

Diseño de un prototipo de sistema informático para la gestión de espacios físicos en ámbitos de la Educación Superior Pública

Sebastian Martinez, Pedro Alfonzo, Sonia I. Mariño

¹ Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura. 9 de Julio 1449. 3400. Corrientes
{seba_martinezz, plalfonzo}@hotmail.com
simarinio@yahoo.com

Resumen.

Se presentan los resultados preliminares referentes al modelado de un sistema informático de gestión de espacios físicos implementable en el ámbito de la Educación Superior pública. Se compone de cuatro secciones. La primera resume el estado del arte, en la segunda se menciona la metodología, en la tercera se sintetizan las funcionalidades de los software generados. Finalmente se exponen las consideraciones finales.

Palabras claves: Educación Superior, administración espacios físicos, gestión, sistemas informáticos

1 Introducción

En los ámbitos de la Educación Superior, la asignación de los de espacios físicos como son los laboratorios, salas de conferencias y aulas rige generalmente bajo la responsabilidad de “bedeles” como son: horario de inicio, horario de finalización y profesor asignado.

Actualmente existen soluciones informáticas orientadas a apoyar la gestión administrativa en la Educación Superior, así como algunas específicas para la administración de aulas [1], [5] y [17].

Se coincide con [3], en que el problema de la asignación óptima de aulas ha sido tratado con diversas técnicas, herramientas y tecnologías. Además, siendo escasos productos software, que determinen la mejor distribución de espacios en función de las características edilicias y de las franjas horarias y de su cupo.

Trabajos elaborados por instituciones de nivel universitario, que abordan esta problemática, desde diversas perspectivas se detectaron. En [3], se presenta el diseño de una solución basada en inteligencia artificial, específicamente utilizando algoritmos genéticos multi-objetivo que realizan la asignación de aulas y docentes en forma conjunta, para automatizar esta tarea y mejorar el uso de los espacios y la asignación de los docentes. Por otra parte, [1] describe un sistema para la generación y asignación de los horarios, el cual permite almacenar la carga académica por aulas, por profesor y por semestre. En [17], se desarrolla un sistema de gestión de asignación de aulas y laboratorios de acorde con las necesidades y requerimiento de

cada carrera. En [5], se propone el modelo de gestión de laboratorios para prácticas docentes en carreras técnicas desarrollado por dicho personal.

Además, existe software de pago desarrollado por empresas que implementan parte del trabajo necesario en el mantenimiento de un laboratorio, por ejemplo [16] y [20], aunque no cubren la totalidad de las necesidades de gestión para un laboratorio ([5]).

También, la revisión permitió detectar los denominados “Sistemas de reservas”, que se utilizan para gestionar salas de conferencias u otro tipo de espacio para los cuales se establece un turno previamente. Además, existen herramientas destinadas a la gestión de alumnos, profesores, notas, salones, aulas entre otras funcionalidades, como por ejemplo MRBS [13], JSchoolManager [8] y Softaula [21], entre otros.

En este trabajo, se propone el diseño, desarrollo y la transferencia de un sistema informático orientado a la gestión de espacios físicos parametrizable, incorporando aspectos relevantes de accesibilidad y acceso a través de tecnología móvil.

El prototipo de sistema informático para la gestión de espacios físicos se modeló considerando como contexto de implementación la Facultad de Ciencias Exactas y Naturales y Agrimensura (FaCENA), sede 9 de Julio, de la Universidad Nacional del Nordeste (UNNE), situada en Corrientes, Argentina con miras a proporcionar una solución fiable y acorde con las necesidades y requerimientos de las distintas carreras.

2 Metodología

La metodología propuesta en la construcción del prototipo de sistema informático para la gestión de espacios físicos consta de las siguientes etapas, basada en la propuesta expuesta por Mariño y Godoy [11].

- Análisis y diseño de un sistema informático para la gestión de espacios físicos, integrando los conceptos previamente estudiados, seleccionados y adoptados para su utilización, con miras a su implantación.
 - Análisis de factibilidad, de requerimientos funcionales y no funcionales del sistema, definición de limitaciones tecnológicas. Se definirá el producto software y se determinará su factibilidad en el ciclo de vida desde la perspectiva de la relación costo-beneficio.
 - Especificación de requerimientos del prototipo. Se detallarán las funciones solicitadas, las interfaces y el rendimiento del producto [9]. Se considerarán incrementos en porcentajes de la funcionalidad total del sistema.
 - Definición de seguridad en el acceso a la información. Los aspectos de seguridad en el acceso y manipulación de los datos, darán lugar a establecer medidas para prevenir cualquier tipo de problemas tanto externos como internos que puedan influir en el desempeño normal de las organizaciones. En los productos desarrollados se diferenciarán dos subsistemas: de administración o back-end y el acceso público o front-end. Asimismo, se definirán distintos perfiles de usuarios, cuyas funciones se representarán en diagramas proporcionados por modelos y técnicas de la ingeniería de requerimientos, como son diagramas de

- casos de uso, diagramas de secuencias, entre otros [18]. En la Figura 1, se ilustra la interacción de los usuarios con el sistema.
- Diseño de interfaces. Los diversos perfiles de usuarios a la cual estará dirigido los productos de software determinarán el diseño del sistema informático. La tendencia actual se orienta a diseñar interfaces centradas en los usuarios, rescatando conceptos de accesibilidad [12] y [19].
 - Diseño de la base de datos. Consistirá en diseñar la base de datos y sus posibles relaciones con otras fuentes de datos.
 - Diseño del prototipo. Consistirá en elaborar el plan del prototipo, fijadas las restricciones con el usuario, se deben disponer al menos de algunas funcionalidades. Se refiere al análisis detallando cómo se trabajará, los módulos a implementar y las funciones a usar.
 - Documentación. Se elaborarán documentaciones de soporte al análisis y diseño de las soluciones propuestas.
- Desarrollo del sistema información.
- Selección de herramientas. Para la construcción del sistema se estudiarán y seleccionarán las herramientas más apropiadas. Se enfatizará en el paradigma FLOSS (Free/ Libre Open Source Software) [15] y [24]; y en tecnologías estándares como la plataforma Java.
 - Desarrollo. A partir de la experiencia desarrollada, fundamentada en el desarrollo de prototipos incrementales o evolutivos y la experimentación con otros ciclos de vida [2], [14] y [22] se procederá a la programación del sistema informático.
 - Integración de la calidad en el proceso de desarrollo. Existen diversos Modelos de Proceso Software y Normas de Calidad, a través de los cuales se miden o certifican los procesos de desarrollo (ISO/IEC 15504-2, ISO 90003, CMMI) y los productos [4], [6] y [7]. Además, la accesibilidad son algunos de los factores relevantes en la calidad de software, por lo cual se considerará en la construcción del sistema informático. A modo de ejemplo se mencionan algunas herramientas para medir la accesibilidad Lynx Browser [10], TAW [23], TextAloud [25], Web Developer Extensión [26].
 - Documentación. Se elaborarán documentaciones de soporte al desarrollo de la solución propuesta.
- Implementación y evaluación del sistema informático.
- Migración de los datos. Si existieran datos digitales, se prevé la migración de los datos desde antiguas fuentes de datos, como a priori de la implementación del sistema propuesto.
 - Implementación. Como todo proceso de desarrollo de producto informático, se realizará la implementación, previendo su posterior evaluación con el propósito de generar información de realimentación. Asimismo, la presentación de versiones, constituye un medio de obtener datos para refinar el sistema, de modo que al final del proyecto el resultado cubra los requerimientos.

- Evaluación del sistema informático con miras a realimentación. Se diseñarán y aplicarán instrumentos orientados a valorar los productos e implantaciones realizadas. Se construirán aquellos más adecuados al tipo de información a reunir, en relación con los objetivos, recursos y población investigados.
- Resguardo de la información. Los datos almacenados y los códigos desarrollados se exportarán regularmente con el propósito de asegurar la información de la institución.
- Documentación. Se elaborarán documentaciones de soporte a la implementación de las soluciones propuestas.
- Actualización y mantenimiento del sistema. La actualización y mantenimiento, tiene razón considerando modificaciones por: i) nuevos requerimientos o cambios en la administración de la información, ii) fallas detectadas por el uso cotidiano, iii) aplicación de innovadoras tecnologías y herramientas que surgieran en el mercado.
- Capacitación. Definido el entorno y con el objeto de propiciar el empleo eficiente del sistema informático en los destinatarios y a fin de asegurar su implantación, se planificarán instancias de capacitación ad-hoc.

3 Resultados

Desarrollo del prototipo: La principal característica que se pretende brindar con esta aplicación es la accesibilidad que brinda la Web. Se utilizó modelado UML (Unified Markup Language) para esquematizar las diferentes funciones que debería brindar el sistema en cuestión. Se optó por desarrollar el sistema aplicando la metodología de Prototipos, ya que la misma facilita un modelo a programador, hay partes del sistema que no están del todo claras y generalmente el Usuario “destinatario” no especifica todos los requerimientos al inicio del ciclo de vida del producto tecnológico.

El diseño de los prototipos se llevó a cabo, mediante modelado UML, se utilizaron: i) Casos de usos: para representar a los actores del sistema y obtener una visión general de las principales funcionalidades. ii) Conversaciones de los casos de uso para comprender que debería hacer cada uno, y iii) Diagramas de secuencia para identificar los diferentes flujos de información necesarios.

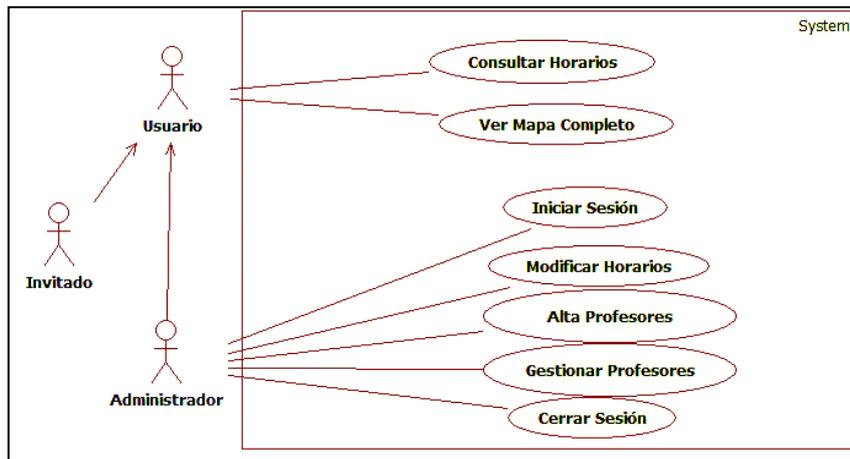


Figura 1. Diagrama de caso de uso General.

Se graficaron los principales diagramas a fin de representar la complejidad que representa el problema. En la Figura 1 se muestra el diagrama de caso de uso trazo grueso del sistema. La Figura 2 ilustra un caso de uso de trazo “fino”, detallándose las funciones del actor “Administrador” y “Gestionar Profesores”. En la Tabla 1 se representa la conversación correspondiente al caso de uso con su Curso Normal y Alternativo. La Figura 3 presenta el Diagrama de Secuencia correspondiente al Curso normal de la conversación. En la Tabla 2 se describe la conversación con la modificación de los horarios, llevada a cabo por el actor “Administrado” con su correspondiente diagrama de secuencia de curso normal según se detalla en la Figura 4.

En la Tabla 3 se describe la conversación correspondiente con las altas de profesores, llevada a cabo por el actor “Administrado” con su correspondiente diagrama de secuencia de curso normal en la Figura 5.

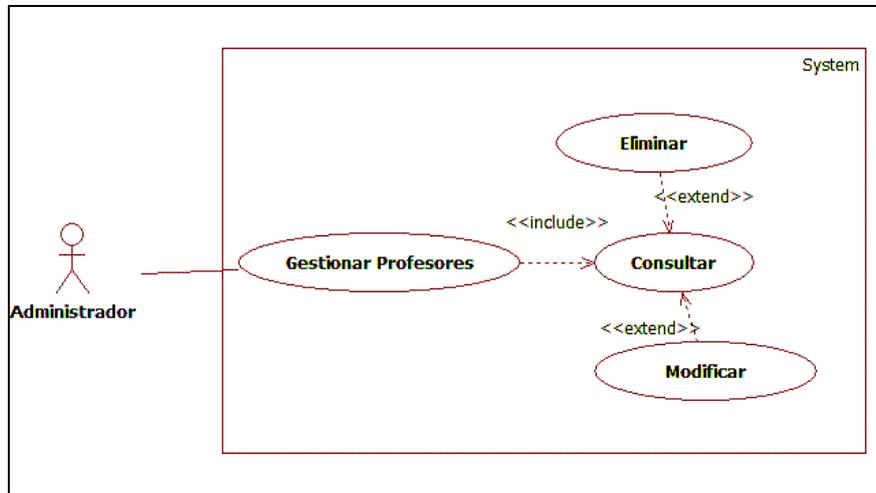


Figura 2. Diagrama de caso de uso Gestionar Profesores.

Caso de uso: “Gestionar Profesores”.

Actor: Administrador.

Descripción: Permite gestionar los profesores del sistema mostrando la lista completa. (Baja y Modificación de los mismos).

Tabla 1. Diagrama de conversación “Gestionar Profesores”.

Acción	Curso Normal	Curso Alternativo
1. Sistema - muestra la lista completa de profesores cargados en el sistema.		
2. Administrador – Selecciona una opción.	2.1.1 Opción - Modificar datos del profesor.	2.2.1 Opción - Baja del profesor en el sistema.
	2.1.2 Sistema – Consulta datos del profesor seleccionado y Muestra pantalla de modificación.	2.2.2 Sistema – Da de baja el profesor seleccionado e informa mediante un mensaje en pantalla.
	2.1.3 Administrador – Ingresa datos Modificados.	2.2.3 Vuelve a la pantalla inicial. Fin Caso de Uso.
	2.1.4 Administrador – Confirma.	
	2.1.5 Sistema – Guarda datos y muestra mensaje en pantalla.	
	2.1.6. Vuelve a la pantalla inicial. Fin Caso de Uso.	

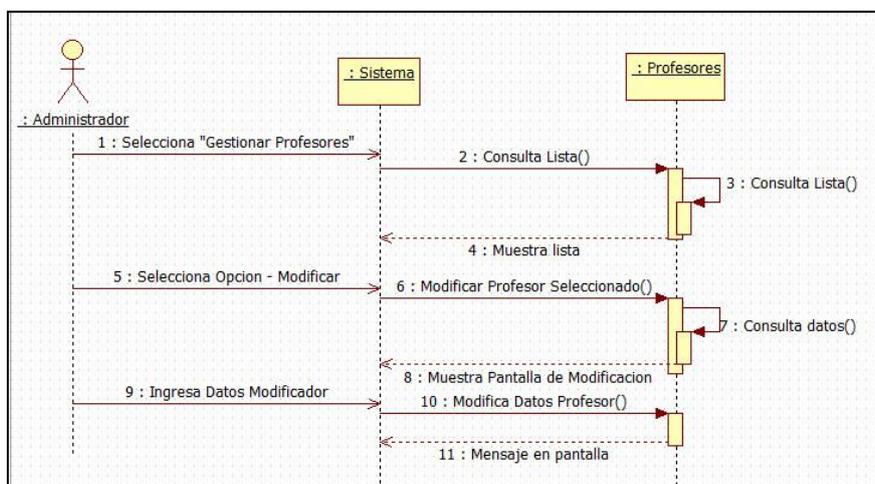


Figura 3. Diagrama de secuencia de “Gestionar Profesores – Curso Normal”.

Caso de uso: “Modificar Horarios”.

Actor: Administrador.

Descripción: Permite modificar los horarios asignados a las distintas aulas del sistema.

Tabla 2. Diagrama de conversación “Modificar Horarios”.

Acción	Curso Normal	Curso Alternativo
1. Administrador – Selecciona “modificar Horarios”		
2. Sistema – Consulta lista de aulas cargadas y las muestra.		
3. Administrador – Selecciona Aula a modificar.		
4. Sistema - Consulta datos del aula seleccionada.		
5. Sistema – Muestra pantalla de modificación junto con los datos del aula.		
6. Administrador – Selecciona “Hora Inicio” – “Hora Fin” – “Día” – “Carrera”.		
7. Sistema – Verifica Campos de datos.	7.1 Opción – Datos Correctos	7.2 Opción – Datos Incorrectos
	7.1.1 Muestra siguiente pantalla de modificación	7.2.1 “Mensaje de Error”, Vuelve al paso 4.
8. Administrador – Selecciona “Materia” – “Profesor” – “Cantidad de Alumnos”		
9. Sistema – Verifica Campos de datos.	9. Opción – Datos Correctos	9.2 Opción – Datos Incorrectos
	9.1.1 Muestra siguiente pantalla de modificación.	9.2.1 “Mensaje de Error”, Vuelve al paso 7
10. Sistema – Muestra datos completados hasta el momento.		
11. Administrador – Selecciona opción.	11.1 Opción – Dar Alta	11.2 Opción – Cancelar TODO
	11.1.1 Sistema – Da el alta efectiva del horario en el aula correspondiente.	11.2.1 Fin Caso de Uso.
12. Sistema Muestra pantalla inicial de modificación.		
13. Fin Caso de Uso		

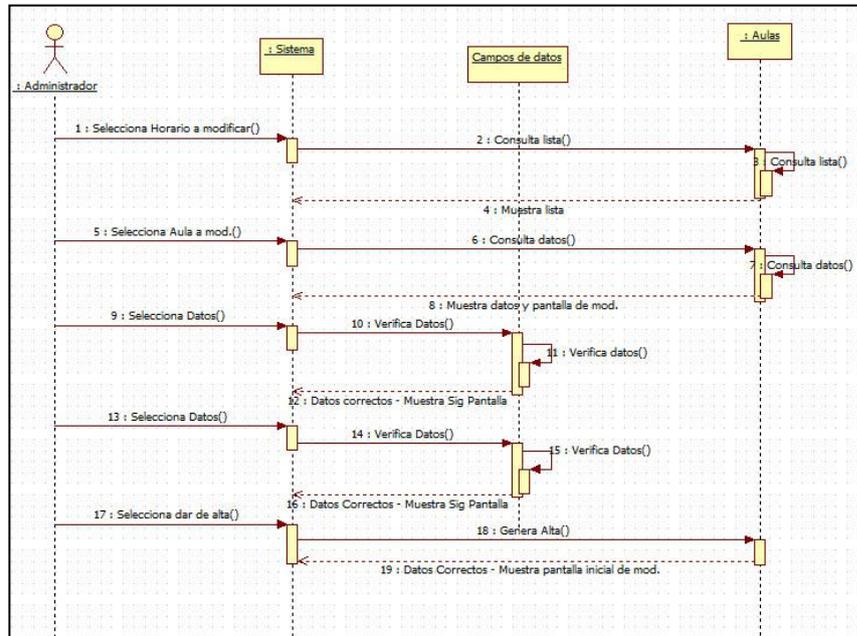


Figura 4. Diagrama de secuencia “Modificar Horarios”.

Caso de uso: “Alta Profesores”.

Actor: Administrador.

Descripción: Permite generar las altas de los profesores en el sistema.

Tabla 3. Diagrama de conversación “Alta de Profesores”.

Acción	Curso Normal	Curso Alternativo
1. Administrador – Selecciona “Alta Profesores”		
2. Sistema – Muestra pantalla de altas		
3. Administrador – Ingresa los datos		
4. Sistema – Verifica Datos ingresados y DNI.	4.1 Opcion – Datos Correctos	4.2 Opción – Datos Incorrectos
	4.1.1 Sistema – Da de alta los datos.	4.2.1 Sistema - Limpia Formulario.
	4.1.2 Sistema – Muestra mensaje “Alta de profesor Correcta”	
	4.1.3 Sistema – Limpia formulario.	
6. Fin Caso de Uso		

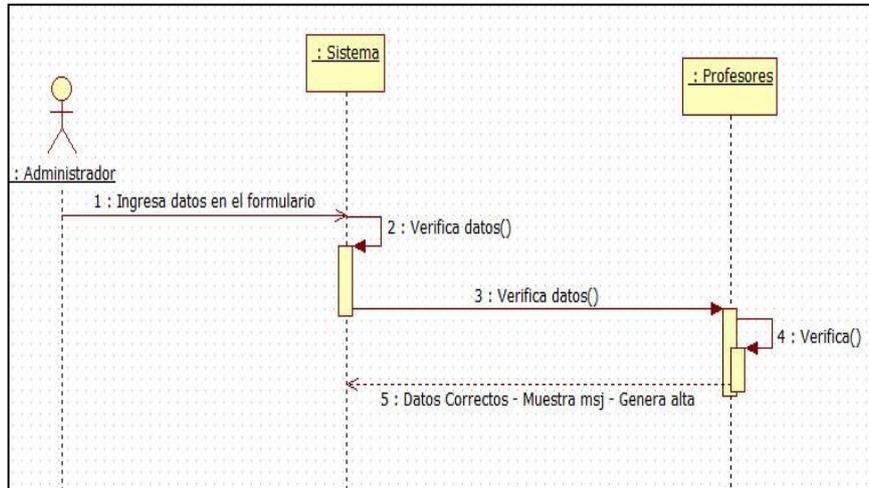


Figura 5. Diagrama de secuencia “Alta Profesores”.

Para el desarrollo del sistema, se ideó un modelo de datos (Figura 6), en la que posible almacenar toda la información necesaria para cumplir con los requerimientos establecidos. En las Figuras 7 y 8, se muestran algunas de las interfaces disponibles.

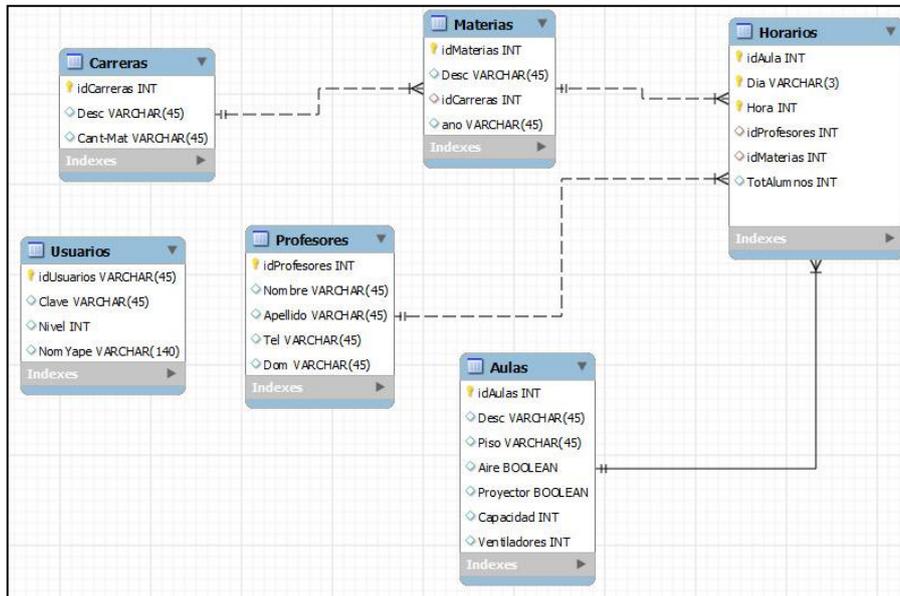


Figura 6. Estructura de la Base de Datos y sus relaciones



Figura 7. Algunas interfaces disponibles para el administrador



Figura 8. Algunas interfaces disponibles para consultas

4 Conclusión

En este trabajo se describieron los avances/resultados preliminares logrados en las primeras etapas del ciclo de vida propuesto en la sección metodología. Se considera de relevancia abordar una minuciosa especificación de requerimientos ya que este análisis constituye una explicitación del conocimiento elicitado en la organización adoptante del sistema informático.

Considerando la complejidad de la gestión de espacios físicos en instituciones de Educación Superior Públicas, especialmente aquellas orientadas a la contención de numerosas carreras con diversos perfiles profesionales se esperan que la implementación de la solución informática permita mejorar el proceso de administración, asignación y presentación de horarios de los espacios físicos. Su adopción permitiría apoyar la toma de decisiones vinculadas a infraestructura edilicias y brindar una diversidad de consultas a los distintos perfiles de usuarios sin restricciones espacio-temporales

Referencias

1. Alomoto Cuvi, L.; Morales Hernández, E.: Sistema de gestión de horarios en las aulas de la Facultad de Ingeniería Informáticos y de computación. Proyecto de tesis. Escuela Politécnica Nacional. Facultad de Ingeniería en Sistemas. En: <http://bibdigital.epn.edu.ec/handle/15000/925>.
2. Boehm, B.: A Spiral Model of Software Development and Enhancement. IEEE. 61-72pp. 1998.
3. Cancelo, F.; Cababie, N; Barrera, G.; López De Luise, D.: Un nuevo enfoque para asignación óptima de múltiples recursos. Universidad de Palermo, ITLab, AIGROUP.
4. Capability Maturity Model.: Integration (CMMISM), Version 1.1. CMMISM for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing. (CMMI-SE/SW/PPD/SS, V1.1). Software Engineering Institute, Carnegie Mellon University. 2002.
5. Gil Martínez, J., Abarca, Hernández Sáez, A., Zubizarreta Ugalde, J.: Modelo de Gestión y Seguridad en Aulas Informáticas. Escuela Politécnica Superior de la Universidad de Alicante. En: <http://www.dtic.ua.es/grupoM/recursos/articulos/JDARE-05-E.pdf>.
6. IEEE. 2001: Guide to the Software Engineering Body of Knowledge. SWEEBOOK. 2001. En: <http://es.scribd.com/doc/37120492/IEEE-SWEBOK-Software-Engineering>.
7. ISO/IEC 9126-1 2001. Software engineering — Product quality — Part 1: Quality model.
8. JSchoolManager.: En: <http://aulamagica.wordpress.com/2008/02/19/software-de-gestion-escolar-y-libre>.
9. Lores, J.: La interacción persona-ordenador. Ed. AIPO. ISBN: 84-607-2255-4.
10. Lynx Browser. En <http://invisible-island.net/lynx/>.
11. Mariño, S.; Godoy, V.: Sistemas y TIC: Técnicas y herramientas. PI f013-11 Acreditado por la SGCyT. UNNE. Resol. 142/12.
12. Navarro, I.; Fonseca, D.: Accesibilidad web en entornos culturales. Anuales de la 8ª Conferencia Iberoamericana en Sistemas, Cibernética e Informática - CИСCI 2009. En <http://www.iiis.org/CDs2009/CD2009CSC/CISC12009/PapersPdf/C205OK.pdf>.
13. MRBS.: En: <http://mrbs.sourceforge.net>.
14. Pressmann R.: Ingeniería de Software: Un Enfoque Práctico. Ed. Pearson Education, S.A., Madrid. Edition 7ª. 2007.

15. Quigley E.; Gargenta M.: PHP y MySql practico para diseñadores y programadores web. Ed. Anaya Multimedia. 2008.
16. Rembo.: En: <http://www.rembo.com>
17. Rodríguez, J.; Galio, G.: Desarrollo e implementación de un Sistema de Gestión de Asignación de Aulas y Laboratorios. Facultad de Ingeniería en Electricidad y Computación. Licenciatura en sistemas de información. Escuela Superior Politecnica del Litoral. En: <http://www.dspace.espol.edu.ec/bitstream/123456789/9103>.
18. Rumbaugh, J.; Booch, G.; Jacobson, I.: El Lenguaje Unificado de Modelado. Manual de Referencia. Ed. Pearson Educacion. 2007.
19. Segovia, C.: Accesibilidad e Internet... para que todas las personas, con distintas capacidades o recursos, puedan acceder a Internet. En: http://www.archena.es/files/accesibilidad_e_internet.pdf.
20. Symantec Ghost Solution Suite. En: <http://www.symantec.com/es/mx/ghost-solution-suite>.
21. Softaula.: En: <http://softaula.net/>.
22. Sommerville, I.: Ingeniería del Software. Ed. Prentice Hall. 2005.
23. TAW. Test de Accesibilidad Web "TAW". En: <http://www.tawdis.net>.
24. Tramullas Saz J.; Garrido Picazo, P.: Software Libre para Servicios de Información Digital. Ed. Prentice-Hall. 2006.
25. TextAloud. En: http://www.nextup.com/files/manualtrans/Spanish_TextAloudManual.html.
26. Web Developer Extensión. En: <https://addons.mozilla.org/es-ES/firefox/addon/60/>.