., G. BOOCH AND J. RUMBAUGH (1999): **The Unified Software Development Process**. Addison-Wesley Longman, Boston.
- [11] HUNTER, G.M. (2004): Information Systems & Small Business: Research Issues. **Journal of Global Information Management**. 14, 1-5.
- [12] ISO (2003): ISO 10006:2003: Quality management systems-Guidelines for quality management in Projects. <http://www.iso.org>
- [13] MCDONALD, S. and H. M. EDWARDS (2007): Who Should Test Whom?. Examining the use and abuse of personality tests in software engineering. **Communications of the ACM**. 50, 67-71.
- [14] MONTONI, M. A., G. SANTOS, K. V. LIMA, A. R. ROCHA, G. H. TRAVASSOS, S. FIGUEIREDO, S. MAFRA, A. Albuquerque, P. MIAN (2005): Enterprise-Oriented Software Development Environments to Support Software Products and Processes Quality Improvement. **Lecture Notes in Computer Science**, 3547, 12-16.
- [15] NGO-THE, A. and G. RUHE. (2008): A Systematic Approach for Solving the Wicked Problem of Software Release Planning. **Soft Computing**, 12, 95-108.
- [16] PMI (2004): **A Guide to the Project Management Body of Knowledge (PMBOK® Guide)**.. Third ed. Project Management Institute, Pennsylvania.
- [17] PRESSMAN, R. S. (2004): **Software Engineering: A Practitioner's Approach**. McGraw-Hill Science, New York.
- [18] RYAN, R. (2007): IT Project Management: Infamous Failures, Classics Mistakes, and Best Practices. **Mis Quarterly Executive**. 6, 67-78.
- [19] URQUIZA, A. (2007): Aplicación de modelos de competencias a la gestión de sistemas de información. **Revista de Innovación, Calidad e Ingeniería del Software**, 3, 23-37.
- [20] YANG, H. L. and J. H. TANG (2004): Team Structure and Team Performance in IS Development: A Social Network Perspective. **Information & Management**, 41, 335-349.