**CONSTRUCCIÓN DE UNA APLICACIÓN WEB PARA LA ELABORACIÓN DE DIAGRAMAS DE CLASE, ORIENTADA BAJO EL MODELO DE SERVICIOS DE INTERNET CLOUD COMPUTING**

**Gabriel Leonardo Díaz Cárdenas**1, 2, **Marco Antonio Adarme Jaimes**1, 3

1 Programa de Ingeniería de Sistemas, Universidad Francisco de Paula Santander. Avenida Gran Colombia No. 12E-96B Colsag. San José de Cúcuta, Norte de Santander, Colombia.

2 Estudiante, gabrielleonardodc@ufps.edu.co.

3 Director del Proyecto de Grado, madarme@ufps.edu.co.

**RESUMEN**

El presente artículo expone el proceso de construcción de la herramienta CLASS Modeler; se trata de una aplicación web que permite diseñar diagramas de clase UML de manera online, así como la generación de código fuente Java a partir de ellos. Esta aplicación fue el resultado del proyecto de grado titulado: CONSTRUCCIÓN DE UNA APLICACIÓN WEB PARA LA ELABORACIÓN DE DIAGRAMAS DE CLASE, ORIENTADA BAJO EL MODELO DE SERVICIOS DE INTERNET CLOUD; cuyo enfoque gira en torno al desarrollo de una herramienta tecnológica que sirva de apoyo al proceso educativo de los estudiantes del Programa de Ingeniería de Sistemas de la Universidad Francisco de Paula Santander. En este caso el tema de investigación son las herramientas CASE, que constituyen un aspecto importante en el desarrollo de software y resultan de gran ayuda al momento de poner en práctica conceptos como abstracción y modelamiento de sistemas.

CLASS Modeler fue desarrollada usando lenguaje Java a nivel empresarial; el estándar J2EE constituyó el ambiente principal de desarrollo de la herramienta, incorporando tecnologías avanzadas en la construcción de aplicaciones con arquitectura basada en capas. Adicionalmente, se fundamenta en la especificación formal del lenguaje UML en su versión 2.4.1 definida por el Object Management Group [1], utilizando una implementación en lenguaje Java del metamodelo definido, lo cual permitió la representación del diagrama en forma de un modelo de objetos en dicho lenguaje. A lo largo del artículo se explica de manera muy general el proceso de desarrollo de la aplicación, abordando temas técnicos de arquitectura y tecnología utilizadas, así como la aplicación de un proceso híbrido entre dos metodologías de desarrollo de software muy conocidas, Programación Extrema XP y El Proceso Unificado UP.

**Palabras Clave:** Aplicación Web, Diagramas de Clase, UML, Herramientas CASE, Desarrollo de Software, Java.

1. **INTRODUCCIÓN**

En proyectos de investigación realizados anteriormente dentro de la Universidad Francisco de Paula Santander [2], se determinó que uno de los principales factores de deserción y mortalidad académica en el Programa de Ingeniería de Sistemas, es la falta de práctica y comprensión de conceptos como abstracción y modelamiento de sistemas, así como la falta de ambientes para realizar actividades de refuerzo y afianzamiento.

Considerando lo anterior, se hace notoria la necesidad de disponer de mecanismos que ayuden a reforzar los conceptos aprendidos en clase y así disminuir esta problemática que afecta el programa. Es por ello que se planteó la posibilidad de diseñar una nueva herramienta enfocada en las necesidades particulares del ambiente académico.

Si bien se conocía de antemano la existencia software especializado en el diseño de diagramas UML, se determinó también, en base a observaciones y manifestaciones de estudiantes y docentes, que la tecnología existente no era suficiente en algunos casos. Debido que la mayoría es software que se instala de manera nativa, varias de estas aplicaciones requieren de configuración y actualización manual, lo cual resulta en un esfuerzo para adecuar ambientes de trabajo para grupos de estudiantes; por no mencionar la complejidad implícita que representa el uso de estas herramientas por parte de estudiantes que están iniciando sus estudios universitarios.

Se planteó el desarrollo de una aplicación RIA[[1]](#footnote-1), que realizara las funciones mínimas de una herramienta CASE para el diseño de diagramas de clase UML, y que proporcionara características propias de una aplicación web como, portabilidad, facilidad de acceso y uso.

CLASS Modeler, nombre que se le dio a la aplicación, se clasifica dentro del modelo de servicios SaaS establecido por Cloud Computing[[2]](#footnote-2); el cual define el software como un servicio que puede ser accedido bajo demanda desde cualquier lugar y en cualquier momento.

La herramienta permite únicamente el diseño de diagramas de clase UML y genera código fuente en lenguaje Java; esto debido a que el lenguaje ha sido adoptado por el Programa de Ingeniería de Sistemas para la enseñanza de programación desde el inicio de la carrera.

1. **DESARROLLO DEL PROYECTO**
   1. **METODOLOGIA**

En términos de metodología, se optó por realizar un hibrido entre dos metodologías XP y UP. Esto permitió enfocar el desarrollo de manera iterativa y adaptable a los cambios, sin dejar de lado los lineamientos un proceso robusto y bien documentado.

Dentro de la metodología se definieron los siguientes aspectos:

* Historias de Usuario
* Roles del Proyecto
* Actores del Sistema
* Iteraciones
* Arquitectura del Sistema
* Tecnologías

Las historias de usuario contenían la descripción de las funcionalidades a implementar, las cuales a su vez, fueron producto de un análisis de las herramientas CASE más utilizadas por estudiantes y docentes del programa académico.

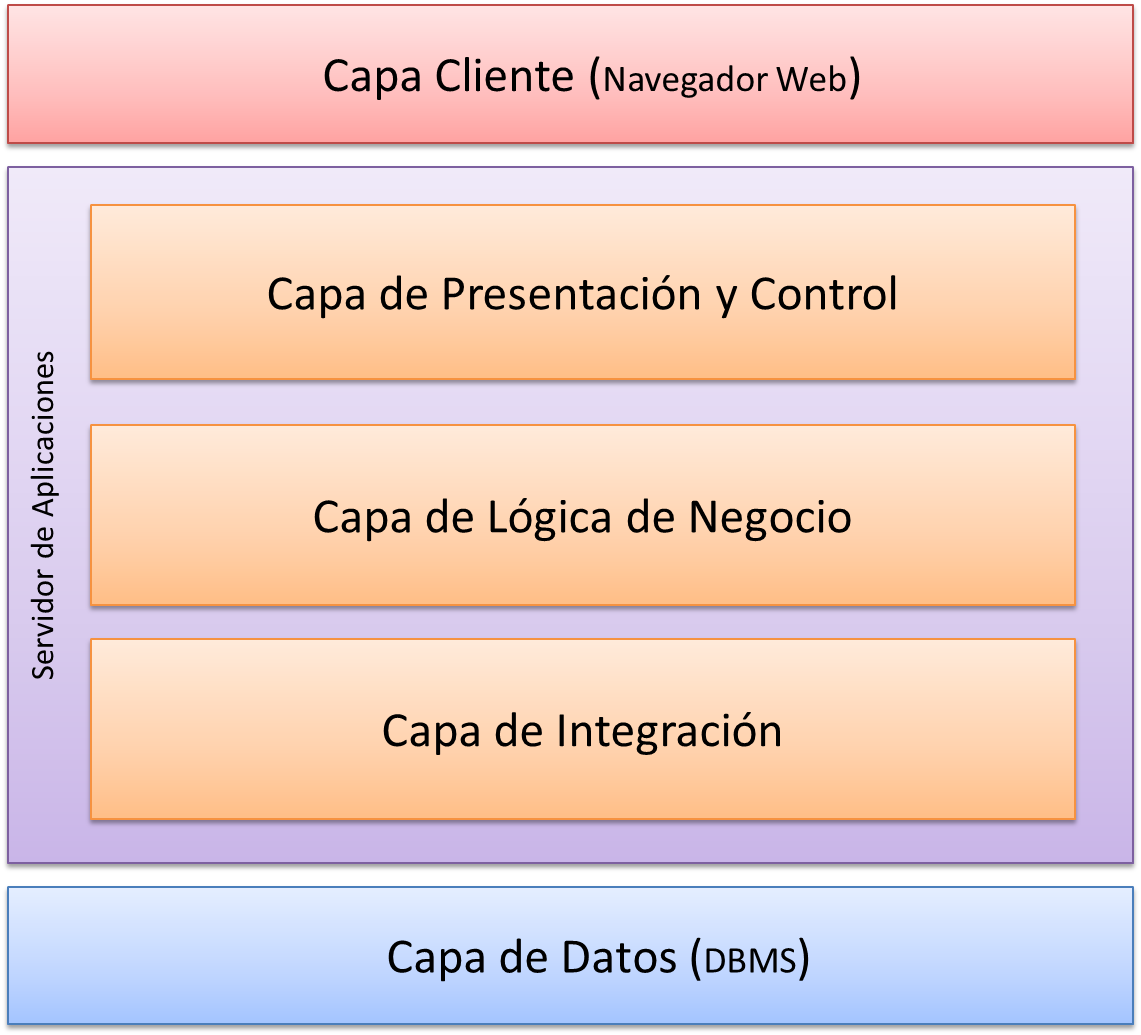
Estas historias se clasificaron en base a su prioridad y complejidad, y se determinaron las iteraciones del proyecto, cada iteración entregaba una versión del sistema en construcción.

Cada iteración define una serie de actividades, dentro de las cuales se destacan el diseño de las pruebas de unidad realizadas a los servicios EJB implementados para manejar la lógica de negocio de la aplicación. En la sección

* 1. **ARQUITECTURA**

La arquitectura seleccionada para el proyecto fue basada en capas, en donde cada capa tiene una labor específica y se comunica con las demás mediante interfaces bien definidas.

La siguiente figura muestra la distribución de las capas.



Hablar de patrones

MVC

DAO

* 1. **TECNOLOGIAS UTILIZADAS**

Las tecnologías utilizadas se clasifican dentro de cada capa de la arquitectura del sistema, es así como cada tecnología cumple con su labor específica dentro de su ambiente de ejecución.

La siguiente tabla menciona las tecnologías utilizadas dentro de cada capa.

|  |  |
| --- | --- |
| **Capa** | **Tecnología** |
| Cliente | jQuery / Javascript |
| CSS |
| HTML |
| XML |
| mxGraph |
| Presentación y Control | JSF |
| PrimeFaces |
| Facelets |
| XHTML |
| Servlets |
| XML Parser |
| Lógica de Negocio | Enterprise JavaBeans |
| UML2/EMF |
| JavaMail |
| Integración | JPA |
| Datos | MySQL |

* 1. **DISEÑO DE DIAGRAMAS**

Hablar de mxGraph

Hablar de SVG

Hablar de Javascript

* 1. **UML A CÓDIGO FUENTE**

Hablar de UML2

Hablar de StringTemplate

Hablar del Metamodelo

**PRUEBAS DE UNIDAD**

1. **RESULTADOS**

Se obtuvo una aplicación web que permite diseñar diagramas de clase UML y generar código fuente en lenguaje Java a partir del diagrama construido. De esta manera se obtiene un mecanismo adicional que sirve como apoyo al proceso educativo de los estudiantes, permitiendo practicar y reforzar conceptos de modelamiento y abstracción de sistemas.

1. **CONCLUSIONES**

Con el desarrollo de CLASS Modeler, se logró construir una herramienta de apoyo para el proceso de aprendizaje de los estudiantes del Programa de Ingeniería de Sistemas de la Universidad Francisco de Paula Santander; muy útil para la realización de prácticas sobre el modelado de diagramas de clase, promoviendo aspectos importantes como el trabajo en equipo y la colaboración.

CLASS Modeler proporciona beneficios que resultan atractivos a la hora de utilizar una herramienta de modelado UML, tales como la facilidad de acceso, inmediatez a la hora de diseñar diagramas y portabilidad de la información. CLASS Modeler solo necesita una conexión a internet y un navegador web para ser usado, y no requiere instalación de complementos como Flash o Shockwave.

CLASS Modeler se puede clasificar dentro del modelo de servicios SaaS, donde el soporte lógico y los datos de la aplicación se encuentran en un servidor de TI y los clientes acceden bajo demanda mediante un navegador web.

Un aspecto muy importante del proyecto fue la capacidad de convertir el metamodelo de clases de UML a código fuente en lenguaje Java. Esto representó un logro bastante importante debido a la complejidad implícita del proceso y la diferencia que existe entre ambos lenguajes. En algunos casos fue necesario extender el metamodelo para soportar conceptos propios de Java, como son los Tipos Genéricos, utilizados a la hora de crear colecciones de objetos.

**REFERENCIAS**

1. **OMG Object Management Group.** Unified Modeling Language Superstructure Definition. [En línea] 6 de Agosto de 2011. http://www.omg.org/spec/UML/2.4.1/Superstructure/PDF/. formal/2011-08-06.

2. **Amaya Torrado, Yegny Karina y Herrera Angarita, Lady Torcoroma.** *Identificación de las causas que generan problemas en el aprendizaje de fundamentos de programación de computadores en las facultades de ingenieira de sistemas de las universidades de la ciudad de Cúcuta.* Cúcuta N. de S. : Universidad Francisco de Paula Santander, Biblioteca Eduardo Cote Lamus, 2003. TIS 371.334A489i.

1. Rich Internet Applications, son aplicaciones web que tienen la mayoría de características de las aplicaciones de escritorio normales. [↑](#footnote-ref-1)
2. Cloud Computing, es un paradigma que permite ofrecer servicios de computación a través de internet. [↑](#footnote-ref-2)