

## **Estudio de la Metodología Midas y la Plataforma Rails para el Desarrollo de un Sistema Web de Control de Proyectos**

**A. García, C. Hinojosa y R. Reyes**

*Departamento de Ciencias de la Computación, Escuela Politécnica del Ejército, Sangolquí, Ecuador*  
[andres.garcia@abcos.com.ec](mailto:andres.garcia@abcos.com.ec), {[chinojosa](mailto:chinojosa@espe.edu.ec), [rreyes](mailto:rreyes@espe.edu.ec)}@espe.edu.ec

**RESUMEN:** El estudio fue desarrollado utilizando la metodología Midas y la plataforma Rails para la creación de un sistema Web de control de proyectos. El análisis de compatibilidad entre la metodología y la plataforma se realizó comparando las iteraciones de Midas con el Modelo Vista Controlador que propone Rails. La elección de estas dos herramientas fue gracias a que Rails permite crear sistemas robustos, confiables y seguros, que pueden ser mantenibles y administrables en corto tiempo. Por otro lado, gracias a la Metodología Midas se pudo realizar de una manera eficiente el análisis, diseño y construcción del sistema de Control de Proyectos, los lineamientos dados por la metodología facilitaron el desarrollo del trabajo y se pudo lograr un proceso de desarrollo adecuado y el resultado fue un sistema de calidad, cumpliendo con las necesidades establecidas por los usuarios. La elección de las tecnologías a utilizar en el desarrollo del presente sistema Web contribuyó al éxito del mismo, se utilizaron herramientas de código abierto: plataforma Rails, lenguaje de programación Ruby, servidor Web Apache y base de datos MySQL. No solo se creó un sistema, sino que se mostró las ventajas de utilizar Software Libre en la creación de Sistemas de Información, obteniendo así productos de calidad con mayores funcionalidades, servicios y un ágil desarrollo.

**Palabras clave:** Modelo vista controlador, desarrollo Web, metodología Midas, Rails

**ABSTRACT:** The study was conducted using the methodology Midas and the Rails platform to create a Web project control. The analysis of compatibility between the methodology and the platform was made by comparing the iterations of Midas with Model View Controller proposed Rails. The choice of these two tools was thanks to the Rails to create robust, reliable and secure, which can be maintained and managed in a short time. On the other hand, thanks to the Methodology Midas could efficiently perform the analysis, design and construction of the Project Control system, the limits given by the methodology facilitated the development of work and to reach an adequate development process and the result was a quality system complying with the requirements set by users. The choice of technologies used in the development of this Web system contributed to its success, we used open source tools: platform Rails, Ruby programming language, Apache Web server and MySQL database. Not only created a system, but showed the benefits of using Free Software in the creation of Information Systems, thus obtaining quality products with more features, services, and agile development.

**Keywords:** Model View Controller, Web development, Midas Methodology, Rails Platform

## **1. INTRODUCCIÓN**

El acelerado crecimiento de la tecnología y el surgimiento de nuevos lenguajes de programación, ya sean gratuitos o propietarios, han generado una competitividad considerable entre las plataformas desarrolladas. Por esta razón, “*Ruby on Rails*” presenta como su principal ventaja la sencillez y productividad al momento de desarrollar la programación orientada a objetos [2], permitiendo programar de una manera más intuitiva, a diferencia de sus principales competidores como lo son PHP y Java.

Midas es una metodología para aplicaciones orientadas a la Web que propone la utilización de modelos mediante un proceso iterativo e incremental, dichos procesos permiten un desarrollo de sistemas ahorrando tiempo, ya que utiliza prácticas extraídas de metodologías ágiles como XP (*Xtreme Programming*) [6]. La principal ventaja de la metodología Midas, está en las diferentes iteraciones tanto en las Especificaciones de Requerimientos de Software como en el desarrollo de sistemas, las cuales se han definido para satisfacer las necesidades de los clientes como de los desarrolladores.

La combinación de estas dos tecnologías como lo son Midas y Ruby on Rails representan un reto al momento de desarrollar un sistema Web, ya que se deben aprovechar todas las ventajas que proveen cada una [1].

Frente a este escenario, como contribución, el presente artículo se basa en la presentación de la experiencia obtenida en el desarrollo de una aplicación Web, utilizando la metodología Midas y la plataforma Rails. Para cumplir con los objetivos propuestos, inicialmente se realizó una investigación documental bibliográfica, de donde se obtuvieron los lineamientos, conceptos y procedimientos que permitieron realizar el estudio comparativo y el desarrollo del producto software.

El resto del artículo ha sido organizado como sigue: La sección 2 describe los métodos de la investigación documental y la aplicación de la metodología Midas en la plataforma Rails. La sección 3 detalla la implementación del sistema de control de proyectos con la metodología y la plataforma propuesta. En la sección 4 se muestra la evaluación de los resultados de utilizar Midas y la plataforma Rails para el desarrollo de un sistema de control de proyectos. En la sección 5, se analizan los trabajos relacionados. Finalmente, en la sección 6, se presentan las conclusiones y líneas de trabajo futuro sobre la base de los resultados obtenidos.

## **2. MÉTODOS**

### ***2.1 Método de Investigación Documental***

La recopilación de la información se la inició mediante Internet, por ser temas relativamente nuevos había poca documentación. En los foros de discusión y al compartir conocimientos de desarrolladores experimentados, se pudo obtener información relevante.

Fue necesaria la aplicación de una investigación documental bibliográfica de fuentes primarias y secundarias internacionales [4], en donde la metodología Midas y la plataforma Rails han tenido éxito en organizaciones mundialmente reconocidas.

### ***2.2 Aplicación de la metodología Midas en Ruby on Rails***

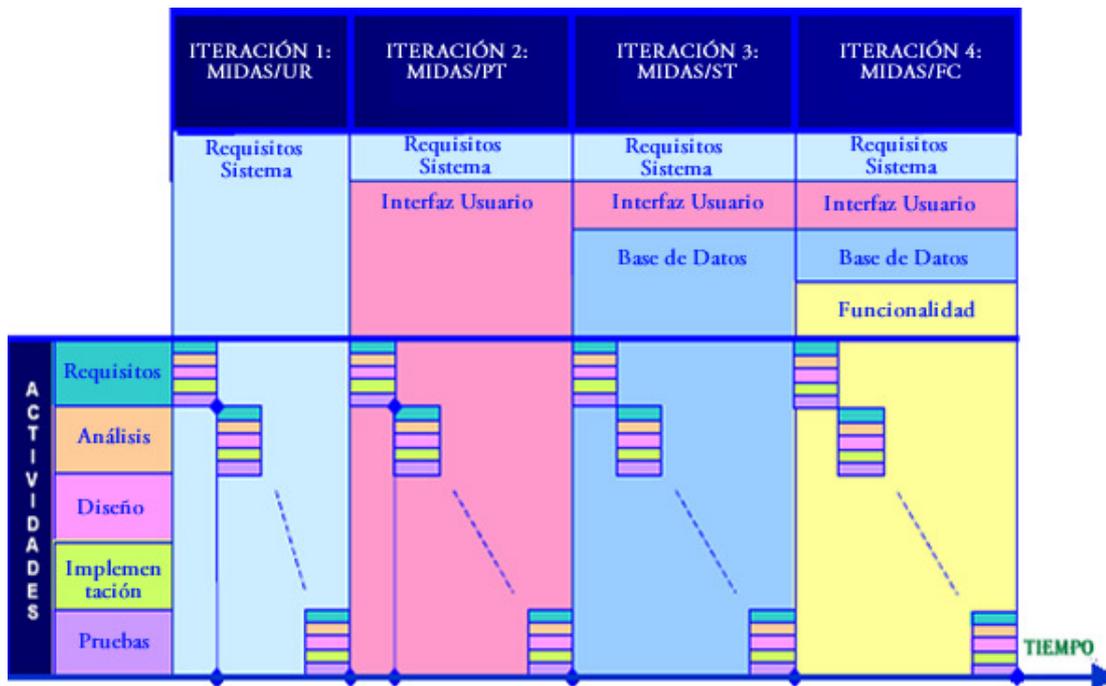
Posteriormente, se definieron las iteraciones de Midas de la siguiente manera:

Dentro de la Iteración 1 (ver Fig. 1) se plasmaron todos los modelos independientes de computación los cuales se definen con los requisitos del sistema y se especifican los casos de uso, esta iteración no interactúa con Ruby on Rails ya que se realizan los diagramas del análisis de requerimientos del sistema [3].

Posteriormente, en la Iteración 2 (ver Fig. 1) se desarrolló el modelo conceptual de datos y se realizó el primer prototipo del sistema. La utilización de Rails en esta iteración es importante porque se diseña las

interfaces de los usuarios y facilita el realizar un prototipo real del sistema gracias a su generación automática de código.

En la Iteración 3 (ver Fig. 1) se definió el modelo conceptual de datos, se implementaron e integraron los diseños lógicos de los datos. Rails facilita la conexión e implementación de la base de datos, ya que se definen las tablas mediante el ingreso de código, y el usuario puede migrar las tablas dentro de la base de datos.



**Figura 1:** Ciclo de vida del desarrollo del SIW con Midas [6] [8]

Finalmente, en la Iteración 4 (ver Fig. 1) se realizó el modelo de descripción del lenguaje del servicio Web *Web Services Description Language* (WSDL) que define la funcionalidad del sistema basado en el modelo de casos de uso y en el de composición de servicios.

La plataforma Rails utiliza el patrón de desarrollo Modelo Vista Controlador el cual permitió definir y desarrollar cada una de las iteraciones de Midas.

**Modelo.**- Representa los datos y las relaciones entre ellos (llamado también lógica de negocios). Consiste en las clases que representan a las tablas de la base de datos. Son gestionadas por ActiveRecord. Por lo general, lo único que tiene que hacer el programador es heredar de la clase ActiveRecord::Base, y el programa averiguará automáticamente qué tabla usar y qué columnas tiene. Aquí se aplicó la Iteración 3 de Midas [10].

**Vista.**- La interfaz de usuario. Representa cómo se muestran los datos de las clases del Controlador. Con frecuencia en las aplicaciones Web la vista consiste en una cantidad mínima de código incluido en HTML. Aquí se aplicó la Iteración 2 de Midas [10].

**Controlador.**- Quien recibe los eventos solicitados a través de la vista, los empuja al modelo, y genera/refresca a la vista. Describe a la interacción del usuario e invocan a la lógica de la aplicación, que a su vez manipula los datos de las clases del Modelo y muestra los resultados usando las Vistas. En las aplicaciones Web basadas en MVC, los métodos del controlador son invocados por el usuario usando el navegador Web. Aquí se aplicó la Iteración 4 de Midas [10].

### **3. IMPLEMENTACION**

En esta sección se describe la metodología Midas con la plataforma Rails en el desarrollo de un sistema Web de control de proyectos, para lo cual se dividió según las iteraciones de Midas de manera secuencial:

#### ***3.1 Iteración 1: Requisitos del sistema***

Para proceder a la implementación, en primer lugar se tomaron los requisitos del sistema mediante entrevistas a los diferentes usuarios y actores del sistema [5]. Las técnicas aplicadas fueron principalmente grabar las conversaciones, documentarlas y revisarlas paulatinamente con los clientes. Esta es la única iteración en la que la plataforma Rails no tiene ninguna participación.

#### ***3.2 Iteración 2: Interfaz de usuario***

Una vez documentados y aprobados los requisitos del sistema, se procede a realizar las interfaces de los usuarios, para lo cual Rails permitió mediante la Vista representar con código HTML y Ruby cada una de las pantallas del sistema [11]. La mezcla de código entre HTML y Ruby fue de gran ayuda, ya que permitió posicionar los elementos de una forma fácil e intuitiva.

Para un mayor control de los archivos, Rails creó la carpeta Views, en la cual se almacenaron todas las interfaces de usuario, en archivos de extensiones .html.erb.

#### ***3.3 Iteración 3: Base de datos***

Se definió, realizó, implementó e integró los modelos conceptual y lógico de la base de datos, con ayuda de los requisitos del sistema [9]. Rails mediante el Modelo y la migración de la base de datos permitió la implementación del modelo lógico de los datos, dentro de los cuales se validaron los datos para asegurar la integridad y veracidad de los mismos.

Para un mayor control de los archivos, Rails creó la carpeta Models y Database Migrations [15], en las cuales se almacenaron todos los modelos y creaciones de la base de datos, en archivos de extensiones .rb.

#### ***3.4 Iteración 4: Funcionalidad***

Por último, se desarrolló el sistema tomando en cuenta los casos de uso. En esta iteración se desarrollaron los Controladores de Rails, tomando en cuenta las interfaces de los usuarios y la base de datos. Se integró a todo el Modelo Vista Controlador que ofrece Rails [7] mediante las iteraciones de Midas.

Para un mayor control de los archivos, Rails creó la carpeta Controllers, en la cual se almacenaron todos los controladores del sistema, en archivos de extensiones .rb.

El sistema inicia con una pantalla de ingreso en la que se solicita el nombre de usuario y password asignados por el gerente de sistemas; se da la opción de ingresar a través de dos usuarios:

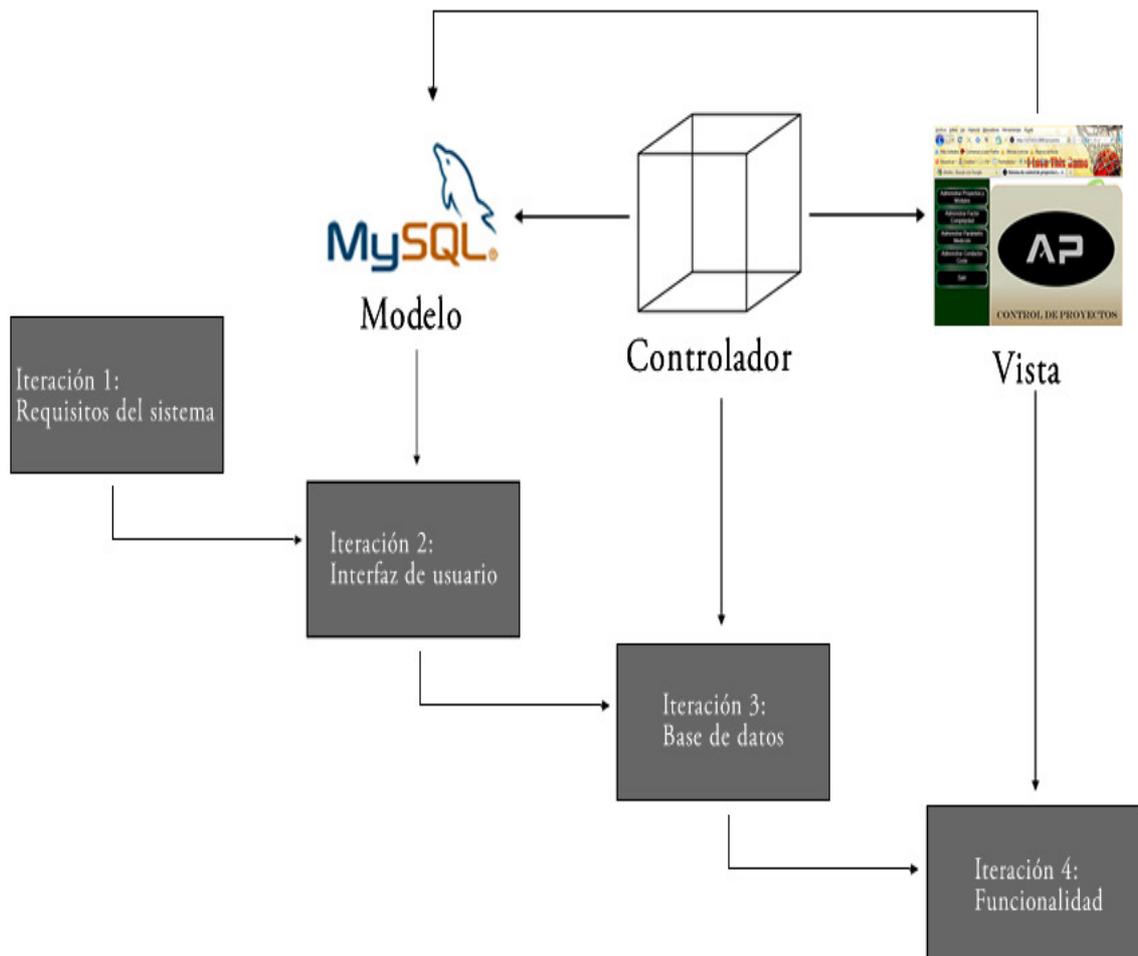
- i. Administrador de proyectos.- Con la opción de ingresar a administrar proyectos y módulos para crear nuevos proyectos, editar uno ya existente, eliminarlo y obtener reportes, consultar estimaciones y avances, además se puede ingresar los valores de conductores de coste. Se pueden crear nuevos módulos, editarlos, eliminarlos, establecer avances, los valores de factores de complejidad y parámetros de medición.

A través del cálculo de los factores de cada proyecto se tiene el factor de complejidad, parámetros de medición y conductor de coste que permiten realizar el cálculo del esfuerzo, tiempo, productividad y personal necesario para el desarrollo.

- ii. Gerente de sistemas.- Con la opción de ingresar a administración de proyectos y módulos para hacer consultas acerca del avance y estimaciones de los proyectos ingresados al sistema. Y, administrador de usuarios, para crear, editar o eliminar usuarios.

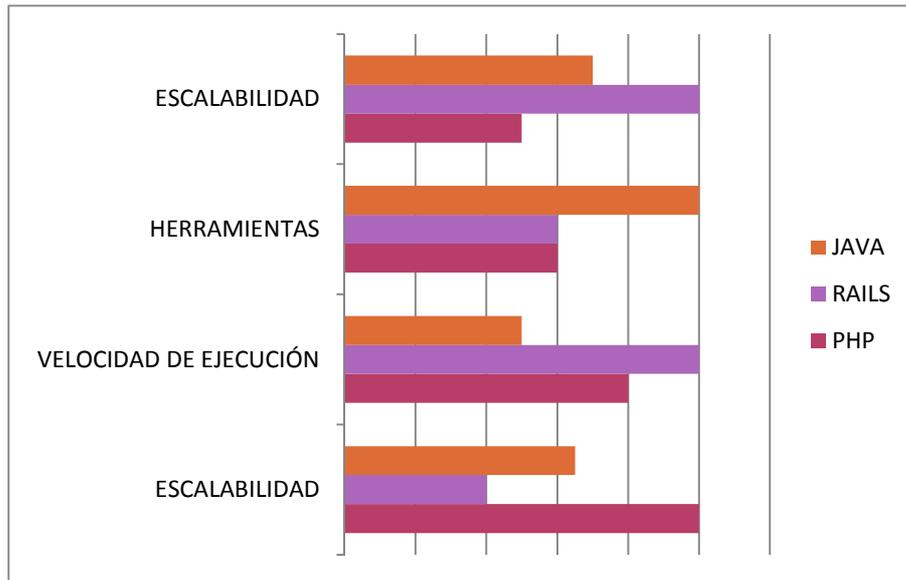
#### 4. EVALUACIÓN DE RESULTADOS

Como resultado se obtuvo que la metodología Midas y la plataforma Rails son completamente compatibles en su desarrollo, ya que las iteraciones de Midas y el patrón MVC que utiliza Rails se complementan entre sí como lo muestra la Fig. 2.



**Figura 2:** Comparación de la metodología Midas con la plataforma Rails

Además, la plataforma Rails con su lenguaje de programación Ruby frente a sus principales competidores como lo son PHP y Java tiene facilidad de mantenibilidad y mayor velocidad de ejecución, pero su debilidad lo demuestra en la escases de herramientas y escalabilidad como lo demuestra la Fig. 3.



**Figura 3:** Comparación entre Java vs Php vs Rails [14]

El sistema de control de proyectos permitió obtener resultados reales del esfuerzo, tiempo, productividad y el personal que necesitan para realizar proyectos específicos (ver Tabla 1). Permitted ser eficientes y eficaces en los recursos que destinan a los proyectos que se están ejecutando. Este software estableció parámetros a los sistemas, que por su naturaleza intangible es difícil valorar, permitiendo determinar costos y personal [12].

**TABLA 1:** Estimaciones de proyectos Web de la Empresa Adgesproject Cía. Ltda.

Proyecto	Punto De Función	Esfuerzo (Personas / Mes)	Tiempo (Meses)	Productividad (Lineas de Código (LDC) / Esfuerzo)	Personal (Personas)
Florícolas	0.0	0.00	0.00	0.00	0.00
Mapa1	140.97	2.00	3.25	465.11	0.61
secob	130.02	0.08	0.95	429.06	0.08

El sistema de control de proyectos permitió el control del avance de los proyectos (ver Tabla 2) para tomar decisiones sobre planes de contingencia dentro de la empresa y cumplir con los plazos acordados con el cliente evitando pérdidas económicas y problemas legales por el incumplimiento de contratos suscritos [13].

Estableció un proceso de seguimiento de las fases de desarrollo del sistema, verificando el tiempo requerido en cada una.

**TABLA 2:** Control de avance de un proyecto Web de la Empresa Adgesproyect Cía. Ltda.

cajeros					
Hito	Fecha Inicio	Fecha Entrega	Fecha Finalización	Estado	Observación
asasx	2009-10-06	2009-10-15	2009-10-30	Análisis	asd
asasx	2009-10-06	2009-10-15	2009-10-30	Análisis	asd
111	2009-09-29	2009-10-08	2009-10-23	Análisis	11

## 5. TRABAJOS RELACIONADOS

La plataforma Rails es relativamente nueva, han pasado 4 años desde su publicación, está en su etapa de despliegue por lo que son escasos los sistemas desarrollados con esta plataforma. En la investigación no se encontraron sistemas de control de proyecto desarrollados utilizando la metodología Midas en la plataforma Rails.

## 6. CONCLUSIONES Y TRABAJO FUTURO

La aplicación fue desarrollada en la plataforma Rails, lenguaje de programación Ruby y demás herramientas de código abierto para la implementación del sistema a nivel de programación, servidor Web y base de datos. En el lenguaje de programación Ruby aplica el concepto de “todo es objeto”, debido a que, incluso los tipos de datos los maneja como objetos independientes lo cual permite la reutilización del código, facilita su mantenimiento y agiliza el desarrollo del software. Midas se ha definido mediante un proceso iterativo e incremental, basado en las propuestas de Investigación en Acción, utilizando casos de estudio, los mismos que han servido para determinar los problemas y necesidades, desde un punto de vista metodológico e ingenieril, en el desarrollo de SIW. Pero, además de los casos de estudio, una vez especificada la metodología, Midas se ha aplicado a distintos casos de prueba, lo que ha permitido refinarla. Las iteraciones de Midas son compatibles con el patrón de desarrollo Modelo Vista Controlador que aplica Ruby on Rails, lo cual permite el trabajo óptimo entre estas dos tecnologías haciéndolas complementarias. Se recomienda el uso de NetBeans para el desarrollo del sistema, porque es un Entorno de Desarrollo Integrado de código abierto para diferentes lenguajes de programación entre ellos Ruby utilizando la plataforma Rails, además posee una interfaz gráfica que permite a los desarrolladores el ordenamiento del código y facilita la utilización de la herramienta. Se recomienda grabar las conversaciones con los usuarios al momento de obtener los requerimientos del sistema, de esa manera se facilita la comprensión y documentación de los mismos. Para la seguridad e integridad de los datos se recomienda a los usuarios poner contraseñas seguras y no dejarlas en blanco. Se recomienda realizar pruebas informales periódicamente con los usuarios para garantizar la calidad del producto. Se recomienda la aplicación de Midas utilizando la plataforma Rails ya que sus procesos tanto en las iteraciones como en el patrón MVC son compatibles en su desarrollo.

Como trabajo a futuro se creará un sistema de Inteligencia de Negocios (Business Intelligence BI) para el manejo de estadísticas de los proyectos, saber sus pérdidas y ganancias, además de la ayuda en la toma de decisiones para proyectos posteriores.

## AGRADECIMIENTOS

Agradecemos a la Escuela Politécnica del Ejército (ESPE), que a través de la facilidad de recursos económicos, técnicos y humanos, de parte del Departamento de Ciencias de la Computación han hecho realidad la consecución de este proyecto. Un cordial agradecimiento a la Ingeniera Cecilia Hinojosa, y al Ingeniero Rolando Reyes por haber compartido sus conocimientos desinteresadamente, siendo una guía y respaldo a lo largo de la elaboración del proyecto. A mis padres por su amor, comprensión y apoyo incondicional, que han fomentando valores y enseñanzas que nos han ayudado durante toda la carrera. A mis profesores, por sus valiosos consejos y por compartir sus amplios conocimientos y experiencia. Y a todas aquellas personas que de una u otra forma, colaboraron o participaron en la realización de este proyecto, hago extensivo mi más sincero agradecimiento.

## Referencias Bibliográficas

- [1]. D. Thomas, C. Fowler, A. Hunt; "Programming Ruby - The Pragmatic Programmer's Guide", 2da Edición, Octubre 2004.
- [2]. D. Thomas, D. Hansson; "Agile Web Development with Rails", 2005.
- [3]. C. Hibbs, B. Tale; "Ruby on Rails: Up and Running", 2006.
- [4]. P. Roger, "Ingeniería De Software Un Enfoque Práctico", Mc. Graw Hill, Madrid – España, 2002.
- [5]. S. Perdita, P. Rob, W. Addison; "Utilización de UML en ingeniería de software con objetos y componentes", Octubre 2007.
- [6]. P. Cáceres; "An approach for Navigation Model Construction from the Use Cases Model".
- [7]. O. Bini, "Practical JRuby on Rails Web 2.0 Projects: Bringing Ruby on Rails to the Java™ Platform", 2007.
- [8]. V. De Castro, J. Cavero, B. Vela, P. Cáceres; "Seminario de Lenguajes y Sistemas Informáticos", 22 de enero de 2003.
- [9]. OMG. Object Management Group, en: <http://www.omg.org/>.
- [10]. Desarrollo Orientado a Objetos con UML, en: <http://www.clikear.com/manuales/uml/>.
- [11]. Linux para todos, en: <http://www.linuxparatodos.net/portal/staticpages/index.php?page=servidor-Web/>.
- [12]. Somerville, Ian (2005). Ingeniería del software. Séptima edición. Editorial Pearson Educación.
- [13]. I., Booch, G. y Rumbaugh J. Addison Wesley; "El Proceso Unificado de Desarrollo de Software", 2001.
- [14]. E. Marcos Dykinson; "Investigación en Ingeniería del Software vs. Desarrollo Software", 2003.
- [15]. W3C Web Services Description Language (WSDL); "W3C Working Draft", marzo 2003, en <http://www.w3.org>.

	<b>E S P E</b> ESCUELA POLITÉCNICA DEL EJÉRCITO CAMINO A LA EXCELENCIA	<b>Informes: <a href="mailto:marketing@espe.edu.ec">marketing@espe.edu.ec</a> Marketing 3949400 Ext. 3001</b>
<b>UNIDAD DE GESTIÓN DE POSTGRADOS</b>		