

**INFORME DE PRÁCTICA EMPRESARIAL
PRAGMA S.A.**

SEBASTIAN HOLGUIN RIVERA

**CORPORACIÓN UNIVERSITARIA LASALLISTA
FACULTAD DE INGENIERÍAS
INGENIERÍA INFORMÁTICA
CALDAS- ANTIOQUIA
2012**

**INFORME DE PRÁCTICA EMPRESARIAL
PRAGMA S.A.**

SEBASTIAN HOLGUIN RIVERA

Trabajo de grado para optar el título de Ingeniería Informática

**ASESOR
ALEJANDRO DIAZ BETANCOURT
INGENIERO EN SISTEMAS**

**CORPORACIÓN UNIVERSITARIA LASALLISTA
FACULTAD DE INGENIERÍAS
INGENIERÍA INFORMÁTICA
CALDAS- ANTIOQUIA
2012**

Nota de Aceptación

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Ciudad y fecha (día, mes, año)

AGRADECIMIENTOS

Esta es una experiencia que marca dentro de mí un recuerdo imborrable, quisiera agradecer a todos los docentes que han acompañado mi formación como profesional en especial a los señores: Hugo Elías López Reyes, Jairo Miguel Vergara, Cesar Ruiz Jaramillo, Juan Carlos Quintero, Mauricio Bedoya y más personas que participaron directa e indirectamente en todo este proceso de consecución de metas y sueños.

Agradezco a mi familia en especial a mis padres piezas importantes que día a día me brindaron su apoyo moral para obtener tan anhelado título.

A Alejandro Díaz Betancourt, asesor de tesis, por guiarme y aconsejarme para alcanzar este gran logro.

TABLA DE CONTENIDO

	Pág
INTRODUCCIÓN	23
1. JUSTIFICACIÓN	25
2. OBJETIVOS	26
2.1 GENERAL	26
2.2 ESPECÍFICOS	26
3. RESEÑA	27
3.1 HISTORIA	27
3.2 MISIÓN	27
3.3 VISIÓN	27
3.4 PRAGMA S.A.	28
4. MARCO TEÓRICO	30
4.1 ESTADO DEL ARTE O ANTECEDENTES	30
4.1.1 Portales Familia	30
4.1.1.1 Nosotras Online	30
4.1.1.2 Cosas de Familia	31
4.2 EPISERVER FRAMEWROK CMS	32
4.2.1 Que es CMS	33
4.2.2 Características	34
4.3 MICROSOFT VISUAL STUDIO .NET	36

4.3.1 Características	37
4.3.2 Que es C#	37
4.4 ORACLE	38
4.4.1 Características	38
4.4.1.1 Ventajas	38
4.4.1.2 Inconvenientes	39
4.5 JAVASCRIPT	40
4.5.1 Que es jQuery	40
4.5.2 Características	41
4.6 CASCADING STYLE SHEETS	43
4.6.1 Características	43
4.7 FIREBUG	44
4.7.1 Historia	45
5. DISEÑO METODOLOGICO	46
5.1. DESCRIPCIÓN DEL PROYECTO	46
5.1.1 Información general	47
5.1.3 Descripción del plan proyecto.	48
5.1.3.1 Organización del proyecto	48
5.1.3.2 Recursos	51
5.2 METODOLOGIA DE TRABAJO TSP/PSP	52
5.2.1 Team Software Process	52
5.2.1.1 Objetivos	52

5.2.1.2 Antecedentes	53
5.2.1.3 Estructura de TSP	53
5.2.1.4 Problemas Comunes	54
5.2.1.5 Metodología TSP	54
5.2.1.6 Lanzamientos TSP	54
5.2.1.7 Plan de Calidad	55
5.2.1.8 Ciclo de Vida TSP	56
5.2.2 Personal Software Process	56
5.2.2.1 Modelo PSP	56
5.2.2.2 Características	57
5.2.2.3 Pasos a Seguir	57
5.2.2.4 Ventajas y Desventajas	58
5.2.2.5 Ciclo de Vida PSP	58
5.3 LECCIONES APRENDIDAS	59
5.4 RESULTADOS	59
6. CONCLUSIONES	60
BIBLIOGRAFIA	61
ANEXO	64

INDICE DE FIGURAS

		Pag.
Figura 1	Logo Pragma S.A.	27
Figura 2	Home Actual Nosotras Online	30
Figura 3	Home Actual Cosas de Familia	31
Figura 4	EPiServer Centro de Informes	34
Figura 5	Firebug	45
Figura 6	Diagrama Descripción del Sistema	48
Figura 7	Diagrama de Componentes	49
Figura 8	Diagrama de Clases	50
Figura 9	Objetivo TSP	52
Figura 10	Estructura TSP	53
Figura 11	Porcentaje de Defectos Encontrados	55
Figura 12	Ciclo de vida PSP	58

LISTA DE TABLAS

		Pág.
Tabla 1	Descripción del sistema	49
Tabla 2	Vistas dinámicas	50

LISTA DE ANEXOS

	Pág.
ANEXO A. Cronograma de Actividades.	64

GLOSARIO

ALGORITMOS: En matemáticas, ciencias de la computación y disciplinas relacionadas, un algoritmo (del latín, dixit algorithmus y éste a su vez del matemático persa Al Juarismi) es una lista bien definida, ordenada y finita de operaciones que permite hallar la solución a un problema. Dado un estado inicial y una entrada, a través de pasos sucesivos y bien definidos se llega a un estado final, obteniendo una solución. Los algoritmos son objeto de estudio de la algoritmia.

APLICACIÓN: es un tipo de programa informático diseñado como herramienta para permitir a un usuario realizar un o diversos tipos de trabajo. Esto lo diferencia principalmente de otros tipos de programas como los sistemas operativos (que hacen funcionar al ordenador), las utilidades (que realizan tareas de mantenimiento o de uso general), y los lenguajes de programación (con el cual se crean los programas informáticos).

AUTOMATIZACIÓN: es el uso de sistemas o elementos computarizados para controlar maquinarias y/o procesos industriales substituyendo a operadores humanos.

BACKUP: Es la copia total o parcial de información importante del disco duro, CDs, bases de datos u otro medio de almacenamiento. Esta copia de respaldo debe ser guardada en algún otro sistema de almacenamiento masivo, como ser discos duros, CDs, DVDs o cintas magnéticas (DDS, Travan, AIT, SLR, DLT y VXA). Los backups se utilizan para tener una o más copias de información considerada importante y así poder recuperarla en el caso de pérdida de la copia original.

BASES DE DATOS: es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. En la actualidad, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital (electrónico), que ofrece un amplio rango de soluciones al problema de almacenar datos.

CALIDAD: Calidad tiene muchas definiciones, pero la básica es aquella que dice que aquel producto o servicio que nosotros adquiramos satisfaga nuestras expectativas sobradamente. Es decir, que aquel servicio o producto funcione tal y como nosotros queramos y para realizar aquella tarea o servicio que nos tiene que realizar. Con todo y a pesar de esta definición el término "Calidad" siempre será entendido de diferente manera por cada uno de nosotros, ya que para unos la Calidad residirá en un producto y en otros en su servicio posventa de este producto, por poner un ejemplo. Lo cierto es que nunca llegaremos a definir exactamente lo que representa el término Calidad a pesar de que últimamente este término se haya puesto de moda.

CHECKLIST: Los checklist o listas de comprobación, son un elemento cotidiano que muchas personas utilizan para asegurarse de que están cumpliendo algo. En este caso, las listas de comprobación miden el progreso del cumplimiento de los objetivos del software y ver aquellas áreas en las que se necesita trabajar.

CIBERESPACIO: es una realidad virtual que se encuentra dentro de los ordenadores y redes del mundo. El ciberespacio es un tema recurrente en la ciencia ficción. El término "ciberespacio" proviene de la novela de William Gibson Neuromante, publicada en 1984, y a su vez de una obra anterior del mismo autor, Burning Chrome. El 8 de febrero de 1996, en Davos, Suiza, John Perry Barlow escribió la Declaración de independencia del ciberespacio en la que exhortaba a los gobiernos a no ejercer soberanía sobre el ciberespacio, definido por el mismo como "el nuevo hogar de la Mente".

CMS: Un sistema de gestión de contenidos (en inglés Content Management System, abreviado CMS) es un programa que permite crear una estructura de soporte (framework) para la creación y administración de contenidos, principalmente en páginas web, por parte de los administradores, editores, participantes y demás roles.

Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio web. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio web sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al sistema y otro de nivel superior (moderador o administrador) que permite que estos contenidos sean visibles a todo el público (los aprueba).

CPU: La unidad central de procesamiento o CPU (por el acrónimo en inglés de central processing unit), o simplemente el procesador o microprocesador, es el componente en una computadora digital que interpreta las instrucciones y procesa los datos contenidos en los programas de la computadora. Las CPU proporcionan la característica fundamental de la computadora digital (la programabilidad) y son uno de los componentes necesarios encontrados en las computadoras de cualquier tiempo, junto con el almacenamiento primario y los dispositivos de entrada/salida.

CSS: Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.

EMPOTRADO: En informática, software que reside en memoria de solo lectura y se utiliza para controlar productos y sistemas de los mercados industriales y de consumo. El software empotrado puede ejecutar funciones muy limitadas y curiosas (p. Ej.: el control de las teclas de un horno de microondas) o suministrar una función significativa y con capacidad de control (p. Ej.: funciones digitales en un automóvil, tales como control de la gasolina, indicaciones en el salpicadero, sistemas de frenado, etc.).

FICHERO: Un archivo informático o fichero es un conjunto de bits almacenado en un dispositivo periférico. Un archivo es identificado por un nombre y la descripción de la carpeta o directorio que lo contiene. Los archivos informáticos se llaman así porque son los equivalentes digitales de los archivos en tarjetas, papel o microfichas del entorno de oficina tradicional. Los archivos informáticos facilitan

una manera de organizar los recursos usados para almacenar permanentemente datos en un sistema informático.

GPL: Licencia Pública General de GNU o más conocida por su nombre en inglés GNU General Public License o simplemente su acrónimo del inglés GNU GPL, es una licencia creada por la Free Software Foundation a mediados de los 80, y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios. Existen varias licencias "hermanas" de la GPL, como la licencia de documentación libre de GNU (GFDL) que cubre los artículos de la Wikipedia, la Open Audio License, para trabajos musicales, etcétera, y otras menos restrictivas, como la LGPL, o la LGPL (Lesser General Public License, antes Library General Public License), que permiten el enlace dinámico de aplicaciones libres a aplicaciones no libres.

HARDWARE: corresponde a todas las partes físicas y tangibles de una computadora, sus componentes eléctricos, electrónicos, electromecánicos y mecánicos; sus cables, gabinetes o cajas, periféricos de todo tipo y cualquier otro elemento físico involucrado.

HTML: El HTML (Hyper Text Markup Language) es el lenguaje con el que se escriben las páginas web. Es un lenguaje de hipertexto, es decir, un lenguaje que permite escribir texto de forma estructurada, y que está compuesto por etiquetas, que marcan el inicio y el fin de cada elemento del documento.

Un documento hipertexto no sólo se compone de texto, puede contener imágenes, sonido, vídeos, etc., por lo que el resultado puede considerarse como un documento multimedia.

Los documentos HTML deben tener la extensión html o htm, para que puedan ser visualizados en los navegadores (programas que permiten visualizar las páginas web). Los navegadores se encargan de interpretar el código HTML de los documentos, y de mostrar a los usuarios las páginas web resultantes del código interpretado.

INGENIERÍA: es el conjunto de conocimientos y técnicas científicas aplicadas, que se dedica a la resolución u optimización de los problemas que afectan directamente a la humanidad. En ella, el conocimiento, manejo y dominio de las matemáticas y física, obtenido mediante estudio, experiencia y práctica, se aplica con juicio para desarrollar formas eficientes de utilizar los materiales y las fuerzas de la naturaleza para beneficio de la humanidad y del ambiente. Pese a que la ingeniería como tal (transformación de la idea en realidad) está intrínsecamente

ligada al ser humano, su nacimiento como campo de conocimiento específico viene ligado al comienzo de la revolución industrial, constituyendo uno de los actuales pilares en el desarrollo de las sociedades modernas.

INTERFAZ: conocida también como GUI (del inglés graphical user interface) es un tipo de interfaz de usuario que utiliza un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Habitualmente las acciones se realizan mediante manipulación directa para facilitar la interacción del usuario con la computadora. Surge como evolución de la línea de comandos de los primeros sistemas operativos y es pieza fundamental en un entorno gráfico. Como ejemplo de interfaz gráfica de usuario podemos citar el entorno de escritorio del sistema operativo Windows, el X-Window de Linux o el de Mac OS X, Aqua. En el contexto del proceso de interacción persona-ordenador, la interfaz gráfica de usuario es el artefacto tecnológico de un sistema interactivo que posibilita, a través del uso y la representación del lenguaje visual, una interacción amigable con un sistema informático.

INTERNET: es un conjunto descentralizado de redes de comunicación interconectadas, que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial. Sus orígenes se remontan a 1969, cuando se estableció la primera conexión de computadoras, conocida como ARPANET, entre tres universidades en California y una en Utah, Estados Unidos.

LENGUAJE DE PROGRAMACIÓN: es un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Es utilizado para controlar el comportamiento físico y lógico de una máquina. Aunque muchas veces se usan los términos 'lenguaje de programación' y 'lenguaje informático' como si fuesen sinónimos, no tiene por qué ser así, ya que los lenguajes informáticos engloban a los lenguajes de programación y a otros más, como, por ejemplo, el HTML (lenguaje para el marcado de páginas web que no es propiamente un lenguaje de programación). Un lenguaje de programación permite a uno o más programadores especificar de manera precisa sobre qué datos debe operar una computadora, cómo estos datos deben ser almacenados o transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar relativamente próximo al lenguaje humano o natural, tal como sucede con el lenguaje Léxico. Una característica relevante de los lenguajes de programación es precisamente que más de un programador puedan tener un conjunto común de instrucciones que puedan ser comprendidas entre ellos para realizar la construcción del programa de forma colaborativa.

LICENCIAMIENTO: El licenciamiento de un software le otorga derecho legal de ejecutar y utilizar el software. Un contrato de licenciamiento controla el uso de la licencia de un software. Normalmente los contratos de licenciamiento permiten que el software sea ejecutado en un número limitado de PCs y que se realicen copias sólo con propósitos de respaldo. Microsoft tiene varios programas de licenciamiento, cada uno diseñado para diferentes necesidades.

LINQ SQL: Language Integrated Query es un lenguaje de consultas creado para facilitar la explotación de los datos sin importar el tipo de fuente de datos utilizada. Permite consultar información en tecnologías tan diferentes como ficheros XML, bases de datos relacionales o colecciones fuertemente tipadas.

LOC: Sigla de líneas de código.

OBSOLESCENCIA: es la caída en desuso de máquinas, equipos y tecnologías motivada no por un mal funcionamiento del mismo, sino por un insuficiente desempeño de sus funciones en comparación con las nuevas máquinas, equipos y tecnologías introducidos en el mercado.

PATENTE: es un derecho negativo, otorgado por el Estado a un inventor o a su causahabiente (titular secundario). Este derecho permite al titular de la patente impedir que terceros hagan uso de la tecnología patentada, y por lo tanto el titular de la patente es el único que puede hacer uso de la tecnología que reivindica en la patente. Las patentes no son de duración indefinida si no que caducan después de un período determinado que normalmente es de veinte años. Después de la caducidad de la patente cualquier persona puede hacer uso de la tecnología de la patente sin la necesidad del consentimiento del titular de ésta.

PIRATERÍA: La piratería de software es la copia, reproducción, utilización o fabricación no autorizadas de productos de software protegidos por las leyes de copyright internacionales y de los Estados Unidos.

PROCESS DASHBOARD: Es un programa gratuito creado como iniciativa a dar soporte a equipos que manejan la metodología Personal Software Process y Team Software Process.

PROGRAMA: Conjunto unitario de instrucciones que permite a un ordenador realizar funciones diversas, como el tratamiento de textos, el diseño de gráficos, la resolución de problemas matemáticos, el manejo de bancos de datos.

REDUNDANCIA: en bases de datos se refiere a la repetición inútil de datos.

REQUERIMIENTOS: En la ingeniería de sistemas, un requerimiento (del inglés requirement: 'requisito') es una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio. Se usa en un sentido formal en la ingeniería de sistemas o la ingeniería de software.

En la ingeniería clásica, los requisitos se utilizan como datos de entrada en la etapa de diseño del producto. Establecen qué debe hacer el sistema, pero no cómo hacerlo.

La fase de captura, análisis y registro de requisitos puede estar precedida por una fase de análisis conceptual del proyecto. Esta fase puede dividirse en recolección de requisitos, análisis de consistencia e integridad, definición en términos descriptivos para los desarrolladores y un esbozo de especificación, previo al diseño completo.

ROBOT: es una entidad virtual o mecánica artificial. En la práctica, esto es por lo general un sistema electromecánico que, por su apariencia o sus movimientos, ofrece la sensación de tener un propósito propio. La palabra robot puede referirse tanto a mecanismos físicos como a sistemas virtuales de software, aunque suele aludirse a los segundos con el término de bots.

REQUISITOS FUNCIONALES: Son aquellos requisitos que hacen referencia a una funcionalidad que debe tener el sistema.

REQUISITOS NO FUNCIONALES: Son propiedades que debe tener el sistema y que a pesar que no influyen directamente sobre la funcionalidad, deben ser tenidos en cuenta, algunos requisitos no funcionales pueden ser: Tiempos de respuesta, confiabilidad, mantenimiento, requisitos de almacenamiento, capacidades de banda ancha si se trata de una aplicación web.

SCARAB: Para facilitar la administración de incidencias o requerimientos entre Pragma y sus clientes, se seleccionó como herramienta de apoyo Scarab.

Scarab es un sistema de seguimiento de incidencias o nuevos requerimientos tanto técnicos como no técnicos, permitiendo el ingreso de información a través de formularios, a los usuarios involucrados con una incidencia le llega un correo electrónico informando sobre cambios de estados, además permite a los usuarios realizar consultas personalizadas dependiendo de sus necesidades.

SEO: El posicionamiento en buscadores, posicionamiento web u Optimizador de motores de búsqueda (SEO por sus siglas en inglés, de Search Engine Optimization) es el proceso de mejorar la visibilidad de un sitio web en los diferentes buscadores, como Google, Yahoo! o Bing de manera orgánica, es decir sin pagarle dinero al buscador para tener acceso a una posición destacada en los resultados.

La tarea de optimizar la estructura de una web y el contenido de la misma, así como la utilización de diversas técnicas de linkbuilding, linkbaiting o contenidos virales con el objetivo de aparecer en las primeras posiciones de los resultados de los buscadores (cuando un usuario busca por una determinada palabra clave o keyword), es conocida como SEO, sigla en inglés que significa Search Engine Optimizer, que traduce, 'Optimizador de motores de búsqueda'.

La aplicación de técnicas SEO suele ser más intensa en sitios web con mucha competencia y lo que se pretende con su aplicación es el posicionarse por encima de los competidores por determinadas palabras clave.

Las técnicas SEO pueden ser desmedidas y afectar los resultados naturales de los grandes buscadores por lo que si incumplen las cláusulas y condiciones de uso de los mismos pueden ser consideradas, en algunos casos, como una forma de SPAM, el spamdexing.

El trabajo es amplio, ya que el posicionamiento involucra al código de programación, al diseño y a los contenidos. También nos referimos a SEO para definir las personas que realizan este tipo de trabajo.

SISTEMA DE GESTIÓN DE BASES DE DATOS: Son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

SOFTWARE: La palabra «software» se refiere al equipamiento lógico o soporte lógico de una computadora digital, y comprende el conjunto de los componentes lógicos necesarios para hacer posible la realización de una tarea específica, en contraposición a los componentes físicos del sistema (hardware). Tales componentes lógicos incluyen, entre otros, aplicaciones informáticas tales como procesador de textos, que permite al usuario realizar todas las tareas concernientes a edición de textos; software de sistema, tal como un sistema operativo, el que, básicamente, permite al resto de los programas funcionar adecuadamente, facilitando la interacción con los componentes físicos y el resto de las aplicaciones, también provee una interfaz para el usuario. No hay un consenso sobre qué máquinas pueden ser consideradas robots, pero sí existe un acuerdo general entre los expertos y el público sobre que los robots tienden a hacer parte o todo lo que sigue: moverse, hacer funcionar un brazo mecánico,

sentir y manipular su entorno y mostrar un comportamiento inteligente, especialmente si éste comportamiento imita al de los humanos o a otros animales.

TECNOLOGÍA: es el conjunto de conocimientos que permiten construir objetos y máquinas para adaptar el medio y satisfacer nuestras necesidades. Es una palabra de origen griego, τεχνολογος, formada por tekne (τεχνη, "arte, técnica u oficio") y logos (λογος, "conjunto de saberes"). Aunque hay muchas tecnologías muy diferentes entre sí, es frecuente usar el término en singular para referirse a una de ellas o al conjunto de todas. Cuando se lo escribe con mayúscula, tecnología puede referirse tanto a la disciplina teórica que estudia los saberes comunes a todas las tecnologías, como a educación tecnológica, la disciplina escolar abocada a la familiarización con las tecnologías más importantes.

WEBAPPS: En la ingeniería software se denomina aplicación web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web (HTML, JavaScript, Java, asp.net, etc.) en la que se confía la ejecución al navegador. Las aplicaciones web son populares debido a lo práctico del navegador web como cliente ligero, así como a la facilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software a miles de usuarios potenciales. Existen aplicaciones como los webmails, wikis, weblogs, tiendas en línea y la propia Wikipedia que son ejemplos bien conocidos de aplicaciones web.

RESUMEN

Principalmente, lo que se buscó con este documento es evidenciar el proceso de la práctica realizada en la empresa Pragma S.A.

Durante la práctica se tuvo la oportunidad de apoyar en diferentes procesos de desarrollo e implementación de portales familia, estos portales tienen como objetivo el posicionamiento y divulgación de la marca familia. La metodología empleada PSP (Personal Software Process) contribuyeron al mejoramiento de la calidad de los procesos personales y junto a la otra metodología CMMI TSP (Team Software Process) la disminución de defectos de los productos entregados a los clientes, garantizando la calidad del software desarrollado en Pragma.

ABSTRACT

Mainly, what is sought in this document is to show the process of practice done in the company Pragma SA

During practice we had the opportunity to support various development processes and implementation of family portals, these portals are aimed at positioning the brand and disclosure of family The methodology used PSP (Personal Software process) contributed to improving the quality of personal processes and methodology along with the other CMMI TSP (Team software process) the reduction of defects of goods delivered to customers, ensuring the quality of software developed by Pragma.

INTRODUCCIÓN

Pragma S.A. es una empresa con 15 años de experiencia en la creación y desarrollo de soluciones de negocio basadas en Internet y medios relacionados, es una empresa de clase mundial, que compite con una serie de productos y servicios enfocados en las siguientes líneas de negocio:

- SOA y BPM: PRAGMA es una compañía experta en facilitar la agilidad de los negocios a través de la consultoría, diseño y optimización, y automatización de procesos, a través de la aplicación de los paradigmas de Service Oriented Architecture (SOA) y Business Process Management (BPM).
- Portales: Nos hemos especializado en crear, facilitar y promover la comunicación y el contacto de las entidades corporativas con su público objetivo y su grupo de trabajo. Por ello contamos con un equipo de arquitectos de soluciones, desarrolladores, diseñadores gráficos y arquitectos de información, capaces de generar soluciones coherentes, versátiles, escalables, y seguras.
- Fábrica de Software: PRAGMA se ofrece como el aliado tecnológico perfecto para el desarrollo de nuevas soluciones de software a la medida y para la asesoría en el mantenimiento y evolución de las inversiones realizadas en activos de software existentes. La eficiencia en la gestión del ciclo de desarrollo y en la gestión de proyectos, apalancada en la aplicación constante de los lineamientos de TSP™, el Proceso Unificado y el Project Management Institute, son la base sobre la cual PRAGMA garantiza el éxito de los proyectos que emprende, ya sea bajo esquemas de contratación fija o flexible (bolsas de horas y fee de recursos).

La empresa cuenta con clientes como, Bancoomeva, Bancolombia, Banco de Bogotá, Colpatría, Enlace Operativo, Familia, ING, ISAGEN, Protección, Banco Santander, Imusa, Éxito, Consejo Nacional de Operación y Camara net.

Esta última línea de negocio fábrica de software es en la cual se enfocara todo mi trabajo empresarial como Ingeniero de Proyectos, el cliente objetivo es Familia el cual innovara sus portales de Nosotras, Cosas de Familia y Familia Institucional

los cuales son destacados no solo a nivel nacional sino también a nivel internacional.

Estos portales se desarrollaran en Microsoft .Net mediante el lenguaje C# utilizando la framework EPiServer la cual facilita la administración y gestión de sitios o portales CMS, también se gestionara el acceso a datos mediante base de datos Oracle y Microsoft SQL Server.

Pragma para su desarrollo está fundamentado sobre las metodologías PSP (Personal Software Process) y TSP (Team Software Process), la cuales están enfocadas en la búsqueda de la calidad de cada desarrollo y mejores prácticas personales y de equipo, estas dos metodologías están sustentadas y vigiladas por el SEI (Software Engineering Institute) el cual vigila los procesos de desarrollo de software a nivel mundial.

1. JUSTIFICACIÓN

Debido a las actividades y proyectos realizados dentro de la compañía esta se fundamenta en ayudar y capacitar profesionales con mejores conocimientos tecnológicos y académicos, ayudando al desarrollo con calidad del personal brindando capacidades necesarias para resolver problemas y adaptarse al medio con gran facilidad y buen desempeño.

La empresa PRAGMA S.A enfoca sus productos en las siguientes líneas: integración y automatización de procesos, desarrollo de portales y fábrica de software, con clientes que quieren ser los mejores y no se conforman en su medio. Por ello requieren de personas que aspiren a tener mejores conocimientos y habilidades para aplicarlos en sus líneas de negocio, personas competentes en lo técnico y lo humano, honestas, respetuosas con los clientes, compañeros y el entorno de trabajo, con sentido común para encontrar soluciones, prácticas que lleven a Pragma a ser cada vez mejor y a lograr sus objetivos.

Con nuestra participación activa en Pragma, la Corporación Universitaria Lasallista podrá establecer un vínculo empresarial, poniendo a disposición practicantes íntegros e idóneos, reconocidos por su excelencia ética, académica y laboral, si los resultados entregados durante los próximos 6 meses se ajustan a las necesidades de la compañía el valor ganado personal representara la permanencia dentro de la misma, teniendo la posibilidad de ampliar mi experiencia laboral y personal.

El principal activo sobre el que está construido la compañía es el conocimiento, para ello utiliza el 10% del tiempo total de los empleados en enriquecimiento, para el avance técnico y tecnológico. De esta manera busca que el personal sea más competente, y estén relacionadas con las actividades del negocio.

Otro factor importante en el que tiene foco, es un entorno laboral satisfactorio, en el cual inciden dos factores, en primer lugar un clima laboral agradable en donde las relaciones de amistad y respeto sean mejores, y en segundo lugar proveer una infraestructura adecuada con un entorno idóneo para el aprendizaje, productividad, innovación, conocimiento, socialización.

2. OBJETIVOS

2.1 OBJETIVO GENERAL

- Desarrollo de los sitios Nosotras Online, Cosas de Familia y Familia Institucional los cuales pertenecen al cliente Familia.

2.2 OBJETIVOS ESPECIFICOS

- Adquirir nuevos conocimientos académicos, éticos, morales y sociales.
- Desarrollar nuevas habilidades de trabajo grupal y personal mediante las metodologías TSP y PSP.
- Implementar soluciones de calidad las cuales sean útiles para el negocio de nuestro cliente Familia.

3. RESEÑA

NOMBRE: PRAGMA S.A.

3.1 HISTORIA

Es una organización fundada en 1996 que nació con el objetivo de asesorar a las grandes empresas colombianas en el uso y aprovechamiento de Internet y sus tecnologías relacionadas dentro sus procesos productivos y comerciales. Durante estos años, Pragma se ha consolidado como una de las principales empresas en la prestación de servicios relacionados con Desarrollo de Software de Aplicaciones de Misión Crítica, Marketing Digital y Comercio Electrónico, Automatización de Procesos y la Integración de Sistemas de Información. Actualmente, Pragma es el principal proveedor de soluciones y consultoría alrededor de la Gestión de Riesgo con su producto "Cero" (Control Estratégico del Riesgo), con el cual llega a los mercados de Perú y México.

3.2 MISIÓN

Incrementar el valor del negocio de nuestros clientes, entregando servicios y soluciones innovadoras basadas en las Tecnologías de la Información, con un equipo de excelencia y estableciendo relaciones de largo plazo.

3.3 VISIÓN

Ser una empresa reconocida por su excelencia en el mercado de las tecnologías de la información, generando valor para sus clientes, empleados, accionistas y la sociedad.

Figura 1. Logo Pragma S.A,



Fuente: pragma

3.4 PRAGMA S.A.

Somos 150 personas en Bogotá y Medellín integrando la tecnología con la estrategia y la creatividad. Eso nos hace Integrales.

Desde 1996 somos el socio estratégico de muchas de las más importantes compañías del país.

Somos ingenieros, comunicadores, diseñadores, publicistas y administradores, aprovechando las capacidades interactivas y analíticas de la red.

Creemos en los retos, aumentar la rentabilidad y competitividad de nuestros clientes en sus proyectos de tecnologías de información, implementar soluciones IT de misión crítica, posicionar marcas en Internet generando experiencias para los usuarios.

Creemos en Internet porque facilita todo el público realmente objetivo, la reducción de la carga operacional de procesos empresariales

Creemos en la flexibilidad para adaptarnos a su imagen y a su plataforma de software y así como creemos hacemos que su compañía obtenga mejores resultados con la mejor relación costo - beneficio que puede encontrar.

PROTECCIÓN

"Cuando alguien me pregunta cómo me ha ido con Pragma como proveedor la primera claridad es que Pragma para mí no es un proveedor, es un aliado estratégico. Con el pasar de los años y los proyectos hemos descubierto que juntos logramos grandes metas. El acompañamiento ha sido constante..."

"Trabajamos en equipo, tenemos una comunicación constante, fluida, asertiva y apoyada en las herramientas de trabajo que la hacen metódica además, eso contribuye en gran medida al orden y éxito en la labor. Fomentamos las buenas ideas, la construcción colectiva, damos cabida a lo impensable y lo que es mejor: Lo logramos juntos".

Paula Gaviria
Analista Gerencia Comercial en Protección S.A
Protección

CERO

"El Proceso de implementación de nuestra necesidad fue muy bien coordinado entre el Banco Santander y PRAGMA, recibiendo de este toda la colaboración que estos casos requieren..."

"...Destaco el acompañamiento que nos brindan y la oportunidad en la solución de los problemas que surgen.."

"...las personas que conforman el equipo de Pragma conocen el trabajo que desarrollan y acompañan al usuario en resolver los inconvenientes que se presentan con la inmediatez que la implementación demanda"

Eugenio Cárdenas Mahecha
Gerente de Riesgo Operativo
Banco Santander

FAMILIA

"Pragma tiene un equipo de profesionales altamente calificados y todos los trabajos son de excelente calidad."

"Es una empresa comprometida con el progreso de sus clientes."

Maria Elvira Uribe
Jefe de Mercadeo Relacional Pequeñín

4. MARCO TEÓRICO

4.1 ESTADO DEL ARTE O ANTECEDENTES

En esta sección se dará a mostrar los proyectos y su estado actual en el mercado y lo que representa el cliente para la compañía y ver que tanto se adaptan los nuevos rediseños de los portales a las necesidades del cliente Familia.

4.1.1 Portales Familia

4.1.1.1 Nosotras Online

La idea consiste en conectarnos con el target el cual son las jovencitas en Latinoamérica, captar su fidelidad interactuando con ellas, los hechos 100.400 visitantes mensuales; Las adolescentes de Latinoamérica están conectadas con Nosotras Online, una marca amiga y confidente, que guía, educa y divierte. Conocimiento personalizado de cada cliente, al integrar su experiencia online con sus consumos offline, Nos sentimos orgullosos de haber hecho: Diseño gráfico e implementación de su sitio web, de sus estrategias y piezas publicitarias online, creando experiencias de marca diferenciadora e impactante.

Generación de reportes de comportamiento online, para mejorar la estrategia y afinar las soluciones de contenido, diseño y de software brindadas.

Figura 2. Home Actual Nosotras Online



Fuente: nosotrasonline

4.1.1.2 Cosas de Familia

La idea consiste en ir entendiendo las necesidades de las mamás y amas de casa, brindarles soluciones para hacer más fácil su vida, los hechos: 12.600 visitantes mensuales, Familias conectadas con interactuando con Cosas de Familia, una marca altamente posicionada, vista como una amiga, una guía para disfrutar la vida en familia. Nos sentimos orgullosos de haber hecho:

Estrategia para llegar acertadamente a este nicho de clientes.

Desarrollos de software a la medida con soluciones verdaderamente útiles para su estrategia web, generando fidelidad hacia la marca.

Diseño gráfico de su sitio web y de sus estrategias y piezas publicitarias online, creando experiencias de marcas diferenciadoras e impactantes.

Creación de contenidos útiles, relevantes, generadores de fidelidad, así como la administración de contenidos mediante software de Pragma.

Generación de reportes de comportamiento online, para mejorar la estrategia y afinar las soluciones de contenido, diseño y de software brindadas.

Famositos

"Conectarnos con el corazón de los niños acercándolos al mundo de FAMILIA": eso fue lo que obtuvimos con www.famositos.com, junto con Any y Miguel. Así les dimos el Planeta Azul y el Planeta Rosa, uno para niños y otro para niñas, con secciones como: tests, compartir fotos, juegos, descargas, lanzamientos de productos. Todo con una orientación exitosa, donde puedan divertirse, experimentar y aprender. Y siempre captando su información, posicionando marca, manteniéndonos cerca en la etapa infantil y preadolescente del target.

Figura 3. Home Actual Cosas de Familia



Fuente: cosasdefamilia

4.2 EPISERVER FRAMEWORK CMS

EPiServer es un diseño sueco sistema de publicación web (Web Content Management System , CMS web, sistemas de gestión de contenidos). EPiServer CMS se utiliza para muchos empleados pueden trabajar juntos para desarrollar una organización más grande del sitio web , así como las páginas web públicos que intranet (en el sentido de la inter-red) y extranet (login páginas protegidas para los clientes, por ejemplo). Estos webmasters no necesita ningún software especial en sus equipos, pero se puede editar las páginas web de los navegadores. Editores web no necesita tener conocimientos de HTML y otras tecnologías subyacentes, pero en general el uso del WYSIWYG de edición, similar a la de procesamiento de textos en general.

Web EPiServer CMS basado en Microsoft, . NET plataforma. EPiServer socios de personalizar el sistema y construir nuevos módulos a la misma. EPiServer CMS desarrollado por la empresa sueca EPiServer AB, cuya sede se encuentra en Government Street, de Estocolmo .

4.2.1 Que es CMS

Un sistema de gestión de contenidos (en inglés Content Management System, abreviado CMS) es un programa que permite crear una estructura de soporte (framework) para la creación y administración de contenidos, principalmente en páginas web, por parte de los administradores, editores, participantes y demás roles.

Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio web. El sistema permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio web sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al sistema y otro de nivel superior (moderador o administrador) que permite que estos contenidos sean visibles a todo el público (los aprueba).

4.2.2 Características

- **Contenido Dinámico**

El contenido dinámico permite a los editores para añadir "dinámico" a las partes el contenido de la página que, en tiempo de render, obtendrá los datos de otras fuentes. Esas fuentes pueden ser el contenido de otras páginas o información de otros sistemas, tales como datos financieros en un sistema ERP.

EPiServer CMS R2 incluye un adaptador, que permite a los editores para mostrar la información desde cualquier página en el sistema en la página actual. Esto es muy útil si desea volver a utilizar (maestro) de contenido, como texto legal o la información del producto en varias páginas. Si se edita el contenido principal, se cambiará automáticamente en cualquier página que utiliza un contenido dinámico.

EPiServer CMS contenido dinámico también permite a los desarrolladores para crear bloques de funcionalidad en el código, que luego pueden ser utilizados más tarde por un editor de una página en el editor de HTML. Esto significa que se puede adjuntar cualquier tipo de función o de la información y ponerla a disposición para el editor.

Para más información sobre el contenido dinámico se puede encontrar en los siguientes artículos en EPiServer Mundial.

- **Proveedor de página personalizada (Enterprise Edition)**

El proveedor de la página le permite trabajar con cualquier tipo de datos como si se tratara de EPiServer páginas CMS. A menudo lo desea, puede gestionar toda la información la compañía de un punto - el modelo de proveedor de la página lo hace posible.

Mediante la implementación de una interfaz hacia la fuente de datos, puede trabajar con cualquier tipo de datos como si se tratara de EPiServer páginas CMS. Es posible que, por ejemplo, conectar una base de datos existente de artículos nuevos o los registros de productos a partir de una base de datos de producto. Cuando se trabaja en EPiServer CMS que ahora será capaz de utilizar la información como cualquier otro contenido de la página e incluso puede actualizar los datos subyacentes, por ejemplo, artículos de prensa o de los productos. Un XmlPageProvider también se incluye en la instalación de EPiServer CMS R2.

El proveedor de página personalizado sólo está disponible en la versión Enterprise Edition de EPiServer CMS 5.

- **El contenido del canal**

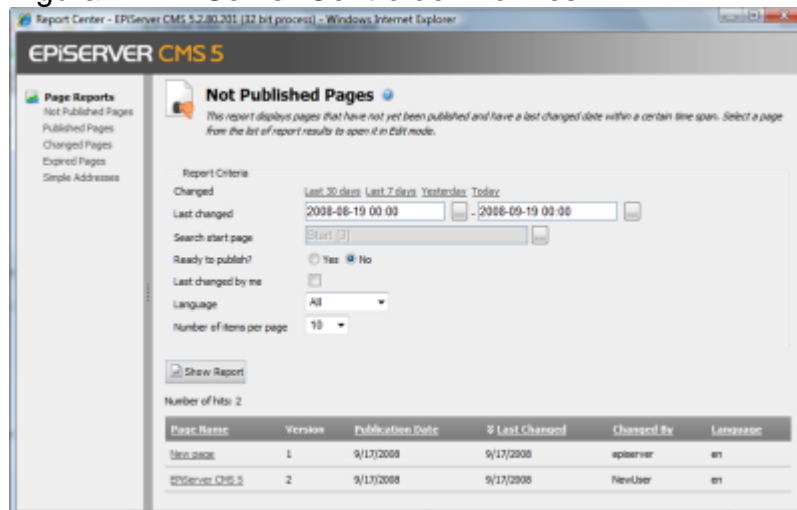
Canal de contenido le permite agregar una gran cantidad de datos externos en EPiServer CMS. Páginas o archivos se crean, eliminan o actualizan en función de una secuencia XML de entrada. Nosotros lo usamos internamente cuando sincronizar los datos de sitios de SharePoint en EPiServer CMS a través de Connect para SharePoint.

Canal de contenido se encuentra disponible desde EPiServer CMS 5 Service Pack1.

- **Centro de informes**

¿Quieres saber qué páginas se han publicado o modificado durante un período determinado de tiempo? ¿Quieres saber cuál de estas páginas han sido modificadas por usted? EPiServer CMS Centro de informes hace que sea fácil para que usted pueda encontrar rápidamente la información sobre su sitio Web para ayudarle con su trabajo diario. EPiServer CMS R2 incluye cinco informes incorporados.

Figura 4. EPiServer Centro de Informes



Fuente: episerver

Los siguientes cinco informes integrados se incluyen en EPiServer R2 CMS.

- **Páginas cambiadas**

Este informe muestra las páginas que han cambiado la fecha dentro de un cierto período de tiempo. Esto hace posible que los editores sigan las páginas que han cambiado y cuándo.

- **Páginas expiradas**

El informe Páginas de vencimiento es una manera fácil de encontrar páginas que sean pronto a punto de ser archivado o con discapacidad. Esto le permite actuar en consecuencia, añadir un poco de tiempo antes de la expiración o modificar el contenido.

- **Las páginas no publicadas**

Este informe muestra las páginas que aún no han sido publicadas y tienen una fecha cambió por última vez en un lapso de tiempo determinado.

- **Páginas publicadas**

Este informe muestra las páginas publicadas en un lapso de tiempo determinado. Usted puede optar por buscar entre las páginas que han publicado o todas las páginas publicadas.

- **Las direcciones simples**

El informe se ocupa de simple es una manera fácil a la vista de todas las direcciones simple que se usa en el sistema. Editores ya no tienen que mirar a través de todas las páginas que posiblemente tenga una dirección sencilla con el fin de encontrar todas las direcciones simples que se utilizan.

- **Mejoras en la redacción**

En EPiServer CMS 5 R2 hay varias características nuevas y mejoradas editorial. Todo para apoyar el trabajo diario del editor y administrador. Cinco de estos merecen especial atención.

- **Editor de imágenes**

Trabajar con imágenes en EPiServer CMS R2 es fácil! El editor de imágenes integrado que permite recortar las imágenes, la escala y el tamaño utilizado en el sitio Web. El editor de imágenes está disponible en el Administrador de archivos y la función Insertar imagen en el editor.

- **Convertir páginas**

En el modo de administración ahora puede convertir las páginas de un tipo de página a otra. Esta funcionalidad puede ser utilizada, por ejemplo, si ha creado las páginas basadas en el tipo "malo" de la página o desea combinar las páginas de distintos tipos en un tipo común.

- **Multi-link de Propiedad**

Ahora es posible guardar una colección de enlaces en una página, no sólo vínculos individuales. Esta es una característica útil, por ejemplo, si desea crear listas de enlaces a páginas relacionadas.

- **Formas de gestión nuevos**

En EPiServer CMS R2 es muy fácil de manejar formas, incluso si su sistema contiene cientos de formas. El editor de formularios ha sido rediseñado y ahora también es compatible con Firefox e Internet Explorer. Algunas características de menor importancia también se han añadido.

Browser mejorado el soporte

EPiServer CMS R2 tiene, en general, el apoyo mejorado para Firefox.

- **Installation Manager**

EPiServer CMS 5 R2 se entrega a través de un Gestor de instalación de nuevo encapsulado dentro de un estándar de MSI-package. Esto significa que ahora puede instalar y desinstalar EPiServer CMS como cualquier otro producto en el servidor. La instalación se divide en dos partes: la instalación de los componentes de EPiServer base de CMS y la instalación de sitios Web. También puede seleccionar un "One Click Install" escenario de lo que es muy fácil y rápido de instalar una primera versión de EPiServer CMS.

El Gestor de instalación es también fundamental para la instalación del producto en el futuro. Se basa en la energía Scripts Shell, que hace que sea fácil para cualquier empresa de terceros para utilizar el mismo instalador para sus productos también.

4.3 MICROSOFT VISUAL STUDIO .NET

Microsoft Visual Studio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión .NET 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles.

4.3.1 Características

El compilador de Visual Basic x.0 genera código que requiere librerías de enlace dinámico DLL para que funcione, en algunos casos llamada MSVBVMxy.DLL (acrónimo de "MicroSoft Visual Basic Virtual Machine x.y", donde x.y es la versión) y en otros VBRUNXXX.DLL ("Visual Basic Runtime X.XX").

Estas DLL proveen las funciones implementadas en el lenguaje, conteniendo rutinas en código ejecutable que son cargadas bajo demanda. Además existe un gran número de bibliotecas DLL, que facilitan el acceso a la mayoría de las funciones del sistema operativo y también la integración con otras aplicaciones.

En el IDE de Visual Basic se puede ejecutar el programa en desarrollo, "al vuelo" o en el modo intérprete (en realidad pseudo-compila el programa muy rápidamente y luego lo ejecuta), y también se permite la generación del programa en código ejecutable (exe).

Tal programa generado en disco puede luego ser ejecutado fuera del ambiente de programación (incluso en modo stand alone, dependiendo de los requisitos de DLL's), aunque será necesario que las librerías DLL requeridas se encuentren instaladas en el sistema para su apropiada ejecución.

Visual Basic provee soporte para empaquetado y distribución, es decir, permite generar un módulo instalador que contiene el programa ejecutable y las bibliotecas DLL necesarias para él. Con ese módulo la aplicación generada se distribuye y puede ser instalada en cualquier equipo (con sistema compatible).

Así como bibliotecas DLL, hay numerosas aplicaciones de terceros que disponen de variadas funciones y mejoras para Visual Basic, incluyendo también para empaquetado y distribución.

4.3.2 Que es C#

C# (pronunciado si sharp en inglés) es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270). C# es uno de los lenguajes de programación diseñados para la infraestructura de lenguaje común.

Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de la plataforma .NET, similar al de Java, aunque incluye mejoras derivadas de otros lenguajes.

El nombre C Sharp fue inspirado por la notación musical, donde '#' (sostenido, en inglés sharp) indica que la nota (C es la nota do en inglés) es un semitono más alta, sugiriendo que C# es superior a C/C++. Además, el signo '#' viene de cuatro '+' pegados.¹

Aunque C# forma parte de la plataforma .NET, ésta es una API, mientras que C# es un lenguaje de programación independiente diseñado para generar programas sobre dicha plataforma. Ya existe un compilador implementado que provee el marco Mono - DotGNU, el cual genera programas para distintas plataformas como Windows, Unix y GNU/Linux.

4.4 ORACLE

Oracle es un sistema de gestión de base de datos objeto-relacional (o ORDBMS por el acrónimo en inglés de Object-Relational Data Base Management System), desarrollado por Oracle Corporation.

Se considera a Oracle como uno de los sistemas de bases de datos más completos, destacando:

- Soporte de transacciones,
- Estabilidad,
- Escalabilidad.
- Soporte multiplataforma.

Su dominio en el mercado de servidores empresariales ha sido casi total hasta hace poco, recientemente sufre la competencia del Microsoft SQL Server de Microsoft y de la oferta de otros RDBMS con licencia libre como PostgreSQL, MySQL o Firebird. Las últimas versiones de Oracle han sido certificadas para poder trabajar bajo GNU/Linux.

4.4.1.2 Inconvenientes

- Una de las versiones más recientes de Oracle es la 8 punto algo (Aunque ya está la 9i). Y es que desde el lanzamiento original de la 8 se sucedieron varias versiones con correcciones, hasta alcanzar la estabilidad en la 8.0.3. El motivo de tantos fallos fue, al parecer, la remodelación del sistema de almacenamiento por causa de la introducción de extensiones orientadas a objetos.
- El mayor inconveniente de Oracle es quizás su precio. Incluso las licencias de Personal Oracle son excesivamente caras, en mi opinión. Otro problema es la necesidad de ajustes. Un error frecuente consiste en pensar que basta instalar el Oracle en un servidor y enchufar directamente las aplicaciones clientes. Un Oracle mal configurado puede ser desesperantemente lento.
- También es elevado el coste de la formación, y sólo últimamente han comenzado a aparecer buenos libros sobre asuntos técnicos distintos de la simple instalación y administración.

4.5 JAVASCRIPT

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos,³ basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador... aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

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

4.5.1 Que es jQuery

jQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC.

jQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privativos.¹ jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

Las empresas Microsoft y Nokia anunciaron que incluirán la biblioteca en sus plataformas.² Microsoft la añadirá en su IDE Visual Studio³ y la usará junto con los frameworks ASP.NET AJAX y ASP.NET MVC, mientras que Nokia los integrará con su plataforma Web Run-Time

4.5.2 Características

Javascript es muy fácil de aprender para quien ya conoce lenguajes similares como el C++ o Java, pero, dada su simplicidad sintáctica y su manejabilidad, no es tampoco difícil para quien se acerca por primera vez a este lenguaje. Sin embargo, esto puede ser un arma de doble filo porque la simplicidad se basa en una disponibilidad de objetos limitada, por lo que algunos procedimientos, aparentemente muy sencillos, requieren script bastante complejos.

La característica principal de Javascript, de hecho, es la de ser un lenguaje de scripting, pero, sobre todo, la de ser el lenguaje de scripting por excelencia y, sin lugar a dudas, el más usado. Esta particularidad conlleva una notable serie de ventajas y desventajas según el uso que se le deba dar y teniendo en cuenta la relación que se establece entre el mecanismo cliente-servidor. Para explicar con

pocas palabras dicha relación, podemos decir que el servidor envía los datos al cliente y estos datos pueden llegar en en dos formatos diversos: en formato texto (o ASCII) o en formato binario (o código máquina: El cliente sabe cómo comprender sólo el formato binario (es decir, la secuencia de 1 y 0), por lo que si los datos llegan en este formato son inmediatamente ejecutables (y, por desgracia, no dejan abierta la posibilidad de efectuar controles), mientras que si el formato es diverso tienen que ser interpretados y traducidos al formato binario y, por tanto, el cliente necesitará un filtro, o mejor dicho, un intérprete que sepa leer estos datos y los pueda traducir al binario. Los datos en formato texto son visibles al usuario como simples combinaciones de caracteres y de palabras y son, por tanto, fáciles de manipular, si bien requieran más tiempo para su interpretación a causa de los distintos pasos y de las transformaciones a las que deben someterse para que el cliente pueda comprenderlos: Los datos en formato binario, sin embargo, son difíciles de comprender por el usuario, pero inmediatamente ejecutables por el cliente ya que no requieren fases intermedias.

Sentada esta premisa, podemos dividir los lenguajes que suelen utilizarse en el web en cuatro tipologías:

- HTML: está en formato texto y no es un lenguaje en el sentido tradicional, sino un empaginador para consentir colocar los objetos en la página con las características indicadas. Naturalmente, por esta peculiaridad, tiene un carácter estático y no interactúa con el usuario ni puede tomar decisiones sino por lo que se refiere a los formularios. Para su interpretación necesita un navegador.
- Lenguajes compilados: son lenguajes bastante complejos en los que la fuente (un file de texto con las operaciones que hay que ejecutar) se compila en código máquina y se empaqueta en un ejecutable que puede utilizarse sólo de la forma y para las operaciones para las que se ha proyectado.
- Lenguajes semicompilados: en realidad, a esta clase pertenece tan sólo Java porque es un lenguaje compilado en un formato intermedio entre los file ASCII y el file binario. Dicho formato se llama bytecode y es interpretado en el cliente por una máquina virtual llamada Java Virtual Machine. De esta manera, a la recepción, dicha máquina completa la compilación y hace que el file sea ejecutable.
- Lenguajes interpretados: son los que resultan muy similares al HTML, pero que tienen mayores potencialidades porque permiten efectuar controles y

operaciones complejas. Se envían en en file ASCII y, por tanto, con código en claro que el navegador interpreta y ejecuta línea por línea en modalidad runtime.

El concepto de script queda bien expresado con una comparación que podemos leer en el texto de Michael Moncur sobre Javascript, recientemente publicado en Italia por Tecniche Nuove. La explicación es muy eficaz y merece citarse: script en inglés significa "guión". De hecho, el uso es exactamente éste: el navegador lee una línea, la interpreta y la ejecuta, después pasa a la sucesiva y hace lo mismo, y así hasta el cierre del script.

4.6 CASCADING STYLE SHEETS

El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets, del que toma sus siglas. CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML1 (y por extensión en XHTML).

El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

La información de estilo puede ser adjuntada como un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "<style".

4.6.1 Características

El modo de funcionamiento de las CSS consiste en definir, mediante una sintaxis especial, la forma de presentación que le aplicaremos a:

- Un web entero, de modo que se puede definir la forma de todo el web de una sola vez.
- Un documento HTML o página, se puede definir la forma, en un pequeño trozo de código en la cabecera, a toda la página.

- Una porción del documento, aplicando estilos visibles en un trozo de la página.
- Una etiqueta en concreto, llegando incluso a poder definir varios estilos diferentes para una sola etiqueta. Esto es muy importante ya que ofrece potencia en nuestra programación. Podemos definir, por ejemplo, varios tipos de párrafos: en rojo, en azul, con márgenes, sin ellos...

La potencia de la tecnología salta a la vista. Pero no solo se queda aquí, ya que además esta sintaxis CSS permite aplicar al documento formato de modo mucho más exacto. Si antes el HTML se nos quedaba corto para maquetar las páginas y teníamos que utilizar trucos para conseguir nuestros efectos, ahora tenemos muchas más herramientas que nos permiten definir esta forma:

- Podemos definir la distancia entre líneas del documento.
- Se puede aplicar indentado a las primeras líneas del párrafo.
- Podemos colocar elementos en la página con mayor precisión, y sin lugar a errores.
- Y mucho más, como definir la visibilidad de los elementos, márgenes, subrayados, tachados...

Y seguimos mostrando ventajas, ya que si con el HTML tan sólo podíamos definir atributos en las páginas con píxeles y porcentajes, ahora podemos definir utilizando muchas más unidades como:

- Píxeles (px) y porcentaje (%), como antes.
- Pulgadas (in)
- Puntos (pt)
- Centímetros (cm)

4.7 FIREBUG

Firebug es una extensión de Firefox creada y diseñada especialmente para desarrolladores y programadores web. Es un paquete de utilidades con el que se puede analizar (revisar velocidad de carga, estructura DOM), editar, monitorizar y depurar el código fuente, CSS, HTML y JavaScript de una página web de manera instantánea e inline.

Firebug no es un simple inspector como DOM Inspector, además edita y permite guardar los cambios, un paso por delante del conocido Web Developer. Su

atractiva e intuitiva interfaz, con solapas específicas para el análisis de cada tipo de elemento (consola, HTML, CSS, Script, DOM y red), permite al usuario un manejo fácil y rápido. Firebug está encapsulado en forma de plug-in o complemento de Mozilla, es Open Source, libre y de distribución gratuita.

Con Firebug es posible editar el código HTML de una página y comprobar el resultado al instante, sin tener que recargar.

Esa misma acción puede realizarse sobre las hojas de cascada en estilo (CSS), sin tener que editar el fichero que las contiene. Los cambios también se verán en vivo.

Firebug incluye una pequeña regla que te ayudará en las tareas de diseño, para que no yerres ningún pixel.

Otras características interesantes de FireBug son:

- Supervisar la actividad de red.
- Depurar el código Javascript.
- Explorar el DOM (Document Object Model).

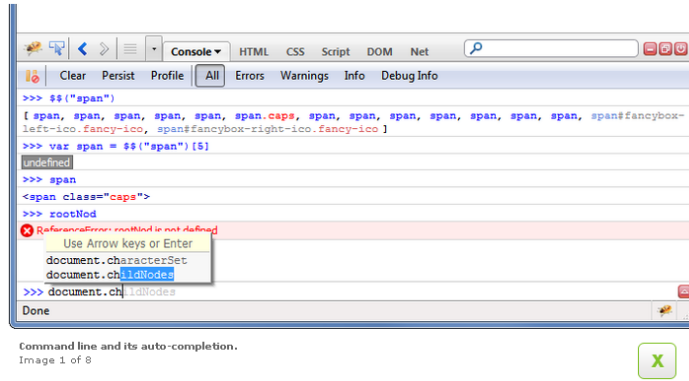
4.7.1 Historia del proyecto

La primera versión del proyecto (versión 0.2) de 21 kb fue lanzada el 12 de enero de 2006. Cinco días más tarde vio la luz la versión 0.2.1 y dos días después la 0.2.2 con una apariencia notablemente mejorada y que incluía una ventana para el inspector más completa. Además mantenía visible el menú del navegador en la consola y mostraba una solapa para inspeccionar el JavaScript mucho más depurada. Durante ese mismo mes se sacaría una versión más, la 0.2.3, que podía ser instalada en Flock (navegador de código abierto basado en tecnologías de Mozilla y programado en torno a Gecko) y Deer Park (versión de prueba del hoy Firefox 1.1).

La versión 0.3 aparece a finales de marzo de 2006 con numerosos cambios, especialmente en la forma en la que trabaja el inspector e incluía novedades como un inspector de eventos, editor DOM y la posibilidad de ver el código fuente en vivo. Luego vendrían las versiones 0.3.1, 0.3.2 (que podía ser instalada en Firefox 2.0 alpha), 0.4 y 0.4.1 (compatible con Firefox 2 y primeros pasos hacia una versión estable en Mac Os X).

La versión 1.0 aparece en enero de 2007, un año después del inicio del proyecto y con un tamaño de 291 kb. En los cuatro primeros meses de este año han salido las versiones 1.01, 1.02, 1.03, 1.04 y 1.05, evolucionando como es lógico en su propio código y usabilidad.

Figura 5. Firebug



Fuente: wikipedia

5. DISEÑO METODOLOGICO

También denominada “material y métodos” o “procedimientos”, el diseño metodológico es la descripción de cómo se va a realizar la investigación, o también;

- La estrategia utilizada para comprobar una hipótesis o un grupo de hipótesis.
- La determinación de las estrategias y procedimientos que servirán para dar respuesta al problema y comprobar hipótesis.
- El plan de acción del investigador para alcanzar los objetivos del mismo.

Al seleccionar y plantear un diseño se busca maximizar la validez y confiabilidad de la información y reducir los errores en los resultados.

5.1 DESCRIPCION DEL PROYECTO

- Renovar el sitio Web, fortaleciendo el posicionamiento de la marca en las usuarias a través de Internet
- Ampliar los servicios que actualmente tiene el sitio
- Ampliar el segmento para llegar a las usuarias de más de 18 años
- Tener un look más actual
- Proporcionar mayor interacción
- Promover la construcción de la comunidad
- Aumentar el número de visitas por usuario

Para dar solución al problema anteriormente descrito se creará un Sitio Web basado en el CMS EPiServer versión Relate + 2, que permita la administración de contenidos, así como la utilización de módulos contenidos en éste y que son necesarios para la solución, tales como: Encuestas, Blogs, Foros, Eventos.

Se deben realizar también integraciones con el CRM que posee Familia, ya que la información de sus negocios (NOL, CDF, Pequeñín, Tena, Familia institucional y Freshkids) se encuentra centralizada. Entre estas integraciones se encuentra el módulo de seguridad, ya que los usuarios se encuentran en dicha fuente unificada,

para esto se debe realizar un componente de seguridad a la medida, ya que EPiServer solo cuenta con 2 opciones: Membership Provider de .Net y LDAP. Las integraciones se realizan con modelo de Capas desde el Sitio web (EPiServer)

5.1.1 Información general

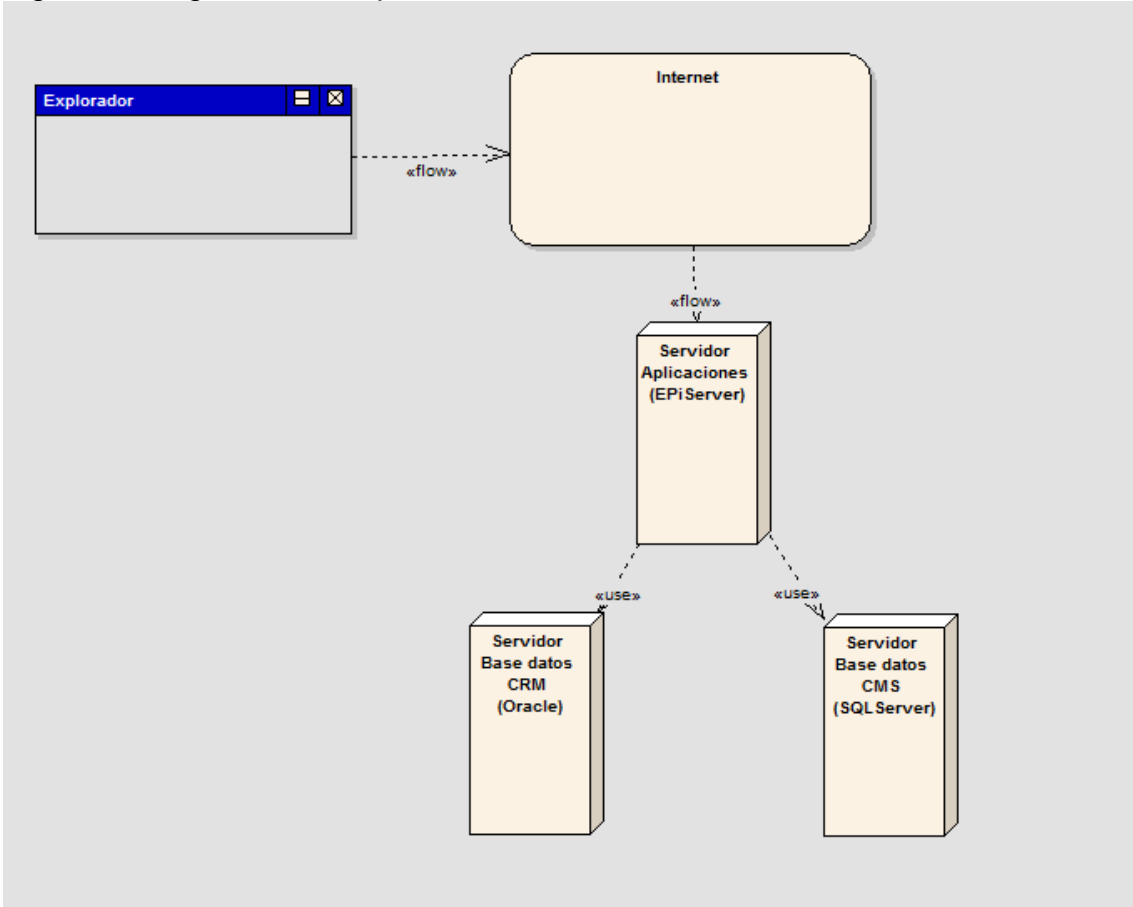
- Los artículos podrán ser publicados en html, flash o video
- Los usuarios podrán suscribirse por RSS para recibir en su mail actualizaciones
- Las usuarias podrán comentar todos los artículos
- Se podrá recomendar y enviar por mail los artículos a las personas que las usuarias consideren
- Los artículos y las opiniones podrán calificarse como “me gusta” o “no me gusta”
- Los artículos serán ilustrados de forma que se combinen ilustraciones gráficas con fotografías reales
- Con el fin de la comercialización a terceros de la página, esta sección deberá tener un espacio importante para la venta de pauta y también para la comunicación de eventos y noticias por parte interna de la marca
- Todos los artículos deberán ser “recomendados” y “comentados” por usuarios del sistema.
- Se tendrá una sub sección en donde las usuarias podrán consultar los artículos que estuvieron en otras ediciones de magazín
- Se tendrá como se usa en sitios web como Amazon, en la parte de abajo un banner en el que se dirá “las personas que leyeron este artículo también visitaron...” con el fin de promocionar una navegación más amplia, no solo de las secciones de Magazín sino de todo el sitio.

5.1.2 Descripción del plan del proyecto

5.1.2.1 Organización del Proyecto

Descripción del sistema (Diagrama de Contexto)

Figura 6. Diagrama Descripción del Sistema



Fuente: wikipedia

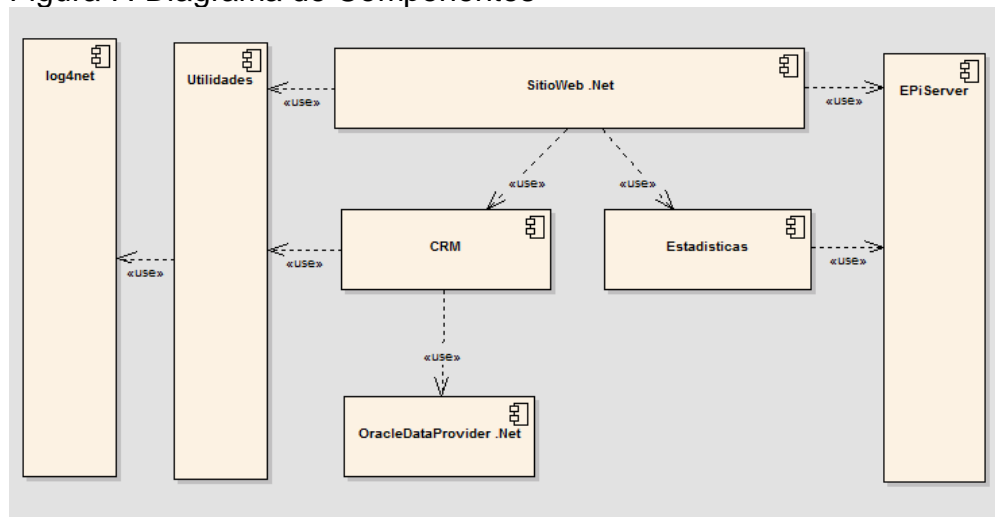
Tabla 1. Descripción del sistema

	Tipo de sistema	Descripción
Servidor Aplicaciones	Plataforma	Sistema donde se encuentra corriendo el sitio web basado en EPiServer
Servidor de Base de datos CRM	Plataforma	Sistema que almacena la Base de datos con el CRM de Familia
Servidor de Base de datos del CMS	Plataforma	Sistema que almacena la Base de datos del CMS.

Vistas Dinámicas

Diagrama de Componentes

Figura 7. Diagrama de Componentes



Fuente: wikipedia

Tabla 2. Vistas dinámicas

	Descripción
Sitio Web .Net	Componente que contiene la lógica de la interfaz del sitio web.
Estadísticas	Contiene la lógica para registrar y obtener información relacionada con el uso de las páginas.
CRM	Contiene la lógica para acceso al CRM de Familia

log4net	Componente para registro de información de bajo de nivel de la aplicación, EPiServer viene integrado con este componente.
EpiServer	Componentes que permiten el acceso a la información del CMS
OracleDataProvider	Componente para acceso a Bases de datos Oracle desde .Net
Utilidades	Contiene las funcionalidades comunes para todos los componentes, como por ejemplo el registro de errores, mensajes de advertencia, etc.

Vistas Estáticas

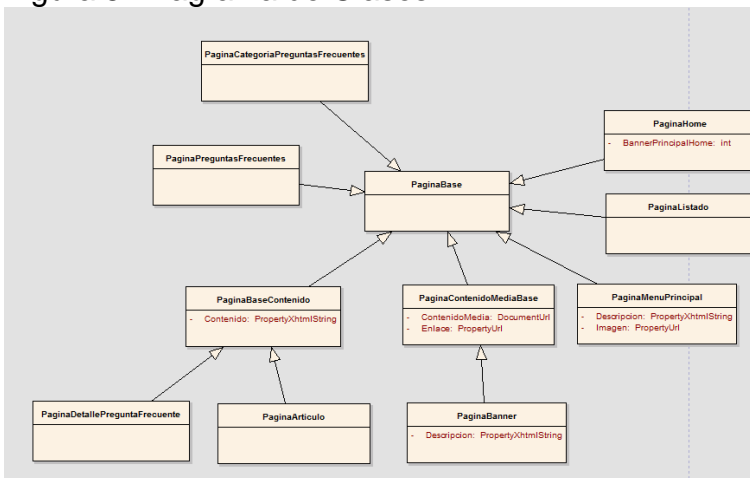
Diagrama de clases

Los diagramas de clases están divididos para cada uno de los componentes con los cuales cuenta el sistema.

Sitio Web. Net

En este componente se cuenta con las plantillas necesarias para el CMS, se describen a continuación las plantillas y sus relaciones (Las plantillas se definen como clases .Net)

Figura 8. Diagrama de Clases



Fuente: wikipedia

5.1.2.2 Recursos

Servidor de aplicaciones:

- Sistema operativo: Alguna de las siguientes opciones:
 - Microsoft Windows Server 2003 SP2 32/64 bit
 - Microsoft Windows Server 2008 SP2 32/64 bit
 - Microsoft Windows Server 2008 R2

- Servidor web: Alguna de las siguientes opciones:
 - Microsoft Internet Information Services (IIS) 6
 - Microsoft Internet Information Services (IIS) 7.0
 - Microsoft Internet Information Services (IIS) 7.5

- Servidor de base de datos CMS: Alguna de las siguientes opciones:
 - Microsoft SQL Server 2005 SP2 32/64 bit
 - Microsoft SQL Server 2008 SP1 32/64 bit
 - Microsoft SQL Server 2008 R2 32/64 bit

- Servidor de base de datos CRM:
 - Oracle 9i

5.2 METODOLOGIA DE TRABAJO TSP/PSP

5.2.1 Team Software Process

Se describirá el TSP, cómo y para quién fue desarrollado, su estructura, una breve explicación de la metodología, resultados de una aplicación real y la versión educativa (TSPi).

5.2.1.1 Objetivos

Es una metodología para dirigir el trabajo de mejora y desarrollo de software además de establecer un entorno donde el trabajo efectivo de equipo sea normal y natural

- Generar un marco basado en PSP
- Desarrollar productos en varios ciclos
- Establecer estándares para medir la calidad y el comportamiento
- Proporcionar métricas para equipos
- Evaluar roles y equipos
- Guías para solución de problemas en equipos.
- Maximizar calidad SW
- Minimizar costos

Figura 9. Objetivo TSP



Fuente: Software Engineering Institute

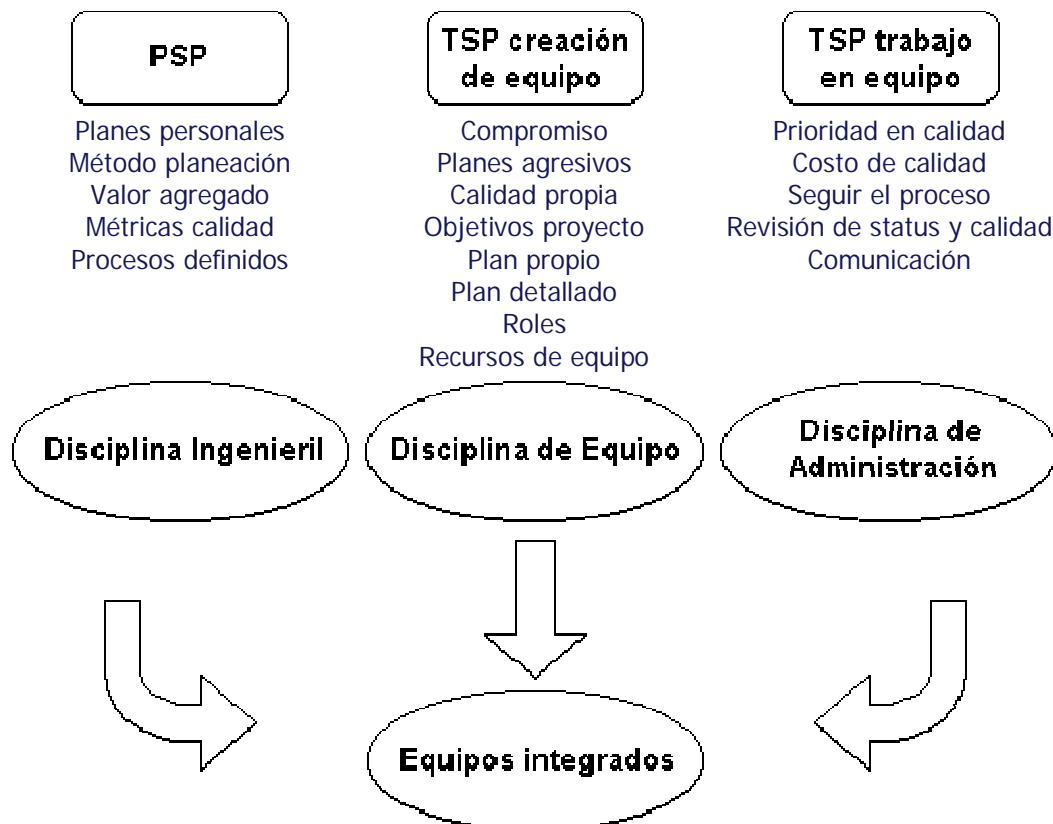
5.2.1.2 Antecedentes

Tsp prosigue las estrategias de calidad americanas que inicio:

- Demming en la industria en 1982.
- Fagan en el proceso de sw 1986.
- W. Humphrey sw, cmm 1987.
- W. Humphrey sw, psp 1995.
- W. Humphrey sw, tsp 1999.

5.2.1.3 Estructura de TSP

Figura 10. Estructura TSP



Fuente: Software Engineering Institute

5.2.1.4 Problemas comunes

- Falta de liderazgo
- Falta de compromiso y ganas de cooperar
- Diferencia en contribuciones
- Falta de confianza
- Falta de calidad
- Mejoras excesivas
- Revisiones entre colegas inefectivas

5.2.1.5 Metodología TSP

Lanzamiento TSP, checklist para planeación

- Establecer productos y objetivos de empresa
- Establecer roles y objetivos de equipo
- Definir estrategia de desarrollo
- Hacer un plan general
- Hacer un plan de calidad
- Balancear el plan (cargas de trabajo)
- Proyecto de riesgos
- Diseñar reporte para administración
- Revisión del plan con administración
- Analisis Postmortem, nuevo equipo revisa proceso

5.2.1.6 Lanzamiento TSP

Programa de reuniones

- Los puntos 1,2,3 seran en el día 1
- Los puntos 4,5,6 seran en el día 2
- Los puntos 7,8 seran en el día 3
- El punto 9 y el análisis postmortem serán en el día 4 o bien al final del día 3

5.2.1.7 Plan de calidad

Se enfatiza en la administración de defectos.

Se basa en los estimados de tamaño e historicos, y estimaran los defectos en cada fase, sino hay historico se basaran en la figura 14.

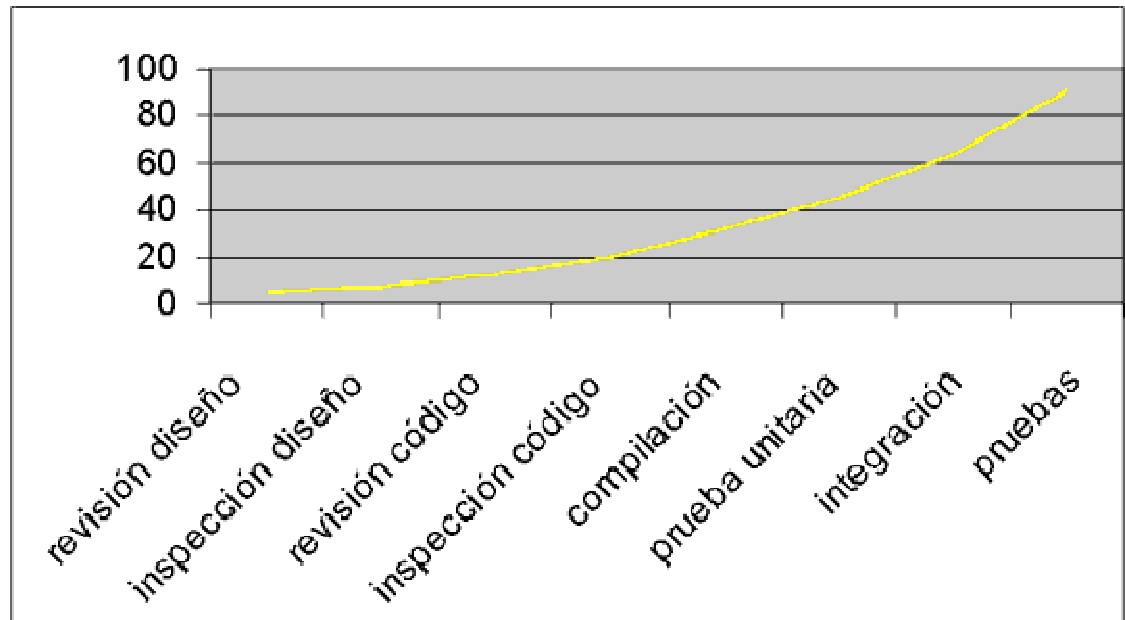
Manejo de la calidad

Ejemplo Plan de Calidad

Nombre: x **Proyecto:** xx **parte:** xxy

Defectos	Plan	Actual
Compilación	140	220
En producto	7	21
Revisión código	23	52

Figura 11. Porcentaje de Defectos Encontrados



Fuente: Software Engineering Institute

5.2.1.8 Ciclo de vida de TSP

- Lanzamiento
- Estrategia
- Planeación
- Requerimientos
- Diseño
- Implementación
- Pruebas
- Postmorten

5.2.2 Personal Software Process

El Personal Software Process, conocido por sus siglas como PSP, es una metodología de reciente creación, proveniente del Instituto de Ingeniería del Software(SEI). PSP es una alternativa dirigida a los ingenieros de sistemas, que les permite mejorar la forma en la que construyen software. Considerando aspectos como la planeación, calidad, estimación de costos y productividad, PSP es una metodología que vale la pena revisar cuando el ingeniero de software está interesado en aumentar la calidad de los productos de software que desarrolla dentro de un contexto de trabajo individual.

5.2.2.1 Características

En PSP todas las tareas y actividades que el ingeniero de software debe realizar durante el proceso de desarrollo de un producto de software, están puntualmente definidas en un conjunto de documentos conocidos como scripts. Los scripts son el punto medular de PSP, por lo que se hace mucho énfasis en que deben ser seguidos en forma disciplinada, ya que de ello dependerá el éxito de la mejora que se busca. Gran parte de las tareas y actividades definidas en los scripts generará en su realización un conjunto de datos, fundamentalmente de carácter estadístico. La aplicación de PSP en varios procesos de desarrollo, y el análisis de la información estadística generada en cada uno de éstos, permitirán al ingeniero de software identificar, tanto sus fortalezas como sus debilidades, y crecer a través de un proceso de autoaprendizaje y auto mejora. La calidad en PSP, es un aspecto fuertemente relacionado con la cantidad de defectos que el producto de software contiene.

En este nivel se introducen algunos métodos aplicables al proceso de desarrollo de software, dentro de un enfoque de proyectos a gran escala, pero sin lidiar con problemas de comunicación y coordinación de los equipos de trabajo.

5.2.2.2 Pasos A Seguir

Los scripts se organizan en cuatro niveles, identificados del 0 al 3, atendiéndose en cada nivel un conjunto de aspectos a mejorar del proceso de desarrollo de software. Al primer nivel se le conoce como 0 o de medición personal, al segundo como nivel 1 o de planeación personal, al tercero, como nivel 2 o de calidad personal, y al cuarto, como nivel 3 o cíclico personal. Cada uno de estos niveles, con excepción del 3, tiene una versión que los extiende, introduciendo tareas y actividades para un mejor manejo de los aspectos de interés en nivel, o bien para incluir nuevos aspectos, verla si la siguiente figura. Cada uno de los niveles extiende los aspectos considerados en el nivel inmediato anterior. Una de las razones de esta clasificación puede ser el que PSP es una metodología de mejora basada en datos estadísticos, los cuales deben ser cuidadosamente recabados por el ingeniero de software; el aumento gradual de la cantidad de datos que debe recolectar el ingeniero introduce, por consiguiente, el cambio en su manera de trabajo de una manera paulatina. Se recomienda un uso incremental de PSP, iniciando con el nivel más bajo durante un primer proyecto de desarrollo y, en proyectos siguientes, ascendiendo a niveles superiores. Los scripts no pueden utilizarse en forma separada o desordenada.

5.2.2.3 Ventajas y Desventajas

PSP es una alternativa, de las muchas que han surgido recientemente, para mejorar el proceso de desarrollo de software. Más que clasificar un conjunto de sentencias como ventajas o desventajas, a continuación se citan algunas características:

- PSP es una metodología basada en estimación. La estimación permite saber cuándo y cómo se desarrollan las tareas de un proceso, por lo que podría citarse como un aspecto importante de esta metodología el estar basada en métricas y estimaciones.
- La información de las métricas y estimaciones se utiliza para evaluar y mejorar procesos futuros. PSP parte de la premisa que, si el ingeniero de software conoce sus fortalezas y debilidades, puede establecer las

acciones necesarias para erradicar o explotar los aspectos identificados en la forma en que desarrolla software.

- Aunque lo mencionado en el punto anterior podría sonar bastante atractivo, la forma de llegar a ese auto conocimiento puede resultar tediosa y, en el peor de los casos, una pesadilla para el desarrollador. Salvo muy pocas excepciones, los ingenieros de software nunca realizan procedimientos formales para conocer la forma en que trabajan, no saben con exactitud cuántas líneas de código generan por hora, cuánto tiempo invierten al corregir un error, cuánto tiempo invierten en pruebas, etcétera.
- Los pasos de registro de información a detalle en el nivel de medición pueden resultar frustrantes cuando se tiene presión de tiempo.
- En los scripts de PSP no se incluyen tareas y actividades para la etapa de análisis de requerimientos. Siempre se parte de una definición de requerimientos que no va a cambiar.
- Aún no existe una herramienta automatizada que facilite el registro y análisis de datos generados por la aplicación de PSP.

5.2.2.4 Ciclo de vida PSP, fases

Figura 12. Ciclo de vida PSP



Fuente: Software Engineering Institute

5.3 LECCIONES APRENDIDAS

- Estructurar mejor los diferentes diagramas existentes para el desarrollo de un software.
- Realizar diferentes tipos de consultas a una base de datos.
- Programación básica en el lenguaje Microsoft Visual Studio CSharp.
- Conocimiento de herramientas como episerver, .net y oracle, entre otras.
- Utilización de la metodología TSP y PSP.

5.4 RESULTADOS

Los resultados del proyecto en curso han sido satisfactorios hemos pasado de metodologías de trabajo basadas en RUP o estándares anteriores del desarrollo tradicional a una metodología que ha innovado el trabajo en equipo e incrementado la experiencia laboral a nivel personal de cada recurso del proyecto, ahora cada integrante de un proyecto está presente en cada una de las etapas y responsable no solo tanto de su labor específica, sino que pasa a ser parte de la gerencia, interfaz con el cliente, modelos de negocio etc. Mediante los roles TSP que se establecen a partir de cada lanzamiento, en cuanto la calidad la mejora es notable, se pasa de inyectar 5 defectos por cada 500 líneas a un estándar de 1 defecto cada 1000 líneas de código probadas, esto es un avance crucial para las entregas oportunas de cada proyecto no solo en los de familia sino en cualquier otro en el cual se requiera un desarrollo TSP, con esta nueva cultura no solo se mejora la experiencia laboral sino también la experiencia personal de ser no solo tratado como un recurso porcentual de un proyecto sino al contrario un eje o engrane de un sistema de seres humanos que tienen responsabilidades, riesgos y metas idénticas.

6. CONCLUSIONES

En Colombia se requiere que más empresas se interesen por la calidad de nuestros productos, porque su gente se involucre en ella y para ello se requiere adoptar una metodología de organización que nos lleve a ella.

Como se dijo, se está adoptando la metodología PSP/TSP para producir software de calidad, necesitamos resultados inmediatos y una buena organización, mucho mejor que la de nuestros competidores.

La implementación de una metodología organizacional permite a los ingenieros hacer mejor su trabajo, ahorrar tiempo y dinero, poder planear su trabajo y ofrecer mejor calidad.

BIBLIOGRAFIA

¿QUÉ ES SHAREPOINT? [On line]. [Citado 20 junio 2010] URL disponible en: <http://office.microsoft.com/es-hn/sharepoint-server-help/que-es-sharepoint-HA010378184.aspx>

ARQUITECTURA DE INFORMACION. Modelo Navegacional [en línea] <http://www.arquitecturadeinformacion.cl/como/mapa.html> [Citado el 15 de Marzo del 2011]

CIUDAD EDUCATIVA. Software de notas Ciudad Educativa [en línea] http://www.ciudadeducativa.com/cms/index.php?option=com_content&view=article&id=25&Itemid=28 [Citado el 15 de Marzo del 2011]

COLOMBIA MINISTERIO DE EDUCACIÓN NACIONAL. Decreto 1290 del 2009 http://www.mineducacion.gov.co/1621/articles187765_archivo_pdf_decreto_1290.pdf [Citado el 15 de Marzo del 2011]

DATOSOFT. Software de notas datosoft [en línea] <http://www.datosoft.com/colegios.htm> [Citado el 15 de Marzo del 2011]

EDUCOLOMBIA. Software de notas educolombia [en línea] www.educolombia.org [Citado el 15 de Marzo del 2011]

INTRODUCCIÓN A ISA SERVER 2006. [On line]. [Citado 20 junio 2010] URL disponible en: <http://www.microsoft.com/spain/isaserver/prodinfo/overview.msp#EYB>

MAS ADELANTE, SIGNIFICADOS DE RED. [on line]. [Citado 20 junio 2010] URL disponible en: <http://www.masadelante.co>

MICROSOFT TECHNET AUTENTICACIÓN NTLM. [on line]. [Citado 20 junio 2010] URL disponible en: [http://technet.microsoft.com/es-es/library/cc783005\(WS.10\).aspx](http://technet.microsoft.com/es-es/library/cc783005(WS.10).aspx)

MICROSOFT. TechNet. Guía detallada de uso de la Consola de administración de Directivas de grupo. [on line]. [Citado 20 junio 2010] URL disponible en: <http://www.microsoft.com/spain/technet/recursos/articulos/gpmcinad.msp>

MICROSOFT MSDN Active Directory Schema Terminology [on line]. [Citado 20 junio 2010] URL disponible en: [http://msdn.microsoft.com/en-us/library/ms675087\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ms675087(v=vs.85).aspx)

PHPNUKE. Navicat [en línea]
<http://downloads.phpnuke.org/es/download-item-view-a-g-m-b-m/NAVICAT%2BFOR%2BMYSQL.htm> [Citado el 15 de Marzo del 2011]

SISTEMAS MICROINFORMÁTICOS . [on line]. [Citado 20 junio 2010] URL disponible en: <http://html.rincondelvago.com/sistemas-microinformaticos.html>

SISTACNET. Software de notas sistacnet [en línea]
< <http://sistacnet.com/> > [Citado el 15 de Marzo del 2011]

SLIDESHARE. Tipos Pruebas [en línea]
< <http://www.slideshare.net/aracelij/pruebas-de-software> > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Diagramas de casos de uso [en línea]
< http://es.wikipedia.org/wiki/Diagrama_de_casos_de_uso > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Diagramas de clases[en línea]
< http://es.wikipedia.org/wiki/Diagrama_de_clases > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Firebug [en línea]
< <http://es.wikipedia.org/wiki/Firebug> > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Mantis [en línea]
< http://es.wikipedia.org/wiki/Mantis_Bug_Tracker > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Metodología RUP [en línea]
< http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Migración de Datos, [en línea]
< http://es.wikipedia.org/wiki/Migraci%C3%B3n_de_datos > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Modelo MVC [en línea]
< http://es.wikipedia.org/wiki/Modelo_Vista_Controlador > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Mysql [en línea]
< <http://es.wikipedia.org/wiki/MySQL> > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Php [en línea]< <http://es.wikipedia.org/wiki/PHP> > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Phpmyadmin [en línea]
< <http://es.wikipedia.org/wiki/PhpMyAdmin> > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Pruebas [en línea]
< http://es.wikipedia.org/wiki/Pruebas_de_software > [Citado el 15 de Marzo del 2011]

WIKIPEDIA. Help Desk, [en línea]
< http://es.wikipedia.org/wiki/Help_desk > [Citado el 15 de Marzo del 2011]

ANEXOS

ANEXO A. Cronograma de Actividades

Plan de Trabajo Personal				
Fase	Tarea	Horas	Fecha	Semana
REQ	CDF Home I1 REQ	4,0	22/08/2011	1
DLD	Banner contacto Detailed Design	1,5	22/08/2011	1
DLDR	Banner contacto DLD Review	0,8	22/08/2011	1
TD	Banner contacto Test Development	0,2	22/08/2011	1
CODE	Banner contacto Code	1,4	22/08/2011	1
CR	Banner contacto Code Review	0,7	22/08/2011	1
COMPILE	Banner contacto Compile	0,0	22/08/2011	1
UT	Banner contacto Unit Test	1,1	22/08/2011	1
DLD	Campos Buscador Detailed Design	0,7	22/08/2011	1
DLDR	Campos Buscador DLD Review	0,3	22/08/2011	1
TD	Campos Buscador Test Development	0,1	22/08/2011	1
CODE	Campos Buscador Code	0,6	22/08/2011	1
CR	Campos Buscador Code Review	0,3	22/08/2011	1
COMPILE	Campos Buscador Compile	0,0	22/08/2011	1
UT	Campos Buscador Unit Test	0,5	22/08/2011	1
DLDINSP	Pata Facebook DLD Inspection	1,3	22/08/2011	1
CODEINSP	Pata Facebook Code Inspection	1,3	22/08/2011	1
DLD	Registro Paso 2 Detailed Design	6,2	22/08/2011	1
DLDR	Registro Paso 2 DLD Review	3,1	29/08/2011	2
TD	Registro Paso 2 Test Development	1,0	29/08/2011	2
CODE	Registro Paso 2 Code	5,6	29/08/2011	2
CR	Registro Paso 2 Code Review	2,8	29/08/2011	2
COMPILE	Registro Paso 2 Compile	0,0	29/08/2011	2
UT	Registro Paso 2 Unit Test	4,4	29/08/2011	2
DLD	Registro Paso 3 Detailed Design	3,3	29/08/2011	2
DLDR	Registro Paso 3 DLD Review	1,7	29/08/2011	2
TD	Registro Paso 3 Test Development	0,5	29/08/2011	2
CODE	Registro Paso 3 Code	3,0	05/09/2011	3
CR	Registro Paso 3 Code Review	1,5	05/09/2011	3
COMPILE	Registro Paso 3 Compile	0,0	05/09/2011	3
UT	Registro Paso 3 Unit Test	2,4	05/09/2011	3

IT	Prueba de Intergracion I1 CDF	3,0	05/09/2011	3
REQ	Para Nosotras I4 Parte 1 REQ	8,0	05/09/2011	3
REQINSP	NOL Diseño REQINSP	3,4	05/09/2011	3
DLDINSP	ASPX Preguntar Consultorio DLD Inspection	1,3	05/09/2011	3
CODEINSP	ASPX Preguntar Consultorio Code Inspection	2,6	12/09/2011	4
DLD	ASPX Mensaje Enviado Consultorio Detailed Design	2,0	12/09/2011	4
DLDR	ASPX Mensaje Enviado Consultorio DLD Review	1,0	12/09/2011	4
TD	ASPX Mensaje Enviado Consultorio Test Development	0,3	12/09/2011	4
CODE	ASPX Mensaje Enviado Consultorio Code	1,8	12/09/2011	4
CR	ASPX Mensaje Enviado Consultorio Code Review	0,9	12/09/2011	4
COMPILE	ASPX Mensaje Enviado Consultorio Compile	0,0	12/09/2011	4
UT	ASPX Mensaje Enviado Consultorio Unit Test	1,4	12/09/2011	4
DLD	ASPX Listado Mias Consultorio Detailed Design	2,0	12/09/2011	4
DLDR	ASPX Listado Mias Consultorio DLD Review	1,0	12/09/2011	4
TD	ASPX Listado Mias Consultorio Test Development	0,3	12/09/2011	4
CODE	ASPX Listado Mias Consultorio Code	1,8	12/09/2011	4
CR	ASPX Listado Mias Consultorio Code Review	0,9	12/09/2011	4
COMPILE	ASPX Listado Mias Consultorio Compile	0,0	12/09/2011	4
UT	ASPX Listado Mias Consultorio Unit Test	1,4	12/09/2011	4
DLD	ASPX Interna Producto Detailed Design	3,3	12/09/2011	4
DLDR	ASPX Interna Producto DLD Review	1,7	12/09/2011	4
TD	ASPX Interna Producto Test Development	0,5	12/09/2011	4
CODE	ASPX Interna Producto Code	3,0	19/09/2011	5
CR	ASPX Interna Producto Code Review	1,5	19/09/2011	5
COMPILE	ASPX Interna Producto Compile	0,0	19/09/2011	5
UT	ASPX Interna Producto Unit Test	2,4	19/09/2011	5
DLD	ASPX Lo Nuevo Detailed Design	0,7	19/09/2011	5
DLDR	ASPX Lo Nuevo DLD Review	0,3	19/09/2011	5
TD	ASPX Lo Nuevo Test Development	0,1	19/09/2011	5
CODE	ASPX Lo Nuevo Code	0,6	19/09/2011	5
CR	ASPX Lo Nuevo Code Review	0,3	19/09/2011	5
COMPILE	ASPX Lo Nuevo Compile	0,0	19/09/2011	5
UT	ASPX Lo Nuevo Unit Test	0,5	19/09/2011	5
DLD	ASPX Interna Girls Detailed Design	2,0	19/09/2011	5
DLDR	ASPX Interna Girls DLD Review	1,0	19/09/2011	5
TD	ASPX Interna Girls Test Development	0,3	19/09/2011	5
CODE	ASPX Interna Girls Code	1,8	19/09/2011	5
CR	ASPX Interna Girls Code Review	0,9	19/09/2011	5
COMPILE	ASPX Interna Girls Compile	0,0	19/09/2011	5
UT	ASPX Interna Girls Unit Test	1,4	19/09/2011	5
DLD	ASPX Listado Descargas Detailed Design	4,4	19/09/2011	5

DLDR	ASPX Listado Descargas DLD Review	2,2	19/09/2011	5
TD	ASPX Listado Descargas Test Development	0,7	26/09/2011	6
CODE	ASPX Listado Descargas Code	4,0	26/09/2011	6
CR	ASPX Listado Descargas Code Review	2,0	26/09/2011	6
COMPILE	ASPX Listado Descargas Compile	0,0	26/09/2011	6
UT	ASPX Listado Descargas Unit Test	3,2	26/09/2011	6
IT	Prueba de integracion I1-i2-i3-i4-i5-i6NOL	3,0	26/09/2011	6
DLD	Cargar Foto Detailed Design	4,0	26/09/2011	6
DLDR	Cargar Foto DLD Review	2,0	26/09/2011	6
TD	Cargar Foto Test Development	0,6	26/09/2011	6
CODE	Cargar Foto Code	3,6	03/10/2011	7
CR	Cargar Foto Code Review	1,8	03/10/2011	7
COMPILE	Cargar Foto Compile	0,0	03/10/2011	7
UT	Cargar Foto Unit Test	2,8	03/10/2011	7
DLDINSP	Datos Personales DLD Inspection	1,6	03/10/2011	7
CODEINSP	Datos Personales Code Inspection	1,6	03/10/2011	7
DLD	Descripcion de la Familia - Edición Detailed Design	3,1	03/10/2011	7
DLDR	Descripcion de la Familia - Edición DLD Review	1,6	03/10/2011	7
TD	Descripcion de la Familia - Edición Test Development	0,5	03/10/2011	7
CODE	Descripcion de la Familia - Edición Code	2,8	03/10/2011	7
CR	Descripcion de la Familia - Edición Code Review	1,4	03/10/2011	7
COMPILE	Descripcion de la Familia - Edición Compile	0,0	03/10/2011	7
UT	Descripcion de la Familia - Edición Unit Test	2,2	03/10/2011	7
DLDINSP	Banner Como Se mide el ranking DLD Inspection	0,6	03/10/2011	7
CODEINSP	Banner Como Se mide el ranking Code Inspection	0,6	03/10/2011	7
DLDINSP	Banner Beneficios del mes DLD Inspection	0,6	03/10/2011	7
CODEINSP	Banner Beneficios del mes Code Inspection	0,6	03/10/2011	7
DLD	Banner Actividades Para subir Ranking Detailed Design	4,0	10/10/2011	8
DLDR	Banner Actividades Para subir Ranking DLD Review	2,0	10/10/2011	8
TD	Banner Actividades Para subir Ranking Test Development	0,6	10/10/2011	8
CODE	Banner Actividades Para subir Ranking Code	3,6	10/10/2011	8
CR	Banner Actividades Para subir Ranking Code Review	1,8	10/10/2011	8
COMPILE	Banner Actividades Para subir Ranking Compile	0,0	10/10/2011	8
UT	Banner Actividades Para subir Ranking Unit Test	2,8	10/10/2011	8
DLD	Banner Producto Favorito Detailed Design	1,4	10/10/2011	8
DLDR	Banner Producto Favorito DLD Review	0,7	10/10/2011	8
TD	Banner Producto Favorito Test Development	0,2	10/10/2011	8
CODE	Banner Producto Favorito Code	1,3	10/10/2011	8
CR	Banner Producto Favorito Code Review	0,7	10/10/2011	8
COMPILE	Banner Producto Favorito Compile	0,0	10/10/2011	8
UT	Banner Producto Favorito Unit Test	1,0	10/10/2011	8

DLDINSP	Banner Lo Más DLD Inspection	4,9	17/10/2011	9
CODEINSP	Banner Lo Más Code Inspection	4,9	17/10/2011	9
DLD	Banner Cosas de casa Detailed Design	4,0	17/10/2011	9
DLDR	Banner Cosas de casa DLD Review	2,0	17/10/2011	9
TD	Banner Cosas de casa Test Development	0,6	17/10/2011	9
CODE	Banner Cosas de casa Code	3,6	17/10/2011	9
CR	Banner Cosas de casa Code Review	1,8	17/10/2011	9
COMPILE	Banner Cosas de casa Compile	0,0	17/10/2011	9
UT	Banner Cosas de casa Unit Test	2,8	17/10/2011	9
DLD	Banner Sección Rotador Detailed Design	5,3	24/10/2011	10
DLDR	Banner Sección Rotador DLD Review	2,7	24/10/2011	10
TD	Banner Sección Rotador Test Development	0,8	24/10/2011	10
CODE	Banner Sección Rotador Code	4,8	24/10/2011	10
CR	Banner Sección Rotador Code Review	2,4	24/10/2011	10
COMPILE	Banner Sección Rotador Compile	0,0	24/10/2011	10
UT	Banner Sección Rotador Unit Test	3,8	24/10/2011	10
DLDINSP	Banner Participa y Gana DLD Inspection	1,6	24/10/2011	10
CODEINSP	Banner Participa y Gana Code Inspection	1,6	31/10/2011	11
DLD	Ultimos Articulos Creados Detailed Design	4,0	31/10/2011	11
DLDR	Ultimos Articulos Creados DLD Review	2,0	31/10/2011	11
TD	Ultimos Articulos Creados Test Development	0,6	31/10/2011	11
CODE	Ultimos Articulos Creados Code	3,6	31/10/2011	11
CR	Ultimos Articulos Creados Code Review	1,8	31/10/2011	11
COMPILE	Ultimos Articulos Creados Compile	0,0	31/10/2011	11
UT	Ultimos Articulos Creados Unit Test	2,8	31/10/2011	11
DLD	ASPX Tags Detailed Design	0,7	31/10/2011	11
DLDR	ASPX Tags DLD Review	0,3	31/10/2011	11
TD	ASPX Tags Test Development	0,1	31/10/2011	11
CODE	ASPX Tags Code	0,6	31/10/2011	11
CR	ASPX Tags Code Review	0,3	31/10/2011	11
COMPILE	ASPX Tags Compile	0,0	31/10/2011	11
UT	ASPX Tags Unit Test	0,5	31/10/2011	11
DLD	Buscador Receta Detailed Design	8,8	07/11/2011	12
DLDR	Buscador Receta DLD Review	4,4	07/11/2011	12
TD	Buscador Receta Test Development	1,4	07/11/2011	12
CODE	Buscador Receta Code	8,0	07/11/2011	12
CR	Buscador Receta Code Review	4,0	07/11/2011	12
COMPILE	Buscador Receta Compile	0,0	07/11/2011	12
UT	Buscador Receta Unit Test	6,3	14/11/2011	13
DLDINSP	Banner rotador DLD Inspection	2,1	14/11/2011	13
CODEINSP	Banner rotador Code Inspection	2,1	14/11/2011	13

DLDINSP	En Esta Seccion DLD Inspection	1,6	14/11/2011	13
CODEINSP	En Esta Seccion Code Inspection	1,6	14/11/2011	13
DLD	Control tabulador de contenido de Famipuntos Detailed Design	6,2	14/11/2011	13
DLDR	Control tabulador de contenido de Famipuntos DLD Review	3,1	14/11/2011	13
TD	Control tabulador de contenido de Famipuntos Test Development	1,0	21/11/2011	14
CODE	Control tabulador de contenido de Famipuntos Code	5,6	21/11/2011	14
CR	Control tabulador de contenido de Famipuntos Code Review	2,8	21/11/2011	14
COMPILE	Control tabulador de contenido de Famipuntos Compile	0,0	21/11/2011	14
UT	Control tabulador de contenido de Famipuntos Unit Test	4,4	21/11/2011	14
DLD	Banner Destacado de Receta Categoria Detailed Design	4,0	21/11/2011	14
DLDR	Banner Destacado de Receta Categoria DLD Review	2,0	21/11/2011	14
TD	Banner Destacado de Receta Categoria Test Development	0,6	21/11/2011	14
CODE	Banner Destacado de Receta Categoria Code	3,6	28/11/2011	15
CR	Banner Destacado de Receta Categoria Code Review	1,8	28/11/2011	15
COMPILE	Banner Destacado de Receta Categoria Compile	0,0	28/11/2011	15
UT	Banner Destacado de Receta Categoria Unit Test	2,8	28/11/2011	15
DLD	Recetas creadas por los usuarios Detailed Design	4,0	28/11/2011	15
DLDR	Recetas creadas por los usuarios DLD Review	2,0	28/11/2011	15
TD	Recetas creadas por los usuarios Test Development	0,6	28/11/2011	15
CODE	Recetas creadas por los usuarios Code	3,6	28/11/2011	15
CR	Recetas creadas por los usuarios Code Review	1,8	28/11/2011	15
COMPILE	Recetas creadas por los usuarios Compile	0,0	28/11/2011	15
UT	Recetas creadas por los usuarios Unit Test	2,8	28/11/2011	15
DLDINSP	Comentarios Recientes en Recetas DLD Inspection	1,6	28/11/2011	15
CODEINSP	Comentarios Recientes en Recetas Code Inspection	1,6	05/12/2011	16
DLD	Catalogo de productos Detailed Design	6,2	05/12/2011	16
DLDR	Catalogo de productos DLD Review	3,1	05/12/2011	16
TD	Catalogo de productos Test Development	1,0	05/12/2011	16
CODE	Catalogo de productos Code	5,6	05/12/2011	16
CR	Catalogo de productos Code Review	2,8	05/12/2011	16
COMPILE	Catalogo de productos Compile	0,0	05/12/2011	16
UT	Catalogo de productos Unit Test	4,4	12/12/2011	17
REQ	FI Nuestros productos I2 REQ	10,0	12/12/2011	17
DLDINSP	Global - Contáctenos DLD Inspection	2,0	12/12/2011	17
CODEINSP	Global - Contáctenos Code Inspection	2,0	12/12/2011	17
DLDINSP	Global - General DLD Inspection	0,6	12/12/2011	17
CODEINSP	Global - General Code Inspection	0,6	12/12/2011	17
DLD	Home - Encuestas Detailed Design	4,0	12/12/2011	17
DLDR	Home - Encuestas DLD Review	2,0	19/12/2011	18
TD	Home - Encuestas Test Development	0,6	19/12/2011	18
CODE	Home - Encuestas Code	3,6	19/12/2011	18

CR	Home - Encuestas Code Review	1,8	19/12/2011	18
COMPILE	Home - Encuestas Compile	0,0	19/12/2011	18
UT	Home - Encuestas Unit Test	2,8	19/12/2011	18
DLD	FI Detalle de Producto Detailed Design	11,7	19/12/2011	18
DLDR	FI Detalle de Producto DLD Review	5,9	26/12/2011	19
TD	FI Detalle de Producto Test Development	1,8	26/12/2011	19
CODE	FI Detalle de Producto Code	10,6	26/12/2011	19
CR	FI Detalle de Producto Code Review	5,3	26/12/2011	19
COMPILE	FI Detalle de Producto Compile	0,0	26/12/2011	19
UT	FI Detalle de Producto Unit Test	8,4	02/01/2012	20
IT	Prueba de integracion i1 i2 FI	3,0	02/01/2012	20
DLDINSP	Disminución del riesgo de contaminación DLD Inspection	0,3	02/01/2012	20
CODEINSP	Disminución del riesgo de contaminación Code Inspection	0,3	02/01/2012	20
DLD	Responsabilidad con la comunidad Detailed Design	6,2	02/01/2012	20
DLDR	Responsabilidad con la comunidad DLD Review	3,1	02/01/2012	20
TD	Responsabilidad con la comunidad Test Development	1,0	02/01/2012	20
CODE	Responsabilidad con la comunidad Code	5,6	09/01/2012	21
CR	Responsabilidad con la comunidad Code Review	2,8	09/01/2012	21
COMPILE	Responsabilidad con la comunidad Compile	0,0	09/01/2012	21
UT	Responsabilidad con la comunidad Unit Test	4,4	09/01/2012	21
DLDINSP	Responsabilidad con el medio ambiente DLD Inspection	0,3	09/01/2012	21
CODEINSP	Responsabilidad con el medio ambiente Code Inspection	0,3	09/01/2012	21
DLD	Certificados Detailed Design	0,7	09/01/2012	21
DLDR	Certificados DLD Review	0,3	09/01/2012	21
TD	Certificados Test Development	0,1	09/01/2012	21
CODE	Certificados Code	0,6	09/01/2012	21
CR	Certificados Code Review	0,3	09/01/2012	21
COMPILE	Certificados Compile	0,0	09/01/2012	21
UT	Certificados Unit Test	0,5	09/01/2012	21
DLD	Manuales de Uso Detailed Design	0,7	09/01/2012	21
DLDR	Manuales de Uso DLD Review	0,3	09/01/2012	21
TD	Manuales de Uso Test Development	0,1	09/01/2012	21
CODE	Manuales de Uso Code	0,6	09/01/2012	21
CR	Manuales de Uso Code Review	0,3	09/01/2012	21
COMPILE	Manuales de Uso Compile	0,0	09/01/2012	21
UT	Manuales de Uso Unit Test	0,5	09/01/2012	21
ST	NOL Plan de pruebas ST	4,0	09/01/2012	21
ST	CDF Plan de pruebas ST	4,0	16/01/2012	22
ST	FI Plan de pruebas ST	4,0	16/01/2012	22
DOC	NOL Manual Usuario DOC	12,0	16/01/2012	22
DOC	CDF Manual Usuario DOC	12,0	23/01/2012	23

DOC	FI Manual Usuario DOC	12,0	23/01/2012	23
PM	SYSTEM Postmortem	2,0	30/01/2012	24
		517,54		