

EXPERIENCIA DE DESARROLLO DE UNA APLICACIÓN WEB UTILIZANDO LA METODOLOGIA DE MODELADO RUP Y UML

LUIS ANIBAL MOSCOSO CALVOPIÑA

Departamento de Eléctrica y. Electrónica, Escuela Politécnica del Ejército Sede Latacunga, Marquéz de Maenza S/N Latacunga, Ecuador,
Email: luismoscoso2008@gmail.com

RESUMEN: En este artículo se presenta la experiencia del desarrollo de un sistema de información académico, utilizando la metodología de modelado RUP, y un diseño orientado a objetos. Se presentan las características principales y los beneficios de esta propuesta de desarrollo de software orientado a la web. El principal problema que han acarreado los sistemas web a lo largo de los años es la falta de enfoque, sobre el proceso que se debe manejar durante el desarrollo de éste. La metodología que se explicara a lo largo del presente artículo es RUP (RATIONAL UNIFIED PROCESS), esta metodología es uno de los procesos más generales de los existentes actualmente, ya que en realidad está pensado para adaptarse a cualquier tipo de proyecto.

ABSTRACT: This article presents the experience of developing an academic information system, using the RUP modeling methodology, and an object-oriented design. We present the main features and benefits of the proposed development of web-oriented software. The main problem that web systems have led over the years is the lack of focus on the process that must be handled during the development of this. The methodology will be explained throughout this article is RUP (Rational Unified Process); this methodology is one of the more general processes currently existing, because it actually is designed to fit any type of project.

I. INTRODUCCIÓN.

Internet es una red de redes a escala mundial de millones de computadoras. Al contrario de lo que se piensa comúnmente, Internet no es sinónimo de World Wide Web. Ésta es parte de aquella, siendo la World Wide Web uno de los muchos servicios ofertados en la red Internet.

Algunos de los servicios disponibles en Internet aparte de la Web son el acceso remoto a otras máquinas (telnet y ssh),

transferencia de ficheros (FTP), correo electrónico, grupos de discusión, y chat o conversaciones en línea (IRC o Jabber). El acceso a Internet se realiza tanto desde el hogar como desde el lugar de trabajo o de un cibercafé.

Para mejora el desarrollo de los sistemas orientados a la web los desarrolladores se ha enfocado sus esfuerzos en mayor proporción en el proceso, dejando a un lado las herramientas de desarrollo, lo cual genera un problema de entendimiento y comunicación con el usuario final. La solución para la problemática se ha ido construyendo con el pasar de los años, surgiendo estándares y metodologías de desarrollo que permiten que los sistemas vayan tomando más fuerza y cubran más campos dentro de la sociedad. Una metodología es aquella guía que se sigue a fin realizar las acciones propias de una investigación, identifica qué hacer y cómo actuar cuando se quiere obtener algún tipo de investigación.

El objetivo de este artículo es mostrar la experiencia que se tuvo al desarrollar un sistema orientado a la web utilizando la metodología RUP, además de cada una de las herramientas que se utilizaron en las distintas fases del proceso. Entre las herramientas podemos nombrar al lenguaje de modelado unificado UML. La herramienta CASE utilizada fue STARUML porque permite reducir los tiempos de desarrollo de los sistemas orientados a objetos gracias a que utiliza UML. También cabe destacar en la fase final de pruebas se utilizo la norma ISO 9126, que un estándar internacional para la evaluación del Software

Este artículo está estructurado de la siguiente manera: la sección 2 explica la metodología RUP, y las diferentes fases

que proceso y desarrollo. La sección 3 muestra las diferentes herramientas utilizadas durante el proceso de desarrollo. En la sección 4 mostraremos cada uno de los parámetros de evaluación se la norma ISO 9126. En la sección 5 se muestran los resultados que se obtienen después de aplicar un estándar para la evaluación del Software. En la sección 6 se muestra la conclusión final tras realizar el proyecto de software y finalmente en la sección 7 se muestran las referencias bibliográficas.

II. METODOLOGIA RUP (RATIONAL UNIFIED PROCESS)

RUP es uno de los procesos más generales de los existentes actualmente, ya que en realidad está pensado para adaptarse a cualquier tipo de proyecto. El ciclo de vida de RUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y haciendo hincapié en las distintas actividades. En las iteraciones de cada fase se hacen diferentes esfuerzos en diferentes actividades

1. Inicio (puesta en marcha)
2. Elaboración (definición análisis y diseño)
3. Construcción (implementación)
4. Transición (fin del proyecto y puesta en producción)

FASE DE INICIO.

Durante la fase de inicio las iteraciones tienen mayor énfasis en actividades de modelado del negocio y de requisitos.

FASE DE ELABORACIÓN.

En la fase de elaboración, las iteraciones se orientan al desarrollo de ciclos de desarrollo de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación.

FASE DE CONSTRUCCIÓN.

Se implementan las clases y objetos en ficheros fuente, binarios, ejecutables y demás. El resultado final es un sistema ejecutable.

FASE DE TRANSICIÓN.

La actividad tiene como objetivo, producir con éxito distribuciones del producto y distribuirlo a los usuarios. Las actividades implicadas incluyen:

DURANTE TODO EL PROYECTO.

Se vigila el cumplimiento de los objetivos, gestión de riesgos y restricciones para desarrollar un producto que sea acorde a los requisitos de los clientes y los usuarios.

III. HERRAMIENTAS Y TECNOLOGÍAS DE DESARROLLO DEL SOFTWARE.

➤ ARQUITECTURA CLIENTE/SERVIDOR

La arquitectura cliente servidor es una relación entre procesos corriendo en máquinas separadas, que interactúan mediante un mecanismo de pasaje de mensajes pedido de servicio y respuesta WEB. Donde el servidor es un proveedor de servicios y el cliente es un consumidor de servicios.

Cliente, es el que inicia un requerimiento de servicio, donde el requerimiento inicial puede convertirse en múltiples requerimientos de trabajo a través de las redes LAN o WAN. La ubicación de los datos o de las aplicaciones es totalmente transparente para los clientes.

Servidor Web, es un programa que se ejecuta continuamente en un computador, manteniéndose a la espera de peticiones de ejecución que le hará un cliente o un usuario de Internet. El servidor Web se encarga de contestar a estas peticiones de forma adecuada, entregando como resultado una página Web o información de todo tipo de acuerdo a los comandos solicitados. [WEB1]

➤ MODELO VISTA CONTROLADOR.

Un problema muy frecuente para los programadores es que tienen problema en la solución o mejora de código existente. Para los casos en los que hay que resolver un problema parecido a algo que ya tienen hecho, mejorar el aspecto de un programa, mejorar su algoritmo, etc. Esta tarea se

facilita mucho si a la hora de programar tienen la precaución de separar el código en varias partes que sean susceptibles de ser reutilizadas sin modificaciones.

En casi cualquier programa que hagan podemos encontrar tres partes bien diferenciadas:

- a) Por un lado, está el problema que se trata de resolver. El problema suele ser independiente de cómo queramos que nuestro programa recoja los resultados o cómo queremos que los presente. El código constituiría el modelo.
- b) Otra parte clara es la presentación visual que queramos hacer de la aplicación. El código es la vista. La llamaré interfaz gráfica por ser lo más común, pero podría ser de texto, de comunicaciones con otro programa externo, con la impresora, etc.
- c) La tercera parte de código es aquel código que toma decisiones, algoritmos, etc. Es código que no tiene que ver con las ventanas visuales ni con las reglas de nuestro modelo.

Tras este tipo de ordenación, si se quiere reaprovechar cosas en futuros programas, está claro que el modelo debe ser independiente. Las clases (o funciones y estructuras) del modelo no deben ver a ninguna clase de los otros grupos. De esta forma podremos compilar el modelo en una librería independiente que podremos utilizar en cualquier programa que hagamos. Siguiendo con el orden de posibilidad de reutilización, el controlador podría (y suele) ver clases del modelo, pero no de la vista. De esta forma, el cambio de interfaces gráficas no implicará retocar el algoritmo y recompilarlo, con los consiguientes riesgos de estropearlo además del trabajo del retoque.

La vista es lo más cambiante, así que podemos hacer que vea clases del modelo y del controlador. Si cambiamos algo del controlador o del modelo, es bastante seguro que tengan como mínimo que recompilar la interface gráfica. [WEB1]

➤ LENGUAJE UNIFICADO DE MODELADO – UML.

El lenguaje unificado de modelado se ha vuelto el lenguaje de modelado estándar usado en análisis y diseño orientado a

objetos.

Popular lenguaje de modelado de sistemas de software. Se trata de un lenguaje gráfico para construir, documentar, visualizar y especificar un sistema de software. Entre otras palabras, UML se utiliza para definir un sistema de software. Posee la riqueza suficiente como para crear un modelo del sistema, pudiendo modelar los procesos de negocios, funciones, esquemas de bases de datos, expresiones de lenguajes de programación, etc. Para ello utiliza varios tipos diferentes de diagramas.

Algunos programas gratuitos para modelar en UML son: ArgoUML, DÍA, gModeler, MonoUML, StarUML, TCM, Umbrello Herramienta, UMLet.

UML no es un método de desarrollo. Es un estándar para representar los diagramas del análisis al diseño y de este al código. No son una serie de pasos que llevan a producir código a partir de unas especificaciones. UML al no ser un método de desarrollo es independiente del ciclo de desarrollo que vayas a seguir, puede encajar en un tradicional ciclo en cascada, o en un evolutivo ciclo en espiral o incluso en los métodos ágiles de desarrollo. [WEB1]

➤ APACHE.

Servidor Web de distribución libre y de código abierto, siendo el más popular del mundo desde abril de 1996, con una implementación actual del 50% del total de servidores Web del mundo (hasta agosto de 2007). Apache fue la primera alternativa viable para el servidor Web de Netscape Communications, actualmente conocido como Sun Java System Web Server. Apache es desarrollado y mantenido por una comunidad abierta de desarrolladores bajo el auspicio de la Apache Software Foundation. La aplicación permite ejecutarse en múltiples sistemas operativos como Windows, Novell NetWare y los sistemas basados en Unix. La versión 2 del servidor Apache fue una reescritura sustancial de la mayor parte del código de Apache 1.x, enfocándose en una mayor modularización y el desarrollo de una capa de portabilidad, el Apache Portable Runtime. Apache 2.x incluyó multitarea en UNIX, mejor soporte para plataformas no Unix (como Windows), una nueva API Apache y soporte para IPv6.

Características de Apache:

- Soporte para los lenguajes Perl, python, tcl y PHP
- Módulos de autenticación: mod_access, mod_auth y mod_digest
- Soporte para SSL y TLS
- Permite la configuración de mensajes de errores personalizados y negociación de contenido
- Permite autenticación de base de datos basada en SGBD

Uso de Apache es principalmente usado para servir páginas Web estáticas y dinámicas en la WWW. Apache es el servidor Web del popular sistema XAMP, junto con MySQL y los lenguajes de programación PHP/ Perl/ Python. La "X" puede ser la inicial de cualquier sistema operativo, si es Windows: WAMP, si es el Linux: LAMP, etc. [WEB1]

➤ **PHP.**

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas. Es usado principalmente en interpretación del lado del servidor pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas.

PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor. Fue creado originalmente por Rasmus Lerdorf en 1994, sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.

PHP es un lenguaje interpretado de propósito general ampliamente usado y que está diseñado especialmente para desarrollo Web y puede ser insertado dentro de código HTML. Generalmente se ejecuta en un servidor Web, tomando el código en PHP como su entrada y creando páginas Web como salida. Puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas sin costo alguno. Es también el módulo Apache más popular entre las computadoras que utilizan Apache como servidor Web. La más reciente versión principal del PHP fue la versión 5.2.6 de 1 de mayo de 2008. Cuando el cliente hace una petición al servidor para

que le envíe una página Web, el servidor ejecuta el intérprete de PHP. Éste procesa el código PHP solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF, Flash, así como imágenes en diferentes formatos.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como UNIX y Windows, y puede interactuar con los servidores de Web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI. Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia GNU, existe además un entorno de desarrollo integrado comercial llamado Zend Studio. [WEB1]

➤ **JAVASCRIPT.**

JavaScript es un lenguaje de Alto Nivel con capacidades orientadas a objetos, incrustado en los navegadores como Netscape e Internet Explorer, es decir, no se necesita instalar ningún software adicional pues estos navegadores tienen a JavaScript integrado, lo único que se requiere es agregar los scripts en los documentos HTML bajo la sintaxis propia tanto de JavaScript como de HTML. El lenguaje fue inventado por Brendan Eich en la empresa Netscape Communications, que es la que desarrolló los primeros navegadores Web comerciales. Apareció por primera vez en el producto de Netscape llamado Netscape Navigator 2.0.

Tradicionalmente, se venía utilizando en páginas Web HTML, para realizar tareas y operaciones en el marco de la aplicación únicamente cliente, sin acceso a funciones del servidor. JavaScript se ejecuta en el agente de usuario al mismo tiempo que las sentencias van descargándose junto con el código HTML. [WEB1]

➤ **AJAX.**

Acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo Web para crear

aplicaciones interactivas. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

Ajax es una tecnología asíncrona, en el sentido de que los datos adicionales se requieren al servidor y se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. JavaScript es el lenguaje interpretado en el que normalmente se efectúan las funciones de llamada de Ajax mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En cualquier caso, no es necesario que el contenido asíncrono esté formateado en XML. [WEB1]

➤ HOJAS DE ESTILO.

Las hojas de estilo representan un avance importante para los diseñadores de páginas Web, al darles un mayor rango de posibilidades para mejorar la apariencia de sus páginas. En los entornos científicos en que la Web fue concebida, la gente estaba más preocupada por el contenido de sus páginas que por su presentación. A medida que la Web era descubierta por un espectro mayor de personas de distintas procedencias, las limitaciones del HTML se convirtieron en fuente de continua frustración, y los autores se vieron forzados a superar las limitaciones estilísticas del HTML. Aunque las intenciones han sido buenas mejorar la presentación de las páginas Web, las técnicas para conseguirlo han tenido efectos secundarios negativos.

➤ MySQL

MySQL es un sistema de gestión de base de datos, multi hilo y multi usuario. La siguiente es una lista de algunas de las características de MySQL:

- Escrito en C y en C++
- Probado con un amplio rango de compiladores diferentes.
- Funciona en diferentes plataformas.
- APIs disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, y Tcl.

Uso completo de multi-threaded mediante threads de kernel. Pueden usarse fácilmente múltiple CPUs si están disponibles.

Puede mezclar tablas de distintas bases de datos en la misma consulta (como un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite la verificación basada en el host. Las contraseñas son seguras porque todo el tráfico de contraseñas está encriptado cuando se conecta con un servidor.

Soporte a grandes bases de datos. Se ha usado MySQL Server con bases de datos que contienen 50 millones de registros. También existen usuarios que usan MySQL Server con 60.000 tablas y cerca de 5.000.000.000.000 de registros. [WEB1]

IV NORMA ISO 9126.

ISO 9126 es un estándar internacional para la evaluación del Software. El estándar ISO 9126 está conformado en cuatro partes las cuales dirigen, respectivamente, lo siguiente: modelo de calidad, métricas externas, métricas internas y calidad en las métricas de uso. El modelo de calidad establecido en la primera parte del estándar, ISO 9126-1, clasifica la calidad del software en un conjunto estructurado de características y sub-características de la siguiente manera [ART1]:

a) Funcionalidad, un conjunto de atributos que se relacionan con la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen lo indicado o implica necesidades.

b) Fiabilidad, un conjunto de atributos relacionados con la capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período de tiempo establecido.

c) Usabilidad, conjuntos de atributos relacionados con el esfuerzo necesitado para el uso, y en la valoración individual de tal uso, por un establecido o implicado conjunto de usuarios.

d) Mantenimiento, conjunto de atributos relacionados con el esfuerzo necesitado para modificar las especificaciones.

V. RESULTADOS OBTENIDOS:

➤ FUNCIONALIDAD.

La relación que nos permite calcular el punto función es el siguiente:

$$PF = \text{Cuenta Total} \times (\text{Grado de confiabilidad} + \text{Tasa de error} \Sigma Fi)$$

Donde:

PF: Medida de funciones

Cuenta total: Es la sumatoria del producto de factor ponderación y valores de los parámetros.

Grado de confiabilidad: Es la confiabilidad estimada del sistema.

Tasa de error: Probabilidad subjetiva estimada del dominio de la información.

ΣFi: Son valores de ajuste de complejidad.

Los valores obtenidos después de aplicar el parámetro de funcionalidad son los siguientes:

$$\text{Cuenta total} = 331$$

$$\text{Grado de confiabilidad} = 65\%$$

$$\text{Tasa de error} = 1\%$$

$$\Sigma Fi = 24$$

Con la obtención de los anteriores datos calculamos:

$$PF = 331 \times (0.65 + 0.01 \times 24) \quad PF = 294.59$$

Si consideramos el máximo valor de reajuste de complejidad como $\Sigma Fi = 70$

$$PF = 331 \times (0.65 + 0.01 \times 70) \quad PF = 446.85$$

La relación obtenida entre ambos puntos es:

$$PF / PF \text{ máximo} = 294.59 / 446.85 = 0.66$$

Por lo que se concluye que la funcionalidad del sistema es de 66%.

➤ FIABILIDAD.

Para calcular el índice de error tomamos 8 ejecuciones en una semana durante un mes y medio se tiene reemplazando tenemos:

$$F(t) = 0.81 e^{(-1/5) \times 6} = 0.24$$

Siendo 0.24 la probabilidad de fallo. Por lo que el software tendrá una confiabilidad de 76%.

➤ USABILIDAD.

Para medir la usabilidad del sistema contamos las funciones evidentes al usuario y las comparamos con el número total de funciones del sistema.

$$x = a/b$$

$$a = 29$$

$$b = 32$$

$$x = 29/32$$

$$x = 0.91$$

Esto implica que casi el 91% de las funciones del sistema son entendibles a los usuarios.

MANTENIMIENTO.

Para verificar la estabilidad del sistema es decir el índice de madurez del software (IMS), se probó con los cambios que ocurrieron en el desarrollo del software.

Para ello tenemos:

$$IMS = [MT - (Fc + Fa + Fe)] / MT$$

Dónde:

MT: Número de módulos en la versión actual

Fc: Número de módulos en la versión actual que se han cambiado

Fa: Número de módulos en la versión actual que se han añadido

Fe: Número de módulos en la versión actual que se han eliminado

Considerando una ponderación de:

| Intervalo | Calificación |
|----------------------------|--------------|
| $0\% \leq IMS < 25\%$ | Deficiente |
| $25\% \leq IMS < 50\%$ | Suficiente |
| $50\% \leq IMS < 75\%$ | Buena |
| $75\% \leq IMS \leq 100\%$ | Optima |

Remplazando datos tenemos:

$$IMS = [10 - (1 + 2 + 0)] / 10 = 0.7$$

Tomando en cuenta las ponderaciones se concluye que 70% está en el intervalo con calificación de Buena que implica aceptablemente estable.

VI. CONCLUSIONES.

En este artículo se muestra como se puede desarrollar un sistema web implementando la metodología RUP. De esta metodología se trata de exponer de manera rápida cada una de las fases de proceso dentro del

desarrollo de software así como las distintas actividades que se deben realizar durante el proyecto.

Las herramientas y tecnologías de desarrollo de software que se sirvieron de apoyo dentro del proyecto también fueron descritas y analizadas dentro del presente documento.

En el caso práctico demostramos los resultados obtenidos tras evaluar la calidad del software en base a la norma ISO 9126, de acuerdo a los parámetros de funcionalidad, usabilidad, confiabilidad y mantenimiento; demostrando un rendimiento bastante aceptable del sistema.

VII. REFERENCIAS BIBLIOGRAFICAS.

Rational Software. Presentación disponible en <http://www.rational.com/uml> como UMLconf.zip editorial Pearson Educación

[ART1] Artículo No. 1 "CALIDAD DEL SOFTWARE", Juan Manuel Cueva Lovelle, conferencia 21 de octubre de 1999, Grupo GIDIS, Universidad Nacional de la Pampa cueva@lsi.uniovi.es Fecha: 2008-09-15 hrs.: 18:46 p.m.

[WEB1]
http://es.wikipedia.org/wiki/Ingenieria_de_software Fecha: 2008-09-15 hrs.: 18:46 p.m