

The logo for Universidad de las Ciencias Informáticas (UCi) features the letters 'UCi' in a bold, sans-serif font. A thin, curved line arches over the letters, and a small square is positioned above the 'i'.

Universidad de las Ciencias
Informáticas

Facultad 1

**Solución informática para la
gestión de personal en el núcleo
del Sistema de Gestión
Universitaria**

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

**Autores: Lisbethy Rodríguez Carrazana
Carlos Luis Martínez Leyva**

**Tutores: Ing. Ana María Sánchez González
Ing. Norges Sánchez Tumbarell
Lic. Alejandro Anasagasti Cámara**

La Habana, Junio 2012



La adquisición de cualquier conocimiento es siempre útil al intelecto, que sabrá descartar lo malo y conservar lo bueno.

Leonardo Da Vinci

Declaración de autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo. Autorizamos a dicho centro para que haga el uso que estime pertinente con este trabajo.

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

Lisbethy-Rodríguez-Carrazana
Firma-del-Autor

Carlos-Luis-Martínez-Leyva
Firma-del-Autor

Ing.-Ana-María-Sánchez-González
Firma-del-Tutor

Ing.-Norges-Sánchez-Tumbarell
Firma-del-Tutor

Lic.-Alejandro-Anasagasti-Cámbara
Firma-del-Tutor

Datos de contacto

Lisbethy Rodríguez Carrazana

Correo: lcarrazana@estudiantes.uci.cu

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Carlos Luis Martínez Leyva

Correo: clmartínez@estudiantes.uci.cu

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Ing. Norges Sánchez Tumbarell

Correo: nsanchez@uci.cu

Graduado en la Universidad de las Ciencias Informáticas en la graduación del curso correspondiente a los años 2006-2007. Durante su trayectoria como profesor ha impartido clases de las asignaturas Matemática III y Matemática IV. Pertenece al Centro de Informatización Universitaria (CENIA) donde se desempeña como Líder de Proyecto de Pregrado y posee la categoría docente de Instructor.

Ing. Ana María Sánchez González

Correo: amsanchez@uci.cu

Graduada en la Universidad de las Ciencias Informáticas en la graduación del curso correspondiente a los años 2007-2008. Durante su trayectoria como profesor ha impartido clases de las asignaturas Tele Informática, Seguridad Informática, Comercio electrónico, Ingeniería de software y Práctica profesional. Pertenece a la Facultad 1, dpto. ISW y PP y está prestando servicios en el Centro de Informatización Universitaria (CENIA) donde se desempeña como analista de Pregrado y posee la categoría docente de Instructor.

Lic. Alejandro Anasagasti Cámara

Correo: anasagasti@uci.cu

Graduado en el Instituto Superior Pedagógico “Enrique J. Varona” en la graduación correspondiente a los años 2008-2009. Pertenece al Centro de Informatización Universitaria (CENIA) donde se desempeña como desarrollador en el Proyecto de Pregrado.

Dedicatoria

Le dedico este Trabajo de Diploma a mi familia, ustedes con sus ejemplos me han formado. Porque en todo momento han confiado en mi y me han dejado tomar mis propias decisiones.

A mi madre, eres la mejor mujer que he conocido en esta vida y espero que algún día mis hijos puedan sentir por mí toda la admiración, respeto, amor y confianza que tengo en ti, que a pesar de muchas dificultades siempre encontraste una manera de ayudarme.

A mi abuela, "Mima" gracias por ser esa persona especial que junto a mi madre ha estado ahí, pendiente de mis problemas sabes toda la admiración que siento por ti, que no se olviden que donde yo este siempre estaré orgulloso de tener una madre y abuela como ustedes.

Carlos Luis Martínez Leyva

Dedico mi Trabajo de Diploma a mi tata lindo, para que le sirva de guía en el largo camino que le queda por andar.

A mi abuelo Carrazana, que de él aprendí a superar las cosas y a enfrentarme a la vida sin miedo.

A mi abuela Deisy por aguantar mis resabios, por brindarme su paz y ecuanimidad. A los dos por quererme tanto, educarme y criarme como uno de sus hijos. Hasta el punto de sentir y saber que soy para ustedes "su niña".

A mi abuela Orfelina por tenerme presente en cada momento.

A mi abuelo Juanito por brindarme su cariño y amor.

A los dos por hacerme sentir que soy uno de sus nietos preferidos.

A mis padres, que siempre están celosos para saber a quién quiero más. ¡Mami, papi, esta dedicatoria va por igual! Los quiero mucho a los dos y a cada uno los llevo siempre en mi corazón, tomándolos como ejemplo a cada paso que voy dando en mi vida.

Lisbethy Rodríguez Carrazana

Agradecimientos

Carlos Luis Martínez Leyva

Agradezco a todas esas personas que me ayudaron a tener esta oportunidad de graduarme como ingeniero. A mi familia en general por darme tanto apoyo y formarme como persona, cada uno de ellos han aportado en mí valores importantes.

A mi mamá y abuela que son ese ejemplo a seguir por su perseverancia, por entenderme y apoyarme en momentos difíciles.

A mi padre, del cual siento un gran orgullo cuando me comparan contigo por lo trabajador que eres y por darme esa segunda familia: mi madre Odalis, mis hermanos Helen y Erick, que son especiales, nunca me olvido de ustedes.

A Humberto, tu me criaste y gracias a ti llegué tan lejos, por inculcar en mi hábitos de estudio y de superación.

A mi tío Graciél que es un ejemplo a seguir por todos los valores que tiene.

A mi hermano Gianni, sabes que te quiero y apoyo.

A Rayko, Amílcar, Alberto y Adrian ustedes más que amigos son mi hermanos, tantos años de amistad han significado mucho para mi.

A mis chicas especiales Daimarys, Eli, Rosy y Rosali, ustedes han sabido ocupar un pedacito de mi corazón. Nunca olvidare el tiempo que pasamos juntos y las historias de todo lo que hicimos y quedan por hacer serán recordadas. Ustedes son una parte importante de mí. Para ustedes ¡Aquí estaré siempre!

A todos esas personas que he conocido durante estos 5 años en la universidad, que compartimos un aula, una piscina, una fiesta. Muchos están otros no pero nunca los olvidare.

A mi compañera de tesis por tener esa paciencia conmigo.

A mis profes y tutores: Norges, Alejandro, Ana María, Yasmani por su ayuda durante la realización de este trabajo ustedes son especiales.

Lisbethy Rodríguez Carrazana

En este momento tan especial de mi vida quisiera agradecer a todas aquellas personas que de una forma u otra me ayudaron para que este sueño se hiciera realidad.

Agradezco a mi familia por ser mi fuente de inspiración.

A mi hermano, por seguir mi ejemplo y apoyarme en todo, gracias mi niño lindo.

A mis abuelos, por ser los mejores abuelos del mundo. ¡Los adoro!

A mi mamá que me ha demostrado que siempre se puede, a pesar de los obstáculos.

A mi padre, por ser el padre más guapo del mundo, por aprenderse siempre el nombre de mis amistades y por ser “mi chicharrón”.

A mis tíos y primos, en especial a Tita y Mary.

A Arlen y su familia, por tratarme como un integrante más. Niño te quiero mucho.

A Lucy, por tratarme siempre como su hermana y a su linda familia.

A mi amigo Sandy, por su fortaleza de aguantarme desde pre-escolar. Flaco tu si que sabes encantar a las personas. ¡Aquí tienes una amiga para toda la vida!

A mis amigas del politécnico María Teresa y Lisette María.

Gracias a los amigos que nunca pensé conocer. Esos que conocí al entrar a la universidad.

A mis tres bichos, “Amaurys, Tony y Yoshi”. Espero que nuestra amistad perdure.

A Yaima, Yisel, Greisy, Karelia, gracias chicas por su apoyo y cariño.

A mis amigos del aula y del proyecto que siempre me apoyaron: Yosvany, Valladares,

Adrián, Dayamí, José Miguel, Enrique, Claudia, Iván, Dianela, Rodaín, a todos.

A los muchachos tan guapos que conocí en 4to año, ojalá los hubiera conocido el primer día de clases. Reinaldo, Odrey, Rubiel, Israel, gracias por apoyarme tanto. ¡Los quiero mucho! Tata te quiero como a mi hermano.

A mis profes en especial a Ailec, Ana Maria, Aylín y Norges.

A mi compañero de tesis por hacerme llorar tanto.

Carlos y Lisbethy

Le agradecemos a la Revolución Cubana y en especial a Fidel Castro por darnos la oportunidad de cumplir nuestros sueños de hacernos ingenieros.

A nuestros tutores por su paciencia y dedicación.

A nuestros compañeros de proyecto por estar ahí cuando lo necesitamos.

A nuestros familiares por ser el motor impulsor en nuestras vidas.

A todas las personas que de una forma u otra nos ayudaron.

Resumen

Informatizar los procesos universitarios es un reto para todas las universidades. Dentro de cada proceso existente, se encuentran vinculadas las personas, por lo que, es imprescindible que se maneje su información. En la Universidad de las Ciencias Informáticas se están desarrollando sistemas para alcanzar este objetivo. Uno de estos productos es el Sistema de Gestión Universitaria, perteneciente al Centro de Informatización Universitaria. Hasta el momento, esta gestión de las personas no cumple con las expectativas del centro. El objetivo de la presente investigación fue desarrollar el módulo de Personal en el núcleo del Sistema de Gestión Universitaria, para mejorar los procesos asociados a la información de las personas, facilitar su entendimiento y organización. Con la utilización del módulo propuesto se administra y configura toda la información de las personas, estando organizadas en varias categorías. Con la propuesta planteada se resuelve el problema de obtención de datos, al incluir la configuración de los tipos de fuentes de información, tales como: base de datos, directorio activo, archivos en formato Excel y CSV. La solución propuesta constituye una aplicación Web implementada con herramientas, tecnologías y lenguajes libres tales como: Visual Paradigm, NetBeans, PostgreSQL, UML, entre otras. Para guiar el desarrollo del producto se utilizó un proceso de desarrollo con enfoque ágil orientado a la calidad del producto.

Palabras clave: gestión de personal, gestión de RR.HH. gestión universitaria, universidad.

Abstract

Computerize the university processes is a challenge for all universities. Within each existing process, people are linked, so it is essential that your information is handled. At the University of Informatics Sciences are developing systems to achieve this goal. One such product is the University Management System, part of the University Center of Informatization. So far, the management of people does not meet the expectations of the center. The objective of this research was to develop the module in the core staff of University Management System to improve the processes associated with information from people, facilitate their understanding and organization. By using the proposed module manages and configures all the information of people, being organized in several categories. The proposal addresses the issue raised is data collection, to include the configuration of the types of information sources, such as database, Active Directory, Excel files and CSV format. The proposed solution is a web application implemented with tools, technologies and free languages such as Visual Paradigm, NetBeans, PostgreSQL, UML, among others. To guide the product development process was used agile development approach focused on product quality.

Keywords: personnel management, HR management university management, university.

Índice de contenido

Introducción	1
Capítulo 1 Fundamentación teórica.....	6
1.1 Introducción.....	6
1.2 Breve descripción del objeto de estudio	6
1.2.1 Recursos Humanos.....	6
1.2.2 Gestión de Recursos Humanos (GRH).....	7
1.2.3 Gestión de personal en los Recursos Humanos.....	8
1.3 Tendencias en el uso de sistemas de gestión de personal.....	9
1.3.1 Sistemas de gestión de personal en el mundo	9
1.3.2 Sistemas de gestión de personal en Cuba	13
1.3.3 Sistemas de gestión de personal en la Universidad de las Ciencias Informáticas	13
1.3.4 Valoración de los sistemas existentes	16
1.4 Entorno de desarrollo	16
1.4.1 Lenguajes de desarrollo	16
1.4.2 Tecnologías de desarrollo	19
1.4.3 Herramientas de desarrollo	22
1.5 Proceso de desarrollo de <i>software</i>	24
1.5.1 Proceso de desarrollo de <i>software</i> con enfoque ágil basado en CMMI nivel 2	24
1.6 Conclusiones parciales.....	25
Capítulo 2 Concepción de la propuesta de solución.....	26
2.1 Introducción.....	26
2.2 Modelo de dominio	26
2.3 Descripción del módulo de Personal	28
2.3.1 Características del módulo de Personal	29
2.3.2 Usuarios relacionados con el módulo	30
2.3.3 Integración del módulo al Sistema de Gestión Universitaria	30
2.4 Requisitos	31
2.4.1 Captura de requisitos	31

2.4.2	Especificación de requisitos	34
2.5	Estilos y patrones arquitectónicos	37
2.6	Patrones de diseño	39
2.6.1	Patrones GRASP	39
2.6.2	Patrones GOF	40
2.7	Estándares de diseño.....	41
2.7.1	Pauta cromática	41
2.7.2	Componentes.....	42
2.7.3	Mensajes.....	42
2.8	Modelo de despliegue	43
2.9	Modelo de datos.....	45
2.10	Conclusiones parciales.....	46
Capítulo 3 Implementación y validación de la solución informática		47
3.1	Introducción.....	47
3.2	Estándares de codificación.....	47
3.2.1	Identación, llaves de apertura y cierre, tamaño de líneas	47
3.2.2	Conversión de nomenclaturas	47
3.2.3	Estructuras de control	48
3.3	Técnicas de programación	48
3.3.1	Programación modular	48
3.3.2	Programación orientada a objetos (POO).....	48
3.4	Tratamiento de errores.....	49
3.5	Validación de la solución propuesta	49
3.5.1	Validación de requisitos.....	50
3.5.2	Estrategia de prueba	52
3.6	Conclusiones parciales.....	58
Conclusiones		59
Recomendaciones		60
Bibliografía referenciada		61
Bibliografía consultada.....		67

Anexos.....	69
Anexo 1 Modelo de entrevista.....	69
Anexo 2 Requisitos del cliente	69
Anexo 3 Requisitos del producto.....	70
Anexo 4 Especificación de requisitos funcionales	70
Anexo 5 Especificación de requisitos no funcionales	77
Anexo 7 Modelo de datos	80
Anexo 8 Diseños de caso de prueba basado en requisitos	83
Anexo 9 Casos de prueba de integración	97

Índice de tablas

Tabla 1 Definición de conceptos para procesos de gestión de personal.....	27
Tabla 2 Información sensible de los subsistemas del Sistema de Gestión Universitaria.....	29
Tabla 3 Listado de requisitos funcionales.....	32
Tabla 4 Requisitos funcionales.....	33
Tabla 5 Requisitos no funcionales.....	34
Tabla 6 Especificación de requisito: RFP_1_Buscar Persona.....	35
Tabla 7 Nodos y conexiones del modelo de despliegue.....	43
Tabla 8 Criterios para validar requisitos del cliente.....	50
Tabla 9 Criterios para validar requisitos del producto.....	51
Tabla 10 DCP_Buscar Personas. Parte 1.....	53
Tabla 11 DCP_Buscar personas. Parte 2.....	54
Tabla 12 No conformidades encontradas por iteraciones.....	56
Tabla 13 Caso de prueba de integración Int_1.....	57
Tabla 14 Caso de prueba de integración Int_14.....	57
Tabla 15 Caso de prueba de integración Int_15.....	58

Índice de figuras

Figura 1 Áreas de procesos que intervienen en el nivel 2 de CMMI	25
Figura 2 Modelo de dominio.....	28
Figura 3 MVC en GUUD.....	39
Figura 4 Pauta cromática	41
Figura 5 Modelo de despliegue para el módulo de Personal.....	44

Introducción

Desde la antigüedad se han buscado diferentes procedimientos para recopilar la información, haciéndola prevalecer en el tiempo para utilizarla. Dentro de la información que se ha necesitado salvaguardar está la de las personas, debido a que cada individuo posee una gran cantidad de características que son fundamentales a la hora de realizar procesos en los cuales estén vinculados.

La información de las personas tiene más énfasis en la esfera de los Recursos Humanos (RR.HH.), puesto que se centra en definir, desarrollar procesos encaminados a obtener y coordinar a las personas de forma coherente para la consecución de los objetivos de una organización. Los Recursos Humanos atienden las relaciones entre personas que forman parte de la plantilla de la empresa y la propia organización, para llegar al mutuo beneficio entre ambas partes. No obstante, este tipo de información se maneja en cualquier ámbito de la sociedad desde un censo de población hasta en una institución educacional.

Con el constante desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC), se ha encontrado una solución para recopilar la información de manera digital y que esta sea gestionada con mayor facilidad. Cuba ha evolucionado en este tema y junto con el avance de las TIC se han creado nuevos proyectos para el beneficio de la sociedad; propiciando así la creación de la Universidad de las Ciencias Informáticas (UCI) en el año 2002.

La UCI se considera una Universidad Productiva. Término que fue usado en Cuba para caracterizar a una institución educacional capaz de relacionar el conocimiento universitario con la producción para la sociedad, incluida la ampliación de la capacidad universitaria de captar recursos a partir de sus conocimientos(GONZÁLEZ, et all, 2007).

Como Universidad Productiva está compuesta por centros que son los encargados de desarrollar los productos que se comercializan tanto en Cuba como en otros países.

El Centro de Informatización Universitaria (CENIA) como centro productivo de la UCI tiene varios departamentos encaminados a la gestión de la información de las diferentes líneas de desarrollo que contiene la universidad. Uno de los departamentos es el de Gestión Universitaria en el que se propone informatizar¹ todas las áreas de procesos que se realizan en la universidad, dentro de las que se encuentran: Pregrado, Postgrado, Producción, Investigación, Ingreso, Ubicación Laboral, Laboratorio, Residencia, Extensión Universitaria, Cooperación Internacional, Biblioteca y Formación.

Todas las áreas de procesos mencionadas anteriormente trabajan con la información de las personas que se encuentran en la universidad, por lo que, están estrechamente vinculadas y se hace necesario

¹ *Informatizar: proceso de aplicar sistemas o equipos informáticos al tratamiento de la información.*

que todas estén en una perfecta sincronización en un núcleo para que no existan problemas a la hora de realizar la Gestión Universitaria.

Se ha detectado que, hasta el momento, en el sistema que se está desarrollando para el CENIA, la administración del personal se realiza con dificultades por los elementos que se describen a continuación:

Existe insuficiente validación de la información de las personas, debido a que existen problemas con la base de datos del núcleo, principalmente en la tabla "Persona" donde los atributos que identifican a un individuo están propensos a duplicarse, puesto que, a la hora de insertar los datos no cuentan con la nomenclatura requerida. Donde la estrategia que existe para crear el campo identificador de la persona es la de unir el número del carné de identidad más el número de serie del mismo; esto provoca que en ocasiones cuando se inserta el número de serie con otra estructura pueda duplicarse una información en la tabla.

Otro problema que se detectó es que no se pueden introducir datos de personas que no sean cubanas, ya que las mismas no tienen carné de identidad, provincia, ni municipio. Además no todas las personas cubanas poseen la misma identificación, existiendo así varias longitudes para el tamaño de la identificación y varios tipos de identificaciones.;

Se encontró que no se tienen en cuenta las insuficiencias en las especificaciones de cada área de procesos. Como por ejemplo: Pregrado brinda la dirección de los estudiantes, siendo estos los únicos que pueden tener este dato y Cooperación necesita para realizar los trámites de pasaportes la dirección de las personas, donde no todas las personas que se tramitan son estudiantes.

Además, no se tiene claro el procedimiento para cambiar la información de las personas en el tiempo, porque cada área luego de recoger los datos de las personas los modifica sin que el núcleo se entere. Por lo anteriormente planteado se define como **problema de la investigación**: ¿Cómo mejorar la administración y configuración de los procesos de personal en el negocio universitario para facilitar su entendimiento y organización?

El **objeto de estudio** de la presente investigación lo constituyen los procesos de gestión de los Recursos Humanos y como **objetivo general** se propone: desarrollar el módulo Personal en el núcleo del Sistema de Gestión Universitaria, para mejorar los procesos asociados a la información de las personas, facilitar su entendimiento y organización.

Para dar cumplimiento al objetivo general se proponen los siguientes objetivos específicos:

- Fundamentar los elementos teóricos de los procesos de gestión de personal.
- Definir el análisis y diseño de la solución propuesta.
- Implementar el módulo Personal.

- Validar las funcionalidades generadas en la construcción de la propuesta de solución.

El **campo de acción** está centrado en los procesos de gestión de personal.

La investigación se sustenta en la siguiente **idea a defender**: si se logra administrar y configurar los procesos de personal mediante la creación de un módulo que agrupe varias funcionalidades como la de: gestionar la información de las personas, configurar los datos y configurar las fuentes de información; se contribuirá a que los procesos queden organizados para un mayor entendimiento.

Con el propósito de guiar y perfilar el trabajo hacia el alcance de los objetivos trazados, se definieron las siguientes **tareas de investigación**:

1. Determinación de los conceptos fundamentales asociados a los procesos de la gestión de personal en los Recursos Humanos.
2. Caracterización de las tendencias de los sistemas para la gestión de personal en lo internacional, nacional e institucional.
3. Estudio de las soluciones informáticas existentes para la administración de la información de las personas.
4. Estudio del entorno de desarrollo a utilizar.
5. Estudio del proceso de desarrollo con enfoque ágil a utilizar.
6. Definición de los patrones de diseños a utilizar en la propuesta de solución.
7. Identificación de los requisitos funcionales y no funcionales necesarios para desarrollar el análisis y diseño.
8. Diseño de los prototipos de interfaz.
9. Diseño del esquema de la base de datos del sistema.
10. Definición de los principales elementos de la arquitectura de *software*.
11. Implementación de las funcionalidades identificadas.
12. Validación de la solución propuesta.
13. Diseño de los casos de prueba basados en los requisitos de *software*.

Para la realización y culminación satisfactoria de la presente investigación que está dirigida a la creación de nuevos conocimientos, se hace necesaria la utilización de los métodos científicos.

Dentro de los **métodos teóricos** a utilizar está el **histórico-lógico**, se utiliza este método en el análisis de la base teórica para conocer los antecedentes sobre la utilización de los procesos de gestión de personal. Se utiliza en el estudio de las tecnologías, procesos de desarrollo a emplear y en la descripción de las soluciones informáticas existentes.

Otro método a utilizar es el **analítico-sintético**, el análisis permite el desglose material o mental del objeto a investigar, con el propósito de hallar los elementos fundamentales que lo componen. La síntesis permite la integración tanto material o mental de los elementos y relaciones esenciales del objeto, con el objetivo de determinar cualidades y rasgos específicos. Este método posibilita la búsqueda bibliográfica de los elementos fundamentales de la investigación.

Por otro lado, la **modelación** permite realizar los modelos asociados al proceso de desarrollo seleccionado.

De los **métodos empíricos** se utiliza la **entrevista**: en la investigación se utiliza para precisar el problema a resolver, así como las necesidades existentes en la gestión de personal.

Justificación de la investigación

Dado los antecedentes y el planteamiento del problema es válido preguntarse: ¿por qué y para qué esta investigación? Esta investigación surge por la necesidad de erradicar los problemas existentes con la administración de las personas en la gestión universitaria. Con el desarrollo de la misma se podrá obtener como aportes que: se homogenizará el proceso de gestión de personal para todas las áreas de los procesos que componen el Sistema de Gestión Universitaria, teniendo en cuenta las características de cada proyecto. Se reducirá el riesgo de ocurrencia de réplicas de personas. Además, se obtendrá como aporte práctico: la recopilación bibliográfica sobre los elementos teóricos asociados a la gestión de personal, la documentación asociada al proceso de desarrollo de la propuesta y un módulo que contenga los elementos necesarios para configurar y entender los procesos asociados a la información de las personas.

Estructuración del trabajo por capítulos

Para lograr una mayor organización en la investigación, el presente trabajo estará estructurado por tres capítulos que abarcarán todo el proceso para el desarrollo de la solución informática, adicionalmente, contará con conclusiones, recomendaciones, bibliografías y anexos.

En el **Capítulo 1 Fundamentación teórica**: se presentan los elementos teóricos que sirven de base a la investigación del problema planteado, en este capítulo se analizan los principales conceptos relacionados con el objeto de estudio. Realizando así una descripción de los sistemas existentes y una explicación de herramientas y proceso de desarrollo a utilizar en la construcción de la propuesta de solución.

En el **Capítulo 2 Concepción de la propuesta de solución**: se especifican los principales elementos que caracterizan la propuesta de solución, se explica el modelo de dominio asociado a la gestión de la información de las personas en el negocio universitario. Se refleja cómo se obtienen y especifican los

requisitos, los prototipos de interfaz y estándares de diseño a utilizar. Se proponen los patrones de diseño y el patrón de arquitectura a emplear en el desarrollo de la propuesta. Además, se presenta el modelo de datos de la propuesta.

Capítulo 3 Implementación y validación de la solución informática: Se presentan las técnicas de programación utilizadas, los estándares de codificación. Además, se especificará la validación de la solución propuesta, así como las técnicas y estrategias de prueba seleccionadas.

Capítulo 1 Fundamentación teórica

1.1 Introducción

Para realizar una sólida investigación es necesario tener bien formulado el marco teórico, donde todos los conceptos, sistemas existentes y herramientas estén en estrecha relación con lo que se desea investigar. En el presente capítulo se abordan temas relacionados con el objeto de estudio de la investigación y el campo de acción de la misma. Se analizan aspectos referentes a las principales tendencias en la informatización de los procesos de gestión de la información de las personas en el área de los Recursos Humanos (RR.HH.) mundialmente, nacional e institucional.

1.2 Breve descripción del objeto de estudio

A finales del siglo XIX y a principios del siglo XX con el desarrollo industrial que hubo en el mundo se hizo necesario tener un control de las personas, los líderes comenzaron a lograr metas mediante el esfuerzo de grupos organizados y tuvieron que realizar funciones de personal básicas, dando paso así a la gestión de los RR.HH. En la presente investigación se toma como objeto de estudio los procesos de gestión de los Recursos Humanos. Para poder comprender como funciona esta esfera se debe tener presente los principales conceptos que tiene vinculado. Además, de conocer cómo influye la gestión de personal dentro de la administración de los recursos.

1.2.1 Recursos Humanos

El término de Recursos Humanos es usado para describir a los individuos que componen la fuerza de trabajo de una organización. Esta área es la encargada de establecer políticas y estrategias relacionadas con la gestión de las personas. Se encarga de obtener y coordinar a las personas dentro de un organismo, de forma que consigan las metas establecidas. Cada individuo que forma parte de los Recursos Humanos se caracteriza por desempeñar una variada lista de tareas específicas a cada sector(CHIAVENATO, 1999).

En la Administración de Empresas², se denomina RR.HH. al trabajo que aporta el conjunto de los empleados o colaboradores de esa organización. No obstante, es frecuente llamar así a la acción que se ocupa de seleccionar, contratar, formar, emplear y retener a los colaboradores. Además, lo ve como una disciplina que estudia todo lo atinente a la actuación de las personas en el marco de una organización (ALLES, 2011).

Dentro de las principales funciones que tienen los RRHH se encuentran:

- Empleo: proporciona a la empresa en todo momento el personal necesario, tanto cuantitativo

²Administración de Empresas: Ciencia social que estudia a la organización, (empresas) y la manera como se gestionan los recursos, procesos y resultados de sus actividades.

como cualitativo, para desarrollar óptimamente los procesos de producción con un criterio de rentabilidad económica. Pueden ser aditivos (adicionar personal a la empresa) o sustractivos (disminuir personal de la organización).

- Gestión de personal: encargada del manejo burocrático desde que una persona ingresa en la empresa hasta que la abandona.
- Retribución: se trata en conseguir buscar una estructura de salarios que cumpla 3 condiciones: Motivador, Internamente Equitativo y Externamente competitivo.
- Dirección y desarrollo de RR.HH.: se refiere a la necesidad de que los individuos crezcan dentro de la organización.
- Relaciones laborales: tiene que ver con el tratamiento de conflictos entre los recursos que se manejan.
- Servicios sociales: consiste en el establecimiento de unas medidas voluntarias por parte de la empresa para la mejora del clima laboral.

El recurso humano es el único recurso vivo y dinámico de la organización y decide el manejo de los demás, que son físicos o materiales. Además, constituyen un tipo de recurso que posee una vocación encaminada al crecimiento y desarrollo(CHIAVENATO, 1999).

1.2.2 Gestión de Recursos Humanos (GRH)

La gestión no es más que el proceso mediante el cual se obtiene, despliega o utiliza una variedad de recursos básicos para apoyar los objetivos de la organización. Donde las principales funciones son las de planificar, controlar, organizar y dirigir(DÍAZ, 2005).

Con el progresivo avance de la implantación de los Recursos Humanos en las organizaciones la necesidad de gestionarlos es inminente, por lo que, se han ido desarrollando conceptos que abarcan la gestión de los RRHH, estos conceptos van encaminados a un mismo objetivo, logrando así una mayor eficiencia a la hora de gestionar a los recursos vivos dentro de la empresa.

Dentro de los conceptos desarrollados de GRH se pueden tomar los siguientes:

La GRH es la organización y trato de las personas en el trabajo, de manera que desarrollen plenamente sus capacidades. Donde los mismos contribuyan al logro de los objetivos de la empresa y al mismo tiempo obtengan, mediante la actividad que ejecuta, su propia realización como seres humanos (BARQUERO, et all, 2005).

La GRH tiene como esencia reunir diferentes tipos de personas para lograr un propósito común. Desempeñan una función importante cuando se trata de obtener el máximo de los empleados y proporcionarles un ambiente laboral que satisfaga sus necesidades a corto y largo plazo (BOHLANDER, 2008).

Según Juan Carlos Rodríguez³, en su artículo "*El modelo de Gestión de Recursos Humanos*", define el término de GRH como: el conjunto de actividades, técnicas e instrumentos que se desarrollan o aplican en una empresa; cuyo objetivo consiste en situar a la persona idónea en el puesto adecuado en el momento oportuno, formada y motivada para contribuir eficazmente a la consecución de los objetivos de la organización. Además, plantea que debe considerarse como GRH a las actividades, técnicas e instrumentos cuyo objetivo es facilitar una salida ordenada de la organización de aquellas personas que no son necesarias para la obtención de los resultados de negocio (RODRIGUEZ, 2004).

Luego de haber presentado los diferentes conceptos, en la investigación se asume que en una empresa la GRH se encarga de obtener y coordinar a las personas de una organización, de forma que logren o alcancen las metas establecidas. En esta gestión los hombres son un recurso que es necesario desarrollar, considerándose así el hombre como primer recurso estratégico de la organización.

1.2.3 Gestión de personal en los Recursos Humanos

Para lograr una excelente administración de los RR.HH. dentro de la empresa es necesario que su principal recurso: las personas, sean gestionadas, para poder lograr los fines de la organización. El principal objetivo de esta gestión es lograr una interacción con cada individuo que forme parte del capital humano y tener un estricto control de sus capacidades y sus relaciones tanto fuera como dentro de la empresa.

La administración de personal es una de las funciones que se realiza en los RR.HH. Dentro de la Administración de Empresa se reconoce como un área interdisciplinaria; donde la misma tiene como objeto de estudio y de acción la administración de las personas vinculadas de forma directa o indirectamente a una empresa o conjunto de empresas, a partir de la búsqueda de una coherencia cultural mínima en su dimensión laboral (WERTHER, et al, 2008).

Para realizar la gestión de personal es necesaria la creación de un ambiente de trabajo positivo, agradable y productivo. La buena gestión de empleados se logra con un personal que trabaja duro, disfruta de sus puestos de trabajo y se mantiene motivado.

Según lo expresado por Idalberto Chiavenato⁴, en su libro "*Administración de Recursos Humanos*" la gestión de personal posee 5 procesos esenciales que son interdependientes; pero a su vez están muy relacionados porque depende de la organización, debido a que, cada una los implementa de forma

³Juan Carlos Rodríguez: autor de libros como: "*El Modelo De Gestión De Recursos Humanos*"; "*Desenvolupament Del Factor Humà*"; "*Publicidad Y Comunicación Corporativa En La Era Digital*"; "*Teoría de las relaciones laborales. Desafíos*."

⁴Idalberto Chiavenato: Profesor brasileño conocido internacionalmente por la excelencia de sus trabajos en *Administración y Recursos Humanos*. Poseedor de dos títulos de Doctor Honoris Causa por universidades extranjeras por su contribución a la *Administración General y de Recursos Humanos*.

diferente. Dentro de los procesos se encuentran: provisión, aplicación, mantenimiento, desarrollo, seguimiento y control del personal(CHIAVENATO, 1999).

Se puede concluir que la gestión de personal en cada institución va a ser diferente, debido a que no todas necesitan la misma información de los individuos, ni se interesan por algunas de las relaciones que posee este. Aunque la mayor tendencia es la de abarcar toda la información desde los elementos fenotípicos y características físicas de cada persona, para tener un elevado nivel en su empresa porque dependen del personal para que las demás esferas estén sincronizadas.

1.3 Tendencias en el uso de sistemas de gestión de personal

Con el continuo intercambio de información que existe en el mundo y el progreso de la economía, una de las principales tendencias de las empresas es: tener un excelente control de sus trabajadores; posibilitando la obtención de buenos resultados en los negocios que se desarrollan. Para lograr tal objetivo se han dedicado a introducir la informatización de los procesos de RR.HH. (dentro de estos procesos la gestión de personal) mediante el desarrollo de *software*. Teniendo así la posibilidad de tener concentrada la información de los individuos en bases de datos que permiten una adecuada administración del principal recurso de la empresa.

Otra tendencia que existe para lograr un elevado control en los procesos de las empresas es la creación de Sistemas de Planificación de Recursos Empresariales, o ERP (por sus siglas en inglés, *Enterprise Resource Planning*). Estos sistemas son paquetes de *software* compuestos de módulos que son capaces de gestionar la información que se genera en determinado organismo. Dentro de estos módulos que se desarrollan se encuentran los de: Recursos Humanos, Ventas, Finanzas y Producción. Muchas empresas han adoptado estos sistemas para poder controlar a su personal. Los ERP también se pueden implementar en las universidades para lograr una solución a sus problemas.

Desde la década del 90 las universidades han optado por crear sistemas de Planificación Empresarial de Recursos Universitarios (ERP Universitarios) con el fin de informatizar todos los procesos que se realizan en la universidad. En Cuba se ha empezado con este impulso recientemente, no obstante, la gestión de personal en la administración de los RR.HH. se hace a través de *software* que permite el control de las personas que trabajan en la institución. En los próximos epígrafes se explican las características de sistemas internacionales, nacionales e institucionales que fueron estudiados en la presente investigación.

1.3.1 Sistemas de gestión de personal en el mundo

ERP University

Es un ERP que se está implantando en las universidades de Perú. Está diseñado para mejorar y

automatizar los procesos de las instituciones educativas de nivel universitario. Utiliza un único punto de control, con acceso desde una computadora o dispositivo portátil con Internet⁵; este objetivo se logra mediante la eliminación de duplicación de datos y procesos, la contribución de la integridad de la información, fácil actualización y reducción de costos en Tecnologías de Información y Comunicación(ERP UNIVERSITY, 2012).

En general, una de las ventajas del ERP University es que permite gestionar sus procesos de manera simple y eficiente siguiendo el modelo de Universidad Digital. Este sistema brinda soluciones basadas en las buenas prácticas de negocios y en una plataforma tecnológica libre, lo cual significa una independencia en el uso de la aplicación la cual tiene un funcionamiento en cualquier sistema operativo. Este paquete de gestión es rápido, ágil, versátil y de fácil uso para el usuario final, sin perder su potencialidad además de contar con el soporte técnico integral. Cumple con todos los requisitos exigidos por la Norma ISO 9001:2008 y CONEAU, adicionalmente, reduce el tiempo de la implementación para el proceso de acreditación(ERP UNIVERSITY, 2012).

Dentro de las principales características de este ERP está la de gestionar toda la información académica de los estudiantes en la universidad, llevando un control de los mismos mediante expedientes y reportes. Por lo que, los datos de los estudiante los maneja el módulo de Gestión Académica(ERP UNIVERSITY, 2012).

La GRH en el ERP se gestiona mediante un módulo nombrado Gestión de Recursos Humanos Con este se permite conocer el perfil potencial de los empleados; accediendo así a la realización de evaluaciones de personal de manera ágil, veraz y constante. Soluciona y reduce la complejidad de la gestión de nóminas, contrataciones y rendimiento del personal y evaluación del mismo. A través del módulo se puede realizar la función de gestionar el personal, permitiendo administrar los datos de las personas que estén en la institución. Las funcionalidades más importantes que se permiten gestionar de las personas son: configuración de maestros, registros de altas, datos personales, datos laborales, contratos de personal (ERP UNIVERSITY, 2012). El ERP University cuenta también con los módulos de: gestión de servicios TIC, gestión de la calidad, gestión logística, gestión financiera, gestión de investigación, gestión de producción, gestión de procesos y gestión de la documentación. Estos módulos complementan el trabajo en la institución para lograr que la gestión académica se efectúe de manera integral.

⁵ *Internet: Red de computadoras alrededor de todo el mundo que comparten información unas con otras por medio de páginas o sitios.*

Esta solución informática tiene como desventaja que solo gestiona la información de las personas de acuerdo con las características de su país, imposibilitando la configuración de estos datos según el lugar donde esté implantado el producto.

ERP UJI

El ERP UJI es un sistema de La Universidad Jaime I de España. Este sistema se realizó sobre plataforma Linux. Fue creado para la formalización de los procesos de negocio, para obtener un sistema integrado y brindarle un soporte a la entidad tanto en el área económica, académica, RR.HH. y servicios (CARRO, et al, 2009).

Dentro de los aspectos destacables del producto está que permite el modelado XML de los principales tipos de documentos tales como: actas, certificados, nóminas. Usan las tecnologías propietarias de Oracle (CARRO, et al, 2009). [11]

Dentro de sus principales módulos están los de estudiantes e investigación, donde manejan la información académica de cada uno de ellos. Además, tiene el módulo de personal donde se manejan los datos de las personas.

Este ERP recoge solo la información de los estudiantes y profesores no gestiona a otro tipo de personas. Este sistema presenta el inconveniente de que utiliza la tecnología de Oracle por lo que no cumple con las políticas del Centro de Informatización Universitaria que están encaminadas al uso de tecnologías libres.

Sistema de información universitaria (SIU)

Este sistema forma parte del Programa de la Educación Superior (PRES) de Argentina. Con la creación de este producto el principal objetivo era crear un sistema universitario nacional provisto de información confiable, íntegra y oportuna para una correcta asignación de los recursos del Estado. [15] Comenzó a funcionar desde 1996. Genera soluciones para la utilización del recurso de la información de las universidades, actuando así mismo como facilitador para la incorporación de nuevas tecnologías. El Programa SIU se compone de ocho *software* para las Universidades Nacionales, donde los cuales se listan a continuación:

- SIU- Comechingones - Gestión presupuestaria, financiera y contable.
- SIU- Pampa - Gestión de personal.
- SIU- Guaraní - Gestión de alumnos.
- SIU - Wichi - Consultas gerenciales sobre gestión presupuestaria y de personal.
- SIU - Araucano - Estadística de Alumnos.

- SIU - DataWarehouse - Soporte para toma de decisiones.
- ComDocII - Gestión de expedientes.
- SIU - Bibliotecas - Prestaciones generales para bibliotecas.

Como se menciona anteriormente en el sistema SIU el encargado de la gestión de personal es el SIU-Pampa, este es un sistema que es implantado en varias universidades de Argentina. Su principal objetivo es solucionar la administración de información sobre el personal y facilitar la liquidación a través de un “Legajo Único por Agente”⁶. Mantiene el legajo del empleado actualizado y la información legal es actualizada constantemente.

En el legajo electrónico único del empleado se encuentran sus datos personales, familiares, niveles de estudio, cargos y la historia de los cargos mantenidos por la persona en su relación con el organismo, por lo que el legajo se mantiene aún en el caso que una persona no posea más cargos vigentes en dicho organismo(CANOSA, et all, 2009).

Dentro de las características que posee este sistema se pueden mencionar las siguientes:

- Sistema integrado con otras instituciones.
- La base de datos de los agentes permite mantener una única fuente de información para la gestión de los Recursos Humanos en la institución.
- Incorpora las “reglas del negocio” de las distintas instituciones que van requiriendo nuevas funcionalidades y las deja disponibles, previa parametrización, para el resto de los usuarios. Esto enriquece el sistema y evita que existan diversas versiones.

Dentro de los servicios que se prestan están:

- Administración del Legajo Electrónico Único.
- Liquidación basada en datos del Legajo Electrónico.
- Captura y Consulta Descentralizada.
- Validación automática de Incompatibilidades.
- Control de Planta.
- Administración de Licencias.
- Alto nivel de parametrización.
- Flexibilidad en la definición de conceptos.
- Información Gerencial.
- Información para otros sistemas.

De este sistema se pudo obtener como aporte para la investigación los datos que se gestionan en el

⁶ Legajo Único por Agente: conjunto de datos únicos legales que posee una persona.

legajo único por agentes. Sin embargo, este sistema no se adopta en la propuesta de solución porque los datos de la dirección no se adecuan a lo que necesita el subsistema de Cooperación para realizar sus trámites de pasaporte.

1.3.2 Sistemas de gestión de personal en Cuba

GREHU

GREHU es un sistema de gestión de Recursos Humanos que fue desarrollado por Raúl Martínez Rodríguez⁷ en el año 1997. El mismo permite registrar, procesar y gestionar de forma integrada las principales funciones que se desarrollan en la dirección de Recursos Humanos de una entidad laboral. Este sistema posee varios subsistemas, entre ellos: selección y contratación, inventario de personal, control de las sanciones y amonestaciones, evaluación del desempeño, trabajo con los dirigentes y cuadros y procesamiento de las nóminas; relacionándose entre sí a través del sistema principal (MARTÍNEZ, 2008).

El sistema fue programado en *FOXPRO*⁸ para Windows debido a las dificultades que el mismo brinda para la gestión de las bases de datos y para la generación de forma rápida de pantallas, consultas, reportes, etc. (MARTÍNEZ, 2008).

Es un *software* que se encuentra implantado en varias de empresas de Cuba. El empleo de esta herramienta en los departamentos de personal y en los de economía ha implicado una más ágil realización del trabajo por parte de sus trabajadores (LÓPEZ, 2006).

Este sistema aunque gestiona la información de las personas no resuelve el problema del país debido a que fue realizado para utilizar con *software* privativo. Además, no todas las instalaciones gestionan de igual forma los recursos vivos que tienen en las empresas.

1.3.3 Sistemas de gestión de personal en la Universidad de las Ciencias Informáticas

UCID, Línea de Capital Humano, Módulo Registro de personas

La Unidad de Compatibilización Integración y Desarrollo (UCID) es desarrollada en la Universidad de las Ciencias Informáticas por parte de los cadetes que pertenecen a las FAR (Fuerzas Armadas Revolucionarias). Este sistema posee varias líneas de desarrollo, una de ellas es la de Capital Humano que está encaminada a conocer las habilidades, valores y conocimientos de las personas. Dentro de esta línea se desarrolla el *Módulo de registro de Personal* el cual contiene toda la información necesaria de cada individuo. Los elementos de este sistema son configurables, es desarrollado en el

⁷ Raúl Martínez Rodríguez: ingeniero en Sistemas Automatizados de Dirección, Doctor en Ciencias Técnicas, Profesor Titular, Centro de Estudios de Ingeniería de Sistemas, (CEIS), Instituto Superior Politécnico José Antonio Echeverría Cujae.

⁸ FoxPro: es un lenguaje de programación procedural, orientado a objetos que posee un Sistema Gestor de Bases de Datos

marco de trabajo Sauxe⁹.

Dentro del módulo se pueden encontrar las siguientes funcionalidades:

- Información básica
- Datos de contacto.
- Datos de tallaje.
- Ocupación Actual.
- Afiliaciones.
- Domicilio.
- Idiomas.
- Estudios Concluidos.
- Categorización.
- Números de identidad.
- Sanciones.

Entre los aportes que trajo el estudio de este sistema a la investigación, está la identificación de los principales datos que se pueden administrar de una persona que se encuentre en la universidad. También se identificó los tipos de identidad que puede tener una persona en el país.

Este módulo a pesar de que abarca un gran cúmulo de datos de las personas no va a formar parte de la propuesta de solución, debido a que no se gestionan los datos de la entidad; donde los mismos son fundamentales a la hora de realizar los procesos universitarios. Además, se está desarrollando en un marco de trabajo que no es el que utiliza el Departamento de Gestión Universitaria para crear el Sistema de Gestión Universitaria.

Sistema de Gestión Universitaria

El Sistema de Gestión Universitaria es un producto cubano que se está desarrollando en la Universidad de las Ciencias Informáticas. Es un ERP Universitario que incluye varias líneas de desarrollo agrupadas en las áreas de procesos: Pregrado, Postgrado, Producción, Investigación, Ingreso, Ubicación Laboral, Laboratorio, Residencia, Extensión Universitaria, Cooperación Internacional, Biblioteca y Teleformación. La principal ventaja de este ERP es precisamente la de integrar todos los procesos mencionados anteriormente, para que la gestión universitaria se realice con uniformidad.

⁹ Sauxe; es un marco de trabajo novedoso, que integra varias tecnologías libres, como: ZendFramework, PHPDoctrine, ezComponent, ExtJS y que cuenta con su propio framework ZendExt

El producto cuenta con un núcleo que tiene procesos de apoyo que trae como ventaja agrupar la información que es común para todos los subsistemas. Dentro de los procesos que se realizan en el núcleo se encuentran: la estructura y composición, seguridad, configuración, trazas, estadísticas, entre otros.

El Sistema de Gestión Universitaria es un sistema Web basado en tecnologías Web, siendo estas asequibles. Es desarrollado bajo licencia libre y es multiplataforma¹⁰. Los elementos del sistema son configurables, permitiendo así su adaptabilidad en otros lugares.

Dentro del Sistema de Gestión Universitaria las personas no son administradas de la misma forma debido a que no se cuenta con un módulo que controle a las mismas. La información de los trabajadores se obtiene del sistema ASSET (Sistema de gestión integral) que se encarga de administrar la información de cada uno. La información de los estudiantes se obtiene a través del Subsistema de Pregrado del módulo Personal y Secretaría. Cada subsistema utiliza la poca información que hay en el núcleo para realizar sus procesos y además el Subsistema de Residencia entra al sistema las personas que son visitantes o familiares que permanecen en la universidad.

Para la propuesta de solución no se puede utilizar el módulo de Personal y Secretaría que forma parte del subsistema de Pregrado. A pesar de que el módulo proporcione la información de los estudiantes, el mismo no gestiona todos los datos de superación de las personas que son necesarios en los otros. De estos datos solo se brinda el nivel escolar, dejando de gestionar la categoría científica, la categoría investigativa y el grado académico que son fundamentales para el subsistema de Postgrado. Además, en este módulo no se permite la entrada de personas que sean extranjeras.

ASSET (Sistema de gestión integral)

Es un Sistema de Gestión de Recursos Humanos de tipo ERP, de factura foránea se comenzó a utilizar a partir del año 1997. Está concebido para el área Económica y de Recursos Humanos de ministerios, entidades, centros de investigación, institutos, etc. Este sistema puede ser adaptado a los diversos objetos sociales de cualquier entidad (ASSETS, 2004).

El sistema cuenta con un módulo para gestionar los Recursos Humanos. Este está diseñado para calcular las nóminas y controlar los recursos laborales de una entidad de los cuales se pueden controlar íntegramente: empleados, la estructura organizativa de la entidad y su plantilla. Siempre que se introducen altas, bajas y otros movimientos de personal, se actualiza automáticamente el registro de empleados (catálogo de empleados) y se generan los reportes correspondientes (con el formato oficial) (ASSETS, 2004).

¹⁰ *Multiplataforma: es un término usado para referirse a los programas, sistemas operativos, lenguajes de programación, u otra clase de software, que puedan funcionar en diversas plataformas.*

Es posible modificar plantillas e introducir cambios en la estructura organizativa, crear nuevos cargos y realizar conversiones de plazas. En la UCI se utiliza este sistema para el personal que si es plantilla para poder controlar todos los procesos relacionados con ellos.

Este sistema no es multiplataforma y al ser un *software* privativo incurriría en gastos de licencia y soporte para el país, por lo que, no se puede utilizar directamente en la propuesta de solución.

1.3.4 Valoración de los sistemas existentes

Del estudio desarrollado de los distintos sistemas se obtuvo como resultados:

1. La gran mayoría de los sistemas abarcan una amplia gama de funcionalidades que van encaminadas a la gestión de los RR.HH., pero no se centran fundamentalmente en la administración del personal, sino que lo manejan a la hora de realizar varias de las acciones para la que están destinados, debido a que cada empresa maneja este término de forma diferente.
2. Ninguno de estos sistemas agrupa las necesidades de la universidad para realizar la gestión de la información de las personas a la hora de realizar los procesos universitarios, debido a que no permiten el tratamiento de personas que no sean cubanos.
3. Los sistemas: GREHU, ASSET usan el sistema operativo Windows por lo que no contribuyen a la política de migración a Software Libre del Estado Cubano y el ERP UJI utiliza tecnologías privativas.
4. Se identificó las principales categorías para agrupar la información de las personas.
5. Se seleccionaron las principales características que se gestionan de las personas que estén de acorde con las especificaciones de la gestión universitaria.

1.4 Entorno de desarrollo

Para darle solución al problema planteado en la investigación, se hizo necesario el estudio de los elementos que van a conformar el entorno de trabajo de la misma. A la hora de realizar este estudio se tuvo en cuenta que la solución informática que se pretende construir va a estar integrado al Sistema de Gestión Universitaria, por lo que, se estudió el entorno de desarrollo que este sistema usa y a partir de ahí se seleccionaron todos los elementos necesarios para integrarlos a la investigación. A continuación se presentan todas las herramientas, lenguajes y tecnologías con sus principales características.

1.4.1 Lenguajes de desarrollo

Según la Real Academia Española¹¹, en la informática un lenguaje es un conjunto de signos y reglas que permite la comunicación con un ordenador. Con la evolución de la Informática han surgido los lenguajes de modelado y de programación que facilitan el trabajo a la hora de desarrollar un producto.

¹¹ La REAL ACADEMIA ESPAÑOLA (RAE): es una institución que tiene como finalidad básica fijar la norma que regula el uso correcto de la lengua española armonizando su esencial unidad con la diversidad de su realización.

A continuación se describen los lenguajes que se pretenden utilizar para la confección de la propuesta de solución.

Hojas de estilo en cascada (CSS) (versión 2)

CSS (acrónimo de *Cascading Style Sheets*, en español: Hojas de Estilo en Cascada) es un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML. El W3C (acrónimo de *World Wide Web Consortium*) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los navegadores. La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación (JUBE, 2008).

Lenguaje de marcado de hipertexto (HTML) (versión 4)

HTML (acrónimo de *Hypertext Markup Language*, en español: Lenguajes de Marcado de Hipertexto) es el formato de los documentos de la WWW (acrónimo de *World Wide Web*). Los documentos HTML son archivos de texto plano que pueden ser creados mediante cualquier editor de texto, aunque también existen programas específicos para editar HTML, concebidos específicamente para editar páginas Web en ese lenguaje.

HTML no permite definir de forma estricta la apariencia de una página, aunque en la práctica, se utiliza también como un lenguaje de presentación. Estos archivos HTML se leen en un navegador Web (Netscape Navigator, Microsoft Explorer, Mozilla, Opera, etc.) La presentación de la página es muy dependiente del navegador o browser utilizado ya que el mismo documento no produce el mismo resultado en la pantalla si se visualiza con uno u otro, o sea, HTML se limita a describir la estructura y el contenido de un documento, y no el formato de la página y su apariencia.

Lenguaje de modelado

En el proceso de desarrollo de *software* se utilizan los lenguajes de modelado. Estos lenguajes se utilizan para interpretar los componentes¹² de una estructura. Pueden representarse de diferentes formas y dentro de ellas está la gráfica, donde cada símbolo representa un concepto y las líneas identifican las relaciones existentes entre ellos.

En la actualidad, de los lenguajes de modelados existentes uno de los más utilizados es el UML, este ha sido adoptado por la gran mayoría de empresas que se dedican a desarrollar soluciones informáticas, en el epígrafe siguiente se podrá conocer una breve descripción de este lenguaje.

Lenguaje Unificado de modelado (UML) (versión 2.1)

¹² *Componente: parte física y reemplazable de un sistema que se ajusta a, y proporciona la realización de, un conjunto de interfaces.*

UML (acrónimo de *Unified Modeling Language*, en español: Lenguaje Unificado de Modelado) es "un lenguaje estándar" para la escritura de modelos de *software*. UML se puede utilizar para visualizar, especificar, construir y documentar artefactos de un sistema de *software*. [8]

Contiene construcciones organizativas que permiten la agrupación de los modelos por paquetes, posibilitando la división de grandes sistemas en piezas de trabajo manejables en un entorno de desarrollo complejo (BOOCH, et al, 2005).

Un modelo UML está compuesto por tres clases de bloques de construcción: los elementos, relaciones y diagramas. Como método formal de modelado aporta las siguientes ventajas:

- Mayor rigor en la especificación.
- Permite realizar una verificación y validación del modelo realizado.
- Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos) (PODESWA, 2005).

Lenguajes de programación

Los lenguajes de programación son herramientas que permiten crear programas y *software*. Los mismos se pueden clasificar en lenguajes interpretados (conocidos como intérpretes) y compilados (conocidos como compiladores) (GARCÍA, et al, 2007). En la presente investigación se seleccionaron varios lenguajes que posibilitarán la construcción de la propuesta de solución. A continuación se especifican cada uno.

Hipertexto Pre-Procesado (PHP) (versión 5.3)

PHP (acrónimo de *Hypertext Preprocessor*, en español: Hipertexto Pre-Procesado) es un lenguaje de programación interpretado que posibilita la generación dinámica de contenidos en un servidor Web. Se destacan entre sus características principales su alto rendimiento, potencia, facilidad de aprendizaje y los pocos recursos que consume. Entre las características que posee este lenguaje y que lo convierten en una potente herramienta están: (GUTMANS, et al, 2005)

- Es un lenguaje multiplataforma.
- Soporte sólido para Programación Orientada a Objetos.
- Incorpora bibliotecas que contienen funciones integradas para realizar útiles tareas relacionadas con la Web.
- Es un *software* de código abierto¹³.
- Soporta bases de datos entre las que se pueden mencionar: MySQL y PostgreSQL.

¹³ Código abierto: en inglés; *Open Source*, es el término con el que se conoce al software distribuido y desarrollado libremente

JavaScript (versión 1.6)

JavaScript es un lenguaje de programación que permite a los desarrolladores crear acciones en sus páginas Web, puede ser utilizado por profesionales y para quienes se inician en el desarrollo y diseño de sitios Web. No requiere de compilación, por lo que, es un lenguaje interpretado donde el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos(BERNAL, 2009).

Es orientado a objetos, debido a que dispone de herencia, la cual se realiza siguiendo el paradigma de programación basada en prototipos, ya que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad(BERNAL, 2009).

1.4.2 Tecnologías de desarrollo

Una tecnología es un conjunto de teorías y de técnicas que permiten el aprovechamiento práctico del conocimiento científico. Las tecnologías de desarrollo han estado en un lugar primordial en la construcción de productos gracias al avance de las TIC, debido a que, posibilitan que el producto se desarrolle en un ambiente agradable y que cuente con buena calidad. Dentro de las más utilizadas están los marcos de trabajo, las librerías y otros. En los siguientes epígrafes se explican las tecnologías que se van a utilizar en el desarrollo de la investigación.

Marco de trabajo

Un marco de trabajo (en inglés: *framework*), dentro del ambiente de desarrollo de *software*, es una estructura de soporte definida en la cual otro

Típicamente, un marco de trabajo es un conjunto de proyecto de *software* puede ser organizado y desarrollado. bibliotecas, herramientas y normas a seguir que ayudan a desarrollar aplicaciones. Estos permiten la reutilización de código y la estandarización del desarrollo(LAFOSSE, 2010).

En la investigación se propone utilizar el marco de trabajo GUUD (acrónimo creado con las iniciales de los departamentos del centro CENIA: Gestión Universitaria, Universidad Digital y Gestión Documental) que es la unión del marco de trabajo CodeIgniter y JQuery, este marco es el propuesto por el grupo de arquitectura del centro CENIA. A continuación se describen por separados los dos marcos de trabajos que conforman la propuesta.

CodeIgniter (versión 1.7.3)

CodeIgniter es un marco de trabajo para PHP. Es adecuado para desarrollos que requieran mucho rendimiento, que ejecutan muchas versiones de PHP con diferentes configuraciones. Una de sus mayores ventajas es la documentación que se ofrece en internet. Su principal objetivo es ayudar a que

los desarrolladores puedan realizar proyectos mucho más rápido que creando toda la estructura desde cero(CODEIGNITER, 2012).

CodeIgniter permite concentrarse en el desarrollo del proyecto en cuestión, minimizando la cantidad de código necesaria para realizar las tareas. CodeIgniter usa el patrón de diseño arquitectónico Modelo-Vista-Controlador como paradigma de arquitectura de desarrollo, la cual separa en 3 capas distintas: la representación de datos, el interfaz de usuario y el controlador de eventos respectivamente(CODEIGNITER, 2012).

JQuery (versión 1.3.2)

JQuery es un nuevo tipo de biblioteca o marco de trabajo de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, permitiendo manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX¹⁴(acrónimo de *Asynchronous JavaScript And XML*, en español: JavaScript Asíncrono y XML) al sistema. JQuery al igual que otras librerías, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código. Es decir, con las funciones propias de esta librería se logran grandes resultados en menos tiempo y espacio(JQUERY, 2010).

Con JQuery se ahorra muchas líneas de código, se mejora el tiempo de creación y depuración, esta tecnología tiene licencia para uso en cualquier tipo de plataforma, personal o comercial.

GUUD (versión 1.0)

En la investigación al utilizar el marco de trabajo GUUD se incorporan novedades y modificaciones en su infraestructura, a continuación se listan las mismas.

Del lado del cliente:

1. Se implementaron una serie de *widgets* para utilizarlos de interfaz de algunos de los *widgets* base de *jquery-ui* como por ejemplo el *date*, el *tab* (ambos son interfaces de los *widgets* de mismo nombre de *jquery-ui*) y el *popup* (interfaz del *dialog* de *jquery-ui*). Además de los ya mencionados, se implementaron otros nuevos entre los que se encuentran: *attach*, menú, *message*, *tooltip*, *form* (se construyó con la unión de los *plugins form* de JQuery el cual se utiliza para el envío de formularios AJAX y el *validate* utilizado para validar formularios), *grid* (utiliza como *plugin* el *jqgrid*), *multiselect* (para hacer selecciones múltiples), *navbar* (para la creación de barras de navegación), *tree* (para la creación de árboles) y el *graph* (utiliza la librería *Highchart*).
2. Se le implementó un *plugin* a JQuery para el manejo de espacios de nombre e internacionalización.

¹⁴ AJAX: es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications)

3. Se implementaron funciones comunes para todo el sistema (contenidas en los archivos *core.js* y *common.js*) entre las que se destacan: *loadIn*, *getDataJson*, *createSelect*, *isArray*, *isFunction*, *site_url*.

Del lado del servidor (hechas a Codelgniter):

1. Se le agregó manejo de excepciones y mensajes.
2. Se le implementó el IOC (acrónimo de *Inversion of Control*, en español: Inversión de Control) para la interacción entre módulos.
3. Se le añadió la característica de la modularidad o sea que una aplicación pueda dividirse en módulos. Codelgniter no cuenta con esta posibilidad.
4. Se añadieron, modificaron y extendieron los *helpers* o asistentes.

Sistema Gestor de Base de Datos (SGBD)

Un Sistema Gestor de Bases de Datos (SGBD) o DBMA (acrónimo de *DataBase Management System*) es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta (RAMOS, 2007).

Un SGBD permite definir los datos a distintos niveles de abstracción y manipularlos, garantizando la seguridad e integridad de los mismos. Algunos ejemplos de SGBD son Oracle, DB2, PostgreSQL, MySQL, MS SQL Server, etc (RAMOS, 2007).

Las características de un SGBD son: abstracción de la información, independencia, redundancia mínima, consistencia, seguridad, integridad, respaldo y recuperación, control de la concurrencia.

PostgreSQL (versión 8.4.1)

PostgreSQL se distribuye bajo la licencia de PostgreSQL, un liberal de licencia de código abierto, similar a las licencias BSD¹⁵ (acrónimo de *Berkeley Software Distribution*) o MIT¹⁶. Ofrece soporte al lenguaje SQL, integridad de transacciones y extensibilidad de tipos de datos. PostgreSQL, además la instalación es ilimitada. PostgreSQL está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo (GONZÁLEZ, 2011).

Servidor Web Apache (versión 2.2.2)

Apache es un servidor Web configurable, robusto y estable. Su licencia permite modificar su código fuente, incluyéndolo así dentro de los impresionantes productos del *software* libre que son utilizados en

¹⁵ Licencia BSD: es una licencia de software libre permisiva y tiene menos restricciones en comparación con otras, estando muy cercana al dominio público

¹⁶ Licencia MIT: es una licencia de software, no tiene copyright, lo que permite su modificación y es muy parecida a la licencia BSD en cuanto a efectos

la actualidad. Es multiplataforma, permite la creación de ficheros log según la necesidad del administrador, posibilitando un mayor control de la información (APACHE,2010).

1.4.3 Herramientas de desarrollo

Las herramientas de desarrollo posibilitan el trabajo a la hora de realizar diagramas, clases, interfaces y otros elementos cuando se está construyendo el *software*. Sirven de apoyo para la construcción de los artefactos que se generan. En el mundo una de las herramientas más utilizadas en el desarrollo de producto son las Case. En el siguiente epígrafe se describen sus principales características.

Herramientas Case

Las herramientas CASE (acrónimo de *Computer Aided Software Engineering*, en español: Ingeniería de Software Asistida por Computadora) son *software* de apoyo al desarrollo, mantenimiento y documentación informatizados(CAMPDERRICH, 2003).

Son diversas aplicaciones informáticas destinadas a aumentar la efectividad en el desarrollo de *software* y sistemas, en términos de productividad y calidad, contribuyendo a una reducción de costos de producción.

Entre los beneficios ofrecidos por Case se encuentra la facilidad para la revisión de aplicaciones, brinda soporte para el desarrollo de prototipos de sistemas. Permite la generación de códigos (SOMERVILLE, 2005).

Visual Paradigm (versión8.0)

Herramienta CASE que acelera el desarrollo de aplicaciones, sirviendo de intermediario visual entre arquitectos, analistas y diseñadores de *software*, mediante un ambiente de modelado superior que posibilita un trabajo más fácil y dinámico (VISUAL PARADIGM, 2012).

Es multiplataforma y cuenta con una versión libre para la comunidad (*Community Edition*). Genera la documentación del sistema en los formatos PDF¹⁷, HTML y el formato de documentos de Microsoft Word y permite importar proyectos de otras herramientas de modelado como Rational Rose, Erwin y Microsoft Visio. Soporta la revisión ortográfica, brindando sugerencias para los idiomas: inglés, español, francés, alemán y portugués (VISUAL PARADIGM, 2012).

Esta versión introduce una serie de nuevas características, que incluyen: proteger con contraseñas los diagramas y el ajuste del tamaño de la aplicación de la fuente (VISUAL PARADIGM, 2012).

Entorno Integrado de Desarrollo (IDE)

¹⁷ PDF: acrónimo del inglés *Portable Document Format (Formato de Documento Portátil)*, es un formato de almacenamiento de documentos, desarrollado por la empresa Adobe Systems.

Un IDE es un programa que comprende un entorno de programación amigable para uno o varios lenguajes, brindando facilidades al programador, tales como: un editor de código, un compilador, un depurador y, opcionalmente, un constructor de interfaz gráfica.

NetBeans (versión 6.7.1)

NetBeans tiene una doble licencia; CDDL (acrónimo de *Common Development and Distribution License*, en español: Desarrollo Común y Licencia de Distribución) y (GPL2 GNU *General Public License versión 2 with Classpath exception*, en español: Versión de la Licencia Pública General 2, con excepción de rutas de clases) (NETBEANS, 2012).

Es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. El IDE NetBeans es un producto libre y gratuito sin restricciones de uso (NETBEANS, 2012).

PgAdmin III (versión 1.10.5)

PgAdmin III es una aplicación gráfica para administrar el gestor de bases de datos PostgreSQL, siendo la más completa y popular de código abierto. Es capaz de gestionar versiones a partir de la PostgreSQL 7.3 ejecutándose en cualquier plataforma. Está diseñado para responder a las necesidades de todos los usuarios, desde escribir consultas SQL simples hasta desarrollar bases de datos complejas (NORIEGA, 2010).

El interfaz gráfico soporta todas las características de PostgreSQL y facilita enormemente la administración. La aplicación también incluye un editor SQL con resaltado de sintaxis, un editor de código de la parte del servidor y un agente para lanzar *scripts* programados. La conexión al servidor puede hacerse mediante conexión TCP/IP y puede encriptarse mediante SSL (acrónimo de *Secure Sockets Layer* - Protocolo de Capa de Conexión Segura) para mayor seguridad. Está soportado por la licencia BSD (NORIEGA, 2010).

Pencil Project (versión 1.3.4)

Esta herramienta es una simple extensión de Firefox que cuenta con una considerable potencia y flexibilidad. Es la alternativa libre y multiplataforma para realizar el prototipado. Permite crear proyectos con varias pantallas, en las cuales es posible añadir cualquier elemento de una GUI (acrónimo de *Graphical User Interface*, en español: Interfaz Gráfica de Usuario) típica (botones, tablas, cajas de activación, deslizadores). Cuenta con multitud de elementos Web. Una de las características más importante es que permite alinear objetos para que el conjunto quede ordenado (PENCIL PROJECT, 2010).

1.5 Proceso de desarrollo de *software*

Un proceso de desarrollo de *software* no es único, tiene como propósito la producción eficaz y eficiente de un producto *software* que reúna los requisitos del cliente. No existe uno que sea efectivo para todos los contextos de proyectos de desarrollo, aunque existe un conjunto de actividades fundamentales que se muestran presentes en todos (PRESSMAN, et all, 2005).

Dentro de estas actividades está la especificación de *software*, el diseño e implementación, la validación y la evolución. Además, Pressman¹⁸ menciona un conjunto de “actividades protectoras”, que se aplican a lo largo de todo el proceso de desarrollo del *software*. Dentro de estas se encuentran: el seguimiento y control de proyecto, las revisiones técnicas, la garantía de la calidad del software, las mediciones, la gestión de riesgo, etc (PRESSMAN, et all, 2005).

1.5.1 Proceso de desarrollo de *software* con enfoque ágil basado en CMMI nivel 2

En la actualidad existen varias tendencias a reducir el valor, el tiempo de desarrollo, la agilidad y la fiabilidad del producto. [16] En la presente investigación se definió usar un proceso de desarrollo con enfoque ágil, debido a las oportunidades que brinda el manifiesto, dentro de las cuales están: (DÍAZ DE MERA, 2011)

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas.
- Desarrollar *software* que funciona más que conseguir una buena documentación.
- La colaboración con el cliente más que la negociación de un contrato.
- Responder a los cambios más que seguir estrictamente un plan.

Junto con este enfoque se define basar este proceso en el nivel 2 de CMMI (acrónimo de *Capability Maturity Model Integration*, en español: Modelo de Madurez y Capacidad Integrado). CMMI se presenta como un modelo de mejora de procesos en el desarrollo de *software* que puede complementarse a la norma ISO 9001. Fue desarrollado por el Instituto de Ingeniería de Software (SEI, acrónimo de *Software Engineering Institute*) de la Universidad Carnegie Mellon y publicada su primera versión en el año 2000 (CHIRISSIS, 2003).

Este basamento del modelo de calidad va a estar encaminado específicamente al nivel 2 de la capacidad de los procesos. Con el alcance de este nivel se permite que los procesos sean gestionados y además de ejecutarse, se planifica, se revisa y se evalúa para comprobar que cumple con los requisitos mediante la interacción de las 7 áreas fundamentales las cuales se pueden observar en la Figura 1.

¹⁸ Pressman: ingeniero de software estadounidense, autor y consultor. Presidente de la compañía Pressman & Associate. Reconocido autor de obras que se utilizan en la Ingeniería de Software y en la mejora de procesos.



Figura 1 Áreas de procesos que intervienen en el nivel 2 de CMMI

Estas áreas de procesos van a guiar el proceso de desarrollo del producto durante todo su ciclo de vida para lograr que tenga un alto grado de calidad y pueda ser comercializado.

1.6 Conclusiones parciales

Con la realización de este capítulo se puede concluir que:

1. La gestión de personal abarca la interacción de cada persona que esté presente en la institución, lleva un estricto control de las relaciones de los individuos, tanto dentro como fuera de la institución y se encarga de recopilar toda la información de cada recurso vivo de la empresa.
2. Los sistemas privativos estudiados no se pueden utilizar en la propuesta de solución, debido a que el país incurriría en gastos elevados a la hora de dar soporte y estos no cumplen con las políticas de migración de Cuba.
3. Los sistemas que usan Software Libre no cumplen con las necesidades existentes para la Gestión Universitaria, por lo que, no se pueden incluir en la propuesta de solución.
4. De los sistemas se pudo identificar las principales características de las personas que pudieran formar parte de la propuesta de solución.
5. Las herramientas estudiadas facilitarán el trabajo a la hora de realizar la solución informática.

Capítulo 2 Concepción de la propuesta de solución

2.1 Introducción

Para la realización de un sistema informático se van desarrollando artefactos que, luego de ser integrados, posibilitan responder a las necesidades del cliente. Estos artefactos se van creando por etapas, que van desde la declaración del problema y los requisitos que necesita el sistema, hasta las pruebas y la liberación del producto. Los procesos de desarrollo con enfoque ágil proveen una guía de ayuda para realizar la planificación, actividades y documentos, permitiendo que el cliente se sienta como parte del equipo de desarrollo y pueda ir revisando todos los artefactos que se generen, además de reducir drásticamente los tiempos de desarrollo pero manteniendo una alta calidad.

En el presente capítulo se describirá el módulo de Personal para el núcleo del Sistema de Gestión Universitaria. Se analizará sus funcionalidades y elementos principales, utilizando el proceso de desarrollo con enfoque ágil basado en el nivel 2 de CMMI, del que se incluyen algunos de sus artefactos: el modelo de dominio y la especificación de requisitos. Igualmente se incluirán la arquitectura y patrones de diseño utilizados, así como la descripción del modelo de datos del sistema.

2.2 Modelo de dominio

Para poder desarrollar la propuesta de solución y encontrar las nuevas funcionalidades, es necesario entender cómo se desarrolla este proceso en la actualidad, este estudio se desarrolló mediante el modelado de dominio, debido a que, según Craig Larman¹⁹ este modelo: “explica los conceptos significativos en un dominio del problema” (LARMAN, 1999).

Mediante la realización de este modelo se pueden definir los principales conceptos, sus asociaciones y los atributos. Además de que contribuye a esclarecer la terminología de la nomenclatura del dominio(LARMAN, 1999).

Como se menciona en el estudio que se realizó al Sistema de Gestión Universitaria en el Capítulo 1 la información se obtiene de sistemas externos, simulando así la nutrición heterótrofa de las plantas que es la que obtiene sus alimentos de otros seres. También obtienen información de un propio subsistema teniendo semejanzas con la nutrición autótrofa de las plantas que significa que se alimenta por sí mismo.

Para gestionar el personal en el núcleo, actualmente, se piden los datos de las personas de tres fuentes de información principales, dentro de las que están: el Sistema ASSET, el Sistema de terceros y Pregrado. Toda esta información el encargado de la base de datos la recibe y la almacena para una posterior utilización. En la Tabla 1 se muestran los principales conceptos identificados:

¹⁹Craig Larman: experto en informática canadiense. Autor de varios libros que se enfocan al desarrollo iterativo incremental, desarrollo ágil de software, el análisis orientado a objetos, diseño orientado a objetos entre otros.

Tabla 1 Definición de conceptos para procesos de gestión de personal

Conceptos	Descripción
Estudiante	Persona que estudia en la universidad en los cursos que se ofertan, puede ser estudiante del curso regular diurno o el curso por encuentro.
Trabajador	Persona que trabaja en la universidad, que puede ser adjunto al centro, profesor, técnico, auxiliar de limpieza, etc.
Tercero	Persona que brinda servicio en la universidad.
Sistema ASSET	Sistema que se utiliza para guardar, crear y gestionar el expediente laboral de los trabajadores mientras se encuentren en la universidad.
Subsistema Pregrado	Subsistema que está integrado al Sistema de Gestión Universitaria que brinda la posibilidad de gestionar los procesos docentes que se realizan en la escuela con los estudiantes.
Sistema de Terceros	Sistema donde se registran los datos de las personas terceras a la universidad.
Sistema de Gestión Universitaria	Sistema encargado de gestionar todos los procesos que ocurren en la universidad.
Subsistema de Gestión Universitaria	Subsistema que gestiona los procesos que tengan vinculados en sus negocios. Cada subsistema corresponde a un área de desarrollo de la universidad. Dentro de los subsistemas que se encuentran en el Sistema de Gestión Universitaria están: Pregrado, Postgrado, Producción, Investigación, Ingreso, Ubicación Laboral, Laboratorio, Residencia, Extensión Universitaria, Cooperación Internacional, Biblioteca y Teleformación

Teniendo en cuenta la identificación de los conceptos anteriormente explicados se pudo estructurar el modelo de dominio que se presenta en la Figura 2:

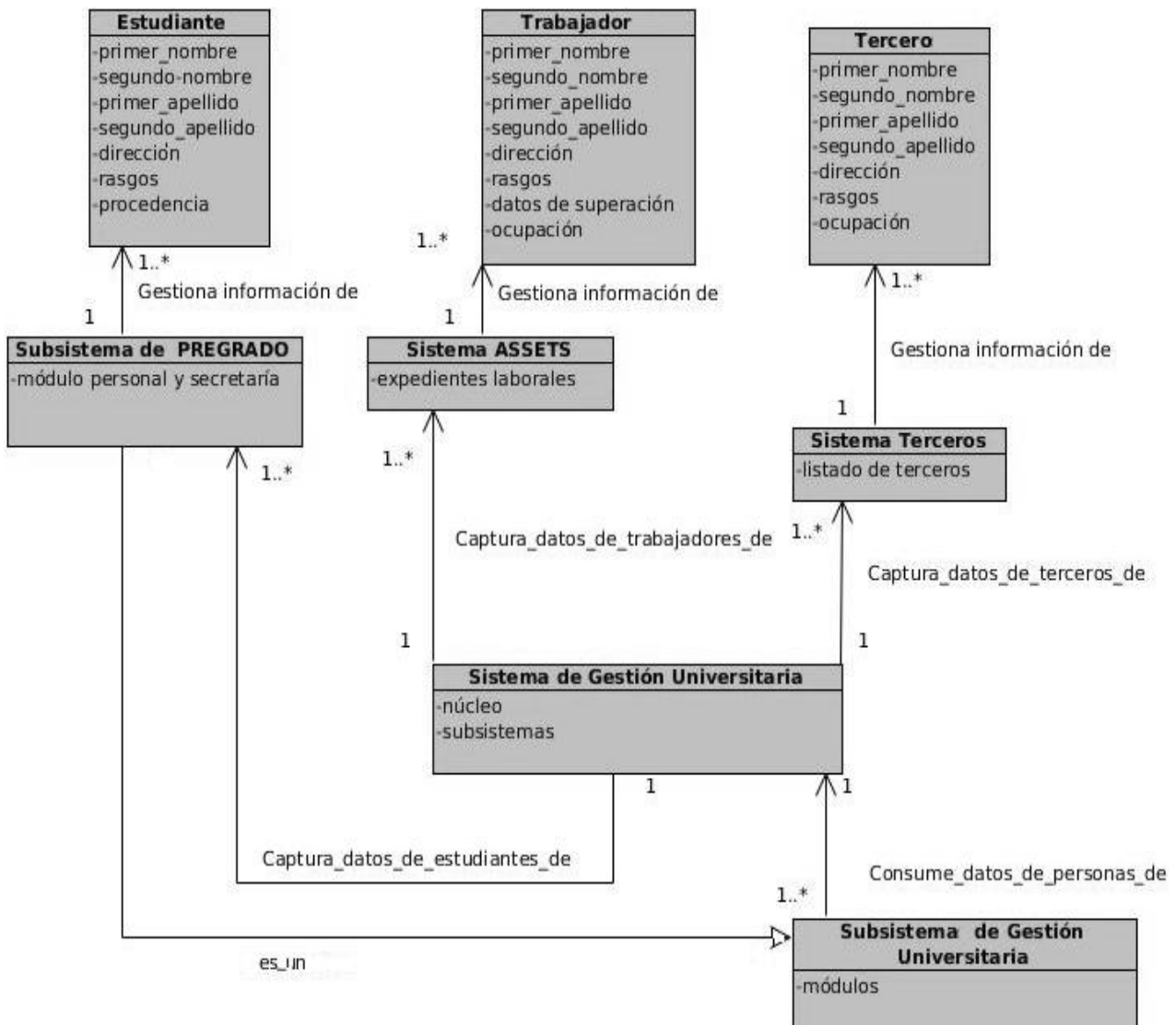


Figura 2 Modelo de dominio

2.3 Descripción del módulo de Personal

Respondiendo al problema de la investigación se hace necesario que la administración de personal en el núcleo del Sistema de Gestión Universitaria sea configurable y que tenga nuevas funcionalidades. Permitiendo así, que toda la información de las personas que se manejan en cada negocio quede concentrada en el núcleo. Respondiendo a estas necesidades se hace imprescindible desarrollar el módulo de Personal.

Al no incluir ninguno de los sistemas estudiados en la propuesta de solución porque no contaban con todos los elementos necesarios, se estudió con más énfasis cada uno de los subsistemas de Gestión Universitaria. Este estudio se realiza para encontrar la información sensible que manejan de los datos

de las personas, para así poder encontrar las nuevas funcionalidades. En la Tabla 2 que se muestra a continuación se puede encontrar las informaciones obtenidas a través de las entrevistas estructuradas que se aplicaron a los jefes de cada uno de los subsistemas. El modelo de la entrevista se puede encontrar en el Anexo 1.

Tabla 2 Información sensible de los subsistemas del Sistema de Gestión Universitaria

Pregrado
Carné de identidad, número de serie del carné de identidad, provincia, tomo, folio, registro civil, primer nombre, segundo nombre, primer apellido, segundo apellido, sexo, ojos, raza, lugar de nacimiento, residencia permanente, apartamento, no, entre, localidad país, provincia, municipio, teléfono, estado civil, cantidad de hijos, nacionalidad, huérfano y organizaciones políticas.
Postgrado
Carné de identidad, primer nombre, segundo nombre, primer apellido, segundo apellido, grado académico, categoría investigativa, categoría científica.
Residencia
Primer nombre, segundo nombre, primer apellido, segundo apellido, sexo.
Cooperación
Carné de identidad, serie del carné de identidad, primer nombre, segundo nombre, primer apellido, segundo apellido, sexo, estado civil, estatura (cm), color de la piel, color de los ojos, color del cabello, ciudadanía, lugar de nacimiento, calle, número, apto, entre1, entre2, circunscripción, localidad, carretera, km, localidad, finca, provincia, municipio.
Investigación
Primer nombre, segundo nombre, primer apellido, segundo apellido.

Luego del estudio realizado se identificaron las principales funcionalidades de la propuesta de solución. Como el módulo estaría integrado al núcleo del Sistema de Gestión Universitaria no será necesario implementar un gran número de funcionalidades como las relacionadas con la autenticación del usuario, la gestión de roles, entre otras. En próximos epígrafes se abordará mejor la integración del módulo al núcleo del Sistema de Gestión Universitaria y su relación con los demás subsistemas.

2.3.1 Características del módulo de Personal

El módulo de Personal estará compuesto por tres agrupaciones funcionales donde cada una de ellas va a contar con una serie de funcionalidades. A continuación se explican cada una de las mismas.

Funcionalidad “Personas”

Esta funcionalidad agrupa las acciones referentes a la búsqueda, creación y modificación de la información de personas. El buscador trae como aporte que se añaden nuevos criterios de búsqueda

avanzadas, que permitirá la búsqueda sobre una fuente de mayor cantidad de características de las personas. A su vez las búsquedas sencillas permiten una rápida búsqueda de información.

Además, en esta funcionalidad se brinda la posibilidad de crear y modificar los datos. Esto trae como aporte que se puedan ingresar en el sistema información de personas que tengan distintos tipos de identidad, es decir, personas que cuenten con diferente identificación (pueden ser: pertenecientes a la FAR, personas extranjeras o cubanas).

En las opciones de crear y modificar se pudo clasificar en 9 agrupaciones la información referente a cada persona para realizar la gestión universitaria. Dentro de las cuales se encuentran: la información nominal, la dirección particular, los rasgos fenotípicos, los datos personales, los datos de contacto, las afiliaciones, los datos de superación y los datos de la entidad.

Funcionalidad “Fuentes de información”

Esta funcionalidad tiene gran importancia debido a que mediante ella es que se van a obtener los datos de las personas de sistemas externos, su surgimiento está en la necesidad de hacer el sistema configurable por si en un futuro se implanta en otras universidades.

En ella se brinda la posibilidad de crear y modificar una configuración de fuente, donde las mismas pueden ser: base de datos, archivos Excel, archivos CSV, archivos Access y LDAP (Directorio activo).

Estas configuraciones se utilizarán a nivel de dato para volcar la información y luego normalizarla.

Funcionalidad “Configuración”

La funcionalidad de configuración brinda como lo dice su nombre configurar los elementos que componen la información de la persona, por si estos cambian en un momento determinado. Se brinda la oportunidad de crear y modificar los mismos.

2.3.2 Usuarios relacionados con el módulo

El módulo de Personal gestiona información sensible de las personas, por lo que, es necesario que accedan solamente las personas indicadas. El módulo solamente será visible para los siguientes usuarios:

- Grupo de informatización universitaria: hace referencias a las personas encargadas de controlar los procesos horizontales del núcleo, donde el rol que tendrían sería el de administrador.

2.3.3 Integración del módulo al Sistema de Gestión Universitaria

Como ya se había mencionado con anterioridad el módulo de Personal va a pertenecer al núcleo del Sistema de Gestión Universitaria por lo que se establece comunicación con ellos de la siguiente forma:

Módulo Seguridad: la comunicación con este módulo va a permitir la autenticación de los usuarios y gestión de los permisos sobre las funcionalidades del módulo.

Módulo Configuración: permitirá realizar las configuraciones del módulo, tanto globales, como las que son usadas por más de un subsistema.

Módulo Estructura y Composición: gestiona todo lo referente a las responsabilidades que tiene una persona en la universidad, de este módulo se obtendrán todos los datos necesarios para saber el área y cargo de cada persona.

La solución informática propuesta se integrará externamente con el Sistema ASSET y con el módulo de Pregrado para obtener los datos de las personas.

2.4 Requisitos

Existen muchos conceptos de requisitos de *software*. Según se especifica en el glosario de términos de la IEEE un requisito es: (IEE, 2012)

- Una condición o capacidad necesaria por un usuario para resolver un problema o alcanzar un objetivo.
- Una condición o capacidad que debe cumplir un sistema o componentes del sistema para satisfacer un contrato, estándar, especificación u otro documento formalmente impuesto.
- Una representación documentada de una condición o capacidad.

Por otro lado, los autores Sommerville y Sawyer asumen que un requisito es: “una especificación que se debería implementar. Son descripciones de cómo se debe comportar el sistema, o de un atributo o propiedad del sistema. Puede ser una restricción en el proceso de desarrollo en un sistema” (SOMMERVILLE, et all, 1997).

Todos estos autores llegan a la conclusión de que un requisito es necesario a la hora de realizar un producto, que son una guía de lo que debe tener, de lo que debe hacer y que deben estar especificados en algún documento donde el cliente y el equipo de desarrollo tengan un mismo entendimiento, además de validarlos. En la presente investigación atendiendo a estos conceptos se realizó la captura de requisitos mediante diferentes técnicas. Especificando cada uno y validándolos. En los siguientes epígrafes se abordará acerca de la captura y especificación de los requisitos, la validación de los mismos se tratará más adelante en el Capítulo 3.

2.4.1 Captura de requisitos

La captura de los requisitos se refiere a de donde vienen los requisitos del *software* y cómo el ingeniero de *software* puede recogerlos. Es la primera etapa en la construcción de una comprensión del problema que el *software* quiere solucionar (SWEBOK, 2004).

Técnicas de captura de requisitos

Dentro de las técnicas de captura de requisitos que se emplearon para la investigación se encuentran:

Entrevistas: se realizaron entrevistas estructuradas con las personas que se verían afectadas por el uso del sistema propuesto, ya sean usuarios finales que interactúan con el sistema como miembros de la organización en la que se desarrolla el proceso de negocio a informatizar, con el objetivo de entender el dominio del problema y sus necesidades.

Escenarios: esta técnica constituye un valioso medio para proporcionar contexto a las exigencias del consumidor. En la investigación se usa para propiciar un marco de trabajo con el usuario al realizarle preguntas como: “y sí” y “cómo se hace esto”.

Prototipos: es una herramienta valiosa para clarificar requisitos confusos. Pueden actuar de una manera similar a los escenarios. En la investigación se usa para que el usuario entienda mejor qué información necesitan proporcionar.

Las técnicas de captura de requisitos permiten identificar los requisitos del cliente y del producto o sistema, los cuales para mayor información se pueden encontrar en el Anexo 2 y 3. Los requisitos del producto o del sistema se derivaron en 66 requisitos funcionales que son lo que el sistema debe hacer y 23 requisitos no funcionales que son los que definen propiedades y restricciones del sistema. A continuación en la Tabla 3 se muestra el listado de requisitos funcionales capturados.

Tabla 3 Listado de requisitos funcionales

Requisitos funcionales	
RFP_1: Buscar persona	RFP_34: Mostrar categoría investigativa
RFP_2: Crear datos de persona	RFP_35: Crear categoría investigativa
RFP_3: Modificar datos de persona	RFP_36: Modificar categoría investigativa
RFP_4: Mostrar persona	RFP_37: Ver detalles de categoría investigativa
RFP_5: Ver detalle de personas	RFP_38: Mostrar categoría docente
RFP_6: Mostrar tipo de identidad	RFP_39: Crear categoría docente
RFP_7 Ver detalles de tipo de identidad	RFP_40: Modificar categoría docente
RFP_8 Crear tipo de identidad	RFP_41: Ver detalles de categoría docente
RFP_9 Modificar tipo de identidad	RFP_42: Mostrar grado académico
RFP_10 Mostrar color de ojos	RFP_43 Crear grado académico
RFP_11: Crear color de ojos	RFP_44 Modificar grado académico
RFP_12: Modificar color de ojos	RFP_45 Ver detalle de grado académico
RFP_13: Ver detalles de color de ojos	RFP_46 Mostrar nivel escolar
RFP_14: Mostrar color de pelo	RFP_47 Crear nivel escolar
RFP_15: Crear color de pelo	RFP_48 Modificar nivel escolar
RFP_16: Modificar color de pelo	RFP_49 Ver detalles de nivel escolar
RFP_17: Ver detalles de color de pelo	RFP_50 Mostrar huérfano
RFP_18: Mostrar estado civil	RFP_51 Crear huérfano

RFP_19: Crear estado civil	RFP_52 Modificar huérfano
RFP_20: Modificar estado civil	RFP_53 Ver detalles de huérfano
RFP_21: Ver detalles de estado civil	RFP_54 Mostrar configuraciones de fuentes de información
RFP_22: Mostrar ocupación	RFP_55 Crear configuración de fuente de información de base de datos
RFP_23: Crear ocupación	RFP_56 Crear configuración de fuente de información de Excel
RFP_24: Modificar ocupación	RFP_57 Crear configuración de fuente de información de CSV
RFP_25: Ver detalles de ocupación	RFP_58 Crear configuración de fuente de información de LDAP
RFP_26: Mostrar raza	RFP_59 Modificar configuración de fuente de información de base de datos
RFP_27: Crear razas	RFP_60 Modificar configuración de fuente de información de Excel
RFP_28: Modificar razas	RFP_61 Modificar configuración de fuente de información de CSV
RFP_29: Ver detalles de razas	RFP_62 Modificar configuración de fuente de información de LDAP
RFP_30:Mostrar tipo de discapacidad	RFP_63 Ver detalles de configuración de fuente de información de base de datos
RFP_31: Crear tipo de discapacidad	RFP_64 Ver detalle de configuración de fuente de información de Excel
RFP_32: Modificar tipo de discapacidad	RFP_65 Ver detalle de configuración de fuente de información de CSV
RFP_33: Ver detalles de tipo de discapacidad	RFP_66 Ver detalles de configuración de fuente de información de LDAP

Los requisitos funcionales fueron sometidos a una evaluación para encontrar la complejidad de los mismos. En la Tabla 4 se puede encontrar el resultado de evaluación y la clasificación que le corresponde, esta clasificación que puede ser: alta, media o baja. Los requisitos no funcionales también fueron clasificados en la Tabla 5 se puede encontrar el resultado de esa clasificación.

Tabla 4 Requisitos funcionales

Prioridad	Cantidad
Alta	35

Media	14
Bajo	17
Total	66

Tabla 5 Requisitos no funcionales

Prioridad	Cantidad
Usabilidad	5
Seguridad	3
Eficiencia	2
Soporte	2
Restricciones de diseño	5
Hardware	3
Requisitos para la documentación de usuarios en línea y ayuda del sistema	2
Interfaz	1
Interfaz de hardware	3
Total	26

2.4.2 Especificación de requisitos

La especificación de requisitos muchos autores lo definen como un documento que especifica los requisitos para un sistema o componente. Por lo general se incluyen las funciones requisitos adicionales, los requisitos de interfaz y normas de desarrollo (IEE, 2012).

En la investigación al usar un proceso de desarrollo con enfoque ágil, se definen distintos artefactos, uno de ellos es la plantilla de “Especificación de Requisitos” en la que se describen cada uno de los requisitos identificados luego de sacar los requisitos del producto y del cliente.

El documento de “Especificación de Requisitos” brinda la posibilidad de ver la vista del sistema, así como la descripción del mismo y un catálogo de requisitos que recoge la descripción de los requisitos funcionales y no funcionales.

La tabla para la descripción de los requisitos funcionales definida en el catálogo, está estructurada para recoger toda la información posible del requisito. Dentro de sus campos se encuentran:

- No: contiene el código por el que se conocerá al requisito. La nomenclatura está compuesta por *RFP_número* donde: *RF* significa que es un requisito funcional, *P* hace referencia al nombre del módulo Personal y el *número* corresponde a un número consecutivo que se le asigna a cada requisito.
- Nombre: contiene el nombre que se le asigna al requisito que se está describiendo.
- Descripción: contiene la descripción del requisito.

- Complejidad: contiene la complejidad que se le asigna a cada funcionalidad, dividiéndose estas en 3 (Alta, Media y Baja).
- Prioridad para el cliente: contiene la prioridad para el cliente, donde las mismas toman los mismos valores que la complejidad.
- Prototipo: contiene el prototipo de interfaz generado en la herramienta del pencil.
- Campos: contiene el nombre de los campos que se muestran en el prototipo de interfaz.
- Tipos de datos: contiene el tipo de dato que le corresponde a cada campo en la base de datos.
- Reglas o Restricciones: cada campo tiene reglas que se deben cumplir a la hora de implementarlos, este campo contiene esas reglas y restricciones.
- Observaciones: contiene todas las observaciones que tiene el requisito funcional tales como mensajes, precondiciones, entre otras.

En la Tabla 6 se muestra un ejemplo de la especificación del requisito *Buscar persona* que corresponde a una de las funcionalidades más importantes del módulo la cual tiene una complejidad *Alta* por lo que es arquitectónicamente significativa. El resto de las descripciones de los requisitos funcionales se pueden encontrar en el Anexo 4.

Tabla 6 Especificación de requisito: RFP_1_Buscar Persona

No	Nombre	Descripción	Complejidad	Prioridad para el cliente
RFP_1	Buscar persona	<p>Permite buscar una o varias personas en el sistema a través de varios criterios de búsqueda.</p> <p>Para buscar por criterios simples se puede utilizar el nombre, apellidos, carnet de identidad, usuario y solapín.</p> <p>Además se pueden utilizar criterios de búsqueda avanzadas como: nacionalidad, ciudadanía, sexo, color de piel, color de ojos, cargo, área, provincia, municipio.</p> <p>Para realizar esta búsqueda se selecciona dentro de la agrupación funcional Personal la funcionalidad personas. Mostrándose en la pantalla el buscador con sus criterios de búsqueda una vez que se llenan todos los datos aparece un listado con los resultados encontrados.</p>	Alta	Alta
Prototipo				

Campos	Tipos de datos	Reglas o restricciones
Criterio	varchar	Admite caracteres alfanuméricos.
Filtros de búsqueda	varchar	Permite seleccionar un filtro de búsqueda mostrando otro campo para realizar la búsqueda.
Sexo	varchar	Permite seleccionar el sexo que puede ser Masculino o Femenino.
Ciudadanía	varchar	Permite seleccionar una ciudadanía.
Nacionalidad	varchar	Permite seleccionar una nacionalidad.
Color de ojos	varchar	Permite seleccionar un color de ojos definido.
Color de piel	varchar	Permite seleccionar un color de piel que se encuentre definido.
Cargo	varchar	Permite seleccionar los cargos existentes en el sistema.
Área	varchar	Permite seleccionar las áreas existentes en el sistema.
Provincia	varchar	Permite seleccionar una provincia.
Municipio	varchar	Permite seleccionar un municipio.
Observaciones	<ol style="list-style-type: none"> 1. Si el resultado de la búsqueda muestra el listado vacío, se debe a que no existe una persona con esos datos en el sistema. 2. La búsqueda retornará todas las coincidencias con el valor encontrado. 3. Los filtros de búsquedas de sexo, color de piel, color de ojo, área, 	

		cargo, provincia y municipio mostrarán los valores que estén creados en su configuración.
--	--	---

Los requisitos no funcionales dentro de la plantilla de requisitos son descritos según el tipo que corresponde. Son identificados por un código donde la nomenclatura está compuesta por *RNFP_número* donde: *RNF* significa que es un requisito funcional, *P* hace referencia al nombre del módulo Personal y el *número* corresponde a un número consecutivo que se le asigna a cada requisito. Para la construcción del módulo se han especificado varios requisitos no funcionales de interfaz e interfaz de hardware que corresponde con la siguiente descripción:

RNFP_23 Interfaz Web: la interfaz debe ser sencilla con colores suaves a la vista y sin cúmulo de imágenes u objetos que distraigan al cliente del objetivo.

RNFP_24 La comunicación entre el servidor WEB y la base de datos se lleva a través del protocolo TCP/IP.

RNFP_25 La comunicación entre el cliente y el servidor de aplicaciones se lleva a través del protocolo de conexión segura HTTPS.

RNFP_26 La comunicación entre el servidor y el directorio activo se hará mediante el protocolo LDAP.

La descripción de los demás requisitos no funcionales se puede encontrar en el Anexo 5.

2.5 Estilos y patrones arquitectónicos

Se puede definir un estilo arquitectónico, como un conjunto de reglas de diseño que identifican las clases de componentes y conectores que se pueden utilizar para componer un sistema o subsistema, junto con las restricciones locales o globales de la forma en que la composición se lleva a cabo (DÍAZ DE MERA, 2011).

Dentro de los diferentes grupos de estilos existentes se encuentran los estilos jerárquicos. Para la estructuración de la arquitectura del sistema se tomó el estilo de cliente/servidor. Este consiste en varios clientes distribuidos en diferentes nodos, conectados en red a uno o varios nodos servidores, donde el servidor puede atender a varios clientes a la vez. En los nodos clientes se encuentran la presentación de usuario y en los nodos servidores la lógica del negocio. Este estilo trae como ventaja que se pueden simplificar las actualizaciones y se puede dar mantenimiento al sistema.

Por otro lado, para conformar la arquitectura del sistema también están los patrones arquitectónicos que expresan un problema recurrente, de diseño muy específico, y presentan una solución para él, desde el punto de vista del contexto en el que se presenta. Por lo que, los patrones arquitectónicos expresan esquemas de organización estructural fundamentales para los sistemas de *software*. Estos se ocupan de cuestiones que están más cerca del diseño, la práctica, la implementación, el proceso, el

refinamiento y el código. Los patrones arquitectónicos, se han materializado con referencia a lenguajes y paradigmas también específicos de desarrollo.

En la investigación al utilizar el marco de trabajo GUUD que está compuesto por CodeIgniter y JQuery, se utiliza el patrón arquitectónico Modelo-Vista-Controlador, debido a que CodeIgniter usa este acercamiento para lograr la flexibilidad, reusabilidad y escalabilidad.

Con el uso de este patrón se separa la lógica y la presentación. El patrón Modelo-Vista-Controlador se divide en tres partes fundamentales:

El modelo: representa la estructura de datos. Típicamente sus clases de modelo contendrán funciones que lo ayudarán a recuperar, insertar y actualizar información en la base de datos.

La vista: es la información que es presentada al usuario, normalmente, será una página Web, pero en CodeIgniter, una vista también puede ser un fragmento de una página como un encabezado o un pie de página.

El controlador sirve como un intermediario entre el Modelo, la Vista y cualquier otro recurso necesario para procesar la petición HTTPS y generar una página Web.

Una vez descrito el funcionamiento del patrón Modelo-Vista-Controlador se pasará a explicar como queda evidenciada su utilización dentro del marco de trabajo sobre el cual se desarrolla la solución. El marco de trabajo cuenta, en su interior, con carpetas definidas para cada uno de los subsistemas (Ver Figura 3) que forman parte del *software* a desarrollar. Al mismo tiempo, cada uno de estos subsistemas, teniendo en cuenta que el marco de trabajo cuenta con la característica de modularidad, define carpetas para cada uno de sus módulos dentro de las cuales queda evidenciado el uso del patrón de la siguiente manera: la carpeta *views* donde serán almacenadas las páginas o fragmentos de páginas web que mostrarán la información al usuario,

La carpeta *controllers* donde se encuentran los archivos que son los encargados de procesar la información y cargar cualquier recurso necesario para procesar la solicitud HTTP, la carpeta *models* donde se encuentran los archivos que contienen una serie de funciones que permiten devolver, insertar y actualizar la información de la base de datos y por último la carpeta *libraries* en la cual están los archivos que sirven de intermediarios entre el controlador y el modelo, además de ser usadas para obtener información de los restantes módulos. Esta última como una adaptación realizada para disminuir la cantidad de responsabilidades en el controlador.

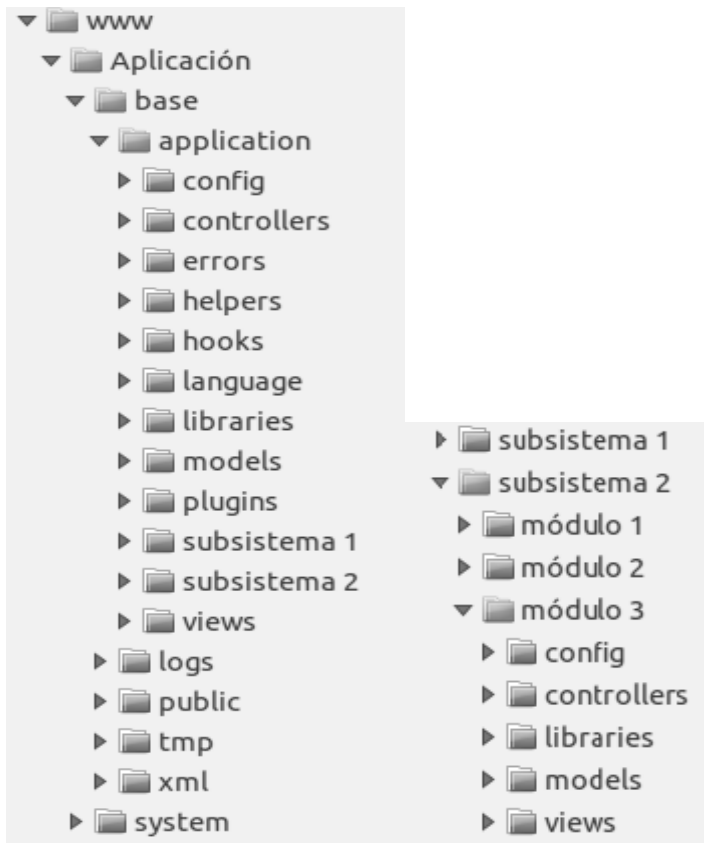


Figura 3 MVC en GUUD.

2.6 Patrones de diseño

Según Craig Larman, en su libro *“UML y Patrones: Introducción al análisis y diseño orientado a objetos”* un patrón es una pareja de problema/solución con un nombre y que es aplicable a otros contextos, con una sugerencia sobre la manera de usarlo en situaciones nuevas (LARMAN, 1999).

Los patrones no se proponen descubrir ni expresar nuevos principios de la ingeniería del *software*. Todo lo contrario, intentan: codificar el conocimiento, las expresiones y los principios ya existentes (LARMAN, 1999).

Como se menciona anteriormente a un patrón se le asigna un nombre, teniendo esto como ventaja que apoya el agrupamiento y la incorporación del concepto al sistema cognitivo y a la memoria, además facilita la comunicación. Dentro de los patrones que se utilizan en el diseño están dos grupos fundamentales los patrones conocidos como GRASP y GOF, teniendo cada una de estas agrupaciones varios patrones que permiten que se apliquen en la solución de un problema.

2.6.1 Patrones GRASP

GRASP (acrónimo de General Responsibility Assignment Software Patterns, en español: Patrones Generales de Software para Asignar Responsabilidades), nombre elegido para indicar la importancia

de captar (en inglés: *grasping*) estos principios, si se quiere diseñar eficazmente el *software* orientado a objetos (LARMAN, 1999).

Para la realización del sistema se utilizaron dentro de los patrones GRASP los patrones: Experto, Creador, Alta Cohesión, Bajo Acoplamiento y Controlador. A continuación se dará una breve descripción de estos patrones:

Experto: con este patrón se le asigna una responsabilidad a la clase que cuenta con la información necesaria para cumplirla. Aplicando este patrón, las librerías serán los expertos, debido a que, son las que cuentan con la información necesaria para cumplir responsabilidades sobre los elementos de negocio. Este patrón brinda beneficios como la conservación del encapsulamiento y el soporte de un bajo acoplamiento y una baja cohesión.

Creador: este patrón guía la asignación de responsabilidades relacionadas con la creación de objetos. En la propuesta de solución la clase *loader* que es el objeto load de las clases controladoras va a cumplir la función de creador, debido a que, es la encargada de cargar los elementos del marco de trabajo, dígame, librerías y modelos. Con el patrón creador se le da soporte a un bajo acoplamiento.

Bajo acoplamiento: este patrón estimula asignar una responsabilidad a una clase, de modo que su colaboración no incremente tanto el acoplamiento con otras clases, al nivel que produzca los resultados negativos propios de un alto acoplamiento. El CodeIgniter tiene este patrón nivelado, pues permite el uso de los componentes de forma individual.

Alta cohesión: este patrón mejora la calidad y facilidad del diseño, genera un bajo acoplamiento y promueve la reutilización. En la propuesta de solución se evidencia cuando entre los componentes hay dependencia.

Controlador: este patrón ofrece una guía para tomar decisiones apropiadas en la elección de los controladores de eventos. Su utilización propicia que las operaciones del sistema se manejen en la capa de dominio de los objetos, y no en la de presentación. En el módulo de Personal las clases controladoras serán las que se encargan de obtener los datos, enviarlos a las librerías y a las vistas actuando así como el patrón controlador.

2.6.2 Patrones GOF

Los patrones GOF constituyen un catálogo de 23 patrones de diseño básicos publicados en el libro *Design Patterns: Elements of Reusable Object-Oriented Software*. GOF significa *Gang of Four* (Banda de los Cuatro), forma en que se conoce al grupo de los 4 autores (GANMA, et all, 2009).

Algunos de los patrones GOF empleados en la implementación del marco de CodeIgniter son *Singleton* (en español: Instancia Única), *Observer* (en español: Observador) y *Mediator* (en español: Mediador) los que se describen a continuación (GANMA, et all, 2009).

Singleton: El empleo de este patrón permite asegurar que una clase posee solamente una instancia, proporcionando un punto de acceso global al objeto instanciado (MESTRAS, 2004). En la investigación se evidencia en todas las clases controladoras que son instancias únicas.

Mediator: Como su nombre lo dice el uso de este patrón permite encontrar un mediador, en la propuesta de solución aplicando este patrón las librerías funcionan como mediadoras entre las clases controladoras y los modelos o acceso a datos.

Observer: En la clase *loader* que es el objeto load de las clases controladoras es donde se evidencia este patrón, esta clase es la que se encarga de cargar los elementos del marco de trabajo dígame, librerías, modelos y se encarga de actualizar la controladora instanciada.

2.7 Estándares de diseño

El módulo Personal al integrarse al Sistema de Gestión Universitaria es necesario que cumpla con las mismas pautas y estándares y diseño que posee el sistema. Trayendo como principal ventaja la uniformidad en la estructura de todas las páginas Web, definiéndose un mapa de navegación y una taxonomía general para todo el módulo.

La vista de gestión de procesos estará dividida en las 11 áreas tales como: subprocesos horizontales, nombre de la aplicación, buscador, líneas de procesos, módulos, nombre de usuario, traza, línea de progresos, contexto, menú de módulos y pie de página. La selección de todas estas áreas va a validar los requisitos no funcionales de usabilidad. Debido a que, el usuario que interactúe con el sistema no tiene que acceder a tantos niveles para realizar la acción deseada.

2.7.1 Pauta cromática

Otro aspecto a tener en cuenta para el diseño son las pautas cromáticas a utilizar. Dentro de los requisitos no funcionales que se identificaron están los de interfaz, donde se especificó que los colores que se usarían para realizar el producto serían colores suaves a la vista del usuario. El Sistema de Gestión Universitaria propone un conjunto de colores que cumplen con estas condiciones, por lo que, para la realización de la propuesta de solución se adoptará las pautas cromáticas definidas en la Figura 4.



Figura 4 Pauta cromática

2.7.2 Componentes

Cumpliendo con los estándares definidos, se han seleccionado para la construcción del módulo varios componentes que van a facilitar su implementación y la uniformidad de todas las funcionalidades.

Dentro de los componentes se encuentran:

- Text Box: componente que crea un control de entrada de una sola línea.
- Text Área: componente que crea un control de entrada de varias líneas.
- Combo Box Simple: componente que contiene una lista de opciones con menús desplegables que permiten elegir una de las múltiples opciones que posee.
- Check Box: componente para listar opciones, permitiendo la elección de más de un componente.
- Picture Grid: componente que permite la creación desde columnas con simples celdas de texto, hasta columnas con texto con formato, imágenes y dato específico. La diferencia radica en que en sus filas el primer elemento es una foto.

2.7.3 Mensajes

Además de definir los componentes que estarán presentes en la investigación se definen los tipos de mensajes con los que contará el módulo. Dentro de los 3 tipos de mensajes definidos se encuentran: mensajes de información, mensajes de error y mensajes de confirmación. A continuación se describen cada uno de ellos:

Los mensajes de información para el cliente se utilizan cuando se crea un elemento o se modifica, dentro de estos mensajes están:

- El elemento ha sido creado satisfactoriamente.
- El elemento ha sido modificado satisfactoriamente.

Los mensajes de error se utilizan para mostrar al usuario cuando ha realizado una acción incorrecta. Estos mensajes pueden salir encima del componente o en forma de ventana. Dentro de los mensajes están:

En forma de ventana:

- El elemento ya existe.

Encima del componente:

- Campo requerido.
- Ha excedido el número de letras permitidas para una palabra.
- Entre al menos n caracteres.
- No más de 200 caracteres.
- Extensión no aceptada.

Los mensajes de confirmación son los que se utilizan cuando es necesario asegurarse que el usuario desea realizar una acción deseada, por ejemplo: cuando se va a cancelar una acción, es necesario asegurarse que eso es lo que desea. Dentro de los mensajes están:

- ¿Está seguro de realizar la acción?

2.8 Modelo de despliegue

En el libro *El Proceso Unificado de Desarrollo* se define al modelo de despliegue como: un modelo que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Se utiliza como entrada fundamental en las actividades de diseño e implementación debido a que la distribución del sistema tiene una influencia principal en su diseño (JABCONSON, et al, 2004).

En el modelo de despliegue cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar. Los nodos poseen relaciones que representan medios de comunicación entre ellos (JABCONSON, et al, 2004).

El módulo de Personal se ejecutará sobre un nodo servidor Web que tendrá contacto con: un nodo de base de datos, un nodo que representa un servidor de directorio activo, un nodo que representa el servidor de PhotoStore y un cierto número de nodos clientes. En la Tabla 7 se muestra la comunicación que existe entre cada nodo del modelo de despliegue y en la Figura 5 se muestra como quedó conformado el modelo.

Tabla 7 Nodos y conexiones del modelo de despliegue

Nodos	Conectores/Protocolos
Servidor Web- Servidor Base de Datos	TCP/IP
Servidor Web- Cliente	HTTPS
Servidor Web-Servidor de Aplicaciones	LDAP
Servidor Web- <i>PhotoStore</i>	TCP/IP

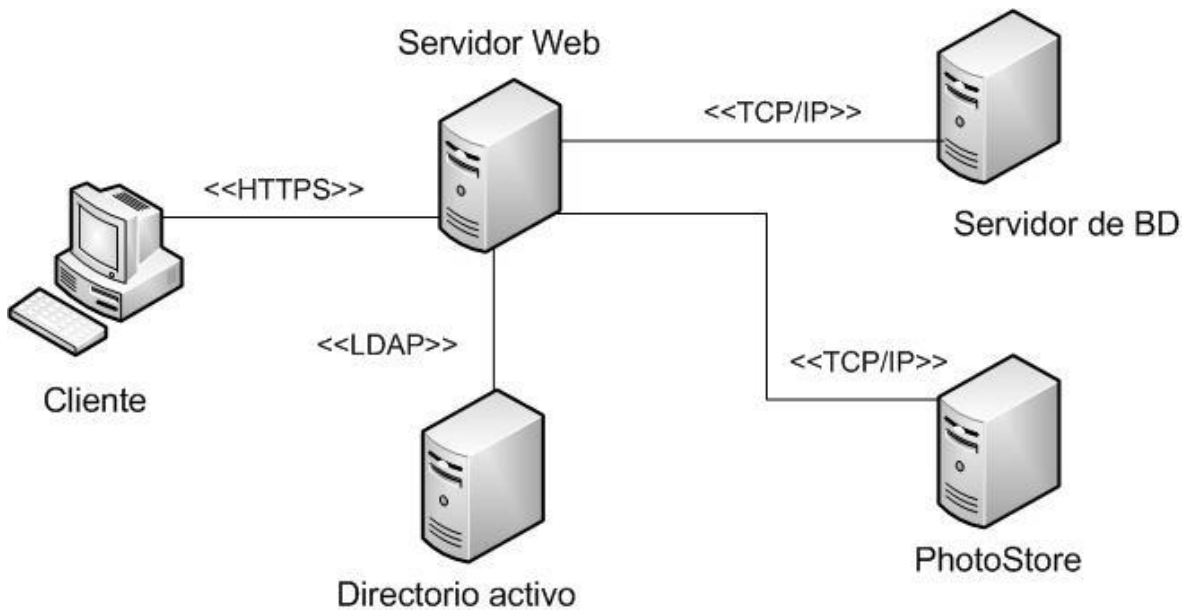


Figura 5 Modelo de despliegue para el módulo de Personal

- **PC Cliente:** su función es acceder al módulo e interactuar con el mismo según sus necesidades.
- **Servidor Web:** en este nodo es donde descansa la capa de presentación del sistema, la cual es accedida por las máquinas clientes a través de un navegador Web. Contiene además, toda la funcionalidad del sistema.
- **Servidor de Base de Datos:** es el encargado de almacenar toda la información generada del sistema.
- **Servidor de PhotoStore:** es el encargado de almacenar todas las fotos de las personas que se encuentran en la universidad.
- **Directorio activo:** es el encargado de almacenar toda la información referente al usuario y las contraseñas de las personas.
- **<<HTTPS>>:** el Protocolo Seguro de Transferencia de Hipertexto es un protocolo de red basado en HTTP por lo que está orientado a transacciones sin estado, es decir, que no guarda ninguna información sobre conexiones anteriores y sigue el esquema petición-respuesta entre un cliente y un servidor. La principal diferencia entre ellos es que este está destinado a la transferencia segura de datos de hipertexto, en otras palabras, es la versión segura de HTTP.
- **<<TCP/IP>>:** la familia de protocolos de Internet es un conjunto de protocolos de red en la que se basa Internet y que permite la transmisión de datos entre redes de computadoras. Denominada en muchas ocasiones conjunto de protocolos TCP/IP en referencia a los dos protocolos más importantes que la componen: Protocolo de Control de Transmisión (*TCP*) y

Protocolo de Internet (*IP*), que fueron los dos primeros en definirse, y que son los más utilizados de la familia.

- <<LDAP>>: acrónimo de *Lightweight Directory Access Protocol*, en español: Protocolo Ligero de Acceso a Directorios. LDAP es un conjunto de protocolos abiertos usados para acceder a información guardada centralmente a través de la red. La mayor ventaja de LDAP es que se puede consolidar información para toda una organización dentro de un repositorio central.

2.9 Modelo de datos

Toda la información que se recopila de las personas debe estar almacenada. En la investigación se realizó el modelo de datos precisamente para permitir salvaguardar la información. Este modelo fue generado en el Visual Paradigm, el mismo se puede encontrar en el Anexo 7. Por el gran tamaño de este modelo se hizo necesario dividirlo en partes.

En el modelo de datos para la gestión de la información de las personas la tabla principal es la tabla *tb_dpersona*, en esa tabla se guardará la información básica de las mismas, además, contará con el identificador del tipo de identidad, donde la misma posibilita conocer si la persona es extranjera, cubana o de las FAR. Para la gestión de las fuentes de configuración está estructurada por varias tablas que posibilitan crear las configuraciones, en donde se guarden los datos para realizar el mapeo de la información según la clasificación.

Para la creación del modelo se tuvo en cuenta los estándares de nomenclaturas propuestos por el grupo de desarrollo del departamento de Gestión Universitaria. En el modelo el esquema está compuesto por el nombre *sq_gestion_personal* donde *sq* representa el tipo. Las tablas de entidades o datos están compuestas por la nomenclatura *tb_d+<texto>*. Las tablas de los nomencladores se pueden identificar por la nomenclatura *tb_n+<texto>*. Las tablas de relación se identifican por la nomenclatura *tb_r+<texto>*. Las llaves primarias tienen la nomenclatura *id_+<texto>* y las llaves primarias se identifican por la nomenclatura *<tabla>_<campo>_fkey*.

Además de las nomenclaturas, se definió utilizar el patrón de base de datos que se describe a continuación:

Llaves subrogadas:

Este patrón es muy utilizado pues se decide generar una llave primaria única para cada entidad en vez de usar un atributo identificador en el contexto dado. Normalmente se usa enteros en columnas *identity* o *GUID* (acrónimo de: *Global Unique Identifier*, en español: Identificador Único Global) que están demostrados que no se repiten o con una probabilidad extremadamente baja. Permite que las tablas sean más fáciles de consultar por el identificador dado que se conoce el mismo tipo de todos en cada tabla (BLAHA, 2010).

El modelo propuesto anteriormente permite resolver el problema de la duplicación de los datos de las personas en las tablas, siendo el mismo uno de los problemas por lo que surge la investigación. Con los estándares adoptados se posibilita que el modelo quede organizado, para que las personas que necesiten trabajar con el mismo tengan un entendimiento a la hora de realizar algún cambio o consultar directamente algún dato.

El patrón de base de datos planteado erradica en gran medida los errores de duplicación en la información de las personas. Gracias a la facilidad que brinda el patrón de crear un identificador único, se formuló una estrategia para crear el id de las personas donde el mismo va a estar compuesto por el carné de identidad, el número de serie del carné de identidad, el primer nombre, el primer apellido y el segundo apellido.

2.10 Conclusiones parciales

En este capítulo quedaron establecidos de forma clara los principales conceptos del dominio, así como las principales funcionalidades, se propuso la arquitectura a utilizar y el modelo de datos de la propuesta.

Por lo que, se puede concluir que:

- Las funcionalidades descritas en la propuesta de solución posibilitará que al implementarse las mismas la información de las personas van a ser configurables y van a posibilitar una mejor administración.
- La arquitectura y estándares propuestos posibilitarán que se cuente con los componentes y elementos necesarios para realizar la implementación.

Capítulo 3 Implementación y validación de la solución informática

3.1 Introducción

En el proceso de desarrollo de *software* luego de especificar y diseñar el producto se pasa a la implementación, los programadores se apoyan en los artefactos generados y presentados en el capítulo anterior para realizar esta tarea. Luego de culminado este paso, se procede a realizar el proceso de pruebas que no deja de ser importante en el desarrollo de un sistema, debido a que permite conocer la calidad del mismo y evaluarla luego de ser sometido a condiciones especiales.

En el presente capítulo se incluyen los temas de estándares, técnicas de codificación y tratamiento de errores. Además, se incluye la estrategia de prueba seleccionada para probar la propuesta de solución y los artefactos que se generaron.

3.2 Estándares de codificación

Un estándar según la Real Academia Española es: un modelo, norma, patrón o referencia. Además, es una especificación que regula la realización de ciertos procesos o la fabricación de componentes para garantizar la interoperabilidad²⁰.

Para el Sistema de Gestión Universitaria se estandarizó al igual que el diseño, el código, permitiendo de esa forma que exista legibilidad y que el mismo esté organizado para su mantenimiento. A continuación se especifican varios de los estándares de código que se asumen para la construcción del módulo de Personal.

3.2.1 Identación, llaves de apertura y cierre, tamaño de líneas

Se define usar una indentación sin tabulaciones, con un equivalente a 4 pasos. El uso de las llaves “{ }” será en una nueva línea. Para mantener la legibilidad del código, la longitud de las líneas debe ser aproximadamente de 75-80 caracteres.

3.2.2 Conversión de nomenclaturas

Constantes: Las constantes siempre deben ser en mayúsculas, con caracteres de subrayado “_” para separar palabras en caso de nombres compuestos.

Clases: Las clases siempre comienzan con mayúscula, en caso de nombre compuesto las palabras se separan con el carácter subrayado “_” y el resto en minúscula.

²⁰ *Interoperabilidad: condición necesaria para que los usuarios (humanos o mecánicos) tengan un acceso completo a la información disponible.*

3.2.3 Estructuras de control

Se incluye *if*, *for*, *foreach*, *while*, *switch*, entre las estructuras de control y en los paréntesis debe de existir un espacio. Se recomienda utilizar siempre llaves de apertura y cierre, incluso en situaciones en las que técnicamente son opcionales. Esto aumenta la legibilidad y disminuye la probabilidad de errores lógicos.

3.3 Técnicas de programación

Cuando se crea un producto el mismo debe tener la flexibilidad suficiente para ser modificable en el momento que se requiera. Estos deben ser claros, simples, con el fin de poder ser leídos e interpretados de forma fácil. Para lograr este objetivo se debe asumir en la programación técnicas que permitan la estandarización (PRESSMAN, et all, 2005).

Es ahí donde entran a desempeñar un papel muy importante las técnicas de programación, en las últimas décadas con el desarrollo de la Informática es muy común escuchar que los diversos lenguajes y programas que se crean están orientados a objetos y que soportan la programación modular (PRESSMAN, et all, 2005). En los siguientes epígrafes se explicarán estas dos técnicas que son adoptadas para la creación del módulo.

3.3.1 Programación modular

La programación modular es uno de los métodos para el diseño más flexible y de mayor *performance* para la productividad de un programa. En este tipo de programación el programa es dividido en módulos, cada uno de los cuales realiza una tarea específica, codificándose independientemente de otros módulos. Cada uno de estos es analizado, codificados y puestos a punto por separado. (BONANATA, 2003)

Se propone utilizar esta técnica de programación debido a la estructura que tiene el Sistema de Gestión Universitaria, en la que se dividen en módulos existiendo un módulo común para ellos posibilitando que la seguridad de los módulos sea independiente.

Además, esta técnica posibilita que haya diferentes programadores trabajando a la vez sobre el producto. También un módulo puede ser codificado sin afectar a los demás, incluso sin alterar su función principal.

3.3.2 Programación orientada a objetos (POO)

En los últimos años la frase “orientado a objetos”, se ha vuelto muy popular, escuchándose a cada momento frases como: “sistemas operativos orientados a objetos”, “lenguajes orientado a objetos” y “programación orientado a objetos”. Dentro de los conceptos generales más utilizados en el modelo orientado a objeto se encuentran: la abstracción, la encapsulación y modularidad. Y con respecto a la

programación son: los objetos, las clases, los métodos, el envío y recepción de mensajes, la herencia y el polimorfismo (BONANATA, 2003).

En la POO encapsular significa que se reúne y controla todo el grupo resultante en un conjunto y no de forma individual. La abstracción es un término externo al objeto, que controla la forma en que es visto por los demás. La herencia se define como una jerarquía de clases derivadas y la relación entre estas, donde se comparte la estructura y el comportamiento de una o más clases consideradas como clases padres. El polimorfismo constituye la definición de múltiples clases con funcionalidades diferentes, pero con métodos o propiedades denominados de forma idéntica

3.4 Tratamiento de errores

En la construcción de un *software* se debe tener en cuenta el tratamiento de errores que no es más que la acción de identificar, localizar, analizar y eliminar los errores que se producen en este proceso. A menudo, cuando se termina un producto, este no tiene un correcto funcionamiento, debido a que, a la hora de entrar los datos en el sistema no hay estabilidad, produciendo errores que atentan contra la calidad.

El tratamiento de errores que debe poseer los sistemas automatizados debe estar encaminados tanto a los errores que se producen a la hora de los usuarios introducir datos, como los errores que puedan ser generados por el comportamiento incorrecto de los componentes internos, por ejemplo: las consultas de base de datos.

En el módulo Personal, los datos que introduce el usuario son validados en los formularios del JQuery mediante Java Script, mostrando mensajes de error encima de los componentes que indican al usuario cuáles datos están incorrectos o cuales datos son obligatorios.

En cuanto al tratamiento de errores en los componentes internos, se implementa en las controladoras las validaciones y cuando se detecta una posible excepción se usan las definidas en un archivo XML.

3.5 Validación de la solución propuesta

Cuando se construye un *software* la calidad no se puede probar, sino que se va construyendo a lo largo del desarrollo del mismo. Cuanto más errores se encuentren en las revisiones, más se podrá asegurar la calidad de la propuesta de solución.

En la investigación al utilizar un proceso de desarrollo con enfoque ágil basado en el nivel 2 de CMMI se deben aplicar validaciones a los requisitos y realizar pruebas durante el ciclo de vida del producto. En los siguientes epígrafes se podrá encontrar las validaciones que se le hicieron a los requisitos y la estrategia de prueba a utilizar.

3.5.1 Validación de requisitos

La validación de requisitos se define como: un proceso o una reunión durante la cual los requisitos de *software* son presentados a clientes, gerentes u otras partes interesadas para hacer comentarios o aprobación (IEE, 2012).

Dentro de los parámetros a comprobar en la validación están:

- La validez: donde no basta con preguntar a un solo usuario, todos los potenciales usuarios deben tener puntos de vista distintos y necesitar otros requisitos.
- La consistencia: no debe haber contradicciones entre unos requisitos y otros.
- La completitud: deben estar todos los requisitos.
- El realismo: se pueden implementar con la tecnología actual.
- La verificabilidad: tiene que existir alguna forma de comprobar que cada requisito se cumple.

En el proceso de desarrollo se definen varios documentos para la validación de requisitos que permiten identificar los criterios para validar los requisitos del cliente y del producto. Se generan los documentos “*Criterios para validar requisitos del cliente*” y “*Criterios para validar requisitos del producto*” permitiendo así, conocer si el requisito queda aprobado después de evaluar todos los criterios definidos.

En la Tabla 8 se puede encontrar las preguntas que se realizan para validar los requisitos en el artefacto de “*Criterios para validar requisitos del cliente*” y en la Tabla 9 se encuentran las preguntas que se realizan para validar los requisitos en el artefacto “*Criterios para validar requisitos del producto*”.

Tabla 8 Criterios para validar requisitos del cliente

Criterios
¿El proveedor del REQ es un proveedor válido?
¿El REQ está identificado como único?
¿El REQ es modificable?
¿El REQ no es ambiguo?
¿El REQ está completo?
¿El REQ es congruente con otros REQ relacionados?
¿El REQ puede ser implementado?
¿El REQ puede ser probado?
¿El resultado de la evaluación de impacto es positivo?
¿El REQ está correcto?
¿El REQ es traceable?

Tabla 9 Criterios para validar requisitos del producto

Criterios
¿Están identificados los elementos de entrada?
¿Están identificados los elementos de salida?
¿El requisito es dado por el superior determinado en el organigrama del proyecto?
¿El requisito no es ambiguo?
¿Es técnicamente factible?
¿Puede ser verificado?
¿Está correcto?
¿El resultado de la evaluación de impacto es positivo?
¿El requisito es traceable?

Con la aplicación de los criterios para validar requisitos del cliente y del producto se obtuvo el 100 % de los requisitos aprobados. Estos requisitos fueron clasificados en 4 requisitos del cliente y 8 requisitos del producto.

Además de validar los requisitos mencionados anteriormente, se realizaron revisiones a los requisitos funcionales y no funcionales mediante la técnica de revisión de requisitos donde se realizó reuniones entre varios miembros del proyecto con el fin de revisar el documento de *“Especificación de requisitos”*. Este proceso abarcó los aspectos que se mencionan a continuación:

- Verificaciones de validez: los requisitos deben cumplir con las necesidades del cliente.
- Verificaciones de consistencia: los requisitos no deben contradecirse en las especificaciones escritas, no deben existir restricciones o descripciones opuestas a las reglas definidas.
- Verificaciones de completitud: los requisitos deben incluir todas las funcionalidades propuestas por el cliente, satisfacer de manera general todas las necesidades acordadas.

Como resultado se identificaron inconsistencias en las especificaciones como:

- Opciones en el área de íconos flotantes innecesarios.
- Falta de concordancia entre la complejidad de la especificación y la que existe en el artefacto *“Evaluación de requisitos”*.
- Componentes innecesarios en los prototipos de interfaz.
- No se registran datos necesarios para el cliente.

Todas estas inconsistencias fueron erradicadas permitiendo que se asegure estos elementos en la calidad del *software*.

3.5.2 Estrategia de prueba

Según Pressman una estrategia de prueba del *software*: integra los métodos de diseño de caso de pruebas en una serie bien planeada de pasos que desembocará en la eficaz construcción del *software*. La estrategia proporciona un mapa que describe los pasos, además de cuánto esfuerzo, tiempo y recursos consumirán (PRESSMAN, et all, 2005).

En la investigación se proponen utilizar algunas de las estrategias definidas por Pressman en su libro *Ingeniería de software: Un enfoque práctico*, en donde se pretende que el producto se pruebe desde la parte más pequeña del *software* hasta la más grande en forma de espiral usando los niveles, tipos, técnicas y herramientas de prueba (PRESSMAN, et all, 2005).

Uno de los niveles de prueba a utilizar es la prueba de unidad con los métodos de caja negra con la técnica de casos de prueba. Otra estrategia que se propone es la de utilizar pruebas de integración mediante la generación de casos de prueba.

Pruebas de unidad

Las pruebas de unidad se centran en la verificación de los elementos más pequeños del *software* que se puedan probar (PRESSMAN, et all, 2005). Se prueba la interfaz del módulo para asegurar que la información fluye de forma adecuada hacia y desde la unidad de programa que está siendo probada. Además, se examinan las estructuras de datos locales para asegurar que los datos que se mantienen temporalmente conservan su integridad mediante los pasos de ejecución del algoritmo.

La interfaz del sistema puede ser probada mediante el método de **caja negra**, también denominada prueba de comportamiento, esta se centra en los requisitos funcionales del *software*.

Las estructuras de datos pueden ser probadas mediante el método de **caja blanca** o estructural, que se basan en un minucioso examen de los detalles procedimentales del código a evaluar, por lo que es necesario conocer la lógica del programa.

Método de caja negra

El método de caja negra se refiere a las pruebas que se llevan a cabo sobre la interfaz del *software*. Una prueba de caja negra examina algunos aspectos del modelo fundamental del sistema sin tener mucho en cuenta la lógica interna del *software* (PRESSMAN, et all, 2005).

En la propuesta de solución se aplicó el método de caja negra con la técnica de **casos de prueba** en donde se intenta demoler el sistema con todas las posibles combinaciones de datos y su resultado radica en tener una alta probabilidad de mostrar un error no descubierto hasta el momento.

Para la aplicación de esta técnica se generaron los artefactos de “*Diseño de caso de prueba basado en requisitos*”. Por cada requisito funcional se creó un documento en donde se recogen todos los datos necesarios para probar la interfaz. Dentro de esos datos se encuentran:

- Condiciones de ejecución: contiene como su nombre lo indica todas las condiciones que deben existir para que el caso de prueba sea ejecutado sin dificultad.
- SC_ Nombre del requisito a probar: corresponde al nombre que se le da al requisito en el documento de “*Especificación de requisitos*”.
- Escenario: en este campo se nombran los diferentes escenarios a probar donde cada uno estará compuesto por las siglas *EC* y una numeración consecutiva *1.1... 1.n + el nombre del escenario*.
- Descripción: corresponde como dice su nombre a la descripción del escenario a probar.
- Variable 1...n: este campo va a estar compuesto por dos partes: la primera podrá tomar los valores *V*, *I* y *NA* (válido, inválido y no aplica respectivamente), la segunda estará conformada por un ejemplo según el escenario que se pretende realizar.
- Respuesta del sistema: se escribe el resultado que se espera al realizar la prueba.
- Flujo central: corresponde a los pasos a desarrollar para probar la funcionalidad que se indicó.

En las Tablas 10 y 11 se puede encontrar un ejemplo de diseño de casos de prueba para el requisito RFP_1 Buscar persona, debido a la extensión del diseño de caso de prueba se hizo necesario dividir la tabla. El resto de los diseños de casos de prueba se pueden encontrar en el Anexo 8.

Tabla 10 DCP_Buscar Personas. Parte 1

Condiciones de ejecución: El usuario debe estar autenticado como administrador.							
SC Buscar personas							
Escenario	Descripción	Filtro de búsqueda simple	Filtro de búsqueda con avanzada	Sexo	Nacionalidad	Ciudadanía	Color de piel
EC 1.1 Buscar datos correctamente.	Mediante este escenario se mostrará al usuario el listado de personas existentes en el sistema.	V El usuario dará un espacio	NA	NA	NA	NA	NA
		V José Antonio	NA	NA	NA	NA	NA
		V Pérez	NA	NA	NA	NA	NA
		V 8,8E+10	NA	NA	NA	NA	NA
		V jabo	NA	NA	NA	NA	NA
		V Pepe a	NA	NA	NA	NA	NA
		V	NA	NA	NA	NA	NA

		E12152					
		V A	V Se selecciona el filtro sexo	V Femenino	NA	NA	NA
		V lisb carr	V Se selecciona el filtro nacionalidad	NA	V Cubana	NA	NA
		V pepe E122365	V Se selecciona el filtro ciudadanía	NA	NA	V Española	NA
		V lola cab 89526323568	V Se selecciona el filtro color de piel	NA	NA	NA	V Blanca
		V ljbabella E20203	V Se selecciona el filtro color de ojos	NA	NA	NA	NA
		V Pérez Paz Gabriel	V Se selecciona el filtro cargo	NA	NA	NA	NA
		V Esperanza E	V Se selecciona el filtro área	NA	NA	NA	NA
		V Mará Caridad	V Se selecciona el filtro provincia	NA	NA	NA	NA
		V 88112536520 E56201 pjperéz	V Se selecciona el filtro municipio.	NA	NA	NA	NA
		V dfcastro	V Se seleccionan todos los filtros	V Masculino	V Cubana	V Cubana	V Blanca
EC 1.4 No existen elementos creados.	Mediante este escenario en caso de que no exista creado ningún elemento se muestra un listado vacío.	V El usuario dará un espacio.		NA	NA	NA	NA

Tabla 11 DCP_ Buscar personas. Parte 2

Cargo	Área	Provincia	Municipio	Respuesta del sistema	Flujo central

NA	NA	NA	NA	El sistema muestra el listado de todas las personas del sistema.	El usuario una vez autenticado en el sistema. Selecciona el módulo Personal y dentro de la Agrupación Funcional "Personal" selecciona la funcionalidad Personas llena los campos de búsqueda y presiona el botón "Buscar"
NA	NA	NA	NA	El sistema muestra un listado con todas las personas que tengan coincidencia con el elemento introducido.	
NA	NA	NA	NA		
NA	NA	NA	NA		
NA	NA	NA	NA		
NA	NA	NA	NA		
NA	NA	NA	NA		
NA	NA	NA	NA	El sistema mostrará un listado de personas luego de buscar coincidencias entre los filtros simples y los filtros de búsqueda avanzada.	
NA	NA	NA	NA		
NA	NA	NA	NA		
NA	NA	NA	NA		
NA	NA	NA	NA		
V Director	NA	NA	NA		
NA	V Facultad 1				
NA	NA	V Matanzas	NA		
NA	NA	NA	V Matanzas		
V Auxiliar	V Docente 3	V Ciego de Ávila	V Morón		
NA	NA	NA	NA	El sistema muestra el listado vacío.	El usuario una vez autenticado en el sistema. Selecciona el módulo Personal y dentro de la Agrupación Funcional "Personal" selecciona la funcionalidad Personas. Llena los campos de búsqueda y presiona el botón "Buscar"

Para ejecutar los diseños de casos de prueba generados se definió un plan de iteraciones. Durante la ejecución de esta técnica se detectaron no conformidades, que se clasificaron en no conformidades de interfaz, de validación y de ortografía. En la tabla 12 se puede consultar la cantidad de no conformidades encontradas. Es necesario aclarar que las no conformidades encontradas se resolvieron en un 100%.

Tabla 12 No conformidades encontradas por iteraciones

Iteraciones	No conformidades de ortografía	No conformidades de interfaz	No conformidades de validación
Iteración 1	6	15	20
Iteración 2	4	10	15
Iteración 3	3	7	4
Iteración 4	0	0	0

Pruebas de integración

Las pruebas de integración es una técnica sistemática para construir la estructura del programa mientras que, al mismo tiempo, se llevan a cabo pruebas para detectar errores asociados con la interacción. El objetivo es coger los módulos probados mediante la prueba de unidad y construir una estructura de programa que está de acuerdo con lo que dicta el diseño (PRESSMAN, et all, 2005).

Como en la investigación la solución informática que se propone es un módulo, es necesario probar la integración entre las agrupaciones funcionales, además su integración con los módulos de los que se obtiene información.

Técnica de casos de prueba

En la investigación se utilizan los casos de prueba para probar los elementos mencionados anteriormente. Para cada elemento se creó un caso de prueba en donde se recogen todos los datos necesarios. Dentro de estos datos se encuentran:

- Número de caso de prueba: corresponde al nombre que se le da al caso de prueba está identificado por las siglas *Int_n*, donde *Int* corresponde a la identificación de las pruebas de integración y *n* es un número consecutivo que se le asigna.
- Agrupación a integrar o módulo a integrar: corresponde al nombre del elemento a probar. Depende de lo que se quiera probar.
- Condiciones de ejecución: hace referencia a los pasos a seguir antes de realizar la prueba.
- Descripción de la prueba: hace referencia como lo dice su nombre a la descripción de la prueba.

- Entradas/Pasos de ejecución: hace referencia a los pasos que se deben realizar para ejecutar la prueba.
- Resultado esperado: como lo indica su nombre es el resultado que se espera cuando se ejecutan los pasos.
- Evaluación: hace referencia al resultado de la prueba luego de ser ejecutada, donde el resultado puede ser satisfactorio o insatisfactorio.

A continuación se muestra en las Tablas 13, 14 y 15 ejemplos de casos de prueba de integración. La primera para integrar la funcionalidad de “Tipo de discapacidad” de la agrupación funcional “Configuración” a la agrupación funcional “Personal”. La segunda para integrar el módulo Seguridad al módulo Personas y el tercer diseño para integrar el módulo de Estructura y Composición. Los demás casos de prueba generados se pueden encontrar en el Anexo 9.

Tabla 13 Caso de prueba de integración Int_1

Número de caso de prueba: Int_1	
Funcionalidad a integrar:	Tipo de discapacidad
Condiciones de ejecución:	Se debe crear el tipo de discapacidad sensorial.
Descripción de la prueba:	Comprobar que cuando se cree los datos de una persona y se llenen los rasgos fenotípicos el tipo de discapacidad sensorial se liste.
Entrada/Pasos de ejecución:	El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Llena los datos para crear una nueva persona, seleccionando que es discapacitado y escogiendo el tipo de discapacidad sensorial.
Resultado esperado:	Se lista el tipo de discapacidad sensorial.
Evaluación:	Satisfactoria

Tabla 14 Caso de prueba de integración Int_14

Número de caso de prueba: Int_14	
Módulo a integrar:	Seguridad
Condiciones de ejecución:	Deben existir dominios creados en el módulo de seguridad.
Descripción de la prueba:	Comprobar que cuando aparezca la interfaz para modificar los datos de una persona se pueda escoger un dominio.
Entrada/Pasos de ejecución:	El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de modificar en el área de iconos internos. Se modifican los datos. Cuando aparezca la interfaz de datos de entidad se debe seleccionar un dominio.

Resultado esperado: Se listaron los dominios existentes.

Evaluación: Satisfactoria

Tabla 15 Caso de prueba de integración Int_15

Número de caso de prueba: Int_15

Módulo a integrar: Estructura y composición

Deben existir responsabilidades creadas en el módulo de estructura y

Condiciones de ejecución: composición.

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para modificar los datos de una persona se pueda escoger un cargo.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de modificar en el área de iconos internos. Se modifican los datos. Cuando aparezca la interfaz de datos de entidad se debe seleccionar un cargo.

Resultado esperado: Se listaron los cargos existentes.

Evaluación: Satisfactoria

Los casos de prueba se fueron aplicando de forma ascendente, donde se probaron las integraciones entre las agrupaciones funcionales y luego las integraciones con los módulos del núcleo.

Luego de haber aplicado los casos de prueba se pudo identificar que en el 100 % de casos el resultado fue satisfactorio. Esta propuesta de prueba no abarcó la integración con los subsistemas del Sistema de Gestión Universitaria.

3.6 Conclusiones parciales

Tras haber realizado este capítulo se puede concluir que:

- Los estándares de código propuestos facilitan el trabajo del programador a la hora de implementar el módulo.
- La propuesta de solución estará orientada a objetos debido al uso de las técnicas de programación y herramientas utilizadas.
- La estrategia de prueba propuesta posibilita la detección de errores tanto del código como de la interfaz del módulo, contribuyendo esto a que se eleve la calidad de la propuesta de solución.

Conclusiones

Una vez finalizada la implementación del módulo de Personal, se pudo comprobar el cumplimiento de los objetivos trazados concluyendo que:

- Un estudio detallado de los principales conceptos identificados en el modelo de dominio evidenció que la gestión de personal no se efectuaba de la mejor manera, debido a las incoherencias de las áreas, la poca validación, entre otras dificultades.
- Los sistemas automatizados que fueron estudiados sólo asumen algunos requerimientos necesarios, por lo que, se evidenció la necesidad de desarrollar una solución informática que cumpliera con todas las exigencias de las áreas de procesos universitarios.
- Las herramientas propuestas en el entorno de desarrollo fueron estudiadas y adoptadas para ser empleadas en la solución propuesta.
- Con la arquitectura definida se logró alcanzar un nivel de organización y relación entre las capas especificadas, así como el cumplimiento de las pautas necesarias para el desarrollo de la solución propuesta.
- Con la implementación del sistema quedaron evidenciadas las buenas prácticas de programación a través de un código limpio y legible, ajustado a los estándares especificados y manteniendo un correcto manejo de errores a través de las diferentes vías.

Después de haber realizado las pruebas de software pertinentes se logró una aplicación que cumple con todos los requisitos, dichas pruebas sirvieron para identificar las no conformidades que fueron corregidas en cada iteración.

Recomendaciones

- Con el objetivo de elevar la calidad de la solución informática planteada, se recomienda que se apliquen las pruebas del sistema para probar el *stress* y la resistencia del producto.

•

Bibliografía referenciada

- [1] (ALLES, 2011) ALLES, Martha. Diccionario de términos: Recursos Humanos. [En línea] 1a. ed. Buenos Aires: Granica, 2011. 488p. [Fecha de consulta: 12 de enero de 2012]. Disponible en: <http://books.google.com.cu/books?id=nKnL2ne1QaEC&pg=PA45&dq=Administraci%C3%B3n+de+Personal+en+Recursos+Humanos&hl=es&sa=X&ei=t1y1T7CKDYzCgAea3eniDw&ved=0CGQQ6AEwCQ#v=onepage&q=Administraci%C3%B3n%20de%20Personal%20en%20Recursos%20Humanos&f=false> ISBN: 978-950-641-618-8.
- [2] (ASSETS, 2004) ASSETS: Sistema de Gestión Integral. Assets, 2004. [En línea]. [Fecha de consulta: 22 de octubre de 2011]. Disponible en: <http://www.assets.co.cu/clientes.asp?page=18>
- [3] (BOHLANDER, 2008) BOHLANDER, George y SNELL, Scott. Administración de Recursos Humanos. [En línea]. 14a. ed. Abril Vega Orozco, 2008. 25p. [Fecha de consulta: 22 de noviembre de 2011]. Disponible en: http://books.google.com.cu/books?id=m0voFJyc8MAC&printsec=frontcover&dq=Administraci%C3%B3n+de+Recursos+Humanos&hl=es&sa=X&ei=A0a1T_Nk1OqBB8-sqAl&ved=0CD0Q6AEwAQ#v=onepage&q=Administraci%C3%B3n%20de%20Recursos%20Humanos&f=false. ISBN: 970-686-712-0.
- [4] (BARQUERO, et all, 2005) BARQUERO Corrales, Alfredo. Administración de Recursos Humanos. [En línea]. 1a. ed. San José, Costa Rica: Universidad Estatal a distancia, 2005. 188p. [Fecha de consulta: 20 de noviembre de 2011]. Disponible en: http://books.google.com.cu/books?id=mXjYQZHR00kC&printsec=frontcover&dq=Administraci%C3%B3n+de+Recursos+Humanos&hl=es&sa=X&ei=A0a1T_Nk1OqBB8-sqAl&ved=0CFwQ6AEwBg#v=onepage&q=Administraci%C3%B3n%20de%20Recursos%20Humanos&f=false ISBN: 9977-64-330-X.
- [5] (BERNAL, 2009) BERNAL, Yuri. JavaScript: Desarrollo de aplicaciones comerciales en Java. 10 de junio de 2009 [En línea]. [Fecha de consulta: 20 de enero de 2012]. Disponible en: <http://www.slideshare.net/guest044583b/javascript-1562899>
- [6] (BLAHA, 2010) BLAHA, Michael. Patterns of Data Modeling. Emerging Directions in Database Systems and Applications. Edición ilustrada. [s.l.]: Taylor & Francis Group, 2010. 245p.
- [7] (BONANATA, 2003) BONANATA, Maximiliano. Programación y Algoritmos [en línea]. Edición ilustrada. Buenos Aires: MP Ediciones, 2003 [fecha de consulta: 5 Enero 2012]. Disponible en:

- <http://sunshine.prod.uci.cu/search/programaci%C3%B3n%20orientada%20a%20objetos>. ISBN: 9789875261563
- [8] (BOOCH, et all, 2005) BOOCH, Grady, RUMBAUGH, James y JABCONSON, Ivar. The unified modeling language user guide. 2a. [sl]. ed. Addison Wesley Professional, 2005.125p. ISBN: 0-321-26797-4
- [9] (CAMPDERRICH, 2003) CAMPDERRICH Falgueras, Benet. Ingeniería del software. [En línea]. 1a. ed. Barcelona: UOC, 2003. 260p. [Fecha de consulta: 16 de enero de 2012]. Disponible en: http://books.google.com/cu/books?id=tKTpr4Ah88C&pg=PA30&dq=herramientas+case&hl=es&sa=X&ei=jtq3T4GHA8j0gAfO_Py2Cg&ved=0CEEQ6AEwAg#v=onepage&q=herramientas%20case&f=false ISBN: 84-8318-997-6
- [10] (CANOSA, et all, 2009) CANOSA, Mónica, VERA, Judith y GRANADOS, Andrea. Sistema de Gestión de Personal (SIU-PAMPA). 3 de agosto de 2009. [En línea]. [Fecha de consulta: 15 de diciembre de 2011]. Disponible en: http://dgti.uncoma.edu.ar/index.php?option=com_content&view=article&id=12&Itemid=40
- [11] (CARRO, et all, 2009) CARRO, Martha. Entrevista con José Pascual Gumbau Mezquita. Director del Gabinete de Planificación y Prospectiva Tecnológica. Universidad Jaime I de Castellón. *Sociedad de la información*. [En línea]. No 64. 14 de diciembre de 2009. [Fecha de consulta: 15 de enero de 2012]. Disponible en: <http://www.socinfo.es/numeros/809-u-jaume-i-de-castellon-q-contamos-con-un-erp-propio-que-cubre-el-90-de-los-procesosq?format=pdf> ISSN: 1761-2850
- [12] (CHIAVENATO, 1999) CHIAVENATO, Idalberto. Administración de Recursos Humanos. [En línea]. 5a. ed. Santa Fé, Bogotá: McGraw-Hill, 1999. 580p. [Fecha de consulta: 18 de octubre de 2011]. Disponible en: <http://es.scribd.com/doc/11402610/Administracion-de-RRHH-5ta-Edicion-Idalberto-Chiavenato> ISBN: 85-224-2004-1
- [13] (CHIRISSIS, 2003) CHIRISSIS, Mary Beth. CMMI: Guindelines for Process Integration and Product Improvement. 1a. ed. Adisson Wesley, 2003. 663p. ISBN: 0-321-15496-7
- [14] (CODEIGNITER, 2012) CODEIGNITER. Ellis Lab Inc, 2012. [En línea]. [Fecha de consulta: 30 de enero de 2012]. Disponible en: <http://codeigniter.com/>

- [15] (CRESPO, et all, 2010) CRESPO, Lic. Luis J. y OLMEDO, Lic. Gustavo E. Cátedra: Informática II Sistemas SIU. Tesis (TECNICATURA EN GESTIÓN UNIVERSITARIA). Argentina: Universidad Nacional de Formosa, 2010. 5p.
- [16] (DÍAZ DE MERA, 2011) DÍAZ DE MERA Sánchez, María del Prado. Implicaciones del Espaciado Armonizado Europeo de Seguridad y Calidad Industrial en las Metodologías de Gestión de Proyectos Sostenibles. [En línea]. 1a. ed. Grin Verlag, 2011. [Fecha de consulta: 16 de febrero de 2012]. Disponible en: <http://books.google.com/cu/books?id=ROlaMTAx0W8C&pg=PA278&dq=Manifiesto+%C3%A1gil+%C3%A1gil&hl=es&sa=X&ei=RgS4T6mlFsbDgQfg9pXACg&ved=0CDMQ6AEwAA#v=onepage&q=Manifiesto%20%C3%A1gil%20%C3%A1gil&f=false> ISBN: 978-3-656-17327-4
- [17] (DÍAZ, 2005) DÍAZ Duarte, Lic.Daimara. Toma de decisiones: el imperativo diario de la vida en la organización moderna. *ACIMED*. [En línea]. mayo-jun 2005, v.13 n.3. [Fecha de consulta: 20 de octubre 2011]. Disponible en: http://scielo.sld.cu/scielo.php?pid=S1024-94352005000300010&script=sci_arttext&tlng=pt. ISSN 1561-2880.
- [18] (ERP UNIVERSITY, 2012) ERP UNIVERSITY: Primer ERP basado en Software Libre. ERP University. 7de octubre de 2012. [En línea]. [Fecha de consulta: 12 de enero de 2012]. Disponible en: www.erpuniversity.com
- [19] (GANMA, et all, 2009) GANMA, Erich [et all.]. Design Patterns: Elements of Reusable Object-Oriented Software. ed. Addison-Wesley, 2009. 395p. ISBN: 0201633612.
- [20] (GARCÍA, et all, 2007) GARCÍA, Elena y CUEVAS Roque. Principios básicos de Informática. [En línea] 1a. ed. Madrid: Dykinson S. L. 2007. 62p. [Fecha de consulta: 21 de enero de 2011]. Disponible en: <http://books.google.com/cu/books?id=5zZq8abR9YkC&pg=PA62&dq=Lenguajes+de+programaci%C3%B3n&hl=es&sa=X&ei=LyG3T63LGMr8ggfdx8zQCg&ved=0CFgQ6AEwBg#v=onepage&q=Lenguajes%20de%20programaci%C3%B3n&f=false> ISBN: 978-84-9849-098-5.
- [21] (GONZÁLEZ, 2011) GONZÁLEZ, Carlos D. Curso PostgreSQL, SQL avanzado y PHP: Diseño natural, seguro, accesible. 2011. [En línea]. [Fecha de consulta: 10 de enero de 2012]. Disponible en: <http://www.usabilidadweb.com.ar/postgre.php>.
- [22] (GONZÁLEZ, et all, 2007) GONZÁLEZ Hernández, MSc. Delly L., RODRÍGUEZ Orego, MSc. Vero E y FEBLES Rodríguez, Dr. Juan P. La colaboración en proyectos de investigación.

- Desarrollo en Bioinformática. De la dispersión a la integración: Una propuesta de acciones desde al ámbito de los Estudio Sociales en Ciencia y Tecnología. La Habana, 2007. 62p.
- [23] (GUTMANS, et all, 2005) GUTMANS, Andi, SAETTER Stig y RETHANS, Derick. PHP 5 power programming. [En línea] 1a. ed. Prentice Hall PTR, 2005. 689p. [Fecha de consulta: 13 de febrero de 2012]. Disponible en: http://www.phptr.com/content/images/013147149X/downloads/013147149X_book.pdf ISBN: 013147149X.
- [24] (IEE, 2012) IEEE: Advancing Technology for Humanity. IEEE, 2012 [En línea]. [Fecha de consulta: 15 de febrero de 2012]. Disponible en: http://www.ieee.org/education_careers/education/standards/standards_glossary.html.
- [25] (JABCONSON, et al, 2004) JABCONSON Ivar, BOOCH, Grady y RUMBAUGH, James. El proceso unificado de desarrollo de software. 1a. ed. Madrid: Adison Wesley, 2004. 200p. ISBN: 0201571692
- [26] (JQUERY, 2010) JQUERY. The JQuery Project, 2010. [En línea]. [Fecha de consulta: 3 de febrero de 2012]. Disponible en: http://docs.jquery.com/Release:jQuery_1.3.2.
- [27] (JUBE, 2008) JUBE, Web master, 2008. [En línea]. [Fecha de consulta: 15 de enero de 2012]. Disponible en: <http://jube.es/>.
- [28] (LAFOSSE, 2010) LAFOSSE, Jérôme. Strust 2: El marco de trabajo de desarrollo de aplicaciones Java EE. [En línea]. 2a. ed. Barcelona: ENI, abril 2010. 100p. [Fecha de consulta: 25 de enero de 2011]. Disponible en: <http://books.google.com/cu/books?id=96HHRq6g5x8C&pg=PA11&dq=Framework&hl=es&sa=X&ei=0S-3T9mhDISmgweL86GnCc&ved=0CFAQ6AEwBQ#v=onepage&q=Framework&f=false> ISBN: 978-2-7460-5542-1.
- [29] (LAMARCA, 2011) LAMARCA Lapuente, María Jesús. Hipertexto: El nuevo concepto de documento en la cultura de la imagen. 2011. [en línea] [Fecha de consulta: 19 de noviembre de 2011] Disponible en: <http://www.hipertexto.info/documentos/html.htm>.
- [30] (LARMAN, 1999) LARMAN, Craig. UML y Patrones Introducción al análisis y diseño orientado a objetos. 1a. ed. México: Prentice Hall, 1999. ISBN: 970-17-0261-1
- [31] (LÓPEZ, 2006) LÓPEZ Paz, Carlos R. Aplicación del cuadro de mando integral y el data warehouse a la gestión de los Recursos Humanos. *Revista Industrial*. [En línea]. enero-febrero

2006. vol. XXVII. no. 2-3. [Fecha de consulta: 20 de enero de 2012]. Disponible en: <http://es.scribd.com/doc/54529507/25387645-1>
- [32] (MARTÍNEZ, 2008) MARTÍNEZ Rodríguez, Dr. Raúl. "GREHU: UN SISTEMA INTEGRAL PARA GESTIONAR LOS RECURSOS HUMANOS". 6 de noviembre de 2008. [En línea]. [Fecha de consulta: 30 de noviembre de 2011]. Disponible en: www.gestec.disaic.cu/memorias/98/Trabajos/RaulMRguez.doc.
- [33] (MESTRAS, 2004) MESTRAS, J. P. Patrones de diseño orientado a objetos. [En línea] Universidad Complutense Madrid Facultad de Informática UCM, 2004. [Fecha de consulta: 15 de marzo de 2012]. Disponible en: <http://www.fdi.ucm.es/profesor/jpavon/poo/2.14PDOO.pdf>.
- [34] (NETBEANS, 2012) NETBEANS. Oracle corporation. [En línea]. [Fecha de consulta: 20 de diciembre de 2012]. Disponible en: <http://netbeans.org/>.
- [35] (NORIEGA, 2010) NORIEGA, Juan. 22 de marzo de 2010. [En línea]. [Fecha de consulta: 24 de enero de 2012]. Disponible en: <http://noriegatic.blogspot.com/2010/03/postgre-sql.html>.
- [36] (PENCIL PROJECT, 2010) PENCIL PROJECT: Sketching and Prototyping with Firefox. Evolus, 2010 [En línea]. [Fecha de consulta: 20 de diciembre de 2010]. Disponible en: <http://pencil.evolus.vn/en-US/Home.aspx>
- [37] (PRESSMAN, et all, 2005) PRESSMAN, Roger S y MURRIETA Murrieta Jesús E. Ingeniería del software: Un enfoque práctico. 6a. ed. McGraw-hil Interamericana, 2005. 958p. ISBN: 9780072853186.
- [38] (PODESWA, 2005) PODESWA, Howard. Uml For The It Business Analyst. 1a. ed. Course Technology Ptr, 2005. 56p. ISBN: 1598638688.
- [39] (RAMOS, 2007) RAMOS Martín, Alicia. Operaciones con Base de Datos Ofimáticas y Corporativas. [En línea]. 1a. ed. Barcelona: Thompson, 2007. 393p. [Fecha de consulta: 6 de enero de 2012]. Disponible en: <http://books.google.com.cu/books?id=hwNWRHc79PMC&pg=PA1&dq=sistemas+gestores+de+bases+de+datos&hl=es&sa=X&ei=ijW3T7DeEJK-gAeqKnSCq&ved=0CFkQ6AEwCA#v=onepage&q=sistemas%20gestores%20de%20bases%20de%20datos&f=false> ISBN: 978-84-9732-552-3.
- [40] (RODRIGUEZ, 2004) RODRIGUEZ Serrano, Juan C. El modelo de gestión de recursos humanos. [En línea]. 1a. ed. UOC, 2004. 165p. [Fecha de consulta: 30 de noviembre de 2011].

Disponible en:

http://books.google.com.cu/books?id=F5iqQr_Q9dkC&pg=PA2&lpg=PA2&dq=Rodr%C3%ADguez+Serrano,+J.+C.+%28abril+2004%29.+El+modelo+de+gesti%C3%B3n+de+recursos+humanos+%28Primera+edici%C3%B3n+ed.%29.+Barcelona:+UOC.&source=bl&ots=qMvmJAbRbX&sig=2xcqTEEOAclvSRZ0cr1SU-8qXE4&hl=es&sa=X&ei=aFm1T5ruMYPagAeb4Pn6BA&ved=0CEQQ6AEwAA#v=onepage&q&f=false. ISBN: 84-9788-152-4.

- [41] (SOMMERVILLE, et al, 1997) SOMMERVILLE, Ian y SAWYER, Pete. Requirements Engineering. A Good Practice Guide. 1a. ed. John Wiley & Sons, 1997. 391p. ISBN 0471974447.
- [42] (SOMERVILLE, 2005) SOMERVILLE, Ian. Ingeniería del Software. [En línea]. 7a. ed. Madrid: Pearson Education S.A, 2005. 661p. [Fecha de consulta: 20 de diciembre de 2012]. Disponible en:
http://books.google.com.cu/books?id=gQWd49zSut4C&pg=PA80&dq=Herramientas+Case&hl=es&sa=X&ei=uOG3T-_cE8jppgged2PSiCg&ved=0CEQQ6AEwAA#v=onepage&q=Herramientas%20Case&f=false ISBN: 84-7829-074-5
- [43] (SWEBOK, 2004) SWEBOK. Guide to the Software Engineering Body of knowledge. 1a. ed. Bourque, P., & Dupuis, 2004. 202p. ISBN 0-7695-2330-7
- [44] (APACHE,2010) APACHE Software Foundation, 17 de junio de 2010. [En línea]. [Fecha de consulta: 15 de febrero de 2012]. Disponible en:
<http://tiles.apache.org/2.2/framework/tutorial/basic/concepts.html>
- [45] (VISUAL PARADIGM, 2012) VISUAL PARADIGM. UML tool, business process modeler and database designer for software development team. [En línea]. [Fecha de consulta: 15 de enero de 2012] Disponible en: <http://www.visual-paradigm.com>.
- [46] (WERTHER, et al, 2008) WERTHER, William B y DAVIS, Keith. Administración de personal y recursos humanos. 5a. ed. McGraw-Hill, 2008 - 644 p. ISBN: 9701028929.

Bibliografía consultada

1. ÁLVAREZ, Sara. Arquitectura cliente-servidor. [En línea]. [Fecha de consulta: 25 de febrero de 2012]. Disponible en: <http://www.desarrolloweb.com/articulos/arquitectura-cliente-servidor.html>
2. BASS, L.; CLEMENTS, P., *et al.* Software architecture in practice. 2a. ed. Pearson Education, Inc., 2003, Disponible en: http://books.google.com/cu/books?id=mdilu8Kk1WMC&dq=software+architecture+in+practice&printsec=frontcover&source=bn&hl=es&ei=lrG6S7uqJ8P98AaluKGkCA&sa=X&oi=book_result&ct=result&resnum=4&ved=0CCAQ6AEwAw#v=onepage&q=&f=false. ISBN 0-321-15495-9. 3.
3. FERSCO-WEISS, H. The case for improved software design. (Computer-aided software engineering). Lotus, 1989, vol. v5, nº n8, p. p11(14). Disponible en: <http://find.galegroup.com/gps/infomark.do?&contentSet=IAC-Documents&type=retrieve&tabID=T002&prodId=IPS&docId=A7477636&source=gale&srcprod=CDB&userGroupName=ucinf&version=1.0> ISSN 8756-7334.
4. GIBERT GINESTÁ, M. y PÉREZ MORA, O. *Bases de Datos en PostgreSQL*. Disponible en: http://dataprix.com/files/UOC_OpenSource_bases_de_datos_en_PostgreSQL.pdf.
5. GONZÁLEZ Pupo, Eddy Felix y SALGADO Valdivia, Marcel. Implementación del módulo Personal y Secretaría para el subsistema Gestión de Pregrado del Sistema de Gestión Universitaria. Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informática. Ciudad de La Habana, 2010.
6. HERNÁNDEZ, León, R. A., & Coello González, S. El paradigma cuantitativo de la investigación científica. Ciudad de la Habana 2002.
7. HERNÁNDEZ Sampieri, Roberto, FERNÁNDEZ Collado, Carlos y BAPTISTA Lucio, Pilar. Metodología de la investigación. 4a. ed. [sl]. McGraw-Hill, 2006. 882p. ISBN: 970-10-5753-8.
8. ORELLANA León, Mireydis y ALONSO Beatón, Yurislely. Análisis y diseño para los Procesos de Gestión de Personas. Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. Ciudad de La Habana, junio de 2009.
9. PEÑALVER Romero, G. M. *MA-GMPR-UR2 Metodología Ágil para proyectos de software libre*. Tutor: Ing. Meneses Abad, A. Facultad 10. Universidad de las Ciencias Informáticas (UCI), 2008.

10. POSTGRESQL GLOBAL DEVELOPMENT GROUP. *PostgreSQL*: The world's most advanced open source database [En línea]. [Fecha de consulta: 23 de marzo de 2012]. Disponible en: <http://www.postgresql.org/>
11. WORLD WIDE WEB CONSORTIUM. Extensible Markup Language (XML). [En línea]. Disponible en: <http://www.w3.org/XML/>

Anexos

Anexo 1 Modelo de entrevista

MODELO DE ENTREVISTA

Nombre(s):

Apellido(s):

Proyecto al que pertenece:

Cargo que ocupa:

Preguntas:

¿De dónde se obtiene la información de las personas en su negocio?

¿Cuál es la información de las personas que utilizan en su negocio?

¿Qué información de las personas utiliza más?

¿Cuáles son los datos de las personas que generan más información para su negocio?

¿De esos datos cuáles considera sensibles?

¿Considera que la información de las personas que se maneja en su negocio siempre debe estar actualizada?

¿Qué datos de las personas son los que le hace falta y que no se encuentran en la información que se obtiene?

Anexo 2 Requisitos del cliente

No	Requisitos del cliente
1	Gestionar la información nominal de las personas teniendo en cuenta las especificaciones de cada área.
2	Configurar los datos de las personas.
3	Gestionar la configuración de las fuentes de información.
4	Guardar las configuraciones de cada fuente de información en la base de datos para

realizar el volcado de la información.

Anexo 3 Requisitos del producto

No	Requisitos del cliente
1	El administrador una vez autenticado podrá buscar una o varias personas en el sistema.
2	El responsable una vez autenticado podrá incorporar datos de personas en el sistema.
3	El responsable podrá modificar los datos de personas en el sistema.
4	Se permitirá luego de creada la persona, cambiarle los datos de la entidad.
5	El administrador podrá crear una configuración de fuente de información.
6	Cuando se crea la configuración de una fuente se debe acceder a esa fuente para seleccionar los elementos por los que quieren recoger la información.
7	El administrador podrá modificar la configuración deseada.
8	El responsable del grupo de informatización podrá configurar los datos asociados a la información de la persona.

Anexo 4 Especificación de requisitos funcionales

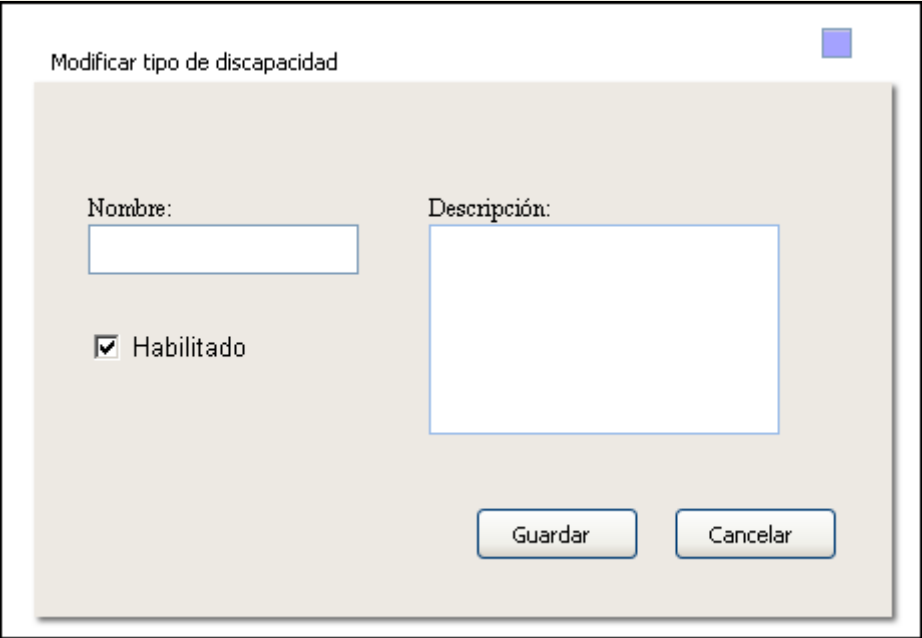
En este anexo solo se hará referencia a cuatro especificaciones de requisitos funcionales. Para buscar más información se puede consultar el expediente del Proyecto Pregrado en donde se encuentran todas las especificaciones de requisitos.

Nº	Nombre	Descripción	Complejidad	Prioridad para cliente
RFP_31	Crear tipo de discapacidad	Permite crear un tipo de discapacidad seleccionando de la agrupación funcional "Configuración" la funcionalidad "Tipo de discapacidad" y en el área de iconos flotantes la opción crear.	Alta	Alta
Prototipo				

	<div style="border: 1px solid black; padding: 10px;"> <p style="text-align: right;">[X]</p> <p>Crear tipo de discapacidad</p> <p>Nombre: <input style="width: 150px;" type="text"/></p> <p>Descripción: <input style="width: 200px; height: 80px;" type="text"/></p> <p><input type="checkbox"/> Habilitado</p> <p style="text-align: center;"> <input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/> </p> </div>		
	Campos	Tipos de Datos	Reglas o Restricciones
	<ul style="list-style-type: none"> • Nombre 	<ul style="list-style-type: none"> • varchar 	<ul style="list-style-type: none"> • Permite la entrada de un total de 50 caracteres donde las palabras no excedan los 30 caracteres. • Permite la entrada de caracteres alfanuméricos y no permite la entrada de caracteres extraños. • La cantidad mínima debe ser de 2 caracteres
	<ul style="list-style-type: none"> • Descripción 	<ul style="list-style-type: none"> • varchar 	<ul style="list-style-type: none"> • Permite la entrada de 0-200 caracteres en total. • Las palabras no deben exceder a los 30 caracteres.

			<ul style="list-style-type: none"> Permite cualquier tipo de caracteres
	<ul style="list-style-type: none"> Habilitado 	<ul style="list-style-type: none"> varchar 	<ul style="list-style-type: none"> Permite seleccionar el estado. Inicialmente va a estar habilitado.
	Observaciones	<ol style="list-style-type: none"> Cuando un elemento se repite el sistema debe mostrar le mensaje de error: El elemento ya existe. Cuando se selecciona la opción Cancelar se muestra el mensaje de error: ¿Está seguro de realizar la acción? Cuando en los campos se introduzcan la cantidad de caracteres requeridos el sistema no debe permitir la entrada de más caracteres. Cuando las palabras excedan a los 30 caracteres el sistema debe mostrar el mensaje de error en rojo encima del componente: Ha excedido el número de letras permitidas para una palabra. Cuando se copien más de 200 caracteres el sistema debe mostrar el mensaje en rojo encima del componente: No más de 200 caracteres. Cuando no se cumpla con la cantidad mínima de caracteres el sistema debe mostrar un mensaje en rojo encima del componente: Entre al menos 2 caracteres. 	

Nº	Nombre	Descripción	Complejidad	Prioridad para cliente
RFP_32	Modificar tipo de discapacidad	Permite modificar un tipo de discapacidad seleccionando de la agrupación funcional Configuración la funcionalidad Tipo de discapacidad y en el área de iconos internos la opción modificar	Alta	Alta
Prototipo				

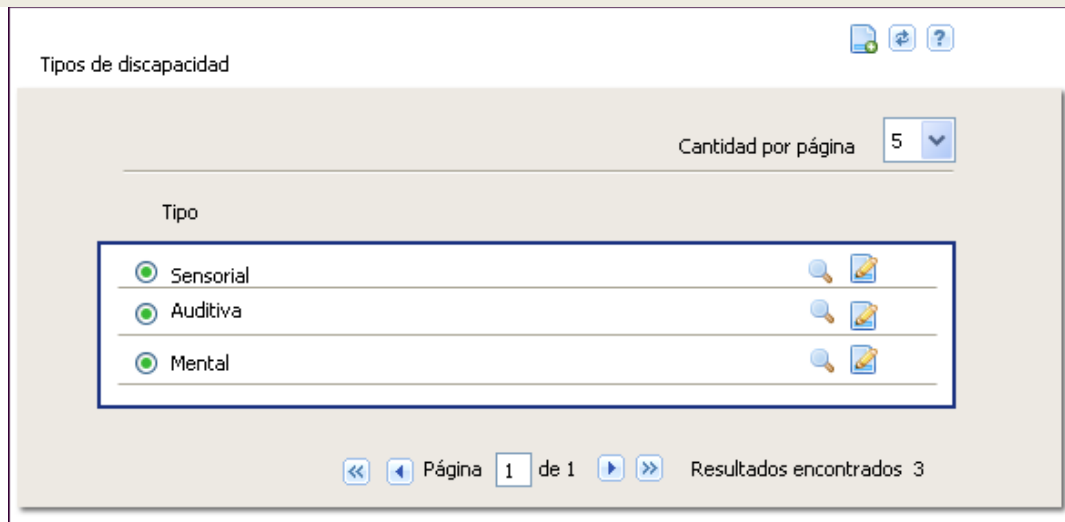
			
	Campos	Tipos de Datos	Reglas o Restricciones
	<ul style="list-style-type: none"> Nombre 	<ul style="list-style-type: none"> varchar 	<ul style="list-style-type: none"> Permite la entrada de un total de 50 caracteres donde las palabras no excedan los 30 caracteres. Permite la entrada de caracteres alfanuméricos y no permite la entrada de caracteres extraños. La cantidad mínima debe ser de 2 caracteres
	<ul style="list-style-type: none"> Descripción 	<ul style="list-style-type: none"> varchar 	<ul style="list-style-type: none"> Permite la entrada de 0-200 caracteres en total. Las palabras no deben exceder a los

			<p>30 caracteres.</p> <ul style="list-style-type: none"> Permite cualquier tipo de caracteres
	<ul style="list-style-type: none"> Habilitado 	<ul style="list-style-type: none"> varchar 	<ul style="list-style-type: none"> Permite seleccionar el estado. Inicialmente va a estar habilitado.
	Observaciones	<p>7. Cuando un elemento se repite el sistema debe mostrar le mensaje de error: El elemento ya existe.</p> <p>8. Cuando se selecciona la opción Cancelar se muestra el mensaje de error: ¿Está seguro de realizar la acción?</p> <p>9. Cuando en los campos se introduzcan la cantidad de caracteres requeridos el sistema no debe permitir la entrada de más caracteres.</p> <p>10. Cuando las palabras excedan a los 30 caracteres el sistema debe mostrar el mensaje de error en rojo encima del componente: Ha excedido el número de letras permitidas para una palabra.</p> <p>11. Cuando se copien más de 200 caracteres el sistema debe mostrar el mensaje en rojo encima del componente: No más de 200 caracteres.</p> <p>12. Cuando no se cumpla con la cantidad mínima de caracteres el sistema debe mostrar un mensaje en rojo encima del componente: Entre al menos 2 caracteres.</p>	

Nº	Nombre	Descripción	Complejidad	Prioridad para cliente
RFP_30	Mostrar tipo de discapacidad.	Permite mostrar el listado de tipos de discapacidad, mostrando el estado en el que se encuentra y el tipo de discapacidad. Brinda en el área de iconos internos la opción de Ver detalles y en el área de iconos flotantes las opciones de	Media	Media

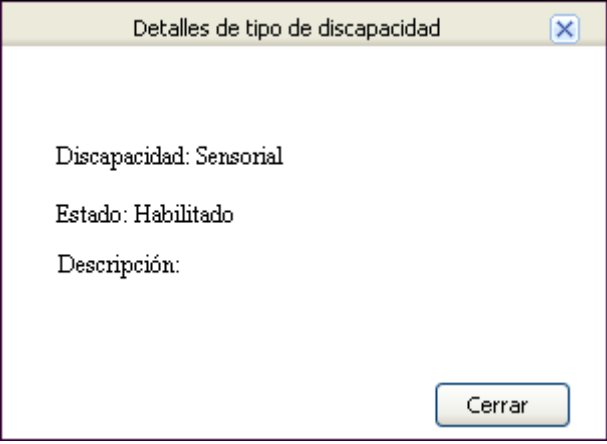
crear, actualizar, exportar y ayuda. Para acceder a esta opción se debe escoger de la agrupación funcional Configuración la funcionalidad Tipo de discapacidad.

Prototipo



Campos	Tipos de Datos	Reglas o Restricciones
•	•	•
Observaciones	<ol style="list-style-type: none"> 1. Los atributos se ordenan alfabéticamente en la lista, ubicando primeramente las habilidades y luego las deshabilitadas. 2. La cantidad de elementos a mostrar en la lista son 5, 10, 15 y 20 según la preferencia del usuario. 	

Nº	Nombre	Descripción	Complejidad	Prioridad para cliente

RFP_33	Ver detalles de tipo de identidad	<p>Permite ver los detalles de los tipos de identidad. Mostrando los datos: Tipos, Descripción y Estado.</p> <p>Para acceder a esta opción se debe escoger de la agrupación funcional Configuración la funcionalidad Tipo de discapacidad y dentro del área de iconos internos la opción Ver detalles.</p>	Baja	Baja
Prototipo				
				
Campos		Tipos de Datos	Reglas o Restricciones	
•		•	•	
Observaciones		<p>1. Al pasar el puntero del mouse sobre el ícono Ver detalles este debe indicar la acción.</p>		

Anexo 5 Especificación de requisitos no funcionales

Requisitos no funcionales	
Usabilidad	
RNFP_1	Facilidad de uso por parte de los usuarios: El sistema debe presentar una interfaz amigable que permita la fácil interacción con el mismo y llegar de manera rápida y efectiva a la información buscada. Debe, además, ser una interfaz de manejo cómodo que posibilite a los usuarios sin experiencia una rápida adaptación.
RNFP_2	Especificación de la terminología utilizada: El sistema debe adaptarse al lenguaje y términos utilizados por los usuarios en la rama abordada con vista a una mayor comprensión por parte del cliente de la herramienta de trabajo.
RNFP_3	Potencialidades de capacitación orientadas a interfaces intuitivas: lo que enaltece la posibilidad de que el usuario aprenda mediante el uso y explotación de la herramienta.
RNFP_4	Menús: El sistema debe presentar una serie de menús tanto laterales como en barra de íconos flotantes que permitan el acceso rápido a la información por parte de los usuarios, aprovechando así las potencialidades de estas estructuras.
RNFP_5	Emplear perfiles de usuario: Diferenciar las interfaces y opciones para los usuarios que accedan al sistema según los diferentes roles que estos tengan dentro del mismo.
Seguridad	
RNFP_6	La seguridad de la base de datos está a nivel de roles, con el fin de mantener la integridad de los datos en función del acceso de cada uno de ellos, trayendo consigo además la protección de la información.
RNFP_7	Políticas de seguridad por usuarios y roles: El sistema debe contar con un grupo de políticas de accesibilidad a las diferentes funcionalidades del mismo en dependencia del nivel de autorización que presente un usuario determinado.
RNFP_8	Registro sistemáticos de incidencias: El sistema debe ser capaz de registrar el accionar del usuario, así como permitir auditorías y exámenes de las trazas tanto en tiempo real como en históricos.
Eficiencia	
RNFP_9	El sistema debe soportar un tiempo de respuesta menor o igual a 5 segundos.
RNFP_10	El sistema debe soportar una conexión simultánea de más de 3000 usuarios.
Soporte	

RNFP_11 El sistema debe brindar como apoyo una Ayuda contextual en la cual se refleja detalladamente la explicación de cada una de las pantallas con sus respectivas funcionalidades.

RNFP_12 Grupo de soporte y asesoría: El sistema contará con un grupo de soporte y asesoría al cliente del producto destinado a brindar asesoría y soporte técnico al mismo.

Restricciones de diseño

RNFP_13 IDE de desarrollo: NetBeans 6.9.

RNFP_14 Sistema Gestor de BD: PostgreSQL 8.4

RNFP_15 Lenguaje de programación: PHP 5.2

RNFP_16 Navegador Web: Mozilla Firefox 6 o superior.

RNFP_17 Marco de trabajo base de desarrollo: GUUD con la integración de CodeIgniter 1.7.2 y JQuery 1.3.2

Hardware

RNFP_18 Para el desarrollo se requiere de una PC Intel Pentium 4 o superior, CPU 3GHZ o superior, 512 MB RAM o superior, 160 GB HDD o superior.

RNFP_19 Para la explotación del cliente se requiere de una PC Pentium 3 o superior, CPU 133 MHZ o superior, 256 RAM mínimo 512 RAM recomendada o superior.

RNFP_20 Para la explotación del servidor se requiere de un CPU Dual Core 2.0 GHZ o superior, memoria RAM de 4 GB (recomendado 6 GB), 250 GB HDD.

Requisitos para la documentación de usuarios en línea y ayuda del sistema

RNFP_21 Manual de usuario: El sistema deberá presentar un manual de usuario, permitiendo con ello un correcto uso de sus funcionalidades y brindarle al usuario una mayor experiencia del trabajo con el mismo.

RNFP_22 Documentación actualizada del grupo de desarrollo: Se precisa que la documentación del sistema esté actualizada en todos los aspectos, fases de trabajo y ciclos de desarrollo del mismo, permitiendo con ello un respaldo tanto ingenieril como legal del desarrollo de dicho sistema.

Interfaz

RNFP_23 Interfaz Web: La interfaz es sencilla con colores suaves a la vista y sin cúmulo de imágenes u objetos que distraigan al cliente del objetivo.

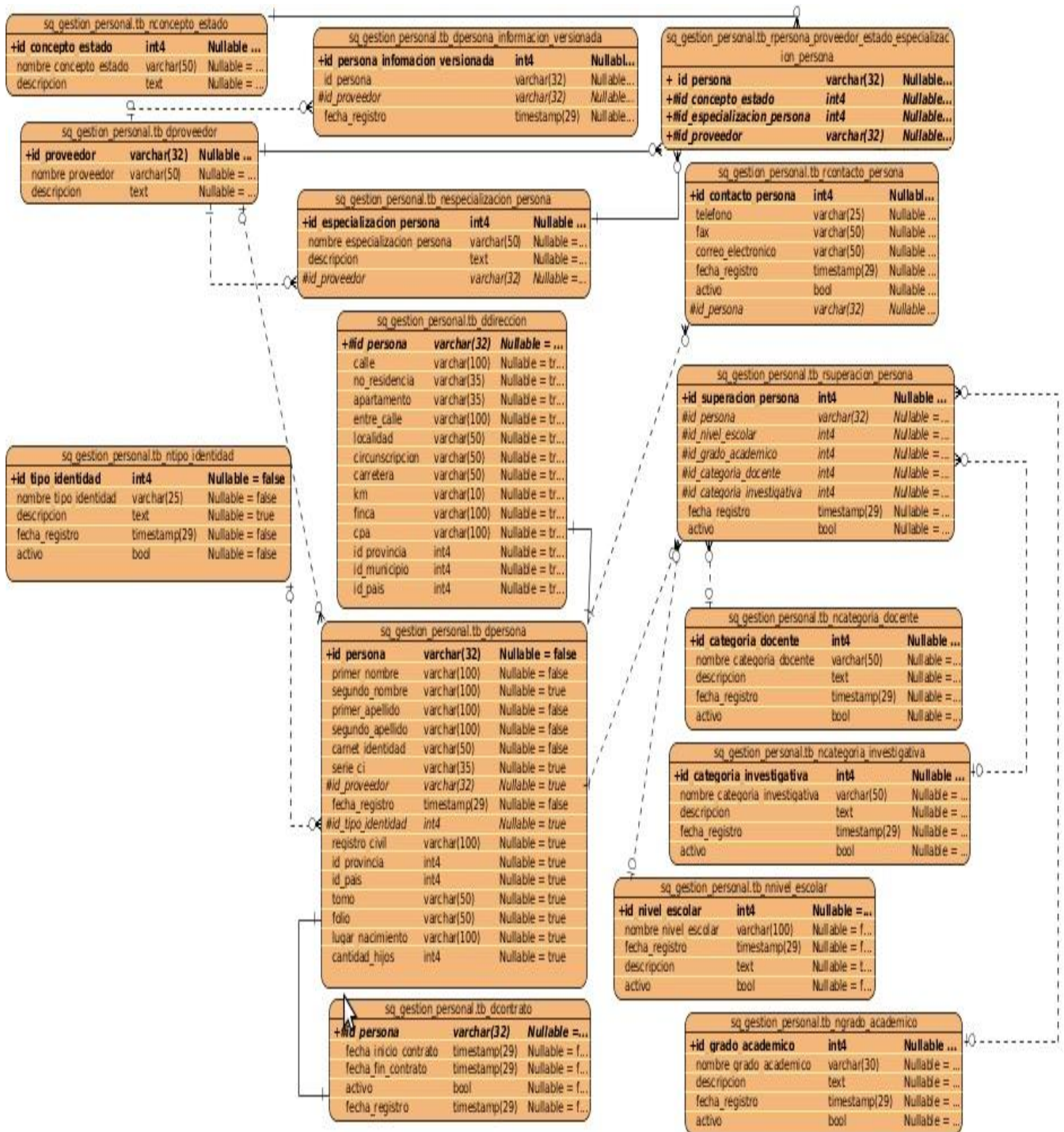
Interfaz de hardware

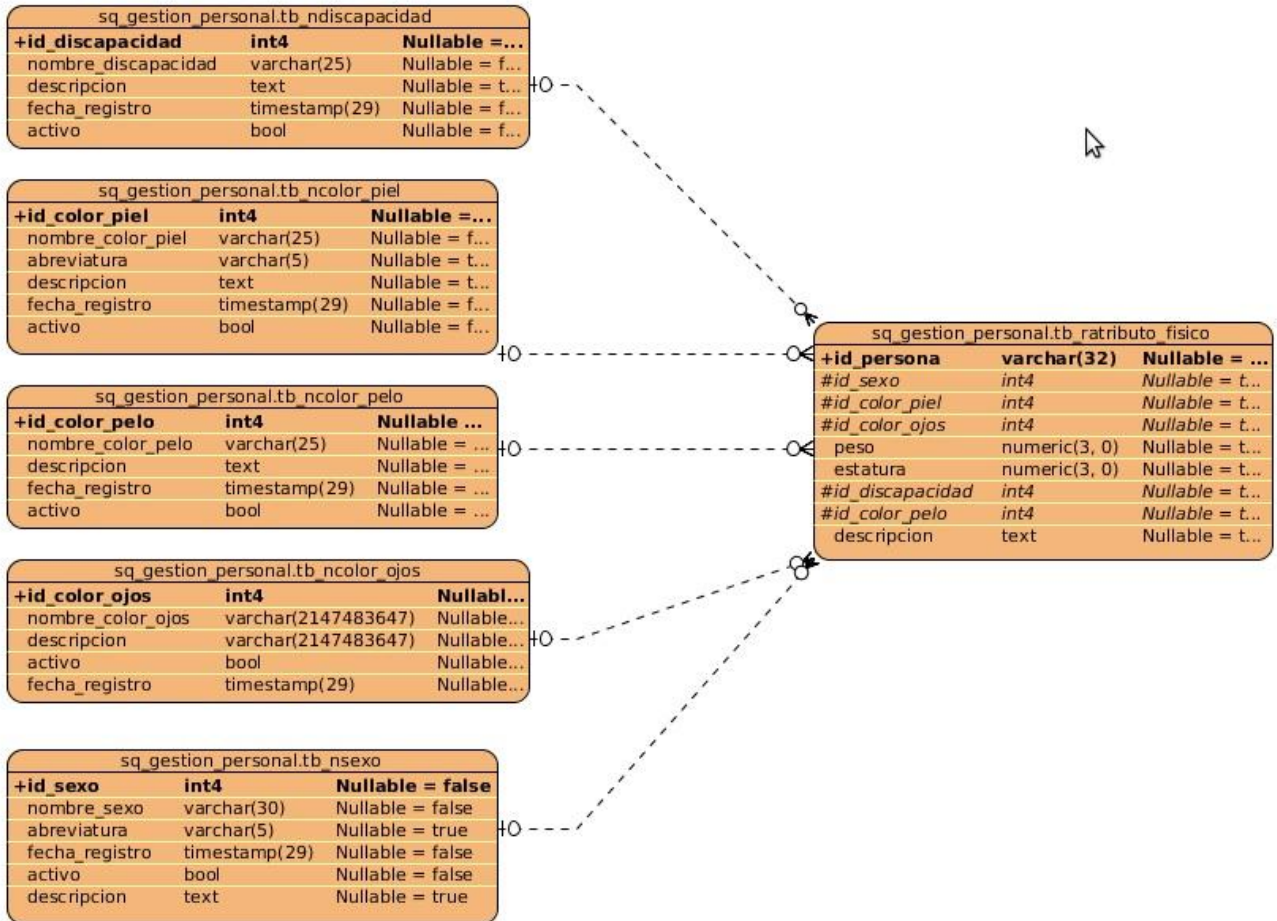
RNFP_24 La comunicación entre el servidor de aplicaciones y la base de datos se lleva a través del protocolo TCP/IP.

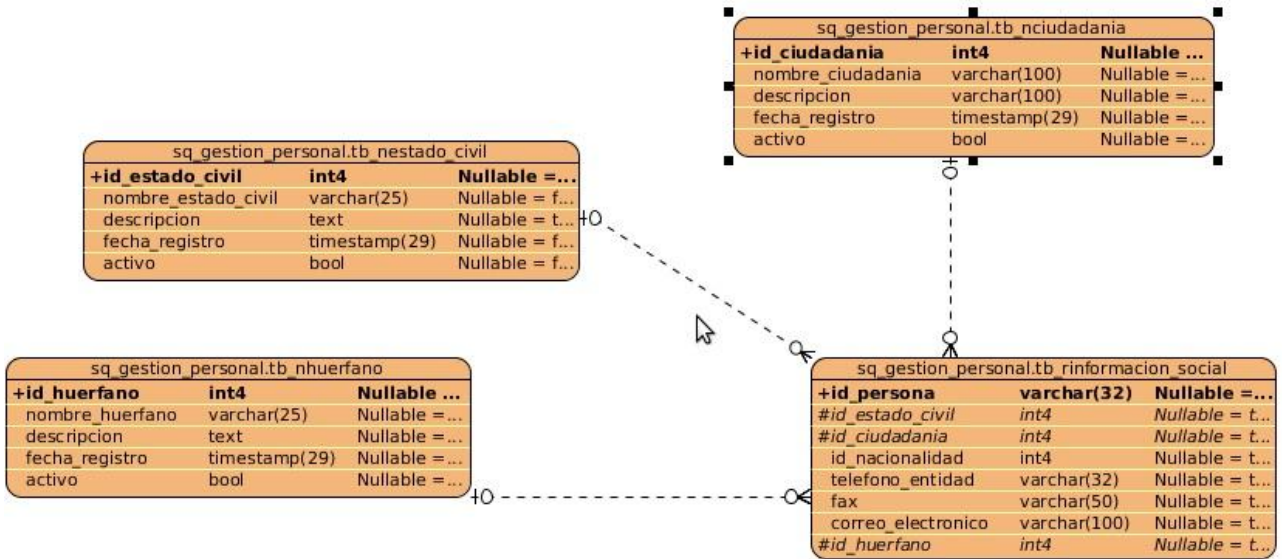
RNFP_25 La comunicación entre el cliente y el servidor de aplicaciones se lleva a través del protocolo de conexión segura HTTPS.

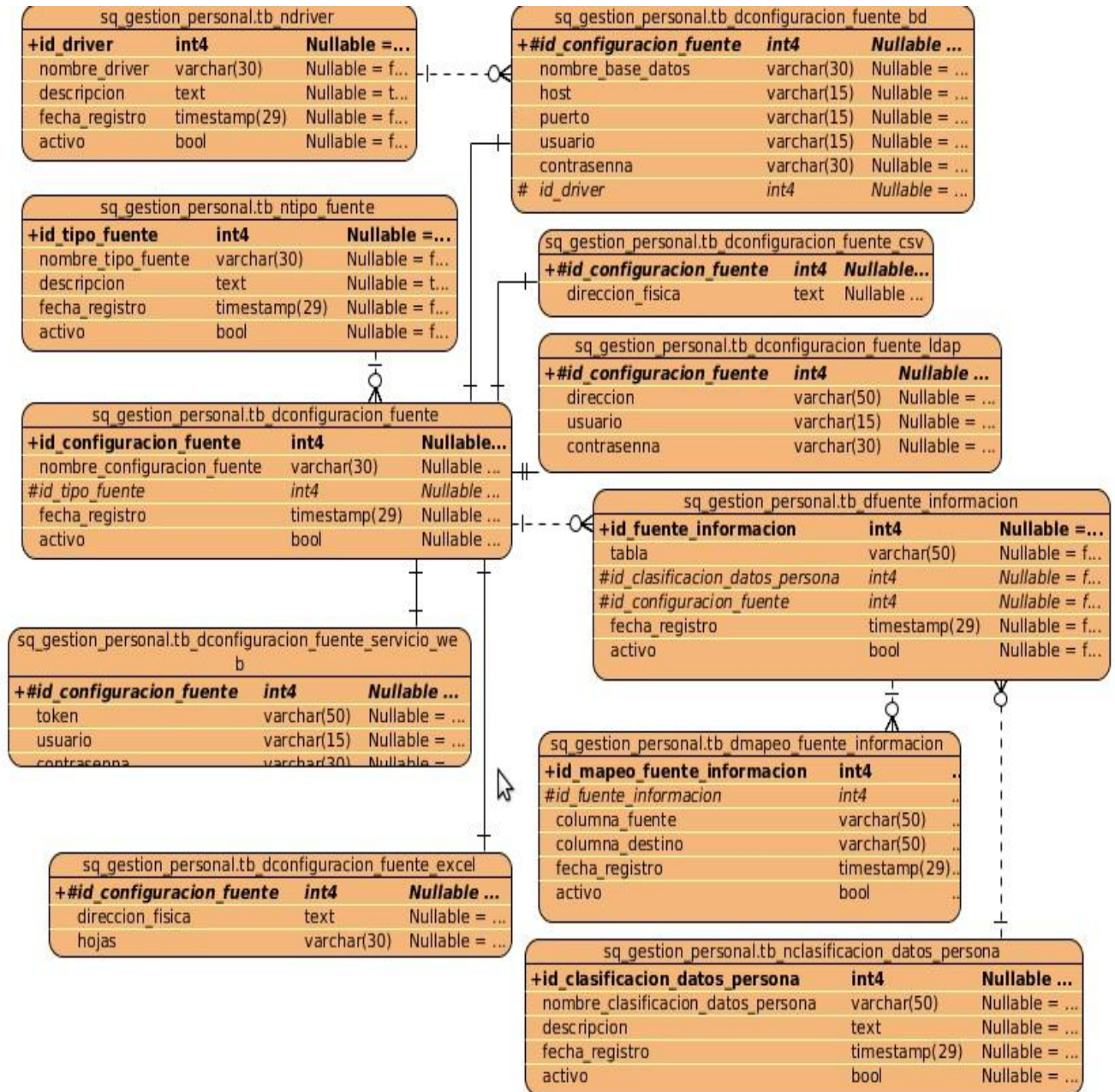
RNFP_26 La comunicación entre el servidor y el directorio activo se hará mediante el protocolo LDAP.

Anexo 7 Modelo de datos









Anexo 8 Diseños de caso de prueba basado en requisitos

En este anexo solo se hará referencia a cuatro diseños de caso de prueba basado en requisitos. Para buscar más información se puede consultar el expediente del Proyecto Pregrado en donde se encuentran todos los diseños.

DCP_ Ver detalles de tipo de discapacidad**Condiciones de ejecución: El usuario debe estar autenticado como administrador.****SC Ver detalles de tipo de discapacidad**

Escenario	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Ver detalles de tipo de discapacidad	Mediante este escenario se muestra toda la información del elemento seleccionado.	El sistema muestra una ventana emergente con los siguientes datos: Nombre Descripción Estado (habilitado/ deshabilitado),	El usuario una vez autenticado en el sistema selecciona la Agrupación Funcional "Configuración" y luego selecciona la funcionalidad Tipo de discapacidad Escoge dentro del área de iconos internos la opción Ver detalles.
EC 1.2 Cerrar la ventana ver detalles por el botón Cerrar	Mediante este escenario se puede cerrar la ventana de Ver detalles.	El sistema cierra la ventana y muestra el listado con los elementos existentes.	El usuario una vez autenticado en el sistema selecciona la Agrupación Funcional "Configuración" y luego selecciona la funcionalidad Tipo de discapacidad Escoge dentro del área de iconos internos la opción Ver detalles.

DCP_ Mostrar tipo de discapacidad**Condiciones de ejecución: El usuario debe estar autenticado como administrador.****SC Mostrar tipo de discapacidad**

Escenario	Descripción	Cantidad por página	Encabezado del Grid	Respuesta del sistema	Flujo central
EC 1.1 Mostrar datos correctamente.	Mediante este escenario se mostrará al usuario el listado de los tipos de discapacidades existentes en el sistema. En el listado se muestra una barra de iconos internos con las acciones a desarrollar sobre cada elemento: Ver detalles. Modificar. Además se muestra una barra de íconos flotantes con las acciones a realizar sobre el contexto: Crear. Actualizar, y Ayuda(próximas iteraciones de desarrollo)	NA	NA	El sistema muestra el listado de los elementos actualizados hasta la fecha.	El usuario una vez autenticado en el sistema. Selecciona del módulo Personal y dentro de la Agrupación Funcional "Configuración" selecciona la funcionalidad Tipo de discapacidad
EC 1.2 Seleccionar correctamente cantidad de elementos por página.	.Mediante este escenario el usuario escoge la cantidad de elementos a mostrar por página (los valores a escoger son 5, 10, 15 y 20).	V	NA	El sistema muestra la cantidad de elementos según la opción escogida(Se puede escoger 5,10,15,20)	El usuario una vez autenticado en el sistema. Selecciona del módulo Personal y dentro de la Agrupación Funcional "Configuración" selecciona la funcionalidad Tipo de discapacidad. El usuario selecciona la cantidad de elementos a mostrar por página.
		El usuario escoge la opción de 5	NA		
		V	NA		
		El usuario escoge la opción de 10	NA		
EC 1.3	Mediante este escenario se da clic en el	V	NA	El sistema ordena	El usuario una vez autenticado en el
		El usuario escoge la opción de 15	NA		
		V	NA		
		El usuario escoge la opción de 20			

<p>Ordenar los elementos de estructura correctamente.</p> <p>EC 1.4 No existen elementos creados.</p>	<p>encabezado del listado ordenándolo así alfabéticamente. (Icono que se muestra como una flecha).</p> <p>Mediante este escenario en caso de que no exista creado ningún elemento se muestra un listado vacío.</p>	<p>Se da clic encima del elemento y se activa la flecha de arriba</p> <p>V</p> <p>Se da clic encima del elemento y se activa la flecha de abajo.</p> <p>NA</p>	<p>NA</p> <p>NA</p>	<p>los elementos de forma ascendente. Ordena primero los habilitados y luego los deshabilitados. El sistema ordena los elementos de forma descendente. Ordena primero los habilitados y luego los deshabilitados. El sistema muestra el listado vacío.</p>	<p>sistema. Selecciona del módulo Personal y dentro de la Agrupación Funcional "Configuración" selecciona la funcionalidad Tipo de discapacidad. El usuario escoge la opción de ordenar alfabéticamente.</p> <p>El usuario una vez autenticado en el sistema. Selecciona del módulo Personal y dentro de la Agrupación Funcional "Configuración" selecciona la funcionalidad Tipo de discapacidad.</p>
---	--	--	---------------------	--	--

DCP_ Crear tipo de discapacidad.**Condiciones de ejecución: El usuario debe estar autenticado como administrador.****SC Crear tipo de discapacidad**

Escenario	Descripción	Nombre	Estado	Descripción	Respuesta del sistema	Flujo central
EC 1.1 Insertar datos correctamente.	Mediante este escenario se inserta en el sistema una nueva tipo de discapacidad.	V	V	V	El sistema actualiza el listado y muestra el mensaje "El elemento ha sido creado satisfactoriamente".	El usuario una vez autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad Selecciona en el área de iconos flotantes la opción de Crear. El usuario llena los datos correctamente y presiona el botón Aceptar.
		Sensitiva	Habilitado	Breve descripción		
		V	V	V		
		Auditiva	Deshabilitado	Vacio		
		V	V	V		
		Se introduce un texto que tenga como máximo 50 caracteres donde las palabras no pueden exceder los 30 caracteres.	Habilitado	Vacio		
		V	V	V		
Mental	Habilitado	Se escribe un texto donde las palabras no excedan los 30 caracteres y el máximo del texto es de 200 caracteres.				
EC 1.2 Insertar elementos repetidos.	Mediante este escenario se introducen datos para insertar un tipo de discapacidad que ya existe en el sistema.	V	V	V	El sistema muestra un mensaje de error indicando: El elemento ya existe.	El usuario una vez autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad. Selecciona en el área de iconos flotantes la opción de Crear. El usuario

		Sensitiva	Habilitado	Breve descripción		llena los datos e inserta un elemento que ya esté en el sistema y presiona el botón Aceptar.
EC 1.3 Insertar datos incompletos.	Mediante este escenario no se introducen todos los datos para insertar un tipo de discapacidad	I	V	V	El sistema muestra encima del componente un mensaje en color rojo indicando: Campo requerido.	El usuario una vez autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad "Tipo de discapacidad". Selecciona en

		Vacío	Deshabilitado	Descripción del requisito.		el área de iconos flotantes la opción de Crear El usuario llena los datos y deja campos obligatorios vacíos y presiona el botón Aceptar.
EC 1.4 Insertar datos incorrectos.	Mediante este escenario se introducen datos incorrectos para insertar una nueva parentesco	I	V	V	El sistema muestra en color rojo encima del componente: Solo se permiten letras.	El usuario una vez autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad Selecciona en el área de iconos flotantes la opción de Crear. El usuario llena los datos de forma incorrecta y presiona el botón Aceptar
		"@#%256^"	Habilitado	Vacio		
		I	V	V	El sistema muestra	

		A	Deshabilitado	Vacio	en color rojo encima del componente el mensaje: Debe entrar al menos 4 letras.	
		I	V	V	El sistema no permite la entrada de más caracteres.	
		(El usuario entra un texto con más de 50 caracteres)	Habilitado	Vacio		
		I	V	V	El sistema muestra en rojo encima del componente: Ha excedido el número de letras permitidas para una palabra.	
		El usuario entra una palabra con más de 30 caracteres.	Habilitado	Vacio		
		V	V	I	El sistema muestra en rojo encima del componente: Ha excedido el número de letras permitidas para una palabra.	
		nueva discapacidad	Habilitado	El usuario entra una palabra que exceda los 30 caracteres.		
		V	V	I	El sistema no permite la entrada de más caracteres.	
		motora	Habilitado	El usuario intenta escribir un texto con más de 200 caracteres.		
		V	V	I	El sistema muestra encima del componente con color rojo: No más de 200 caracteres.	
		sensitiva	Deshabilitado	El usuario pega en el campo más de 200 caracteres.		
EC 1.5	Se cancela la	NA	NA	NA		El usuario una vez

Cancelar operación.	creación del tipo de discapacidad.		autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad Selecciona en el área de iconos flotantes la opción de Crear. El usuario llena o no los datos y presiona el botón Cancelar
---------------------	------------------------------------	--	---

DCP_ Modificar tipo de discapacidad						
Condiciones de ejecución: El usuario debe estar autenticado como administrador						
SC Modificar tipo de discapacidad						
Escenario	Descripción	Nombre	Estado	Descripción	Respuesta del sistema	Flujo central
EC 1.1 modificar datos correctamente.	Mediante este escenario se modifica en el sistema un tipo de discapacidad.	V	V	V	El sistema actualiza el listado y muestra el mensaje "El elemento ha sido modificado satisfactoriamente".	El usuario una vez autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad Selecciona en el área de iconos internos la opción de Modificar. El usuario llena los datos correctamente y presiona el botón Guardar.
		Sensitiva	Habilitado	Breve descripción		
		V	V	V		
		Auditiva	Deshabilitado	Vacio		
		V	V	V		
		Se introduce un texto que tenga como máximo 50 caracteres donde las palabras no pueden exceder los 30 caracteres.	Habilitado	Vacio		
		V	V	V		
		Mental	Habilitado	Se escribe un texto donde las palabras no excedan los 30 caracteres y el máximo del texto es de 200 caracteres.		

EC 1.2 modificar elementos repetidos.	Mediante este escenario se introducen datos para modificar un tipo de discapacidad que ya existe en el sistema.	V	V	V	El sistema muestra un mensaje de error indicando: El elemento ya existe.	El usuario una vez autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad. Selecciona en el área de iconos internos la opción de Modificar. El usuario llena los datos e modifica un elemento que ya esté en el sistema y presiona el botón Guardar.
		Sensitiva	Habilitado	Breve descripción		
EC 1.3	Mediante este	I	V	V	El sistema muestra	El usuario una vez

modificar datos incompletos.	escenario no se introducen todos los datos para modificar un tipo de discapacidad	Vacío	Deshabilitado	Descripción del requisito.	encima del componente un mensaje en color rojo indicando: Campo requerido.	autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad. Selecciona en el área de iconos internos la opción de Modificar. El usuario llena los datos y deja campos obligatorios vacíos y presiona el botón Guardar.
EC 1.4 modificar datos incorrectos.	Mediante este escenario se introducen datos incorrectos para modificar una parentesco	I	V	V	El sistema muestra en color rojo encima del componente: Solo se permiten letras.	El usuario una vez autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad Selecciona en el área de iconos internos la opción de Modificar. El usuario llena los datos de forma incorrecta y presiona el botón Guardar
		"@#%256^"	Habilitado	Vacio		
		I	V	V	El sistema muestra en color rojo encima del componente el mensaje: Debe entrar al menos 4 letras.	
		A	Deshabilitado	Vacio		
I	V	V	El sistema no			

		(El usuario entra un texto con más de 50 caracteres)	Habilitado	Vacio	permite la entrada de más caracteres.	
		I	V	V	El sistema muestra en rojo encima del componente: Ha excedido el número de letras permitidas para una palabra.	
		El usuario entra una palabra con más de 30 caracteres.	Habilitado	Vacio	El sistema muestra en rojo encima del componente: Ha excedido el número de letras permitidas para una palabra.	
		V	V	I	El sistema muestra en rojo encima del componente: Ha excedido el número de letras permitidas para una palabra.	
		nueva discapacidad	Habilitado	El usuario entra una palabra que exceda los 30 caracteres.	El sistema muestra en rojo encima del componente: Ha excedido el número de letras permitidas para una palabra.	
		V	V	I	El sistema no permite la entrada de más caracteres.	
		motora	Habilitado	El usuario intenta escribir un texto con más de 200 caracteres.		
		V	V	I	El sistema muestra encima del componente con color rojo: No más de 200 caracteres.	
		sensitiva	Deshabilitado	El usuario pega en el campo más de 200 caracteres.		
EC 1.5	Se cancela la	NA	NA	NA		El usuario una vez

Cancelar operación.	modificación del tipo de discapacidad.					autenticado en el sistema selecciona la agrupación funcional "Configuración". Seguidamente selecciona la funcionalidad Tipo de discapacidad Selecciona en el área de iconos internos la opción de Modificar. El usuario llena o no los datos y presiona el botón Cancelar
---------------------	--	--	--	--	--	--

Anexo 9 Casos de prueba de integración

Número de caso de prueba: Int_2

Funcionalidad a integrar: Color de ojos

Condiciones de ejecución: Se debe crear el color de ojos :negros

Descripción de la prueba: Comprobar que cuando se cree los datos de una persona y se llenen los rasgos fenotípicos el color de ojos negros se liste.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Llena los datos para crear una nueva persona, seleccionando el color de ojos negros.

Resultado esperado: Se lista el color de ojos: negros.

Evaluación: Satisfactoria

Número de caso de prueba: Int_3

Funcionalidad a integrar: Color de pelo

Condiciones de ejecución: Se debe crear el color de pelo :rojo

Descripción de la prueba: Comprobar que cuando se cree los datos de una persona y se llenen los rasgos fenotípicos el color de pelo: rojo se liste.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Llena los datos para crear una nueva persona, seleccionando el color de pelo: rojo.

Resultado esperado: Se lista el color de pelo: pelo.

Evaluación: Satisfactoria

Número de caso de prueba: Int_4

Funcionalidad a integrar: Razas

Condiciones de ejecución: Se debe crear la raza mestiza.

Descripción de la prueba: Comprobar que cuando se cree los datos de una persona y se llenen los rasgos fenotípicos la raza: mestiza se liste.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Llena los datos para crear una nueva persona, seleccionando la

raza: mestiza.

Resultado esperado: Se lista el color de pelo: pelo.

Evaluación: Satisfactoria

Número de caso de prueba: Int_5

Funcionalidad a integrar: Tipo de identidad

Condiciones de ejecución: Se debe crear el tipo de identidad :FAR

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda escoger el tipo de identidad: FAR.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Cuando aparezca la interfaz se debe seleccionar el tipo de identidad de las FAR.

Resultado esperado: Se lista el tipo de identidad: FAR

Evaluación: Satisfactoria

Número de caso de prueba: Int_6

Funcionalidad a integrar: Tipo de identidad

Condiciones de ejecución: Se debe crear el tipo de identidad :FAR

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda escoger el tipo de identidad: FAR.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Cuando aparezca la interfaz se debe seleccionar el tipo de identidad de las FAR.

Resultado esperado: Se lista el tipo de identidad: FAR

Evaluación: Satisfactoria

Número de caso de prueba: Int_7

Funcionalidad a integrar: Estado civil

Condiciones de ejecución: Se debe crear el estado civil: soltero

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda escoger el estado civil: soltero.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la

funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Se introducen los datos. Cuando aparezca la interfaz de datos personales se debe seleccionar el estado civil: soltero.

Resultado esperado: Se lista el estado civil: soltero

Evaluación: Satisfactoria

Número de caso de prueba: Int_8

Funcionalidad a integrar: Huérfano

Condiciones de ejecución: Se debe crear el tipo de huérfano: madre

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda escoger el tipo de huérfano: madre

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Se introducen los datos. Cuando aparezca la interfaz de datos personales se debe seleccionar el tipo de huérfano: madre.

Resultado esperado: Se lista el tipo de huérfano: madre.

Evaluación: Satisfactoria

Número de caso de prueba: Int_9

Funcionalidad a integrar: Nivel escolar

Condiciones de ejecución: Se debe crear el nivel escolar: 9no grado.

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda escoger el nivel escolar: 9no grado.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Se introducen los datos. Cuando aparezca la interfaz de datos de superación se debe seleccionar el nivel escolar: 9no grado.

Resultado esperado: Se lista el nivel escolar: 9no grado.

Evaluación: Satisfactoria

Número de caso de prueba: Int_10

Funcionalidad a integrar: Grado académico

Condiciones de ejecución: Se debe crear el grado académico: ingeniero.

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda

escoger el grado académico: ingeniero.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Se introducen los datos. Cuando aparezca la interfaz de datos de superación se debe seleccionar el grado académico: ingeniero.

Resultado esperado: Se lista el grado académico: ingeniero.

Evaluación: Satisfactoria

Número de caso de prueba: Int_11

Funcionalidad a integrar: Ocupación

Condiciones de ejecución: Se debe crear la ocupación: obrero.

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda escoger la ocupación: obrero.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Se introducen los datos. Cuando aparezca la interfaz de datos de superación se debe seleccionar la ocupación: obrero.

Resultado esperado: Se lista la ocupación: obrero.

Evaluación: Satisfactoria

Número de caso de prueba: Int_12

Funcionalidad a integrar: Categoría docente

Condiciones de ejecución: Se debe crear la categoría docente: auxiliar.

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda escoger la categoría docente: auxiliar.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Se introducen los datos. Cuando aparezca la interfaz de datos de superación se debe seleccionar la categoría docente: auxiliar.

Resultado esperado: Se lista la ocupación: obrero.

Evaluación: Satisfactoria

Número de caso de prueba: Int_13

Funcionalidad a integrar: Categoría investigativa

Condiciones de ejecución: Se debe crear la categoría investigativa: investigador auxiliar.

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para crear una persona se pueda escoger la categoría investigativa: investigador auxiliar.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de crear en el área de iconos flotantes. Se introducen los datos. Cuando aparezca la interfaz de datos de superación se debe seleccionar la categoría investigativa: investigador auxiliar.

Resultado esperado: Se lista la ocupación: obrero.

Evaluación: Satisfactoria

Número de caso de prueba: Int_16

Módulo a integrar: Estructura y composición

Condiciones de ejecución: Deben existir estructuras creadas en el módulo de estructura y composición.

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para modificar los datos de una persona se pueda escoger un área.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de modificar en el área de iconos internos. Se modifican los datos. Cuando aparezca la interfaz de datos de entidad se debe seleccionar un área.

Resultado esperado: Se listaron los cargos existentes.

Evaluación: Satisfactoria

Número de caso de prueba: Int_17

Módulo a integrar: Configuración

Condiciones de ejecución: Deben existir provincias creadas en el módulo de configuración.

Descripción de la prueba: Comprobar que cuando aparezca la interfaz para modificar los datos de una persona se pueda escoger un área.

Entrada/Pasos de ejecución: El usuario se dirige a la agrupación funcional Personal y accede a la funcionalidad Personas, escoge la opción de modificar en el área de iconos internos. Se modifican los datos. Cuando aparezca la interfaz de datos nominales se debe seleccionar una provincia.

Resultado esperado: Se listaron las provincias existentes.

Evaluación: Satisfactoria
