

Universidad de las Ciencias Informáticas

Facultad 1



**Título: Módulo para el análisis estadístico del resultado
de las evaluaciones.**

**Trabajo de diploma para optar por el título de Ingeniero en
Ciencias Informáticas**

Autores: Cadete Oraldo Jacinto Simón

Cadete Javier Rodríguez López

Tutor: Tte. Ing. Yadir Martínez Vergara

Ciudad de La Habana, 2010.

Año 52 de la Revolución

DECLARACIÓN DE AUTORÍA

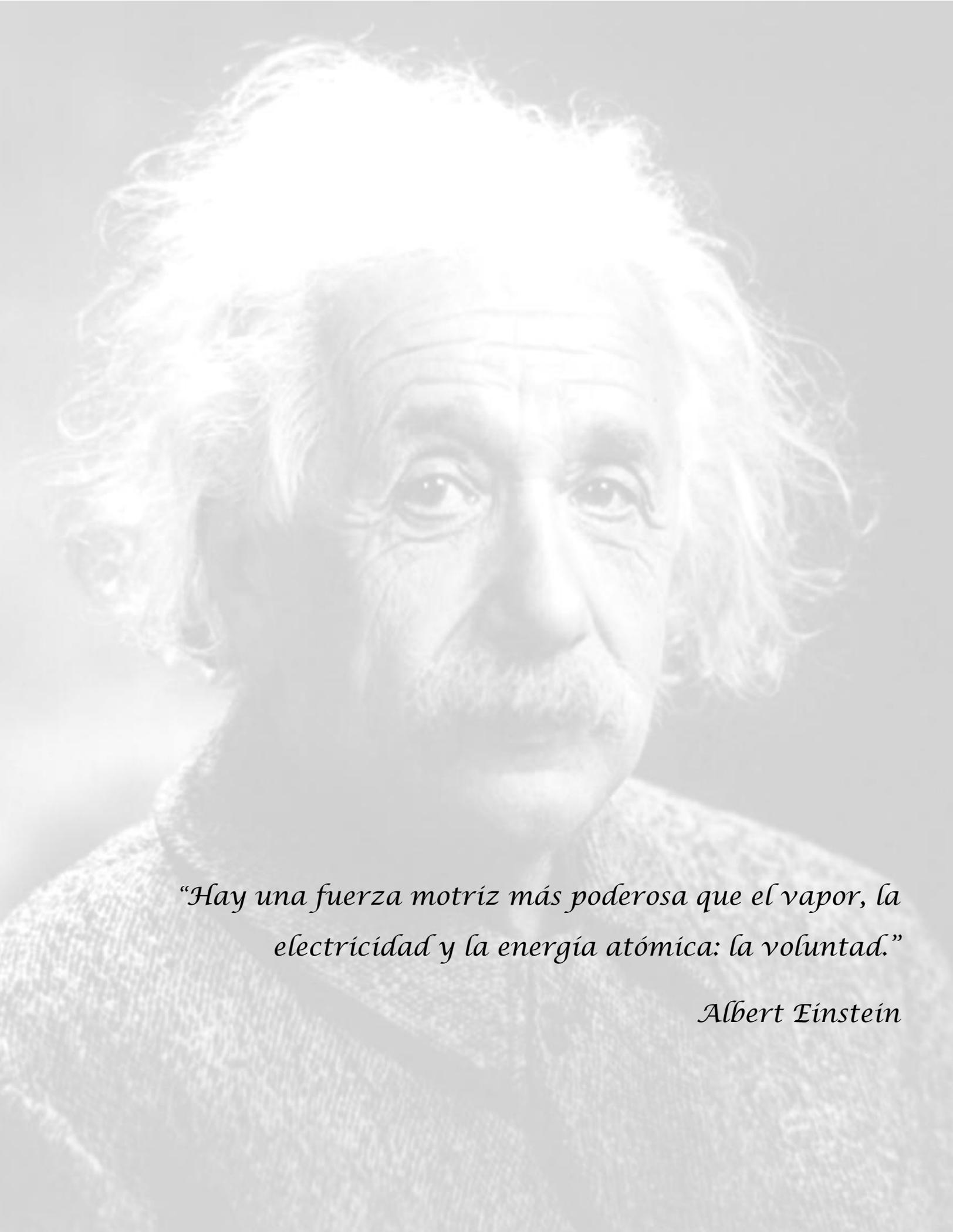
Declaramos ser autores del presente trabajo y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Autor Cdte. Javier Rodríguez López

Autor Cdte. Oraldo Jacinto Simón

Tutor Tte. Ing. Yadir Martínez Vergara



“Hay una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad.”

Albert Einstein

DATOS DE CONTACTO

Tutor: Tte. Ing. Yadir Martinez Vergara

UCID, Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: yvergara@uci.cu

AGRADECIMIENTOS

Con la realización de este trabajo quisiéramos agradecer:

A la Revolución, a Fidel y a la UCI por darnos la oportunidad de formar parte de este proyecto futuro y formarnos como profesionales.

A nuestro tutor el Tte. Ing. Yadir Martínez Vergara por el apoyo brindado.

A Heidy y Greybi por su dedicación y amor incondicional.

A nuestros padres, familiares y amistades que de una forma u otra nos ayudaron para que este sueño se hiciera realidad.

DEDICATORIA

Oraldo

Este trabajo va dedicado con mucho amor y cariño a mis padres por haberme guiado sabiamente, sin su amor, dedicación y apoyo no hubiese sido posible, todo lo que he logrado en la vida se los debo a ustedes los quiero.

A mis tíos por ayudarme en momentos difíciles, a mis abuelos, a mis hermanos y demás familiares, a mis amistades, a mis queridos suegros, a toda la gente que me aprecia y han estado presentes en mi vida.

Javier

Ante todo a mis padres queridos que siempre me han apoyado en cualquier decisión que he tomado aún estando en desacuerdo, a mi hermana, a mis abuelos que me han consentido tanto, a mi tío que ha sido guía y ejemplo a seguir. A mi tía Dania y el Chino por haberme ayudado todos estos cinco años, a mi novia por tener paciencia y entenderme, en fin a toda mi familia.

A los viejos amigos que vienen conmigo desde los camilitos y a todos los nuevos que he conocido aquí en la UCI, gracias por su apoyo.

RESUMEN

En la actualidad la gestión del talento humano es una de las áreas que ha tenido más cambios y transformaciones, de tal forma que hoy en día la visión del área es totalmente diferente de cuando recibía el nombre de Administración de Recursos Humanos (ARH). En muchas entidades se le denomina gestión de personas, administración del capital intelectual, gestión de socios o colaboradores, gestión del talento humano o gestión del capital humano que es el nombre al que se hará referencia en este contexto para que sea de mayor entendimiento.

Las Tecnologías de la Información y las Comunicaciones (TIC) para la gestión del capital humano se han convertido en una operación necesaria para los profesionales de los recursos humanos. Actualmente en Cuba, dichos profesionales manejan grandes volúmenes de información empleando gran cantidad de tiempo y esfuerzo en el análisis de la misma. Todo el contenido se encuentra en formato duro, lo que contribuye al posible deterioro de la información almacenada. Dicha razón contribuyó a que el país tomara la decisión de incluir en el proceso de informatización, la gestión del capital humano, donde desempeña un papel fundamental la gestión de evaluaciones, de ahí la idea de crear una herramienta que permita el análisis de las evaluaciones.

Para el desarrollo de esta idea se decide desarrollar un módulo que contribuya al análisis estadístico del resultado de las evaluaciones durante el proceso evaluativo, para ello se elaboró el estado del arte sobre sistemas informáticos que analicen el resultado de las evaluaciones. Se modeló el proceso de negocio “Analizar resultado de la evaluación”. Se realizó el diseño, implementación y validación de la solución.

Palabras claves: capital humano, evaluación del desempeño, análisis estadístico.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	1
1.1. Introducción	1
1.2. Evaluación del desempeño	1
1.2.1. Proceso de evaluación del Ministerio de Trabajo y Seguridad Social (MTSS).....	2
1.2.2. Proceso de evaluación de los cuadros de las FAR.....	3
1.3. Estado del arte	4
1.3.1. Sistema informático IOn	4
1.3.2. QTraining software para evaluación del desempeño y control de capacitación.....	5
1.3.3. EVAL 360.....	5
1.3.4. Sistema de Registro y Control de Cuadros (SRCC).....	5
1.3.5. Sistema Informativo Atención de los Cuadros de las FAR (SIAC).....	6
1.4. Herramientas, tecnologías y metodologías.....	6
1.4.1. Servidor web.....	7
1.4.2. Lenguajes de programación y tecnologías	9
1.4.3. Sistema gestor de base de datos (SGBD)	13
1.4.4. Frameworks y librerías.....	17
1.4.5. Metodología de desarrollo de software	19
1.4.6. Herramientas CASE	23
1.4.7. Herramientas de desarrollo.....	24
1.5. Conclusiones	27
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	28
2.1. Introducción	28
2.2. Situación problemática.....	28
2.3. Descripción general de la propuesta de sistema	28
2.4. Mapa de procesos.....	29
2.5. Descripción de procesos de negocio	30
2.5.1. Definir reglamento de evaluación del desempeño	31
2.5.2. Definir proceso evaluativo.....	31
2.5.2.1. Realizar evaluación del desempeño	31
2.5.3. Objeto de automatización	31
2.5.3.1. Analizar resultado de la evaluación.....	31
2.5.4. Descripción del proceso “Analizar resultado de la evaluación”.....	32
2.6. Modelo conceptual	36
2.7. Requerimientos funcionales (RF).....	36

2.8. Requerimientos no funcionales	38
2.9. Conclusiones	39
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA	40
3.1. Introducción	40
3.2. Patrones de arquitectura y diseño	40
3.2.1. Patrón Modelo Vista Controlador (MVC).....	40
3.2.2. Patrones generales de software para asignar responsabilidades (GRASP).....	41
3.2.2.1. Patrón Controlador	41
3.2.2.2. Patrón Experto	42
3.2.2.3. Patrón Creador	42
3.2.2.4. Patrón de Alta Cohesión	43
3.3. Diagramas de clases de diseño	43
3.3.1. Diagrama de clases genérico	43
3.3.2. Diagramas específicos	44
3.4. Diagramas de interacción (Secuencia)	46
3.5. Diseño de la base de datos (BD)	47
3.5.1. Diagrama entidad relación de la BD.....	47
3.6. Conclusiones	49
CAPÍTULO 4. IMPLEMENTACIÓN Y PRUEBA	50
4.1. Introducción	50
4.2. Diagrama de componentes	50
4.3. Matriz de integración de componentes interna.....	51
4.4. Prototipos de interfaz de usuario	52
4.5. Mapa de navegación.....	54
4.6. Validación del sistema	54
4.6.1. Pruebas de caja negra.....	54
4.6.2. Diseño de casos de prueba.....	54
4.6.2.1. Diseño del caso de prueba Adicionar modelo matemático.....	54
4.7. Conclusiones	60
CONCLUSIONES	61
RECOMENDACIONES.....	62
BIBLIOGRAFÍA	63
REFERENCIA BIBLIOGRÁFICA.....	64
GLOSARIO DE TÉRMINOS	67

Introducción

En la actualidad el mundo cambia con una rapidez increíble y el uso de las TIC, ha sido uno de los factores que desde su explosión en los años 90 ha permitido que este fenómeno sea cada vez mayor. Las TIC han sido de gran ayuda para el desarrollo de la sociedad, elevando la calidad de vida de las personas y su nivel de productividad. Favorecen la cooperación y colaboración entre distintas entidades, provocan el surgimiento de nuevas profesiones y mercados, además la Internet como herramienta estándar de comunicación, permiten un acceso igualitario a la información y al conocimiento. Dentro del desarrollo de las tecnologías informáticas y las comunicaciones, la informatización de los procesos se ha convertido en una solución que conlleva al éxito de cualquier empresa o institución.

La informatización de los procesos y el uso de las TIC, en general, desempeñan un papel importante en la mejora del sistema empresarial ya que permiten un incremento de los ingresos con respecto a lo planificado. Agregan un valor a las actividades operacionales y de gestión empresarial permitiendo a las empresas obtener ventajas competitivas, permanecer en el mercado y centrarse en su negocio.

A raíz de todos los beneficios que trae consigo el uso de las TIC, el gobierno cubano está aprovechando este recurso con el objetivo de informatizar el país para perfeccionar y agilizar los procedimientos de sus instituciones. La disminución de la importación de software es un factor clave para elevar la economía, permitiendo la reducción del presupuesto destinado a este fin y que este pueda ser aprovechado en otras funciones. La Industria Cubana del Software está llamada a convertirse en una significativa fuente de ingresos para la nación, como resultado del correcto aprovechamiento de las ventajas del capital humano disponible. Desarrollar este tipo de industria se ha convertido en una de las premisas de la Revolución con el objetivo de no solo reducir las importaciones de software, sino, de ser capaz de alcanzar la soberanía e independencia tecnológica.

Actualmente Cuba cuenta con diversas entidades para potenciar la producción de software, entre las que se encuentra la empresa nacional de software Desoft S.A, Albet Ingeniería y Sistemas S.A, esta última es una empresa cubana, cuyo origen y desarrollo se vincula estrechamente a la Universidad de las Ciencias Informáticas (UCI). La UCI está estructurada en facultades docentes las cuales poseen centros de desarrollo de software, donde la fuerza de trabajo de la universidad está conformada por profesionales,

estudiantes y profesores. Uno de los centros de desarrollo es Centro de Informatización de la Gestión de Entidades (CEIGE), el cual está inmerso en el desarrollo de diferentes proyectos. CEDRUX es uno de los productos que está actualmente en desarrollo, el cual cuenta entre sus subsistemas con Capital Humano.

La gestión del capital humano no es más que la ARH o gestión del talento humano que es como se le conoce en la actualidad. La mayoría de las organizaciones consideran su gestión del talento humano como elementos valiosos y en algunos casos irremplazables. Esto se debe a que la gestión de los factores humanos, se convierte en un asunto estratégico y crucial, ya que estos están estrechamente vinculados con los resultados que puede obtener una organización.

El procedimiento para evaluar el recurso humano se denomina evaluación del desempeño y el análisis del resultado de las evaluaciones del desempeño proporciona a la directiva, la información necesaria sobre la eficacia del trabajo que están efectuando sus subordinados para la organización.

El análisis de los resultados de las evaluaciones de desempeño en las entidades del país es un trabajo engorroso, lo requiere de tiempo y esfuerzo excesivo de los especialistas de recursos humanos, lo cual dificulta la oportuna toma de decisiones de los jefes e impide a los especialistas la utilización de métodos matemáticos que descubran las potencialidades, fortalezas y características de los trabajadores que puedan ser aprovechadas en otras áreas de la entidad. Durante todo el proceso se generan grandes volúmenes de información que se encuentran en formato duro y se analizan de forma manual lo que puede conducir a la pérdida de evaluaciones y al posible deterioro de las mismas.

A raíz de la situación anteriormente expuesta se presenta el siguiente **problema a resolver**:

¿Cómo contribuir a mejorar el análisis estadístico del resultado de las evaluaciones durante el proceso evaluativo en las entidades del país?

Por lo que el **objeto de estudio** del presente trabajo es el proceso de evaluación del desempeño.

Derivándose que el **campo de acción** es el análisis estadístico del resultado de las evaluaciones.

Presentando como **idea a defender**: Si se desarrolla un módulo que permita el análisis estadístico del resultado de las evaluaciones entonces se contribuirá al proceso de análisis de las evaluaciones de desempeño durante la gestión del capital humano en las entidades del país.

Para dar solución al problema existente se ha tomado como **objetivo general**:

Modelar e implementar el módulo de análisis estadístico de los resultados de las evaluaciones durante el proceso evaluativo.

Para cumplir el objetivo general del trabajo se trazaron los siguientes **objetivos específicos**:

1. Elaborar el estado del arte.
2. Modelar el proceso de negocio “Analizar resultado de la evaluación”.
3. Capturar, describir y validar los requisitos funcionales y no funcionales del sistema.
4. Realizar el diseño de la solución.
5. Implementar el módulo de análisis estadístico del resultado de las evaluaciones.
6. Validar el módulo implementado.

El presente trabajo está estructurado en cuatro capítulos, tal y como se describe a continuación:

Capítulo 1. Fundamentación Teórica.

Se realiza un estudio de los principales procesos de evaluación que se llevan a cabo en el país centrándose en el subproceso “Análisis del resultado de las evaluaciones”, y del estado del arte a nivel nacional e internacional y se hace referencia a la metodología de desarrollo de software y las herramientas seleccionadas para el desarrollo del trabajo.

Capítulo 2. Características del sistema.

Se muestran todas las características que tendrá el sistema, partiendo de la descripción general de la propuesta del mismo. Se realizan el mapa de procesos, la descripción y modelado de los mismos, se identifican y se describen los requerimientos funcionales y no funcionales.

Capítulo 3. Análisis y diseño del sistema

Los artefactos que se ven vinculados en la construcción de la aplicación Web, son modelados en este capítulo, entre los cuales se encuentran: el modelo de diseño, mediante diagramas de clases del diseño con estereotipos web y diagramas de secuencia, el diagrama de componentes y el modelo de datos; además, se definen los patrones de diseño a utilizar.

Capítulo 4. Implementación y prueba

Se traducen los elementos del diseño a implementación. Se describe cómo los elementos del modelo del diseño se implementan en términos de componentes, se presentan algunas pantallas de la aplicación, el mapa de navegación y se valida el sistema.

Capítulo 1: Fundamentación teórica

1.1. Introducción

En el presente capítulo se define el concepto de evaluación del desempeño y en aras de comprender como se desarrolla un proceso de evaluación en el país, se describen los procesos evaluativos de los cuadros de las Fuerzas Armadas Revolucionarias (FAR) y del Ministerio de Trabajo y Seguridad Social (MTSS), haciendo énfasis en el proceso “Análisis del resultado de las evaluaciones”. Se realiza un estudio del estado del arte a nivel nacional e internacional de los sistemas informáticos que solucionen dicho subproceso y además de las tendencias, técnicas, tecnologías, metodologías de desarrollo de software y herramientas que se utilizarán en la solución al problema.

1.2. Evaluación del desempeño

Todo sistema de gestión del capital humano tanto en Cuba como en el mundo gira entorno a las competencias laborales y está integrado por procesos que se complementan, los cuales se resumen en la siguiente ilustración:

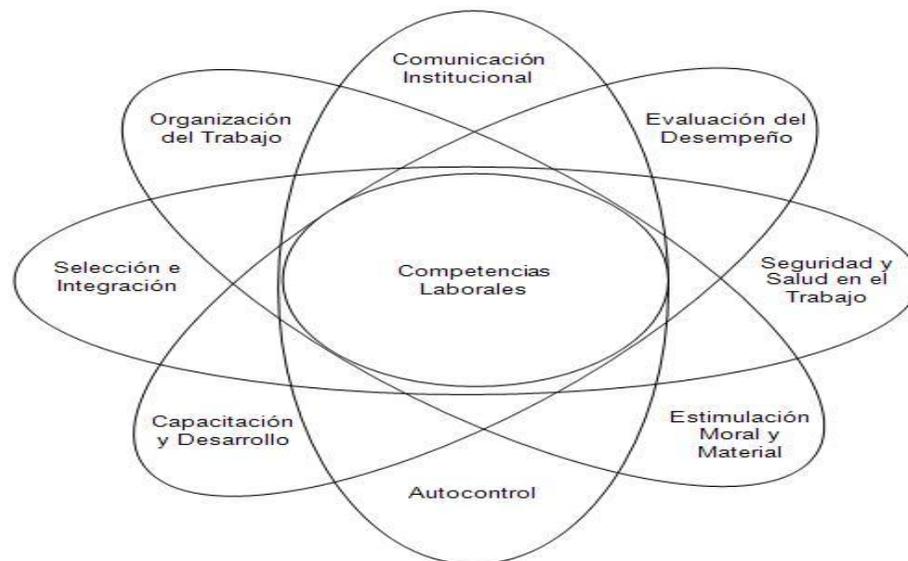


Figura 1.1 Competencias laborales.

La evaluación del desempeño es un proceso que representa un papel importante dentro de la gestión del capital humano y se desarrolla en todas las instituciones del país, debido a que es:

“La medición sistemática e integral del grado de eficacia y eficiencia de la labor realizada por los trabajadores en la actividad que ejecutan, durante un período de tiempo y de su potencial de desarrollo en el ámbito de la empresa, identifica los tipos de insuficiencias y problemas del personal evaluado, sus fortalezas, posibilidades y capacidades, los caracteriza y constituye la base para la elaboración del plan de desarrollo, acorde a las necesidades. Repercute sobre la permanencia, promoción, formación, reconocimiento moral, idoneidad.” [1]

1.2.1. Proceso de evaluación del Ministerio de Trabajo y Seguridad Social (MTSS)

El reglamento de evaluación del desempeño de los trabajadores se elaborará en cada entidad de común acuerdo, con la organización sindical correspondiente a ese nivel, o por la máxima jefatura del centro donde se definen los indicadores para evaluar el desempeño según el marco de trabajo y los objetivos establecidos, así como la definición de la periodicidad que tendrá la evaluación, la custodia de las evaluaciones, las personas facultadas para ejecutar la evaluación y las acciones a tomar como resultado del proceso evaluativo para erradicar las deficiencias detectadas. La elaboración de las planillas de evaluación de cada institución se regirá por dicho reglamento definido para las mismas.

La periodicidad con que se efectúa la evaluación del desempeño es convenida entre la administración y la organización sindical, pudiendo efectuarse cortes parciales de los resultados alcanzados por el trabajador antes de la conclusión de la evaluación. La realización de los cortes parciales le permite a la administración de la entidad laboral, conocer los avances y emprender acciones con respecto a las deficiencias del trabajador, antes de la conclusión del proceso de evaluación. Para la conclusión de la evaluación anual se utilizan los términos de desempeño laboral superior, adecuado y deficiente.

El análisis del resultado del proceso evaluativo tiene como elemento fundamental, la elaboración de un documento que contenga las recomendaciones derivadas de la evaluación, en el cual se reflejan todas las acciones que este debe cumplir en el próximo período con vista a mejorar su desempeño, incluidas las acciones de capacitación y desarrollo individual del trabajador, precisando fechas de cumplimiento para

cada una de ellas, además las valoraciones relacionadas con los indicadores y el resultado conclusivo de la evaluación.

La administración debe llevar un registro en el que aparecen los avances en el desempeño del trabajador y las deficiencias que aún subsisten, así como las recomendaciones que se le realizan. El resultado de la evaluación del desempeño en los cortes parciales y anuales debe analizarse y discutirse con el trabajador.

1.2.2. Proceso de evaluación de los cuadros de las FAR

La evaluación de los cuadros es el subsistema determinante dentro del Sistema de Trabajo con los Cuadros, ya que está presente en todos los subsistemas como el elemento base para la toma de cualquier decisión sobre la prestación del servicio por los oficiales y suboficiales de las FAR.

Las evaluaciones se realizan a todas las categorías de personal (cuadros civiles, oficiales, camilitos, cadetes y alumnos.) de las FAR y a todos los niveles. En el caso de los oficiales existen dos modelos de evaluación, el integral que se realiza cada 3 años y el parcial que se realiza todos los años. No obstante, estos modelos pueden ser modificados en el tiempo. Cada jefe es el encargado de evaluar a los cuadros que le están directamente subordinados, para lo cual puede crear comisiones de análisis de la evaluación.

Los cuadros deben ser evaluados periódicamente y por diversas causas (propuesta de movimiento, tiempo de estancia en el mismo cargo, cumplimiento del período de instrucción). Después de realizadas las evaluaciones, se crea el plan de desarrollo prospectivo y los diferentes tipos de listados de candidatos que deben ser aprobados por el jefe de la nomenclatura del cargo en el caso del listado de candidatos a cargo (cargo, estímulo, ascenso, curso, reserva, prórroga). [2]

El análisis del resultado del proceso evaluativo tiene como elemento fundamental obtener el Informe Resumen del Proceso Evaluativo, este se elabora en dos partes, la primera será elaborada por la Comisión de Evaluación y en ella se describirá el proceso, se procesará la información, se hará un resumen estadístico, y se elaborarán los listados de cuadros con diferentes propuestas resultantes de la evaluación. La segunda parte del Informe estará a cargo de los órganos de Cuadros (o la propia comisión para los niveles Batallón, Compañía y Pelotón) y su contenido fundamental son las valoraciones sobre los

resultados del proceso evaluativo, elaborándose los proyectos de documentos para puntualizar la estrategia en la aplicación de la política de cuadros.

El Informe en su totalidad será analizado por la Comisión de Cuadros (si existe a ese nivel), pasando a la aprobación del jefe de la unidad.

Se confeccionan dos ejemplares y una vez aprobado el informe por la Comisión de Cuadros y firmado por el presidente de la misma, se envía un ejemplar al mando superior.

Teniendo en cuenta como se lleva a cabo el análisis del resultado de las evaluaciones en las FAR y el MTSS se modelará un proceso genérico para las entidades del país.

1.3. Estado del arte

Tras la búsqueda de aplicaciones informáticas que permitan darle solución al problema a resolver, se encontraron las siguientes herramientas que se enfocan en la automatización del proceso de evaluación y desarrollo del Capital Humano permitiendo realizar un análisis del resultado de la evaluación del desempeño.

1.3.1. Sistema informático IOn

Es un software para la Gestión Estratégica del Capital Humano, desarrollado por Cahuala Consultores, en conjunto con la empresa Lempert S.A. IOn proporciona una forma de medir, controlar, gestionar y mejorar la Inteligencia Organizacional de las empresas y permite: incorporar el modelo de gestión por competencias, integrar la administración por objetivos involucrando a todos los niveles de la organización, automatiza el proceso de evaluación de desempeño conjugando la evaluación tradicional con la evaluación 360° y utiliza como fundamentos los resultados obtenidos en las diferentes evaluaciones de desempeño para diseñar actividades orientadas a la mejora del rendimiento de las personas. IOn permite efectuar el proceso de selección on-line, los postulantes pueden completar solicitudes de incorporación al proceso con toda la información necesaria para la empresa. [3]

Está desarrollado 100% en código Java, lo que permite ser utilizado en diferentes plataformas pero tiene un gran inconveniente, es un software privativo y eso traería consigo varias desventajas como: restricciones en el uso, imposibilidad de distribución y sería una opción muy costosa.

1.3.2. QTraining software para evaluación del desempeño y control de capacitación

QTraining es una herramienta para evaluar el desempeño del personal, detectar necesidades de capacitación y administrar los cursos de capacitación de la empresa. Permite evaluar el impacto de las capacitaciones llevadas a cabo y relaciona los objetivos de desempeño con los cursos de capacitación.

Con QTraining queda registrada toda la información de las evaluaciones de desempeño y los cursos de capacitación. Esta información se puede consultar y filtrar en cualquier momento a través de un potente generador de consultas avanzadas que le permite construir prácticamente cualquier consulta sobre la información registrada en la base de datos. Está concebida para una red y fue elaborada por el Grupo Wilsoft. [4]

No es un software libre y su adquisición es muy costosa además se dependería de la empresa propietaria al 100%.

1.3.3. EVAL 360

Fue desarrollado por IO Group Chile Canadá y consiste en una plataforma que permite evaluar desempeño en línea con los objetivos y metas trazadas de los diferentes colaboradores (principio básico de toda evaluación del desempeño), además calcular el cumplimiento de estas mismas a lo largo del año, mostrando gráficamente sus avances, ingresar el plan para el año siguiente y nuevamente medirlo. [5]

Esta herramienta es privativa y eso conllevaría a un costo mayor, además de que las restricciones marcadas por la licencia impedirían adaptar la aplicación a las necesidades existentes.

1.3.4. Sistema de Registro y Control de Cuadros (SRCC)

SRCC es un software para registrar y controlar toda la información pertinente a los cuadros o directivos de cualquier entidad, fue desarrollado en la Unidad de Compatibilización, Integración y Desarrollo de Software para la Defensa (UCID) por profesionales del mismo. Está desarrollado en PHP como lenguaje

de programación de lado del servidor y para lograr una interfaz gráfica agradable, utiliza la librería Ext JS, como gestor de base de datos utiliza PostgreSQL. Cuenta con varios módulos entre los que se encuentra el de evaluación, pero este solo ofrece la posibilidad de registrar, mostrar y filtrar las evaluaciones, pero no permite la configuración de los requisitos que debe tener el proceso evaluativo, ni realizar un análisis estadístico del resultado de dicho proceso.

1.3.5. Sistema Informativo Atención de los Cuadros de las FAR (SIAC)

Este software fue desarrollado por un grupo de profesionales del Ministerio de las Fuerzas Armadas Revolucionarias (MINFAR). Es una herramienta automatizada que permite un mayor nivel de eficiencia, disponibilidad y seguridad de la información referente a los cuadros, garantizando a la vez que los jefes de todos los niveles puedan realizar los procesos de recopilación, análisis y tramitación de la información de manera más fácil. Cuenta con un subsistema para la automatización de las evaluaciones de los cuadros que encierra todas las categorías de personal. [6]

Con dicho sistema, es posible hacer recuperaciones del resumen del proceso evaluativo pero no permite representar gráficamente dichos resultados. Este sistema solo tiene como objetivo cumplir las necesidades de las FAR y no es aplicable a las entidades civiles del país.

Es una aplicación de escritorio desarrollado en Delphi y usa como sistema gestor de bases de datos a SQL Server.

A modo de conclusión del estudio del estado del arte, se puede decir que estos productos cumplen con los objetivos de diseño para los que fueron desarrollados, pero debido a las características de Cuba no se puede pagar la licencia de estos productos, debido al bloqueo económico impuesto al país, además de que no cumplen con algunos de los requisitos del sistema que se desea desarrollar.

1.4. Herramientas, tecnologías y metodologías

A continuación se describen las herramientas, metodologías y tecnologías actuales usadas para dar solución al problema planteado teniendo en cuenta las necesidades existentes y el entorno donde se desplegará el sistema. Las tecnologías usadas fueron tomadas de la arquitectura del proyecto ERP-Cuba.

1.4.1. Servidor web

Un servidor web es un programa que corre sobre el servidor que escucha las peticiones HTTP que le llegan y las satisface. Dependiendo del tipo de la petición, el servidor web buscará una página web o bien ejecutará un programa en el servidor. De cualquier modo, siempre devolverá algún tipo de resultado HTML al cliente o navegador que realizó la petición. El servidor web va a ser fundamental en el desarrollo de las aplicaciones del lado del servidor que se construyan, ya que se ejecutarán en él.

1.4.1.1. Apache

Apache es el servidor web por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. La licencia Apache es una descendiente de la licencia BSD, no es GPL. Esta licencia te permite hacer lo que quieras con el código fuente siempre y cuando se les reconozcas su trabajo. [10]

El Servidor Apache HTTP es un servidor Web de tecnología Open Source sólido y para uso comercial desarrollado por la Apache Software Foundation.

Algunas de sus características:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Trabaja con varios lenguajes de script tales como Perl, PHP y otros además funciona con Java y páginas jsp. Ofreciendo todo el soporte que se necesita para tener páginas dinámicas.
- Permite personalizar las respuestas ante los posibles errores que puedan ocurrir en el servidor.
- Ofrece una alta configurabilidad en la creación y gestión de logs.
- Apache es un servidor altamente configurable de diseño modular. Resulta muy sencillo ampliar las capacidades del servidor Web Apache.

1.4.1.2. IIS

IIS (Internet Information Server), es una serie de servicios para los ordenadores que funcionan con Windows. Los servicios que ofrece son: FTP, SMTP, NNTP y HTTP/HTTPS. Este servicio convierte a un

ordenador en un servidor de Internet o Intranet es decir que en las computadoras que tienen este servicio instalado se pueden publicar páginas web tanto locales como remotamente (servidor web).

El servidor web se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas, por ejemplo Microsoft incluye los de Active Server Pages (ASP) y ASP.NET. También pueden ser incluidos los de otros fabricantes, como PHP o Perl. [11]

Desventajas:

- Software privativo.
- Solo corre sobre plataforma Windows

1.4.1.3. Thttpd

Thttpd es un servidor web de código libre disponible para la mayoría de las variantes de Unix. Se caracteriza por ser simple, pequeño, portátil, rápido, y seguro, ya que utiliza los requerimientos mínimos de un servidor HTTP. Esto lo hace ideal para servir grandes volúmenes de información estática. [12]

Ventajas:

- Simple, ya que sólo maneja el mínimo necesario para poner en práctica el protocolo HTTP, algunas veces un poco más que el mínimo.
- Pequeño, porque también tiene un pequeño tamaño de período de explotación, ya que esto no se divide en dos partes y es muy cuidadoso sobre la asignación de memoria.
- Portátil, pues se compila limpiamente sobre la mayoría de sistemas operativos, expresamente incluyendo FreeBSD, SunOS 4, Solaris 2, BSD/OS, GNU/Linux, OSF.
- Rápido, porque en el empleo típico es sobre todo más rápido que los mejores servidores "destacados" (Apache), y bajo la carga extrema es mucho más rápido.
- Seguro, porque se extiende a grandes longitudes para proteger el servidor web contra ataques de otros sitios.
- El administrador puede decidir restringir la transferencia de archivos de imagen JPEG a en la mayor parte de 20 KB por segundo.

- Los promedios de carga se caen debido a la reducción de la transferencia gráfica.

Desventajas:

- No posee las mismas aplicaciones que se pueden obtener de un software estándar como lo es el Apache.

1.4.1.4. Fundamentación de la selección del servidor web

Se decide utilizar el servidor web Apache por ser multiplataforma, lo que lo convierte en un servidor prácticamente universal, a diferencia de su homólogo el IIS que solo es desarrollado para Windows. Apache es una tecnología gratuita de código fuente abierto, esto le da una transparencia a este software de manera que si se desea ver qué es lo que se está instalando como servidor, se puede conocer, sin ningún secreto, sin ninguna puerta trasera; y por último es altamente configurable de diseño modular por lo que es muy sencillo ampliar sus capacidades.

1.4.2. Lenguajes de programación y tecnologías

1.4.2.1. HTML (Hypertext Markup Language)

HTML acrónimo en inglés de Hypertext Markup Language es el lenguaje predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido que contendrá la página web. El HTML es un lenguaje de marcas que basa su sintaxis en un elemento de base al que se llama etiqueta, a través de estas se van describiendo los elementos del documento, como enlaces, párrafos e imágenes. [13]

El HTML 4 desarrolla el lenguaje HTML con mecanismos para hojas de estilo, ejecución de scripts, marcos, objetos incluidos, soporte mejorado para texto de derecha a izquierda y direcciones mezcladas, tablas más ricas y mejoras en los formularios, ofreciendo mejoras de accesibilidad para personas con discapacidades.

1.4.2.2. XML (Extensible Markup Language)

XML es un Lenguaje de Etiquetado Extensible muy simple, pero estricto que juega un papel fundamental en el intercambio de una gran variedad de datos. Es un lenguaje muy similar a HTML pero su función principal es describir datos y no mostrarlos como es el caso de HTML. XML es un formato que permite la lectura de datos a través de diferentes aplicaciones.

Las tecnologías XML son un conjunto de módulos que ofrecen servicios útiles a las demandas más frecuentes por parte de los usuarios. [14]

XML sirve para estructurar, almacenar e intercambiar información. Este propone una forma estándar de organizar la información, independiente de la plataforma y lenguaje de programación.

1.4.2.3. JavaScript

JavaScript es un lenguaje de programación web que se utiliza principalmente para crear páginas web dinámicas. Este es un lenguaje de programación interpretado por lo que no es necesario compilar los programas para ejecutarlos, su sintaxis es similar al del lenguaje Java, a pesar de su nombre no guarda ninguna relación directa con este lenguaje. Legalmente, JavaScript es una marca registrada de la empresa Sun Microsystems, como se puede ver en: <http://www.sun.com/suntrademarks/>. [15]

Los navegadores modernos interpretan el código JavaScript integrado dentro de las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del DOM.

1.4.2.4. CSS (Cascade Style Sheet)

CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación. [16]

Es un mecanismo simple para añadir estilos (ej., color, tipos de letras, formas) a documentos web.

Esto trae como ventaja la accesibilidad del documento, reduce la complejidad de su mantenimiento y permite visualizar el mismo documento en infinidad de dispositivos diferentes haciendo que su uso sea estándar para la mayoría de los navegadores.

1.4.2.5. AJAX (Asynchronous JavaScript and XML)

AJAX no es una tecnología en sí mismo. En realidad, se trata de varias tecnologías independientes que se unen para lograr cosas realmente impresionantes. AJAX, en resumen, es el acrónimo para Asynchronous JavaScript + XML y el concepto es: Cargar y renderizar una página, luego mantenerse en esa página mientras scripts y rutinas van al servidor buscando, en background, los datos que son usados para actualizar la página solo renderizando la página y mostrando u ocultando porciones de la misma. De esta forma, es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

Las peticiones HTTP al servidor se sustituyen por peticiones JavaScript que se realizan al elemento encargado de AJAX mientras que el acceso a los datos se realiza mediante XMLHttpRequest, objeto disponible en los navegadores actuales. En este caso, la interacción del usuario tampoco se ve interrumpida por recargas de página o largas esperas por la respuesta del servidor. [17]

Las tecnologías que forman AJAX son:

- XHTML y CSS, para crear una presentación basada en estándares.
- DOM, para la interacción y manipulación dinámica de la presentación.
- XML, XSLT y JSON, para el intercambio y la manipulación de información.
- XMLHttpRequest, para el intercambio asíncrono de información.
- JavaScript, para unir todas las demás tecnologías.

1.4.2.6. PHP (Hypertext Preprocessor)

PHP es un acrónimo recursivo que significa PHP Hypertext Preprocessor fue creado por Rasmus Lerdorf, es un lenguaje interpretado de propósito general, ampliamente usado y que está diseñado especialmente para el desarrollo web, para la programación de servidores web de código abierto, es un lenguaje de alto nivel, cuyas órdenes se incrustan entre las etiquetas de las páginas web que han sido escritas en HTML. PHP es software libre esto significa que puede descargarse de Internet, modificar su código y copiarse sin ningún costo. Aunque es un lenguaje multiplataforma, ha sido concebido para entornos UNIX y es en este sistema operativo donde se pueden aprovechar mejor sus prestaciones.

PHP ofrece un sinfín de funciones para la explotación de bases de datos de una manera llana y sin complicaciones. Brinda interfaces para el acceso a la mayoría de las bases de datos comerciales tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite. [18]

A continuación se muestra una pequeña lista de algunas de sus funcionalidades:

- Tratamiento de imágenes
- Gestión de archivos
- Gestión de bases de datos
- Funciones de correo electrónico
- Gestión de archivos PDF, Flash

1.4.2.7. ASP.net

ASP.NET es un framework para aplicaciones web, desarrollado y comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y servicios web XML. Apareció en enero de 2002 con la versión 1.0 del .NET Framework, y es la tecnología sucesora de la tecnología Active Server Pages (ASP). ASP.NET está construido sobre el Common Language Runtime, permitiendo a los programadores escribir código ASP.NET usando cualquier lenguaje admitido por el .NET Framework. Es liviano y muy fácil de programar, tiene muchas utilidades que con una breve línea de aprendizaje pueden ser modificadas a su gusto. [19]

Principal desventaja:

- Tiene que correr en computadoras normales que tengan Windows y un servidor web, por lo que no es un lenguaje multiplataforma.

1.4.2.8. Fundamentación de la selección del lenguaje

Para la selección del lenguaje se realizó una comparación teniendo en cuenta el entorno donde se va a desplegar la aplicación y los requisitos del usuario.

Multiplataforma: Todos los lenguajes expuestos anteriormente son multiplataforma excepto ASP.net

Velocidad de ejecución: PHP es mucho más rápido respecto a ASP.net

Disponibilidad de recursos: Actualmente los lenguajes más utilizados en Internet son PHP y JSP. PHP posee una de las comunidades más grandes en Internet permitiendo grandemente la disponibilidad y reutilización de código.

Familiaridad con el lenguaje: Tanto en la UCI como en las FAR el lenguaje más utilizado para desarrollar aplicaciones web es PHP.

Una vez realizada esta comparación se decide tomar el lenguaje PHP por ser el más recomendable y del lado del cliente JavaScript con CSS y XHTML mediante el framework Ext JS en su versión 2.2.

1.4.3. Sistema gestor de base de datos (SGBD)

Los sistemas de gestión de bases de datos o SGBD (en inglés Database Management System, abreviado DBMS) 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. Se define también como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones:

- Definición de los datos.
- Mantenimiento de la integridad de los datos dentro de la base de datos
- Control de la seguridad y privacidad de los datos
- Manipulación de los datos [20]

Este tipo de sistema facilita el trabajo de los desarrolladores con el manejo de las bases de datos aumentando el nivel de productividad.

1.4.3.1. MySQL

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Funciona en diferentes plataformas. Posee un sistema de privilegios y contraseñas que es muy flexible y seguro, y que permite verificación basada en el host. Las contraseñas son seguras porque todo el tráfico de contraseñas

está encriptado cuando se conecta con un servidor. Soporta a grandes bases de datos. Un índice puede usar prefijos de una columna para los tipos de columna CHAR, VARCHAR, BLOB, o TEXT. MySQL Server tiene soporte para comandos SQL para chequear, optimizar, y reparar tablas.

Ventajas:

- La velocidad a la hora de realizar las operaciones, lo que le hace uno de los gestores que ofrecen mayor rendimiento.
- Su bajo consumo lo hacen apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Tiene una probabilidad muy reducida de corromper los datos, incluso en los casos en los que los errores no se produzcan en el propio gestor, sino en el sistema en el que está.

Desventajas:

- Carece de soporte para transacciones, rollback y subconsultas.
- El hecho de que no maneje la integridad referencial, hace de este gestor una solución pobre para muchos campos de aplicación, sobre todo para aquellos programadores que provienen de otros gestores que sí que poseen esta característica. [21]

No es viable para su uso con grandes bases de datos, a las que se acceda continuamente, ya que no implementa una buena escalabilidad.

1.4.3.2. PostgreSQL

PostgreSQL es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD (<http://www.postgresql.org>). Con cerca de una década de desarrollo, PostgreSQL es el gestor de bases de datos de código abierto más avanzado en la actualidad, ofreciendo control de concurrencia multi-versión, soportando casi todas las sintaxis SQL (incluyendo subconsultas, transacciones, y tipos y funciones definidas por el usuario), cuenta además con un amplio conjunto de enlaces con lenguajes de programación (incluyendo C, C++, Java, Perl, TCL y Python). El número de bases de datos que puede contener es ilimitado.

El desarrollo de PostgreSQL está dirigido por una comunidad bien organizada de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo, de ahí el nivel de aceptación y de calidad de este. [22]

Ventajas:

- Posee una gran escalabilidad. Es capaz de ajustarse al número de Unidades Centrales de Procesamiento (CPUs) y a la cantidad de memoria que posee el sistema de forma óptima.
- Implementa el uso de rollback, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz.
- Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos.
- Requiere pocos recursos de hardware y la simplificación del proceso de administración de licencias de software, que no es necesario cuando se usa software libre.
- Puede lidiar con gran volumen de datos.

Desventajas:

- Tiene un límite de 8K por fila, aunque se puede aumentar a 32K, con una disminución considerable del rendimiento.

1.4.3.3. Oracle

Oracle es básicamente una herramienta cliente/servidor para la gestión de Bases de Datos. Es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio hace que sólo se vea en empresas muy grandes y multinacionales, por norma general. En el desarrollo de páginas web pasa lo mismo: como es un sistema muy caro no está tan extendido como otras bases de datos, por ejemplo, Access, MySQL y SQL Server. [23]

Su principal defecto es su enorme precio, que es de varios miles de euros, además de criticársele la política de suministro de parches de seguridad desde el 2005, lo cual ha incrementado el nivel de exposición de los usuarios en lo que a seguridad respecta.

1.4.3.4. Microsoft SQL Server

Microsoft SQL Server es también un sistema de bases de datos relacional, entre sus características se destacan el soporte de transacciones, su escalabilidad, estabilidad y seguridad. Incluye un potente entorno gráfico de administración. Este sistema constituye la alternativa de Microsoft a otros potentes gestores como Oracle, PostgreSQL o MySQL. Su principal dificultad radica en que no es multiplataforma, solo está disponible en Sistemas Operativos de Microsoft. [24]

Ventajas:

- Soporte de transacciones.
- Gran estabilidad.
- Gran seguridad.
- Escalabilidad.
- Soporta procedimientos almacenados.
- Incluye también un potente entorno gráfico de administración, que permite el uso de comandos, Lenguaje de Definición de Datos (DDL) y Lenguaje de Manipulación de Datos (DML) gráficamente.
- Además, permite administrar información de otros servidores de datos.

Desventajas:

- Tiene soporte solamente en el sistema operativo Windows.
- Es un software privativo.

1.4.3.5. Fundamentación de la selección del SGBD

Después del estudio realizado de los diferentes gestores de base de datos más usados se decide utilizar como SGBD a PostgreSQL por ser multiplataforma, gratuito y por la amplia comunidad que tiene detrás para su desarrollo. Además, permite una fácil integración con el lenguaje de programación seleccionado y se adapta a las especificaciones de la solución.

1.4.4. Frameworks y librerías

1.4.4.1. Ext JS

Ext JS es una librería JavaScript ligera y de alto rendimiento, compatible con la mayoría de navegadores para crear páginas web dinámicas. Ext JS es una librería construida con JavaScript cuya potencia radica en la rica colección de componentes para el diseño de interfaces gráficas de usuario del lado del cliente, haciendo uso extensivo de AJAX.

Entre los componentes que esta librería ofrece se encuentran cuadros de diálogo, menús, tablas editables, paneles, pestañas y todo lo necesario para construir atractivos desarrollos al estilo de Web 2.0. Los datos son obtenidos con AJAX a través de XML y/o JSON. [25]

Ventajas:

- Posee una gran cantidad de widgets.
- El diseño está completamente separado de la funcionalidad dándole más flexibilidad.
- Funciones comunes fáciles de implementar, como validaciones, ventanas (con minimizar y maximizar), grillas editables lo que lo hace muy parecido a aplicaciones de escritorio.
- Posee una amplia documentación y comunidad de desarrollo.

Desventajas:

- El tiempo de aprendizaje es similar al de aprender a programar en un nuevo lenguaje.

1.4.4.2. Doctrine

Doctrine es un potente y completo sistema ORM (Object Relational Mapper) para PHP 5.2.3+ con un con un DBAL (Database Abstraction Layer) incorporado, además posee una potente capa de abstracción de bases de datos (DBAL). Una de sus principales características es la de escribir las consultas de bases de datos en un objeto propiamente orientado al Dialecto SQL llamado DQL (Doctrine Query Language) por sus siglas en inglés inspirado en Hibernate SQL. [26]

Esto ofrece a los desarrolladores una poderosa alternativa frente a SQL manteniendo la flexibilidad y permitiendo ampliamente la reutilización de código.

1.4.4.3. Zend Ext Framework

Es un framework Open Source desarrollado por el grupo de arquitectura de la UCID, está diseñado para PHP 5 y con buenas capacidades de ampliación. Es elaborado a partir de Zend Framework cumpliendo con todas sus características. Este trae de novedoso un controlador vertical para el control de las acciones realizada por las vistas hacia el controlador, un motor de reglas para las validaciones en el servidor. Se le incluyó el IOC para la comunicación entre los módulos o componentes, la integración con el ORM Doctrine Framework para trabajo en la capa de abstracción a base de datos, el Ext JS Framework para el desarrollo de las vistas y un controlador de trazas para controlar las acciones del sistema (acción, excepciones, rendimiento, integración, y excepción de integración).

1.4.4.4. Marco de trabajo Sauxe

El Marco de Trabajo Sauxe se ha estructurado de manera tal que facilite la reutilización de los diferentes componentes y que sea de fácil entendimiento para todos los programadores que desarrollen sobre el mismo. Sauxe utiliza Ext JS para implementar la capa de presentación, se apoya en Zend, una extensión de Zend Framework para el desarrollo de la lógica del negocio y para la gestión de los datos utiliza Doctrine. Este utiliza como patrón arquitectónico Modelo Vista Controlador (MVC). También tiene como propósito insertar la programación orientada a aspectos así como la inversión de controles. El framework en su nueva configuración define que la conexión a la base de datos será configurada en un XML almacenado en la carpeta de recursos comunes del proyecto. Es válido aclarar que este cuenta con un componente de transacciones mediante el cual serán salvados automáticamente los datos de modificaciones e inserciones. Es decir, ya no será necesaria la implementación por parte del programador de las consultas de inserción, éste solamente deberá programar la obtención de los campos de la presentación y el Sauxe será el responsable de que los mismos sean guardados en la base de datos.

Algunas de las ventajas más importantes del nuevo marco de trabajo son:

- Los identificadores de cada tupla de las tablas serán generados automáticamente en la base de datos como una secuencia.
- El antiguo método de try y catch para el lanzamiento de excepciones desaparece, en el nuevo marco con registrar las excepciones en el manager de excepciones, el framework será capaz de realizar un mejor tratamiento de las mismas.
- No será necesario en cada método de inserción de información a la base de datos ejecutar el método correspondiente en la clase del negocio correspondiente, el framework será capaz de que una vez tomados los datos de la presentación salvarlos en la base de datos directamente.

1.4.4.5. Fundamentación de la selección del uso de frameworks y librerías

Se decide utilizar el marco de trabajo Sauxe para el desarrollo de la solución por su gran capacidad de integración entre todos sus componentes, el fuerte trabajo que realiza en el manejo de excepciones.

1.4.5. Metodología de desarrollo de software

El desarrollo de software no es una tarea fácil. Como resultado a este problema ha surgido una alternativa desde hace mucho: la Metodología. Las cuales imponen un proceso disciplinado sobre el desarrollo de software con el fin de hacerlo más predecible y eficiente. Lo hacen desarrollando un proceso detallado con un fuerte énfasis en planificar inspirado por otras disciplinas de la ingeniería. Se puede decir además que una metodología es un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar nuevo software.

Un proceso de software detallado y completo suele denominarse “Metodología”. Las metodologías se basan en una combinación de los modelos de proceso genéricos (cascada, evolutivo e incremental.). Adicionalmente una metodología debería definir con precisión los artefactos, roles y actividades involucrados, junto con prácticas y técnicas recomendadas, guías de adaptación de la metodología al proyecto, guías para uso de herramientas de apoyo.

Existen varias metodologías, por ejemplo: la metodología ágil Programación Extrema (XP), la metodología pesada Proceso Unificado de Desarrollo (RUP), entre otras.

1.4.5.1. UML (Unified Modeling Language)

El Lenguaje Unificado de Modelado (UML), permite la modelación de sistemas con tecnología orientada a objetos. Es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema con gran cantidad de software. UML proporciona una forma estándar de escribir los planos de un sistema, cubriendo tanto las cosas conceptuales, tales como procesos del negocio y funciones del sistema, como las cosas concretas, tales como las clases escritas en un lenguaje de programación específico, esquemas de bases de datos, componentes y software reutilizable. [27]

UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir.

1.4.5.2. BPMN (Business Process Modeling Notation)

La iniciativa de gestión de procesos empresariales (por sus siglas en inglés BPMI) ha desarrollado el estándar Notación para el Modelado de Procesos de Negocio (BPMN). El principal objetivo de BPMN es proporcionar una forma de notación gráfica para expresar los procesos de negocio en un diagrama de procesos de negocio (DBP) que sea fácilmente comprensible tanto para los usuarios del negocio, los analistas de procesos de negocio, los desarrolladores encargados de la aplicación de la tecnología que llevará a cabo esos procesos como a los trabajadores del negocio que van a administrar y supervisar esos procesos. Por lo tanto, BPMN crea un estándar entre el diseño del proceso de negocio y el proceso de implementación. [28]

BPMN proporciona un medio sencillo de comunicar información de procesos a los usuarios de otros negocios, los ejecutores de procesos, clientes y proveedores.

1.4.5.3. RUP (Rational Unified Process)

RUP es el resultado de varios años de desarrollo y uso práctico en el que se han unificado técnicas de desarrollo, a través del UML, y trabajo de muchas metodologías utilizadas por los clientes. En un proyecto de desarrollo de software RUP define "Quién" debe hacer "Qué", "Cuándo" y "Cómo" debe hacerlo, además permite la ejecución iterativa del proyecto y el control de riesgos.

Los verdaderos aspectos que definen el Proceso Unificado de software se resumen en tres frases claves:

- Guiado por los Casos de Uso.
- Centrado en la Arquitectura.
- Iterativo e incremental.

En RUP se han agrupado las actividades en grupos lógicos definiéndose 9 flujos de trabajo principales. Los 6 primeros son conocidos como flujos de ingeniería y los tres últimos como de apoyo. Según la metodología el ciclo de vida de un proyecto se divide en las siguientes fases:

- Fase de inicio.
- Fase de elaboración.
- Fase de construcción.
- Fase de transición.

RUP implementa las siguientes mejores prácticas asociadas al proceso de Ingeniería de Software: Gestión de los requisitos, desarrollos iterativos, uso de arquitecturas basadas en componentes, desarrollo visual del software (con UML), verificación continua de la calidad del software, gestión de los cambios. [27]

Esta metodología conlleva a un uso exhaustivo de documentación durante todo el ciclo de un proyecto y por su grado de complejidad puede no resultar muy adecuada ya que si se va a usar en un proyecto pequeño sería muy tedioso usar todas las fases en un período corto de tiempo.

1.4.5.4. Proceso de desarrollo y Gestión de Proyectos de software

El proceso de desarrollo de software propuesto en la UCID, describe la secuencia de actividades de alto nivel para la construcción y desarrollo de soluciones. El desarrollo de cualquier proyecto de software consiste en cinco fases secuenciales: Inicio, Modelación, Construcción, Explotación Experimental y Despliegue, conocidas como el ciclo de vida del proyecto que generalmente definen, qué trabajo técnico se hará en cada fase, cuándo se generarán los entregables, cómo se revisarán, verificarán y validarán en estas, quién está involucrado en cada fase y cómo controlar y aprobar cada una. Las etapas o fases permiten mejorar el planeamiento, ejecución y control de un proyecto, se caracterizan por la conclusión y la aprobación de uno o más artefactos entregables siendo estos verificables. Al finalizar cada fase se

verifica el cumplimiento de los objetivos y se evalúa si se pasa o no a la siguiente ya que la consecución exitosa de cada fase es indispensable para poder continuar con el proyecto.

El modelo de desarrollo de software propuesto en la UCID se logra con la combinación entre los modelos basado en Componentes y el Iterativo e Incremental.

Iterativo e incremental: Es un enfoque en el que el ciclo de vida está compuesto por iteraciones, estas son pequeños procesos compuestos de varias actividades cuyo objetivo es entregar una parte del sistema parcialmente completo, probado, integrado y estable. Todo el software es integrado en cada entrega de cada iteración hasta obtener el producto de software completo en la última iteración. En cada iteración se obtiene como resultado un incremento.

Basado en componentes: Lleva a alcanzar un mayor nivel de reutilización de software, aún en contextos distintos de aquellos para los que fue diseñado. Permite que las pruebas sean ejecutadas probando cada uno de los componentes antes de probar el conjunto completo de componentes ensamblados. Cuando existe un débil acoplamiento entre componentes, el desarrollador es libre de actualizar y/o agregar componentes según sea necesario, sin afectar otras partes del sistema. Dado que un componente puede ser construido y luego mejorado continuamente, la calidad de una aplicación basada en componentes mejorará con el paso del tiempo.

Orientado a procesos: La modelación de proceso de negocio permite realizar una rápida y profunda exploración del dominio del problema, con el fin de lograr comprensión por parte del equipo de desarrollo de los procesos que se realizan actualmente en la entidad y la relación que existe entre estos. De esta forma, se van determinando necesidades operacionales, así como restricciones que presenta la entidad, obteniéndose finalmente un entendimiento del negocio para dar paso a la fase inicial del sistema.

1.4.5.5. Fundamentación de la selección de la metodología de desarrollo

Se decide utilizar el Proceso de desarrollo y Gestión de Proyectos de software por su fortaleza y flexibilidad, junto a BPMN para modelar el negocio por procesos; aprovechando la claridad y transparencia que este brinda para modelar los procesos del negocio.

1.4.6. Herramientas CASE

Las herramientas CASE (Computer Aided Software Engineering en español Ingeniería de Software Asistida por Ordenador) de modelado, permiten aplicar la metodología de análisis y diseño orientados a objetos y abstraerse del código fuente, estos programas de ayuda dan asistencia a los analistas, diseñadores y desarrolladores durante todo el ciclo de vida de desarrollo de un software.

1.4.6.1. Visual Paradigm for UML

Visual Paradigm para UML es una herramienta UML profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. [29]

Es una herramienta fácil de usar que soporta notación UML, entorno de creación de diagramas para BPMN, ingeniería inversa, generación de código, generador de informes y permite editar las figuras.

1.4.6.2. Rational Rose Enterprise

Rational Rose Enterprise es una herramienta CASE que soporta notación UML y un ambiente de modelado que soporta la generación de código a partir de modelos en Ada, ANSI C++, C++, CORBA, Java™/J2EE™, Visual C++® y Visual Basic®. Proporciona un lenguaje común de modelado para el equipo que facilita la creación de software de calidad mucho más rápido. [30]

Como desventaja tiene que solo es compatible con la familia de sistemas operativos de Windows.

1.4.6.3. Fundamentación de la herramienta CASE seleccionada

Se decide utilizar como herramienta CASE a Visual Paradigm principalmente por permitir el modelado de diagramas para BPMN y por ser multiplataforma.

1.4.7. Herramientas de desarrollo

Las herramientas de desarrollo son aplicaciones que tienen cierta importancia en el desarrollo de un producto de software. Pueden ser de importancia vital (como un compilador, un ensamblador, o un editor) o de importancia secundaria, como un Entorno de Desarrollo Integrado (IDE).

1.4.7.1. Zend Studio for Eclipse

Zend Studio for Eclipse es la siguiente generación en la familia Zend Studio IDE. Esta nueva liberación se basa en la probada tecnología de Zend Studio y el proyecto Eclipse PHP Developers Tools (PDT). Zend Studio for Eclipse crea un IDE profesional que proporciona a PHP capacidades de desarrollo con el soporte de Eclipse en varios idiomas y plugins que crea los mundos más poderosos para la IDE de PHP. [31]

Tiene soporte superior para el desarrollo de modernas aplicaciones AJAX al ofrecer potentes capacidades para PHP, la mejora de soporte de JavaScript, profunda integración a Zend Framework y Dojo, posibilitando que el desarrollo sea realizado en un tiempo récord.

Ofrece un sinfín de ventajas entre las que se encuentran:

- Una comunidad de millones de usuarios y miles de desarrolladores
- Integración mejorada con Zend Framework e Integración con Zend Platform
- Cientos de plugins
- Soporte básico de depuración de scripts PHP

1.4.7.2. Aptana Studio

Es un entorno integrado de desarrollo para la elaboración de aplicaciones webs dinámicas que empleen PHP, Ruby, Ruby on Rails y Python. Cuenta con módulos que permiten utilizarlo para crear programas mediante la tecnología Adobe Air y utilidades para el iPhone. Está basado en Eclipse y se puede encontrar para las tres plataformas mayoritarias (Windows, Mac y GNU/Linux).

Su integración con dos nuevas herramientas de Aptana amplían sus posibilidades casi al infinito. Estas son Aptana Jaxer, un servidor AJAX, y Aptana Cloud, un servicio de hosting complementario.

Aptana Studio posee un asistente de código que ayuda al programador en la escritura de los diferentes lenguajes, CSS y JavaScript entre ellos. En el caso del HTML, puede mostrar todos los elementos pertenecientes a este lenguaje y sus propiedades.

Aptana Studio contiene también información de soporte para los principales navegadores web: IE, Firefox, Opera, Netscape y Safari. [32]

Entre las funciones más destacadas de Aptana Studio se pueden mencionar:

- Asistente de código para HTML y JavaScript.
- Librerías AJAX (jQuery, prototype, scriptaculous, Ext JS, Dojo, YUI y Spry entre otras).
- Conexión vía FTP, SFTP, FTPS y Aptana Cloud.
- Herramientas para trabajo con base de datos.
- Marcado de sintaxis mediante colores.
- Compatible con extensiones para Eclipse (existen más de 1000).
- Explorador de código en forma de árbol.
- Librerías populares AJAX/JavaScript.
- Visor de errores y advertencias.

1.4.7.3. EMS PostgreSQL Manager 3

EMS SQL Manager for PostgreSQL es una poderosa herramienta gráfica para la administración y desarrollo de PostgreSQL Database Server (servidor de bases de datos PostgreSQL). PostgreSQL Manager soporta todas las nuevas características de PostgreSQL incluyendo espacios de tablas (tablespaces), argumentos nombrados (named arguments) en funciones y otras más. Ofrece una gran variedad de herramientas poderosas para usuarios avanzados, tales como Visual Database Designer (diseñador visual de base de datos), Visual Query Builder (constructor visual de consultas), y un poderoso editor de objetos binarios (BLOB) para satisfacer todas sus necesidades. PostgreSQL Manager cuenta con una nueva y avanzada interfaz gráfica de usuario con un sistema asistente bastante descriptivo, tan claro en su uso que ni un principiante se podrá confundir. [33]

Características principales:

- Nueva interfaz gráfica de usuario último modelo.
- Ágil navegación y administración de base de datos.
- Administración sencilla de todos los objetos PostgreSQL.
- Herramientas de manipulación de datos avanzada.
- Administración efectiva de seguridad.
- Excelentes Herramientas visuales y de texto para elaboración de consultas.
- Acceso al servidor PostgreSQL a través del protocolo HTTP.
- Impresionantes opciones de exportación e importación de datos.
- Poderoso diseñador visual de base de datos.
- Asistentes fáciles de usar que realizan mantenimiento de tareas de PostgreSQL.

1.4.7.4. AppServer 2.5.6

Es una herramienta Open Source para Windows que facilita la instalación de Apache, MySQL y PHP en una sola herramienta, esta característica facilita la tarea al usuario ya que se configuran las aplicaciones de forma automática. AppServer instalará en tu ordenador en solamente unos segundos Apache, PHP, MySQL y phpMyAdmin, dejando las aplicaciones configuradas para su funcionamiento inmediato.

AppServer en su última versión (AppServer 2.6.0) incluye Apache 2.2.8: servidor HTTP multiplataforma. PHP 6.0.0-dev: lenguaje de programación dinámico que utilizan la mayoría de gestores de contenidos más populares. Se integra a la perfección con MySQL y Apache. MySQL 6.0.4-alpha: gestor de bases de datos, rápido y seguro. PhpMyAdmin-2.10.3: interfaz gráfica de administración para MySQL.

Una vez instalado AppServer, estarán disponibles dos servidores, uno web y otro de base de datos, configurados de manera local, esto permitirá realizar todas las pruebas necesarias a una aplicación web antes de lanzarla a la red.

1.4.7.5. Navegador Mozilla Firefox

El navegador escogido es Mozilla Firefox por cumplir muy bien con los estándares de la W3C para la web, por ser un navegador libre, multiplataforma y por la amplia comunidad que tiene por detrás para su desarrollo y su mantenimiento. Se requiere Mozilla Firefox 2.2 o superior.

Además, posee entre sus ventajas lo siguiente:

- JavaScript más rápido: Firefox potencia el desarrollo web, con JavaScript incluyendo cierres, generadores de expresión y reducción de matrices.
- Ampliación a página completa: La función de aumento permite maximizar todos los elementos del diseño uniformemente.
- Protección contra programas espías: Firefox no permitirá que una página web descargue, instale, o ejecute programas en un ordenador sin el consentimiento del usuario. Este avisará siempre que se descargue o instale un software.
- Antiphishing y Antimalware: Firefox posee la protección más avanzada contra los tipos malos de la red.

1.5. Conclusiones

En este capítulo se realizó un análisis del estado del arte a nivel nacional como internacional referente a aplicaciones que se enfoquen en la automatización del proceso de evaluación y desarrollo del Capital Humano y que permitan realizar un análisis del resultado de las evaluaciones. Se presentó un resumen de las tecnologías y herramientas a utilizar para llevar a cabo el producto de software. Se determinó utilizar como lenguaje del lado del cliente a JavaScript auxiliándose de las librerías de Ext JS en su versión 2.2 y lenguaje del lado del servidor PHP, como entorno de desarrollo se tomó a Aptana Studio y Zend Studio for Eclipse respectivamente, como gestor de bases de datos a PostgreSQL, como marco de trabajo se escogió el Marco de Trabajo Sauxe y como metodología el proceso de Desarrollo y Gestión de Proyectos de Software con BPMN auxiliándose de la herramienta CASE Visual Paradigm y como navegador la versión de Moxilla Firefox 2.2 o superior. La decisión de utilizar las mismas fue tomada por la dirección del proyecto siguiendo la política establecida por el centro para el desarrollo de software.

Capítulo 2: Características del sistema

2.1. Introducción

En el presente capítulo se muestran las características que tendrá el sistema. Se realizan el mapa de procesos, la descripción de los mismos y el modelado del proceso “Analizar resultado de la evaluación”, se identifican y se describen los requerimientos funcionales y no funcionales.

2.2. Situación problemática

La administración del país evalúa el desempeño de sus subordinados mediante planillas de evaluación, aprobadas de forma centralizada. En la mayoría de los casos los seminarios que se imparten a los jefes sobre el tema no llegan a los evaluadores, trayendo consigo complejidad y mala realización del llenado de la información por parte de los mismos. Al mismo tiempo, los pasos a seguir por los evaluadores se hacen muy complejos, cada vez que se realiza un análisis de una persona en un período de tiempo determinado, se tiene que buscar en todo su expediente realizando un gran esfuerzo humano.

El análisis de los resultados de las evaluaciones de desempeño en las entidades del país es un trabajo engorroso, que requiere de tiempo y esfuerzo excesivo de los especialistas de recursos humanos, lo cual dificulta la oportuna toma de decisiones de los jefes e impide a los especialistas la utilización de métodos matemáticos que descubran las potencialidades, fortalezas y características de los trabajadores que puedan ser aprovechadas en otras áreas de la entidad; así como evitar el posible deterioro de la información acumulada.

2.3. Descripción general de la propuesta de sistema

Para dar cumplimiento a los objetivos previamente planteados, el módulo propuesto constará de dos componentes: uno para gestionar los modelos matemáticos asociados a una planilla evaluativa y otro para gestionar reglas.

El componente gestionar modelos matemáticos permitirá adicionar, modificar y eliminar los modelos asociados a una planilla evaluativa. El objetivo principal de un modelo matemático es clasificar el resultado de una evaluación de acuerdo con los hechos asociados al modelo.

El componente gestionar reglas permitirá adicionar, modificar y eliminar las reglas asociadas a una planilla evaluativa. Una regla está definida por una ecuación matemática que puede estar compuesta por operadores aritméticos, operadores lógicos, indicadores, escalas y a su misma vez por otras reglas.

2.4. Mapa de procesos

Un proceso es un conjunto de fases sucesivas de un fenómeno durante el transcurso de sucesos. Es la marcha hacia un fin determinado. [34]

Un mapa de procesos ofrece una visión general del sistema de gestión. En él se representan los procesos que componen el sistema así como sus principales relaciones.

Propósito:

Mostrar los procesos de negocio que abarcan el proceso de evaluación, así como sus entradas, salidas y relaciones.

Mapa de procesos de negocio

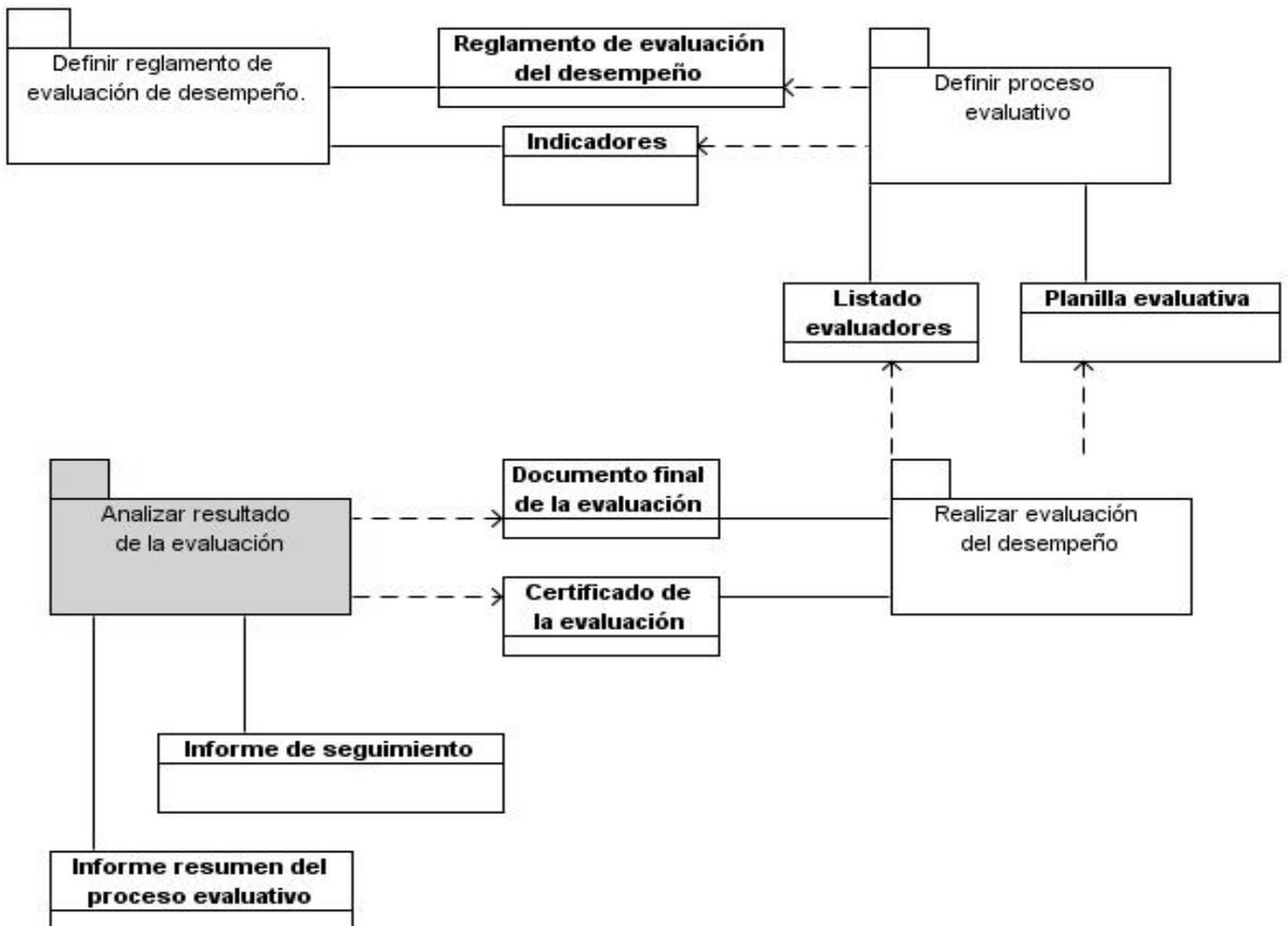


Figura 2; *Error! Utilice la ficha Inicio para aplicar 0 al texto que desea que aparezca aquí..2* Mapa de procesos de negocio.

2.5. Descripción de procesos de negocio

El proceso evaluativo tiene cuatro momentos principales:

1. Definir el reglamento de evaluación del desempeño.
2. Definir el proceso evaluativo.
3. Realizar la evaluación del desempeño.
4. Analizar el resultado de la evaluación.

El presente trabajo hará énfasis en la etapa “Analizar el resultado de la evaluación”.

2.5.1. Definir reglamento de evaluación del desempeño

Cada entidad elaborará de común acuerdo con la organización sindical correspondiente a ese nivel, el reglamento de evaluación del desempeño de los trabajadores, lo cual se inscribe en el Convenio Colectivo de Trabajo.

2.5.2. Definir proceso evaluativo

El objetivo de definir el proceso evaluativo, es preparar las condiciones necesarias para crear un proceso evaluativo de calidad; para esto se tiene como entrada el Reglamento de evaluación del desempeño y los indicadores presentando como salida el listado de los evaluadores y la planilla evaluativa. Aquí se definen los procesos evaluativos que se llevarán a cabo en la entidad, los tipos de evaluaciones, plantillas evaluativas, el cronograma y los evaluadores y/o las comisiones evaluadoras en correspondencia con el reglamento de evaluación de desempeño aprobado.

2.5.2.1. Realizar evaluación del desempeño

El objetivo de realizar evaluación es desarrollar el proceso de evaluar al trabajador, presentando como entrada el listado de los evaluadores y la planilla evaluativa y como salida el documento final de la evaluación y el certificado de la evaluación. El evaluador recopila información relacionada con el desempeño del trabajador que le permitirá elaborar el documento final de la evaluación. Analiza con el trabajador los señalamientos y las recomendaciones de la evaluación final. En caso de que el trabajador no esté de acuerdo se le permite realizar una reclamación según los términos establecidos en la legislación vigente.

2.5.3. Objeto de automatización

2.5.3.1. Analizar resultado de la evaluación

El objetivo de analizar resultado de la evaluación es obtener estadísticas del proceso evaluativo; para su obtención posee como entrada el documento final de la evaluación y el certificado de la evaluación, obteniendo como salida el informe resumen del proceso evaluativo y el informe de seguimiento. En este

proceso se realizan análisis de los procesos evaluativos, dando como resultado, informes del estado del proceso. Permite darle seguimiento a las recomendaciones recogidas en el documento final de la evaluación. [35]

Entre sus actividades fundamentales están:

- Obtener resúmenes estadísticos.
- Obtener los listados de personas.

En el marco de este trabajo se abordará el subproceso “Analizar resultado de la evaluación”.

2.5.4. Descripción del proceso “Analizar resultado de la evaluación”

Objetivos	<ul style="list-style-type: none"> • Realizar un análisis del proceso evaluativo dando como resultado, informes del estado del proceso. • Controlar el proceso de custodia del certificado de la evaluación en el expediente del trabajador y la información a los trabajadores. • Darle seguimiento a las recomendaciones recogidas en el documento final de la evaluación. • Controlar evaluación.
Evento(s) que lo generan	<p>Para el caso de un análisis durante el proceso evaluativo el comienzo del proceso de evaluación de desempeño.</p> <p>Para el caso del análisis de las recomendaciones cuando la dirección de la entidad determine necesario.</p>
Precondiciones	<p>Para el caso de un análisis durante el proceso evaluativo debe haber comenzado un período evaluativo.</p> <p>Para el caso del análisis de las recomendaciones deben existir las evaluaciones de desempeño en el expediente del trabajador.</p>
Pos condición	Impresión de los reportes estadísticos.

Marco Legal	<ul style="list-style-type: none"> • Variante para el Manual de Procedimientos del Trabajo con los Cuadros. Capítulo 11 • Resolución No. 21 del 2007 del MTSS “Evaluación del desempeño de los trabajadores” • Resolución No. 8 del 2005 del MTSS “Reglamento general sobre Relaciones laborales”. Capítulo X • Decreto No. 281 del 2007, “Reglamento para la implantación y consolidación del Sistema de Dirección y Gestión empresarial estatal”. • Normas Cubanas NC 3000-2007 “Sistema de gestión integrada de capital humano”.
Clientes internos	<ul style="list-style-type: none"> • Definir reglamento de evaluación de desempeño. • Realizar Evaluación de desempeño.
Clientes externos	<ul style="list-style-type: none"> • Actualizar Expediente Laboral. • Capacitación y Desarrollo
Entradas	<ul style="list-style-type: none"> • Documento final de la evaluación. • Certificado de la evaluación • Informe de Seguimiento
Salidas	Informe sobre los resultados del proceso evaluativo

Tabla 2.1 Descripción del proceso de negocio Analizar resultado de la evaluación

Para ver el diagrama de dicho proceso remitirse al anexo [1].

Descripción del flujo básico: El proceso se puede iniciar por varias acciones. En el caso que sea un análisis para dar un seguimiento a las recomendaciones realizadas en las evaluaciones parciales y anuales sin pertenecer a un proceso evaluativo específico ir al flujo alterno Realizar Seguimiento. En el caso de que el análisis sea durante el proceso evaluativo para chequear el estado del proceso, seguir flujo básico Realizar análisis del proceso evaluativo.

1. Analizar estado del proceso evaluativo: En el momento que se decida los evaluadores realizarán un análisis para comprobar el estado en que se encuentra el proceso evaluativo. Se discuten los

resultados y se revisa el cumplimiento del cronograma, nivel de completamiento de las evaluaciones por áreas, se analiza el estado de las evaluaciones de desempeño realizadas. Se analizan las recomendaciones, el estado de cumplimiento, la evaluación por indicadores y se chequea si se sigue con el reglamento establecido para asegurar el grado de avance del proceso. Para ello se consultarán el Documento final de la evaluación y el Reglamento de evaluación de desempeño.

En el caso que el proceso evaluativo haya concluido se seguirá el flujo alternativo 2.a.1 Realizar informe final.

- 2. Realizar Resumen:** Una vez concluido el análisis del proceso se realiza un resumen donde se refleja los resultados conclusivos del mismo. Se obtiene como resultado el Resumen estado del proceso y en algunos casos el Informe de seguimiento.
- 3. Enviar resumen del proceso evaluativo:** El Resumen del estado del proceso se envía a la Administración para que tengan conocimiento del estado del proceso evaluativo en el momento que se decidió realizar el análisis.
- 4. Recibir Resumen del proceso:** La Administración recibe el Resumen de estado del proceso evaluativo.
- 5. Concluye el proceso.**

Pos-condiciones: Queda documentado todo el análisis del proceso evaluativo.

Salidas:

1. Informe sobre los resultados del proceso evaluativo.
2. Informe de seguimiento.

Descripción de los flujos paralelos

“No aplicable”.

Flujos alternos:

1. a Realizar Seguimiento

1. a.1 Analizar Recomendaciones: Los evaluadores analizan las recomendaciones realizadas en evaluaciones anteriores y chequean el estado de cumplimiento de cada uno de ellas creando el Informe de Seguimiento que permitirá registrar el grado de avance de las recomendaciones realizadas a los evaluados y las observaciones que se deseen señalar del análisis realizado.

1. a.2 Archivar Informe de Seguimiento: Se archiva el informe de seguimiento para posteriores análisis.

1. a.3 Concluye el proceso.

Pos-condiciones:

Queda creado el Informe de seguimiento

Salidas: Informe de seguimiento

2. a Analizar fin de período evaluativo.

2. a.1 Realizar Informe final: Los evaluadores crean un informe donde se resume el proceso evaluativo y actualiza el informe de seguimiento. Se analizan los datos recopilados del proceso evaluativo y se realizan reportes estadísticos que permitirán mostrar de una manera más detallada los resultados del proceso. Se utilizan como entradas los documentos finales de evaluación y se obtienen como salidas el Informe resumen del proceso evaluativo y el Informe de seguimiento.

2. a.2 Enviar Informe para aprobación: Se envía el informe a la administración para su aprobación.

2. a.3 Recibir Informe: Se recibe el informe de la administración para su aprobación.

2. a.4 Aprobar Informe: Se aprueba el informe por la administración.

2. a.5 Comunicar Colectivo Laboral: Se comunica a los trabajadores los resultados del proceso evaluativo y las conclusiones del mismo reflejadas en el informe final.

2. a.6 Recibir Información: Los trabajadores reciben las informaciones referentes al proceso evaluativo realizado.

2. a.7 Concluye el proceso.

Pos-condiciones: Queda creado el informe resumen del proceso evaluativo.

Salidas: Informe resumen del proceso evaluativo.

2.6. Modelo conceptual

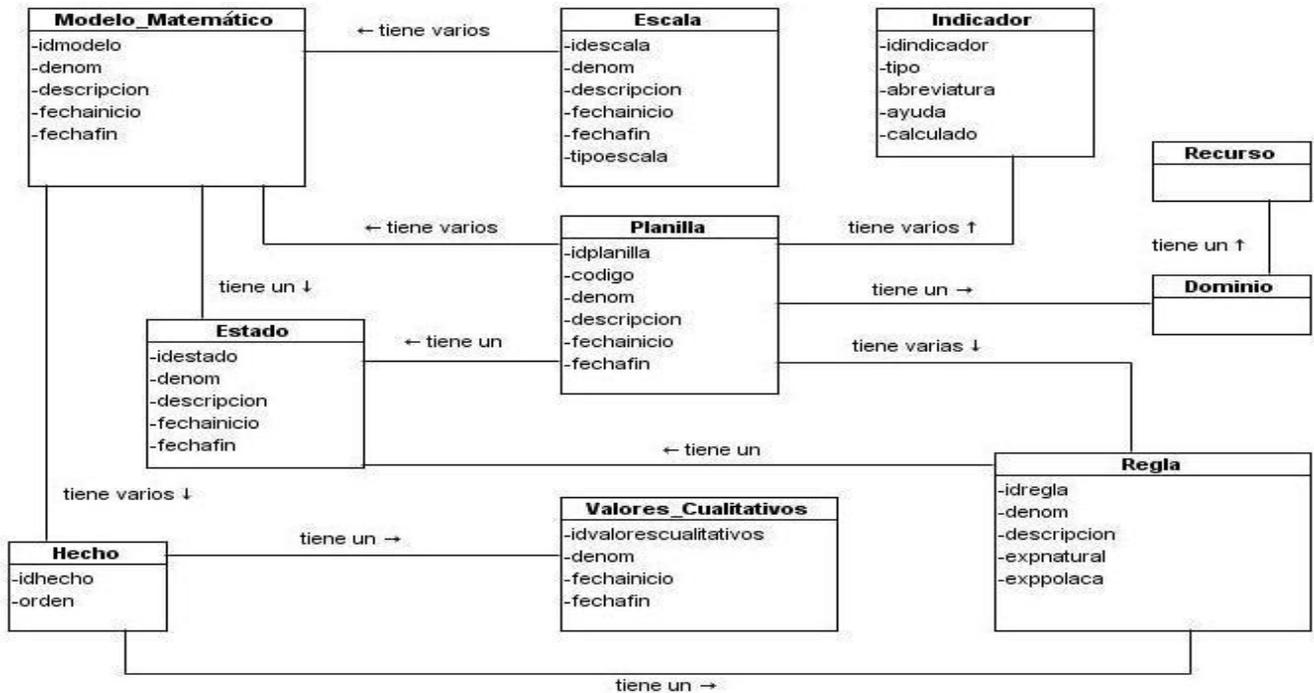


Figura 2; *Error! Utilice la ficha Inicio para aplicar 0 al texto que desea que aparezca aquí..3* Modelo conceptual.

Para obtener más información remitirse al diccionario de datos en el Anexo [2].

2.7. Requerimientos funcionales (RF)

Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir. A continuación se muestra un listado de requisitos que el sistema deberá cumplir.

R1. Buscar planilla evaluativa.

El requisito funcional Buscar planilla evaluativa permitirá encontrar planillas de una forma fácil. Incluye los siguientes requisitos.

R1.1 Filtrar por dominio.

R1.2 Buscar por denominación.

R2. Gestionar regla.

El requisito funcional Gestionar regla servirá para gestionar todas las reglas con que cuenta un modelo matemático. Este incluye los siguientes requerimientos:

R2.1 Adicionar regla.

R2.2 Modificar regla.

R2.3 Eliminar regla.

R2.4 Listar reglas.

R3. Gestionar modelo matemático.

El requisito Gestionar modelo matemático permitirá gestionar los modelos matemáticos asociados a una planilla evaluativa. Este incluye los siguientes requerimientos:

R3.1 Adicionar modelo matemático.

R3.2 Modificar modelo matemático.

R3.3 Eliminar modelo matemático.

R3.4 Listar modelos matemáticos.

R3.5 Asociar hecho. Referencia (R4).

R4 Gestionar hecho.

El requisito Gestionar hecho permitirá gestionar los hechos asociados a un modelo matemático. Este incluye los siguientes requerimientos:

R4.1 Adicionar hecho.

R4.2 Eliminar hecho.

R4.5 Listar hechos.

Para obtener más información remitirse al Anexo [3].

2.8. Requerimientos no funcionales

Los requerimientos no funcionales son las propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. En muchos casos los requisitos no funcionales son fundamentales en el éxito del producto. Normalmente, están vinculados a requerimientos funcionales, es decir, una vez se conozca lo que el sistema debe hacer se puede determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser. [36]

Para el sistema propuesto se han definido los siguientes requisitos no funcionales:

Usabilidad: El sistema podrá ser usado por personas con conocimientos básicos sobre el manejo de ordenadores. Se emplearán barras de progreso para indicar el estado de los procesos que por su complejidad requieran de un tiempo de procesamiento apreciable por los usuarios. El software tendrá siempre visible la opción de Ayuda, lo que posibilitará un mejor entendimiento por parte de los usuarios de sus funcionalidades.

Seguridad: Los usuarios deberán estar correctamente autenticados por sus roles correspondientes, para lograr visualizar cada uno de los elementos relacionados con el alcance de su trabajo, y a su vez, tener acceso a los elementos necesarios para su desarrollo laboral.

Portabilidad: El sistema podrá ser ejecutado sobre los sistemas operativos GNU/Linux o Windows, permitiendo una fácil migración haciendo uso de estándares y tecnologías de código abierto.

Software: Para trabajar con la aplicación se debe disponer de un sistema operativo GNU/Linux, Windows 98 o superior. Navegador web con capacidad de interpretación de JavaScript, CSS y AJAX. Preferiblemente Mozilla Firefox 2.2 o superior.

Requerimiento de Ayuda y Documentación: El sistema debe contar con una ayuda general en la página principal, que guíe al usuario al trabajar en el sistema. Estará disponible en cada una de las interfaces, de

manera que los usuarios tengan conocimiento de las funcionalidades del mismo y logren hacer un mejor uso de ellas.

2.9. Conclusiones

En este capítulo quedaron definidos los requisitos funcionales y no funcionales que debe tener la aplicación así como la especificación por escrito de los RF, se obtuvo el mapa de procesos y la descripción referente al diagrama del proceso de negocio “Analizar resultado de la evaluación” y se definió el mapa conceptual para dicho proceso.

Capítulo 3: Análisis y diseño del sistema

3.1. Introducción

En el presente capítulo se realiza una descripción del sistema a automatizar, explicando cada uno de los patrones de diseño a utilizar, para un mayor entendimiento se representa gráficamente los diagramas de interacción (secuencia), diagramas de clases del diseño con estereotipos web y el diagrama entidad relación de la base de datos, realizándose una breve descripción de cada uno de ellos.

3.2. Patrones de arquitectura y diseño

Los patrones son un amplio repertorio de principios generales basados en la experiencia que guían la creación de un software. Para que una solución sea considerada un patrón debe ser aplicable a diferentes problemas en distintas circunstancias. Los patrones de arquitectura expresan un esquema organizativo estructural fundamental para sistemas de software especificando subsistemas con sus responsabilidades, detallando un grupo de recomendaciones para organizar componentes. Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software.

3.2.1. Patrón Modelo Vista Controlador (MVC)

Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones Web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página, el modelo es el sistema de gestión de base de datos y el controlador representa la Lógica de negocio. [37]

Modelo: Esta es la representación específica de la información con la cual el sistema opera. Administra el comportamiento y los datos del dominio de aplicación, responde a requerimientos de información sobre su estado y responde a instrucciones de cambiar el estado. El acceso a datos está implementado por clases PHP denominadas [Dat/Eval][requisito_x] ó [Dat/Nom][requisito_x] encargadas de la lectura de datos y las [requisito_x]Model se especializan en las funcionalidades de insertar, modificar y eliminar.

Vista: Representa el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario. Las vistas son las encargadas de enviar las peticiones de los usuarios a las clases controladoras y representar los resultados obtenidos, dichas vistas están implementadas por ficheros JavaScript:

[requisito_x].js (encargada de la lógica de interfaz), ext_base.js (encargada del manejo de las solicitudes, respuestas y manejo de componentes de Ext JS), ext_all.js (encargada de la creación de los componentes visuales de la vista) y ucid_all.js (contiene funciones definidas por el equipo de arquitectura en el marco de trabajo con el objetivo de incluir extensiones propias al framework). Todos estos ficheros son llamados desde un fichero [requisito_x].phtml que es el encargado de representar la interfaz de usuario.

Controlador: Controla el flujo entre la vista y el modelo (los datos). Responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y en la vista. Las clases PHP controladoras tendrán el nombre de [requisito_x Controller].php y son las encargadas de recibir las peticiones de los usuarios desde la vista y hacer llamadas a las funciones necesarias para dar cumplimientos a las mismas, son las clases responsables de mediar entre la capa modelo y la vista.

3.2.2. Patrones generales de software para asignar responsabilidades (GRASP)

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. [38]

Las responsabilidades están relacionadas con las obligaciones de un objeto en cuanto a su comportamiento. Con el objetivo de alcanzar mayor calidad en el diseño se tuvieron en cuenta los siguientes patrones de diseño GRASP.

3.2.2.1. Patrón Controlador

Problema: ¿Quién debería encargarse de atender un evento del sistema?

Solución: Asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema, a una clase que represente una de las siguientes opciones:

- El “sistema” global (controlador de fachada).
- La empresa u organización global (controlador de fachada).
- Algo en el mundo real que es activo (por ejemplo, el papel de una persona) y que pueda participar en la tarea (controlador de tareas).

Aplicación: En el sistema cada requisito funcional cuenta con una clase [requisito_xController].php que es la encargada de manejar los eventos y la lógica del negocio del sistema relacionados al mismo.

3.2.2.2. Patrón Experto

Problema: ¿Quién debiera ser el responsable de conocer la información?

Solución: La responsabilidad de realizar una labor es de la clase que tiene o puede tener los datos involucrados (atributos). Una clase, contiene toda la información necesaria para realizar la labor que tiene encomendada.

Hay que tener en cuenta que esto es aplicable mientras se consideren los mismos aspectos del sistema:

- Lógica de negocio
- Persistencia a la base de datos
- Interfaz de usuario

Aplicación: El sistema implementa este patrón en forma de servicios; ejemplo de su aplicación: la clase modelosController.php al necesitar datos del nomenclador estado y de los modelos matemáticos obtiene la información través de las clases NomEvalEstado.php y DatEvalmodelo.php, las cuales son las encargadas del acceso a datos.

3.2.2.3. Patrón Creador

Problema: ¿Quién debería ser responsable de crear una nueva instancia de alguna clase?

Solución: Este patrón como su nombre lo indica es el que crea, el guía la asignación de responsabilidades relacionadas con la creación de objetos, se asigna la responsabilidad de que una clase B cree un Objeto de la clase A solamente cuando:

- B contiene a A
- B es una agregación (o composición) de A
- B almacena a A
- B tiene los datos de inicialización de A (datos que requiere su constructor)

- B usa a A.

Aplicación: Las clases [requisito_x Controller].php, son responsables de crear instancias de las clases [requisito_x]Model para realizar las funciones de insertar, modificar y eliminar datos en la base de datos e instancias de la clase [Dat/Nom][requisito_x] o en caso de necesitar cargarlos.

3.2.2.4. Patrón de Alta Cohesión

Problema: ¿Cómo mantener la complejidad dentro de límites manejables?

Solución: Asignar una responsabilidad de modo que la cohesión siga siendo alta.

Aplicación: Cada elemento del diseño realiza una labor única dentro del sistema, no desempeñada por el resto de los elementos, aunque los modelos matemáticos están relacionados con las reglas no se decide la implementación de una clase general sino la creación de clases para cada proceso de gestión (modelosController.php y reglasController.php) evitando la sobrecarga, compresión y reutilización de funcionalidades.

3.3. Diagramas de clases de diseño

El diagrama de clase es el diagrama principal de diseño y análisis para un sistema, estos son de estructura estática que muestran las clases del sistema y sus interrelaciones (incluyendo herencia, agregación y asociación.). Los diagramas de clases son el pilar básico del modelado con UML, siendo utilizados tanto para mostrar lo que el sistema puede hacer (análisis), como para mostrar cómo puede ser construido (diseño).

3.3.1. Diagrama de clases genérico

El objetivo del diagrama de clases genérico es mostrar de forma general la estructura de los diagramas de clases del diseño del sistema, especificando en color gris claro los paquetes comunes para todos los diagramas de clases específicos de cada requerimiento funcional.

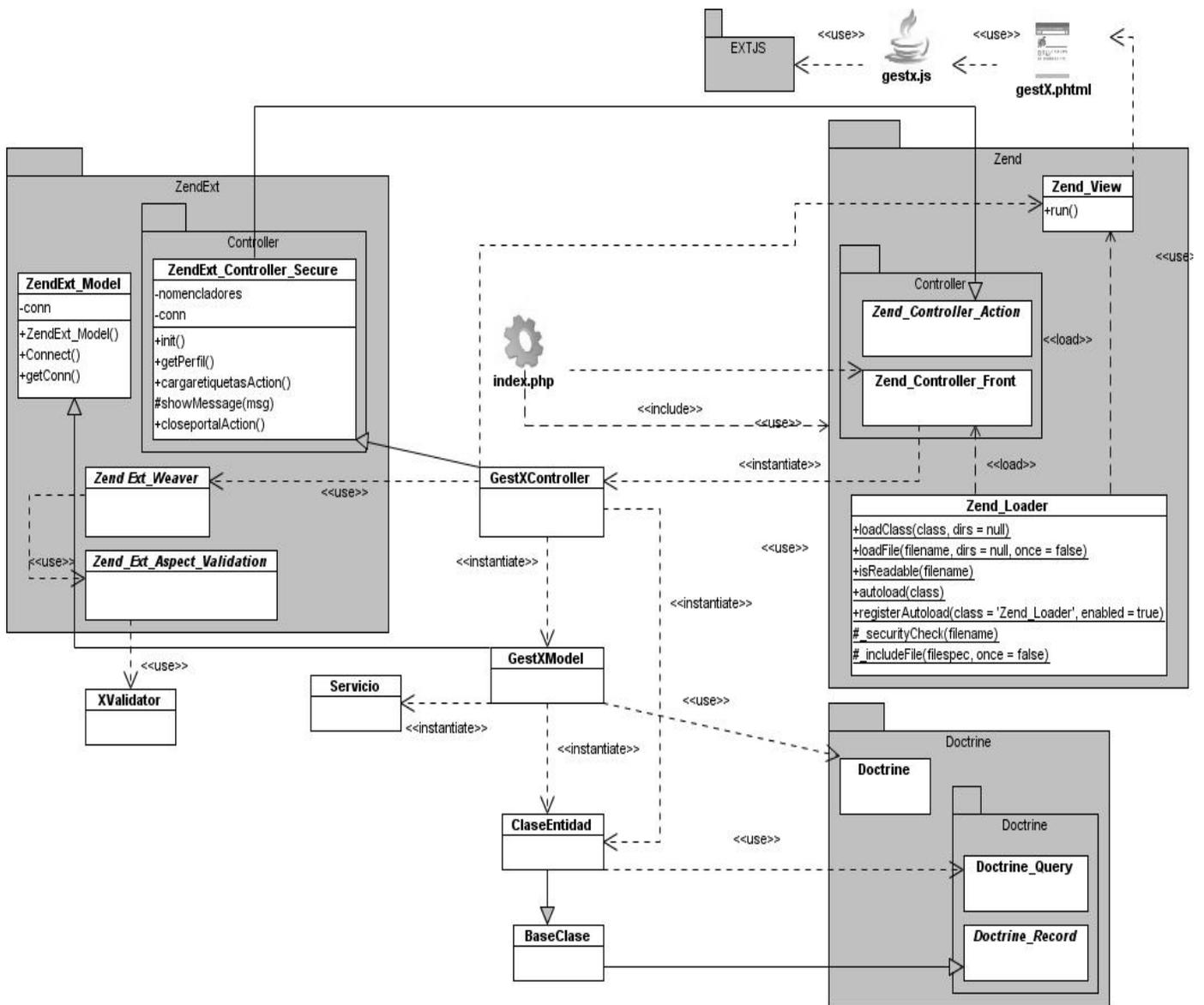


Figura 3.4 Diagrama de clases genérico.

3.3.2. Diagramas específicos

En estos diagramas se omiten los paquetes y clases comunes del diagrama de clases genérico representando lo específico para cada diagrama de clase del diseño.

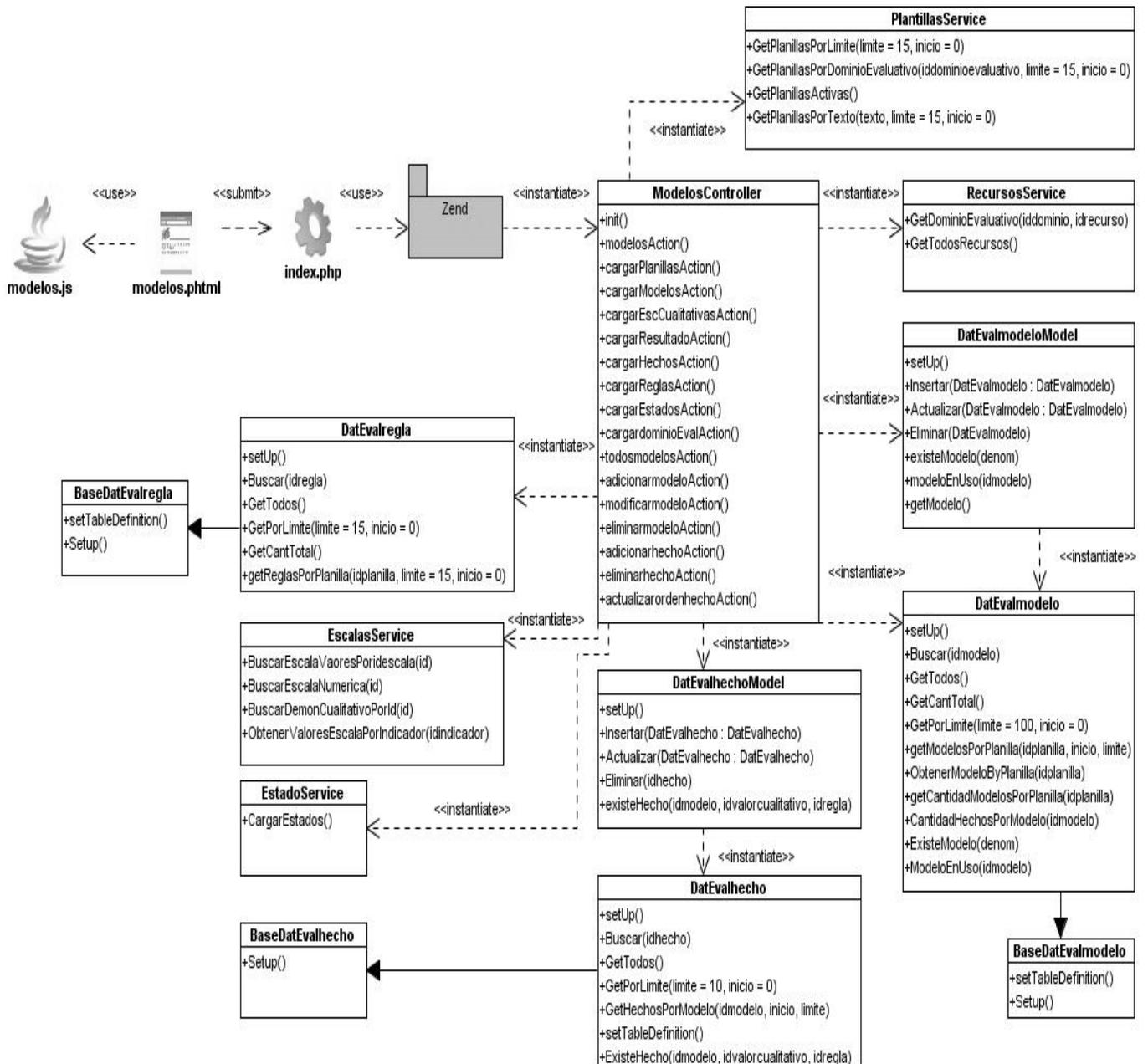


Figura 3.5 Diagrama de clases del diseño: Gestionar modelos matemáticos.

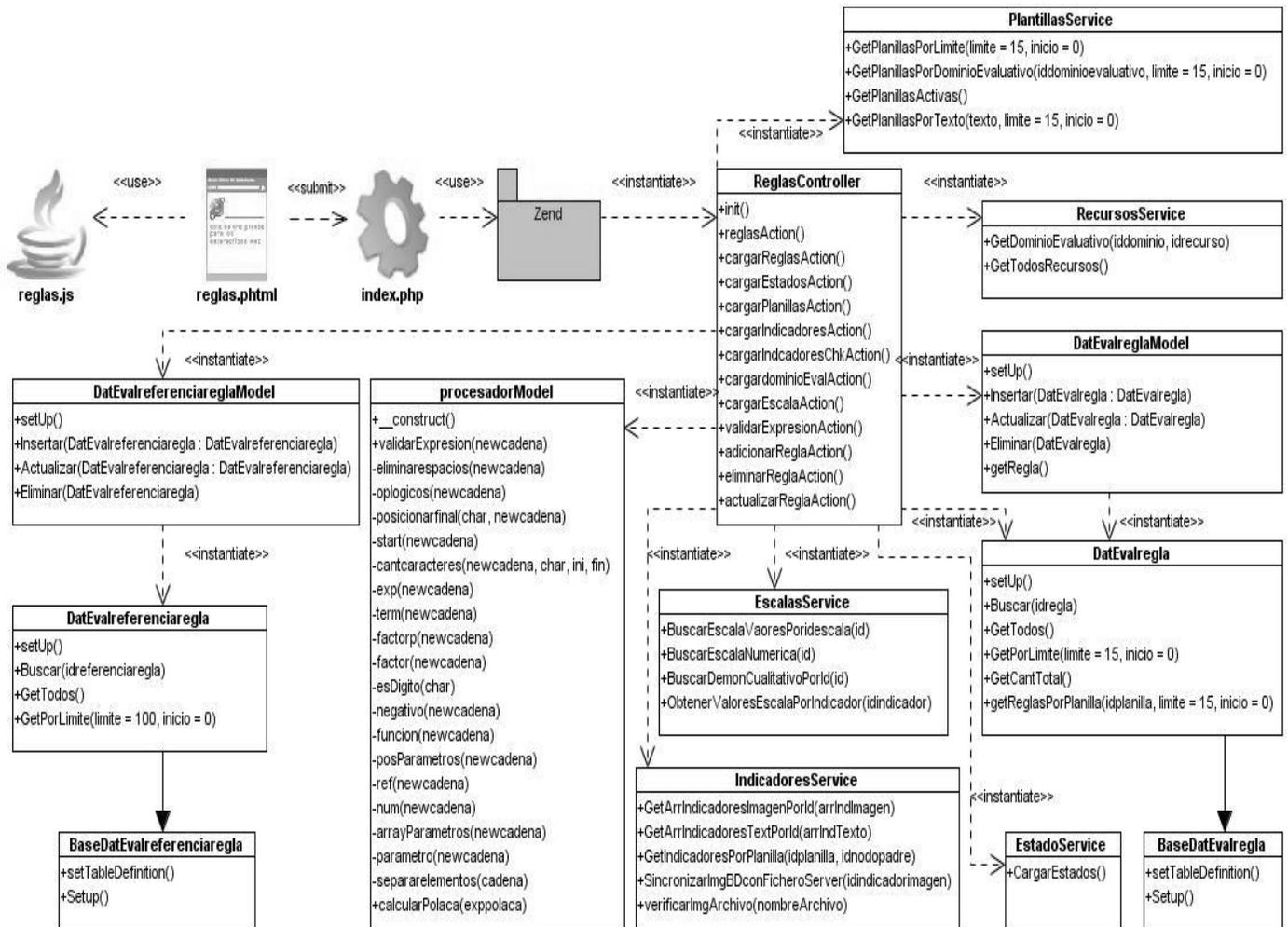


Figura 3.6 Diagrama de clases del diseño: Gestionar reglas.

Para obtener más información de los diagramas de clases del diseño remitirse al anexo[4].

3.4. Diagramas de interacción (Secuencia)

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo. A continuación se muestra el diagrama de secuencia para adicionar un modelo matemático.

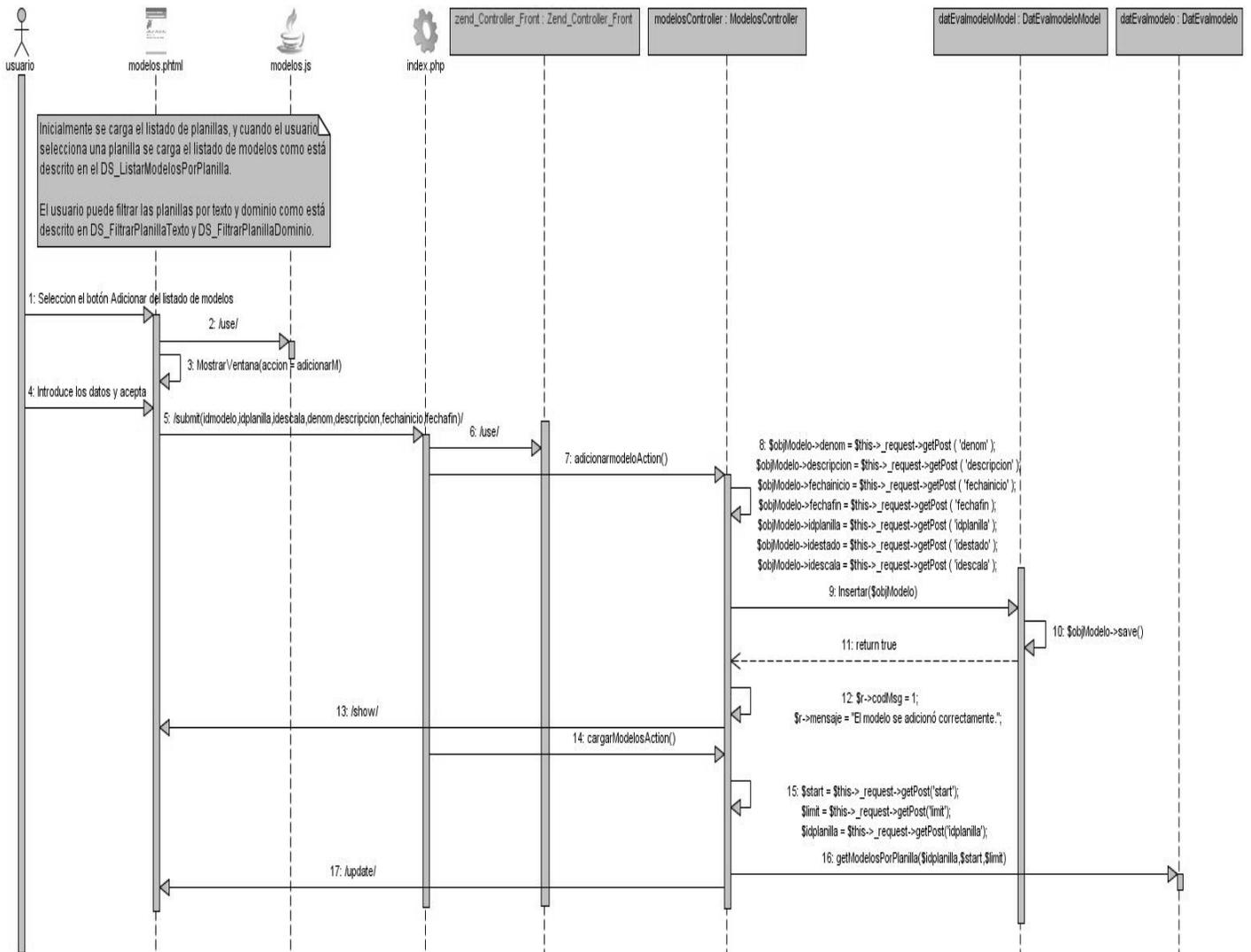


Figura 3: Diagrama de secuencia del requisito Gestionar modelos matemáticos, escenario Adicionar modelo matemático.

Para consultar los demás diagramas de secuencia consulte el Anexo [5]

3.5. Diseño de la base de datos (BD)

3.5.1. Diagrama entidad relación de la BD

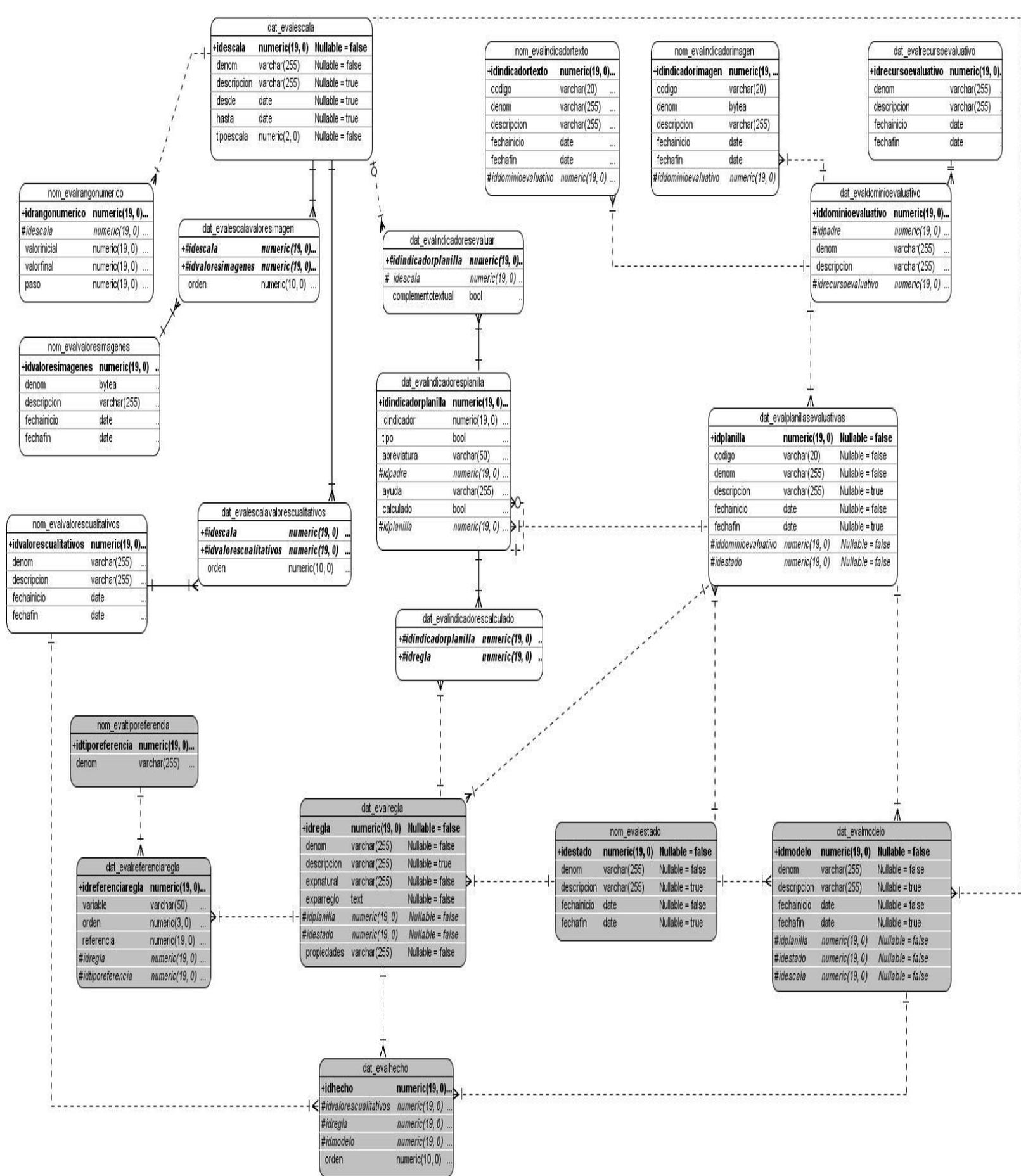


Figura 3.8 Diagrama entidad relación de la base de datos.

Para ver la descripción de las tablas de la base de datos consulte el Anexo [6]

3.6. Conclusiones

En este capítulo se alcanzó la definición de la arquitectura del sistema especificando las clases correspondientes al patrón MVC. Se realizaron los diagramas de clases del diseño con estereotipos web aplicando los patrones de diseño descritos así como los diagramas de secuencia y el diagrama de entidad relación de la base de datos.

Capítulo 4. Implementación y prueba

4.1. Introducción

En el presente capítulo se comienza la implementación con los resultados obtenidos del diseño. Se describe cómo los elementos del modelo del diseño se implementan en términos de componentes, se muestra el mapa de navegación de los componentes “Regla” y “Modelo matemático” y algunos de los diseños de casos de pruebas que apoyarán las pruebas de caja negra.

4.2. Diagrama de componentes

Un diagrama de componentes representa la estructura de un sistema o subsistema de software en componentes y la relación entre estos. El siguiente diagrama de componentes muestra de forma global los paquetes de componentes que implementan el subsistema de evaluaciones dentro del módulo capital humano.

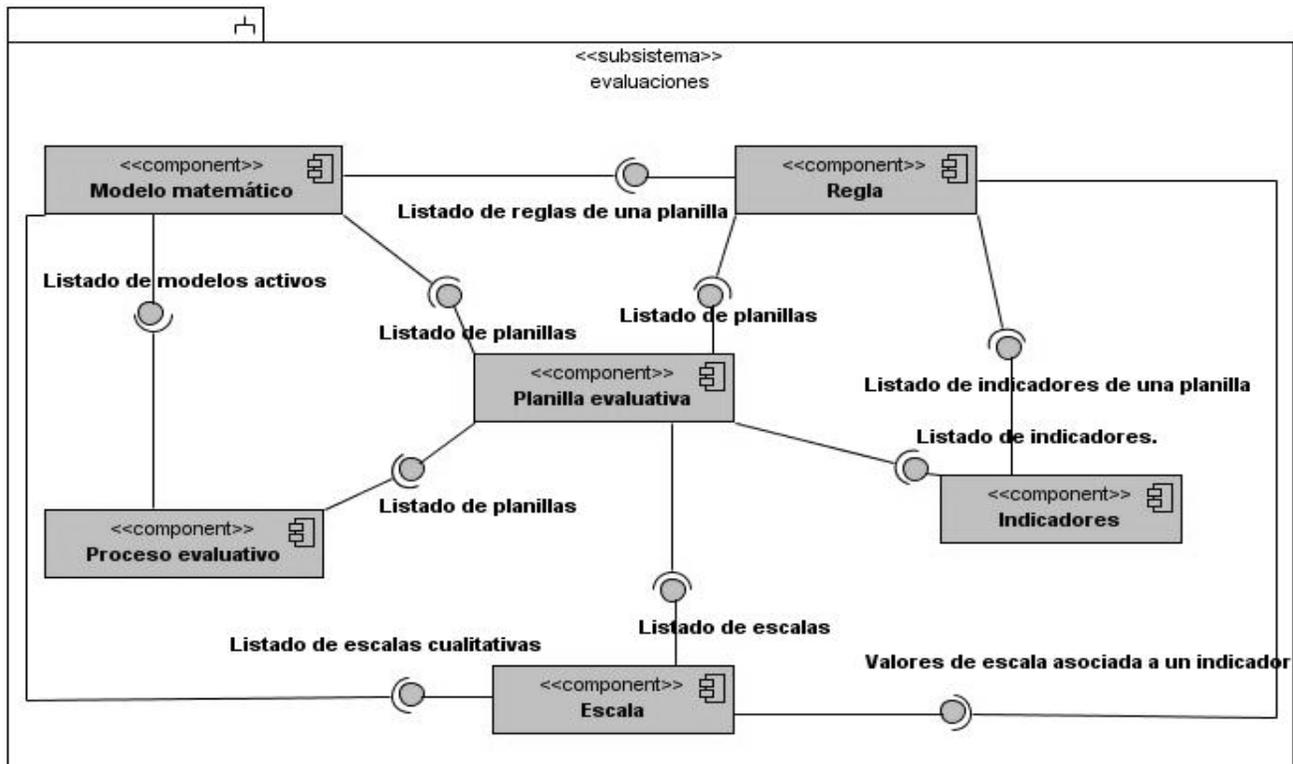


Figura 4.9 Diagrama de componentes global.

4.3. Matriz de integración de componentes interna

		Componentes Internos			
Componentes Internos	Modelos matemáticos	Reglas	Planillas	Indicadores	Escalas
Proceso evaluativo	Listado de modelos activos	—	Listado de planillas	—	—
Modelos matemáticos	—	Listado de reglas de una planilla	Listado de planillas	—	Listado de escalas cualitativas
Reglas	—	—	Listado de planillas	Listado de indicadores de una planilla	Valores de una escala asociada a un indicador
Planillas	—	—	—	Listado de indicadores	Listado de escalas

Tabla 4.1 Matriz de integración de componentes interna

4.4. Prototipos de interfaz de usuario

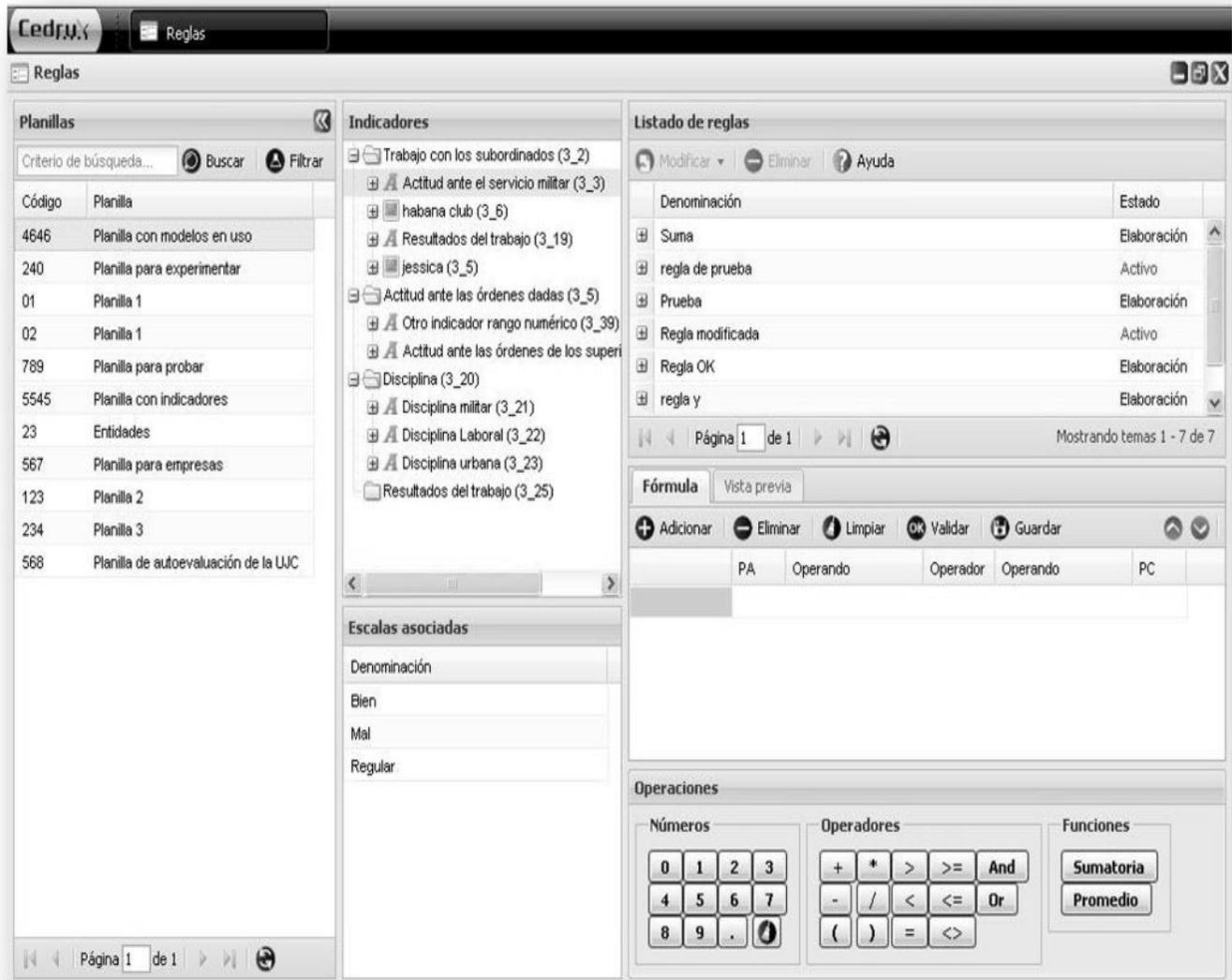


Figura 4.10 Prototipo de interfaz de usuario Gestionar regla.

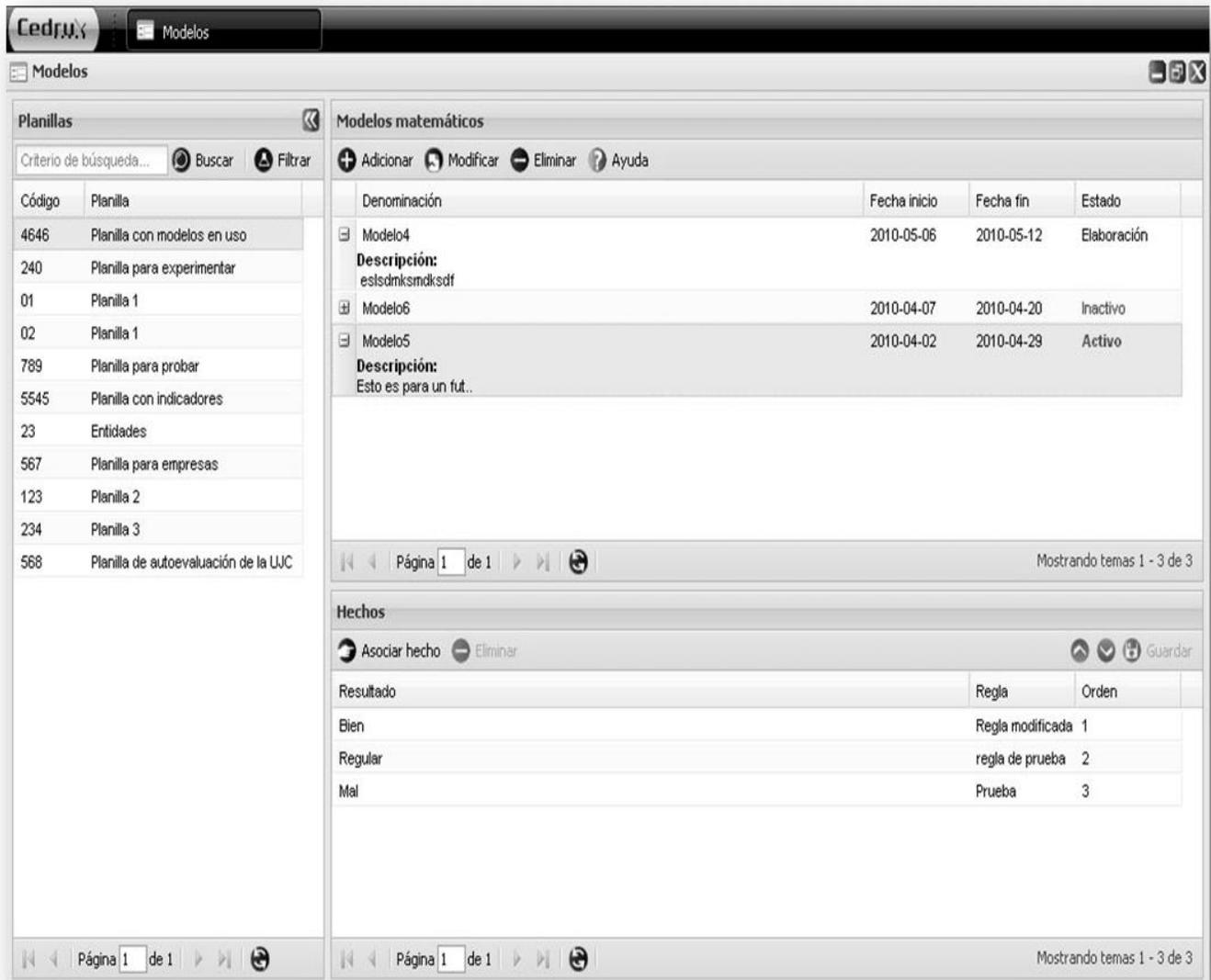


Figura 4.11 Prototipo de interfaz de usuario Gestionar modelo matemático.

Para consultar los demás prototipos de interfaz consultar el Anexo [3].

4.5. Mapa de navegación

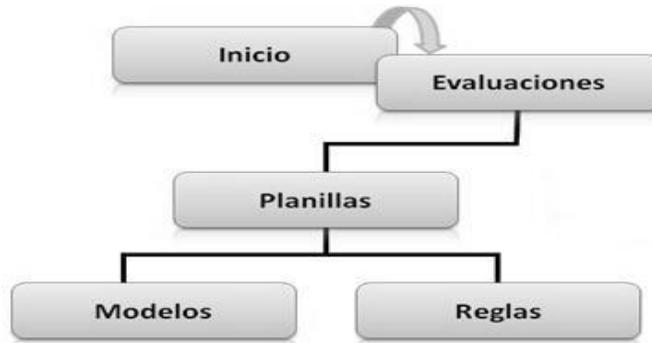


Figura 4. Mapa de navegación.

4.6. Validación del sistema

4.6.1. Pruebas de caja negra

Las pruebas de caja negra son aquellas que se enfocan directamente en el exterior del módulo (pruebas a nivel de interfaz de usuario), sin importar el código. Este tipo de prueba no es aplicable a los módulos que trabajan en forma transparente al usuario.

4.6.2. Diseño de casos de prueba

Los casos de prueba son un conjunto de condiciones o variables bajo las cuales el analista determinará si el requisito de una aplicación es parcial o completamente satisfactorio.

4.6.2.1. Diseño del caso de prueba Adicionar modelo matemático

Condiciones de ejecución:

- El usuario debe estar autenticado en el sistema.
- Se debe ir a **Inicio/Evaluaciones/Planillas/Modelos.**

Requisitos a probar

Nombre del requisito	Descripción general	Escenarios de pruebas	Flujo del escenario
Gestionar modelo matemático	El objetivo de este requisito es gestionar los modelos matemáticos de una planilla evaluativa para poder obtener un resultado y a partir de ahí dar un criterio de evaluación. Además, permite asociar hechos a los modelos matemáticos con el objetivo de calificar con criterios cualitativos las reglas que se le apliquen a una planilla evaluativa.	EP 1.1 Adicionar modelo matemático. Correcto	<ul style="list-style-type: none"> • Se selecciona la opción Evaluaciones/Planillas/ Modelos en el menú principal del sistema. • Se muestra la interfaz Modelos con la parte derecha deshabilitada. • Se selecciona una planilla evaluativa. • Se cargan los modelos correspondientes a la planilla seleccionada y se habilita la parte derecha de la interfaz. • Se oprime el botón Adicionar. • Se muestra la ventana de adicionar modelo matemático, con los campos: denominación, descripción, fecha de inicio, fecha fin, y con el estado por defecto Elaboración. • El usuario introduce los datos. • Se oprime el botón Aceptar. • El sistema muestra un mensaje a modo de información diciendo que el modelo se ha insertado con éxito.

		<p>EP 1.2: Adicionar modelo matemático Incorrectamente.</p>	<ul style="list-style-type: none"> • Se selecciona la opción Evaluaciones/Planillas/ Modelos en el menú principal del sistema. • Se muestra la interfaz Modelos con la parte derecha deshabilitada. • Se selecciona una planilla evaluativa. • Se cargan los modelos correspondientes a la planilla seleccionada y se habilita la parte derecha de la interfaz. • Se oprime el botón Adicionar. • Se muestra la ventana de adicionar modelo matemático, con los campos: denominación, descripción, fecha de inicio, fecha fin, y con el estado por defecto Elaboración. • El usuario deja campos en blanco o introduce el nombre de un modelo ya existente. • Se oprime el botón Aceptar. • Si se dejaron campos en blanco el sistema marca en color rojo los campos que son obligatorios. • Si introduce el nombre de un modelo ya existente muestra un mensaje de error diciendo que el modelo ya existe.
--	--	---	---

		<p>EP 1.3: Cancelar operación.</p>	<ul style="list-style-type: none"> • Se selecciona la opción Planillas/ Modelos en el menú principal del sistema. • Se muestra la interfaz Modelos con la parte derecha deshabilitada. • Se selecciona una planilla evaluativa. • Se cargan los modelos correspondientes a la planilla seleccionada y se habilita la parte derecha de la interfaz. • Se oprime el botón Adicionar. • Se muestra la ventana de adicionar modelo matemático, con los campos: denominación, descripción, fecha de inicio, fecha fin y el estado. • El usuario introduce los datos. • Se oprime el botón Cancelar. • El sistema cancela las operaciones.
		<p>EP 1.4: Ver ayuda</p>	<ul style="list-style-type: none"> • Se oprime el botón Ayuda. • Se muestra la ayuda detallada de la funcionalidad.

Tabla 4.2 Requisitos a probar.

Juego de datos a probar

Id del escenario	Escenario	Denominación	Fecha de inicio	Fecha fin	Escala	Descripción	Respuesta del sistema	Resultado de la prueba
EP 1.1	Adicionar modelo matemático	V(852mod459)	V(5/05/2010)	V(vacío)	V (Si selecciona una escala.)	V(@2**Estoes un modelo de prueba **2@)	Se adiciona el modelo matemático y se muestra el mensaje de Información.	Satisfactoria
		V(Modelos58)	V(6/10/2010)	V(9/11/2010)	V (Si selecciona una escala.)	V (Vacío.)	Se adiciona el modelo matemático y se muestra el mensaje de Información.	Satisfactoria
EP 1.2	Adicionar modelo matemático incorrectamente.	I(vacío)	I(vacío)	V(vacío)	I(vacío)	V(vacío)	Se muestra el campo en color rojo y se muestra un mensaje indicando el campo que sea obligatorio llenar.	Satisfactoria
		I(vacío)	V(5/08/2010)	V(9/12/2010)	V (Si selecciona una escala.)	V (Esto es un ejemplo).		Satisfactoria
		V(Modeloabv44)	I(vacío)	V(18/7/2010)	V (Si selecciona una escala.)	V(vacío)		

		I(vacío)	I(vacío)	V(23/9/2010)	V (Si selecciona una escala.)	V(ssjhdsslos)		
		V(aksjajk0e33)	I(vacío)	V(19/2/2010)	I(vacío)	V(asksjsds//55****)		
		V(2Modelitoxz)	I(ddd/er/85)	I(87sss)	V(Si selecciona una escala)	V(¡#%\$%^&*)	Se muestran en color rojo los campos que tengan formato incorrecto.	Satisfactoria
		V(896666)	V(9/7/2010)	I(kjmio sdfj89343/*)	V(Si selecciona una escala)	V(2OAKAKA)		Satisfactoria
		V(sskmsak888)	I(aa89652)	V(26/1/2010)	V(Si selecciona una escala).			
EP 1.3	Cancelar operación.	NA	NA	NA	NA	NA	El sistema cancela las operaciones	Satisfactoria
		V(modelo487)	V(13/4/2010)	V(15/6/2010)	V (Si selecciona una escala).	V(marcado)	El sistema cancela las operaciones	Satisfactoria
		Cualquier combinación	Cualquier combinación	Cualquier combinación	Cualquier combinación	Cualquier combinación	El sistema cancela las operaciones	Satisfactoria

EP 1.4	Ver ayuda.	NA	NA	NA		NA	Se muestra la ayuda detallada de la funcionalidad.	Satisfactoria
--------	------------	----	----	----	--	----	--	---------------

Tabla 4.3 Juego de datos a probar.

Para consultar los diseños de casos de prueba remitirse al Anexo [7].

Con el objetivo de encontrar la mayor cantidad de anomalías en los componentes de modelos y reglas, el equipo de desarrollo diseñó y ejecutó 8 diseños de casos de pruebas, obteniendo 5 no conformidades no significativas en la primera iteración, una vez solucionadas en la segunda iteración se obtuvo un resultado satisfactorio para cada una de las combinaciones de datos por escenario.

4.7. Conclusiones

En este capítulo quedó descrito como los elementos del diseño se implementaron en términos de componentes, se obtuvo además la matriz de integración de componentes interna, los prototipos de interfaz de usuario, el mapa de navegación así como el diseño de casos de prueba.

Conclusiones

Con la realización de este trabajo se elaboró el estado del arte, realizando un estudio en el ámbito nacional e internacional de las herramientas que automatizan el proceso de análisis estadístico de los resultados de las evaluaciones y de gestión del capital humano. Se modeló el proceso del negocio “Analizar resultado de la evaluación” para las entidades del país, lográndose de este modo un mayor entendimiento de la problemática por parte de los clientes y los analistas del sistema. Se realizó el diseño de una solución, que satisface las necesidades de los usuarios.

Se logró implementar y validar el módulo de análisis estadístico de los resultados de las evaluaciones durante el proceso evaluativo, que incorporado al subsistema Capital Humano del proyecto ERP Cuba, facilitará el análisis estadístico del resultado de las evaluaciones y contribuirá a satisfacer las necesidades de las entidades del país. Este constituirá un aporte social significativo en las entidades tanto civiles como de las FAR, pues reducirá en gran medida parte del trabajo manual que hasta hoy se viene realizando y proporcionará una mayor eficiencia en el análisis del resultado de las evaluaciones.

Recomendaciones

Con el objetivo de mejorar la solución planteada se agregan las siguientes recomendaciones:

Continuar un desarrollo iterativo e incremental de los componentes Modelos y Reglas obteniendo como resultado un incremento en cada iteración que permita perfeccionar el funcionamiento de dichos componentes.

Incorporar al componente de Reglas un mayor nivel de complejidad que permita construir fórmulas matemáticas con más operadores y funciones predefinidas.

Se recomienda el uso del componente de Reglas y Modelos para otras líneas de producción, además de la implantación del producto y la revisión por especialistas de calidad del mismo.

Bibliografía

1. **Chiavenato, Idalberto.** *Gestión del Talento Humano.* [ed.] Lily Solano Arévalo. Bogotá : Quebecor World Bogotá S.A., 2002. ISBN: 958-41 -0288-5.
2. **Rolando Alfredo Hernández León, Sayda Coello González.** *El paradigma cuantitativo de la investigación científica.* Ciudad de la Habana : Editorial Universitaria, 2002. ISBN: 959-16-0343-6.
3. **UCID.** *Proceso de Desarrollo y Gestión de Proyectos de Software.* Ciudad de la Habana : s.n., 2009.
4. **UCID, ERP-Cuba UCI.** *Manual del Marco de Trabajo (ARQUCID).Proyecto ERP Cuba.* Ciudad de la Habana : s.n., 2009.
5. *Vistos desde otros horizontes. Revista de Software libre ATIX.* **Esteban Saavedra López.** 2009.
6. **Shea Frederick, Colin Ramsay, Steve 'Cutter' Blades.** *Learning Ext JS.* [ed.] Swapna V. Verlekar. Birmingham : Packt Publishing Ltd., 2008. ISBN 978-1-847195-14-2.
7. **Mehdi Achour, Friedhelm Betz, Antony Dovgal, Nuno Lopes, Hannes Magnusson.** *php. php.* [En línea] © 1997-2010 the PHP Documentation Group, 11 de abril de 2010. [Citado el: 6 de abril de 2010.] <http://www.php.net/manual/es/>.
8. **Jonathan H. Wage, Roman S. Borschel,Guilherme Blanco, Benjamin Eberlei.** *doctrine. Doctrine ORM for PHP Table of Contents.* [En línea] Doctrine Project, 11 de 2009. [Citado el: 6 de abril de 2010.] <http://www.doctrine-project.org/projects/orm/1.2/docs/manual/en>.

Referencia Bibliográfica

1. **Cartaya, Alfredo Morales.** *Resolución No. 21.* Ciudad de la Habana : s.n., 2007.
2. **FAR.** Manual de procedimientos. Ciudad de la Habana : s.n., 2008
3. **Lempert.** Lempert S.A. [En línea] [Citado 1 de febrero de 2010.] <http://www.lempert.com.ar>.
4. **Wilsoft.** Wilsoft. [En línea] [Citado el: 1 de febrero de 2010.] <http://www.wilsoft-la.com/qtraining.htm>.
5. **Asesorías e Inversiones Comunidades Ciudadanas S.A y miembro de la Red de Diarios Ciudadanos.** El repuerto . [En línea] 31 de diciembre de 2008. [Citado el: 1 de febrero de 2010.] <http://www.elrepuerto.cl/admin/render/noticia/16818>.
6. **MINFAR.** Modelo de Negoco SIAC. Ciudad de la Habana : s.n., 2009.
7. **Milenium.** InformáticaMilenium. [En línea] 15 de enero de 2007. [Citado el: 2 de febrero de 2010.] <http://www.informaticamilenium.com.mx/Paginas/espanol/sitioweb.htm>.
8. **DIMAGIN.** DIMAGIN Web Development. [En línea] [Citado el: 2 de febrero de 2010.] http://www.dimagin.net/es/contenido.php?t_id=6.
9. **Jesús Vegas.** El Servidor Web. [En línea] 21 de marzo de 2002. [Citado el: 2 de febrero de 2010.] <http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node20.html>
10. **Ciberaula.** Ciberaula. [En línea] [Citado el: 29 de enero de 2010.] http://linux.ciberaula.com/articulo/linux_apache_intro/.
11. **IIS.** IIS. [En línea] [Citado el: 29 de enero de 2010.] <http://www.iis.net/>.
12. **ACME.** ACME Laboratories. [En línea] [Citado el: 29 de enero de 2010.] <http://www.acme.com/software/thttpd/>
13. **Juan R. Pozo.** HTML con Clase. [En línea] [Citado el: 3 de febrero de 2010.] <http://html.conclase.net/w3c/html401-es/intro/intro.html>.
14. **W3C.** W3C. [En línea] [Citado el: 3 de febrero de 2010.] <http://www.w3c.es/divulgacion/guiasbreves/tecnologiasXML>.
15. **Eguíluz Pérez, Javier.** Introducción a JavaScript. [En línea] [Citado el: 25 de enero de 2010.] <http://www.librosweb.es/javascript/capitulo1.html>.

16. **Eguíluz Pérez, Javier.** Introducción a CSS. [En línea] [Citado el: 25 de enero de 2010.] <http://librosweb.es/css/capitulo1.html>.
17. **Eguíluz Pérez, Javier.** Introducción a AJAX. [En línea] [Citado el: 7 de febrero de 2010.] <http://librosweb.es/ajax/capitulo1.html>.
18. **Álvarez, Rubén.** desarrolloweb.com. [En línea] [Citado el: 29 de enero de 2010.] <http://www.desarrolloweb.com/articulos/303.php>.
19. **Scribd.** Lenguaje de programación para Web: ASP. [En línea] [Citado el: 29 de enero de 2010.] <http://www.scribd.com/doc/2413026/Lenguaje-de-programacion-para-Web-ASP>.
20. **Alvarez, Sara.** DesarrolloWeb.com. [En línea] 31 de julio de 2007. [Citado el: 3 de febrero de 2010.] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
21. **MySQL Enterprise.** MySQL. [En línea] [Citado el: 4 de febrero de 2010.] <http://www.mysql.com/>.
22. **ArPug.** ArPUG (Grupo de usuarios PostgreSQL de argentina). [En línea] [Citado el: 29 de enero de 2010.] <http://www.arpug.com.ar/trac/wiki/TutorialPostgreSql>.
23. **Guiarte Multimedia S.L.** desarrolloweb.com. [En línea] [Citado el: 4 de febrero de 2010.] <http://www.desarrolloweb.com/articulos/840.php>.
24. **Microsoft.** Microsoft. [En línea] 25 de noviembre de 2005. [Citado el: 4 de febrero de 2010.] <http://www.microsoft.com/spain/sql/productinfo/features/top30features.mspx>.
25. **Ext JS.** Ext JS. [En línea] [Citado el: 4 de febrero de 2010.] <http://www.extjs.com>.
26. **Doctrine Project.** doctrine. [En línea] [Citado el: 5 de febrero de 2010.] <http://www.doctrine-project.org/>.
27. **Jacobson.** El Lenguaje Unificado de Modelado. Addison-Wesley. I.B(1999).
28. **Martin Owen, Jog Raj.** Business Process Management Notation. [En línea] [Citado el: 6 de febrero de 2010.] http://www.bpmn.org/Documents/6AD5D16960.BPMN_and_BPM.pdf.
29. **Free Download Manager.ORG.** Free Download Manager. [En línea] [Citado el: 6 de febrero de 2010.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/
30. **Grupo Soluciones Innova S.A.** Rational Rose Enterprise. GSiNNOVA Grupo Soluciones. [En línea] [Citado el: 7 de febrero de 2010.] <http://www.rational.com.ar/herramientas/roseenterprise.html>.

31. **Zend Corporation.** Zend Studio for Eclipse Proven Zend Studio technology powered by Eclipse. [En línea] [Citado el: 8 de febrero de 2010.] <http://static.zend.com/topics/Studio-for-Eclipse-Datasheet-600.pdf>.
32. **Media Ingea SL .** UP to DOWN. [En línea] [Citado el: 9 de febrero de 2010.] <http://aptana-studio.uptodown.com/>.
33. **PostgreSQL Global Development Group.** PostgreSQL. [En línea] 1 de junio de 2005 [Citado el: 9 de febrero de 2010.] <http://www.postgresql.org/about/news.329>.
34. **Real Academia Española.** *Real Academia Española.* [En línea] [Citado el: 25 de 3 de 2010.] http://buscon.rae.es/drael/SrvltConsulta?TIPO_BUS=3&LEMA=proceso.
35. **Sanzano, Fidel Jimenez.** *Mapa de procesos de negocio: "Evaluación del desempeño nivel 2". Capital humano. Centro de soluciones de Gestión.* 2010.
36. *Conferencia 6. Fase de inicio. Disciplina de Requisitos. Semana:7. UCI.* Ciudad Habana : s.n., 2008-2009.
37. *Arquitectura y Patrones de diseño. Ingeniería de Software. Actividad:5 Tema 1 Semana:3. UCI.* Ciudad Habana : s.n., 2008-2009.

Glosario de términos

Aplicaciones Web: Se le 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 (browser). También se les denomina aplicaciones de servidor pues los cómputos y el procesamiento de información se realizan en el servidor. Por lo cual el cliente para interactuar e intercambiar información solo necesita de un ordenador con un navegador web.

Análisis estadístico: La estadística es una ciencia con base matemática referente a la recolección, análisis e interpretación de datos, que busca explicar condiciones regulares en fenómenos de tipo aleatorio.

Browser: Navegador web (del inglés, web browser) es un programa que permite visualizar la información que contiene una página web. El navegador interpreta el código, HTML generalmente, en el que está escrita la página web y lo presenta en pantalla permitiendo al usuario interactuar con su contenido y navegar hacia otros lugares de la red mediante enlaces o hipervínculos.

Capital intelectual: Es el conocimiento intelectual de esa organización, la información intangible (que no es visible, y por tanto, no está recogida en ninguna parte) que posee y que puede producir valor.

Capital Humano: Conjunto de conocimientos, entrenamientos y habilidades poseídas por las personas que las capacitan para realizar labores productivas con distintos grados de complejidad y especialización. Es el valor resultante de la aportación de las personas a la organización.

Entidad: Una organización o una sección de una organización que, para fines contables, se distingue de otras organizaciones como una unidad económica por separado. Es el nombre o razón social con el que se conoce a instituciones públicas y privadas, organismos nacionales e internacionales y empresas.

Escala: Una escala está conformada por un conjunto de valores. Las escalas pueden ser de diferentes tipos: escalas cualitativas, escalas de imagen y escalas numéricas.

FTP: Protocolo de Transferencia de Archivos, en informática, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP (Transmission Control Protocol), basado en la

arquitectura cliente-servidor. Desde un equipo cliente se puede conectar a un servidor para descargar archivos desde él o para enviarle archivos, independientemente del sistema operativo utilizado en cada equipo.

Framework: En el desarrollo de software, un framework es una estructura conceptual y tecnológica de soporte definida, normalmente con artefactos o módulos de software concretos, con base en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros programas para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

GNU/Linux: GNU/Linux es uno de los términos empleados para referirse a la combinación del núcleo o kernel libre similar a Unix denominado Linux, que es usado con herramientas de sistema GNU. Su desarrollo es uno de los ejemplos más prominentes de software libre; todo su código fuente puede ser utilizado, modificado y redistribuido libremente por cualquiera bajo los términos de la GPL (Licencia Pública General de GNU) y otra serie de licencias libres.

Hechos: Son reglas matemáticas que están conformados por un valor cualitativo y una ecuación o regla matemática para clasificar el resultado de una planilla evaluativa. Los hechos se asocian a los modelos matemáticos con el objetivo de clasificar el resultado de un modelo matemático asociado a una planilla evaluativa.

HTTP: Hypertext Transfer Protocol (en español protocolo de transferencia de hipertexto) es el protocolo usado en cada transacción de la World Wide Web, es decir, este protocolo define la comunicación entre el software navegador de internet y el software servidor que publica las páginas consultadas.

HTTPS: Hypertext Transfer Protocol Secure (en español: Protocolo seguro de transferencia de hipertexto), más conocido por sus siglas HTTPS, es un protocolo de red basado en el protocolo HTTP, destinado a la transferencia segura de datos de hipertexto, es decir, es la versión segura de HTTP.

IOC: Inversión de control (Inversion of Control en inglés, IOC) es un método de programación en el que el flujo de ejecución de un programa se invierte respecto a los métodos de programación tradicionales, en los que la interacción se expresa de forma imperativa haciendo llamadas. Tradicionalmente el

programador especifica la secuencia de decisiones y procedimientos que pueden darse durante el ciclo de vida de un programa mediante llamadas a funciones. En su lugar, en la inversión de control se especifican respuestas deseadas a sucesos o solicitudes de datos concretas, dejando que algún tipo de entidad o arquitectura externa lleve a cabo las acciones de control que se requieran en el orden necesario y para el conjunto de sucesos que tengan que ocurrir.

Modelo matemático: Un modelo matemático es uno de los tipos de modelos científicos, que emplea algún tipo de formulismo matemático para expresar relaciones, proposiciones sustantivas de hechos, variables, parámetros, entidades y relaciones entre variables y/o entidades u operaciones, para estudiar comportamientos de sistemas complejos ante situaciones difíciles de observar en la realidad.

Open Source: Código abierto es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre.

Planilla evaluativa: Planilla evaluativa es un documento que recoge una serie de indicadores por los cuales se va a evaluar determinado recurso.

Proceso evaluativo: Implica una serie de pasos, que no son independientes entre sí, están interrelacionados y se condicionan unos a otros, es decir, la evaluación actúa como un sistema. Esto es decisivo porque mediante la evaluación se puede obtener la información necesaria que se necesite para otros procesos.

Regla: Es una fórmula matemática que puede estar compuesta por operadores aritméticos, operadores lógicos, indicadores, escalas o a su misma vez por otras reglas.

SMTP: Protocolo Simple de Transferencia de Correo, es un protocolo de la capa de aplicación. Protocolo de red basado en texto utilizado para el intercambio de mensajes de correo electrónico entre computadoras u otros dispositivos.

Software: Término en inglés para describir a los programas de computación. 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 tareas específicas; en contraposición a los componentes físicos del sistema, llamados hardware.

SSL: Secure Sockets Layer -Protocolo de Capa de Conexión Segura- (SSL) y Transport Layer Security -Seguridad de la Capa de Transporte- (TLS), su sucesor, son protocolos criptográficos que proporcionan comunicaciones seguras por una red, comúnmente Internet.

Web: World Wide Web o simplemente Web, cuya traducción podría ser la Red Global Mundial, abreviado como “WWW” o “3W” es el universo de información accesible a través de Internet, una fuente inagotable del conocimiento humano. Su característica sobresaliente es el texto remarcado, un método para referencias cruzadas instantáneas que permite vincular un texto disponible en la Web con otros documentos, eso indica que usted puede realizar un clic para ser transferido a otro sitio. La exploración en el Web se realiza por medio de un software especial denominado Browser o Explorador, con el uso de este en la Web se tiene acceso a millones de páginas de información.