

Universidad de las Ciencias Informáticas

Facultad CITEC



“Sistema informático para la gestión de los recursos humanos en la Facultad de Ciencias y Tecnologías Computacionales de la Universidad de las Ciencias Informáticas”.

**TRABAJO DE DIPLOMA PARA OPTAR POR EL TÍTULO DE INGENIERO
EN CIENCIAS INFORMÁTICAS**

Autor:

Karen Alonso Tejera

Tutores:

Dr.C Rosario Garza Ríos

Ing. Bárbara Bron Fonseca

La Habana, junio de 2017

“Año 59 de la Revolución”

DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas (UCI) los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste se firma la presente a los ____ días del mes de _____ del año 2017.

Firma de la autora
Karen Alonso Tejera

Firma de la tutora
Dr.C Rosario Garza Ríos

Firma de la tutora
Ing. Bárbara Bron Fonseca

Datos de contacto

Tutora:

Dra. Rosario Garza Rios: Profesor Titular de la Universidad Tecnológica de la Habana José Antonio Echeverría, vinculado a la docencia desde 1980, impartiendo asignaturas relacionadas con la disciplina de Matemática Aplicada en las carreras de Ingeniería Industrial e Informática; presenta experiencia como tribunal de tesis, tutor y oponente en pregrado y postgrado.

Correo electrónico: rosariog@ind.cujae.edu.cu

Tutora:

Ing. Bárbara Bron Fonseca: Graduada en el año 2012 de Ingeniera en Ciencias Informáticas en la Universidad de las Ciencias Informáticas. Se desempeñó como Especialista B en Informática en la Empresa de Tecnologías para la Defensa XETID. Actualmente es profesora del departamento de ISW en la facultad CITEC de la UCI. Posee publicaciones en revistas científicas y memorias de eventos. Certificado como especialista en calidad para revisiones de Software. Presenta experiencia como tutor y oponente en pregrado.

Correo electrónico: bbron@uci.cu

Agradecimientos

- *Agradezco a mis padres y hermana que me apoyaron siempre para llegar hasta aquí, que han sido incondicionales.*
- *A mi amor Joanny por tenerme paciencia y darme ánimo durante este camino recorrido.*
- *Le agradezco a mi hija por brindarme pequeños momentos de su tiempo para estudiar y a las abuelas de mi hija por su tiempo también que le dedicaron para ayudarme en los estudios.*
- *Le doy gracias a todos mis profesores que me exigieron siempre lo máximo para superarme cada día más.*
- *A mi antiguo grupo que creamos una pequeña familia todos diferentes pero unidos. Y a mi nuevo grupo que me aceptó como una más y obtuve su ayuda incondicional para terminar este último año.*
- *Le agradezco a mis tutoras porque no tengo palabras para agradecerle todo lo que me ayudaron y me guiaron durante toda mi carrera.*
- *A mi jefe y amigo Omar que fue muy paciente con mi días de estrés por las pruebas.*
- *A mis amigos Apsara, Aimara, Yohana y Harold que siempre estuvieron pendiente de mis estudios y dando fuerzas para continuar.*
- *A todos los trabajadores de la facultad por ayudarme y guiarme durante la carrera.*

Resumen

El factor humano es esencial en todas las empresas productivas del mundo. El uso de sistemas de gestión de recursos humanos se ha convertido en un mecanismo clave para lograr el éxito de cualquier empresa. En la Facultad de Ciencias y Tecnologías Computacionales de la Universidad de las Ciencias Informáticas (UCI) se han informatizado algunos subprocesos de la gestión de los recursos humanos pero aún quedan actividades claves que no han sido insertadas en un sistema informático lo que la hace vulnerable a la generación de un número de errores de impacto alto.

La presente investigación describe una solución a la problemática antes planteada, trazándose como objetivo la implementación de un sistema para la gestión de los recursos humanos. Para guiar el proceso de desarrollo se utilizó la metodología Open UP. El sistema fue implementado en lenguaje de programación Java 8.1.1 en el IDE de desarrollo NetBeans, como gestor de base de datos PostgreSQL en su versión 9.3 y se obtuvo como resultado un sistema capaz de llevar la trazabilidad de las informaciones claves para la toma de decisiones y que genera reportes estadísticos de la gestión de los recursos humanos.

Palabras Claves: *Recursos humanos, sistema de gestión, reportes, toma de decisiones.*

Abstract

The human factor is essential in every productive enterprise in the world. The use of human resources management systems has become a key mechanism to achieve the success of any company. Some sub-processes of human resources management have been computerized in Faculty 6 of the University of Computer Science (UCI) but there are still key activities that have not been inserted in a computer system, which makes it vulnerable to the generation of a Number of high impact errors. The present research describes a solution to the above problematic, aiming at the implementation of a system for the management of human resources. To guide the development process, the Open UP methodology was used. The system was implemented in Java programming language 8.1.1 in the NetBeans development IDE, as a PostgreSQL database manager in its version 9.3 and resulted in a system capable of tracking the key information for making Decisions. And that generates statistical reports of the management of the human resources.

Keywords: Human resources, management system, reports, decision making.

Índice

Introducción	1
Capítulo 1: Fundamentos teóricos	5
Introducción	5
1.1. Conceptos asociados al sistema informático para la gestión de recursos humanos.	5
1.2. Sistemas gestión de recursos humanos.....	7
1.2.1. SIRHU, Versión: 4.0 / Año: 2009.....	7
1.2.2. Orange HRM, Versión: 2.6.0.2/ Año: 2011.....	7
1.2.3 CEDRUX, Versión: 1.0/ Año: 2010.	8
1.2.4 GREHU, Versión: 4.0/ Año: 2012.	9
1.3. Tecnologías, herramientas y metodologías a considerar	10
1.3.1. Metodología de Desarrollo de Software	10
1.3.2. Herramienta CASE	11
1.3.3. Lenguaje de programación.....	11
1.3.4. IDE de desarrollo.....	12
1.3.5. Sistemas Gestores de Base de Datos (SGBD).....	12
Conclusiones del capítulo	13
Capítulo 2: Descripción del sistema.....	15
Introducción	15
2.1. Propuesta de solución	15
2.2. Modelo de Negocio.....	16
2.2.1. Actores del negocio.....	16
2.2.2. Trabajadores del negocio.....	16
2.3. Diagrama de casos de uso del negocio	17
2.4. Diagrama de actividades del modelo de negocio.....	17
2.5. Descripción de los casos de usos del negocio.....	18
2.6. Extracción de Requisitos	20
2.6.1. Requisitos funcionales	20
2.6.2. Requisitos no funcionales	23

2.7. Modelo de casos de uso.....	24
2.7.1 Actores del sistema	25
2.8. Diagrama de casos de uso.....	25
2.9. Descripción de casos de uso.....	26
Conclusiones del capítulo	31
Capítulo 3: Análisis y diseño del sistema.....	33
Introducción	33
3.1. Modelo del diseño	33
3.2. Descripción de patrones arquitectónicos.....	33
3.2.1 Patrones arquitectónicos.....	33
3.2.2 Patrón arquitectónico Modelo – Vista – Controlador.....	34
3.3. Patrones de diseño.....	36
3.3.1 Patrones GRASP	36
3.3.2 Patrones Gof	38
3.4. Diagrama de clases del diseño	39
3.5. Modelo de datos	40
3.6. Diagrama de componentes	41
3.7. Diagrama de despliegue.....	42
Conclusiones del capítulo	43
Capítulo 4: Implementación y Análisis de los Resultados	44
Introducción	44
4.1. Código fuente	44
4.1.1 Estándares de codificación	44
4.1.2 Nombres de clases y métodos	44
4.1.3 Estructura	45
4.2. Pantallas principales de la aplicación.....	46
4.3. Niveles de Prueba	48
4.4. Tipos de Prueba	49
4.5. Métodos de Prueba	49
4.5.1 Pruebas funcionales de caja negra	49

4.5.2	Resultados e interpretación de las pruebas	56
4.6.	Resultados alcanzados.....	57
	Conclusiones del capítulo	58
	Conclusiones generales.....	59
	Recomendaciones	60
	Referencias bibliográficas.....	61
	Bibliografía	63
	Anexos	65

Índice de figuras

Figura 1: Propuesta de solución.	16
<i>Figura 2: Caso de uso del negocio.</i>	<i>17</i>
Figura 3: Diagrama de actividades: Generar Prenómina.....	18
Figura 4: Diagrama de actividades: Elaborar Informes.....	18
Figura 5: Diagrama de casos de uso del sistema.....	26
Figura 6: Patrón arquitectónico Modelo Vista Controlador.	35
Figura 7: Ejemplo de clase que usa el patrón experto.....	37
Figura 8: Ejemplo de clase que usa el patrón creador.	37
Figura 9: Ejemplo de clase que usa el patrón Bajo acoplamiento.	38
<i>Figura 10: Ejemplo de clase que usa el patrón Controlador.....</i>	<i>38</i>
Figura 11: Ejemplo de clase que usa el patrón Compositor.	39
Figura 12: Diagrama de clases del diseño.....	40
Figura 13: Diagrama Entidad Relación.	41
Figura 14: Diagrama de componentes.....	42
Figura 15: Diagrama de despliegue.....	43
Figura 16: Estándar de codificación CamelCase.....	45
Figura 17: Estándar de codificación UpperCamelCase.....	45
Figura 18: Página principal de Recursos Humanos: Persona.	46
Figura 19: Página principal de Adicionar Persona: Persona.	47
Figura 20: Página principal de Agregar Categoría: Categoría.....	48
Figura 21: Página principal de Agregar Nomencladores: Nomencladores.	48
Figura 22: Clasificación de las NC identificadas. Primera iteración.....	56

Figura 23: Clasificación de las NC identificadas. Segunda iteración..... 57

Figura 24: Iteraciones de las pruebas funcionales. 57

Índice de tablas

Tabla 1: Comparación entre la soluciones existente.	10
Tabla 2: Actores del negocio.	16
Tabla 3: Trabajadores del negocio.	16
Tabla 4: Caso de uso del negocio: Generar Prenómina.	18
Tabla 5: Caso de uso del negocio: Elaborar Informe.	19
Tabla 6: Actores del sistema.	25
Tabla 7: Descripción de caso de uso: Gestionar persona.	26
Tabla 8. DCP Adicionar Persona	50
Tabla 9. DCP Generar reportes de categorías colectivas	52
Tabla 10. DCP Generar reporte de evaluación individual.	54

Introducción

El mundo se encuentra inmerso en un desarrollo científico y tecnológico impulsado por la informática, contribuyendo a que la sociedad se transforme aceleradamente, tanto económica, social y políticamente, con el objetivo fundamental de alcanzar eficiencia en los procesos y eliminar los errores.

Las Tecnologías de la Información y las Comunicaciones (TIC) son esenciales para mejorar la productividad, calidad y funcionamiento de las instituciones, redefiniendo los paradigmas de ejecutar, gestionar y almacenar. Cuba no está exenta del vertiginoso avance de las tecnologías, por lo que le confiere un alto nivel de prioridad a la informatización de la sociedad cubana y de sus principales procesos.

La Universidad de las Ciencias Informáticas (UCI), también se encuentra inmersa en el proceso de informatización del país y es vanguardia en la implantación de cada uno de los sistemas que en ella se desarrollan. Un proceso clave en la universidad es la gestión de sus recursos humanos. El uso de sistemas de gestión de recursos humanos se ha convertido en un factor clave para lograr el éxito de cualquier empresa, proporcionando mayor control y centralización de la información, con el fin de lograr una mejor precisión en las tareas que se realizan.

En la Facultad de Ciencias y Tecnologías Computacionales (CITEC) de la UCI se han informatizado algunos subprocesos de la gestión de los recursos humanos como son el autorizo para medios tecnológicos, generar la planificación de la guardia obrera, pago por conceptos de viático y la gestión de los activos fijos, pero aún quedan actividades claves que no han sido insertadas en el sistema informático. La pre Nómina mensual aún se genera de forma manual, por lo que es vulnerable a la introducción de errores humanos, las hojas de firma se hacen aún con el método tradicional. No se lleva a cabo la identificación de los trabajadores atendiendo a su área. No existen mecanismos de alerta oportuna que identifiquen cuando a un trabajador docente está próximo a vencerse la categoría docente en el 90 % de los casos esto se identifica un mes antes del vencimiento de las categorías a través de terceros. La evaluación de desempeño se hace a través de documentos pero no existen registros que permitan la trazabilidad y análisis de cada trabajador de forma individual.

A partir de los resultados identificados en auditorías realizadas y la investigación realizada por la autora se evidenció que el proceso de gestión de recursos humanos de la Facultad CITEC aún es vulnerable a la generación de un número de errores de impacto alto y que se resumen a continuación:

- No existe un flujo de trabajo organizado lo que provoca demora en el plazo 3 a 5 días en la entrega de la documentación necesaria.
- Existen errores humanos en la concordancia entre los registros y la nómina final, lo cual demora la entrega de la nómina con los errores corregidos. Esto ocurre con promedio 3 trabajadores mensuales.
- No es posible realizar reportes en tiempo real por lo imposibilita la toma de decisiones.
- No existen mecanismos de alerta oportuna cuando ocurre alguna incidencia.
- La trazabilidad de los resultados individuales y colectivos es engorrosa.

A partir de la problemática planteada se define como **problema de la investigación**: ¿Cómo contribuir a la gestión de los recursos humanos en la Facultad CITEC de la UCI?

Se define como **objeto de estudio** de la investigación: los procesos de gestión de recursos humanos, enmarcado en el **campo de acción**: los sistemas para la gestión de los recursos humanos.

Como **objetivo general** se plantea: Desarrollar un sistema para contribuir a la gestión de los recursos humanos en la Facultad CITEC de la UCI.

Para dar continuidad al objetivo general, se plantea como **objetivos específicos** los siguientes:

1. Identificar los sistemas relacionados con la gestión de los recursos humanos.
2. Seleccionar las herramientas y metodologías para el desarrollo del sistema.
3. Diseñar el sistema para gestionar los recursos humanos.
4. Implementar las funcionalidades para lograr el despliegue del sistema para gestionar los recursos humanos.
5. Garantizar el correcto funcionamiento del sistema aplicando pruebas de funcionalidad para gestionar los recursos humanos con el objetivo de identificar y corregir posibles no conformidades.

A partir de lo antes planteado se define como **idea a defender**: Con el desarrollo de un sistema informático se contribuirá a la gestión de los recursos humanos en la Facultad CITEC de la UCI.

Para dar cumplimiento al objetivo general se definen las siguientes **tareas de investigación**:

- Elaborar el marco teórico referencial relacionado con los sistemas para la gestión de recursos humanos.
- Seleccionar las principales tecnologías y herramientas que se ajustan a la situación real del negocio para el desarrollo de la propuesta de solución.
- Elaborar el diseño del sistema para gestionar los recursos humanos.
- Realizar la implementación del sistema para gestionar los recursos humanos.
- Aplicar pruebas de funcionalidad al sistema para gestionar los recursos humanos con el objetivo de identificar y corregir posibles no conformidades.

Para obtener los conocimientos necesarios con la finalidad de hacer posible el cumplimiento del objetivo trazado en la presente investigación, se llevó a cabo un estudio en el que se utilizaron los siguientes métodos de la investigación científica:

Métodos empíricos usados para poder determinar el grado de complejidad que adquiere el problema y para identificar si quedaron o no satisfechas todas las necesidades previstas a través de:

Entrevistas: Aplicada a directivos, técnicos y especialistas de la Facultad de Ciencias y Tecnologías Computacionales de la UCI, con el objetivo de obtener información sobre el proceso de gestión de recursos humanos, y a partir de las respuestas obtenidas, identificar algunas de las irregularidades y deficiencias iniciales que se cometen en dicho proceso.

Métodos teóricos: Usados para estudiar las características del objeto de investigación que no son observables directamente, para apoyar el análisis y la síntesis y su relación con otros fenómenos:

- **Analítico-Sintético:** Aplicado por el autor para descomponer el problema de investigación en elementos por separados y profundizar en el estudio de cada uno de ellos, para luego sintetizarlos en la solución propuesta. Entre los elementos identificados se encuentran la función de los sistemas de gestión de recursos humanos y el manejo de los indicadores críticos que son básicos en las organizaciones nacionales, de manera que permita construir correctamente el marco teórico de la investigación.

- **Histórico-Lógico:** Aplicado por el autor para constatar teóricamente cómo han evolucionado los sistemas de información en el control de las actividades laborales, los cuales varían a partir de las leyes y cambios aplicados en dependencia de las normas internacionales y para profundizar en los lineamientos establecidos en las normativas de los recursos humanos de forma tal que permita un análisis real a la hora de identificar los problemas que existen en la gestión de estos procesos.

El presente trabajo de diploma se ha estructurado de la siguiente forma:

Capítulo 1: Fundamentación teórica

Se especifican los conceptos que son tratados a lo largo del documento y que tienen vital importancia para la comprensión del mismo. Se realiza un estudio de los sistemas de gestión de recursos humanos, utilizadas hoy en Cuba y en el mundo y se describen y justifican las diferentes herramientas, metodologías y lenguajes de programación a utilizar durante la construcción del sistema.

Capítulo 2: Características del Sistema

Se describe las características del sistema a desarrollar. Se define el modelo de dominio, los requisitos funcionales y no funcionales del sistema para comprender mejor su funcionamiento, así como, los actores, diagrama de casos de uso del sistema y los patrones de casos de uso utilizados.

Capítulo 3: Análisis y Diseño del Sistema

Está relacionado con el diseño del sistema. Se describen los patrones de diseño empleados. Se presentan los diagramas de clases del diseño, el diagrama entidad relación, el diagrama de componentes y el diagrama de despliegue del sistema con el objetivo de tener una idea concreta, de cómo quedará.

Capítulo 4: Implementación y Análisis de los Resultados

Está enfocado a la fase de implementación para dar solución a los requerimientos funcionales identificados. Se describen los estándares de codificación, el tratamiento de errores para darle solución al sistema propuesto y se demuestra la solución del problema a resolver.

Capítulo 1: Fundamentos teóricos

Introducción

El presente capítulo introduce los conceptos básicos relacionados con el funcionamiento de los sistemas que se encargan de gestionar los recursos humanos. Son analizados los sistemas profesionales de corte nacional e internacional, que son usados actualmente y son seleccionadas las herramientas y metodologías a utilizar en el desarrollo del software, justificando el uso de cada una de ellas.

1.1. Conceptos asociados al sistema informático para la gestión de recursos humanos.

Gestión de la información

La gestión de la información representa la forma de organizar, evaluar, presentar y comparar los datos en un determinado contexto, controlando su calidad, de manera que esta sea veraz, oportuna, significativa, exacta y útil; y que esta información esté disponible en el momento que se le necesite. Ella se encamina al manejo de la información, documentos, metodologías, informes, publicaciones, soportes y flujos en función de los objetivos estratégicos de una organización (Ledo, 2012).

La gestión de la información se vincula con la generación y la aplicación de estrategias, el establecimiento de políticas, así como con el desarrollo de una cultura organizacional y social dirigida al uso racional, efectivo y eficiente de la información en función de los objetivos y metas trazadas en materia de desempeño y de calidad.

Un Sistema de Información está constituido por los métodos y procedimientos establecidos para registrar, procesar, resumir e informar sobre las operaciones de una entidad. La calidad de la información que brinda el sistema afecta la capacidad de los directivos y ejecutivos para adoptar decisiones adecuadas que permitan controlar las actividades de la entidad (Contraloría General de la República, 2009).

Gestión de los recursos humanos

La gestión de recursos humanos son las actividades coordinadas para dirigir y controlar una organización, que permiten materializar la política laboral, que se aplican con la participación activa y efectiva de los trabajadores en la planificación, organización, dirección, control y evaluación de los recursos humanos, que determinan o inciden en el desempeño de la facultad.

Actualmente los recursos humanos se consideran un medio activo y base para el desarrollo estratégico de las empresas. El reto de los Recursos Humanos es lograr un desempeño laboral superior para alcanzar la máxima eficacia y eficiencia de las organizaciones de producción y servicios para el sostenido desarrollo económico y social del país.(Concepción 2011)

En el proceso de gestión de los recursos humanos intervienen todos los miembros activos de la Facultad, entendiéndose por tales: la decana, el vicedecano administrativo, los jefes de área, técnico de recursos humanos. Para poner en funcionamiento a estos miembros activos se realiza la articulación de las funciones sociales y la definición de las políticas del personal las cuales consideran los objetivos de la organización, sobre la base del uso de instrumentos administrativos y reglamentarios. La gestión de recursos humanos se basa en planear, organizar, desarrollar y controlar aquellas técnicas capaces de promover un desempeño eficiente por parte del personal empleado.

La administración se puede ver también como un proceso esta tiene varias funciones en la facultad:

- Determinar las necesidades del personal.
- Decidir si contratar empleados de forma temporal o permanente en base a las necesidades.
- Seleccionar y potenciar a los empleados más eficientes.
- Supervisar el trabajo de los empleados.
- Asegurar la buena marcha del grupo y las relaciones entre los empleados.
- Redactar los documentos y políticas de los empleados.
- Asegurar un alto rendimiento.
- Administrar la pre Nómina.
- Asegurar la igualdad de oportunidades entre los empleados.
- Combatir la discriminación.
- Resolver posibles problemas referentes al trabajo.
- Asegurar que las prácticas de la empresa se rigen en base a varias regulaciones.
- Trabajar la motivación de los empleados.

Del mismo modo, los administradores han de trabajar en sus propias capacidades interpersonales. Una vez sabiendo manejar esto, es posible que el trabajo resulte más efectivo.(Díaz 2009)

1.2. Sistemas gestión de recursos humanos

A continuación, se expone un análisis sobre algunos de los sistemas que presentan como principal característica la gestión de recursos humanos, utilizados hoy en Cuba y en el mundo. Posteriormente se realizará una comparación para definir características que pudieran aportar de alguna manera al desarrollo de la propuesta de solución. Se justifica la necesidad de elaborar un nuevo sistema adecuado a las características propias de la facultad CITED de la UCI.

1.2.1. SIRHU, Versión: 4.0 / Año: 2009.

Durante la investigación se analizó una herramienta en La universidad de Córdoba para la implementación del control de los recursos humanos basado en la integración de los recursos humanos en la universidad. El software **SIRHU** ha sido desarrollado por el equipo de Servicio de informática de la universidad. El producto permite gestionar todos los aspectos relacionados con el personal al servicio de una universidad. Está compuesto por 7 módulos:

1. Planificación: permite conocer las necesidades de personal de la Universidad, se materializan en una plantilla teórica, valoradas económicamente.
2. Selección: cubre el periodo previo a la incorporación del trabajador.
3. Registro del personal: consiste en la gestión de la relación del trabajador.
4. Formación: gestiona los cursos y actividades formativas que realiza el personal al servicio de la Universidad.
5. Retribuciones: se encarga de determinar qué cantidades se han de abonar periódicamente a los trabajadores en pago a sus servicios. Gestiona además la relación con los organismos externos que intervienen en este proceso.
6. Gestión de becas: gestiona el registro de los becarios y el pago de las becas cuando éstas son gestionadas por la propia Universidad.
7. Ordenación Académica: gestiona la actividad docente de los profesores de la Universidad, determinando las asignaturas y créditos que imparten.(Tetilla 2009)

1.2.2. Orange HRM, Versión: 2.6.0.2/ Año: 2011.

Otra de las aplicaciones analizadas fue una especializada en gestionar los recursos humanos de la empresa Orange HRM. Este sistema permite acceso con rol de empleado o de administrador, a los distintos módulos que permiten identificar a los empleados, escribir notas e informes, analizar el rendimiento y horas de trabajo, administrar vacaciones y bajas. Consta de 8 módulos:

1. Módulo de Administrador: permite definir la información general y estructura de la empresa, definir los puestos de trabajo de la empresa, las categorías salariales.
2. Módulo de Personal: permite gestionar a los empleados y consultar su información.
3. Módulo del Empleado: permite gestionar y editar la información del trabajador, como datos de contacto, direcciones, personas de contacto alternativas, su propia foto.
4. Módulo de Informes: genera informes personalizables según las necesidades, para uno o un determinado número de empleados. Pueden definirse multitud de criterios a la hora de generar un informe, pudiendo guardar estos como plantillas para no repetir las mismas tareas.
5. Módulo de Permisos: permite una completa gestión de los días de permiso de los trabajadores, definir los posibles tipos de permiso, establecer los días festivos, de vacaciones y los tipos de jornada laborables.
6. Módulo de Tiempo y Asistencia: sirve para contabilizar horas o rentabilizar el tiempo, también llamado “time tracking”, permite registrar la cantidad de tiempo de un trabajador en una determinada tarea, también permite definir los días no laborables.
7. Módulo de Beneficios: es una plataforma integrada para administrar tareas relacionadas con los beneficios. Permite definir coberturas médicas y definir el esquema de la nómina de la empresa.
8. Módulo de Reclutamiento: es una solución completa para el proceso de contratación, también permite a los profesionales de recursos humanos generar plantillas y documentación para optimizar el proceso de contratación. (Torres 2011)

1.2.3 CEDRUX, Versión: 1.0/ Año: 2010.

Se investigó sobre un sistema desarrollado en la Universidad De Las Ciencias Informáticas llamado CEDRUX. El sistema es el encargado de mejorar los problemas que presenta la gestión del

capital humano dentro de un proyecto. Está conformado por cinco procesos centrales y dentro de ellos sub procesos como son:

1. Planificación del capital humano: consiste en desarrollar una estrategia que permita cubrir las futuras necesidades de esos recursos, en la implantación de un proyecto; es el proceso por el cual el jefe de implantación deberá mantener seguridad en el número correcto de personas, con las capacidades adecuadas.
2. Selección: se encuentra relacionado con el suministro de personas al proyecto. Este proceso responde por los insumos humanos e implica todas las actividades relacionadas con reclutamiento y elección de personas, así como su integración a las actividades organizadas.
3. Desarrollo: El desarrollo del potencial humano en los proyectos implica, no simplemente pensar en términos de aumento de la cantidad y la calidad del capital inteligente o recurso humano.
4. Evaluación: es un proceso destinado a determinar y comunicar a las personas, la forma en que están desempeñando su trabajo, y en principio, a elaborar planes de mejora.
5. Atención del Capital Humano: Las personas son capaces de realizar proyectos sumamente exitosos impulsados por sentimientos motivacionales que se van construyendo a partir de un deseo en el que se depositan creencias, valores y expectativas de la vida, resulta imposible pensar en un proyecto de cualquier clase sin involucrar la motivación, lo que nos impulsa a creer que se puede realizar cualquier cosa que nos proponamos, que no hay barreras que nos impidan alcanzar la cima.(Ing. Virtudes Milagro Figueredo Lara 2010).

1.2.4 GREHU, Versión: 4.0/ Año: 2012.

El sistema de Gestión de Recursos Humanos (GREHU), desarrollado en el centro de Estudios de Ingeniería en Sistemas (CEIS) de la Facultad de Ingeniería Industrial del Instituto Superior Politécnico “José Antonio Echeverría” (CUJAE), actualmente se encuentra en migración de todos sus módulos a una nueva tecnología.

El sistema, de forma general permite gestionar los Recursos Humanos (RRHH) de diferentes negocios de manera integral, independientemente de las diferencias existentes entre ellos, y obtiene una gran cantidad de información sobre estos RRHH. Es un sistema complejo compuesto por un módulo principal y 16 módulos secundarios donde cada uno se corresponde con alguna de las áreas clave de la GRH tales como: Inventario de Personal, Selección y Contratación,

Prenóminas, Nóminas, Pago por Resultados, Capacitación del Personal, Control de Dirigentes, Control de Asistencia. (Hernández 2012)

Tabla 1: Comparación entre la soluciones existente.

Parámetros a medir	SIRHU	Orange HRM	CEDRUX	GREHU
Tecnología	Privativa	Libre	Libre	Libre
Plataforma	Windows	Linux, Windows	Linux, Windows	Migración
Costo	si	si	no	si
Soporte	si	si	si	si
Gestión de prenomina	no	no	no	si

Al concluir su posterior análisis se constató que las soluciones existentes, no es posible adaptarlos a los procesos de gestión de recursos humanos definidos hoy en la universidad ya que no se tienen en cuenta los procesos claves para la gestión de estos, aunque si se tomaron las mejores prácticas de cada uno de ellos como punto de partida para desarrollar la presente investigación.

1.3. Tecnologías, herramientas y metodologías a considerar

Las tecnologías, herramientas y metodologías escogidas para el desarrollo del sistema se exponen en el siguiente subtema, para lo cual se analizaron las distintas alternativas y las necesidades del proyecto. El principal elemento que se tuvo en consideración fue que las herramientas debían ser compatibles con el sistema que ya fue implementado, multiplataformas, sin restricciones de uso.

1.3.1. Metodología de Desarrollo de Software

Una metodología es una guía para el desarrollo del software; que hace posible que todo el personal de un proyecto se vincule y pueda entenderse, estas son un conjunto de procedimientos, técnicas y ayuda a la documentación para el desarrollo de productos software, en la que se indican, paso a paso, todas las actividades a realizar para lograr informatizar el proceso deseado; muestran qué personas deben participar en el desarrollo de las actividades y el papel que desempeñan. Además, detallan la información que se debe producir como resultado de una actividad y la información necesaria para comenzarla.

Metodología de Desarrollo de Software (Open Up)

Open UP es una metodología de desarrollo de software, basada en RUP (Rational Unified Process), que contiene el conjunto mínimo de prácticas que ayudan a un equipo de desarrollo de software a realizar un producto de alta calidad, de una forma eficiente. Open UP, es un proceso unificado, iterativo e incremental, que se centra en el desarrollo colaborativo de software para generar sistemas de calidad. Los elementos que forman Open Up son tareas, disciplinas, artefactos y procesos. El ciclo de vida de un proyecto, según la metodología Open UP, permite que los integrantes del equipo de desarrollo aporten con micro-incrementos, que pueden ser el resultado del trabajo de unas pocas horas o unos pocos días. El progreso se puede visualizar diariamente, ya que la aplicación va evolucionando en función de este micro-incremento. El objetivo de Open UP es ayudar al equipo de desarrollo, a lo largo de todo el ciclo de vida de las iteraciones, para que sea capaz de añadir valor de negocio a los clientes, de una forma predecible, con la entrega de un software operativo y funcional al final de cada iteración. El ciclo de vida del proyecto provee a los clientes de: una visión del proyecto, transparencia y los medios para que controlen la financiación, el riesgo, el ámbito, el valor de retorno esperado (Infante A., 2013).

Todo proyecto en Open UP consta de cuatro fases: inicio, elaboración, construcción y transición. Cada una de estas fases se divide a su vez en iteraciones (OpenUp Basic. Eclipse Foundation, 2011)

Metodología de Desarrollo de Software seleccionado

Se escoge como metodología de desarrollo de software a Open Up, que permite al sistema integrarse a requerimientos cambiantes que se pueden dar durante el desarrollo del proyecto, es una metodología ágil, que presenta documentación, muy favorable para proyectos que presentan pocos integrantes y es aplicable a sistemas de bajo costo.

1.3.2. Herramienta CASE

Visual Paradigm es una herramienta visual de Ingeniería de Software para el modelado. Brinda una colección de menús, barras de herramientas y ventanas que forman el área de trabajo, lo cual permite crear diferentes tipos de diagramas en un ambiente completamente visual, manejar de manera rápida aplicaciones de mayor calidad y a un menor coste. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación (Martín T., 2013).

1.3.3. Lenguaje de programación

Teniendo en cuenta que una de las restricciones del sistema es que debe ser multiplataforma, se decide trabajar con el lenguaje de programación Java 8.1.1 ya que su principal facilidad es la de permitir desarrollar aplicaciones para distintas tecnologías (de escritorio, móvil, web...). Posee una librería de clases bastante completa que se puede aprovechar gracias a que es un lenguaje orientado a objetos que corre sobre la máquina virtual.

1.3.4. IDE de desarrollo

Para desarrollar, normalmente solo es necesario un editor de texto, un intérprete o compilador y una terminal de líneas de comando, un Entorno de Desarrollo Integrado (IDE). Esto simplifica el trabajo y ahorra tiempo de desarrollo.

NetBeans 8.0

NetBeans es una aplicación de código abierto diseñada para el desarrollo de aplicaciones portables entre las distintas plataformas. Dispone de soporte para crear interfaces gráficas de forma visual, desarrollo de aplicaciones web, control de versiones, entre otras. Es un programa con Licencia Común de Desarrollo y Distribución (CDDL) que permite su uso libremente, brinda el código del mismo y es totalmente gratuita. Permite escribir, compilar, hacer un debut, ensamblar y desplegar aplicaciones y aunque está escrito en Java, brinda soporte para toda clase de lenguajes de programación. Funciona en sistemas operativos compatibles con la máquina virtual Java (Windows XP, Vista, Windows 7, Ubuntu 9.10, Solaris, Mac OS X 10.5 o superior).

IDE de desarrollo seleccionado

Para el desarrollo de la aplicación informática se seleccionó NetBeans, por permitir escribir, compilar, ensamblar y desplegar aplicaciones, además de brindar soporte para varios lenguajes de programación mediante el empleo de plugins. Facilita el completamiento para los lenguajes de programación HTML, CSS, JavaScript y PHP permitiendo tracear este último.

1.3.5. Sistemas Gestores de Base de Datos (SGBD)

Un SGBD es un conjunto de programas que permiten crear y mantener una Base de datos, asegurando su integridad, confidencialidad y seguridad. Permite almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las aplicaciones más usuales son para la gestión de empresas e

instituciones públicas, además de ser ampliamente utilizado en entornos científicos con el objeto de almacenar la información experimental (Días Roberto, 2010).

PostgreSQL 9.3

PostgreSQL es un SGBD Objeto-Relacional basado en el proyecto Postgres, de la Universidad de Berkeley. Es una derivación libre de este proyecto y utiliza el lenguaje SQL. Fue pionero en muchos de los conceptos del sistema objeto-relacional actual. Este proyecto lleva más de una década de desarrollo, siendo hoy día, el sistema libre más avanzado, soportando la gran mayoría de las transacciones SQL y control concurrente. Entre sus principales ventajas se destacan las siguientes:

- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.
- Posee una gran escalabilidad, haciéndolo idóneo para su uso en sitios Web que atienden un gran número de solicitudes.
- Puede ser instalado un número ilimitado de veces sin temor de sobrepasar la licencia.
- Posee estabilidad y confiabilidad legendarias.
- Es extensible a través del código fuente disponible sin costos adicionales.
- Es multiplataforma, disponible en la mayoría de los sistemas operativos.
- Permite implementar reglas, vistas, disparadores, subconsultas y funciones.
- Posee herramientas para generar SQL portable para compartir con otros sistemas compatibles con SQL.

Gestor de Base de datos seleccionado

Después de analizar las características y prestaciones de los SGBD mencionados anteriormente, se aprecian las ventajas que brinda PostgreSQL. Este sistema provee de gran capacidad de almacenamiento, consistencia, escalabilidad y rendimiento bajo grandes cargas de trabajo. Es un SGBD objeto-relacional, distribuido bajo la licencia BSD13 y su código fuente se encuentra disponible libremente. Por lo tanto, se decide desarrollar solución propuesta en esta investigación haciendo uso del SGBD PostgreSQL en su versión 9.3

Conclusiones del capítulo

Una vez realizado el estudio de la literatura científica, se pudo identificar la ineficiencia de los cuatro sistemas investigados, ya que no se adecua a la necesidad de gestión de recursos humanos en la facultad

CITEC. Por lo que se plantea un sistema informático que garantice las necesidades requeridas, utilizando la metodología de desarrollo de software Open Up y las herramientas y técnicas necesarias para guiar el proceso de desarrollo de software.

Capítulo 2: Descripción del sistema

Introducción

En el presente capítulo se detallan las características del sistema, a partir de la descripción general de la propuesta de solución. Se modela el mapa de procesos del negocio e identifican los requisitos funcionales y no funcionales del sistema con la respectiva descripción de cada uno de ellos y se traza la línea base de la arquitectura del sistema en cuestión.

2.1. Propuesta de solución

A partir de las deficiencias identificadas anteriormente en la Facultad CITEC de la UCI en el proceso de gestión de los Recursos Humanos y para dar solución al problema planteado en la investigación, se propone desarrollar un sistema que permita generar la prenómina de todas las áreas de la facultad teniendo en cuenta cada una de las incidencias individuales y el control de los recursos humanos, permitiendo una trazabilidad de los datos en tiempo real, el cual estará compuesto por los siguientes módulos.

- **Módulo Recursos Humanos:** Permite la gestión de usuarios en el sistema, la creación de las áreas claves y asociar personal a cada una de las áreas, la asignación de roles, almacena información sobre la evolución docente y científica de cada trabajador. Es aquí donde se asocian las incidencias mensuales a cada trabajador y se realiza la evaluación de los trabajadores.
- **Módulo Administración:** Permite generar la prenómina por cada una de las áreas creadas en el sistema y permite la creación de los nomencladores propios de los recursos humanos.
- **Módulo Configuración:** Permite la configuración de alertas a partir de una paleta del sistema y que previenen la ocurrencia de eventos de impacto alto para la organización. Configura además los horarios laborales y a partir de esta crea turnos de trabajo y genera las hojas de firma vinculadas a un área y un horario establecido previamente.
- **Módulo Reporte:** Emite informes del comportamiento de la categoría docente, investigativa y científica por área y de manera individual. Reportes de evaluaciones y de incidencias.

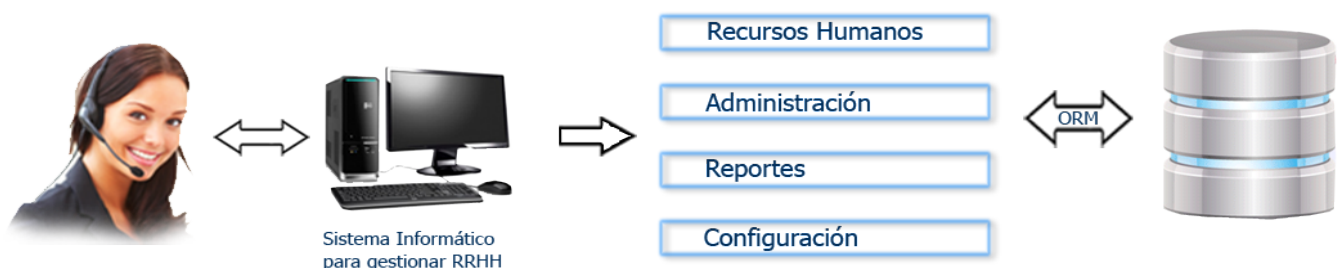


Figura 1: Propuesta de solución.

2.2. Modelo de Negocio

Un Proceso de Negocio es una colección de actividades que, tomando una o varias clases de entradas, crean una salida que tiene valor para un cliente. Estos procesos del negocio se representan a través de los mapas de procesos, los cuales ofrecen una visión general del sistema. Con el Modelo del Negocio se pretende llegar a un mejor entendimiento de la organización donde se va a implantar el producto. Los objetivos principales de esta actividad son entender la estructura y la dinámica de la organización para la cual el sistema va a ser desarrollado (organización y objetivo) y de esta manera poder derivar los requerimientos del sistema necesarios para apoyar la organización objetivo (Días Roberto, 2010).

2.2.1. Actores del negocio

Los actores del negocio son roles que juegan los software, personas, grupos u organizaciones que interactúan con un negocio determinado, en este caso el negocio se desarrolla en la Facultad de ciencias y tecnologías computacionales de la Universidad de las ciencias informáticas y en esta organización los actores del negocio se identifican a continuación:

Tabla 2: Actores del negocio.

Actor	Descripción
Especialista de RRHH	Encargado solicitar todos los datos correspondientes para generar la nómina de los trabajadores.

2.2.2. Trabajadores del negocio

Los trabajadores del negocio son una representación abstracta de los seres humanos o de un sistema que representa los roles del desarrollo de las actividades dentro de un caso de uso del negocio. Siendo la abstracción que ayuda a identificar todas las mejoras en los procesos de negocio.

Permite analizar los efectos que se pueden obtener debido a la automatización de estos procesos para la organización. En la tabla siguiente se muestra la descripción de los trabajadores del negocio para el modelado del sistema.

Tabla 3: Trabajadores del negocio.

Trabajador	Descripción
------------	-------------

Decano.	Encargado de revisar y aprobar los procesos de recursos humanos.
Vicedecano Administrativo.	Encargado controlar y mandar a ejecutar los procesos (visualizar reportes y gestionar componentes).
Técnico de RRHH	Encargado de ejecutar todos los procesos referentes a recursos humano.
Jefe de Area	Encargados de reportar las evaluaciones e incidencias de sus trabajadores.

2.3. Diagrama de casos de uso del negocio

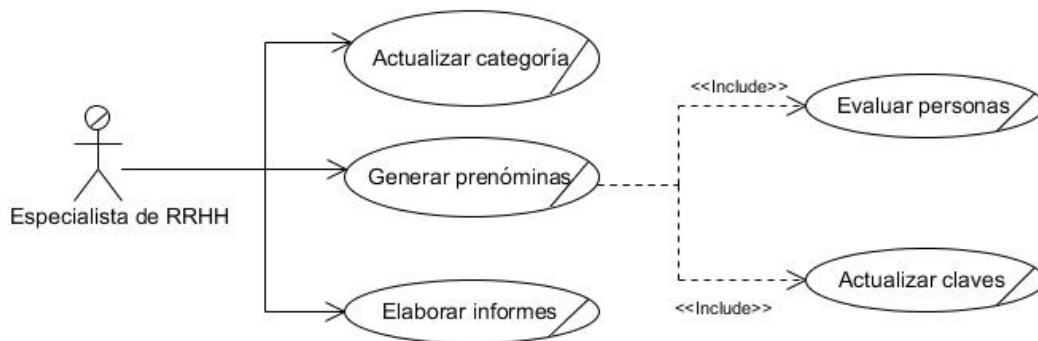


Figura 2: Caso de uso del negocio.

2.4. Diagrama de actividades del modelo de negocio

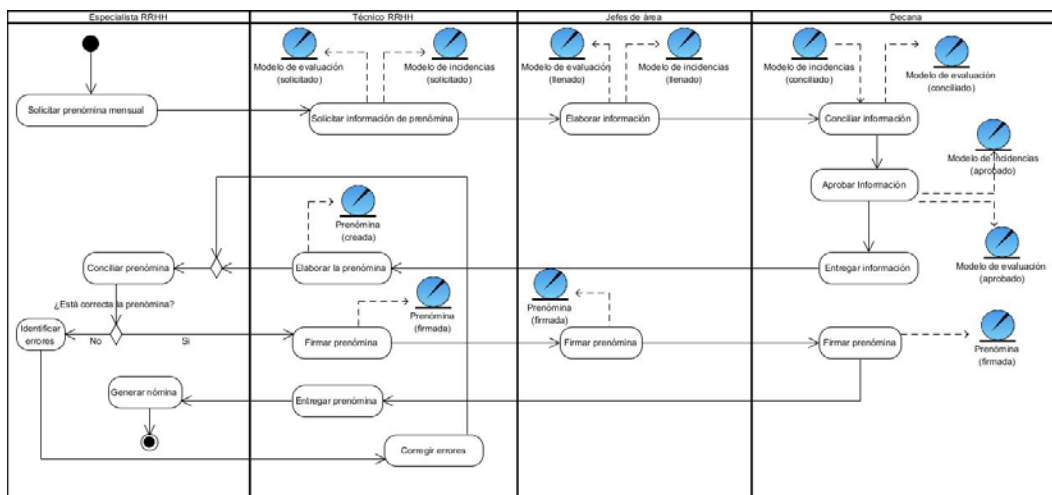


Figura 3: Diagrama de actividades: Generar Prenómina.

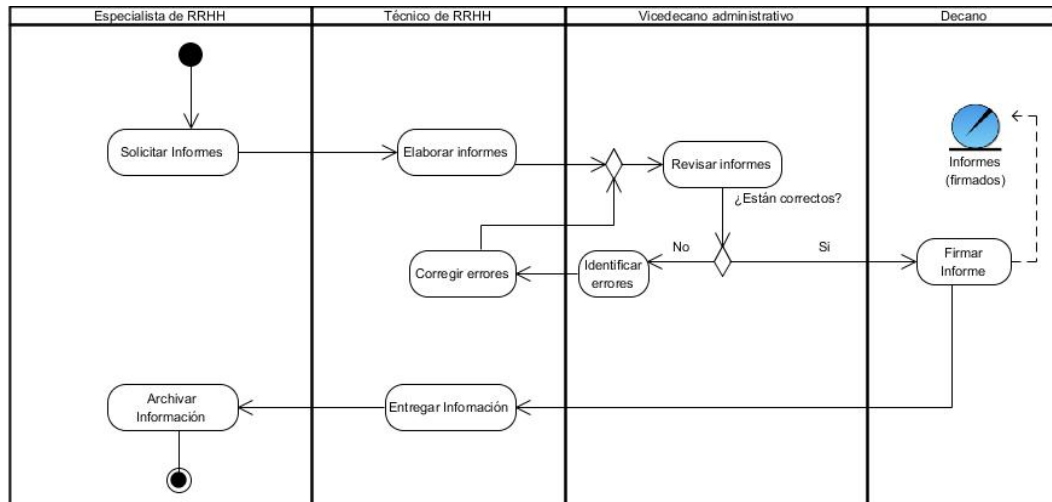


Figura 4: Diagrama de actividades: Elaborar Informes.

2.5. Descripción de los casos de usos del negocio

Tabla 4: Caso de uso del negocio: Generar Prenómina.

Caso de Uso:	Generar Prenómina
Actores:	Especialista RRHH (Inicia)
Trabajadores:	Técnico RRHH, Jefe de Área, Decana.
Propósito:	Generar la prenómina de todas las áreas de la facultad de un mes.
Resumen:	El CUN se inicia cuando el especialista de RRHH solicita la prenómina al técnico de RRHH de la facultad. Luego el técnico de RRHH solicita información para generar la prenómina como son incidencias en el mes y evaluaciones de desempeño a los jefes de departamento. Los jefes de áreas concilian las evaluaciones con la decana y esta aprueba o no. La información aprobada es entregada al técnico de capital humano quien elabora la prenómina y la concilia con el especialista de RRHH. Si todos los datos están correctos se procede a la firma de la misma por todos los responsable jefe de áreas, técnico RRHH y el decano. Sino están correctos los datos se corrige y se firma. Se entrega al especialista de RRHH y este genera la nómina.
Casos de uso asociados:	
Flujo de Trabajo	
Acción del Actor	Respuesta del Negocio
1. El especialista de RRHH pide la prenómina al técnico de RRHH.	2. El técnico de RRHH pide al jefe de área las evaluaciones y las incidencias de sus trabajadores del mes.

	3. El jefe de área elabora la información.
	4. el jefe de área concilia con el decano la información elaborada.
	5. El decano aprueba la información entregada y se la entrega al técnico de RRHH.
	6. El técnico de RRHH elabora la prenómina y concilia con el especialista RRHH.
7. El especialista de RRHH concilia la información.	8. El técnico RRHH con la información correcta procede a la firma de la prenómina.
	9. El jefe de área firma la prenómina.
	10. El decano firma la prenómina.
11. El especialista de RRHH elabora la nómina.	
Prioridad:	Crítico
Mejoras:	
Cursos Alternos	
Acción del Actor	Respuesta del Negocio
7.1) Si el especialista de RRHH ve algún error regresa la información al técnico de RRHH.	
	7.2) El técnico de RRHH corrige el error.
7.3) El especialista de RRHH vuelve a conciliar la información.	
	7.4) Se va a la acción 8 del flujo Normal de eventos.

Tabla 5: Caso de uso del negocio: Elaborar Informe.

Caso de Uso:	Elaborar Informe
Actores:	Especialista RRHH (Inicia)
Trabajadores:	Técnico RRHH, Jefe de Área, Decana.
Propósito:	Generar informes solicitados por el especialista de RRHH
Resumen:	El CUN se inicia cuando el especialista de RRHH solicita la información o un reporte. Luego el técnico de RRHH genera el reporte. Se lo entrega al vicedecano administrativo para que lo revise. Si la información es la correcta se le entrega al decano para que la firme. Sino esta correcta la información el técnico de RRHH la corrige se le entrega a vicedecano administrativo. Se continúa el flujo normal. El técnico de RRHH se le entrega al especialista y este lo archiva.
Casos de uso	

asociados:	
Flujo de Trabajo	
Acción del Actor	Respuesta del Negocio
1. El especialista de RRHH pide la información al técnico de RRHH.	2. El técnico de RRHH elabora el reporte solicitado.
	3. El técnico de RRHH se la entrega al vicedecano administrativo para que la revise.
	4. El vicedecanato administrativo revisa la información.
	5. El vicedecanato administrativo entrega al decano la información correcta para que la firme.
	6. El técnico de RRHH se le entrega al especialista RRHH.
7. El especialista de RRHH archiva la información solicitada.	
Prioridad:	Crítico
Mejoras:	
Cursos Alternos	
Acción del Actor	Respuesta del Negocio
	4.1) Si la información no es correcta el técnico de RRHH corrige la información.
	4.2) Se va a la acción 5 del flujo Normal de eventos.

2.6. Extracción de Requisitos

La especificación de los requisitos de software es una descripción completa del comportamiento del sistema que se va a desarrollar. Contiene requisitos funcionales que son las capacidades o condiciones que el sistema debe cumplir. En el que se incluye un conjunto de casos de uso que describen las interacciones que tendrán los usuarios con el software y contiene, además, requisitos no funcionales que son las propiedades o cualidades que el producto debe tener, que imponen restricciones en el diseño o la implementación. En esta etapa se debe generar una información clara y precisa de los aspectos más relevantes del producto, ya que esta actividad es el hilo conductor de todo el desarrollo del software.

2.6.1. Requisitos funcionales

Los Requisitos Funcionales (RF) definen las condiciones y funciones que el sistema será capaz de realizar. Luego de efectuar las actividades propuestas en la etapa de obtención de la Ingeniería de Requisitos se obtuvieron los siguientes requisitos funcionales:

Autenticar usuario

RF 1. Autenticar usuario.

Gestionar persona

RF 2. Adicionar Persona (Nombre y Apellidos, Número de expediente, Número de solapín, Cargo, Área, Categoría docente, Fecha de categoría, Categoría Científica, Grupo de ocupación, Fecha de ingreso, antigüedad, evaluación anual, militancia, sexo, Dirección, Residente).

RF 3. Modificar persona

RF 4. Eliminar Persona.

RF 5. Listar Persona.

RF 6. Ver persona.

Gestionar Prenómina

RF 7. Crear Prenómina.

RF 8. Generar Prenómina

RF 9. Listar Prenómina

RF 10. Eliminar Prenómina.

RF 11. Modificar Prenómina.

RF 12. Exportar Prenómina.

Gestionar Claves

RF 13. Crear clave.

RF 14. Modificar clave.

RF 15. Eliminar clave.

Generar claves

RF 16. Adicionar Clave

RF 17. Modificar clave. Cambiar clave en la prenómina*

RF 18. Eliminar clave.

RF 19. Listar clave.

Componente evaluación

RF 20. Crear evaluación

RF 21. Asignar evaluación

RF 22. Calcular nocturnidad.

RF 23. Calcular días trabajados.

RF 24. Generar Observación.

Gestionar hojas de firma

RF 25. Adicionar hojas de firma (Nombre y Apellidos, Número de expediente, Cargo, Área , Horario(regular ,Irregular,)

RF 26. Modificar hojas de firma

RF 27. Eliminar hojas de firma.

RF 28. Listar hojas de firma.

RF 29. Generar Hoja de firma

RF 30. Exportar hojas de firma.

RF 31. Calcular Días trabajados.

RF 32. Asignar horario.

Gestionar horario

RF 33. Adicionar horario (Días del mes, sábados, domingos, Feriados, Laborables, festivos).

RF 34. Modificar horario.

RF 35. Eliminar horario.

RF 36. Listar horario.

Generar Reportes

RF 37. Reporte colectivo de categoría docente

RF 38. Reporte de incidencias colectivas

RF 39. Reporte colectivo de evaluaciones

Gestionar turnos

RF 40. Adicionar horario

RF 41. Modificar horario.

RF 42. Eliminar horario.

RF 43. Buscar horario.

RF 44. Generar hoja de firma.

Gestionar alarmas

- RF 45. Adicionar alerta
- RF 46. Modificar alerta.
- RF 47. Eliminar alerta.
- RF 48. Buscar alerta.

2.6.2. Requisitos no funcionales

Los Requisitos no Funcionales (RNF) son propiedades o cualidades, que pueden usarse para juzgar la operación de un sistema en lugar de su funcionalidad. El sistema debe satisfacer las siguientes condiciones o cualidades: que el producto debe tener.

Requerimientos de Software

El Software se debe ejecutar sobre el sistema operativo Linux o Windows.

El lenguaje de programación será Java 8.0.

El IDE de desarrollo será NetBeans.

Como gestor de base de datos se usará PostgreSQL en su versión 9.3.

Requerimientos de Hardware

Estaciones de Trabajo:

- Se requiere tarjeta de red.
- Se requiere que tenga al menos 1 GB de memoria RAM.
- Procesador Dual Core como mínimo.

Servidores:

- Se requiere tarjeta de red.
- Se requiere que tenga al menos 1 GB de RAM.
- Procesador Dual Core como mínimo.

Requerimientos de Restricciones en el diseño y la implementación

- El análisis y diseño del sistema estará basado en la metodología de desarrollo Open Up con el uso del lenguaje de modelado UML 2.0.
- Se utilizará Visual Paradigm 8.0 como herramienta CASE para el modelado de los artefactos que se generan con cada flujo de trabajo.

- La base de datos se implementará en PostgreSQL SQL versión 9.3 o superior.

Requerimientos de Apariencia o interfaz externa

- Diseño perfectamente encuadrado para resoluciones de 800x600, pero preparado para verse en otras resoluciones.
- Para la realización del diseño se tendrá en cuenta el control y la transparencia como eje principal en torno al cual girará el sistema, además debe permitir el uso de colores claros y legibles.

Requerimientos de Usabilidad

- Los usuarios deben tener un conocimiento básico sobre informática para poder trabajar con el software a desarrollar.

Requerimientos de Portabilidad

- El sistema debe ser multiplataforma, es decir la propuesta de solución puede ser manejable en el sistema operativo Linux o Windows.

Requerimientos de Seguridad Confidencialidad

Confidencialidad:

- La información manejada por el sistema deberá estar protegida de acceso no autorizado y divulgación.
- Los usuarios accederán a la información correspondiente a cada uno, ya que el sistema brinda un mecanismo de autenticación que garantiza lo antes planteado.

Integridad:

- La información manejada por el sistema será objeto de cuidadosa protección contra estados inconsistentes y corrupción; a través de roles y permisos de los usuarios autenticados (R-Back).

Disponibilidad:

- A los usuarios autorizados se les deberá garantizar el acceso a la información solicitada en todo momento, mientras que los no autorizados no presentarán los mismos privilegios (no tendrán acceso a la información).

Encriptación:

La contraseña de la base de datos se encuentra protegida, por el algoritmo de encriptación md5.

2.7. Modelo de casos de uso

El modelo de casos de uso del sistema permite que los desarrolladores de software y los clientes lleguen a un acuerdo sobre los requisitos, es decir sobre las condiciones y posibilidades que debe cumplir el sistema. Además, proporciona la entrada fundamental para el análisis, diseño y pruebas.

2.7.1 Actores del sistema

Para el acceso a las funcionalidades que se desarrollarán se cuenta con varios actores, que se definen a continuación.

Tabla 6: Actores del sistema.

Actor	Descripción
Vicedecano Administrativo.	Encargado de la gestión de los usuarios del sistema y de gestionar la información referente a todas las áreas. Puede realizar informes.
Jefe de Área.	Usuario del sistema encargado de evaluar los trabajadores y pueden visualizar la información existente en el mismo, de acuerdo al nivel de acceso que posean.
Técnicos en RRHH	Usuarios del sistema que puede reportar incidencias, gestiona todos los módulos en el sistema.
Decano.	Usuario del sistema puede visualizar toda la información existente en el sistema.

2.8. Diagrama de casos de uso

Un diagrama de casos de uso del sistema (DCU) representa gráficamente a los procesos y su interacción con los actores. Además facilita el entendimiento de los procesos realizados por el sistema para el desarrollador.

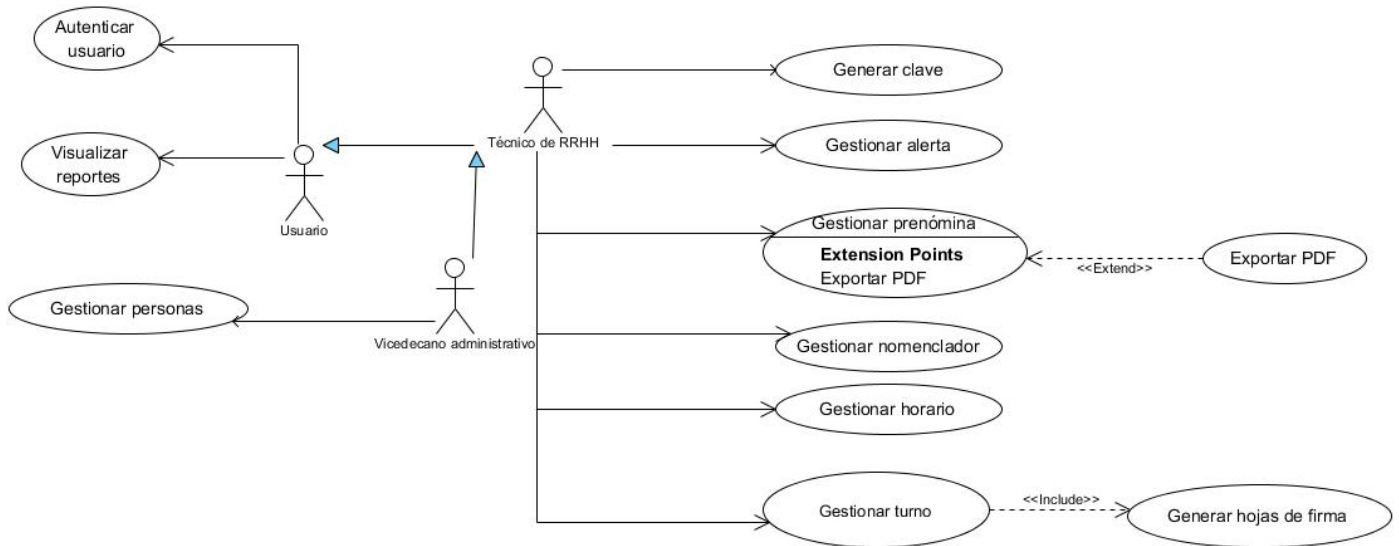


Figura 5: Diagrama de casos de uso del sistema.

2.9. Descripción de casos de uso

Tabla 7: Descripción de caso de uso: Gestionar persona.

Objetivo	Gestionar persona.
Actores	Técnico de RRHH (inicia).
Resumen	El Técnico de RRHH escoge en el sistema una opción en la parte superior del menú para la gestión de persona y se muestra la lista de persona, con un menú de búsquedas y con la posibilidad de adicionar, modificar y eliminar.
Complejidad	Alta.
Prioridad	Crítica.
Precondiciones	Tiene que tener permiso y estar autenticado para realizar las acciones.
Postcondiciones	Se adiciona, modifica, elimina o busca personas.
Flujo de eventos	
Flujo básico	
Actor	Sistema

1.	Selecciona la opción "Persona" en el menú del superior del sistema.	
2.		Muestra un listado con las personas almacenadas y posibilita al técnico de RRHH ejecutar las siguientes acciones: Adicionar persona. Ver Sección 1 Modificar persona. Ver Sección 2 Eliminar persona. Ver sección 3. Buscar persona. Ver sección 4.
3.	El Técnico de RRHH escoge una de las cuatro opciones.	
4.		Finaliza el caso de uso.

Sección 1: "Adicionar persona"

Flujo básico

	Actor	Sistema
1	Selecciona la opción "Adicionar" de la página principal de los criterios.	
2		Muestra una ventana con un formulario de registro con los siguientes campos: Se llenan los campos: <ul style="list-style-type: none"> • Nombre y apellidos: Solo acepta letra. • Número de expediente: solo acepta valores numéricos y es único. • Número de solapin: acepta números y letras y es único. • Cargo: Su valor se gestiona en los nomencladores. • Sexo: solo admite: femenino y masculino. • Área: Su valor se gestiona en los

		<p>nomenccladores.</p> <ul style="list-style-type: none"> • Grupo ocupacional: Su valor se gestiona en los nomenccladores. • Fecha de contrato: Fecha en que se contrato en la facultad. • Militancia: Su valor se gestiona en los nomenccladores. Puede ser vacio • Antigüedad: Solo se acepta números de dos cifras. Puede ser vacio. • Residencia UCI: Si se selecciona la opción Residencia UCI Se habilitan los campos Apartamento y Teléfono. No es obligatorio. • Dirección particular: Si se selecciona la opción Dirección particular se habilitan los campos para introducir la dirección. No es obligatorio. • Observaciones: Acepta todo. Puede ser vacio.
3	Se presiona la opción Adicionar .	
4		<p>Se validan los campos, si ya existe una persona con el número de expediente introducido se muestra el mensaje "El número de expediente pertenece a otra persona".</p> <p>Si hay un campo vacio se muestra un mensaje "Rellene todos los campos de persona".</p>

5	Se presiona el botón Aceptar	
6		Se adiciona a la persona al sistema. Se muestra el mensaje " La persona ha sido adicionada satisfactoriamente".
Flujo alterno		
"Escoge la opción cancelar."		
	Actor	Sistema
5.1	Selecciona la opción "Cancelar", que se encuentra en el borde inferior derecho, de la página principal de la sección "Adicionar persona".	
5.1.1		Finaliza el flujo alterno.
Flujo alterno		
"Escoge la opción Aplicar."		
5.2	Selecciona la opción "Aplicar", que se encuentra en el borde inferior derecho, de la página principal de la sección "Adicionar persona".	
5.2.1		Se adiciona a la persona al sistema. Se muestra el mensaje " La persona ha sido adicionada satisfactoriamente". Se limpian los campos para adicionar una nueva persona.
Sección 2: "Solicita la eliminación del criterio"		
Flujo básico		
	Actor	Sistema

1	Selecciona la opción “Eliminar” de la página principal de las evidencias.	
2		Se muestra una ventana con la lista de criterios.
3	Selecciona el criterio que desee y da clic en el botón “Eliminar”.	
4		Se muestra un mensaje “Desea eliminar el criterio”
5	El Vicedecano Administrativo da clic en el botón “sí”.	
6		Se muestra un mensaje de notificación “Criterio eliminado correctamente”.
Flujo alterno		
“Escoge la opción cancelar.”		
	Actor	Sistema
1	Selecciona la opción “Cancelar”, que se encuentra en el borde inferior derecho, de la página principal de la sección “Eliminar criterio”.	
2		Finaliza el flujo alterno.
Sección 3: “Solicita la actualización del criterio”		
Flujo básico		
	Actor	Sistema
1	Selecciona la opción “Actualizar” de la página principal de los criterios.	
2		Se muestra un formulario (nombre y descripción).
3	El Vicedecano Administrativo modifica los	

	campos del formulario de criterios y da clic en el botón "Aceptar".	
4		Se muestra un mensaje de notificación "Criterio actualizado correctamente".
Flujo alterno		
"Escoge la opción cancelar."		
1	Selecciona la opción "Cancelar", que se encuentra en el borde inferior derecho, de la página principal de la sección "Actualizar criterio".	
2		Finaliza el flujo alterno.
Sección 4: "Solicita la búsqueda del criterio"		
Flujo básico		
	Actor	Sistema
1	Selecciona la opción "Buscar" de la página principal de los criterios.	
2		Permite la búsqueda del criterio por el nombre del mismo (nombre).
3	El Vicedecano Administrativo realiza la búsqueda del criterio y da clic en el botón "Buscar".	
4		Muestra el criterio buscado en caso de existir, sino muestra el mensaje "El criterio no se encuentra registrado".
Requisitos funcionales	no	Seguridad.

Conclusiones del capítulo

La utilización de un modelo de negocio bien definido y la generación de los artefactos del mismo, permitieron conocer, el funcionamiento de todos los procesos que se llevan a cabo en la gestión de los

recursos humanos de la facultad CITEC de la UCI. La descripción de actores y trabajadores que intervienen en el proceso, permitió una mejor comprensión del contexto organizacional y fueron definidos los requisitos funcionales y no funcionales del sistema. Presentando la propuesta de solución mayor organización estructural y organizativa.

Capítulo 3: Análisis y diseño del sistema

Introducción

Está relacionado con el diseño del sistema; se describen los patrones de diseño empleados. Se presentan los diagramas de clases del diseño, diagramas de interacción y diagrama de despliegue del sistema con el objetivo de tener una idea de cómo quedará.

3.1. Modelo del diseño

Con el objetivo de crear una representación de software que proporcione detalles acerca de las estructuras de los datos, las arquitecturas, las interfaces y los componentes del software que son necesarios para implementar el sistema se elaboró un modelo del diseño siguiendo los principios definidos para realizar el diseño por la literatura científica.

3.2. Descripción de patrones arquitectónicos

Los patrones ayudan al arquitecto a definir la composición y el comportamiento del sistema de software, y una combinación adecuada de ellos permite alcanzar los requerimientos de calidad. A continuación se describen los utilizados en el desarrollo de la propuesta de solución.

3.2.1 *Patrones arquitectónicos*

Los patrones arquitectónicos expresan el esquema de organización estructural fundamental para sistemas de software. Provee un conjunto de subsistemas predefinidos, especifica sus responsabilidades e incluye reglas y pautas para la organización de las relaciones entre ellos. Propone que son plantillas para arquitecturas de software concretas, que especifican las propiedades estructurales de una y tienen un impacto en la arquitectura de subsistemas. La selección de un patrón arquitectónico es, por lo tanto, una decisión fundamental de diseño en el desarrollo de un sistema de software (Buschmann F et al. 1996).

Los patrones arquitectónicos:

- Definen la estructura básica de una aplicación.
- Pueden contener o estar contenidos en otros patrones.
- Proveen un subconjunto de subsistemas predefinidos, incluyendo reglas y pautas para su organización.

- Son una plantilla de construcción.

3.2.2 Patrón arquitectónico Modelo – Vista – Controlador

La Arquitectura de Software es la organización fundamental de un sistema encarnada en sus componentes, las relaciones entre ellos y el ambiente y los principios que orientan su diseño y evolución.

La necesidad del manejo de la arquitectura de un sistema de software nace con los sistemas de mediana o gran envergadura, que se proponen como solución para un problema determinado. En la medida que los sistemas de software crecen en complejidad, dado por el número de requerimientos o por el impacto de los mismos, se hace necesario establecer medios para el manejo de esta complejidad. En general, la técnica es descomponer el sistema en componentes que agrupan aspectos específicos del mismo y que al organizarse, de cierta manera constituyen la base de la solución de un problema en particular.

MVC son las siglas de Modelo Vista Controlador, que es un patrón de arquitectura de software cuya función es subdividir una aplicación en tres módulos que corresponden a la vista del usuario (la interfaz a la que accede el usuario), una lógica de control para captar los eventos que el usuario ha generado a través de la interfaz, y un modelo que gestiona los datos según le indique la lógica de control.

Modelo: Esta es la representación específica de la información con la cual el sistema opera y se compone por el Sistema de Gestión de Base de Datos y la lógica de negocio. La lógica de negocio asegura la integridad de estos y permite derivar nuevos datos. El Sistema de Gestión de Base de Datos (SGBD) será el encargado de almacenar los cambios en los datos (agregar datos, editarlos o borrarlos) producidos por la lógica de negocio; ejemplos de SGBD son MySQL, Oracle.

Vista: Este presenta el modelo en un formato adecuado para interactuar, usualmente la interfaz de usuario. Por lo tanto, la vista es la encargada de presentar los datos al usuario y la interfaz necesaria para modificarlos. Un ejemplo de tecnología podría ser las JSP que, mediante el servidor, muestra los datos y los formularios que constituyen la vista para que pueda interactuar con la aplicación.

Controlador: Este responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista. Por lo general, el controlador sería la unidad central que comunica la vista con el modelo y viceversa, asociando los eventos del usuario con los cambios que se producirán en el modelo y devolviendo los datos resultantes que genere el modelo a la vista que corresponda.

El patrón arquitectónico Modelo – Vista – Controlador (MVC) divide una aplicación interactiva en tres componentes. El “modelo” contiene la información central y los datos. Las “vistas” despliegan información al usuario. Los “controladores” capturan la entrada del usuario.

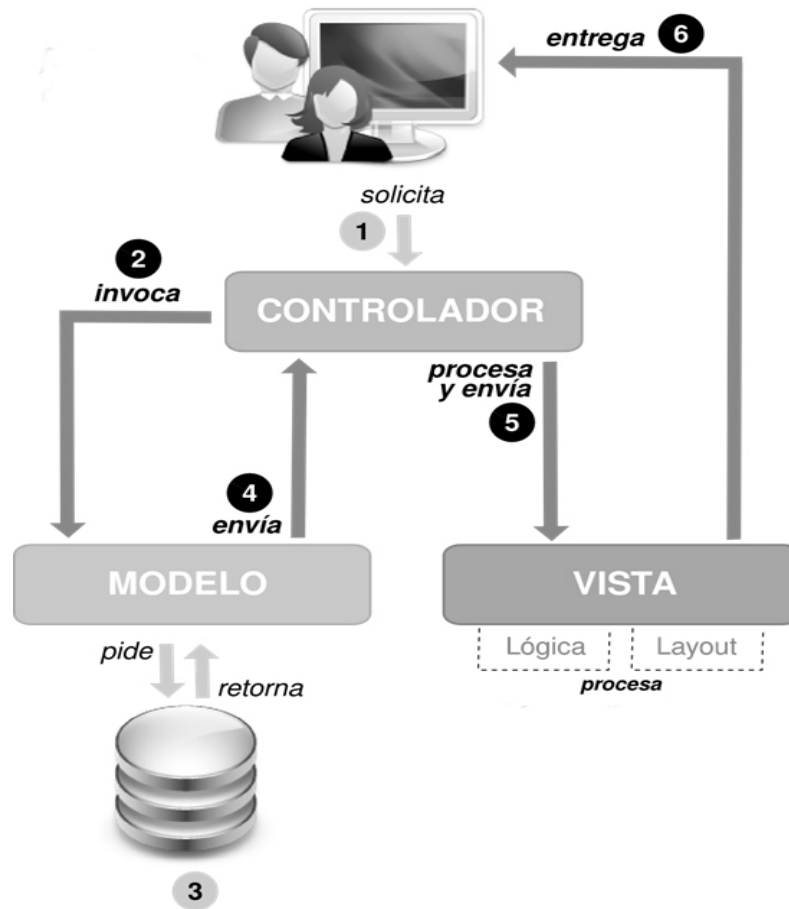


Figura 6: Patrón arquitectónico Modelo Vista Controlador.

Aunque se pueden encontrar diferentes implementaciones de MVC, el flujo que sigue el control generalmente es el siguiente:

- El usuario interactúa con la interfaz de usuario.
- El controlador recibe (a través de la interfaz) la notificación de la acción solicitada por el usuario. Es decir, el controlador gestiona el evento que llega desde la vista producida por un usuario.
- El controlador accede al modelo, ya sea con el fin de consultar datos o actualizarlos, posiblemente modificándolo de forma adecuada a la acción solicitada por el usuario.

- El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se refleja los cambios en el modelo. El modelo no debe tener conocimiento directo sobre la vista. Por lo general, el controlador no pasa objetos de dominio (el modelo) a la vista, aunque puede dar la orden a la vista para que se actualice. Sin embargo, en algunas implementaciones, la vista no tiene acceso directo al modelo, dejando que el controlador envíe los datos del modelo a la vista.
- La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.

3.3. Patrones de diseño

Un patrón de diseño provee un esquema para refinar los subsistemas o componentes de un sistema de software, o las relaciones entre ellos. Describe la estructura comúnmente recurrente de los componentes en comunicación, que resuelve un problema general de diseño en un contexto particular (Buschmann F et al. 1996).

3.3.1 Patrones GRASP

Los Patrones de Principios Generales para Asignar Responsabilidades (GRASP por sus siglas en inglés) describen los principios fundamentales del diseño de objetos y la asignación de responsabilidades, expresados como patrones (Cardoso E, 2004).

- **Experto:** Este patrón es muy utilizado para asignar responsabilidades. Una clase contiene la información necesaria para llevar a cabo sus funcionalidades. Permite conservar el encapsulamiento, debido a que los objetos se valen de su propia información para cumplir lo que se le pide. Proporciona que las clases cuenten con la funcionalidad requerida, brindando así una alta cohesión.

Esto se aplica en sistema en clases como: Fecha.java.


```

public Date FechaActualDate() throws ParseException {
    java.util.Date fecha = new Date();
    SimpleDateFormat sd = new SimpleDateFormat("yyyy/MM/dd");
    String fe = sd.format(fecha);
    return sd.parse(fe);
}

public String FechaActualString() {
    java.util.Date fecha = new Date();
    SimpleDateFormat sd = new SimpleDateFormat("yyyy/MM/dd");
    return sd.format(fecha);
}

```

Figura 7: Ejemplo de clase que usa el patrón experto.

- **Creador:** Asignarle a una clase la responsabilidad de crear una instancia de otra. Dentro del sistema este patrón se evidencia en las acciones de los controladores, las cuales crean objetos del modelo o los formularios que representan las entidades.

Se evidencia en clases como: DocumentoHojaFirma.java donde se crea una nueva instancia para crear un documento.

```

public void Exportar(Persona persona, Date fecha) throws FileNotFoundException {
    try {
        Document documento = new Document(PageSize.A4, 25, 25, 50, 50);
    }
}

```

Figura 8: Ejemplo de clase que usa el patrón creador.

- **Bajo Acoplamiento:** Determina el nivel de dependencia de una clase con respecto a otras. Una clase con bajo acoplamiento no depende de muchas otras. Es muy útil para clases que cambian constantemente y para obtener un grado alto de reutilización. Con este patrón no se afectan por cambios en otros componentes ya que están por separado.

Se utiliza como ejemplo en la siguiente clase: documentosDibujados.java. Esta clase instancia a la clase PdfpCell.java, que utiliza la librería lowagie.

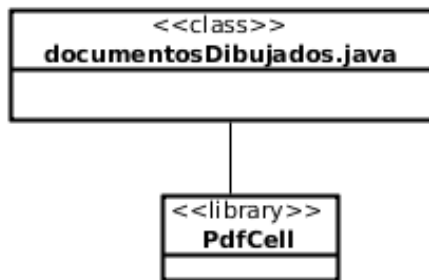


Figura 9: Ejemplo de clase que usa el patrón Bajo acoplamiento.

Controlador: Es el encargado de asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase. Se evidencia el uso de este patrón, ya que para cada petición o evento que se genere en el mismo, existe un controlador con la responsabilidad de obtenerla y devolver una respuesta. La respuesta puede ser mostrar una vista, ejecutar un método, devolver un mensaje.

Este patrón se ve reflejado con el uso del método `showMessageDialog` de la clase `JOptionPane.java`.

```

if (l.Leer().equals("Alerta")) {
    JOptionPane.showMessageDialog(this, "No existe el fichero de la Base de Datos", "Atención", JOptionPane.ERROR_MESSAGE);
} else {
    jTextField2.setText(l.Leer());
}

```

Figura 10: Ejemplo de clase que usa el patrón Controlador.

- **Alta Cohesión:** Este patrón se encarga de guiar el diseño. Permite que las clases del diseño realicen las funcionalidades necesarias para cumplir con su responsabilidad. Mejora la claridad y facilidad para entender el diseño. Busca soluciones para asignar los métodos a las clases de forma coherente, completa y relacionada, permitiendo el cambio, poniendo toda la información que se necesita controlar a la vista en el mismo fichero. Fomenta la reutilización.

De este patrón se encuentran los bean. Ejemplo `área.java`, `incidencia.java`.

3.3.2 Patrones Gof

Los patrones GoF (Gang of Four o “Pandilla de los Cuatro” en español), describen las formas comunes en que diferentes tipos de objetos pueden ser organizados para trabajar unos con otros. Tratan la relación Entre clases, la combinación clases y la formación de estructuras de mayor complejidad. Nos permiten crear grupos de objetos para ayudarnos a realizar tareas complejas.

Compositor (Composite): Patrón de tipo estructura. La jerarquía de clases AWT se ha diseñado según el patrón compositor trata objetos simples y compuestos de manera uniforme. Sirve para diseñar clases que agrupen a objetos complejos, los cuales a su vez están formados por objetos complejos o simples. En este caso se utilizó el patrón en las vistas del sistema, ya que se utiliza la librería de contenedores Swin.

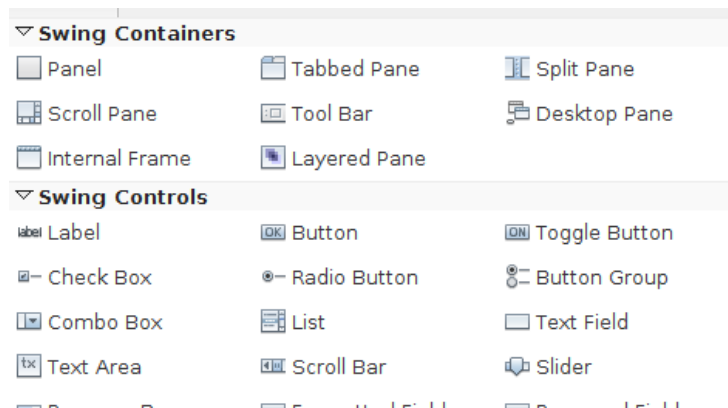


Figura 11: Ejemplo de clase que usa el patrón Compositor.

3.4. Diagrama de clases del diseño

Los diagramas de clases muestran cómo se lleva a cabo la colaboración entre las clases para dar cumplimiento a un requisito determinado. A continuación, se muestra el diagrama de clases del diseño del caso de uso del sistema gestionar usuario y gestionar componentes.

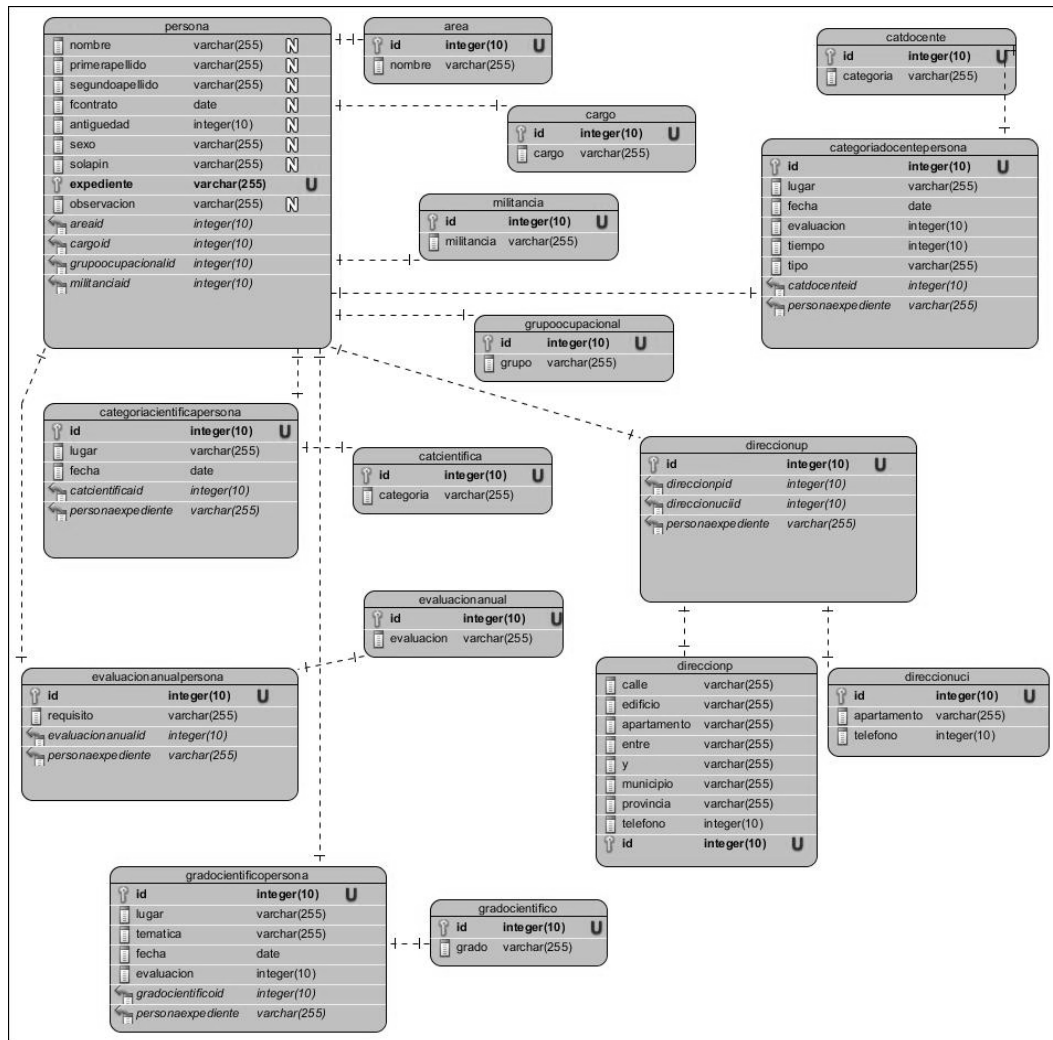


Figura 13: Modelo de datos.

3.6. Diagrama de componentes

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Se realizan con el objetivo de poseer una vista de forma general del sistema a partir de las dependencias e integraciones de los componentes y módulos.

- En el paquete controlador se encuentran las clases controladoras, encargadas de manejar las peticiones de los usuarios a través de los métodos que tienen implementados.

- El paquete modelo agrupa las entidades del sistema, a través de las cuales se realiza el acceso a la base de datos y el paquete vista, agrupa los archivos que permiten visualizar las respuestas que devuelven los controladores al usuario.

A continuación se muestra el diagrama de componentes donde se encuentran los principales componentes del sistema a desarrollar.

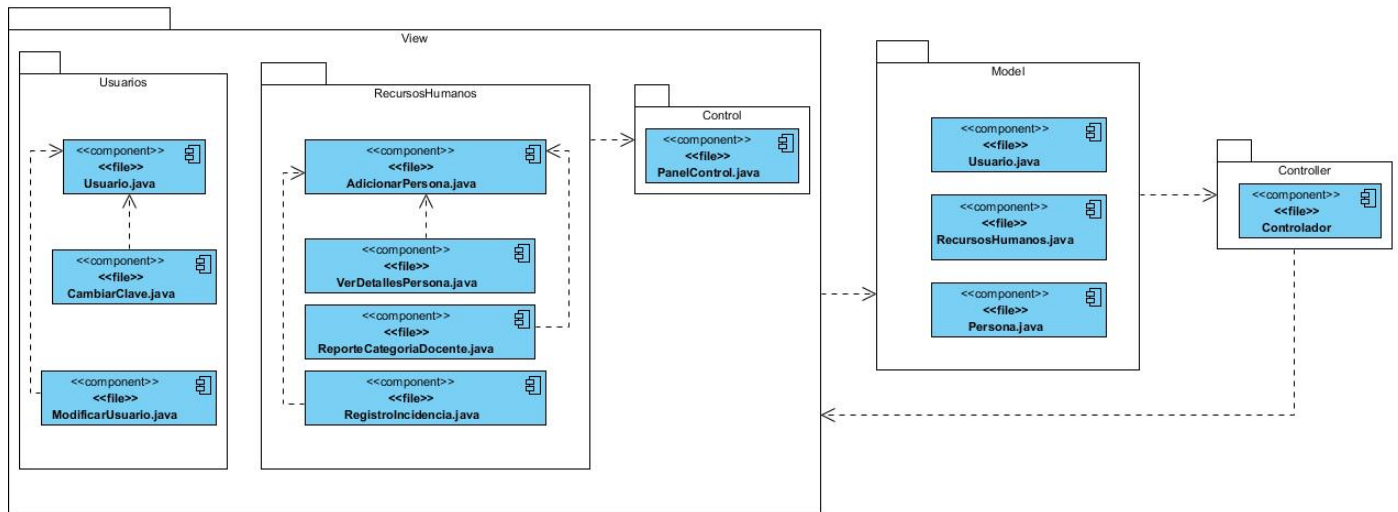


Figura 14: Diagrama de componentes.

3.7. Diagrama de despliegue

El diagrama de despliegue muestra la configuración de los nodos de procesamiento en tiempo de ejecución, los vínculos de comunicación entre ellos y las instancias de los componentes y objetos que residen en ellos. Está compuesto por nodos, dispositivos y conectores. El propósito del modelo de despliegue es capturar la configuración de los elementos de procesamiento y las conexiones entre estos elementos en el sistema.

PC Cliente: Representa las computadoras clientes que se conectan al servidor de aplicaciones, las mismas se comunican con el servidor a través del protocolo TCP/IP.

Servidor de aplicaciones: Representa el servidor donde se encuentra instalada la aplicación. Este accede al servidor de Base de Datos para el manejo de la información mediante el protocolo TCP/IP.

Servidor de Base de datos: Es donde se almacena toda la información de la aplicación.

Impresora: Es el dispositivo que se encarga de la impresión de los documento como hojas de firma, prenómina y reportes generados por el sistema.

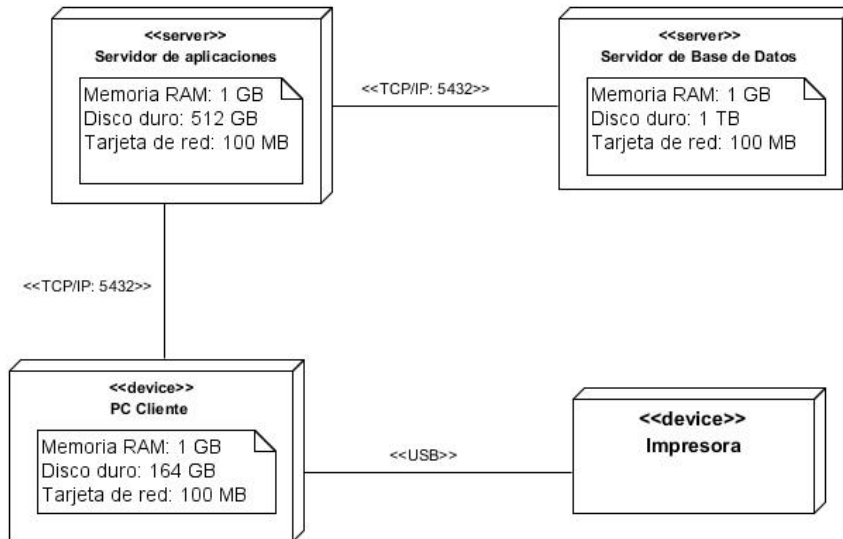


Figura 15: Diagrama de despliegue

Conclusiones del capítulo

La arquitectura definida permitió realizar un diseño robusto y complejo, conforme a las buenas prácticas seguidas por los patrones seleccionados, donde se evidenció la integración de los componentes Modelo Vista Controlador y dichos patrones, mostrando el alto grado de flexibilidad que posee el sistema. Se arribó a que la propuesta de solución, Sistema para la gestión de los recursos humanos, es factible y brinda información valiosa relacionada al control de la organización; que permitió consolidar las bases y principios que guiarán el proceso de implementación y estructuración del sistema.

Capítulo 4: Implementación y Análisis de los Resultados

Introducción

La fase de implementación en el desarrollo de un producto de software, es el mecanismo donde se ponen en práctica todas las descripciones y arquitecturas propuestas en las fases de análisis y diseño, es el complemento del trabajo de las fases que lo preceden dentro del proceso de desarrollo de software. La implementación ofrece una materialización precisa de los requisitos.

Una de las últimas fases del ciclo de vida antes de entregar un software para su explotación es la fase de pruebas, cuyo objetivo es comprobar si este cumple sus requisitos. Dentro de la fase de pruebas pueden desarrollarse varios tipos de pruebas en función de los objetivos de las mismas y de su importancia.

4.1. Código fuente

Para obtener una versión funcional de la aplicación se deben implementar los componentes que se han definido, como resultado se obtienen archivos que contienen el código fuente de la aplicación. El código fuente de un software es un conjunto de líneas de texto que son las instrucciones que debe seguir la computadora para ejecutar dicho programa. Por tanto en el código fuente de un programa está escrito su funcionamiento. Estas instrucciones son escritas en un lenguaje de programación que consiste en un conjunto de símbolos, reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos (Lasso, Iván 2008).

4.1.1 Estándares de codificación

Un estándar de codificación completo comprende todos los aspectos de la generación de código. Si bien los programadores deben implementar un estándar de forma prudente, éste debe tender siempre a lo práctico. Un código fuente completo debe reflejar un estilo armonioso, como si un único programador hubiera escrito todo el código de una sola vez.

Para facilitar el entendimiento del código y fijar un modelo a seguir, se establecieron estándares de codificación. A continuación se muestran algunos de estos estándares utilizados en la implementación del sistema.

4.1.2 Nombres de clases y métodos

Para la definición de las clases y métodos en el código del sistema informático fueron utilizados los estándares de codificación CamelCase y UpperCamelCase. Estos son estilos de escritura que se aplica a frases o palabras compuestas.

```
public function indexAction() { ...9 lines }
public function pdfAction() { ...14 lines }
public function mostrarAction() {
    try { ...12 lines } catch (\Exception $exc) { ...7 lines }
}
public function createAction(Request $request) { ...21 lines }
public function editarAction(Request $request) { ...24 lines }
public function borrarAction() { ...16 lines }
public function evaluarAction() {
```

Figura 16: Estándar de codificación CamelCase

```
class NominaController extends Controller {
```

Figura 17: Estándar de codificación UpperCamelCase.

4.1.3 Estructura

- El código debe usar cuatro espacios en vez de usar el tabulado. Esto minimiza problemas con otras herramientas de desarrollo.
- Las líneas deben tener 80 caracteres o menos, evitando tener más de 120 caracteres.
- Las llaves de apertura en las clases deben ir en la siguiente línea y la llave de cierre debe ir en la siguiente línea después del cuerpo.
- Las llaves de apertura, en los métodos de las estructuras de control debe ir en la misma línea y las llaves de cierre deben de ir después del cuerpo.
- Los paréntesis en las estructuras de control no deben usar espacios antes o después.
- Añadir un solo espacio después de cada limitador de coma.
- Añadir un solo espacio alrededor de los operadores (==, &&,...).

- Usa llaves para indicar el control de la estructura sin tener en cuenta el número de declaraciones que el grupo pueda contener.
- Definir una clase por fichero.
- Declarar las propiedades de clase antes que los propios métodos de clase.

4.2. Pantallas principales de la aplicación

La interfaz de una aplicación permite el flujo de información entre el usuario y el sistema. A continuación se muestran algunos ejemplos de las interfaces del sistema desarrollado:

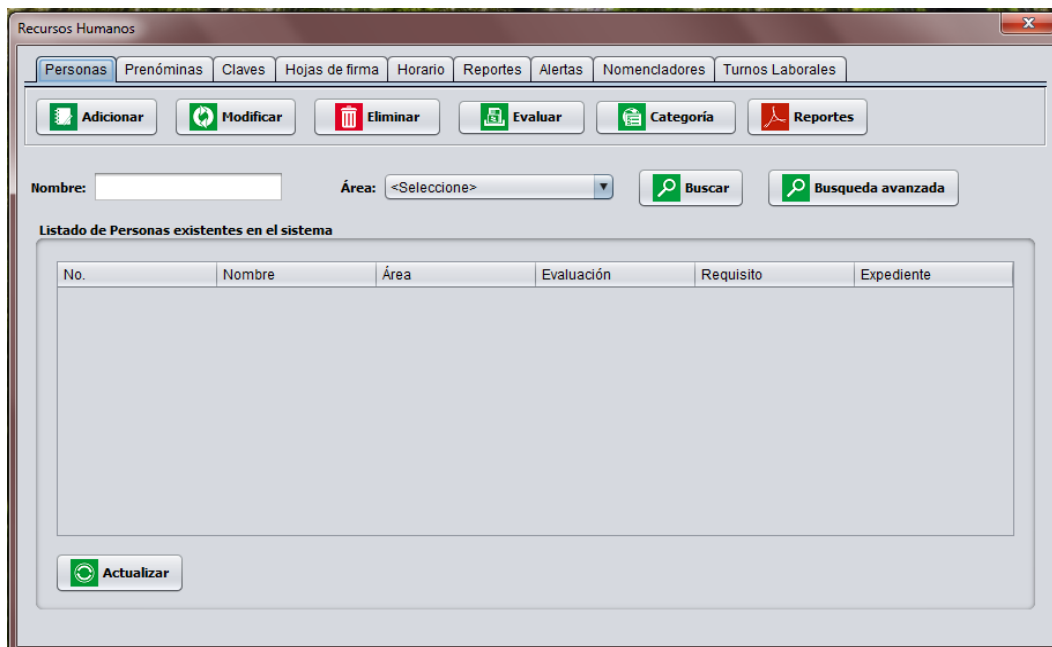


Figura 18: Página principal de Recursos Humanos: Persona.

Adicionar Persona

Datos Generales

Nombre(s): No. expediente:

Primer apellido: Sexo: **femenino**

Segundo apellido: Cargo: **Decano**

Área: **Departamento de tecnología** Grupo ocupacional: **Grupo A**

Fecha de contrato: Solapín:

Antigüedad: Militancia: **UJC**

Residencia UCI Dirección Particular

Apartamento: Calle: Edificio: Apartamento:

Teléfono: Entre: y: Teléfono:

Municipio: Provincia:

Observaciones

Figura 19: Página principal de Adicionar Persona: Persona.

Recursos Humanos

Personas Prenóminas

Nombre:

Listado de Personas existentes

No.	N
1	
2	
3	
4	
5	
6	

Categoría Docente

Profesor: Yulina Hernández González Expediente: 2

Evolución del grado científico

Grado científico	Lugar	Temática	Fecha	Evaluación

Evolución de la categoría docente

Categoría	Lugar	Fecha	Evaluación	Tiempo de la ca...	Tipo

Evolución de la categoría científica

Categoría científica	Lugar	Fecha

Figura 20: Página principal de Agregar Categoría: Categoría.

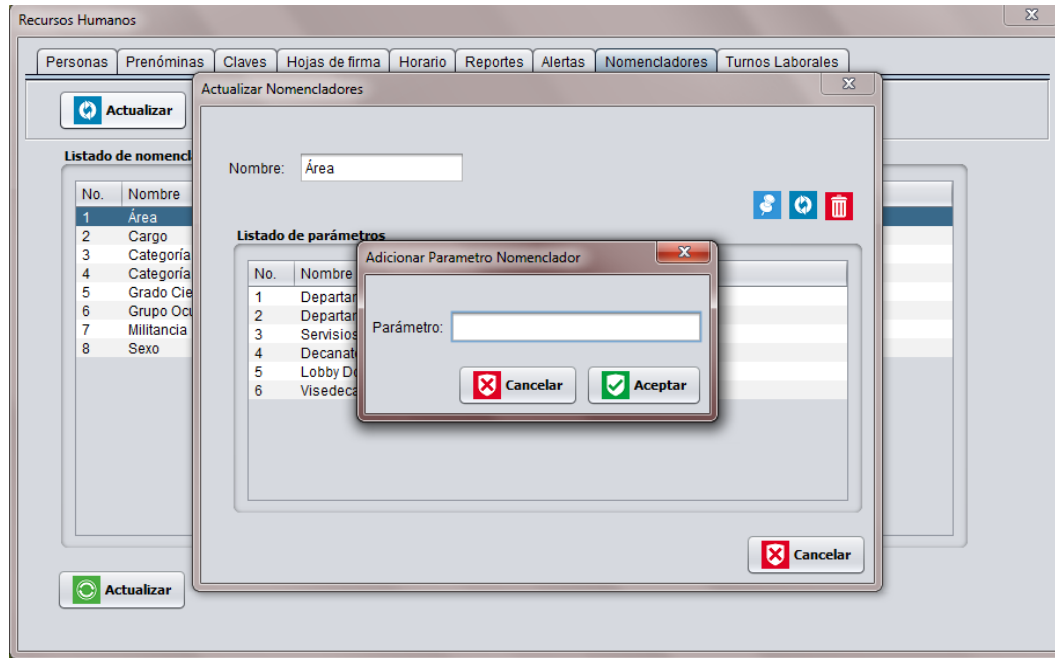


Figura 21: Página principal de Agregar Nomencladores: Nomencladores.

Validación del sistema

Una vez terminada la implementación del producto que se requiere es necesario realizarle pruebas con el objetivo de detectar errores en la aplicación y la documentación; este proceso resulta de gran importancia ya que da una medida de la calidad del mismo, siempre que se lleve a cabo de la forma correcta. A continuación se muestran las pruebas realizadas al sistema y los resultados obtenidos por cada una.

El proceso de pruebas se centra en los procesos lógicos internos del software, asegurando que todas las sentencias se han comprobado, y en los procesos externos funcionales, es decir, la realización de las prueba para la detección de errores. Además son utilizadas para identificar posibles fallos de implementación, calidad o usabilidad de un programa (Pressman, 2010).

4.3. Niveles de Prueba

Los niveles de prueba son diferentes ángulos de verificar y validar en determinados momentos el ciclo de vida del software. Existen diferentes niveles de pruebas como: pruebas funcionales, de sistema y de

aceptación. En el desarrollo de la fase de pruebas del sistema para determinar el índice de control organizacional se aplicarán las pruebas que abarcan el siguiente nivel:

- **Desarrollador:** Se realiza con el objetivo de detectar errores en la implementación de requerimientos.

4.4. Tipos de Prueba

Existen diferentes tipos de pruebas que se pueden aplicar para comprobar el correcto funcionamiento de la aplicación, se seleccionaron los que a continuación se muestran:

- **Pruebas de Funcionalidad:** Se asegura el trabajo apropiado de los requisitos funcionales, incluyendo la navegación, entrada de datos, procesamiento y obtención de resultados.

4.5. Métodos de Prueba

4.5.1 Pruebas funcionales de caja negra

Las pruebas de caja negra, también denominadas pruebas funcionales se centran en los requisitos funcionales del software, es decir, la prueba de caja negra permite al ingeniero de software obtener los conjuntos de condiciones de entrada, que ejerciten completamente todos los requisitos funcionales de un programa.

Por ello se denominan pruebas funcionales, y el probador se limita a suministrarle datos como entrada y estudiar la salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro.

- Las pruebas funcionales que se realizarán a la solución, estarán enfocadas o dirigidas a los casos de uso del sistema para verificar su correcto funcionamiento.
- En este tipo de pruebas se ejecutarán los distintos servicios prestados con datos correctos e incorrectos; en caso de que los datos sean incorrectos se verificará que los mensajes de error sean los deseados y en el caso opuesto que los resultados sean los esperados (Manas,1994).

A continuación se muestran los Diseños de Casos de prueba Adicionar persona, Generar reporte de evaluación individual y Generar reportes de categorías colectivas.

Tabla 8. DCP Adicionar Persona

Escenario	Nombre	Expediente	Solapin	Cargo	Antigüedad	Dirección	Descripción	Respuesta del sistema	Flujo Central
EC 1.1 Adicionar persona correctament e aplicando el adicionar	V(Karen)	V(12356)	V(T114444)	V(Técnico general)	V(8)	V(Ave 51 e/ 206 y 208 #20607)	V(Se adiciona una persona se presiona el botón aceptar)	Se muestra el mensaje " <i>La persona ha sido adicionada satisfactoriamente</i> ". Se cierra el formulario.	1. Se selecciona la opción Componente persona. 2. Seleccione la opción Persona del menú de opciones. 3. Selecciona la opción Adicionar. 4. Se llenan los campos: Se presiona la opción Adicionar. Se validan los campos, Se adiciona a la persona al sistema.
EC 1.2 Adicionar persona correctament e aplicando el aplicar	V(Joanny)	V(12369)	V(T114562)	V(Técnico)	V(10)	V(Calle 4ta e/ laqueruela y gertrudis)	V(Se aplica los cambios a la persona se presiona el botón Aplicar)	Se limpian los campos para adicionar una nueva persona.	1. Se selecciona la opción Componente persona. 2. Seleccione la opción Persona del menú de opciones. 3. Selecciona la opción Adicionar. 4. Se llenan los campos: Se presiona la opción Aplicar. Se validan los campos, <i>Se aplican los cambios a la persona al sistema.</i>

EC1.3 Adicionar persona incorrectamente.	V(Joan)	V(12369)	V(T114458)	V(Especialista general)	V(12)	V(Calle 3ra #331 e/ 33 y 35)	Se muestra un mensaje según el error cometido.	Si ya existe una persona con el número de expediente introducido se muestra el mensaje <i>"El número de expediente pertenece a otra persona"</i> .	<ol style="list-style-type: none"> 1. Se selecciona la opción Componente persona. 2. Seleccione la opción Persona del menú de opciones. 3. Selecciona la opción Adicionar. 4. Se llenan los campos: Se presiona la opción Adicionar. Se validan los campos.
	V(Karen5)	V(14583)	V(114447)	V(Técnico general)	V(2)	V(Calle 2da #67e/ 6 y 8)	Se muestra un mensaje según el error cometido.	Si el nombre contiene números no es válido se muestra el mensaje <i>"El nombre contiene número no es válido "</i> .	
		V(14583)	V(114447)	V(Técnico general)	V(2)	V(Calle 2da #67e/ 6 y 8)	Se muestra un mensaje según el error cometido.	Existen datos vacios se muestra el mensaje <i>"Hay campos vacios rellénelos "</i> .	
EC1.4 Adicionar persona correctamente aplicando el Cancelar	NA	NA	NA	NA	NA	NA	NA (Se presiona el botón Cancelar)	Se cancela la persona y se cierra la ventana.	<ol style="list-style-type: none"> 1. Se selecciona la opción Componente persona. 2. Seleccione la opción Persona del menú de opciones. 3. Selecciona la opción Adicionar. 4. Se llenan los campos: Se presiona la opción Cancelar. Se cierra el formulario.

Tabla 9. DCP Generar reportes de categorías colectivas

Escenario	Subordinación	Fecha Desde	Fecha hasta	Descripción	Respuesta del sistema	Flujo Central
EC 1.1 Hacer reporte de categoría docente colectivo correctamente	V(Omar Mar Cornelio)	V(12-02-17)	V(23-03-17)	V(Se busca el reporte por los datos y se presiona buscar)	Se muestra una tabla con los reportes de las categorías. Se presiona exportar y se exporta el reporte hecho en un documento Excel. Se presiona cerrar y se cierra la ventana.	<ol style="list-style-type: none"> 1. Seleccione la opción componente persona. 2. Seleccione la opción reportes del árbol de la izquierda. 3. Seleccione la opción de categoría docente colectivo. 4. Seleccione el botón reportes. 5. Se llena el campo por donde se desea hacer el reporte. 6. Se llenan los campos de la fecha desde cuando hasta cuando se desea ver la categoría. 6.1 La opción de fecha se va a tomar del almanaque de la computadora. 7. Se presiona el botón generar.
EC 1.2 Hacer reporte de categoría docente colectivo incorrectamente	V()	V(12-02-17)	V(23-03-17)	V(Se busca el reporte por los datos y se presiona buscar)	Si todos los campos están vacíos se muestra un mensaje "Rellene los campos para realizar el reporte".	<ol style="list-style-type: none"> 1. Seleccione la opción componente persona. 2. Seleccione la opción reportes del árbol de la izquierda. 3. Seleccione la opción de categoría docente colectivo. 4. Seleccione el botón reportes.
	V(Yadira Barroso)	V(22-05-17)	V()	V(Se busca el reporte por los datos y se presiona buscar)	Los campos de fecha no pueden estar vacío se muestra un mensaje de "Rellene los campos de fecha para realizar el	<ol style="list-style-type: none"> 5. Se llena el campo por donde se desea hacer el reporte. 6. Se llenan los campos de la fecha desde cuando hasta cuando se desea ver la categoría. 6.1 La opción de fecha se va a tomar del almanaque

					reporte".	de la computadora. 7. Se presiona el botón generar.
EC1.3 Hacer reporte de categoría docente colectivo presionando el botón cancelar.	NA	NA	NA	NA (Se presiona el botón Cancelar)	Se cancela el reporte y se cierra la ventana.	<ol style="list-style-type: none"> 1. Selecciono la opción componente persona. 2. Seleccione la opción reportes del árbol de la izquierda. 5 Selecciona la opción de categoría docente colectivo. 6 Selecciona el botón reportes. 7 Se llena el campo por donde se desea hacer el reporte. 8 Se llenan los campos de la fecha desde cuando hasta cuando se desea ver la categoría. 8.3La opción de fecha se va a tomar del almanaque de la computadora. 9 Se presiona el botón generar.

Tabla 10. DCP Generar reporte de evaluación individual

Escenario	Nombre y Apellido	Fecha Desde	Fecha hasta	Descripción	Respuesta del sistema	Flujo Central
EC 1.1 Hacer reporte de evaluación individual correctamente	V(Bárbara Bron Fonseca)	V(12-02-17)	V(23-03-17)	V(Se busca el reporte por los datos y se presiona buscar)	Se muestra una tabla con los reportes de las evaluaciones. Se puede ver el modelo de evaluaciones presionando e botón Ver Evaluaciones. Se presiona exportar y se exporta el reporte hecho en un documento Excel.	<ol style="list-style-type: none"> 1. Se selecciona la opción componente persona. 2. Seleccione la opción reportes del árbol de la izquierda. 3. Selecciona la opción de Evaluaciones individuales. 4. Se selecciona el trabajador. 5. Se llenan los campos de la fecha desde cuando hasta cuando se desea ver la evaluación. 5.1 La opción de fecha se va a tomar del almanaque de la computadora. 6. Se presiona el botón Generar. 7. Se presiona cerrar y se cierra la ventana.
EC 1.2 Hacer reporte de evaluación individual incorrectamente	V()	V(22-05-17)	V(23-08-17)	V(Se busca el reporte por los datos y se presiona buscar)	Si el campo Nombre está vacío se muestra el mensaje: <i>"Debe introducir el nombre del trabajador"</i> .	<ol style="list-style-type: none"> 1. Se selecciona la opción componente persona. 2. Seleccione la opción reportes del árbol de la izquierda. 3. Selecciona la opción de Evaluaciones individuales. 4. Se selecciona el trabajador. 5. Se llenan los campos de la fecha desde cuando hasta cuando se desea ver la evaluación. 5.1 La opción de fecha se va a tomar del almanaque de la computadora. 6. Se presiona el botón Generar. 7. Se presiona cerrar y se cierra la ventana.
	V(Liudmila de la Osa Iglesias)	V()	V(13-02-17)	V(Se busca el reporte por los datos y se presiona buscar)	Los campos de fecha desde no pueden estar vacío se muestra un mensaje de <i>"Rellene los campos de fecha desde para realizar el reporte "</i> .	
EC1.3	NA	NA	NA	NA (Se presiona	Se cancela el reporte y se	<ol style="list-style-type: none"> 1. Se selecciona la opción componente persona.

<p>Hacer reporte de evaluación individual presionando el botón cancelar.</p>				<p>el botón Cancelar)</p>	<p>cierra la ventana.</p>	<ol style="list-style-type: none"> 2. Seleccione la opción reportes del árbol de la izquierda. 3. Selecciona la opción de Evaluaciones individuales. 4. Se selecciona el trabajador. 5. Se llenan los campos de la fecha desde cuando hasta cuando se desea ver la evaluación. <ol style="list-style-type: none"> 5.1. La opción de fecha se va a tomar del almanaque de la computadora. 6. Se presiona el botón Generar. 7. Se presiona cerrar y se cierra la ventana.
---	--	--	--	---------------------------	---------------------------	---

4.5.2 Resultados e interpretación de las pruebas

El sistema implementado cuenta con un total de 48 requisitos funcionales, para probar cada uno de estos requisitos fue necesario elaborar un diseño de casos de prueba. Cada uno de las No Conformidades (NC) detectadas fueron registradas en el registro de NC del expediente de proyecto de la UCI y fueron debidamente clasificadas.

Se realizaron un total de 3 iteraciones de pruebas. Una primera iteración arrojó un total de 20 no conformidades clasificadas como se muestra en el siguiente gráfico:

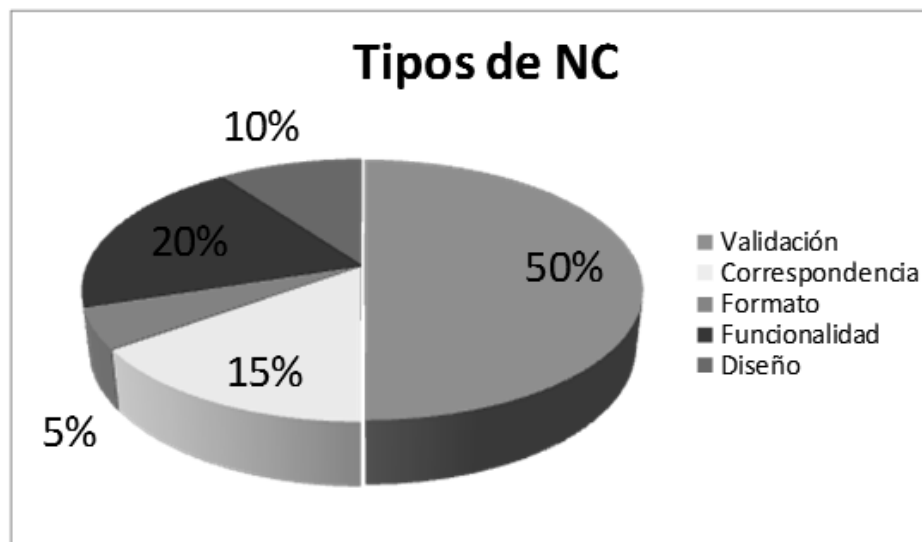


Figura 22: Clasificación de las NC identificadas. Primera iteración.

Las 20 NC detectadas fueron debidamente resueltas y se decidió hacer una segunda iteración de pruebas con el objetivo de verificar que no hayan quedado errores sin ser detectados y revisar que los cambios introducidos al resolver las NC de la primera iteración no habían introdujo nuevos errores.

En la segunda iteración se encontraron 8 NC clasificadas de la siguiente manera:

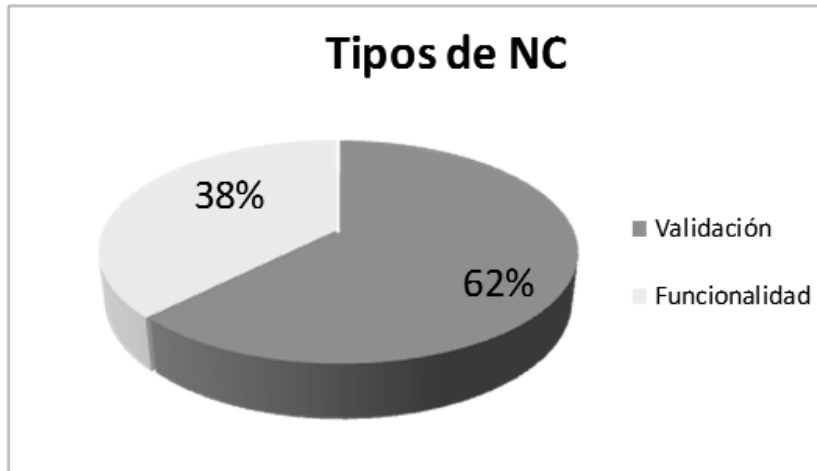


Figura 23: Clasificación de las NC identificadas. Segunda iteración.

La tercera iteración de pruebas se realizó exitosamente y no se encontraron ninguna NC, garantizando de esta manera que fueran satisfechos el total de requisitos especificados por el cliente. Los resultados de las 3 iteraciones se muestran en el siguiente gráfico:

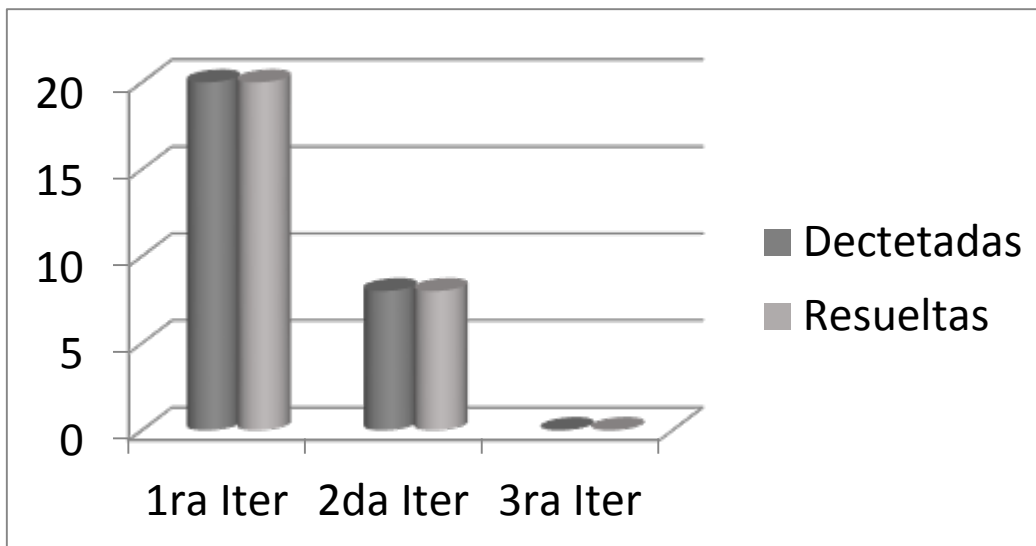


Figura 24: Iteraciones de las pruebas funcionales.

4.6. Resultados alcanzados

Con la implementación del sistema para la gestión de los recursos humanos se obtuvo una herramienta que:

1. **Categoría docente e investigativa:** Permite llevar el control de la categorización docente e investigativa de los profesores de cada una de las áreas de la facultad, implementando mecanismos de alerta oportuna ante la aproximación de vencimiento de una categoría.
2. **Generación de reportes:** Genera reportes estadísticos que apoyan a la toma de decisiones en tiempo real. Estos reportes muestran información actualizada del estado de las Prenóminas, las incidencias individuales y colectivas de los trabajadores, el estado de la categorización de cada profesor.
3. **Hojas de firma:** Genera la hoja de firma de cada trabajador teniendo en cuenta el horario de trabajo, la rotación de horas y los días feriados.
4. **Usuarios y roles:** Permite introducir usuarios al sistema con distintos roles y permisos de los cuales se almacena toda la información necesaria para la gestión de los recursos humanos.
5. **Alertas y notificaciones:** Genera alertas oportunas cuando ocurre un evento importante o está próximo a ocurrir. Estas alertas se muestran en ventanas flotantes con una paleta de colores opcional.

Conclusiones del capítulo

Tras el flujo de implementación y prueba, el sistema quedó desarrollado. Se cumplió con el objetivo planteado en el diseño teórico y se obtuvo como resultado un producto que cumple con el ciento por ciento de las especificaciones emitidas en la Facultad CITEC. Las pruebas realizadas permitieron elevar la calidad de la solución obtenida y corroboraron el correcto funcionamiento de la aplicación.

Conclusiones generales

Una vez concluida la investigación referida al proceso de gestión de los recursos humanos en la facultad CITEC, se obtuvo que:

- El estudio de diferentes fuentes bibliográficas, permitió obtener una visión de las principales ventajas y carencias que poseen los sistemas de gestión de los recursos humanos y la necesidad de un sistema informático que garantice su correcto funcionamiento dando cumplimiento al objetivo de la investigación.
- Con la selección de la metodología desarrollo de software Open Up se generaron varios artefactos, correspondientes a las fases: Modelado de Negocio, extracción de Requisitos, Análisis, Diseño e Implementación; permitió un mejor entendimiento de las funcionalidades a desarrollar, gracias a la gran documentación que presenta, sirviendo de base para la implementación del sistema informático.
- La realización de las pruebas de caja negra contribuyeron a garantizar que el sistema cumple con la calidad requerida para ser desplegado en la facultad CITEC. El sistema cumple adecuadamente con todos los requerimientos planteados por el cliente.

Recomendaciones

El sistema para gestión de los recursos humanos requiere de seguimiento e incremento en las funcionalidades que puede brindar, se recomienda agregar nuevas funcionalidades al sistema como:

- ✓ Integrar las alertas y notificaciones con un servidor de correo para que estas lleguen al correo personal de los roles interesados.

- ✓ Desplegar en otras facultades de la Universidad de la Ciencias Informáticas.

Referencias bibliográficas

BUSCHMANN F, MEUNIER, R., ROHNERT, H., SOMMERLAND, P. y STAL M 1996. Pattern – Oriented Software Architecture. A System of Patterns. Londres.

CARDOSO E, CAMACHO F y NUÑEZ G 2004. Arquitecturas de software. Guia de estudio. . S.I.

CONTRALORÍA GENERAL DE LA REPÚBLICA 2009c. Folleto SistemaCI, Resolución No. 60/11. La Habana.

DIAS ROBERTO 2010. Propuesta de Sistema Automatizado de Informacion Docente para Tecnologia en Villa Clara. [en línea]. [Consulta: 10 junio 2015]. Disponible en: <http://revedumecentro.sld.cu/index.php/edumc/article/view/70/143>.

ECLIPSE FOUNDATION. 2011. OpenUp Basic. S.I.

INFANTE A. 2013. Desarrollo de un prototipo de software para el requerimiento de solicitudes de creditos. Universidad EAN facultad de ingenieria. *Bogota.com*.

LASSO IVAN 2008. Que es el codigo fuente. [en línea]. [Consulta: 17 abril 2015]. Disponible en: <http://www.proyectoautodidacta.com/comics/que es el codigo fuente>.

MANAS, J.A. 1994. Pruebas de programacion. [en línea]. [Consulta: 17 abril 2015]. Disponible en: <http://www.it.uc3m.es/tsps/testinghtml>.

PRESSMAN, R.S. 2010. *Ingeniería del software, un enfoque práctico*. 5. S.I.: Mc Graw Hill.

DÍAZ, J. A. 2009, 'Migración y Perfeccionamiento del Sistema de Gestión de Recursos Humanos GREHU Subsistema Prenómina: Condicionamiento del proceso de pago', CUJAE.

CONCEPCIÓN, Y. B. 2011, *MÓDULO DE PRENÓMINA DE PERSONAL DEL SISTEMA DE GESTIÓN DE RECURSOS HUMANOS DE LA EMPRESA DE LA SAL*, Instituto Superior Politécnico “José Antonio Echeverría”.

HERNÁNDEZ, C. M. 2012, *MIGRACIÓN Y REDISEÑO DEL SISTEMA DE GESTIÓN DE RECURSOS HUMANOS GREHU, MÓDULO CONTROL DE ASISTENCIA*, Instituto Superior Politécnico “José Antonio Echeverría”.

ING. VIRTUDES MILAGRO FIGUEREDO LARA, I. L. C. L., ING. JOHANNY RIVERA LÓPEZ. 2010, 'Modelo para Gestionar el Capital Humano en la Implantación del Sistema Integral de Gestión CedruX.'

TETILLA, C. A. 2009, 'SIRHU. Sistema Integrado de Recursos Humanos'.

SAMPIERI, R. 2006, 'Metodología de la Investigación segunda edición', *Mexico*, p. 136 p.

SAMPIERI, R., COLLADO., C. F. & LUCIO, P. B. 2006, 'Metodología de la investigación', *México* vol. ISBN: 970-10-5753-8.

Bibliografía

BUSCHMANN F, MEUNIER, R., ROHNERT, H., SOMMERLAND, P. y STAL M 1996. Pattern – Oriented Software Architecture. A System of Patterns. Londres.

CARDOSO E, CAMACHO F y NUÑEZ G 2004. Arquitecturas de software. Guía de estudio.S.I.

CASTRO RAÚL 2009. www.cubadebate.cu. www.cubadebate.cu [en línea]. [Consulta: 10 noviembre 2014]. Disponible en: <https://cubadebate.cu/>.

ECLIPSE FOUNDATION. 2011. OpenUp Basic. S.I.

FLORES C y ALFERÉZ, G. 2011. Establecimiento de una Metodología de Desarrollo de Software para la Universidad de Navojoa Usando Open UP. México.

Framework-Ecured. [en línea] 2014. [Consulta: 24 febrero 2015]. Disponible en: <http://www.ecured.cu/index.php/Framework>.

GOLDSTEIN A, WEYL E y LAZIRIS L 2011. HTML5 & CSS3 for the Real World. . S.I.: SitePoint.

GONZALEZ ENRIQUE 2010. Aprender a Programar.

HERNÁNDEZ R A y COELLO S 2002. *El paradigma cuantitativo de la investigación científica*. La Habana: Editorial universitaria.

JACOBSON 2000. *El Proceso Unificado de Desarrollo de Software*. Madrid: Pearson Educación S.A. ISBN 8478290362.

LARMAN, C. 2009. *UML y patrones. Una introducción al análisis y diseño orientado a objetos. 2*. S.I.: s.n.

LOPERA C, GUTIERREZ E y MARTÍN JC 2003. *Indicadores: ciencia y tecnología en países de América Latina*. Lecturas de Economía. S.I.: s.n.

MC GRAW, H. 2001. *Ingeniería de software. Un enfoque práctico. 5*. España: s.n. ISBN 8448132149.

NEGRÍN SIMEÓN 1997. La ciencia y la tecnología en Cuba, Revista Cubana de Medicina Tropical. , vol.

49, no. 3, pp. 153-160.

DÍAZ, J. A. 2009, 'Migración y Perfeccionamiento del Sistema de Gestión de Recursos Humanos GREHU Subsistema Prenómina: Condicionamiento del proceso de pago', CUJAE.

CONCEPCIÓN, Y. B. 2011, *MÓDULO DE PRENÓMINA DE PERSONAL DEL SISTEMA DE GESTIÓN DE RECURSOS HUMANOS DE LA EMPRESA DE LA SAL*, Instituto Superior Politécnico “José Antonio Echeverría”.

HERNÁNDEZ, C. M. 2012, *MIGRACIÓN Y REDISEÑO DEL SISTEMA DE GESTIÓN DE RECURSOS HUMANOS GREHU, MÓDULO CONTROL DE ASISTENCIA*, Instituto Superior Politécnico “José Antonio Echeverría”.

ING. VIRTUDES MILAGRO FIGUEREDO LARA, I. L. C. L., ING. JOHANNY RIVERA LÓPEZ. 2010, 'Modelo para Gestionar el Capital Humano en la Implantación del Sistema Integral de Gestión CedruX.'

BEER, M. Y. C. 1990, 'Dirección de Recursos Humanos. Técnicas de Harvard Business School', *CECSA, México*.

FERNÁNDEZ, L. J. 2006, 'Gestión por competencias. Un modelo estratégico para la dirección de Recursos Humanos. ', *Madrid: , vol. Prentice Hall, Pearson*.

MONDY, R. 2005, 'Administración de los recursos humanos', *México: Editorial Pearson Prentice-Hi*.

Anexos

Anexo 1: Entrevista realizada para identificar errores en el proceso de Gestión de los recursos humanos en la Facultad de ciencias y tecnologías computacionales de la Universidad de las Ciencias Informáticas.

Descripción: El autor realiza la entrevista al departamento del vicedecanato administrativo de la Facultad CITEC, específicamente a la técnico medio en economía Liudmila de la Osa, para detallar el proceso de gestión de los recursos humanos en la Facultad CITEC de la Universidad de las Ciencias Informáticas (UCI), de esta manera se extrae la información que se necesita a través del uso de las técnicas ofimáticas y la propia experimentación de los implicados (entrevistador y entrevistados), la información de la entrevista está presente en el siguiente trabajo respetando las respuesta de los entrevistados.

Preguntas	Respuestas
¿Cómo se gestiona la prenomina de los trabajadores?	Se hace de forma manual, mediante las hojas de firma y reportes que hace los jefes de áreas.
¿Quién es el encargado de hacer las hojas de firmas?	El técnico de recursos humanos, lo hace manual el calcula las rotaciones.
¿Quiénes intervienen en la elaboración de la prenomina?	Los jefes de departamentos, el jefe del vicedecanato administrativo, el técnico de recursos humanos y el decano de la Facultad.
¿Quién verifica los resultados plasmados en la prenomina final?	El técnico de recursos humanos es quien la lleva para que todos los jefes de área la firmen y luego el decano quien es el que

	la revisa al final.
¿Cómo se realizan los reportes de las evaluaciones? ¿Qué tiempo se demora en dar el resultado?	Se hace manual si es individual el reporte se ha mes por mes de la persona implicada, el colectivo es más engorroso. Se demora alrededor de dos días o más dependiendo de la cantidad de información solicitada.
¿Cómo se identifican el vencimiento de las categorías docentes de los trabajadores docentes?	Muy pocas veces se sabe por el profesor la mayoría de las veces nos enteramos porque el especialista de recursos humanos avisa a la facultad, aproximadamente un mes antes de su vencimiento.
¿Qué deficiencias a su entender afectan el desarrollo satisfactorio del proceso?	Bueno errores que se comenten que atrasan la entrega de la pre Nómina, y la demora en la entrega de la información.
¿Existen otros procesos que se han informatizados?	Si existen otros procesos en la facultad que ya se han informatizados.
¿Cuáles son esos procesos ?	La guardia obrera, el pago por concepto de viático y los medios básicos.