

Docencia en la Gestión de Proyectos de Desarrollo de Software: uso de la Dinámica de Sistemas

Ricardo Colomo Palacios, Ángel García Crespo, Antonio Amescua Seco

Dpto. de Informática

Universidad de Carlos III de Madrid

28911 Leganés (Madrid)

e-mail: {ricardo.colomo, angel.garcia, antonio.amescua}@uc3m.es

Resumen

La carencia de gestores competentes de proyectos de desarrollo de software ha sido detectada como uno de los problemas que pueden justificar la denominada “crisis del software”. Una de las soluciones que se apuntan para paliar este problema es el uso de simuladores basados en la dinámica de sistemas para entrenar a este tipo de gestores, necesitados de conocimientos teóricos y formación práctica para el desempeño superior de sus tareas. El presente artículo describe la experiencia docente llevada a cabo en el curso 2004/05 con un doble objetivo: primeramente, desarrollar de forma colaborativa modelos basados en la dinámica de sistemas y, en segundo lugar, utilizar los modelos desarrollados como vehículo experiencial para la formación de alumnos de último curso de Ingeniería Informática en las competencias propias de los gestores de proyectos de desarrollo de software.

1. Introducción

Boehm subrayó la importancia de la gestión de los proyectos de desarrollo de software afirmando que una gestión pobre puede aumentar los costes del software más rápidamente que cualquier otro factor [4], adicionalmente, recientes investigaciones demuestran la conveniencia del empleo de la dinámica de sistemas para la formación de gestores de proyectos de desarrollo de software [16], [18]. Muchas universidades forman a los estudiantes de ingeniería informática para la gestión de proyectos de desarrollo de software mediante lecciones de tipo conferencia

que no aportan la formación práctica, que, para este tipo de proyectos es tan necesaria, como difícil abordar. La simulación está a medio camino entre la clase teórica y la experiencia práctica, y ha sido adoptada por áreas profesionales diversas en campos como el militar, el del entretenimiento o el de la ingeniería civil. La gestión de proyectos de desarrollo de software es altamente compleja, comparable al pilotaje de aviones en condiciones difíciles [16]. Algunas de las competencias que se desarrollan con este tipo de simulaciones son las relativas a planificación, seguimiento y control, mientras que otras competencias, como el liderazgo, la comunicación o el trabajo en equipo, necesitan de otros métodos de entrenamiento.

Teniendo en cuenta, por una parte las carencias de la formación práctica de gestores de proyectos de desarrollo de software, y por otra, los beneficios del uso de herramientas de simulación para la educación en Ingeniería Informática, se ha implantado dentro del marco de una asignatura optativa de quinto curso en la Universidad Carlos III de Madrid la utilización de dinámica de sistemas para la simulación de situaciones relativas al rol de gestor de proyectos.

El artículo, tras realizar una introducción teórica de la Dinámica de Sistemas en el apartado 2, describe la asignatura marco de la iniciativa en el apartado 3, expone los resultados docentes en el punto 4 y concluye en el apartado 5 con las principales conclusiones y mejoras al modelo y su aplicación.

2. Dinámica de Sistemas

2.1. Origen y aplicaciones.

La Dinámica de Sistemas nace a finales de los años 50 cuando Jay Forrester del M.I.T. estudia por encargo de General Electric la demanda de una planta situada en Kentucky. De su análisis surge el artículo “*Industrial Dynamics a major Breakthrough for decision makers*” [9] que posteriormente se amplía en el trabajo “*Industrial Dynamics*” [10]. El propósito de la Dinámica de Sistemas es facilitar la comprensión de la relación entre el comportamiento del sistema a lo largo del tiempo y su estructura, políticas y reglas de decisión [23]. La dinámica industrial, denominación adoptada por Forrester, se transformó en Dinámica de Sistemas en el momento en que se comenzó a usar en diferentes áreas que poseían un sistema continuo: desarrollo urbano, sociología o economía. Actualmente la Dinámica de Sistemas tiene numerosas aplicaciones en ámbitos científicos, sociales o de negocio [5], para lo que ha resultado crucial la aparición de paquetes de software (VenSim, iThink, Powersim Studio...) que han simplificado la codificación de los modelos gracias al empleo de elementos gráficos que sustituyen las sentencias tipo FORTRAN de las primeras herramientas (Dynamo, SimScript, GPSS...).

2.2. La dinámica de sistemas en los proyectos de desarrollo de software.

La aplicación de la dinámica de sistemas en la gestión de proyectos de desarrollo de software tiene su origen en el uso para la gestión de proyectos de la dinámica de sistemas iniciada por Roberts en 1964 [20]. Tomando como referencia la línea de investigación del profesor Roberts, la primera aplicación de la dinámica de sistemas al proceso de desarrollo de software es el modelo desarrollado por Abdel-Hamid y Madnick [1]. El modelo, que plasma los elementos de gestión para un proyecto pequeño o mediano de desarrollo de software utilizando el modelo de construcción en cascada, ha servido de base para numerosos modelos posteriores, que tratan, bien de

complementar el modelo general mediante su aplicación en una determinada organización [13], [14], [22] o bien analizar y resolver problemas de gestión específicos [7], [15], [6], [2].

Mención aparte merece el denominado “Modelo Dinámico Reducido” [21]. El modelo desarrollado en España por investigadores de la Universidad de Sevilla, es obtenido por simplificación del modelo de Abdel-Hamid y Madnick [1], alcanzando la reducción de los diferentes factores un 51%. De esta forma, dada su simplicidad es en especial útil a la hora de mostrar a personal novel en gestión de proyectos de desarrollo de software el comportamiento básico de un proyecto de estas características.

2.3. Los modelos de Dinámica de Sistemas aplicados a la formación de gestores de proyectos de desarrollo de software.

El potencial de los simuladores para el entrenamiento de gestores está plenamente justificado [18]. Los entornos de tipo “simulador de vuelo” enfrentan a los gestores con situaciones reales lo que les permite formarse sin los riesgos del mundo real. El uso específico para el entrenamiento en proyectos de desarrollo de software cuenta con experiencias plenamente documentadas. Por una parte, la *Australian Computer Society* auspicia desde 1994 un simulador del tipo “Juego de Guerra” sobre la temática del desarrollo de software que es actualmente utilizado por diversas firmas de consultoría locales para la formación de sus gestores de proyecto [24].

En el ámbito universitario se ha demandado la presencia de dinámica de sistemas en la totalidad de los estudios de ingeniería [19] y más concretamente en el ámbito de la Ingeniería del Software, diversos autores han implantado con éxito el uso de la dinámica de sistemas para la educación universitaria en la Ingeniería de Software, tanto en Europa [18] como en Estados Unidos [3], [16].

Sin embargo, la probada valía de la modelación para la formación de los gestores de proyectos de desarrollo de software no ha servido para incluir en la recomendación curricular de la Ingeniería del Software [12] (la más reciente de las que constituyen el esfuerzo curricular conjunto de ACM e IEEE relativo a la Informática)

materias relativas a la dinámica de sistemas aplicada a la gestión de proyectos de desarrollo de software.

2.4. Enseñanza de la Dinámica de Sistemas en la educación superior en España.

La enseñanza de la dinámica de sistemas en entornos universitarios para programas de primer y segundo ciclo tiene una distribución limitada. Los campos por los que se extiende su docencia abarcan tanto las ingenierías como los relacionados con la gestión de recursos naturales o biológicos y la simulación del comportamiento humano.

La dinámica de sistemas es aplicada en la docencia de la Ingeniería Informática tanto técnica como superior generalmente para ilustrar la simulación de sistemas continuos mediante los diagramas de Forrester y los causales. Ejemplos de este tipo de docencia son los que se imparten en la Universidad de Málaga, Huelva, Alicante o Almería, entre otras universidades españolas.

En lo que al uso de la dinámica de sistemas para los proyectos de desarrollo de software se refiere, dos iniciativas docentes merecen ser mencionadas. Primeramente, la Universidad del País Vasco, a través de la asignatura optativa de segundo ciclo “Métricas y Modelos en la Ingeniería del Software” [25] apuesta por el entrenamiento en la gestión de proyectos utilizando las herramientas Stella y Vensim y en segundo lugar, la Universidad de Sevilla, cuenta con la asignatura troncal de quinto curso “Ingeniería de Software III” [26] en la que, entre otros contenidos, se aborda el modelo desarrollado por Abdel-Hamid y Madnick [1].

3. La asignatura “Control de la Producción Software”

‘Control de la producción software’ es una asignatura optativa cuatrimestral de quinto curso de 6 Créditos de la titulación: Ingeniero en Informática de la Universidad Carlos III de Madrid. El curso académico 2004/05 ha sido el primero en el que la asignatura se ha impartido, habiéndose matriculado un total de 12 alumnos.

El objetivo de la asignatura es dotar a los alumnos de mecanismos para el desempeño excelente del rol de Jefe de Proyecto mediante la

identificación, definición y evaluación de los parámetros cruciales para la gestión de un proyecto software. Dichos parámetros, tanto cualitativos como cuantitativos, se extraen tanto de conceptos globales de las normas de estandarización como de los modelos de madurez para proyectos de desarrollo de software, y su aplicación tiene el objetivo de proporcionar al gestor un conjunto de factores que reflejen el estado del proyecto y le permitan controlar su avance.

3.1. Dinámica del curso

La asignatura tiene un carácter eminentemente práctico. Tras exponer profusamente el marco teórico expuesto anteriormente, se abordan los conceptos básicos de Dinámica de Sistemas y de los modelos específicos para proyectos de desarrollo de software: los modelos de Abdel Hamid [1] y el Modelo Dinámico Reducido [21]. Asumiendo la máxima del profesor Pestel, autor de la modelización del mundo en 1974 [17], que ha afirmado en alguna ocasión que la técnica de dinámica de sistemas se puede aprender en menos de veinticuatro horas, se ha diseñado un plan por el que el alumno debe entregar, utilizando para ello la herramienta VENSIM [27], primeramente ejemplos simplificados relacionados con la dinámica de los proyectos de desarrollo de software, para posteriormente construir de forma pautada su sistema de control de proyectos de desarrollo de software basado en un subconjunto de indicadores.

La experimentación con las variaciones que los distintos factores implican en la gestión de un proyecto de desarrollo de software supone una experiencia que constituye para el alumno su primera prueba situacional como Jefe de Proyecto, y le enfrenta ante circunstancias que fuera de un modelo dinámico, no se pueden reproducir de forma concurrente.

La nota final depende de diversos factores, entre los que destacan la implementación del modelo realizado utilizando la herramienta VENSIM [27] y el informe que debe elaborar en el que reproduce su solución a una situación concreta en la que inciden diversos factores del modelo. Los criterios numéricos de los diferentes

factores que determinan la nota del alumno se incluyen a continuación:

- Participación (10%).
- Modelización en VENSIM (50%)
- Informe (30%)
- Presentación oral (10%)

4. Resultados obtenidos

Los distintos grupos de alumnos han desarrollado diferentes modelos abordando la problemática de la gestión de proyectos de desarrollo de software, con especial atención a los aspectos de productividad, entre los que se encuentran la productividad grupal, la motivación, las capacidades y conocimientos de los integrantes del proyecto, y la diferenciación entre horas laborales y horas productivas.

Los diversos aspectos que se han modelizado utilizando la herramienta VENSIM [27] están presentes en las figuras 1, 2, 3, y 4 siendo extraídos de diversas publicaciones relacionadas con la gestión de proyectos de desarrollo software [8], [11] y de la experiencia de los alumnos en la realización de proyectos de desarrollo de software en diferentes momentos de la carrera.

Para la realización del modelo se han tomado como referencia los modelos citados [1], [21], simplificando en lo posible su complejidad, pero añadiendo aspectos relacionados con la motivación, del entorno laboral, y de la productividad personal y del equipo, como se muestra en la figura 1, cubiertos con menor profundidad por los modelos de de Abdel Hamid [1] y el Modelo Dinámico Reducido [21], y que se han considerado importantes para la formación de responsables de proyecto sensibilizados con las denominadas *Soft Skills*.

El primer modelo (figura 1) que los estudiantes desarrollan de modo colaborativo presenta cuatro variables de nivel que determinan el comportamiento del sistema:

- Trabajo restante
- Trabajo realizado
- Personal
- Coste

Las variables, en función de los diferentes valores de las constantes permiten detectar el estado del proyecto y actuar de forma simulada sobre las constantes de modo que, bajo la premisa de la invariabilidad del plazo de entrega, los costes y el personal se verán modificados por las acciones que los alumnos emprenden en el entorno de la herramienta.

El modelo cuenta adicionalmente con 88 variables adicionales, entre las denominadas variables auxiliares, cuyos valores no dependen de sus valores en anteriores momentos de la simulación, y constantes, valores que no pueden ser modificados durante la ejecución de una simulación convencional. En este último caso, y con el objeto de servir de “simulador de vuelo” se han utilizado este tipo de objetos para que los alumnos experimenten con sus valores en las diferentes situaciones que se les presentan.

Un caso típico que contempla el modelo es el decrecimiento de alguno de los factores de motivación, típicamente salario, condiciones de trabajo o carrera profesional, que hace descender las horas productivas a través de su incidencia en la productividad, y en consecuencia, los costes. Otra de las contingencias que el alumno debe afrontar y modelizar es la disyuntiva entre la pérdida a corto plazo en la productividad originada por la formación en horario laboral, y los beneficios que reporta a medio plazo en el mejoramiento de capacidades, y por ende, de la productividad.

Como se ha señalado anteriormente, la focalización en los aspectos relacionados con la productividad, no ha permitido en el curso académico 2004/05 el crecimiento del modelo y la interconexión con factores como el reclutamiento o la movilidad del personal asignado al proyecto, que serán abordados en futuros años.

El número de alumnos matriculados en la asignatura, doce en total, permitió la creación de dos grupos de trabajo dedicados al desarrollo del modelo y permitió una supervisión y una colaboración alumno-profesor que cristalizó tanto en la calidad del modelo, como en la reducida tasa de ausentismo durante la docencia de la asignatura.

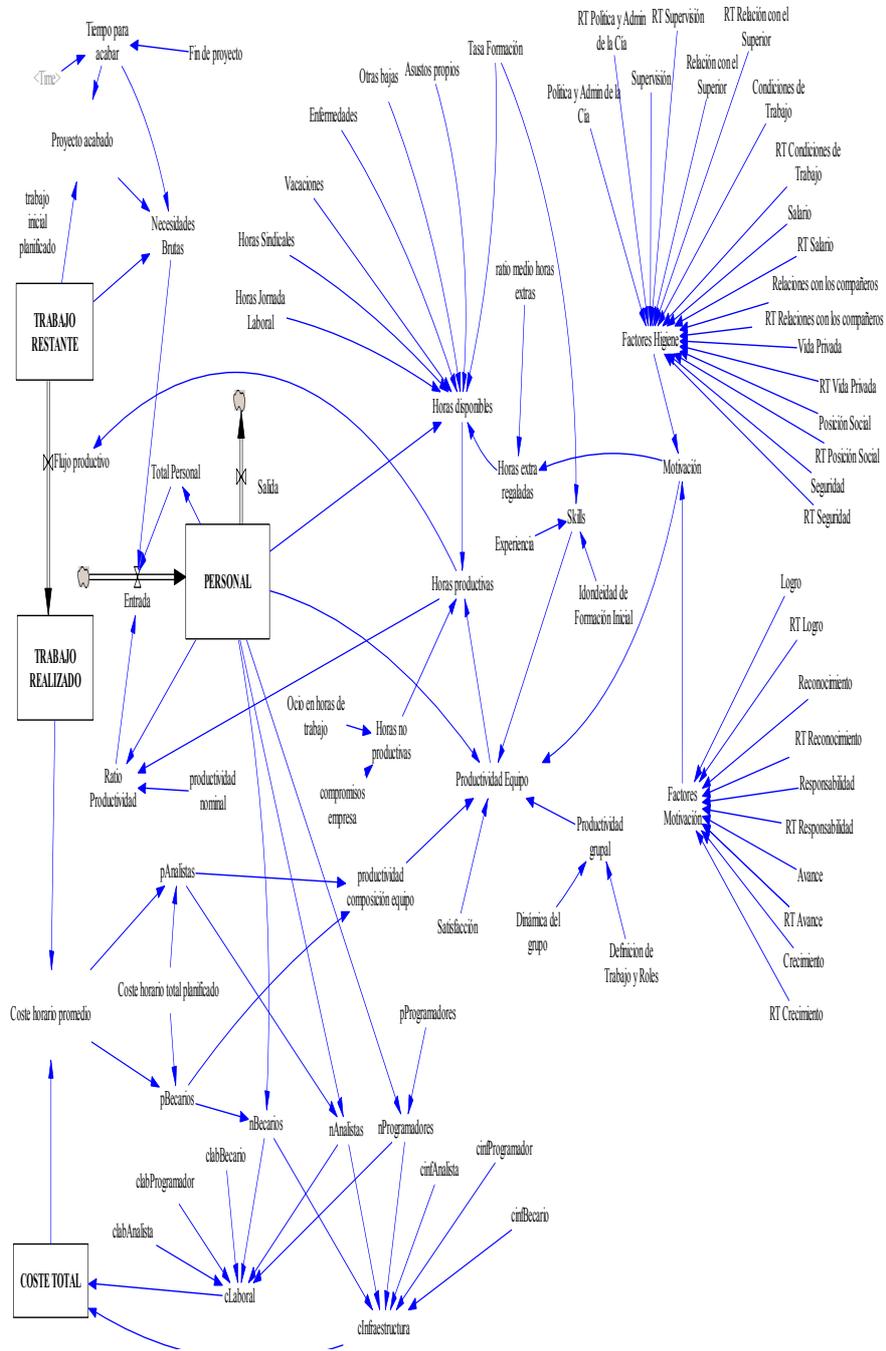


Figura 1. Implementación del modelo de productividad

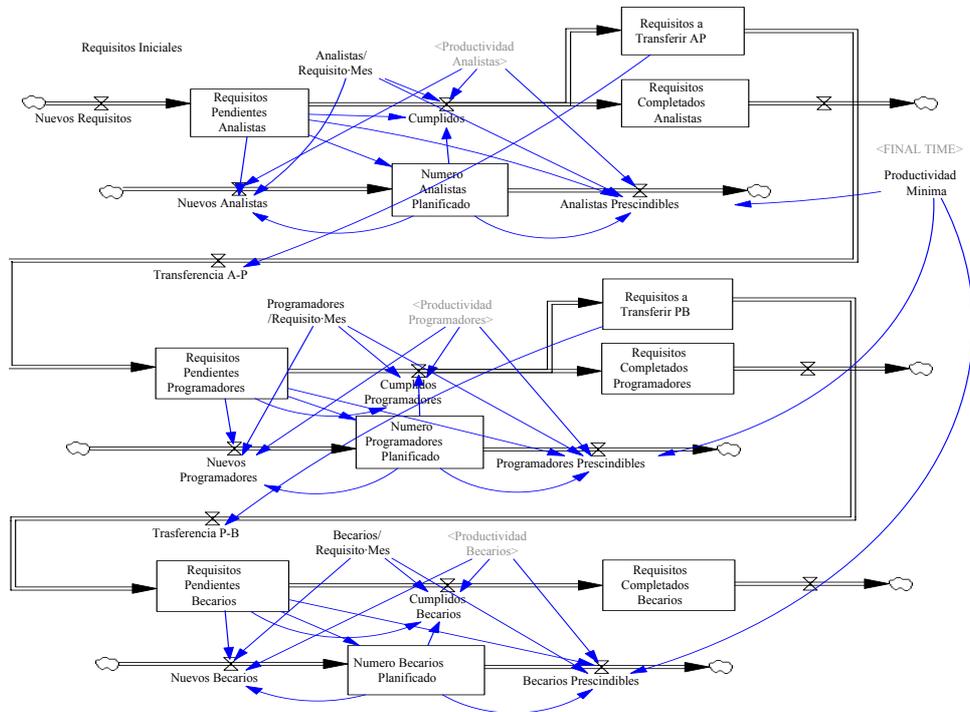


Figura 2. Presión y reductores

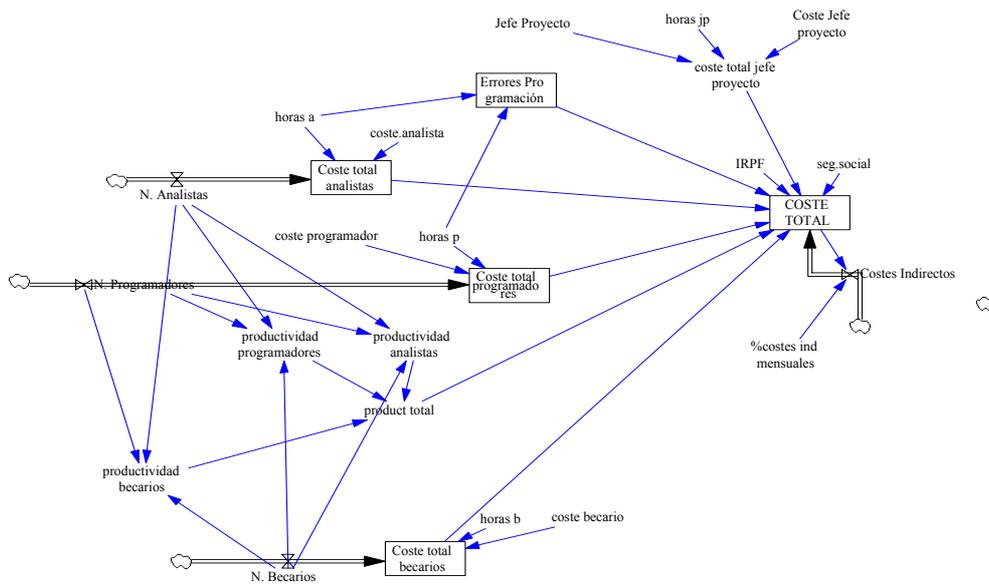


Figura 3. Efectos de la presión y sus reductores

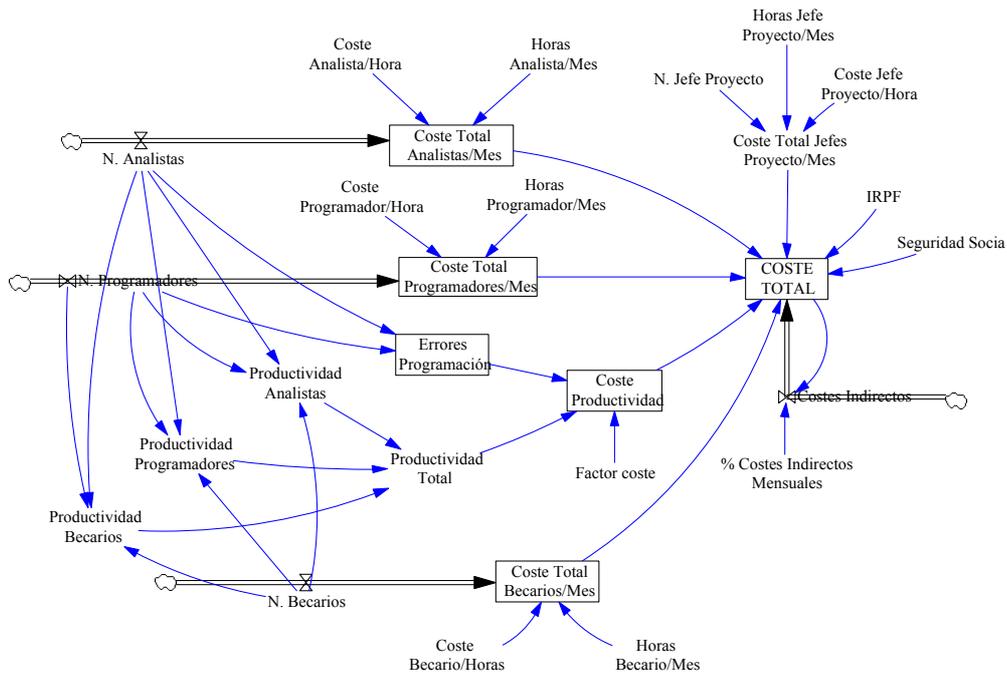


Figura 4. Dimensionamiento de un proyecto en función del tipo de personal

5. Conclusiones

Si bien la inmersión de los profesores de la asignatura ha conllevado un elevado número de horas para la preparación de la misma, dividida en la búsqueda de información, aprendizaje de la herramienta, preparación de las clases teóricas y seguimiento del trabajo práctico de los alumnos, dicho esfuerzo se ha visto recompensado por los resultados obtenidos. La comprensión de los distintos parámetros con los que tiene que 'jugar' un gestor de proyectos y su interrelación han sido fácilmente transmitidos y comprendidos por los alumnos, aunque en determinados momentos veían la herramienta como un fin y no como un medio para analizar los factores. Pero una vez realizado el modelo y puesta en marcha la simulación, su interés aumentaba y podían

observar como una variación de alguno de los factores generaba situaciones en principio imprevistas que tenían la necesidad de gestionar.

Las simulaciones han permitido el debate entre los grupos de trabajo, aumentando la participación e interés del alumnado.

Referencias

- [1] Abdel-Hamid T. K., Madnick S. E., *Software Project Dynamics: An Integrated Approach*, Prentice-Hall 1991.
- [2] Aranda, R.R., Fiddaman, T. & Oliva, R., *Quality microworlds: modeling the impact of quality initiatives over the software product life cycle*. American Programmer, 6, 5, 52-61 1993.

- [3] Baker, A.; Navarro, E.O. & van der Hoek, A. *An Experimental Card Game for Teaching Software Engineering*. Conference on Software Engineering Education and Training. 2003
- [4] Boehm, B., *Software Engineering Economics*, Prentice-Hall. 1981.
- [5] Cauldfield, C.W. & Paul Maj, S. *A Case for system dynamics*. Global Journal of Engineering Education, Vol 6 N° 1, 2002.
- [6] Chichakly, K.J., *The bifocal vantage point: managing software projects from a systems thinking perspective*. American Programmer, 6, 5, 18-25, 1993.
- [7] Collofello, J.S.; Tvedt, J.; Yang, Z., Merrill, D., & Rus I., *Modeling Software Testing Processes*, Proceedings of Computer Software and Applications Conference, 1995.
- [8] Cuevas, G.: *Gestión del Proceso Software*. Centro de Estudios Ramón Areces. Madrid, 2002.
- [9] Forrester, J. W. *Industrial Dynamics: A Major Breakthrough for Decision Makers*. Harvard Business Review 36(4):37-66. 1958
- [10] Forrester, J.W., *Industrial Dynamics*. Waltham: Pegasus Communications, 1961.
- [11] Herzberg, F.: *One more time: How do you motivate employees?*. Harvard Business Review, Sep – Oct 1987, pp 109-120.
- [12] LeBlanc, R. & Sobel, A. K. *Software Engineering* 2004. <http://sites.computer.org/ccse>
- [13] Lin C.Y., *Walking on Battlefields: Tools for Strategic Software Management*, American Programmer, Mayo 1993, pp. 33-40.
- [14] Lin C.Y., Abdel-Hamid T., Sherif J.S., "Software-Engineering Process Simulation Model (SEPS)", Journal of Systems and Software, 38, 1997, pp. 263-277.
- [15] Madachy, R., *System Dynamics Modeling of an Inspection-based Process*, Proceedings of the 18th International Conference on Software Engineering, Berlin, Germany, 1996.
- [16] Merrill, D. & Collofello, J.S.. *Improving Software Project Management Skills using a Software Project Simulator*. Frontiers in Education Conference 1997.
- [17] Mesarovic, M. & Pestel, E., *Mankind at the Turning Point, The second Report to the Club of Rome*, 1974
- [18] Pfahl, D.; Koval, N. & Ruhe, G. *An Experiment for Evaluating the Effectiveness of Using a System Dynamics Simulation Model in Software Project Management Education*. Proceedings of the Seventh International Software Metrics Symposium. 2001
- [19] Radzicki, M.J. & Karanian, B.A. *Why every engineering student should study System Dynamics*. Frontiers in Education Conference. 2002
- [20] Roberts, E.B. *The Dynamics of Research and Development*. Tesis Doctoral, M.I.T., 1964
- [21] Ruiz, M.; Ramos, I. & Toro, M. A simplified model of software Project dynamics. The journal of Systems and Software. 59, 299-309. 2001
- [22] Smith BJ, Nguyen N, Vidale RF, "Death of a Software Manager: How to Avoid Career Suicide through Dynamic Software Process Modeling", American Programmer, May 1993, pp. 10-17.
- [23] Wolstenholme, E.F., *System Enquiry: a System Dynamics Approach*. John Wiley & Sons, 1990.
- [24] Yourdon, E. *Death March*. Prentice Hall. 2004
- [25] <http://www.sc.ehu.es/jiwdocoj/mmis/mmis-info.htm>
- [26] http://www.lsi.us.es/docencia/pagina_asignatura.php?id=16
- [27] <http://www.vensim.com>