

**Universidad de las Ciencias Informáticas
Facultad 3**



**Título: “Análisis y diseño del proceso Competencias
laborales del subsistema Capital Humano del sistema
CedruX”**

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor: Mavis Arias Vázquez

Tutores: MSc. Donel Vázquez Zambrano

Ing. Fidel Jiménez Sanzano

Junio 2011



*“La soberanía del hombre está oculta en la
dimensión de sus conocimientos.”*

Ernesto Ché Guevara

DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconozco 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 _____.

Mavis Arias Vázquez

Firma del Autor

MSc. Donel Vázquez Zambrano

Firma del Tutor

Ing. Fidel Jiménez Sanzano

Firma del Tutor

DATOS DE CONTACTO

Datos de los tutores:

MSc. Donel Vázquez Zambrano

Profesor Instructor graduado de Ingeniería en Ciencias Informáticas con Título de Oro y de Máster en Gestión de Proyectos Informáticos en la Universidad de las Ciencias Informáticas (UCI). Ha cursado y aprobado diecisiete cursos de postgrado. Ha participado en cuatro eventos nacionales y uno internacional como ponente o autor. Tiene en su haber nueve publicaciones científicas.

Correo: dvz@uci.cu

Ing. Fidel Jiménez Sanzano

Graduado de Ingeniería en Ciencias Informáticas con Título de Oro en el año 2009. Ha cursado y aprobado doce cursos de postgrado. Ha participado en 6 eventos nacionales como ponente o autor. Tiene en su haber 2 publicaciones científicas. Actualmente se desarrolla como analista de la línea Capital Humano del centro CEIGE de la facultad 3.

Correo: sanzano@uci.cu

AGRADECIMIENTOS

En primer lugar le quisiera agradecer a mis padres por todo el amor, comprensión, apoyo, y dedicación que me han brindado a lo largo de toda mi vida, por estar siempre presentes, en los buenos y malos momentos, por ser mi luz y mi guía, y por confiar en mí, demostrándome día a día lo afortunada que soy al tener unos padres tan maravillosos.

A mi papito en especial por toda su ayuda con la tesis, por las malas noches que pasamos juntos, por tenerme paciencia; sin él no hubiese podido salir adelante.

A mi mamita por todos sus consejos que han hecho de mí una mejor persona.

A mi hermanita Rosi por mortificarme tanto y alegrarme la vida con sus atrevimientos.

A mi hermano Yúnior, por estar siempre pendiente de mí a pesar de lo lejos que nos encontramos.

A mi novio Yoandry por todo su amor, por apoyarme en los momentos que más lo necesité y alentarme a seguir adelante.

A la familia de mi novio que se ha convertido en mi segunda familia, siempre al tanto de mis cosas.

A Yamilka e Indiana por soportarme durante todos estos años y brindarme su amistad, compartiendo conmigo buenos y malos momentos.

A Leisys, Leidys y Osbel por su gran amistad y por brindarme su ayuda siempre que lo necesité.

A mis compañeras de tesis, en especial a Aylín que la volví loca y siempre tuvo tiempo de ayudarme cada vez que lo necesitaba a pesar de estar también complicada con su tesis, gracias por todo.

A mis tutores, en especial a Fidel que estuvo desde el principio ayudándome en todo lo que me hizo falta.

A mi tribunal y oponente por sus críticas constructivas y su ayuda.

A mi familia por todo su cariño y apoyo.

A todas las personas que he conocido a lo largo de estos cinco años, que me han demostrado su amistad y han contribuido de una forma u otra a hacerme crecer como persona y como profesional.

A todos... muchas gracias

DEDICATORIA

A mis padres que significan todo para mí.

A mi hermanita por ser tan especial.

A mis abuelitos que siempre están pendientes de mí.

A mi familia por todo su cariño y apoyo.

A mi novio por darme tanto amor, fuerzas para seguir y estar ahí siempre que lo necesito.

A mis amistades por compartir conmigo buenos y malos momentos.

A todo el que este trabajo le pueda servir.

RESUMEN

La competencia laboral se ha convertido en los últimos años en un marco de referencia para la organización y el desarrollo del personal en las organizaciones actuales, de ahí que la aplicación de este concepto a la Gestión del Capital Humano se haya extendido actualmente en todo el mundo empresarial, y no son pocos los expertos, especialistas, investigadores y autoridades en general que sobre el tema de las competencias laborales coinciden en opinar que las mismas constituyen una nueva y eficaz alternativa para incrementar el rendimiento laboral y la motivación de los colaboradores.

El país, urgido de transformaciones inmediatas que permitan fortalecer la gestión de las entidades nacionales e incrementar de forma sostenida su productividad y eficiencia, ha abogado también por la aplicación del modelo de Gestión por Competencias y desde hace varios años enfoca sus esfuerzos en el desarrollo de su primer Sistema Integral de Gestión Cedrux que contempla como uno de los procesos claves dentro del Subsistema de Capital Humano, el proceso Competencias Laborales aún por desarrollar.

En el presente trabajo se hace un análisis del proceso antes mencionado y como resultado se provee un diseño adecuado que podrá ser utilizado por los desarrolladores para su implementación e integración al subsistema de Capital Humano del sistema Cedrux.

PALABRAS CLAVES

Capital Humano, Cedrux, Competencias Laborales, Gestión por Competencias.

Tabla de contenido

INTRODUCCIÓN 1

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA 6

 1.1. Introducción 6

 1.2. Gestión por competencias 6

 1.2.1. *Concepto de competencias* 12

 1.2.2. *Clasificación de las competencias* 13

 1.2.3. *Evaluación por competencias* 15

 1.3. Modelo de desarrollo orientado a componentes 17

 1.3.1. *Características* 17

 1.3.2. *Fases del ciclo de vida del proyecto* 18

 1.3.3. *Actividades* 19

 1.3.4. *Artefactos que se generan* 21

 1.3.5. *Tecnología* 22

 1.4. Arquitectura del sistema 22

 1.5. Conclusiones del capítulo 25

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA 26

 2.1. Introducción 26

 2.2. Descripción del proceso del negocio 26

 2.3. Técnicas y métodos empleados en la captura de requisitos 27

 2.4. Modelo conceptual 29

 2.5. Patrones de análisis 30

 2.5.1. *Patrón CRUD* 30

 2.5.2. *Nombres que revelan la intención (IntentionRevealingName)* 31

 2.5.3. *Escenario más Fragmentos (ScenarioPlusFragments)* 31

 2.5.4. *Preciso y legible* 31

 2.5.5. *Completar una única meta* 31

2.5.6. Alternativas Exhaustivas (<i>Exhaustive Alternatives</i>).....	31
2.5.7. Condiciones detectables (<i>Detectable Conditions</i>).....	31
2.5.8. Pasos Nivelados (<i>Leveled Steps</i>).	32
2.6. Requisitos funcionales.....	32
2.6.1. Especificaciones de requisitos.....	33
2.7. Conclusiones del capítulo.....	36
CAPÍTULO 3: DISEÑO DEL SISTEMA.....	37
3.1. Introducción.....	37
3.2. Patrones de diseño.....	37
3.2.1. Patrones GRASP.....	37
3.2.2. Patrones GoF (<i>Gang of Four</i>).	38
3.3. Diagramas de clases del diseño.....	39
3.4. Diagrama entidad relación.....	51
3.5. Diagrama de componentes.....	52
3.6. Conclusiones del capítulo.....	54
CAPÍTULO 4: VALIDACIÓN DE LOS RESULTADOS.....	55
4.1. Introducción.....	55
4.2. Técnicas de validación de requisitos.....	55
4.2.1. Técnica de Prototipado.....	55
4.2.2. Técnica de Diseño de Casos de Prueba.....	55
4.3. Criterios para la verificación de los requisitos.....	56
4.4. Métricas orientadas a clases para evaluar el diseño.....	58
4.4.1. Tamaño operacional de clase (<i>TOC</i>).....	58
4.4.2. Relaciones entre clases (<i>RC</i>).....	61
4.5. Conclusiones del capítulo.....	65
CONCLUSIONES.....	66
RECOMENDACIONES.....	67

REFERENCIAS BIBLIOGRÁFICAS	68
BIBLIOGRAFÍA.....	70
GLOSARIO.....	72

Índice de tablas

Tabla 1: Actividades del modelo de desarrollo orientado a componentes 19

Tabla 2: Resumen de la especificación del requisito Adicionar competencia a la persona..... 33

Tabla 3: Resumen de la especificación del requisito Modificar competencia de la persona 34

Tabla 4: Resumen de la especificación del requisito Eliminar competencia de la persona..... 34

Tabla 5: Resumen de la especificación del requisito Listar competencias de la persona 35

Tabla 6: Resumen de la especificación del requisito Imprimir competencias de la persona 35

Tabla 7: Descripción de la clase GestionarcompetenciasController 40

Tabla 8: Descripción de la clase NomCompetenciaModel..... 41

Tabla 9: Descripción de la clase NomCompetencia 41

Tabla 10: Descripción de la clase GestionarclasificacioncompetenciasController 44

Tabla 11: Descripción de la clase NomClasificacioncompetenciasModel 44

Tabla 12: Descripción de la clase NomClasificacioncompetencias 45

Tabla 13: Descripción de la clase ExpedientecompetenciaspersonaController 46

Tabla 14: Descripción de la clase DatExpedientecompetenciaspersonaModel 47

Tabla 15: Descripción de la clase DatExpedientecompetenciaspersona 48

Tabla 16: Descripción de la clase ProfesiogramaController 49

Tabla 17: Descripción de la clase DatProfesiogramaModel 50

Tabla 18: Descripción de la clase DatProfesiograma 51

Tabla 19: Métrica Tamaño Operacional de Clase (TOC)..... 58

Tabla 20: Rango de valores para la evaluación de la métrica TOC 59

Tabla 21: Resultado de aplicar la métrica TOC 60

Tabla 22: Métrica Relación entre Clases (RC) 61

Tabla 23: Rango para medir el acoplamiento en la técnica RC 62

Tabla 24: Rango de valores para la evaluación de la métrica RC 62

Tabla 25: Resultado de aplicar la métrica RC 63

Tabla de ilustraciones

Ilustración 1: Descripción del ciclo de vida del proyecto ERP	18
Ilustración 2: Mapa de procesos	27
Ilustración 3: Modelo Conceptual	29
Ilustración 4: Diagrama de clases Gestionar competencias	40
Ilustración 5: Diagrama de clases Gestionar clasificación de competencias	43
Ilustración 6: Diagrama de clases Expediente de competencias de la persona.....	46
Ilustración 7: Diagrama de clases Profesiograma	49
Ilustración 8: Diagrama entidad relación	52
Ilustración 9: Diagrama de componentes	53
Ilustración 10: Gráfico de los resultados generales de acuerdo a la cantidad de procedimientos.....	59
Ilustración 11: Gráfico de la responsabilidad por clases.....	60
Ilustración 12: Gráfico de la complejidad de implementación por clases	60
Ilustración 13: Gráfico del nivel de reutilización de las clases	61
Ilustración 14: Gráfico de los resultados generales de acuerdo a las dependencias.	63
Ilustración 15: Gráfico del acoplamiento	63
Ilustración 16: Gráfico de la complejidad de mantenimiento de las clases	64
Ilustración 17: Gráfico de la cantidad de pruebas a realizar	64
Ilustración 18: Gráfico del nivel de reutilización de las clases	64

INTRODUCCIÓN

Los sistemas para la Planificación de los Recursos Empresariales, conocidos internacionalmente como ERP por sus siglas en inglés (Enterprise Resource Planning), se han convertido en poderosas herramientas de importancia trascendental para organizaciones que aspiran a una gestión más eficiente de sus recursos en medio de un escenario económico internacional cada vez más complejo y competitivo.

La economía cubana urgida también de transformaciones inmediatas que permitan fortalecer la gestión de las entidades nacionales e incrementar de forma sostenida su productividad y eficiencia, alcanzar niveles de competitividad internacional y crear mayores oportunidades para insertarse en el mercado mundial, hace que estos sistemas sean vistos también como alternativas viables en nuestro contexto económico y social. Sin embargo, las limitaciones técnico-económicas para adaptar sistemas de este tipo internacionalmente conocidos a las condiciones cubanas, impulsaron la idea de desarrollar un ERP propio, capaz de adaptarse a las particularidades de la economía nacional y que a su vez propiciase la necesaria independencia tecnológica para su desarrollo y mantenimiento. Surge así el programa ERP-Cuba con una expresión comercial ya conocida: Cedrux.

Cedrux es un paquete de soluciones integrales de gestión para las entidades cubanas que cuenta con la atención directa de varios Organismos de Administración Central del Estado (OACE). Se desarrolla por un equipo multidisciplinario compuesto por especialistas de diferentes entidades del país y de la Universidad de las Ciencias Informáticas. En su concepción se han tenido en cuenta las necesidades básicas de las empresas y entidades económicas nacionales, lo establecido en la legislación vigente y las funcionalidades presentes en sistemas similares existentes en el mundo, así como de otros sistemas certificados en Cuba. Actualmente Cedrux se encuentra en fase de prueba en varias instituciones del país para su posterior despliegue y está llamado a convertirse en una plataforma fundamental para el control y gestión de recursos financieros, materiales y de capital humano; un sistema que proporcionará integración de la información y modernización de los procesos de negocio que le permitirá a las entidades mayor eficiencia, productividad y reducción de los costos.

Uno de los subsistemas que forma parte del sistema Cedrux es el de Capital Humano (CH), el cual tiene como objetivo proporcionar las herramientas necesarias para garantizar su adecuada gestión en las organizaciones, encontrándose aún en fase de desarrollo.

El CH es un eslabón fundamental en esta gran cadena de gestión de la entidad, y la forma en que este se gestione podrá influir de manera decisiva en las posibilidades de alcanzar o no las metas y objetivos estratégicos propuestos.

Las tendencias actuales de gestión del CH se enfocan hacia la implementación de un sistema integral basado en competencias, aplicado a procesos claves de la Gestión del Capital Humano (GCH) como son: la organización del trabajo, la integración y selección, la evaluación del desempeño y la capacitación y desarrollo.

La competencia laboral se ha convertido en los últimos años en un marco de referencia para la organización y el desarrollo del personal en las organizaciones actuales, en las que el proceso de adaptarse y adelantarse a los cambios del entorno se presenta como un factor indispensable para alcanzar resultados positivos en la vida laboral.

Así lo evidencian diversas empresas no sólo del mundo desarrollado en nuestro continente y Europa, sino también empresas de varios países latinoamericanos que han incorporado el concepto de Competencias laborales en sus modelos de GCH.

En el caso de Cuba, a pesar de contarse con la base jurídica necesaria para la implementación de la gestión por competencias, la gran mayoría de las empresas continúan aplicando la forma tradicional de gestión, la gestión por funciones. Un breve sondeo efectuado en algunas empresas importantes de la provincia Cienfuegos (ETECSA, Refinería de Petróleo, Polo Petroquímico, Plastimec) evidencia que salvo raras excepciones, las propias áreas de Capital Humano no poseen claridad sobre este novedoso proceso de gestión. En el entorno económico actual constituye un imperativo para las organizaciones cubanas transitar con urgencia hacia la gestión por competencias. En este sentido resulta importante no sólo incorporar en la filosofía actual de gestión los elementos que sustentan la gestión por competencias, sino también poder contar con herramientas informáticas que faciliten su implementación de forma fiable y eficiente.

Por tanto, queda definido como **problema a resolver** que los procesos soportados por el subsistema de Capital Humano de Cedrux no garantizan el trabajo por competencias laborales según establece el modelo cubano para la Gestión Integral de Capital Humano.

Objeto de estudio

Los procesos de Capital Humano.

Campo de acción

El proceso competencias laborales del subsistema Capital Humano del Sistema Integral de Gestión Cedrux.

Para darle solución al problema planteado, se tiene como **objetivo general** realizar el análisis y el diseño del proceso Competencias laborales del subsistema Capital Humano del Sistema Integral de Gestión Cedrux.

Los **objetivos específicos** son:

- Realizar el Diseño teórico y metodológico de la investigación que permita conocer los elementos que la componen y las características referentes al proceso Competencias laborales y su relación con el subsistema Capital Humano del Sistema Integral de Gestión Cedrux.
- Capturar y describir los requerimientos para el proceso Competencias laborales.
- Realizar el diseño del sistema según los requisitos para el proceso.
- Validar el resultado obtenido.

Idea a defender

Si se realiza correctamente el análisis y diseño del proceso competencias laborales, los procesos soportados por el subsistema Capital Humano de Cedrux garantizarán el trabajo según establece el modelo cubano para la Gestión Integral de Capital Humano.

Para el desarrollo del presente trabajo se hace uso de los siguientes **métodos científicos** de investigación:

Métodos teóricos:

Histórico-lógico: Mediante el uso de este método se podrá conocer el estado del arte del proceso competencias laborales, constatándose además la evolución y desarrollo alcanzado por este hasta la actualidad. (Hernández León y otros, 2002)

Análisis-síntesis: Este método es usado para realizar un análisis de toda la documentación y bibliografía referente al proceso competencias laborales, extrayendo de estos los elementos más importantes. (Hernández León y otros, 2002)

Modelación: Este método se utiliza en el momento de generar los artefactos necesarios para el análisis y el diseño del proceso Competencias laborales. (Hernández León y otros, 2002)

Métodos empíricos:

Entrevista: Se realizan entrevistas a personal calificado y conocedor de la actividad de recursos humanos, ayudando a la adquisición de información y a una mejor obtención de captura de requisitos. (Hernández León y otros, 2002)

Aportes esperados:

Análisis y diseño del proceso competencias laborales, posibilitando que los procesos soportados por el subsistema de Capital Humano de Cedrux garanticen el trabajo según se establece en el modelo cubano para la Gestión Integral de Capital Humano.

El presente trabajo se estructura de la siguiente forma:

Capítulo I: Fundamentación teórica.

En este capítulo se realiza el estudio del estado del arte, donde se hará una breve reseña de los principales conceptos asociados a la GCH basada en las competencias laborales, también se dará a conocer el modelo de desarrollo, lenguajes y herramientas que fueron utilizadas para el modelado del proceso competencias laborales.

Capítulo II: Características del sistema

Se realizará un estudio preliminar del sistema. Se van a identificar los conceptos del negocio y se va a realizar también el modelo conceptual. Quedarán especificadas además las técnicas empleadas para la captura de requisitos. Por último se identificarán y describirán los requisitos funcionales que va a presentar la solución a construir.

Capítulo III: Diseño del sistema

En este capítulo se verán los detalles relacionados con el diseño del sistema propuesto, utilizando para su modelado los diagramas de clase, el diagrama de componentes y el diagrama entidad relación, con la aplicación de los patrones correspondientes; realizando posteriormente una descripción de los mismos.

Capítulo IV: Validación de los Resultados

Se realizan las validaciones del proceso aplicando diferentes técnicas y métricas, y se muestran los resultados obtenidos.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

1.1. Introducción

En el presente capítulo se ofrece una visión general sobre la GCH basada en competencias, explicándose sus características y ventajas. Se muestran algunas definiciones expuestas por diferentes autores sobre el concepto de competencia laboral, así como su clasificación y evaluación, y se hará mención de algunas tendencias actuales, tanto internacionales como nacionales, enfocadas a esta moderna forma de GCH (gestión por competencias). Se explicará además el modelo de desarrollo realizado por la dirección del proyecto, el cual será utilizado en el presente trabajo para llevar a cabo su solución y por último se analizará la arquitectura definida por el sistema Cedrux.

1.2. Gestión por competencias

La GCH basada en competencias se reconoce en la actualidad como el camino más eficiente a seguir para lograr una mayor contribución, por parte de los trabajadores, a la competitividad empresarial. Cuanto más integrado sea el equipo de trabajo, y más se aprovechen las cualidades de cada uno de sus integrantes, más fuerte será la empresa.

Aquellas empresas u organizaciones que mejor sepan gestionar su CH, teniendo en cuenta sus cualidades, estarán favorecidas de una gran ventaja competitiva, marcando así la diferencia con el resto.

La gestión por competencias permite evaluar las competencias específicas para cada puesto y favorecer el desarrollo personal de los empleados. (Venegas, 2004)

Según K. Cruz y G. Vega se entiende por Gestión por Competencias “el gerenciamiento que detectará las competencias que requiere un puesto de trabajo para que quien lo desarrolle mantenga un rendimiento elevado o superior a la media; determinará a la persona que cumpla con estas competencias; favorecerá además el desarrollo de competencias tendientes a mejorar aún más el desempeño superior (sobre la media) en el puesto de trabajo y permitirá que el recurso humano de la organización se transforme en una aptitud central y de cuyo desarrollo se obtendrá una ventaja competitiva para la empresa”. (K. Cruz Muñoz, 2001)

La gestión por competencias consiste además en mejorar el rendimiento, el desempeño y la productividad de la Organización; identificar los conocimientos, las capacidades y las actitudes del equipo de trabajo;

mejorar la capacidad de gestión y solución de problemas; definir los cargos exitosos y las rutas profesionales; evaluar la capacidad de las personas y definir los planes de desarrollo individual, y ampliar el alcance a la selección, la compensación, la promoción y la formación. (K. Cruz Muñoz, 2001)

Según la opinión del Dr. Armando Cuesta Santos, “el futuro de la gestión de competencias es sin dudas prometedor. Su complejidad exige rigor científico técnico para hacerla viable de modo sustentable cada vez con mayor eficacia y eficiencia. En correspondencia demanda profesionales bien preparados desde una óptica polivalente o multidisciplinaria. Ya no es novedad destacar que el recurso humano de las organizaciones determina la ventaja competitiva básica”. (Cuesta, 2005)

Realizándose en las empresas una eficiente GCH basándose en las competencias, los conocimientos y las habilidades de los trabajadores aumentarían en gran medida, creando así el desarrollo individual y organizacional, enfocándose en las necesidades de capacitación y apoyando a los sistemas de desarrollo del personal.

Muchas son las ventajas que trae consigo la GCH basada en competencia, Venegas refiere como algunas de ellas: (Venegas, 2004)

- La posibilidad de definir perfiles profesionales que favorecerán a la productividad.
- El desarrollo de equipos que posean las competencias necesarias para su área específica de trabajo.
- La identificación de los puntos débiles, permitiendo intervenciones de mejora que garantizan los resultados.
- La gestión por competencias se basa en objetivos medibles cuantificables y con posibilidad de observación directa.
- El aumento de la productividad y la optimización de los resultados.
- La concientización de los equipos para que asuman la co-responsabilidad de su autodesarrollo. Tornándose un proceso de ganar-ganar, desde el momento en que las expectativas de todos estén atendidas.

Según M. A. Cabezas éstas son otras de las ventajas que brinda la gestión por competencias: (Cabezas, 2001)

- La evaluación del desempeño.

- La eliminación de la costosa e improductiva práctica del adiestramiento tradicional. Cuando se sitúa la gestión por competencias, se evita que los directivos y sus colaboradores pierdan el tiempo en programas de entrenamiento y desarrollo que no tienen que ver con las necesidades de la empresa o las necesidades de los puestos de trabajo.
- La sustitución urgente de las descripciones del cargo como eje de la gestión de los recursos humanos.
- La compensación justa con base en el aporte al valor agregado.

Sin dudas las ventajas que proporciona el modelo de GCH basado en competencias han incidido favorablemente en que muchas empresas en el mundo lo hayan incorporado. La justificación de estos esfuerzos se encuentra en el intento de mejorar los niveles de productividad y competitividad mediante la movilización del conocimiento y de la capacidad de aprender, de la organización. Se hace evidente así, la tendencia de revalorización del aporte humano a la competitividad organizacional. (CINTERFOR/OIT)

En Bolivia, la Institución Nacional de Formación Profesional, el **Instituto Nacional de Formación y Capacitación Laboral (INFOCAL)**, realizó la actualización del Clasificador Nacional de Ocupaciones (CNO) teniendo en cuenta la necesidad de contar con un mapa orientador sobre las distintas ocupaciones existentes en el país, sus diferentes niveles de competencia y funciones. Con la utilización del CNO se logrará un instrumento orientador en la actividad formadora, que servirá para lo referente a la ampliación de instrumentos de identificación de competencias, al insumo para la elaboración o la modificación de los currículos instruccionales de la institución y finalmente contribuir con ello a la modernización del país mediante la formación de recursos humanos altamente competitivos. (Vargas, 2004)

El **Servicio Nacional de Aprendizaje Industrial (SENAI) de Brasil** está desarrollando su modelo de formación basada en competencia laboral. Al efecto ha definido un proyecto estratégico nacional encaminado a dos grandes resultados (Vargas, 2004):

- La elaboración de perfiles y programas de formación por competencias.
- La puesta en marcha de un proceso de reconocimiento de las competencias adquiridas por la experiencia laboral.

En Colombia está presente la formación por competencias en el **Servicio Nacional de Aprendizaje (SENA)** y la Red Nacional de Instituciones de Formación para el Trabajo. El SENA intenta crear un

lenguaje común para el manejo de diferentes temas entre los cuales está el diseño de las normas de competencia laboral, los programas de formación y el reconocimiento de competencias. (Vargas, 2004)

El **Instituto Técnico de Capacitación y Productividad (INTECAP) de Guatemala** ha incorporado el enfoque de competencia laboral en su gestión institucional. Es así como ha conformado el modelo "Norte" por: Normalización Técnica de Competencias. El modelo está siendo aplicado tanto en el diseño de planes y material didáctico, evaluación y certificación de las competencias, como en los procesos de asesoría en la GCH. (Vargas, 2004)

El **Instituto Salvadoreño de Formación Profesional (INSAFORP)**, ha organizado un proceso de desarrollo organizacional que concentra sus esfuerzos en la Formación Basada en Competencia Laboral y en el diseño y promoción de un Sistema Nacional de Formación y Certificación. La formación basada en competencias en el Instituto Nacional de Formación Técnico Profesional (**INFOTEP**) es un camino de avance hacia un Sistema Nacional de Formación Profesional para el Trabajo Productivo en la República Dominicana. El INFOTEP ha desarrollado exitosamente la utilización de la metodología de medición y mejoramiento de la productividad, la cual utiliza el diagnóstico de las competencias que deben ser desarrolladas por los recursos humanos de la empresa, para lograr una plena aplicación y desarrollo del talento humano, dar sentido y pertinencia a las acciones de capacitación y aplicar al desarrollo de los indicadores de productividad de las empresas beneficiarias. La idea central gira en torno al papel de la capacitación para desarrollar las competencias que contribuyen a mejorar la competitividad de la empresa. El Instituto ha documentado varias exitosas aplicaciones de esta metodología que combina formación y gestión de la competitividad. (Vargas, 2004)

Se conocen además experiencias sobre aplicaciones de sistemas normalizados de competencia, bastante difundidos en Inglaterra, Irlanda, Escocia, Australia, enmarcadas dentro de un sistema nacional de formación y certificación. En estos casos, la característica principal de su proyección nacional y la articulación de las instituciones de formación son las necesidades de las instituciones a través de la formación basada en normas de competencias. (Zúñiga, 2004)

Cuba, con la aprobación en 2007 de la NC 3000 (3000, 3001 y 3002) “**Sistema de Gestión Integrada de Capital Humano**”, sentó las bases y proporcionó los elementos y regulaciones necesarias para

implementar en las organizaciones del país la Gestión por Competencias, como estadio superior a la Gestión por Funciones que desarrollaban (y aún desarrollan) la mayoría de las empresas cubanas.

Esta Norma define la gestión por competencias como las actividades coordinadas para dirigir y controlar una organización con un enfoque basado en las competencias laborales y la capacidad de aprendizaje de los colaboradores. Su objetivo es una organización de calidad y la disposición del colectivo integrado para el logro de los objetivos de la organización.

La Norma establece que para llevar a cabo el complejo proceso de la GCH por competencias se debe formar uno o varios Comités de Competencias integrados por expertos, los cuales realizan la identificación de las competencias de los diferentes niveles y las proponen para su aprobación por la alta dirección. Con la aprobación de las propuestas de los Comités de Competencias quedan oficialmente establecidas las competencias laborales y el profesiograma de cada puesto de trabajo de la organización, las cuales deben ser certificadas periódicamente, constituyendo un reconocimiento formal dentro de la organización. La certificación de una competencia demostrada para un trabajo determinado garantiza la calidad de lo que el trabajador es capaz de hacer, de ahí la importancia de que este proceso sea ágil y a la vez abarcador en toda el área de la GCH.

Las competencias son la base del sistema de GCH e interactúan en procesos vitales como son la Selección del personal, la Capacitación y Desarrollo y la Evaluación del desempeño.

A partir de estos preceptos diversas empresas, sobre todo las que se encontraban desarrollando el proceso de Perfeccionamiento Empresarial, desde finales de la primera década del presente siglo iniciaron un proceso de adaptación de los métodos tradicionales de GCH a esta novedosa forma de gestión y algunas de ellas lograron certificarse, entre las que se encuentran la empresa TRAYCO (Transporte de la Construcción) de Pinar del Río, la Empresa de Ingeniería y Proyectos (EIPP) de Ciudad de la Habana, la Empresa Mecánica de Níquel de Holguín, la Empresa Comercializadora de Camagüey, entre otras.

A la par de los cambios que se han venido experimentando en la concepción del proceso de GCH, se ha concedido especial relevancia a la informatización del mismo. Desde hace años se vienen desarrollando y perfeccionando sistemas de gestión integrales en diversas partes del mundo, algunos de los cuales se han convertido en software de referencia mundial. Tales son los casos de SAP, ORACLE, Open Bravo, SAGE, entre otros. Estos Software son productos modulares que pueden ser adquiridos íntegramente o

de forma parcial (módulos específicos) a partir de las necesidades y posibilidades económicas de los países. Son productos que implican altísimos costos de adquisición, adaptación y mantenimiento, generando, además de los gastos, una permanente dependencia tecnológica.

En Cuba, también desde mediados de la primera década de este siglo se han venido realizando intentos por algunas empresas y organismos del país para automatizar actividades de la GCH con el enfoque de la Gestión por competencias, sin embargo sólo uno ha logrado establecerse oficialmente al nivel de implementación y comercialización, el sistema **RH-Expert** desarrollado por la empresa Datazucar del Ministerio del Azúcar (MINAZ). (Nicola, 2011)

En el 2005 el Centro de Información y Gestión Tecnológica (CIGET) de la Delegación Territorial del CITMA en Cienfuegos desarrolló el sistema **RH-CITMA**, con similares propósitos, pero que nunca llegó a utilizarse.

RH-Expert es hoy el único producto oficial del perfil de la gestión del capital humano en Cuba ideado sobre normas específicas relativas a las competencias. Es un software que implementa criterios de la Gestión por Competencias, pero que no abarca todos los procesos que esta incluye. Ofrece soluciones para gestionar lo concerniente a los expedientes laborales, la evaluación del desempeño, seguridad y salud, y capacitación. Hasta la fecha es un producto único con estas características y enfoque en el mercado nacional. Ha sido el principal salto de avance en este sentido toda vez que nació en momentos en que el país, especialmente desde 2006, ha hecho hincapié en la GCH como concepto más abarcador: la persona con su conocimiento, sus habilidades, compromiso con la organización, con sus deseos de hacer las cosas (Nicola, 2011). Como puede apreciarse, a diferencia de Cedrux, RH-Expert es una aplicación ideada para propósitos concretos de la gestión del capital humano.

Según todo lo expuesto anteriormente se puede observar que al integrar el enfoque de competencias a la GCH se obtienen solamente beneficios que ayudarán a desarrollar a las personas en un marco productivo más eficiente y permitirá a las empresas fortalecer sus resultados. Para conocer un poco más acerca del tema, a continuación se mostrarán algunas definiciones del concepto competencias expuestas por diferentes autores.

1.2.1. Concepto de competencias

Sobre competencia se puede contar con tantas definiciones como expertos en la materia, sin embargo se han seleccionado algunas de ellas, intentando construir una gama lo más completa posible. A continuación se mostrarán algunas definiciones expuestas por varios autores:

Según el Dr. Cuesta, las competencias “son habilidades, destrezas, aptitudes, actitudes, rasgos de personalidad, son estilos de dirección, formas de relaciones interpersonales, de experiencias que se adquieren durante toda la vida, tanto personal como laboral”. (Cuesta, 1999)

Cubeiro plantea que son “el conjunto de destrezas, habilidades, conocimientos y características conductuales que, correctamente combinados frente a una situación de trabajo, predicen un desempeño superior, es aquello que distingue el rendimiento excepcional de lo normal y que se observa directamente a través de las conductas de cada empleado, en la ejecución diaria de su cargo”. (Cubeiro, 1998)

Anne Marelli define competencia laboral como “una capacidad laboral, medible, necesaria para realizar un trabajo eficazmente, es decir, para producir los resultados deseados por la organización. Está conformada por conocimientos, habilidades, destrezas y comportamientos que los colaboradores deben demostrar para que la organización alcance sus metas y objetivos”. Y añade además que “son capacidades humanas, susceptibles de ser medidas, que se necesitan para satisfacer con eficacia los niveles de rendimiento exigidos en el trabajo”. (Marelli, 1999)

Agustín Ibarra la define como “la capacidad productiva de un individuo que se define y mide en términos de desempeño en un determinado contexto laboral, y no solamente de conocimientos, habilidades o destrezas en abstracto; es decir, la competencia es la integración entre el saber, el saber hacer y el saber ser”. (Ibarra, 2000)

Bunk puntualiza que “posee competencia profesional quien dispone de los conocimientos, destrezas y aptitudes necesarios para ejercer una profesión, puede resolver los problemas profesionales de forma autónoma y flexible, está capacitado para colaborar en su entorno profesional y en la organización del trabajo.” (Bunk, 1994)

Como se puede observar existe una gran variedad de definiciones sobre este tema, pero para el presente trabajo se tendrá en cuenta, por considerarse la más completa, la aprobada por la NC 3000: 2007, que

plantea que: Las competencias laborales son un “conjunto sinérgico de conocimientos, habilidades, experiencias, sentimientos, actitudes, motivaciones, características personales y valores, basado en la idoneidad demostrada, asociado a un desempeño superior del trabajador y de la organización, en correspondencias con las exigencias técnicas, productivas y de servicios.” Puntualizando además que: “es requerimiento esencial que esas competencias sean observables, medibles y que contribuyan al logro de los objetivos de la organización.” (NC 3000, 2007)

De manera general se puede decir que las competencias laborales son los conocimientos, características y habilidades que posee una persona para ejercer determinada ocupación, que deben ser medibles y que de estas depende el logro que pueda llegar a alcanzar una organización.

1.2.2. Clasificación de las competencias

Muchas son las bibliografías que abordan sobre el tema de competencias laborales, y dentro de éstas sus clasificaciones. Disímiles autores proponen su manera de clasificarlas, y todos de diferentes formas. Las siguientes definiciones fueron las encontradas en las bibliografías estudiadas:

Spencer & Spencer definen cinco principales tipos de competencias:

- ✓ **Motivación:** Los intereses que una persona considera o desea consistentemente. Las motivaciones dirigen y seleccionan el comportamiento hacia ciertas acciones u objetivos y lo alejan de otros.
- ✓ **Rasgos de carácter:** Una predisposición general a conducirse o reaccionar de modo determinado.
- ✓ **Concepto de uno mismo:** Las actitudes, valores o imagen propia de una persona.
- ✓ **Conocimiento:** La información que una persona posee sobre áreas específicas, de hechos o procedimientos, tanto técnicos o interpersonales.
- ✓ **Habilidad:** La capacidad de desempeñar cierta tarea física o mental. Las competencias mentales o cognitivas incluyen pensamiento analítico (procesamiento de información y datos, determinación de causa y efecto, organización de datos y planos) y pensamiento conceptual (reconocimiento de características en datos complejos). (Spencer & Spencer, 1993)

El Instituto Técnico de Capacitación y Productividad (INTECAP) reconoce tres tipos de competencias:

- ✓ **Básicas:** Aquellas de índole formativo que requiere la persona para desempeñarse en cualquier actividad productiva, tales como la capacidad de leer, interpretar textos, aplicar sistemas numéricos, saber expresarse y saber escuchar.
- ✓ **Genéricas:** Son aquellos conocimientos y habilidades que están asociados al desarrollo de diversas áreas ocupacionales; por ejemplo, analizar y evaluar información, trabajar en equipo, contribuir al mantenimiento de la seguridad y higiene en el área de trabajo, planear acciones, entre otras.
- ✓ **Técnicas:** Son aquellas competencias asociadas a conocimientos y habilidades de índole técnicos y que son necesarias para la ejecución de una función productiva. Generalmente se refieren a un lenguaje específico y al uso de instrumentos y herramientas determinadas, por ejemplo, soldar con equipo de oxi-acetileno, preparar el molino para laminado en caliente o evaluar el desempeño del candidato. Se adquieren y desarrollan a través del proceso de capacitación, en el centro de trabajo o en forma autodidacta. (Barrios, 2004)

G. Bunk clasificó las competencias en:

- ✓ **Competencia técnica:** Es el dominio experto de las tareas y contenidos del ámbito de trabajo, así como los conocimientos y destrezas necesarios para ello.
- ✓ **Competencia metodológica:** Implica reaccionar aplicando el procedimiento adecuado a las tareas encomendadas y a las irregularidades que se presenten, encontrar soluciones y transferir experiencias a las nuevas situaciones de trabajo.
- ✓ **Competencia social:** Colaborar con otras personas en forma comunicativa y constructiva, mostrar un comportamiento orientado al grupo y un entendimiento interpersonal.
- ✓ **Competencia participativa:** Participar en la organización de ambiente de trabajo, tanto el inmediato como el del entorno capacidad de organizar y decidir, así como de aceptar responsabilidades.
- ✓ **Competencias técnicas:** Este tipo de competencias incluye las habilidades, técnicas y conocimientos que debe tener la persona para ocupar un puesto determinado y que su desempeño sea eficiente.

- ✓ **Competencias conductuales:** Son las competencias que tienen que ver con las características individuales de cada persona y la motivación que pueda tener o no por el puesto que ocupa. Este tipo de competencias no suelen ser identificadas ni desarrolladas con facilidad. (Bunk, 1994)

Como se puede observar, en las bibliografías consultadas las competencias laborales se clasifican de manera completamente diferente, pero para el presente trabajo no se va a utilizar ninguna de las clasificaciones previamente mencionadas, sino las definidas por el Dr. Armando Cuesta, el cual las clasifica en: (Cuesta, 1999)

- ✓ **Genéricas:** Estas pueden responder a cualquier puesto de trabajo o a la mayoría según su misión, objetivos y metas.
- ✓ **Específicas:** Estas se adecuan a puestos de trabajo, con objetivos de trabajo específicos y pueden cambiar de acuerdo a los escenarios y las estrategias que se traza la organización.
- ✓ **Compuestas:** Son aquellas en la que intervienen varias competencias, son más difíciles de dimensionar y dificultan el análisis de desempeño.
- ✓ **Simple:** Son las que se refieren a una sola competencia para evaluar, siendo fáciles de dimensionar y por lo tanto facilitan su evaluación.
- ✓ **Importantes:** Todas no tienen la misma importancia porque depende de las características del puesto de trabajo y de la misión, objetivos y estrategias del puesto y la organización.
- ✓ **Esenciales:** Dentro de las anteriores hay otras que le llaman esenciales, que debe poseer una persona independientemente del puesto que desempeña.

1.2.3. Evaluación por competencias

Se puede definir el sistema de evaluación como el nexo de unión de los sistemas de gestión de capital humano basados en competencias. El sistema de evaluación vincula la estructura profesional (profesiones y competencias) con los sistemas de desarrollo profesional y retributivo, a través del contraste del perfil ideal definido para cada profesión con la situación profesional de cada persona. (Grande)

El SENAI define la evaluación por competencias como un proceso de recolección de evidencias sobre el desempeño profesional de una persona con el propósito de formarse un juicio sobre su competencia en relación con un perfil profesional e identificar aquellas áreas de desempeño que deban ser fortalecidas, utilizando la formación u otros medios, para llegar al nivel de competencia requerido. (SENAI, 2002)

CONOCER de México ha determinado la evaluación de competencias como el proceso mediante el cual se recogen evidencias sobre el desempeño laboral de un individuo, con el fin de determinar si es competente o aún no, para realizar una función laboral determinada. (Fletcher, 1997)

La evaluación de competencias laborales es definida como un proceso con varios grandes pasos, entre los que se encuentran: (CINTERFOR/OIT)

- ✓ Definición de los objetivos.
- ✓ Recolección de evidencias.
- ✓ Comparación de evidencias con los objetivos.
- ✓ Formación de un juicio (competente o aún no competente).

Algunas de las características principales de la evaluación por competencias son:

- ✓ Está fundamentada en estándares que describen el nivel esperado de competencia laboral.
- ✓ Los estándares incluyen criterios que detallan lo que se considera un trabajo bien hecho.
- ✓ La evaluación es individual, no compara trabajadores entre sí.
- ✓ Configura un juicio para el trabajador evaluado: competente o aún no competente.
- ✓ Se realiza preferentemente, en situaciones reales de trabajo.
- ✓ No se ciñe a un tiempo predeterminado para su realización; es más bien un proceso que un momento.
- ✓ No está sujeta a la terminación de una acción específica de capacitación.
- ✓ Incluye el reconocimiento de competencias adquiridas como resultado de la experiencia laboral.
- ✓ Es una herramienta para la orientación del aprendizaje posterior del trabajador; como tal tiene un importante rol en el desarrollo de las habilidades y capacidades de los evaluados.
- ✓ Es la base para la certificación de la competencia laboral del trabajador.

La evaluación del desempeño posee además disímiles ventajas, entre ellas cabe mencionar:

- ✓ Detectar necesidades de capacitación
- ✓ Descubrir personas claves
- ✓ Descubrir inquietudes de los evaluados
- ✓ Encontrar una persona para un puesto
- ✓ Tomar decisiones sobre salarios y promociones

El proceso de evaluación del desempeño es un complemento importante dentro la GCH y constituye una vía importante para lograr elevados resultados en las empresas y mejores desempeños de los colaboradores.

La evaluación del desempeño se ubica entre las áreas de más impacto para la gestión integral del CH. Para lograr el mejor aporte de esta área al proceso integral, se necesita contar con información histórica del desempeño de los colaboradores. Ya que con la correcta utilización de esta información los directivos logran obtener indicadores de evaluación cada vez más reales para cada cargo, siguiendo los patrones definidos por la experiencia de los mejores trabajadores por puesto de trabajo a lo largo de la historia de la institución. (Martínez, 2009)

1.3. Modelo de desarrollo orientado a componentes

La propuesta del modelo de desarrollo orientado a componentes fue elaborada por el equipo de producción en colaboración con las Líneas de desarrollo del proyecto ERP-Cuba de acuerdo con las necesidades presentadas por cada una de ellas y donde se tuvieron en cuenta los principales riesgos con los que se cuentan en el proyecto. (Equipo de producción, 2007)

1.3.1. Características

El modelo de desarrollo de software propuesto describe la secuencia de actividades de alto nivel para la construcción y desarrollo de soluciones. Se logra con la combinación entre los modelos Orientado a componentes y el Iterativo e incremental. (Miniet y otros, 2009)

✓ Centrado en la arquitectura

La arquitectura determina la línea base, los elementos de software estructurales a partir de los elementos de la arquitectura de negocio. Interviene en la gestión de cambios y diseña la evolución e integración del producto. La arquitectura orienta las prioridades del desarrollo y resuelve las necesidades tecnológicas y de soporte para el desarrollo.

✓ Orientado a componentes

Las iteraciones son orientadas por el nivel de significancia arquitectónicas de los componentes, los mismos son abstracciones arquitectónicas de los procesos de negocio y requisitos asociados que modelan, el componente es la unidad de medición y ordenamiento de las iteraciones.

✓ Iterativo e incremental

Las iteraciones son planificadas y coordinadas con el equipo de arquitectura, los clientes y la alta gerencia. Cada iteración constituye el desarrollo de componentes, los cuales son integrados al término de la integración, permitiendo de esta manera la evolución incremental del producto.

✓ Ágil y adaptable al cambio

El desarrollo de las partes formaliza solamente las características principales de la solución, priorizando los talleres y las comunicaciones entre las personas. Los clientes y funcionales están involucrados en el proyecto y poseen parte de las responsabilidades del éxito del mismo. Los cambios son conciliados semanalmente, discutidos y aprobados.

1.3.2. Fases del ciclo de vida del proyecto

Descripción del ciclo de vida:

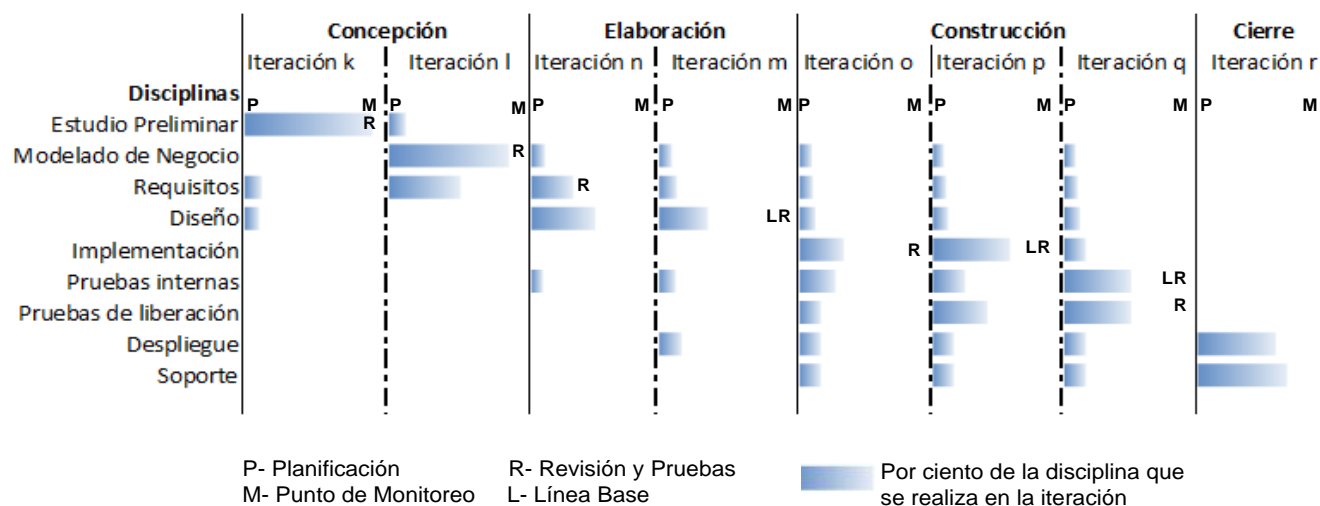


Ilustración 1: Descripción del ciclo de vida del proyecto ERP

Las fases del ciclo de vida del proyecto ERP son: (Equipo de producción, 2007)

- **Concepción**

- Objetivos: Realizar la evaluación de factibilidad del proyecto, de su elaboración y su aprobación definiendo los macro requisitos y macro componentes, definir los procesos de negocios, identificar los requisitos y definir la línea base de los requisitos del proyecto.
- Hito: Concebir el alcance del proyecto.

- **Elaboración**

- Objetivos: Firmar el Acuerdo de Colaboración o Contrato, realizar las descripciones de los requisitos y el diseño arquitectónico de la solución.
- Hito: Establecer la línea base de la arquitectura.

- **Construcción**

- Objetivo: Implementar las funcionalidades del sistema informático.
- Hito: La realización de las pruebas internas y de liberación por parte del Laboratorio Industrial de Pruebas de Software del sistema informático.

- **Cierre**

- Objetivo: Realizar las pruebas piloto y de regresión al software, garantizar la transferencia del sistema al cliente y brindarle mantenimiento al software.
- Hito: Realizar la transferencia de una versión estable junto a su documentación al cliente del sistema informático.

1.3.3. Actividades

El modelo de desarrollo orientado a componentes propone como actividades para el rol de analista las siguientes: (Equipo de producción, 2007)

Tabla 1: Actividades del modelo de desarrollo orientado a componentes

Actividades del Desarrollo	Descripción	Participan
Identificación de Procesos	Se debe identificar, analizar y describir los procesos que se	• Analistas

	llevan a cabo en el negocio que se desea automatizar, con el objetivo de organizar y documentar todas las acciones a tener en cuenta en el análisis para el desarrollo del Software	<ul style="list-style-type: none"> • Especialistas Funcionales • J´de Línea de Desarrollo
Validación de Procesos	Se aprueban que la identificación de los procesos se hizo correctamente y que el equipo tiene plena claridad del negocio a automatizar.	<ul style="list-style-type: none"> • Analistas • Especialistas Funcionales • J´de Línea de Desarrollo
Identificación de Requerimientos	A partir de los procesos identificados se realiza para cada uno de ellos la identificación de los requisitos o funcionalidades que debe cumplir, para que pueda ser realizado dicho proceso.	<ul style="list-style-type: none"> • Analistas • Especialistas Funcionales • J´de Línea de Desarrollo
Validación de Requerimientos	Se validan todos y cada uno de los requisitos identificados para cada uno de los procesos que intervienen en la automatización del negocio deseado.	<ul style="list-style-type: none"> • Analistas • Especialistas Funcionales • J´de Línea de Desarrollo
Taller de Análisis	<p>Se evalúan cada uno de los requerimientos y procesos identificados y validados, a partir de los cuales se desarrollan el mapa de procesos a través del cual:</p> <ul style="list-style-type: none"> • Se agrupan los requerimientos y procesos por componentes. • Se identifican las dependencia entre los componentes, así como los contratos de los mismo en el mismo modulo. • Se establecen las prioridades de desarrollo de cada uno de los componentes, dependiendo de cuan críticos y complejos sean. 	<ul style="list-style-type: none"> • Analistas • Arquitecto de Datos • Arquitecto de Sistema • Diseñador de LN • Desarrollador de IU
Diseño del Modelo de Datos	Partiendo de un modelo lógico obtenido durante la captura de requisitos, se definen las estructuras de base de datos que darán soporte de persistencia a la solución de software orientada a los componentes identificados.	<ul style="list-style-type: none"> • Arquitecto de Datos • Analistas
Creación del Modelo de Datos	Creación de las estructuras y objetos de base de datos en el sistema de gestión seleccionado orientada a los componentes identificados..	<ul style="list-style-type: none"> • Arquitecto de Datos • Analistas
Programación del Modelo de Datos	<p>Consiste en implementar la capa de acceso a datos orientada a los componentes identificados. Tiene que estar creada la base de datos y las entidades de dominio.</p> <ul style="list-style-type: none"> • Realizar los ficheros de mapeo: Consiste en crear 	<ul style="list-style-type: none"> • Arquitecto de Datos • Analistas

	<p>los ficheros de mapeo mediante el Doctrine.</p> <ul style="list-style-type: none"> • Programación de interfaces: Consiste en crear las clases que implementan las interfaces de los DAOs. • Realizar pruebas unitarias: Realizar pruebas unitarias a las implementaciones de los DAOs. 	
Diseño de Lógica de Negocio	Se diseñan los métodos y clases para dar solución a todas las necesidades detectadas durante la identificación de componentes, ajustándose a las funcionalidades previstas.	<ul style="list-style-type: none"> • Diseñadores de LN • Analistas
Diseño IU	Se diseñan las interfaces de interacción con el usuario en dependencia de las funcionalidades y componentes detectados	<ul style="list-style-type: none"> • Diseñadores de IU • Analistas
Reunión de Implementación	Se realiza una breve descripción con los implementadores explicándole las órdenes de desarrollo y explicándole de forma operativa la Lógica de Negocio.	<ul style="list-style-type: none"> • Diseñadores de LN • Analistas • Desarrolladores
Casos de Prueba	Construcción de todos los posibles caminos de ejecución, o escenarios, de cada componente desarrollado. Se obtiene como resultado un listado final con los casos de prueba identificados a partir de los posibles escenarios, los resultados esperados para cada caso y las condiciones o valores requeridos para la ejecución de los distintos escenarios.	<ul style="list-style-type: none"> • Analistas • Especialista de Calidad

1.3.4. Artefactos que se generan

En la fase de Elaboración se realiza el análisis y diseño del sistema, donde el analista de software tiene la responsabilidad de participar en las sesiones de trabajo para identificar, describir y validar los procesos de negocio y los requisitos de software, elaborar los artefactos definidos según los estándares establecidos y participar en los Talleres de Diseño. Generándose los siguientes artefactos: (Equipo de producción, 2007)

- Mapas de procesos de negocio
- Descripción de procesos de negocio
- Modelo Conceptual
- Prototipo de Interfaz de Usuario

- Especificación de requisitos.
- Casos de Prueba

1.3.5. Tecnología

En el presente trabajo, de acuerdo a lo que sugiere el modelo a utilizar y de conjunto con la dirección de la línea de Capital humano se llegó al consenso de utilizar las siguientes notaciones, lenguaje de modelado y herramientas:

La Notación para el Modelado de Procesos de Negocio o BPMN por sus siglas en inglés; para generar el Mapa de procesos y el Modelo conceptual.

El Lenguaje Unificado de Modelado o UML por sus siglas en inglés para generar los diagramas del diseño tales como: diagrama de clases del diseño, diagrama Entidad-Relación para representar la base de datos y el diagrama de Componentes.

La herramienta Visual Paradigm 6.4 para modelar los diagramas del proceso competencias laborales y para el diseño de los prototipos de interfaz de usuario.

1.4. Arquitectura del sistema

Para el desarrollo del sistema Cedrux se decidió adoptar la propuesta de Arquitectura Base definida por la línea de Arquitectura del proyecto ERP-Cuba conformada por las diferentes vistas y estilos arquitectónicos que serán especificados a continuación: (Baryolo y otros, 2008)

➤ Vistas

Vista de Sistema: Propone las partes del software: componentes, conectores, las restricciones y las configuraciones de estas partes, se subdivide en tres vistas fundamentales:

- Vista de Componentes: Encargada de las definiciones de los tipos de componentes posibles a definir en el proyecto, de la especificación de sus características, así como de la composición estructural interna de cada uno de estos componentes.
- Vista de Integración: Encargada de los procesos de integración interna y externa, establece las definiciones, estándares, protocolos de comunicación y reglas de intercambio de información.

- **Vista de Datos:** Encargada de todas las definiciones a nivel de datos, de la integración de los distintos modelos, de los patrones, estándares y definiciones a este nivel.

Vista Tecnológica: Es la base del software, propicia los elementos necesarios para crear el producto, esta a su vez se subdivide en dos vistas:

- **Vista de seguridad:** Chequea e implementa todos los aspectos relacionados con el acceso a la aplicación, la modificación, lectura o eliminación de la información, etc.
- **Vista de presentación:** Encargada de cómo luce el software, cuáles son los colores que lleva la aplicación, cómo son los botones, los vínculos y todos los elementos significativos desde el punto de vista de la presentación.

Vista de Infraestructura: Es la encargada de determinar la plataforma tecnológica a utilizar en la elaboración del producto, la definición y disponibilidad de los distintos servicios telemáticos necesarios en la confección del mismo, así como del diseño de los distintos escenarios de despliegue posibles.

➤ **Estilos arquitectónicos**

Arquitectura Basada en Componentes: Uno de los enfoques en los que actualmente se trabaja es la arquitectura basada en componentes que tiene como objetivo hacer un uso correcto de software reutilizable, para la construcción de aplicaciones mediante el ensamblaje de partes ya existentes. El equipo de arquitectura del proyecto ERP-Cuba dentro de la vista del sistema antes presentada incluye la vista de componente como una de sus subdivisiones; en ese caso propone que todas las funcionalidades levantadas y modeladas en las fases de negocio y requerimientos deben ser expresadas o contenidas en al menos un componente y que las distintas interacciones entre ellos originen funcionalmente la existencia de subsistemas en consecuencia a las dependencias definidas.

Modelo-Vista-Controlador (MVC): El patrón arquitectónico MVC es utilizado para el desarrollo de aplicaciones Web con el fin de separar en tres componentes distintos la interfaz de usuario, la lógica de negocio y los datos persistentes, potenciando la flexibilidad y la adaptabilidad a futuros cambios. Por su parte:

El Modelo: Es la representación de la información que maneja la aplicación. Son los datos puros que puestos en un contexto del sistema son mostrados al usuario por medio del Controlador, proveen de información al usuario o a la aplicación misma.

La Vista: Constituye la representación del modelo en forma gráfica, disponible para la interacción con el usuario. En una aplicación web la "Vista" es la página HTML con contenido dinámico sobre el cual el usuario puede realizar operaciones.

El Controlador: Se encarga de responder a las solicitudes del usuario desde la Interfaz, manejando los diferentes eventos a través de las funcionalidades necesarias y la información perteneciente al Modelo.

Para el desarrollo del Sistema Integral de Gestión Cedrux se decidió trabajar con este patrón, evidenciándose en los framework definidos para cada una de las capas como parte del MVC de cada componente dentro de la aplicación, es decir: para la Vista: Extjs-Framework, el cual es muy utilizado en el desarrollo de aplicaciones Web con tecnología AJAX, para el Controlador: Zend-Framework quien emplea específicamente el estilo Modelo-Vista-Controlador como base de su funcionamiento, y para agilizar el acceso a datos en el Modelo se utilizó Doctrine, un potente y completo sistema de Mapeo Objeto Relacional (ORM). De forma general, según lo descrito con anterioridad, en la arquitectura del Proyecto ERP-Cuba los estilos arquitectónicos que se emplean no se pueden ver de forma independiente sino como un estilo híbrido que comprende numerosas ventajas:

- Desarrollos paralelos: en cada capa.
- Aplicaciones más robustas debido al encapsulamiento.
- Mantenimiento y soporte más sencillo: es más sencillo cambiar un componente que modificar íntegramente una aplicación.
- Mayor flexibilidad: se pueden añadir nuevos módulos para dotar al sistema de nueva funcionalidad.
- Alta escalabilidad: La principal ventaja de una aplicación distribuida bien diseñada es su buen escalado, es decir, que puede manejar muchas peticiones con el mismo rendimiento simplemente añadiendo más hardware.

1.5. Conclusiones del capítulo

En este capítulo se hizo mención de los conceptos necesarios para un mejor entendimiento de la investigación científica en cuestión. Se realizó un estudio de las tendencias actuales, tanto nacionales como internacionales, enfocadas a la GCH por competencias y la importancia que tiene llevar a cabo este proceso en las diferentes empresas y organizaciones. Se abordó acerca del modelo de desarrollo orientado a componentes, a utilizar en el presente trabajo por políticas definidas en el proyecto, quedando claras sus características, fases, actividades que se realizan, artefactos que se generan y tecnología que se utiliza y finalmente se especificó la arquitectura sugerida por el programa ERP-Cuba para el desarrollo del sistema Cedrux.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

2.1. Introducción

En este capítulo se realiza la descripción de la propuesta de solución del presente trabajo, para ello se identifican los conceptos del negocio y se realiza el modelo conceptual. Se especifican además las técnicas y patrones empleados para la captura de requisitos y por último se identifican y describen los requisitos funcionales que debe presentar la solución a construir.

2.2. Descripción del proceso del negocio

El proceso Competencias Laborales es un proceso complejo dentro del área de Capital humano. Para llevarlo a cabo es necesario obtener información de otros procesos que se realizan dentro de esta área, como es el ejemplo de:

- El puesto de trabajo, perteneciente al proceso “Definir puestos de trabajo”. Es un documento que contiene los datos correspondientes al puesto de trabajo, entre ellos el área, el cargo al que éste pertenece y el estado en que se encuentra.
- La persona, proveniente del proceso “Registro de personas”. Se utiliza para obtener todos los datos de la persona, por ejemplo nombre, apellidos, carné de identidad y su categoría ocupacional.
- El trabajador, proveniente del proceso “Movimiento de fuerza de trabajo”. Este documento brinda los datos necesarios de los trabajadores de la entidad, entre los que se encuentran su expediente interno, nombre, apellidos, área, cargo y puesto de trabajo al que pertenecen.

El proceso en cuestión de forma general cuenta con 2 procesos fundamentales, un primer proceso que se realiza para definir las competencias de los diferentes niveles; y un segundo proceso que tiene como objetivo fundamental certificar las competencias de las personas y trabajadores. Para llevar a cabo el primer proceso un Comité de competencias se reúne e identifica las competencias para las diferentes áreas, cargos y puestos de trabajo, las cuales son validadas y aprobadas posteriormente por la alta dirección y recogidas en un solo documento. En el segundo proceso se definen y validan las competencias presentes en las personas y trabajadores, de acuerdo a las competencias identificadas y aprobadas en el proceso anterior.

Al culminar el proceso de competencias laborales se generan una serie de documentos que constituyen salidas hacia otros procesos, estos son:

- Profesiograma: este documento contiene las competencias de las diferentes áreas, cargos y puestos de trabajo de la entidad, y es utilizado en los procesos de Selección y Reclutamiento.
- Expediente de competencias de la persona: documento que contiene las competencias de la persona, y es utilizado en el proceso de Selección.

En la Figura 1 se muestra el mapa de procesos, donde se observan claramente todos los procesos involucrados y las relaciones existentes entre ellos:

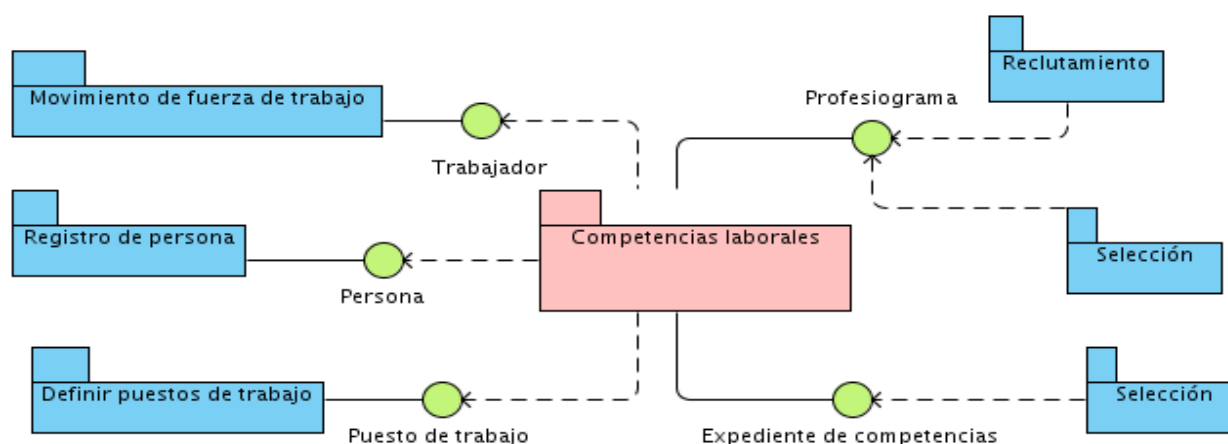


Ilustración 2: Mapa de procesos

2.3. Técnicas y métodos empleados en la captura de requisitos

La ingeniería de requerimientos se realiza para recopilar, analizar y verificar las necesidades del cliente o usuario, la cual tiene como meta fundamental la obtención de una especificación de requisitos de software correcta y completa, razón por la cual durante esta etapa el equipo de desarrollo realiza técnicas que le permitan obtener, documentar y desarrollar todas las tareas de una forma más sencilla. La gran mayoría de estas técnicas son muy útiles y fundamentales para capturar los requisitos del software de manera que permitan obtener la mayor cantidad de información posible del sistema a informatizar. (Chaves, 2006)

A continuación se explican brevemente las empleadas en este trabajo:

Entrevista: Se realiza para comprender el problema y los objetivos del sistema que se desea proponer reuniendo información proveniente de personas o grupos (Chaves, 2006). En este caso el entrevistado fue el funcional Manuel Iglesias con el objetivo de que quedaran definidos correctamente los requisitos necesarios para el proceso de competencias laborales.

Prototipos: Los prototipos son simulaciones del posible producto, que luego son utilizados por el usuario final, permitiendo conseguir una importante retroalimentación en cuanto a si el sistema diseñado con base a los requerimientos recolectados le permite al usuario realizar su trabajo de manera eficiente y efectiva. Este método es muy útil para la captura de los requisitos porque le permite al usuario hacerse una idea de la estructura de la interfaz del sistema pudiéndose incorporar ideas o sugerencias que este haga en función de que el producto final cumpla con sus expectativas (Chaves, 2006). Para realizar los prototipos del presente trabajo se siguió el estándar de diseño de interfaces para las aplicaciones de gestión definido por el proyecto ERP-Cuba.

Glosario: El glosario es una simple lista de términos en donde se explica su significado. En esta lista se incluyen todos los términos que requieren explicación, mejorando así la comunicación intergrupala y la comunicación con el cliente, además de mitigar el riesgo de malos entendidos. Los términos que se incluyen provienen de todas las áreas del proyecto. El glosario se va actualizando durante el transcurso del proceso de IR, perfeccionándolo en cada nuevo ciclo.

Sistemas Existentes: Esta técnica consiste en analizar los distintos sistemas que pueden relacionarse con el sistema que se desea construir. Con esta técnica se pueden analizar las interfaces de usuario y las funciones que se realizan en cada una de ellas, tanto así como la información que se maneja y la forma en que puede ser manipulada (Chaves, 2006). Los sistemas estudiados se pueden encontrar en el capítulo 1 en el epígrafe 1.2 Gestión por competencias.

Arqueología de documentos: Con esta técnica los analistas tratan de determinar posibles requerimientos sobre la base del análisis de la documentación utilizada por la empresa. Se debe recolectar cualquier formulario o documento que sea utilizado para registrar o enviar información. Este método por sí solo no es efectivo, sirve más que nada como complemento de otras técnicas, y ayuda a obtener información que de otra manera sería sumamente difícil conseguir (Fidel, 2009). Para la realización del presente trabajo se utilizaron documentos como: CH-CL-01, Competencias laborales,

Descripción del negocio. ERP-Cuba, Capital Humano, por Crecencio Campos Hung; CH-CL-02, Competencias laborales, Procesos del negocio. ERP-Cuba, Capital Humano, por Crecencio Campos Hung y el grupo de Normas cubanas 3000, 3001, 3002: Sistema de Gestión Integrada de Capital Humano.

2.4. Modelo conceptual

Los modelos conceptuales se utilizan para representar la realidad a un alto nivel de abstracción. Mediante estos se puede construir una descripción de la realidad fácil de entender. En el modelo que se presenta en la Figura 2 se muestran las entidades que conforman el proceso Competencias laborales:

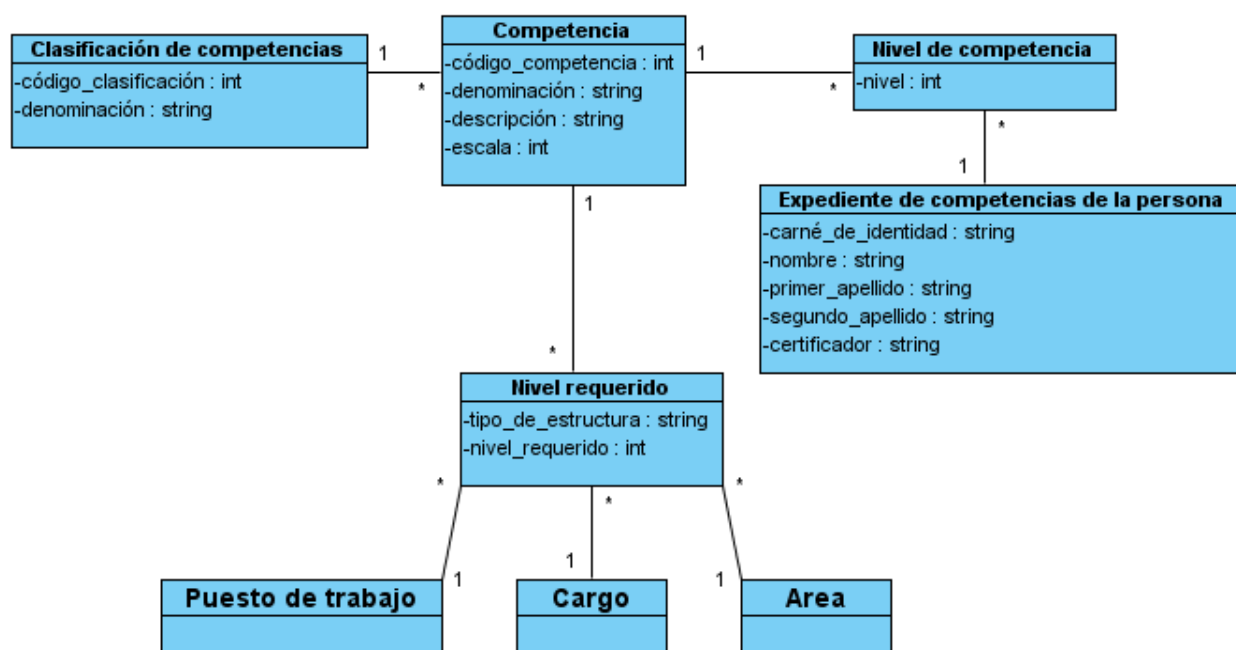


Ilustración 3: Modelo Conceptual

Competencia

Es el documento donde quedan registradas todas las competencias definidas por la empresa, mostrándose de estas su denominación, clasificación y escala. Este documento es utilizado posteriormente en el expediente de competencias de la persona y en el profesiograma para asignarle competencias a determinada persona, trabajador o puesto de trabajo.

Clasificación de competencias

Documento que se realiza con el objetivo de registrar los tipos de competencia definidos por la empresa, los cuales se utilizan posteriormente para darle una clasificación a cada una de las competencias registradas.

Nivel de competencia:

Este representa el nivel que posee la persona en sus competencias, de acuerdo a la forma definida por la empresa para evaluar éstas, ya sea cualitativa o cuantitativamente. Es utilizado en el expediente de competencias de la persona.

Expediente de competencias de la persona:

Documento que se crea para registrar las competencias de la persona o trabajador, mostrándose de cada una de ellas denominación, clasificación, nivel de competencia y la fecha en que fue certificada.

Nivel requerido:

Representa el nivel que se necesita en las competencias para pertenecer a determinada área, cargo o puesto de trabajo, según la escala que se le definió a cada una de las competencias de la empresa. Es utilizado en el expediente de competencias del trabajador.

2.5. Patrones de análisis

Un patrón es una solución a un problema en un contexto, codifica conocimiento específico acumulado por la experiencia en un dominio. Un sistema bien estructurado está lleno de patrones. Cada patrón describe un problema que ocurre una y otra vez, y luego describe el núcleo de la solución a ese problema, de tal manera que se puede usar esa solución un millón de veces más sin hacer jamás la misma cosa dos veces. (Larman)

2.5.1. Patrón CRUD

Acrónimo de Create-Read-Update-Delete (Crear-Leer-Modificar-Eliminar). Conocido como el padre de todos los patrones de capa de acceso. Propone identificar en un caso de uso, todas las diferentes operaciones que se pueden realizar sobre una parte de información de cierto tipo (o sea en una misma entidad), tal como crearla, leerla, actualizarla y eliminarla. Debe ser usado cuando todos los flujos contribuyen al mismo valor de negocio, son cortos y sencillos.

2.5.2. Nombres que revelan la intención (IntentionRevealingName).

El nombre debe reflejar la intención del requerimiento y reflejar un único objetivo e intención que se está intentando lograr. Se debe asignar un nombre apropiado que facilite el manejo del requerimiento, permitiendo tener una vista general del trabajo en su conjunto.

2.5.3. Escenario más Fragmentos (ScenarioPlusFragments).

El usuario debe ser capaz de seguir el camino básico en el que está interesado, en caso contrario le sería muy difícil entender la información que busca. Para ello se describen las acciones del flujo básico como un escenario simple, sin tener en cuenta posibles fallos. Debajo se describen los eventos que explican qué flujo alternativo puede ocurrir. Para comprender mejor este patrón se aconseja consultar la sección 2.6.1 Especificaciones de requisitos.

2.5.4. Preciso y legible

Cada requisito que se escriba debe exactamente describir una Meta única y completa sin ser tan verboso que la audiencia no lo pueda leer o de tan alto nivel que no comunique la suficiente información para entenderlo adecuadamente. Los niveles más altos de formalidad en las especificaciones dan a los desarrolladores un sentido falso de seguridad. Nada puede reemplazar el diálogo con el cliente. Para comprender mejor este patrón se aconseja consultar la sección 2.6.1 Especificaciones de requisitos.

2.5.5. Completar una única meta.

Este patrón plantea que cada requisito debe ser descrito con un objetivo bien definido. Se debe ser consistente entre las metas que se describen. Para comprender mejor este patrón se aconseja consultar la sección 2.6.1 Especificaciones de requisitos.

2.5.6. Alternativas Exhaustivas (ExhaustiveAlternatives).

Este patrón establece que un requerimiento puede tener varias alternativas, identificándose en cada caso el flujo normal de eventos y capturando los posibles fallos. Para comprender mejor este patrón se aconseja consultar la sección 2.6.1 Especificaciones de requisitos.

2.5.7. Condiciones detectables (DetectableConditions).

Un sistema no puede manejar eventos que no pueda detectar, los desarrolladores necesitan conocer qué situaciones detectar. Se debe capturar cada posibilidad razonable, de otra manera, el sistema nunca

estará capacitado para ejecutar el escenario. Descubrir una condición olvidada después de que el sistema ha introducido servicios es incluso más caro. Para comprender mejor este patrón se aconseja consultar la sección 2.6.1 Especificaciones de requisitos.

2.5.8. Pasos Nivelados (LeveledSteps).

Excesivamente largos o excesivamente pequeños pasos en el requisito oscurecen la meta y lo hacen difícil de leer y comprender. Cada escenario debe tener de tres a nueve pasos. Idealmente son todos a niveles similares un nivel de la abstracción justamente debajo de la meta del requerimiento. Esto no quiere decir que un requisito no pueda tener más de 9 pasos si lo requiere. Para comprender mejor este patrón se aconseja consultar la sección 2.6.1 Especificaciones de requisitos.

2.6. Requisitos funcionales

Los requisitos funcionales establecen los comportamientos del sistema. Definen lo que el sistema tiene que hacer, los servicios que debe proporcionar al usuario. A continuación se muestran los requisitos funcionales que fueron definidos de acuerdo con los objetivos planteados:

RF1 Gestionar clasificación de competencias.

RF 1.1 Adicionar clasificación.

RF 1.2 Modificar clasificación.

RF 1.3 Eliminar clasificación.

RF 1.4 Listar clasificaciones.

RF 2 Gestionar competencias.

RF 2.1 Adicionar competencia.

RF 2.2 Modificar competencia.

RF 2.3 Eliminar competencia.

RF 2.4 Listar competencias.

RF 2.5 Imprimir competencias.

RF 3 Gestionar expediente de competencias de la persona.

RF 3.1 Adicionar competencia a la persona.

RF 3.2 Modificar competencia de la persona.

RF 3.3 Eliminar competencia de la persona.

RF 3.4 Listar competencias de la persona.

RF 3.5 Imprimir competencias de la persona.

RF 4: Gestionar profesiograma.

RF 4.1 Adicionar competencia por estructura.

RF 4.2 Modificar competencia por estructura.

RF 4.3 Eliminar competencia por estructura.

RF 4.4 Listar competencias por estructura.

RF 4.5 Imprimir competencias del área.

RF 4.6 Imprimir competencias del cargo.

RF 4.7 Imprimir competencias del puesto de trabajo.

Como se puede observar quedaron definidas cuatro agrupaciones de requisitos, para un total de 21 requisitos funcionales.

2.6.1. Especificaciones de requisitos

En las especificaciones de requisitos se registran las características y condiciones que debe cumplir cada requisito funcional. A continuación se expondrá una breve descripción de los requisitos del Gestionar expediente de competencias de la persona, el resto se encuentra desde el Anexo 1 al Anexo 16, y en el Anexo 17 se muestra la plantilla de Especificación de requisitos utilizada en el proyecto y por tanto para el presente trabajo.

RF Gestionar expediente de competencias de la persona.

Resumen de la especificación del requisito Adicionar competencia a la persona.

Tabla 2: Resumen de la especificación del requisito Adicionar competencia a la persona

Precondiciones	Tiene que existir una persona registrada en el sistema. Tiene que existir al menos una competencia en el sistema.
Flujo de eventos	
Flujo básico	
1	El sistema permite buscar la persona introduciendo el número de carné de identidad o mediante la búsqueda avanzada.
2	El sistema muestra nombre, apellidos y número de carné de identidad.

3	Se introducen los datos del documento: Competencia: se selecciona la competencia que se le va a adicionar a la persona Nivel de competencia: se selecciona el nivel que posee la persona en esa competencia Certificador: se muestra el nombre de la persona que le certifica esa competencia a la persona, es el usuario que está autenticado en el sistema
4	El sistema valida (ver validación 1) los datos introducidos.
5	El sistema valida (Ver validación 2) la ubicación de la persona.
6	Si los datos son correctos el sistema los registra.
7	El sistema confirma el registro de los datos.
8	Concluye el requisito.
Pos-condiciones	
1	Se le adicionó una nueva competencia a la persona.
Validaciones	
1	Validar que el código no se repita.

Resumen de la especificación del requisito Modificar competencia de la persona.

Tabla 3: Resumen de la especificación del requisito Modificar competencia de la persona

Precondiciones	Se le ha registrado a la persona al menos una competencia en el sistema.
Flujo de eventos	
Flujo básico	
1	Se selecciona la competencia a modificar.
2	El sistema muestra y permite editar los datos de la competencia.
3	Se introducen los datos del documento: Competencia: se selecciona la competencia que se le va a adicionar a la persona Nivel de competencia: se selecciona el nivel que posee la persona en esa competencia Certificador: se muestra el nombre de la persona que le certifica esa competencia a la persona, es el usuario que está autenticado en el sistema.
4	El sistema valida (ver validación 1) los datos introducidos.
5	El sistema valida (Ver validación 2) la ubicación de la persona.
6	Si los datos son correctos el sistema los registra.
7	El sistema confirma el registro de los datos.
8	Concluye el requisito.
Pos-condiciones	
1	Se modificaron los datos de la competencia de la persona.
Validaciones	
1	Validar que el código no se repita.

Resumen de la especificación del requisito Eliminar competencia de la persona.

Tabla 4: Resumen de la especificación del requisito Eliminar competencia de la persona

Precondiciones	Se le ha registrado a la persona al menos una competencia en el sistema.
Flujo de eventos	
Flujo básico	
1	Se selecciona la competencia a eliminar.

2	Se solicita confirmación para eliminar la competencia.
3	Si el usuario confirma si elimina la competencia.
4	El sistema confirma la eliminación.
5	Concluye el requisito.
Pos-condiciones	
1	Se eliminó la competencia de la persona.
Validaciones	
1	No se puede eliminar una competencia que se encuentre en uso.

Resumen de la especificación del requisito Listar competencias de la persona.

Tabla 5: Resumen de la especificación del requisito Listar competencias de la persona

Precondiciones	Se le ha registrado a la persona al menos una competencia en el sistema.
Flujo de eventos	
Flujo básico	
1	El sistema muestra un listado de las competencias. Se muestran: Código Denominación Clasificación Nivel de competencia Fecha de certificación
2	Concluye el requisito.
Pos-condiciones	
1	N/A
Validaciones	
1	N/A

Resumen de la especificación del requisito Imprimir competencias de la persona.

Tabla 6: Resumen de la especificación del requisito Imprimir competencias de la persona

Precondiciones	Se le ha registrado a la persona al menos una competencia en el sistema.
Flujo de eventos	
Flujo básico	
1	El sistema permite seleccionar los elementos que se desean imprimir.
2	El sistema imprime los datos (ver formatos de salida).
3	Concluye el requisito.
Pos-condiciones	
1	N/A
Validaciones	
1	N/A

2.7. Conclusiones del capítulo

En el presente capítulo luego de quedar identificados los procesos del negocio se realizó el modelo conceptual, el cual tenía como objetivo identificar las entidades del negocio. Se especificaron además las técnicas y patrones empleados para la captura de requisitos y por último se identificaron y describieron los requisitos funcionales, quedando así satisfechas las necesidades reales de los usuarios y clientes.

CAPÍTULO 3: DISEÑO DEL SISTEMA

3.1. Introducción

En este capítulo se mostrarán los detalles relacionados con el diseño del sistema propuesto, utilizando para su modelado los diagramas de clase, el diagrama de componentes y el diagrama entidad relación, con la aplicación de los patrones correspondientes; realizando posteriormente una descripción tanto de los diagramas como de los patrones utilizados.

3.2. Patrones de diseño

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.

3.2.1. Patrones GRASP

Los patrones GRASP (patrones generales de software para asignación de responsabilidades) del acrónimo General Responsibility Assignment Software Patterns, describen los principios fundamentales de diseño de objetos para la asignación de responsabilidades. Actualmente se conocen nueve patrones GRASP: Experto, Creador, Alta Cohesión, Bajo Acoplamiento, Controlador, Polimorfismo, Fabricación Pura, Induración y No Hables con Extraños. De éstos se utilizaron los siguientes:

Bajo Acoplamiento

Consiste en tener las clases lo menos ligadas entre sí que se pueda, de forma que si se produce una modificación en alguna de ellas, tenga la menor repercusión posible en las demás clases, fortaleciendo así la reutilización, y disminuyendo la dependencia entre las clases.

Alta Cohesión

Este patrón permite que la información que almacena una clase sea coherente y esté en la mayor medida de lo posible relacionada con la clase. Realizando un diseño donde las clases mantengan una alta cohesión se mejora la claridad y facilidad con que se entiende el diseño, se simplifica el mantenimiento y las mejoras de funcionalidad, soporta mayor capacidad de reutilización. (Fidel, 2009)

Experto

Este patrón es el principio básico de asignación de responsabilidades. Indica, por ejemplo, que la responsabilidad de la creación de un objeto o la implementación de un método, debe recaer sobre la clase que contiene toda la información necesaria para crearlo. De este modo se obtendrá un diseño con mayor cohesión y así la información se mantiene encapsulada (disminución del acoplamiento). Se utiliza en las clases controladoras, modelos y de entidad, las cuales poseen funciones específicas de acuerdo con la información que gestionan.

Creador

Ayuda a identificar quién debe ser el responsable de la creación (o instanciación) de nuevos objetos o clases. En los diagramas de clase realizados este patrón se evidencia en el momento en que las clases controladoras se responsabilizan de crear objetos de las modelos, y estas por su parte de crear objetos de las entidades.

Controlador

Sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos del usuario y la que los envía a las distintas clases según el método llamado. Este patrón sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto para aumentar la reutilización de código y a la vez tener un mayor control.

3.2.2. Patrones GoF (Gang of Four).

Son patrones de diseño publicados en el libro Design Patterns: Elements of Reusable Object-Oriented Software conocidos por "Gang of Four" o Pandilla de los cuatro. En este libro se encuentran recopilados un total de 23 patrones clasificados en tres grandes grupos: creacionales, estructurales y de comportamiento. De ellos se utilizaron los que se muestran a continuación:

Creacionales

Se encargan de la creación de los objetos ayudando a que el sistema sea independiente de la creación, composición y representación de los objetos.

- **Singleton:** Este patrón asegura que una clase tendrá solo una instancia y provee un punto de acceso global a la misma. El constructor es privado y el método instance () es el que devuelve la

única instancia de esta clase o la crea si no existe. Se utiliza para las clases que se necesita que exista un solo ejemplar de ellas, evitando de esta forma sobrecargas o problemas de seguridad.

Estructurales

Son los encargados de cómo las clases y objetos están compuestos para formar estructuras más grandes. Los patrones estructurales usan la herencia para componer interfaces u objetos en tiempo de ejecución.

- **Facade (Fachada):** Define una interfaz de alto nivel para un conjunto de interfaces de un subsistema que hace que éste sea más fácil de utilizar. Ejemplo donde se aplica este patrón es en los Servicios, ya que la relación existente entre estos y las clases Controller (Controladoras) posibilita utilizar métodos de otros componentes que se encuentran tanto dentro como fuera del subsistema Capital Humano.

Comportamiento

Plantean algoritmos y la asignación de responsabilidades entre objetos. Estos patrones no solo describen clases y objetos sino también describen la comunicación entre ellos.

- **Cadena de responsabilidad:** permite establecer una cadena de objetos receptores a través de los cuales se pasa una petición formulada por un objeto emisor. Cualquiera de los objetos receptores puede responder a la petición en función de un criterio establecido. En el presente trabajo se utiliza en la mayoría de los diagramas de clases, aplicándose en el tratamiento de excepciones.

3.3. Diagramas de clases del diseño

Los diagramas de clases del diseño describen la estructura de un sistema mostrando sus clases, atributos, operaciones y las relaciones existentes entre ellos. Cada clase del diseño se encarga de realizar una función específica. Las clases controladoras o controller permiten la gestión de información entre la vista y el modelo; las clases del negocio definen la lógica del negocio de la aplicación; las clases del dominio son las responsables de acceder mediante consultas a la base de datos y las clases vista se encargan de mostrar la información recibida del modelo al usuario. Contienen la siguiente información:

- Clases asociadas.
- Métodos
- Navegabilidad
- Dependencias

En las Figuras 4, 5, 6 y 7 se muestran los 4 diagramas de clases del diseño obtenidos y las descripciones de las principales clases:

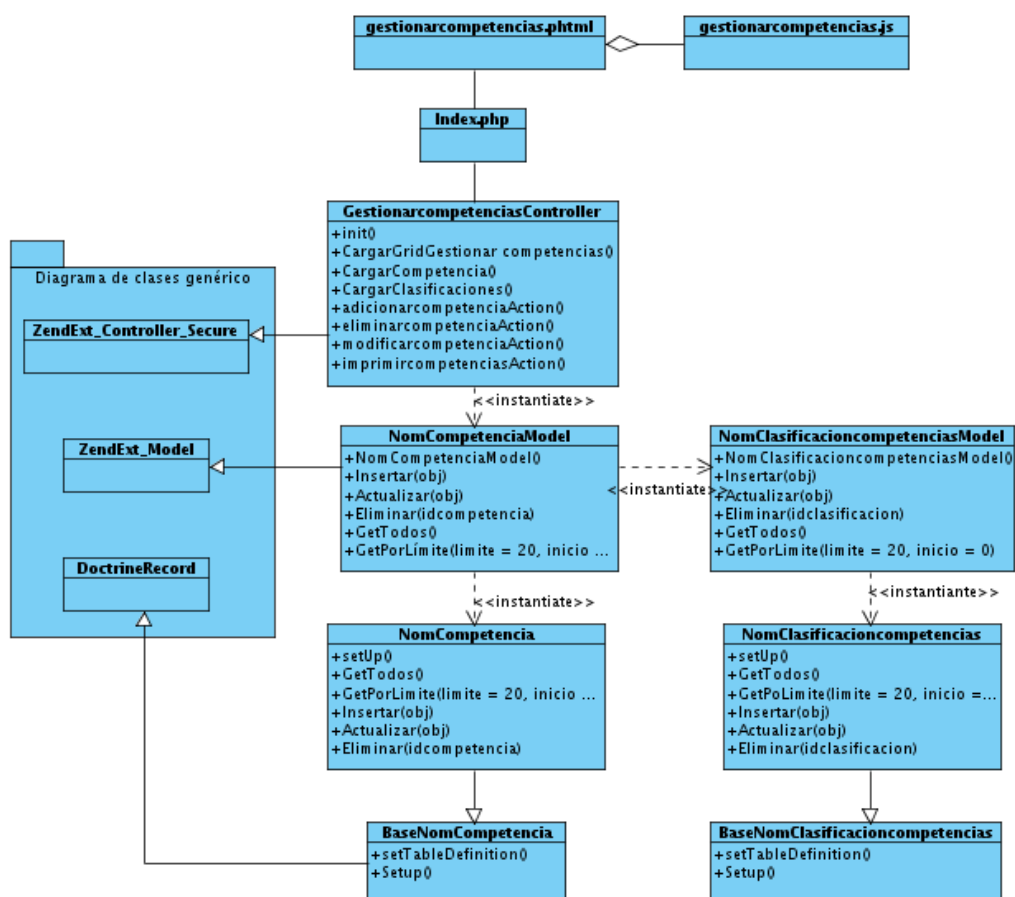


Ilustración 4: Diagrama de clases Gestionar competencias

Tabla 7: Descripción de la clase GestionarcompetenciasController

Nombre: GestionarcompetenciasController	
Tipo de clase: Controladora	
Para cada responsabilidad	
Nombre	Descripción
init()	Se inicializan los parámetros
CargarGridGestionarcompetencias()	Este método se encarga de cargar los datos de la ventana del gestionar competencias.
CargarCompetencia()	Este método se encarga de obtener los datos de la competencia y mostrarlos al usuario al seleccionar modificar

	competencia.
CargarClasificaciones()	Este método carga las clasificaciones que se encuentran registradas en la base de datos y las muestra en el Combo Box clasificación.
adicionarcompetenciaAction()	Este método toma todos los datos de una competencia y los adiciona en la base de datos.
modificarcompetenciaAction()	Este método modifica los datos de una competencia y los guarda en la base de datos.
eliminarcompetenciaAction()	Este método elimina todos los datos de una competencia colocándole una fecha de eliminación en la base de datos.
ImprimircompetenciasAction()	Este método imprime el listado de las competencias existentes en la entidad.

Tabla 8: Descripción de la clase NomCompetenciaModel

Nombre: NomCompetenciaModel	
Tipo de clase: Modelo (Business)	
Para cada responsabilidad	
Nombre	Descripción
NomCompetenciaModel()	Constructor de la clase.
Insertar(obj)	Este método recibe por parámetros un objeto y hace un llamado al método insertar de la clase NomCompetencia del dominio.
Actualizar(obj)	Este método recibe por parámetros un objeto y hace un llamado al método actualizar de la clase NomCompetencia del dominio.
Eliminar(idcompetencia)	Este método recibe por parámetros el id de competencia y hace un llamado al método eliminar de la clase NomCompetencia del dominio.
GetTodos()	Este método hace un llamado al método GetTodos () de la case NomCompetencia del dominio.
GetPorLimite(limite=20, inicio=0)	Este método recibe por parámetros un rango y hace un llamado al método GetPorLimite de la clase NomCompetencia del dominio.

Tabla 9: Descripción de la clase NomCompetencia

Nombre: NomCompetencia	
Tipo de clase: Modelo (Domain)	
Para cada responsabilidad	
Nombre	Descripción
setUp()	Constructor de la clase.
GetTodos()	Este método devuelve todas las competencias que existen en la base de datos.
GetPorLimite(limite=20, inicio=0)	Este método devuelve todas las competencias de acuerdo a un

	rango.
Insertar(obj)	Este método recibe como parámetro un objeto insertándolo en la base de datos.
Actualizar(obj)	Este método recibe como parámetro un objeto actualizándolo en la base de datos.
Eliminar(idcompetencia)	Este método recibe como parámetro un id de competencia y la elimina en la base de datos.

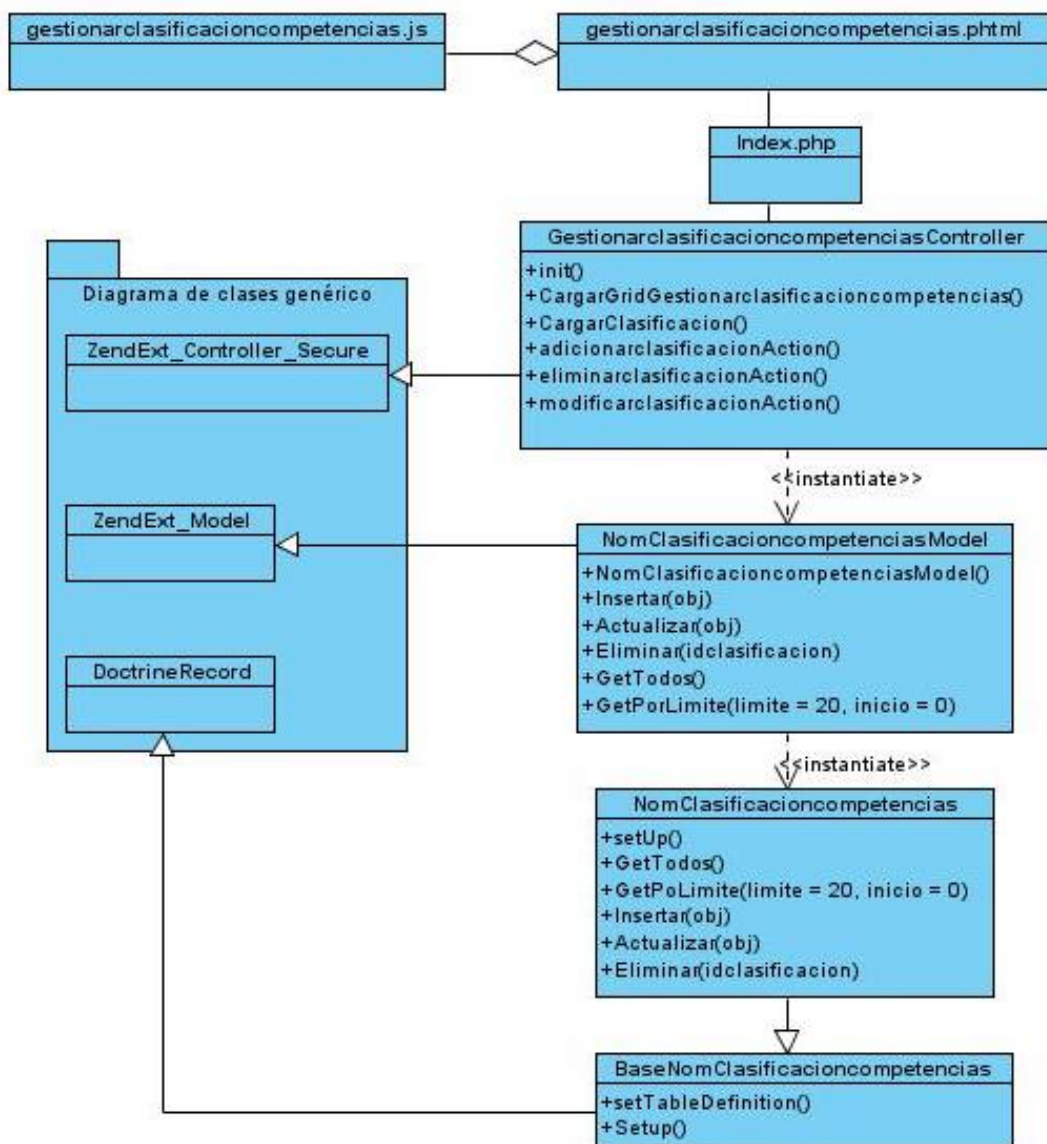


Ilustración 5: Diagrama de clases Gestionar clasificación de competencias

Tabla 10: Descripción de la clase GestionarclasificacioncompetenciasController

Nombre: GestionarclasificacioncompetenciasController	
Tipo de clase: Controladora	
Para cada responsabilidad	
Nombre	Descripción
init()	Se inicializan los parámetros
CargarGridGestionarclasificacioncompetencias() ()	Este método se encarga de cargar los datos de la ventana del gestionar clasificación competencias.
CargarClasificación()	Este método se encarga de obtener los datos de la clasificación y mostrarlos al usuario al seleccionar modificar clasificación.
adicionarclasificacionAction()	Este método toma todos los datos de una clasificación y los adiciona en la base de datos.
modificarclasificacionAction()	Este método modifica los datos de una clasificación y los guarda en la base de datos.
eliminarclasificacionAction()	Este método elimina todos los datos de una clasificación colocándole una fecha de eliminación en la base de datos.

Tabla 11: Descripción de la clase NomClasificacioncompetenciasModel

Nombre: NomClasificacioncompetenciasModel	
Tipo de clase: Modelo (Business)	
Para cada responsabilidad	
Nombre	Descripción
NomCompetenciaModel()	Constructor de la clase.
Insertar(obj)	Este método recibe por parámetros un objeto y hace un llamado al método insertar de la clase NomClasificacioncompetencias del dominio.
Actualizar(obj)	Este método recibe por parámetros un objeto y hace un llamado al método actualizar de la clase NomClasificacioncompetencias del dominio.
Eliminar(idclasificacion)	Este método recibe por parámetros el id de la clasificación y hace un llamado al método eliminar de la clase NomClasificacioncompetencias del dominio.
GetTodos()	Este método hace un llamado al método GetTodos () de la case NomClasificacioncompetencias del dominio.
GetPorLímite(límite=20, inicio=0)	Este método recibe por parámetros un rango y hace un llamado al método GetPorLímite de la clase NomClasificacioncompetencias del dominio.

Tabla 12: Descripción de la clase NomClasificacioncompetencias

Nombre: NomClasificacioncompetencias	
Tipo de clase: Modelo (Domain)	
Para cada responsabilidad	
Nombre	Descripción
setUp()	Constructor de la clase.
GetTodos()	Este método devuelve todas las clasificaciones que existen en la base de datos.
GetPorLímite(limite=20, inicio=0)	Este método devuelve todas las clasificaciones de acuerdo a un rango.
Insertar(obj)	Este método recibe como parámetro un objeto insertándolo en la base de datos.
Actualizar(obj)	Este método recibe como parámetro un objeto actualizándolo en la base de datos.
Eliminar(idclasificacion)	Este método recibe como parámetro un id de clasificación y la elimina en la base de datos.

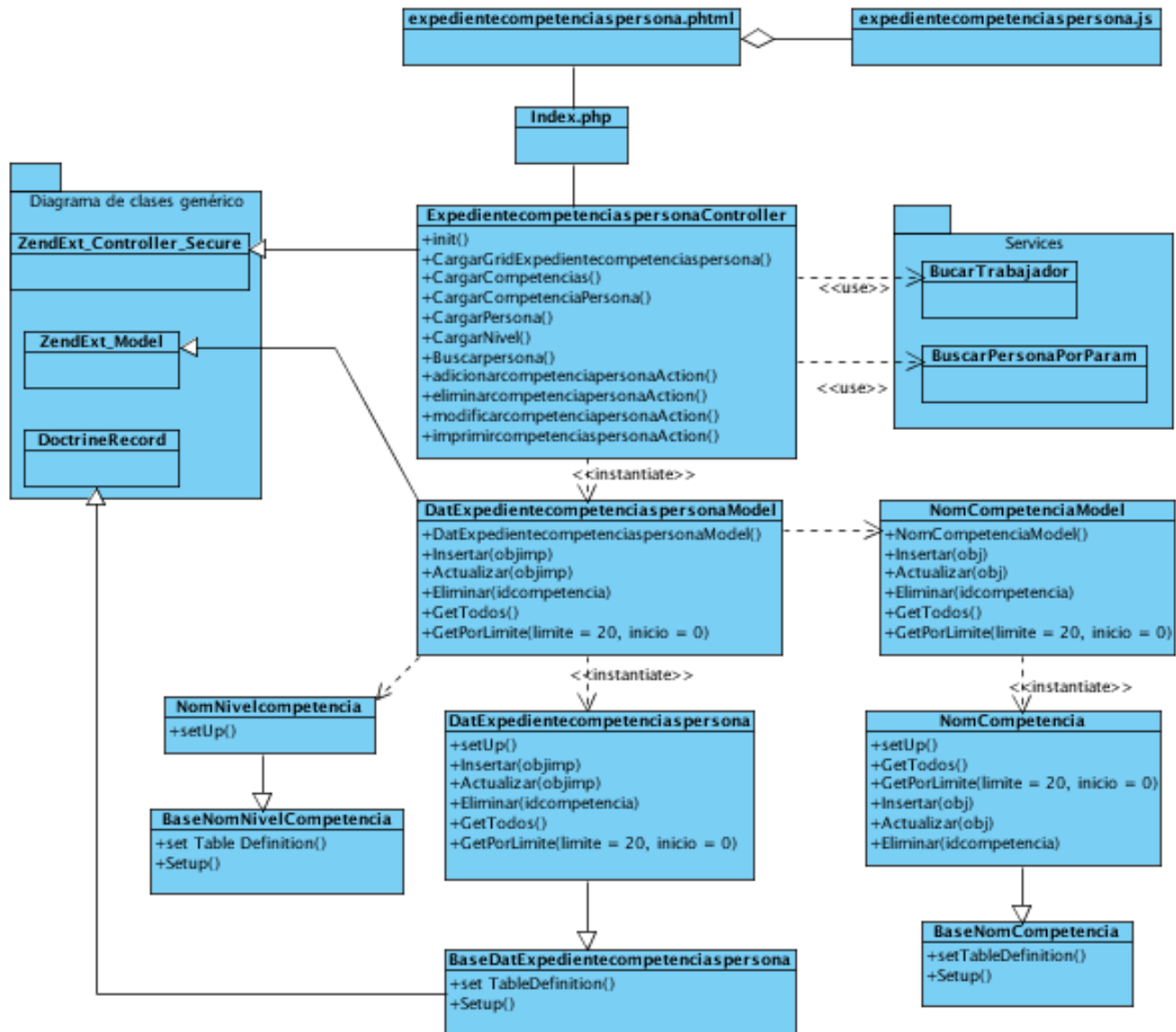


Ilustración 6: Diagrama de clases Expediente de competencias de la persona

Tabla 13: Descripción de la clase ExpedientecompetenciaspersonaController

Nombre: ExpedientecompetenciaspersonaController	
Tipo de clase: Controladora	
Para cada responsabilidad	
Nombre	Descripción

init()	Se inicializan los parámetros
CargarGridExpedientecompetenciaspersona()	Este método se encarga de cargar los datos de la ventana del expediente de competencias de la persona.
CargarCompetencias()	Este método carga las competencias que se encuentran registradas en la base de datos y las muestra en el Combo Box competencia.
CargarCompetenciapersona()	Este método se encarga de obtener los datos de la competencia de la persona y mostrarlos al usuario al seleccionar modificar competencia.
CargarPersona()	Este método se encarga de obtener los datos de la persona y mostrarlos al usuario al seleccionar el Buscar persona.
CargarNivel()	Este método carga los niveles de competencia que se encuentran registrados en la base de datos y los muestra en el Combo Box nivel de competencia.
BuscarpersonaAction()	Este método permite buscar los datos de la persona o trabajador mediante los servicios BuscarPersonaPorParam o BuscarTrabajador respectivamente.
adicionarcompetenciapersonaAction()	Este método toma todos los datos de una competencia de la persona y los adiciona en la base de datos.
modificarcompetenciapersonaAction()	Este método modifica los datos de una competencia de la persona y los guarda en la base de datos.
eliminarcompetenciapersonaAction()	Este método elimina todos los datos de una competencia de la persona colocándole una fecha de eliminación en la base de datos.
ImprimircompetenciaspersonaAction()	Este método imprime el listado de las competencias de la persona existentes en la entidad.

Tabla 14: Descripción de la clase DatExpedientecompetenciaspersonaModel

Nombre: DatExpedientecompetenciaspersonaModel	
Tipo de clase: Modelo (Business)	
Para cada responsabilidad	
Nombre	Descripción
DatExpedientecompetenciaspersonaModel()	Constructor de la clase.
Insertar(obj)	Este método recibe por parámetros un objeto y hace un llamado al método insertar de la clase DatExpedientecompetenciaspersona del dominio.

Actualizar(obj)	Este método recibe por parámetros un objeto y hace un llamado al método actualizar de la clase DatExpedientecompetenciaspersona del dominio.
Eliminar(idcompetencia)	Este método recibe por parámetros el id de la competencia y hace un llamado al método eliminar de la clase DatExpedientecompetenciaspersona del dominio.
GetTodos()	Este método hace un llamado al método GetTodos() de la clase DatExpedientecompetenciaspersona del dominio.
GetPorLimite(limite=20, inicio=0)	Este método recibe por parámetros un rango y hace un llamado al método GetPorLimite de la clase DatExpedientecompetenciaspersona del dominio.

Tabla 15: Descripción de la clase DatExpedientecompetenciaspersona

Nombre: DatExpedientecompetenciaspersona	
Tipo de clase: Modelo (Domain)	
Para cada responsabilidad	
Nombre	Descripción
setUp()	Constructor de la clase.
Insertar(obj)	Este método recibe como parámetro un objeto insertándolo en la base de datos.
Actualizar(obj)	Este método recibe como parámetro un objeto actualizándolo en la base de datos.
Eliminar(idcompetencia)	Este método recibe como parámetro un id de competencia y la elimina en la base de datos.
GetTodos()	Este método devuelve todas las competencias de la persona que existen en la base de datos.
GetPorLimite(limite=20, inicio=0)	Este método devuelve todas las competencias de la persona de acuerdo a un rango.

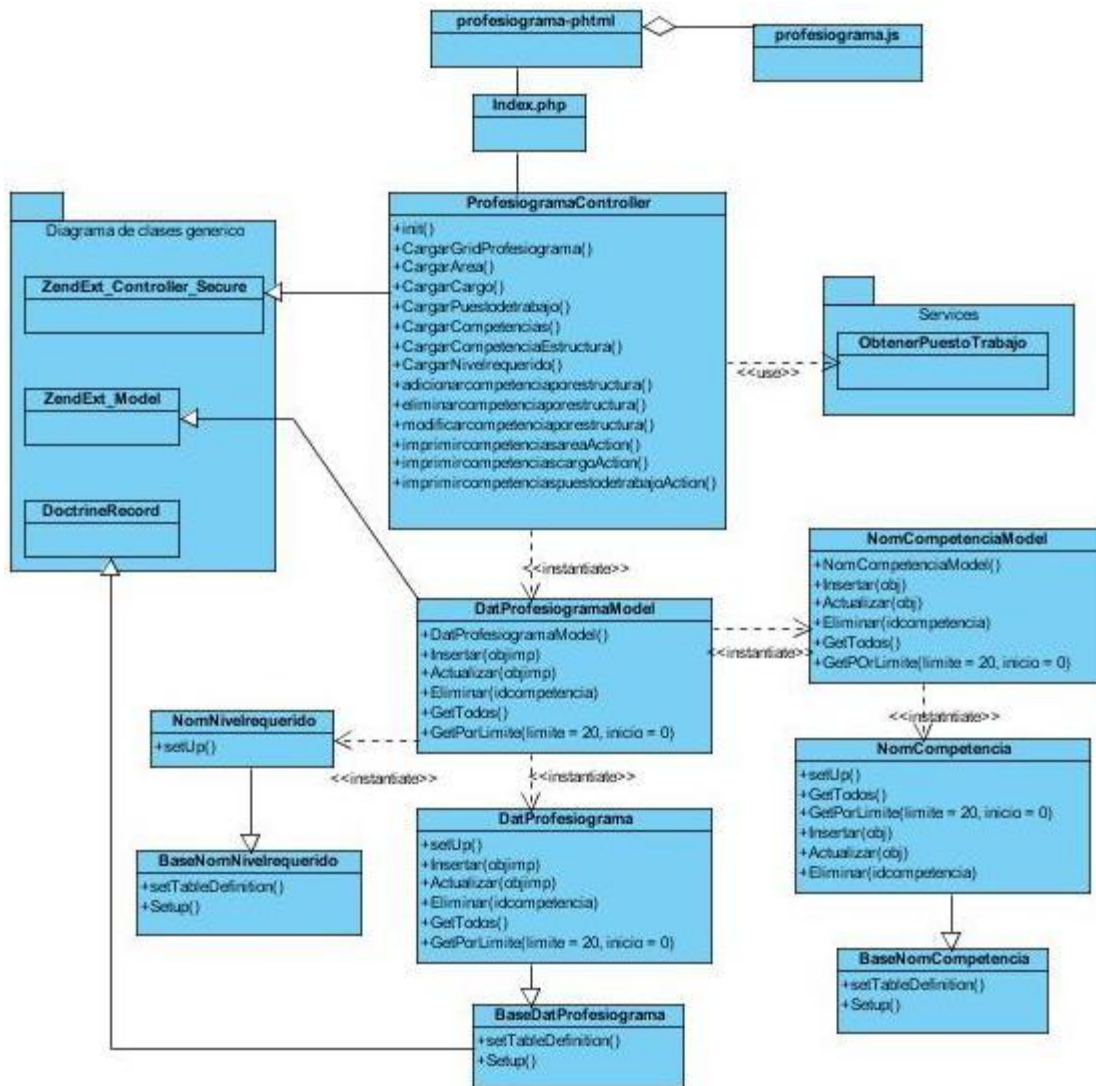


Ilustración 7: Diagrama de clases Profesigramas

Tabla 16: Descripción de la clase ProfesiogramaController

Nombre: ProfesiogramaController	
Tipo de clase: Controladora	
Para cada responsabilidad	
Nombre	Descripción
init()	Se inicializan los parámetros
CargarGridProfesiograma()	Este método se encarga de cargar los datos de la ventana del Profesigramas.

CargarArea() CargarCargo() CargarPuestodetrabajo()	Funcionalidades que se encargan de cargar las áreas, cargos y puestos de trabajo que se encuentran registrados en la base de datos mediante el servicio ObtenerPuestoTrabajo y los muestra organizados jerárquicamente.
CargarCompetencias()	Este método carga las competencias que se encuentran registradas en la base de datos y los muestra en el Combo Box competencia.
CargarCompetenciaEstructura()	Este método se encarga de obtener los datos de la competencia de la estructura y mostrarlos al usuario al seleccionar modificar competencia.
CargarNivelrequerido()	Este método carga los niveles requeridos de competencia que se encuentran registrados en la base de datos y los muestra en el Combo Box nivel requerido.
adicionarcompetenciaporestructuraAction()	Este método toma todos los datos de una competencia de la estructura y los adiciona en la base de datos.
modificarcompetenciaporestructuraAction()	Este método modifica los datos de una competencia de la estructura y los guarda en la base de datos.
eliminarcompetenciaporestructuraAction()	Este método elimina todos los datos de una competencia de la estructura colocándole una fecha de eliminación en la base de datos.
ImprimircompetenciasareaAction()	Este método imprime el listado de las competencias del área existentes en la entidad.
ImprimircompetenciascargoAction()	Este método imprime el listado de las competencias del cargo existentes en la entidad.
ImprimircompetenciaspuestodetrabajoAction()	Este método imprime el listado de las competencias del puesto de trabajo existentes en la entidad.

Tabla 17: Descripción de la clase DatProfesiogramaModel

Nombre: DatProfesiogramaModel	
Tipo de clase: Modelo (Business)	
Para cada responsabilidad	
Nombre	Descripción
DatProfesiogramaModel()	Constructor de la clase.
Insertar(obj)	Este método recibe por parámetros un objeto y hace un llamado al método insertar de la clase DatProfesiograma del dominio.
Actualizar(obj)	Este método recibe por parámetros un objeto y hace un llamado al método actualizar de la clase DatProfesiograma del dominio.

Eliminar(idcompetencia)	Este método recibe por parámetros el id de la competencia y hace un llamado al método eliminar de la clase DatProfesiograma del dominio.
GetTodos()	Este método hace un llamado al método GetTodos() de la clase DatProfesiograma del dominio.
GetPorLimite(limite=20, inicio=0)	Este método recibe por parámetros un rango y hace un llamado al método GetPorLimite de la clase DatProfesiograma del dominio.

Tabla 18: Descripción de la clase DatProfesiograma

Nombre: DatProfesiograma	
Tipo de clase: Modelo (Domain)	
Para cada responsabilidad	
Nombre	Descripción
setUp()	Constructor de la clase.
Insertar(obj)	Este método recibe como parámetro un objeto insertándolo en la base de datos.
Actualizar(obj)	Este método recibe como parámetro un objeto actualizándolo en la base de datos.
Eliminar(idcompetencia)	Este método recibe como parámetro un id de competencia y la elimina en la base de datos.
GetTodos()	Este método devuelve todas las competencias de la estructura que existen en la base de datos.
GetPorLimite(limite=20, inicio=0)	Este método devuelve todas las competencias de la estructura de acuerdo a un rango.

3.4. Diagrama entidad relación

Un diagrama o modelo entidad-relación (DER), es una herramienta para el modelado de datos de un sistema de información, mediante el cual se pretende visualizar los objetos que pertenecen a la Base de Datos como entidades, las cuales tienen unos atributos y se vinculan mediante relaciones.

El diagrama del proceso Competencias laborales se realizó basándose en el modelo conceptual previamente confeccionado, quedando de este modo un total de 8 tablas donde 5 son propias del proceso y 3 son relaciones externas. En la Figura 8 se presenta el diagrama obtenido:

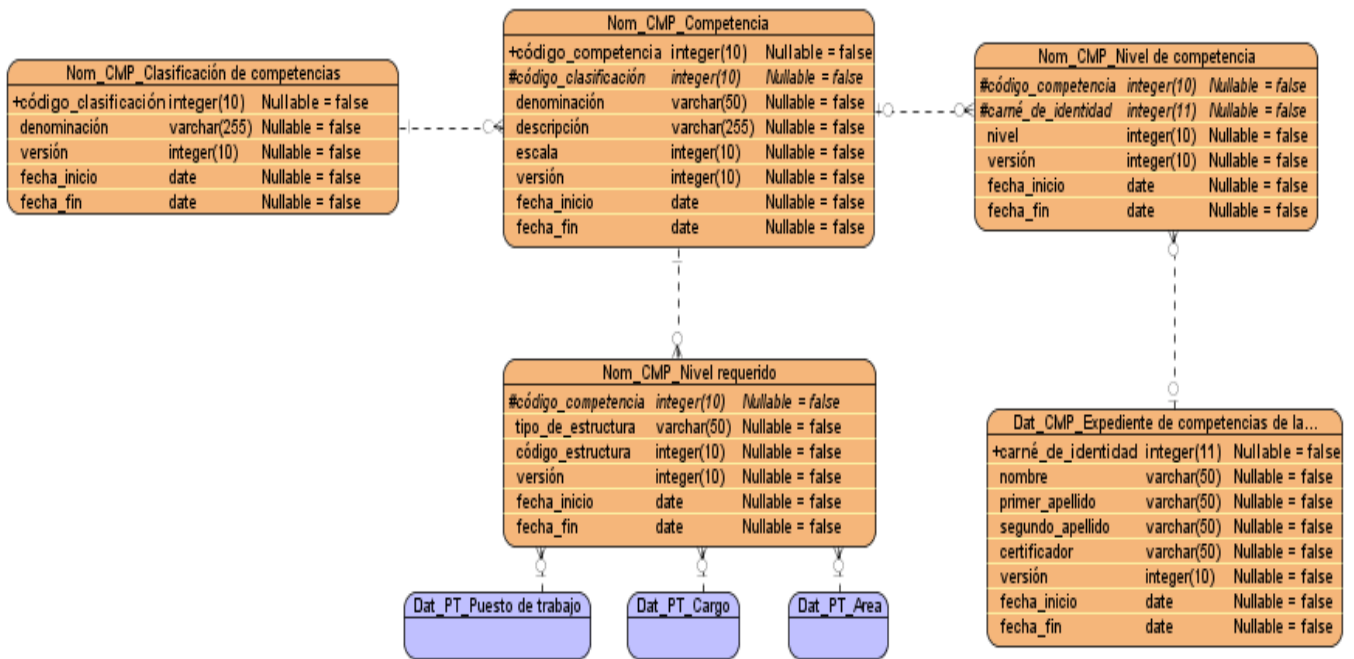


Ilustración 8: Diagrama entidad relación

3.5. Diagrama de componentes

Un diagrama de componentes representa cómo un sistema de software es dividido en componentes, mostrando además las dependencias entre estos. Este tipo de diagrama contiene componentes, interfaces y sus relaciones, puede contener también paquetes que se utilizan para agrupar elementos del modelo. En la figura 9 se presenta el diagrama correspondiente al proceso Competencias laborales:

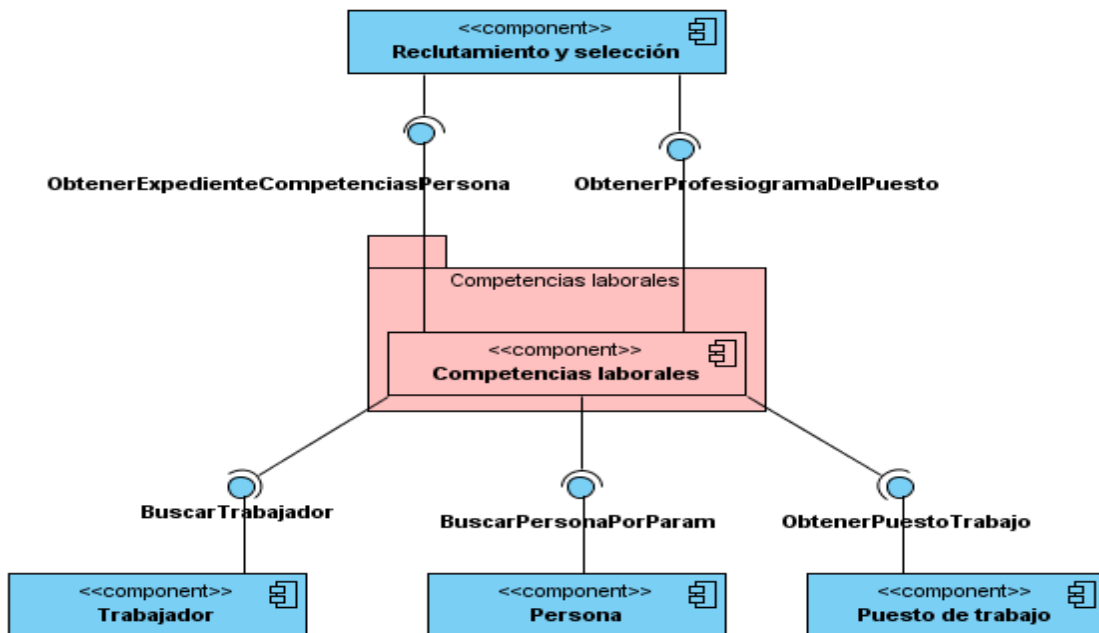


Ilustración 9: Diagrama de componentes

En el diagrama de componentes presentado anteriormente se puede observar la dependencia que existe entre el componente Competencias Laborales y los demás componentes representados, de los cuales se realizará una breve descripción a continuación:

- El componente Competencias laborales tiene relación con el componente “Trabajador” mediante la interfaz **BuscarTrabajador** para buscar los datos del trabajador.
- Se relaciona con el componente “Persona” mediante la interfaz **BuscarPersonaPorParam** para buscar los datos de la persona.
- Se relaciona con el componente “Puesto de trabajo” mediante la interfaz **ObtenerPuestoTrabajo** para obtener el listado de todas las áreas, cargos y puestos de trabajo.
- Se relaciona con el componente “Reclutamiento y Selección” proporcionándole el expediente de competencias para que este a su vez lo utilice para seleccionar a la persona más competente e integrarla a determinado puesto de trabajo, le proporciona además el profesiograma del puesto de trabajo.

3.6. Conclusiones del capítulo

En este capítulo se mostraron los detalles relacionados con el diseño del sistema propuesto, se realizaron los diagramas de clase, el diagrama de componentes y el diagrama entidad relación, realizándose posteriormente una descripción de los mismos. También se definieron los patrones utilizados dándose una breve explicación de cada uno de ellos, mostrándose además de qué forma se aplican en el diseño realizado.

CAPÍTULO 4: VALIDACIÓN DE LOS RESULTADOS

4.1. Introducción

En el presente capítulo se realizará la evaluación de los requisitos capturados, para lo cual se estarán utilizando una serie de técnicas, entre las que se encuentran la de prototipado y la de diseño de casos de prueba, aplicándose posteriormente métricas que permitirán verificar la validez de dichos requisitos. Se validará además el diseño propuesto en el capítulo anterior mediante la aplicación de las métricas Tamaño Operacional de Clase (TOC) y Relación entre Clases (RC).

4.2. Técnicas de validación de requisitos

Para comprobar que los requisitos capturados realmente definen el sistema adecuado es necesario examinarlos, por lo que se realiza su validación mediante una serie de técnicas. Para el presente trabajo se utilizarán las siguientes:

4.2.1. Técnica de Prototipado

Los prototipos de interfaz gráfica se realizan con la intención de que el usuario pueda conocer una versión inicial del sistema que se requiere, de forma que se pueda demostrar y entender mejor cuál será el resultado final, donde el usuario podrá también dar ideas o recomendaciones para que el futuro software cumpla con sus necesidades. Mediante esta técnica es posible corregir los errores que aparezcan estando el software en su fase inicial, evitando de este modo gastos innecesarios y pérdidas de tiempo. Para la construcción del prototipo se utilizó como herramienta el Visual Paradigm 6.4. Para conocer el resultado final ver de los Anexos 18 al 25.

4.2.2. Técnica de Diseño de Casos de Prueba

Los Casos de Prueba se realizan como una forma de probar el sistema, estos incluyen las entradas, las condiciones bajo las que ha de probarse y los resultados esperados. Mediante esta técnica es posible validar y verificar las expectativas de los clientes.

Los requerimientos son esenciales para poder realizar los Casos de Prueba, no obstante también es necesario tener en cuenta otros elementos, entre ellos: riesgos, restricciones, tecnologías, cambios, fallos, etc. Estos Casos de Prueba son conocidos también como pruebas de caja negra, es decir, una prueba que se le realiza al sistema sobre su comportamiento externo. Los datos de prueba se escogen

atendiendo a las especificaciones del problema, sin importar los detalles internos del programa, a fin de verificar que el programa corra bien. La plantilla que se utilizó para realizar los Casos de prueba se encuentra en el Anexo 26 y el Anexo 27.

4.3. Criterios para la verificación de los requisitos

Luego de realizada la validación de los requisitos del proceso Competencias laborales según lo establecido por el proyecto ERP-Cuba, se procede a verificar dichos requisitos, para lo cual se utilizaron los criterios de correctitud, completitud y consistencia.

Correctitud

Esta métrica permite determinar si los requisitos fueron capturados correctamente, de forma que puedan satisfacer las necesidades del negocio y los interesados. Para aplicarla es necesario contar con un registro que contenga las observaciones del negocio realizadas, en el cual se encuentran reflejadas especificidades del negocio como por ejemplo: atributos de un concepto erróneos, omisión de validaciones y restricciones de negocio, omisión de un requisito.

Mediante la validación por el funcional se detectó que en la agrupación de requisitos funcionales Gestionar competencias faltaba ponerle el código a las competencias, lo cual se incluyó en el Registro de observaciones como única observación de negocio realizada.

Para que la correctitud (X) sea aceptable tiene que tomar un valor menor que 0.10, y para hallarla se divide el total de observaciones del negocio realizadas en la validación (D) entre el total de requisitos revisados (T).

$$X = D/T$$

Se encontró solamente una observación del negocio mediante la validación (D) de un total de 21 requisitos funcionales, por lo que:

$$X = 1/21$$

$$X = 0.05$$

El valor obtenido es de $0.05 < 0.10$, por lo que se aceptan las especificaciones de requisitos.

Compleitud

Esta métrica se utiliza para determinar si la especificación de requisitos es completa, para esto se cuenta con un registro de las observaciones de formato (O) realizadas, que son las generadas porque se han omitido secciones que se requieren en la plantilla.

Para que la completitud (X) sea aceptable su valor debe ser mayor que 0.90, y para obtenerlo se le resta 1 a la división entre el total de observaciones de formato (O) y el total de secciones del documento (S):

$$X = 1 - O/S$$

Al obtenerse cero observaciones en el registro de formato para un total de 6 secciones en el documento de especificación de requisitos:

$$X = 1 - 0/6$$

$$X = 1$$

El resultado obtenido es de $1 > 0.90$, por lo que las especificaciones de requisitos realizadas se consideran completas.

Consistencia

Con esta métrica es posible determinar si la especificación de requisitos es consistente, para lo cual se crea un registro de las observaciones de consistencia realizadas que contiene las observaciones realizadas por contradicciones existentes entre los requisitos.

Para que la consistencia (X) sea aceptable debe tomar un valor menor que 0.20 y se obtiene dividiendo el total de observaciones de consistencia entre el total de requisitos revisados:

$$X = C/T$$

Teniéndose 0 observaciones en el registro de consistencia para un total de 21 especificaciones de requisitos revisadas:

$$X = 0/21$$

$$X = 0$$

Se obtuvo un valor de $0 < 0.20$, por lo que las especificaciones de requisitos se consideran consistentes.

De forma general, luego de haber realizado la verificación de la validación de los requisitos se obtuvieron resultados satisfactorios por lo que quedan validados los requerimientos del sistema.

4.4. Métricas orientadas a clases para evaluar el diseño

Las métricas de diseño a nivel de componentes se concentran en las características internas de los componentes del software e incluyen medidas de la cohesión, acoplamiento y complejidad del subsistema. Estas tres medidas pueden ayudar al desarrollador de software a juzgar la calidad de un diseño a nivel de los componentes. (Pressman, 1998)

Para validar el diseño propuesto se utilizaron las siguientes métricas:

4.4.1. Tamaño operacional de clase (TOC)

La métrica tamaño operacional de clase permite medir la calidad de las clases según los atributos Responsabilidad, Complejidad de implementación y Reutilización. De estos atributos de calidad, Responsabilidad y Complejidad de implementación son inversamente proporcionales a la Reutilización, por lo que mientras mayor sea la responsabilidad y complejidad de implementación de una clase, menor será su nivel de reutilización.

Para aplicar esta métrica se tuvieron en cuenta las clases Controladoras y Modelos reflejadas en el diseño de acuerdo a la propuesta planteada, así como la cantidad de procedimientos (métodos) por cada una de ellas, quedando de la siguiente forma:

Tabla 19: Métrica Tamaño Operacional de Clase (TOC)

No	Componente	Clase	Cantidad de procedimientos
1	Competencias laborales	GestionarcompetenciasController	7
2	Competencias laborales	GestionarclasificacioncompetenciasController	5
3	Competencias laborales	ExpedientecompetenciaspersonaController	10
4	Competencias laborales	ProfesiogramaController	13
5	Competencias laborales	NomCompetenciaModel	5
6	Competencias laborales	NomClaificacioncompetenciasModel	5
7	Competencias laborales	DatExpedientecompetenciaspersonaModel	5

8	Competencias laborales	DatProfesiogramaModel	5
---	------------------------	-----------------------	---

Luego de obtener el resultado anterior, es necesario hallar el promedio de procedimientos por clase, quedando para un total de 8 clases, un promedio de 6.88.

Teniendo en cuenta los valores anteriores se procederá a aplicar los criterios de Responsabilidad, Complejidad de implementación y Reutilización de la siguiente manera:

Tabla 20: Rango de valores para la evaluación de la métrica TOC

	Categoría	Criterio
Responsabilidad	Baja	\leq Prom.
	Media	Entre Prom. y 2^* Prom.
	Alta	$> 2^*$ Prom.
Complejidad de implementación	Baja	\leq Prom.
	Media	Entre Prom. y 2^* Prom.
	Alta	$> 2^*$ Prom.
Reutilización	Baja	$> 2^*$ Prom.
	Media	Entre Prom. y 2^* Prom.
	Alta	\leq Prom.

Los resultados obtenidos se muestran a continuación a través gráficos:

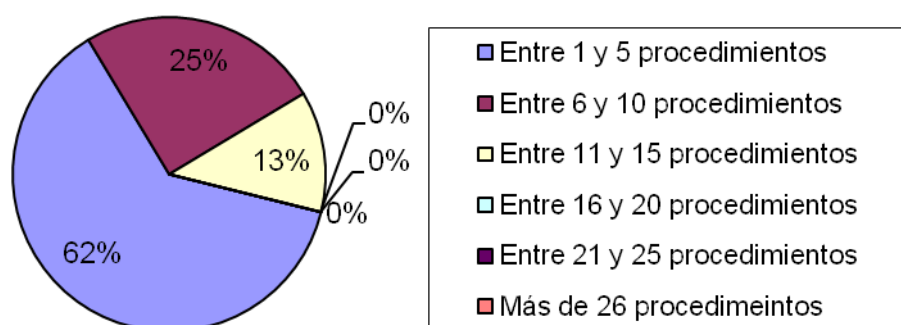


Ilustración 10: Gráfico de los resultados generales de acuerdo a la cantidad de procedimientos.

Obteniéndose como resultado el rango de responsabilidad, complejidad y reutilización de cada una de las clases:

Tabla 21: Resultado de aplicar la métrica TOC

Clase	Responsabilidad	Complejidad	Reutilización
GestionarcompetenciasController	Media	Media	Media
GestionarclasificacioncompetenciasController	Baja	Baja	Alta
ExpedientecompetenciaspersonaController	Media	Media	Media
ProfesiogramaController	Media	Media	Media
NomCompetenciaModel	Baja	Baja	Alta
NomClasificacioncompetenciasModel	Baja	Baja	Alta
DatExpedientecompetenciaspersonaModel	Baja	Baja	Alta
DatProfesiogramaModel	Baja	Baja	Alta

A continuación se mostrarán los gráficos que representan el resultado anterior:

Responsabilidad

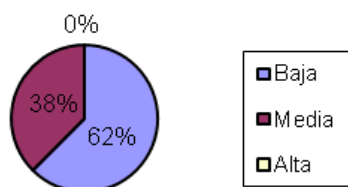


Ilustración 11: Gráfico de la responsabilidad por clases

Complejidad

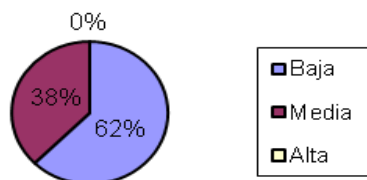


Ilustración 12: Gráfico de la complejidad de implementación por clases

Reutilización

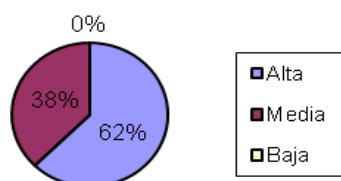


Ilustración 13: Gráfico del nivel de reutilización de las clases

Haciendo un análisis general de lo antes expuesto, se puede observar que todas las clases que conforman el sistema están dentro de la categoría de media y baja, quedando demostrado de esa forma la elevada reutilización, baja complejidad y responsabilidad que presenta el diseño propuesto, por lo que los resultados obtenidos según esta métrica son positivos.

4.4.2. Relaciones entre clases (RC)

Esta métrica está dada por el número de relaciones de uso de una clase con otras. Se le aplica a las mismas clases que le fueron aplicada la métrica TOC, pero en este caso los atributos a medir son: Acoplamiento, Complejidad de mantenimiento, Reutilización y Cantidad de pruebas.

A continuación se presenta la tabla que contiene la información referente a las relaciones de uso de cada una de las clases:

Tabla 22: Métrica Relación entre Clases (RC)

No	Subsistema	Clase	Cantidad de relaciones de uso
1	Capital Humano	GestionarcompetenciasController	0
2	Capital Humano	GestionarclasificacioncompetenciasController	0
3	Capital Humano	ExpedientecompetenciaspersonaController	2
4	Capital Humano	ProfesiogramaController	1
5	Capital Humano	NomCompetenciaModel	1
6	Capital Humano	NomClaificacioncompetenciasModel	0
7	Capital Humano	DatExpedientecompetenciaspersonaModel	2
8	Capital Humano	DatProfesiogramaModel	2

Para esta métrica también es necesario hallar un promedio, pero en este caso, de relaciones de uso por clase, quedando para un total de 8 clases un promedio de valor 1.

Para medir el acoplamiento se utilizan los siguientes valores:

Tabla 23: Rango para medir el acoplamiento en la técnica RC

	Categoría	Criterio
Acoplamiento	Ninguno	0
	Bajo	1
	Medio	2
	Alto	>2

Para el resto de parámetros de calidad es necesario tener en cuenta el promedio hallado anteriormente, los cuales se aplicarán de la siguiente forma:

Tabla 24: Rango de valores para la evaluación de la métrica RC

	Categoría	Criterio
Cantidad de pruebas	Baja	\leq Prom.
	Media	Entre Prom. y $2 \times$ Prom.
	Alta	$> 2 \times$ Prom.
Complejidad de implementación	Baja	\leq Prom.
	Media	Entre Prom. y $2 \times$ Prom.
	Alta	$> 2 \times$ Prom.
Reutilización	Baja	$> 2 \times$ Prom.
	Media	Entre Prom. y $2 \times$ Prom.
	Alta	\leq Prom.

A continuación se muestran los resultados obtenidos luego de haber aplicado los parámetros previamente descritos, tanto en forma de tabla como los gráficos correspondientes.

Tabla 25: Resultado de aplicar la métrica RC

Clase	Acoplamiento	Cantidad de pruebas	Complejidad	Reutilización
GestionarcompetenciasController	Ninguno	Baja	Baja	Alta
GestionarclasificacioncompetenciasController	Ninguno	Baja	Baja	Alta
ExpedientecompetenciaspersonaController	Medio	Media	Media	Media
ProfesiogramaController	Bajo	Baja	Baja	Alta
NomCompetenciaModel	Bajo	Baja	Baja	Alta
NomClaificacioncompetenciasModel	Ninguno	Baja	Baja	Alta
DatExpedientecompetenciaspersonaModel	Medio	Media	Media	Media
DatProfesiogramaModel	Medio	Media	Media	Media

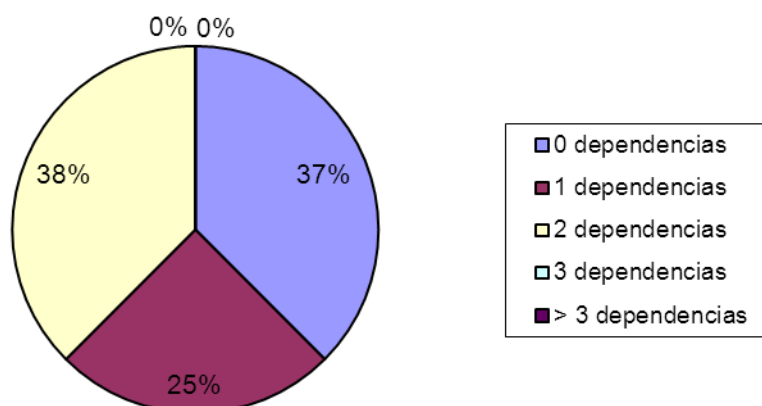


Ilustración 14: Gráfico de los resultados generales de acuerdo a las dependencias.

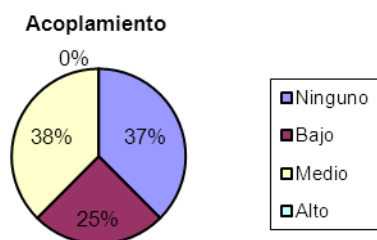


Ilustración 15: Gráfico del acoplamiento

Complejidad de Mantenimiento

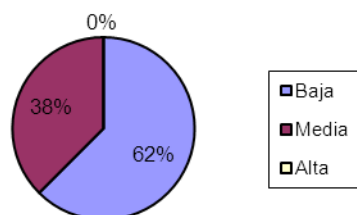


Ilustración 16: Gráfico de la complejidad de mantenimiento de las clases

Cantidad de Pruebas

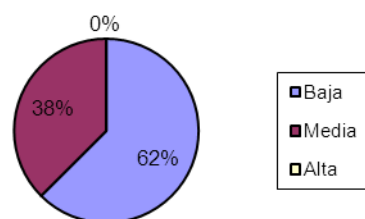


Ilustración 17: Gráfico de la cantidad de pruebas a realizar

Reutilización

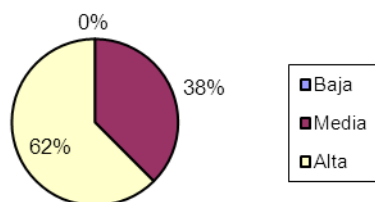


Ilustración 18: Gráfico del nivel de reutilización de las clases

De forma general se puede observar que al aplicar esta métrica los resultados fueron positivos. Existe un bajo acoplamiento entre las clases, ya que el 62% de las clases presenta entre ninguno y bajo acoplamiento, y el 38% de las clases presenta medio acoplamiento. Las clases presentan un elevado nivel de reutilización, ya que el 62% de las clases puede ser reutilizada. De igual manera la cantidad de pruebas y la complejidad de mantenimiento dieron como resultado que el 62% de las clases son fáciles de reparar y la cantidad de pruebas a realizar es corta, lo que demuestra una calidad aceptable del diseño.

4.5. Conclusiones del capítulo

En el presente capítulo se aplicaron una serie de técnicas para validar los requisitos capturados, entre ellas la de prototipos de interfaz y el diseño de casos de prueba, las cuales fueron verificadas posteriormente mediante la aplicación de las métricas correctitud, completitud y consistencia, obteniéndose en cada una de ellas un resultado favorable. Fueron aplicadas además las métricas Tamaño Operacional de Clase (TOC) y Relación entre Clases (RC) para validar el diseño realizado, las cuales arrojaron resultados positivos, quedando demostrada de esta forma la calidad del mismo.

CONCLUSIONES

Luego de haberse realizado el presente trabajo se llegaron a las siguientes conclusiones:

- El estudio del estado del arte permitió una mayor comprensión del proceso de competencias laborales, así como su relación con el subsistema Capital Humano del Sistema Integral de Gestión Cedrux.
- Se logró definir y traducir a requerimientos del sistema las necesidades del cliente.
- Se modelaron los procesos del negocio, además se capturaron, analizaron, especificaron y validaron los requisitos funcionales del sistema.
- La metodología, herramientas y patrones utilizados permitieron que el trabajo se desarrollara con mayor eficiencia.
- Una vez especificados los requisitos, se obtuvo el diseño del proceso Competencias laborales del subsistema Capital Humano del sistema Cedrux, alcanzando así los objetivos trazados.
- Por último tras validar el diseño propuesto se arrojaron resultados positivos con los cuales puede llegarse a la conclusión de que el diseño realizado presenta gran calidad, lo que le permitirá a los desarrolladores mayor facilidad a la hora de implementar el sistema.

RECOMENDACIONES

Luego del trabajo realizado y los resultados obtenidos se recomienda:

- Realizar la implementación del sistema basándose en el diseño propuesto.
- Consultar las referencias bibliográficas utilizadas en el presente trabajo para mayor profundización del tema.
- Realizar la implementación de un sistema que permita llevar a cabo el proceso de certificación de competencias.
- Implementar un sistema que permita realizar la identificación de las competencias de las personas.
- Integrar el proceso de competencias laborales a otros procesos del subsistema Capital Humano como es el caso de Selección e integración.

REFERENCIAS BIBLIOGRÁFICAS

- Baryolo y otros, Oiner Gómez. 2008.** *Arquitectura tecnológica para el desarrollo de software.* 2008.
- Bunk, G. P. 1994.** *La transmisión de las competencias en la formación y perfeccionamiento de profesionales en la RFA, Cedefop.* 1994.
- Cabezas, M. A. 2001.** *Indicadores de Gestión en la Educación Superior como herramienta de la Planificación Estratégica.* 2001.
- Chaves, Michael Arias. 2006.** *La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software.* 2006.
- CINTERFOR/OIT.** Las 40 preguntas más frecuentes sobre Competencias Laborales. [En línea] <http://www.cinterfor.org.uy/public>.
- Cubeiro, J. C. 1998.** *Cómo sacarle fruto a la gestión por competencias.* 1998.
- Cuesta, A. 1999.** *Conformación de los perfiles de Competencias Directivas por el método Delphi.* La Habana : s.n., 1999.
- . **2005.** *Tecnología de Gestión de Recursos Humanos.* 2005.
- Equipo de producción. 2007.** *Modelo de desarrollo orientado a componentes del proyecto ERP-Cuba.* 2007.
- Fidel, Daniellis y. 2009.** *Análisis y diseño del componente Nómina del subsistema Capital Humano del sistema Cedrux.* 2009.
- Fletcher, Shirley. 1997.** *Competencia laboral. Antología de lecturas.* México : s.n., 1997.
- Grande, L. S. 2007.** *Gestión por competencias.* 2007.
- Hernández León y otros, Rolando Alfredo. 2002.** *El paradigma cuantitativo de la investigación científica.* 2002.
- Ibarra, Agustín. 2000.** Formación de Recursos Humanos y competencia laboral. [En línea] 2000. <http://www.cinterfor.org.uy>.
- K. Cruz Muñoz, & Vega Lopez, G. 2001.** *La gestión por competencias: una nueva herramienta en la planificación estratégica del recurso humano.* 2001.
- Marelli, A. 1999.** *Introducción al análisis y desarrollo de modelos de competencia.* 1999.
- Martínez, Ernesto D. Vargas & Yandi Fernández. 2009.** *Definición de un sistema Integral de Gestión de los Recursos Humanos.* 2009.

- Miniet y otros, Yanet Vega. 2009.** *Definición del ciclo de vida de los proyectos de desarrollo.* 2009.
- NC 3000. 2007.** *Sistema de Gestión Integrada de Capital Humano: Vocabulario.* La Habana : s.n., 2007.
- NC 3001. 2007.** *Sistema de Gestión Integrada de Capital Humano: Requisitos.* La Habana : s.n., 2007.
- NC 3002. 2007.** *Sistema de Gestión Integrada de Capital Humano: Implementación.* La Habana : s.n., 2007.
- Nicola, Melbys. 2011.** *Datazucar ofrece soluciones de negocio.* 2011.
- Pressman, Roger. 1998.** *Ingeniería de Software. Un enfoque práctico.* 1998.
- SENAI. 2002.** *Metodología de validación y certificación de competencias.* Brasilia : s.n., 2002.
- Spencer & Spencer, Lyle & Signe M. 1993.** *Competence at work, models superior performance.* 1993.
- Vargas, F. 2004.** La formación basada en competencias en América Latina y el Caribe. Desarrollo reciente. Algunas experiencias. [En línea] 2004.
<http://www.cinterfor.org.uy/public/spanish/region/ampro/cinterfor/temas/complab/observ/vargas/iii>.
- Venegas, C. M. 2004.** Gestión por competencias. [En línea] Mayo de 2004.
<http://www.gestiopolis.com/canales2/gerencia/1/impgercomv.htm>.
- Zúñiga, F. V. 2004.** *40 preguntas sobre competencia laboral.* 2004.

BIBLIOGRAFÍA

- Ana Cristina Zubillaga Rodríguez y otros. 2007.** *Un enfoque de sistemas a las competencias laborales.* 2007.
- D. Zhu, K. Johnson. 2001.** *Requirements Engineering Techniques.* Department of Computer Science. University of Calgary. 2001.
- Dr. Armando Cuesta Santos. 2010.** *Tecnologías de Gestión de Recursos Humanos, Tercera Edición corregida y ampliada.* 2010.
- Fernández, Luis H. 2009.** Patrones de diseño. [En línea] 2009. <http://software.guisho.com/patrones-de-diseño>.
- Figuroa, Armando A. Rosales Leyva y Miguel A. 2009.** *Análisis y diseño del módulo Desarrollo del personal del Sistema de Gestión de Recursos Humanos por competencias en los proyectos productivos de la facultad 3.* 2009.
- García, Elena Cano. 2008.** *La evaluación por competencias en la educación superior.* 2008.
- Giarratana, Matías. 2009.** Modelo de gestión por competencias de los RRHH. [En línea] 2009. <http://winred.com/management/modelo-de-gestion-por-competencias-de-los-rrhh/gmx-niv116-con10980.htm>.
- Gracia, Joaquin. 2005.** Patrones de diseño. Diseño de software orientado a objetos . [En línea] 2005. <http://www.ingenierosoftware.com/analisisydiseno/patrones-diseno.php>.
- 2008.** Herramientas basadas en UML. [En línea] 2008. www.objectsbydesign.com/tools/umltools_byPrice.html.
- 2007.** Ingeniería de software y su relación con las herramientas CASE. [En línea] 2007. http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/rea_c_ji/capitulo_2.html.
- Internacional, V. P. 2006.** *Introducción a los sistemas y herramientas CASE.* 2006.
- Jacobson, Rumbaugh & Booch.** *El proceso unificado de desarrollo de software.*
- José M. Troyas, Lidia Fuentes & Antonio Vallecillo. 2007.** *Metodología orientada a componentes.* 2007.
- Keidy García Lira, Yoena Domínguez & Diana Valdés González. 2008.** *Propuesta de modelado del proceso Ingeniería de requisitos mediante la notación BPMN en los proyectos de informática jurídica.* 2008.

- Larman, Craig.** *UML y Patrones. Una introducción al análisis y diseño orientado a objetos y al proceso unificado.*
- León, Eduardo.** Tutorial Visual Paradigm for UML. [En línea] <http://www.slion2000.blogspot.com>.
- Milán, Ronaldo Castro. 2009.** *Propuesta de procesos para el desarrollo de un Sistema de Emisión de Documentos de Identificación.* 2009.
- Potencier, Fabien y Zaninotto. 2008.** *Symfony, la guía definitiva .* 2008.
- Regalado y Chacón, Yosniel Toste & José A. Martínez. 2008.** *Automatización para ISO y CMMI de la Gestión de Recursos Humanos basados en competencias.* 2008.
- Sanzano, Fidel Jimenez. 2010.** *Sistema de gestión de Capital Humano para el sistema de gestión integral Cedrux.* 2010.
- Sitio oficial de Visual Paradigm. [En línea] <http://www.visual-paradigm.com>.
- Stephen, A. White.** *Introduction to BPMN.* s.l. : IBM Corporation.
- Tedeschi, Nicolás. 2008.** Qué es un patrón de diseño. [En línea] 2008. <http://msdn.microsoft.com/es-es/library/bb972240.aspx>.
- Terreros, Julio Casal.** Desarrollo de software basado en componentes. [En línea] <http://arquitectura-de-software.blogspot.com/2006/05/orm-object-relational-mapping-ii-parte.html>.
- Tobón, Sergio. 2006.** *Las competencias en la educación superior.* Madrid : Universidad Complutense, 2006.

GLOSARIO

Capital Humano: Hace referencia a la riqueza que se puede tener en determinada organización en relación con la cualificación del personal que allí trabaja.

CedruX: Denominación del sistema ERP que se desarrolla para ser usado en las empresas cubanas.

Competencias laborales: Son el conjunto de conocimientos, características y habilidades que posee una persona para ejercer determinada ocupación; deben ser medibles y de estas depende el logro que pueda llegar a alcanzar una organización.

Componente: Es una unidad de composición de aplicaciones software, que posee un conjunto de interfaces y un conjunto de requisitos, y que ha de poder ser desarrollado, adquirido, incorporado al sistema y compuesto con otros componentes de forma independiente, en tiempo y espacio.

Diagrama: Un diagrama es un gráfico que presenta en forma esquematizada información relativa a algún tipo de ámbito.

ERP: Sistema integral de gestión empresarial que está diseñado para modelar y automatizar la mayoría de procesos en la empresa.

Gestión por competencias: Herramienta estratégica indispensable para enfrentar los nuevos desafíos que impone el medio. Es impulsar a nivel de excelencia las competencias individuales, de acuerdo a las necesidades operativas.

Métrica: Es un instrumento que cuantifica un criterio y persigue comprender mejor la calidad del producto, estimar la efectividad del proceso y mejorar la calidad del trabajo realizado.

Proceso: Conjunto de actividades o eventos que se realizan o suceden bajo ciertas circunstancias con un fin determinado.

Requisito: Condición o capacidad que debe cumplir un sistema.

Sistema: Conjunto de elementos que se relación entre sí con el propósito de lograr un objetivo.

Subsistema: Cada una de las partes o módulos que forman un sistema. Cada subsistema abarca aspectos del sistema que comparten alguna propiedad común.