



Ministerio de Educación Superior

Facultad de Matemática, Física y Computación

Trabajo de Diploma

Título: “Diseño e implementación de un sistema para el claustro en Universidad Central “Marta Abreu” de Las Villas”

Autor: Leandro Miguel Pentón Vasconcelos.

Tutores: Ing. Yasniel Pérez Acosta

MSc. Isel Moreno Montes de Oca

Consultante: MSc. Mailin Gonzales Arce

Santa Clara, 2013

Año 55 de la Revolución



Dictamen

El que suscribe, Leandro Miguel Pentón Vasconcelos, hago constar que el trabajo titulado “Diseño e implementación de un Sistema para profesores a tiempo parcial” fue realizado en la Universidad Central “Marta Abreu” de Las Villas como parte de la culminación de los estudios de la especialidad de Ciencias de la Computación, autorizando a que el mismo sea utilizado por la institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos ni publicado sin la autorización de la Universidad.

Firma del autor

Los abajo firmantes, certificamos que el presente trabajo ha sido realizado según acuerdos de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma del tutor

Firma del jefe del Laboratorio

Fecha

"Debemos trabajar por nuestro perfeccionamiento interno (...) como una impulsión constante; cada día analizar, analizar honestamente lo que hemos hecho, corregir nuestro errores y volver a empezar al día siguiente".

Ernesto Che Guevara.

Dedicatoria

A mi madre, a mi hermana y a mi padre que son lo más importante en mi
vida

Agradecimientos

A mis padres y a mi hermana que siempre los tengo en mi corazón.

A mis amigos que siempre estuvieron en las buenas y en las malas a mi
lado.

A todos aquellos que pasaron malas noches junto a mí.

En fin a todos lo que de una forma u otra hicieron este sueño posible.

Resumen

En el presente trabajo se hace uso de las Nuevas Tecnologías de la Información y las Comunicaciones para construir un Sistema de Información para el control de profesores en la Universidad Central “Marta Abreu” de Las Villas. Este sistema abarca los contenidos esenciales del tratamiento de la información, usando **UML** como un lenguaje de modelación; **PHP** y **PostgreSQL** como lenguajes de programación. Posee un ambiente web basado en la tecnología de software libre, y proporciona información a la dirección de recursos humanos de la propia universidad sobre cada uno de los profesores. Facilita además las inscripciones de acuerdo a la categoría para a partir de ello efectuar el pago salarial. Como parte de este trabajo se diseña una base de datos para el manejo de la información sobre profesores a través de un sistema de gestión de bases de datos.

Abstract

In the present work makes use of New Technologies of Information and Communications to build an information system for the control of teachers at the Central University “Marta Abreu” of Las Villas. This system covers the essential content of information processing, using UML as a modeling language, PHP and PostgreSQL as programming languages. It has a web environment based on open source technology, and provides information to the human resources departments own uuniversidad on each of the teachers. It also facilitates the registration according to the category for it from making wage payment. As part of this work was designed a database for managing information through a website management system databases.

Tabla de Contenido

Introducción.....	1
Capítulo 1: Vinculación de las nuevas tecnologías al área de Recursos Humanos en la UCLV.	4
1.1 ¿En qué consiste Recursos Humanos?.....	4
1.1.2 Breve caracterización de la Dirección de Recursos Humanos en la UCLV. ...	5
1.1.3 Objetivos principales de la Dirección de Recursos Humanos en la UCLV.	5
1.2 Impacto de las Nuevas Tecnologías de la Información y las Comunicaciones	6
1.3 Selección de la tecnología adecuada.....	7
1.3.1 Uso de la tecnología en nuestro país.	8
1.4 Software libre de código abierto	8
1.4.1 Razones para usar software libre de código abierto	9
1.4.2 Las tecnologías web	10
1.4.3 La web como divulgación de información	11
1.4.4 Sistema de navegación eficiente	11
1.4.5 Ventajas de la web.....	12
1.4.6 PHP como lenguaje de programación	13
1.4.7 Ventajas de usar PHP	14
1.4.8 Ventajas de usar PHP	15
1.5 Sistema Gestor de base de datos (SGBD).....	16
1.5.1 PostgreSQL como gestor de base de datos.....	16
1.5.2 Ventajas de PostgreSQL.....	17
1.5.3 Desventajas de PostgreSQL.	19
1.5.4 PostgreSQL vs. Access	20
1.6 Herramientas de migración de datos.....	20
1.6.1 Pentaho Data Integration (Kettle)	21
1.6.2 Consideraciones importantes del Kettle.....	21
1.7 Lenguajes de modelado de objetos	23
1.7.1 UML como lenguaje de modelación	24
1.7.2 Conceptos básicos de UML	25
1.7.3 Inconvenientes de UML	26
1.7.4 Perspectivas de UML	26
1.7.5 Utilidad de UML	27

Tabla de Contenidos

1.7.6 UML vs otros lenguajes de modelación	28
Consideraciones finales del capítulo	29
Capítulo 2: Diseño e implementación del Sistema.....	30
2.1 Requisitos del Sistema	30
2.1.1 Requisitos funcionales.....	31
2.1.2 Requisitos no funcionales.....	31
2.2 Modelación conceptual de datos	33
2.2.1 Esquema conceptual de la BD visualizar información del claustro universitario.....	35
2.2.2 Modelo para la visualización de la información del claustro universitario en la UCLV.....	36
2.3 Actor del sistema	37
2.4 Modelo de casos de uso del sistema	38
2.4.1 Formatos de los casos de uso	38
2.4.2 Diagrama de actor y casos de uso del sistema	39
2.5 Diagrama de navegación.....	41
2.6 Arquitectura del sistema Cliente-Servidor.....	42
2.6.1 Diagrama de despliegue	42
2.7 Diagrama de transición de estados.....	43
2.8 SQL como lenguaje de consultas	44
2.8.1 Reportes de empleados implementados mediante consultas SQL	45
2.9 Sistemas basados en Microsoft Access.....	48
2.9.1 Limitaciones del sistema ASSETS de la UCLV	48
Consideraciones finales del capítulo	49
Capítulo 3: Descripción del ambiente del sistema.....	50
3.1 Actualización del sistema	50
3.1.1 Configuración de variables de entorno.....	50
3.1.2 Carga y ejecución de las trasformaciones mediante el Spoon	52
3.2 Ambiente de usuario del Sistema.....	55
Consideraciones finales del capítulo	66
Conclusiones.....	68
Recomendaciones	69
Referencias Bibliográficas	70

Tabla de Contenidos

Introducción

La Gestión de los Recursos Humanos se ha convertido en la actualidad en uno de los eslabones esenciales para el buen funcionamiento de cualquier entidad, dado el incremento de las exigencias cualitativas del personal y la necesidad de obtener calidad y productividad para lograr y mantener los parámetros que se exigen en las organizaciones actuales. Esta calidad depende en gran medida de los clientes internos que son los que la “hacen”, quienes a la vez de formar parte de los procesos sustantivos de la organización deben estar motivados para ofrecer la misma y poseer una formación profesional que les permita ofrecer la calidad que responda a los intereses de la organización. La Gestión de Recursos Humanos constituye, por tanto, un factor básico para que la empresa pueda obtener altos niveles de productividad, calidad y competitividad.

Los impactos de los cambios sustanciales que se han producido en el plano económico, político y tecnológico en el ámbito internacional, han transformado el entorno en el cual operan las organizaciones cubanas.

En este contexto, las Instituciones de Educación Superior se han visto precisadas a elevar su competitividad, específicamente a través de sus Recursos Humanos, por constituir éstos el capital fundamental en el desarrollo de los procesos que en estas entidades se desarrollan.

Las Instituciones de Educación Superior se desarrollan a través de un conjunto de procesos que como un todo deben dar una respuesta pertinente a la sociedad la cual dependerá, no sólo de la calidad y concepción de los procesos identificados como de pertinencia e impacto social, sino también de los procesos denominados de existencia, entre los que se considera la gestión de los recursos.

Los Recursos Humanos constituyen el elemento más importante y dinámico en la gestión económico-financiera, en especial el personal cuya labor tiene que ver directamente con la esfera del registro y control, del análisis, la gestión de los recursos humanos, las finanzas, las estadísticas, la planificación, o que sus funciones tengan una incidencia decisiva en el desenvolvimiento y resultado de dichas actividades.

A pesar de los innegables avances como consecuencia del perfeccionamiento continuo de la gestión de los Recursos Humanos en la Educación Superior, aún persisten

deficiencias e insuficiencias en general y en particular con los sistemas de información de los dispositivos de recursos humanos, donde todavía no se alcanza la eficiencia y efectividad que se requiere.

Uno de los centros pertenecientes a éste organismo es la Universidad Central” Marta Abreu” de Las Villas (UCLV) que es por demás de los mayores centros universitarios del país. Su historia está estrechamente ligada a la vida y obra de Ernesto Che Guevara. Es un baluarte de la Revolución Cubana.

La Dirección de los Recursos Humanos de la UCLV ha dado muestras en los diferentes servicios que brinda que tiene capacidad de dar un paso más en la mejora de la obtención de la información que satisfaga los requerimientos de las facultades y la comunidad universitaria.

Esta dirección consciente de sus potencialidades, está enfrascada en el diseño de un sistema de informático con énfasis en los profesores e investigadores que conforman el claustro universitario, teniendo como antecedente el Sistema de Datos de Recursos Humanos ASSETSS, el cual ofrece múltiples informaciones que abarcan a todos los trabajadores pero carece de informes estadísticos dirigidos a los profesores e investigadores por unidades organizativas que sirva de base para la toma de decisiones y contribuya al cumplimiento de los objetivos y criterios de medida en la universidad. La obtención de las informaciones no son flexibles ni dinámicas. El sistema actual tiene como restricción el permiso de acceso donde solo pueden acceder de forma simultánea cuatro usuarios lo que trae consigo demoras en la entrada y actualizaciones de datos, así como la obtención de informes actualizados.

Por todo lo anteriormente planteado, se tiene el siguiente **problema de investigación**.

Obtener un sistema que proporcione los informes estadísticos actualizados de claustro universitario por unidades organizativas que contribuya al cumplimiento de los objetivos y criterios de medida en la UCLV de manera flexible y dinámica y al que puedan acceder de forma simultánea un número ilimitado de usuarios.

Para dar solución al problema de investigación planteado se propone el siguiente **objetivo general**.

Implementar un sistema para la visualización de la información del claustro universitario comprendido para docentes, investigadores y profesores a tiempo parcial

en la Universidad Central “Marta Abreu” de Las Villas ello usando el enfoque tradicional de bases de datos.

Objetivos Específicos

- ✓ Desarrollar la modelación del sistema a implementar.
- ✓ Diseñar la BD que contendrá la información del claustro de la UCLV.
- ✓ Implementar un sistema que permita la visualización de la información del claustro universitario en la UCLV.

Preguntas de Investigación

- ✓ ¿Qué datos se deben registrar de los profesores que permitan la efectividad del sistema?
- ✓ ¿De qué forma se aprovecharán las aplicaciones anteriores en la representación de los datos?

Valor Práctico

La obtención de un sistema para la visualización de información referente al claustro universitario (esto es, docentes, investigadores y profesores a tiempo parcial) de acuerdo a su categoría. Este sistema posibilita además la preservación de los recursos del estado en cuestiones monetarias

Hipótesis de Investigación

Es posible obtener un sistema para la extracción y muestra de información actualizada del claustro universitario por parte del Departamento de Recursos Humanos en la UCLV que sirva de base para la toma de decisiones y contribuya al cumplimiento de los objetivos y criterios de medida en la UCLV de manera flexible y dinámica y al que puedan acceder de forma simultánea un número ilimitado de usuarios.

Tipo de Investigación

Aplicada.

Capítulo 1: Vinculación de las nuevas tecnologías al área de Recursos Humanos en la UCLV.

La revolución digital ha traído importantes cambios para todos nosotros y también ha provocado una evolución en el entorno empresarial que podemos resumir en la importancia en que una economía digital tiene las nuevas tecnologías, ya que, no solo afectan las relaciones de la organización con los clientes, proveedores e, incluso empleados, sino que son el elemento fundamental para los procesos vitales como pueden ser la producción, las ventas, la contabilidad, etc., y hasta los recursos humanos.

Los departamentos de Recursos Humanos están descubriendo que la tecnología es el aliado perfecto para la optimización de su gestión y de los costos de servicio por empleado. Entre otras cosas les permite concentrarse más en otros roles de valor añadido.

En América Latina, las empresas han ido optando por implementar sistemas de información para la administración de personal donde los empleados, además de recibir información de manera rápida y homogénea, pueden actualizar su información personal, solicitar vacaciones, imprimir el recibo de sueldo, acceder a beneficios e inscribirse en cursos, entre otras ventajas (González Sabín 2005).

1.1 ¿En qué consiste Recursos Humanos?

En la administración de empresas, se denomina Recursos Humanos al trabajo que aporta el conjunto de los empleados o colaboradores de una organización. Lo más frecuente es llamar así a la función o gestión que se ocupa de seleccionar, contratar, formar, emplear y retener a los colaboradores de la organización. Estas tareas las puede desempeñar una persona o departamento en concreto junto a los directivos de la organización.

El objetivo básico que persigue la función de Recursos Humanos es alinear el área de trabajo y profesionales con la estrategia de la organización, lo que permite implantar la estrategia organizacional a través de las personas, quienes son consideradas como los únicos recursos inteligentes capaces de llevar al éxito organizacional y enfrentar los desafíos que hoy día se percibe en la fuerte competencia mundial. Es

imprescindible resaltar que no se administran personas ni Recursos Humanos, sino que se administra con las personas o sea, agentes activos y proactivos dotados de inteligencia, creatividad y habilidades intelectuales (Apaza, Méndez et al. 2010).

Generalmente el Departamento de recursos Humanos se compone por las áreas de reclutamiento y selección, contratación, capacitación, administración y gestión del personal durante la permanencia en la empresa. Dependiendo de la empresa o institución donde la función de Recursos Humanos opere, pueden existir otros grupos que desempeñen distintas responsabilidades vinculados con aspectos como la administración de la nómina de los empleados y el manejo de las relaciones con sindicatos, entre otros.

Para poder ejecutar la estrategia de la organización resulta fundamental la administración de los Recursos Humanos. Para ello se deben considerar los conceptos de comunicación organizacional, el liderazgo, el trabajo en equipo, la negociación y la cultura organizacional.

1.1.2 Breve caracterización de la Dirección de Recursos Humanos en la UCLV.

El sistema empleado por la UCLV para llevar adelante sus tareas se basa en una dirección participativa y con enfoque estratégico, que realiza un proceso de planeación estratégica donde se identifican y señalan las áreas de resultados claves universitarios, con su sistema de objetivos y criterios de medidas. En tanto cada área universitaria ejecuta su propia planeación en función de dar respuesta a la planeación estratégica universitaria. Con frecuencia semestral y anual se establece un chequeo del cumplimiento de los objetivos.

1.1.3 Objetivos principales de la Dirección de Recursos Humanos en la UCLV.

La Dirección de Recursos Humanos de la UCLV tiene como misión proporcionar el capital humano competente para contribuir al logro de metas trazadas, a la toma de decisiones y a la profesionalidad del personal según las competencias requeridas para el funcionamiento de la universidad; centro dirigido a la creación de una cultura basada en la calidad y la excelencia; identificado con una gestión efectiva y un clima

laboral satisfactorio. Por otra parte la visión como organización se trabaja en la implantación del Sistema de Gestión Integrada de Capital Humano.

La Dirección de Recursos Humanos cuenta con un colectivo altamente comprometido con los programas de la Revolución Cubana y los cambios sociales que se derivan de las necesidades del desarrollo sostenible del país, con elevado comprometimiento y protagonismo en los programas de la Revolución que cumpla con los principios de idoneidad demostrada y de las competencias laborales.

Además, se pretende la participación de los trabajadores en el análisis, la toma de decisiones y la solución de los problemas. La capacitación continua y permanente de los recursos humanos afín de tomar profesionales integrales, la inversión para el desarrollo de las competencias laborales, el mejoramiento de las condiciones de trabajo; la prevención de los riesgos y la reducción de la accidentalidad laboral, así como el fortalecimiento de la disciplina y la organización del trabajo; contribuyen al logro de un desempeño laboral superior al incremento de la productividad.

1.2 Impacto de las Nuevas Tecnologías de la Información y las Comunicaciones

El impacto que ha tenido en el mundo el uso de las Nuevas Tecnologías de la Información y las Comunicaciones (NTIC) desde su surgimiento en el pasado siglo, ha marcado un trascendental cambio en la sociedad, jugando un papel fundamental para la divulgación de información. Esto ha hecho posible que esté al alcance de casi todas las personas en el mundo en cuestión de poco tiempo. Con el surgimiento de Internet se ha logrado que las personas puedan acceder a la información desde sus hogares, centros de trabajo o cualquier parte del mundo, todo esto unido a la digitalización de la información, que facilita el almacenamiento de la misma de una forma rápida, segura y compacta.

El recurso que hoy se considera más valioso es la información. Si se quiere alcanzar un objetivo, es preciso acceder a la información pertinente para llegar a tomar las decisiones adecuadas. La gestión de la misma como nuevo concepto de la ciencia de la información permite un control más eficiente sobre una serie de datos, además posibilita una adaptación eficaz a los posibles cambios del entorno.

La universidad ha de contribuir no sólo a preservar toda la cultura alcanzada por la humanidad hasta nuestra época, sino que en nuestro contexto, tiene la importante misión de promover y desarrollar también la cultura científica de la sociedad; esta misión sería incompleta si no tiene como una de sus metas más importantes el fomento y desarrollo de esa cultura científica de la comunidad con que interactúa; por esta razón, potenciar el uso de las TIC como vía más amplia y racional para procesar y difundir la información trae consigo que la promoción de la cultura por la vía extracurricular complemente el desarrollo cultural integral de los estudiantes y se distinga de lo curricular por sus métodos y estilos propios. Las actividades extracurriculares promueven la afición y el interés por la cultura, el arte y la literatura; la cultura física, el deporte y el empleo culto del tiempo libre (González Sabín 2005).

1.3 Selección de la tecnología adecuada

Como parte de la globalización que afecta irremediablemente la economía mundial y otros sectores, la alternativa de software libre se presenta como una opción viable. La monopolización de la tecnología por parte de los países desarrollados no permite un desarrollo equitativo del resto de las naciones y provoca que tengan que pagar altos precios para acceder a ésta.

El software libre es una opción para el desarrollo igualitario de las necesidades de estas técnicas. La importancia del uso de las herramientas de software se ve reflejada en todos los aspectos de la actividad humana. El software libre elimina todo tipo de diferencias y pone a todos en condiciones de igualdad. Es por esto que constituye un nuevo modelo de economía que intenta enfrentarse a la globalización mundial.

Los usuarios demandan interfaces cada vez más completas y funcionalidades más elaboradas, influyendo en el tamaño y la complejidad del producto final. Por ello, los programas deben ser estructurados de manera que puedan ser revisados, corregidos y mantenidos rápida y eficazmente por personas que no necesariamente han colaborado en su diseño y construcción. Esto permite una nueva funcionalidad, mayor seguridad y robustez, funcionando en todas las situaciones que puedan surgir de manera previsible y reproducible (Cernuda 2001).

1.3.1 Uso de la tecnología en nuestro país.

En nuestro país el desarrollo de software constituye una actividad de alta capacidad técnica que depende en lo esencial de la adecuada preparación de los factores humanos involucrados y de las capacidades de comunicación entre los mismos. Es por ello que se ha evidenciado la necesidad de establecer un conjunto de normas y políticas que cohesionen las tendencias del desarrollo de software en la UCLV.

De acuerdo a la política de desarrollo de Software y Sistemas Informáticos de la UCLV, para hacer uso de sistemas de gestión de bases de datos, los requerimientos de persistencia de las aplicaciones son modelados utilizando el tradicional enfoque de bases de datos relacionales lo que implica el uso de MySQL y PostgreSQL.

El desarrollo de Cuba y en particular de la UCLV ha seguido la misma tendencia de desarrollo apreciable en otras comunidades de Internet. Este aspecto respalda la decisión de basar el desarrollo en dos lenguajes principales:

- ✓ PHP para aplicaciones de menor o mediana complejidad y Java en caso de aplicaciones más complejas.

Los sistemas deben seguir una arquitectura basada en WEB para facilitar la gestión centralizada y evitar los altos requerimientos de mantenimiento de las aplicaciones de escritorio (Morales Guedes 2012).

1.4 Software libre de código abierto

El término código abierto fue utilizado por primera vez en 1998 por algunos usuarios de la comunidad del software libre, tratando de usarlo como reemplazo al ambiguo nombre original en inglés del software libre (free software). La filosofía del código abierto orienta su atención en la premisa de que al compartir el código, el programa resultante tiende a ser de calidad superior al software propietario, es una visión técnica.

El movimiento código abierto tiene un decálogo que debe cumplir un código para poder llamarse código abierto, estas son:

- Código fuente: el código fuente debe estar incluido u obtenerse libremente.
- Trabajos derivados: la redistribución de modificaciones debe estar permitida.

- Integridad del código fuente del autor: las licencias pueden requerir que las modificaciones sean redistribuidas sólo como parches.
- Sin discriminación de personas o grupos: nadie puede dejarse fuera.
- Sin discriminación de áreas de iniciativa: los usuarios comerciales no pueden ser excluidos.
- Distribución de la licencia: deben aplicarse los mismos derechos a todo el que reciba el programa
- La licencia no debe ser específica de un producto: el programa no puede licenciarse solo como parte de una distribución mayor.
- La licencia no debe restringir otro software: la licencia no puede obligar a que algún otro software que sea distribuido con el software abierto deba también ser de código abierto.
- La licencia debe ser tecnológicamente neutral: no debe requerirse la aceptación de la licencia por medio de un acceso por clic de ratón o de otra forma específica del medio de soporte del software.

1.4.1 Razones para usar software libre de código abierto

Por la actual situación económica de nuestro país, se ha dificultado la adquisición del software propietario. Por esta razón nuestra sociedad ha tenido que emigrar irremediabilmente para el software libre. La utilización del software libre de código abierto brinda una serie de ventajas que lo catalogan como una buena elección:

Libertad de uso y redistribución: Las licencias de software de fuentes abiertas existentes permiten la instalación del software tantas veces y en tantas máquinas como el usuario desee.

Independencia tecnológica: El acceso al código fuente permite el desarrollo de nuevos productos sin la necesidad de desarrollar todo el proceso partiendo de cero.

Fomento de la libre competencia al basarse en servicios y no licencias: Uno de los modelos de negocio que genera el software de fuentes abiertas es la contratación

de servicios de atención al cliente. Este sistema permite que las compañías que den el servicio compitan en igualdad de condiciones al no poseer la exclusividad del producto del cual dan el servicio. Esto, además, produce un cambio que redundará en una mayor atención al cliente y contratación de empleados, en contraposición a sistemas mayoritariamente sostenidos por la venta de licencias.

Estándares abiertos: Los estándares abiertos permiten una interoperabilidad más alta entre sistemas, evitando incompatibilidades. Los estándares de facto son válidos en ocasiones para lograr una alta interoperabilidad si se omite el hecho de que estos exigen el permiso del propietario y, en su caso, el pago de cuotas.

Sistemas sin puertas traseras y más seguros: El acceso al código fuente permite que tanto expertos como empresas de seguridad de todo el mundo puedan auditar los programas, por lo que la existencia de puertas traseras es ilógica, ya que se pondría en evidencia de manera casi inmediata.

Corrección más rápida y eficiente de fallos: La disponibilidad del código fuente ha demostrado solucionar más rápidamente los fallos de seguridad en el software de fuentes abiertas, posibilidad que no se da en el caso del software propietario.

1.4.2 Las tecnologías web

A partir de la masificación de internet y más aún en los tiempos modernos, la web se convirtió en una gran herramienta, la cual está disponible los 365 días del año. Además, este medio permite relacionarse estrechamente a clientes y proveedores alrededor del mundo, conocerlos mejor, saber cuáles son sus gustos y exigencias, mejorando así la calidad del servicio brindado, ofreciendo oportunamente productos y servicios relacionados a sus necesidades presentes y futuras.

Las tecnologías Web se pueden visualizar como un conjunto de principios y prácticas que marcan una tendencia en la tecnología y en el diseño web; representan un conjunto de tecnologías y paradigmas, como lo son los servicios web, páginas dinámicas, sistemas de información geográfica, tecnologías de interacción de diversas fuentes de información, que facilitan la colaboración y el intercambio de información entre los usuarios. En otras palabras, las tecnologías Web son un conjunto de técnicas y paradigmas resultado de la revolución comercial en la

industria computacional causada por el uso de Internet como plataforma de desarrollo y de consulta de información.

1.4.3 La web como divulgación de información

La Web ha permitido un flujo de comunicación global a una escala sin precedentes en la historia humana. Personas separadas en el tiempo y el espacio, pueden usar la Web para intercambiar, incluso desarrollar mutuamente, sus pensamientos, o alternativamente sus actitudes y deseos cotidianos. Experiencias emocionales, ideas políticas, cultura, idiomas, musicales, negocio, arte, fotografías, literatura... todo puede ser compartido y diseminado digitalmente con muy poco esfuerzo, haciéndolo llegar casi de forma inmediata a cualquier punto del planeta; y todo ello sin utilizar recursos físicos como las bibliotecas o la prensa escrita. Gracias a su carácter virtual, la información en la Web puede ser buscada más fácil y eficientemente que en cualquier medio físico, y mucho más rápido de lo que una persona podría recabar por sí misma a través de un viaje, correo, teléfono, telégrafo, o cualquier otro medio de comunicación.

1.4.4 Sistema de navegación eficiente

Para obtener un sistema de navegación eficiente en un entorno Web es importante seguir un estándar en el diseño del mismo, de lo contrario, puede provocar confusión, que no se sepa dónde buscar lo que se necesita y que lo que se trata de decir con el sistema de navegación no se dé a conocer de la manera adecuada. Es recomendable entonces que la barra de navegación esté situada en el mismo lugar en todas las páginas y con los mismos gráficos, lo que proporcionará seguridad y fluidez en los pasos a seguir.

Uno de los aspectos a tener en cuenta, en el proceso de desarrollo de un entorno Web, es que se debe priorizar la información sustancial. Es importante colocar en la página principal los productos o servicios más relevantes, sugiriendo los pasos a seguir mediante links resaltados en la barra de navegación. De esta forma se despierta el interés por lo que se está ofertando y crea motivación a continuar indagando en el sitio. Como en los entornos Web no todos los contenidos presentan la misma relevancia, la información que juegue un papel secundario o que no sea primordial, debe ser separada de la información más importante y desplazada a un

lugar más común como el fondo de la página. Esto se logra con la utilización de links en forma de texto, lo que contribuye a que no se distraiga la navegación.

La página principal o de bienvenida es la más importante de un entorno Web, entonces es fundamental que se pueda retornar a ella. En la creación de la barra de navegación debe existir un botón o vínculo que indique el regreso a la portada del entorno Web. Así, si se cae en alguna página alterna, siempre se podrá conocer la página inicial y volver a ella. De manera general los entornos Web son más satisfactorios si presentan un diseño atractivo y una navegación placentera y sistematizada. Para alcanzar esta eficacia la barra de navegación debe estar ordenada por categorías priorizando los servicios más importantes del sitio. Es recomendable además que los gráficos sean lo suficientemente ligeros y evitar que la programación y los componentes contengan errores, lo que contribuye a que el entorno Web sea caracterizado como un Web de calidad.

1.4.5 Ventajas de la web

El uso de la tecnología Web para el desarrollo de los sistemas internos de las organizaciones, permite presentar pantallas e interactuar con los usuarios a través de los mismos navegadores que utilizan para acceder a sitios de Internet. El acceso se puede realizar desde cualquier computadora y sistema operativo, solo es necesario una computadora conectada a la red donde se encuentra instalado el sistema. Además, en caso de contar con una conexión permanente a Internet, se puede configurar para el acceso desde cualquier lugar del mundo a través de Internet.

La incorporación fácil de nuevas funciones así como la mantención de las existentes sin distribuir ni instalar software es la clave de su popularidad. Además, estandariza formatos y formas de acceso a la información.

La web ofrece muchas facilidades para la implementación de sistemas entre las que se puede señalar:

- **Almacenamiento:** Un entorno Web brinda posibilidades ilimitadas de almacenamiento y no depende de un lugar específico para guardar los datos y la información pues esta se almacena de forma electrónica.

- **Accesibilidad:** Al tener la posibilidad de estar conectado a Internet, puede accederse desde cualquier parte del mundo.
- **Seguridad:** Las posibilidades de seguridad de un ambiente Web permiten establecer niveles de privilegio para acceder a la información.
- **Rapidez de búsqueda:** Los buscadores Web permiten buscar mediante palabras específicas dentro de todo un gran volumen de información en cortos períodos de tiempo.
- **Transmisión de la información:** La información codificada puede ser transmitida de manera rápida y segura por la red.
- **Análisis:** Un entorno Web puede trabajar conjuntamente con sistemas de cómputo y análisis de datos y de información para mostrar los resultados de manera que se facilite su visualización e interpretación.
- **Despliegue:** La información se muestra en páginas Web, sin tener que disponer de nuevos medios para mostrar la nueva.

La capacidad para actualizar y mantener aplicaciones Web sin distribuir ni instalar software en las computadoras de los clientes es la razón clave de su popularidad que provoca a su vez, que la mayoría de los clientes no necesiten entrenamiento adicional para manipular la interfaz de la aplicación. Además, alojando la aplicación en un servidor Web, se logra una disponibilidad total del producto.

1.4.6 PHP como lenguaje de programación

PHP es un lenguaje de secuencia de comandos de servidor diseñado específicamente para la Web. Dentro de una página Web puede incrustar código PHP, que se ejecutará cada vez que se visite una página. El código PHP es interpretado en el servidor Web y genera código HTML y otro contenido que el visitante verá.

Fue concebido en 1994 y es fruto del trabajo de Rasmus Lerdorf. Ha sido adoptado por otras personas y ha experimentado tres transformaciones importantes hasta convertirse en el producto actual. En octubre de 2002, era utilizado por más de nueve millones de dominios de todo el mundo y su número crece rápidamente.

Constituye un producto de código abierto, lo que quiere decir que puede accederse a su código. Puede utilizarse, modificarse y redistribuirse sin coste alguno. Las siglas

PHP equivalían inicialmente a Personal Home Page (Página de inicio personal) pero se modificaron de acuerdo con la convención de designación de GNU (del inglés, *Gnu's Not Unix*, *Gnu* no es Unix) y ahora equivale a PHP *Hipertext Preprocessor* (*Preprocesador de hipertexto PHP*)(Welling and Thomson 2004).

Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno(BEÁTI 2012).

1.4.7 Ventajas de usar PHP

El gran parecido que posee PHP con los lenguajes más comunes de programación estructurada, como “C” y “Perl”, permiten a la mayoría de los programadores crear aplicaciones complejas con una curva de aprendizaje muy corta. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones. Permite la conexión a diferentes tipos de servidores de bases de datos tales como: MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.

PHP tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, y puede interactuar con los servidores web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI(Lopez 2008).

El lenguaje PHP es multiplataforma, completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos. Además, su código fuente escrito es invisible al navegador y al cliente pues es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.

Presenta la capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad. Destaca su conectividad con MySQL y PostgreSQL. Es un lenguaje libre, por lo que se presenta como una alternativa de fácil acceso para todos. También permite aplicar técnicas de programación orientada a objetos. Sin embargo no requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.

PHP está disponible para una gran cantidad de sistemas operativos diferentes. Además puede escribir código PHP en todos los sistemas operativos gratuitos del tipo Unix, como Linux y *FreeBSD*, versiones comerciales de Unix como Solaris e IRIX, o en las diferentes versiones de Microsoft Windows. Su código funcionará sin necesidad de aplicar ninguna modificación a los diferentes sistemas que ejecute PHP.

PHP incorpora una gran cantidad de funciones integradas para realizar útiles tareas relacionadas con la Web. Puede generar imágenes GIF al instante, establecer conexiones a otros servicios de red, enviar correos electrónicos, trabajar con cookies y generar documentos PDF, todo con unas pocas líneas de código (Gallego Vásquez 2003).

En PHP no solo la velocidad de ejecución, la cual es importante, sino que posibilita no crear demoras en la máquina. Por esta razón no requiere demasiados recursos de sistema. PHP se integra muy bien junto a otro software, especialmente bajo ambientes Unix y cuando se configura como módulo de Apache, está listo para ser utilizado.

La velocidad no sirve de mucho si el sistema se cae cada cierta cantidad de ejecuciones. Ninguna aplicación es 100% libre de bugs, pero teniendo de respaldo una increíble comunidad de programadores y usuarios es mucho más difícil para lo bugs sobrevivir. PHP utiliza su propio sistema de administración de recursos y dispone de un sofisticado método de manejo de variables, conformando un sistema robusto y estable.

Además el sistema debe poseer protecciones contra ataques. PHP provee diferentes niveles de seguridad. Estos pueden ser configurados desde el archivo .ini

1.4.8 Ventajas de usar PHP

El lenguaje PHP promueve la creación de código desordenado con un mantenimiento complejo. Además no posee adecuado manejo de unicode. Es diseñado especialmente hacia un modo de realizar aplicaciones Web que es problemático y obsoleto, por lo tanto es muy difícil de optimizar. Resulta un lenguaje interpretado por lo que se necesita tener el intérprete de **PHP**, y esto trae un alto consumo de las aplicaciones, pues son imposibles de compilar.

1.5 Sistema Gestor de base de datos (SGBD)

Uno de los objetivos fundamentales de un sistema es contar no sólo con recursos de información, sino también con los mecanismos necesarios para poder encontrar y recuperar estos recursos. De esta forma, las bases de datos se han convertido en un elemento indispensable no sólo para el funcionamiento de los grandes motores de búsqueda y la recuperación de información a lo largo y ancho de la Web, sino también para la creación de sedes web, Intranets y otros sistemas de información en los que se precisa manejar grandes o pequeños volúmenes de información. La creación de una base de datos a la que puedan acudir los usuarios para hacer consultas y acceder a la información que les interese es, pues, una herramienta imprescindible de cualquier sistema informativo sea en red o fuera de ella.

Un Sistema Gestor de base de datos (SGBD) es un conjunto de programas que permiten crear y mantener una Base de datos, asegurando su integridad, confidencialidad y seguridad. Por tanto debe permitir: especificar tipos, estructuras y restricciones de datos, guardar los datos en algún medio controlado por el mismo SGBD, realizar consultas, actualizarla, generar informes. Algunas de las características deseables en un SGBD son: Control de la redundancia: La redundancia de datos tiene varios efectos negativos (duplicar el trabajo al actualizar, desperdicia espacio en disco, puede provocar inconsistencia de datos) aunque a veces es deseable por cuestiones de rendimiento y seguridad. Restricción de los accesos no autorizados: cada usuario ha de tener unos permisos de acceso y autorización. Cumplimiento de las restricciones de integridad: el SGBD ha de ofrecer recursos para definir y garantizar el cumplimiento de las restricciones de integridad.

1.5.1 PostgreSQL como gestor de base de datos

PostgreSQL es un servidor de base de datos objeto relacional libre. Incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional, liberado bajo la licencia BSD. Como muchos otros proyectos *open source*, el desarrollo de PostgreSQL no es manejado por una sola compañía sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su

desarrollo, dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).

PostgreSQL comenzó como el proyecto Ingres en la Universidad de Berkeley. Este proyecto, liderado por Michael Stonebraker, fue uno de los primeros intentos en implementar un motor de base de datos relacional. Después de haber trabajado un largo tiempo en Ingres y de haber tenido una experiencia comercial con el mismo, Michael decidió volver a la Universidad para trabajar en un nuevo proyecto sobre la experiencia de Ingres, dicho proyecto fue llamado post-ingres o simplemente POSTGRES.

En proyecto, post-ingres pretendía resolver los problemas con el modelo de base de datos relacional que habían sido aclarados a comienzos de los años 1980. El principal de estos problemas era la incapacidad del modelo relacional de comprender "tipos", es decir, combinaciones de datos simples que conforman una única unidad. Actualmente estos son llamados objetos.

Se esforzaron en introducir la menor cantidad posible de funcionalidades para completar el soporte de tipos. Estas funcionalidades incluían la habilidad de definir tipos, pero también la habilidad de describir relaciones - las cuales hasta ese momento eran ampliamente utilizadas pero mantenidas completamente por el usuario. En POSTGRES la base de datos "comprendía" las relaciones y podía obtener información de tablas relacionadas utilizando reglas (Aliaga Ibarra and Miani Flores 2008).

PostgreSQL utiliza un modelo cliente/servidor y usa *multiprocesos* en vez de *multihilos* para garantizar la estabilidad del sistema. Un fallo en uno de los procesos no afecta el resto y el sistema continuará funcionando (Martin 2011).

1.5.2 Ventajas de PostgreSQL

PostgreSQL proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales tales como DB2 u Oracle. Lo siguiente es una breve lista de algunas de esas características:

DBMS Objeto-Relacional: PostgreSQL aproxima los datos a un modelo objeto-relacional, y es capaz de manejar complejas rutinas y reglas. Ejemplos de su avanzada funcionalidad son consultas: SQL declarativas, control de concurrencia multi-versión, soporte multi-usuario, transactions, optimización de consultas, herencia, y arrays.

Herencia: Las tablas pueden ser configuradas para heredar características de una tabla padre. Los datos son compartidos entre las tablas padre e hija(s). Las tuplas insertadas o eliminadas en la tabla hija serán insertadas o eliminadas en la tabla padre respectivamente.

Altamente Extensible: PostgreSQL soporta operadores, funcionales métodos de acceso y tipos de datos definidos por el usuario.

Soporte SQL Comprensivo: PostgreSQL soporta la especificación SQL99 e incluye características avanzadas tales como las uniones (joins).

Integridad Referencial: PostgreSQL soporta integridad referencial, la cual es utilizada para garantizar la validez de los datos de la base de datos.

API Flexible: La flexibilidad del API de PostgreSQL ha permitido a los vendedores proporcionar soporte al desarrollo fácilmente para el RDBMS PostgreSQL. Estas interfaces incluyen *Object Pascal*, *Python*, *Perl*, *PHP*, *ODBC*, *Java/JDBC*, *Ruby*, *TCL*, *C/C++*, y *Pike*.

Lenguajes Procedurales: PostgreSQL tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/pgSQL. Este lenguaje es comparable al lenguaje procedural de Oracle, PL/SQL. Otra ventaja de PostgreSQL es su habilidad para usar Perl, Python, o TCL como lenguaje procedural embebido.

MVCC: *MVCC*, o Control de Concurrencia Multi-Versión (*Multi-Versión Concurrency Control*), es la tecnología que PostgreSQL usa para evitar bloqueos innecesarios. Si alguna vez ha usado algún DBMS con capacidades SQL, tal como MySQL o Access, probablemente habrá notado que hay ocasiones en las que una lectura tiene que esperar para acceder a información de la base de datos. La espera está provocada por usuarios que están escribiendo en la base de datos.

Resumiendo, el lector está bloqueado por los escritores que están actualizando registros. Mediante el uso de MVCC, PostgreSQL se evita este problema por completo. MVCC está considerado mejor que el bloqueo a nivel de fila porque un lector nunca es bloqueado por un escritor. En su lugar, PostgreSQL mantiene una ruta a todas las transacciones realizadas por los usuarios de la base de datos. PostgreSQL es capaz entonces de manejar los registros sin necesidad de que los usuarios tengan que esperar a que los registros estén disponibles.

Cliente/Servidor: PostgreSQL usa una arquitectura proceso-por-usuario cliente/servidor. Esta es similar al método del Apache 1.3.x para manejar procesos. Hay un proceso maestro que se ramifica para proporcionar conexiones adicionales para cada cliente que intente conectar a PostgreSQL.

Write Ahead Logging (WAL): La característica de PostgreSQL conocida como *Write Ahead Logging* incrementa la dependencia de la base de datos al registro de cambios antes de que estos sean escritos en la base de datos. Esto garantiza que en el hipotético caso de que la base de datos se caiga, existirá un registro de las transacciones a partir del cual podremos restaurar la base de datos.

Esto puede ser enormemente beneficioso en el caso de caída, ya que cualquier cambio que no haya sido escrito en la base de datos pueden ser recuperados usando el dato que fue previamente registrado. Una vez el sistema ha quedado restaurado, un usuario puede continuar trabajando desde el punto en que lo dejó cuando cayó la base de datos (Aliaga Ibarra and Miani Flores 2008).

1.5.3 Desventajas de PostgreSQL.

- Limitaciones al escribir funciones y procedimientos en comparación con Oracle's PL/SQL o Sybase's T-SQL.
- Las tablas espaciosas, tablas particionadas, y con bloqueo altamente complicado siguen siendo ofrecidas por los vendedores propietarios de bases de datos.
- Carencia de herramientas de desarrollo propia.

PostgreSQL hace el análisis de datos mucho más sencillo a través de funcionalidades avanzadas de ANSI SQL 2003, las funciones ventanas, expresiones comunes de tabla

y consultas recursivas. Estas estructuras de consulta aumentan sustancialmente la expresividad del lenguaje SQL de PostgreSQL y permiten a los usuarios realizar preguntas interesantes en una sola gestión.

1.5.4 PostgreSQL vs. Access

Microsoft Access es un sistema de gestión de base de datos relacional. Permite crear ficheros con bases de datos que pueden ser consultados por otros programas. Dentro de un sistema de información entraría dentro de la categoría de Gestión y no en la de Ofimática como algunos creen.

El sistema ASSETS que se encuentra en uso en la UCLV para el control del claustro universitario presenta inconvenientes como:

Entre sus mayores inconvenientes figuran que no es multiplataforma, pues sólo está disponible para sistemas operativos de Microsoft, además inadecuado para grandes proyectos de software que requieren tiempos de respuesta críticos.

Sin embargo el sistema “Claustro UCLV” basado en PostgreSQL posibilita gran volumen de información o de usuarios debido a que es un sistema multiplataforma. Presenta una alta concurrencia permitiendo que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Posee gran variedad de tipos nativos como son: números de precisión arbitraria, direcciones IP (IPv4 e IPv6), entre otras. Soporta funciones que retornan "filas", donde la salida puede tratarse como un conjunto de valores que pueden ser tratados igual a una fila retornada por una consulta (query en inglés).

1.6 Herramientas de migración de datos

El acto de mover datos, o "migración de datos", suele ocurrir cuando se introducen nuevos servidores, bases de datos o computadoras. También ocurre cuando los conjuntos de datos y repositorios se unen en una única fuente. Los almacenes de datos contienen datos de cientos y miles de bases de datos diferentes y plataformas. En la práctica, un almacén de datos puede compararse con una plataforma de migración de datos que funciona en tiempo real, puesto que los datos siempre están disponibles. Pero la mayoría de las compañías y personas no utilizan los almacenes

de datos y en su lugar realizan tareas de movimiento de datos o migración desde un lugar a otro por varias razones.

1.6.1 Pentaho Data Integration (Kettle)

El producto Pentaho Data Integration es una solución de código abierto diseñada para utilizar el paradigma "extraer", "transformar" y "cargar", que es habitual en los entornos de tecnologías de bases de datos. La teoría es que los datos se convierten desde varios formatos extrayéndolos primero de la fuente, transformándolos hasta el formato de destino necesario y después cargándolos en una base de datos de destino. El producto tiene una interfaz gráfica de usuario que permite arrastrar y soltar para sus operaciones. Hay disponibles versiones básicas y avanzadas, pero la edición de la comunidad tiene ayuda propia y está disponible como un producto de código abierto. La versión profesional de pago proporciona un nivel más alto de funcionalidad para el uso en las organizaciones.

Muchas necesidades surgen cuando se necesita migrar las aplicaciones o las bases de datos. Puede que sea tan simple como una mejora a una nueva versión del sistema, o puede que implique cambiar a una nueva base de datos o aplicación. Después de una fusión o de una adquisición, a menudo se retiran las aplicaciones redundantes, pero los datos tienen que ser preservados en el sistema de supervivencia. Después de una migración, el viejo sistema y el nuevo sistema tendrán que coexistir.

La necesidad de crear sistemas encaminados a garantizar el funcionamiento de la economía cubana en las nuevas circunstancias y en el marco de las transformaciones organizativas que se están desarrollando en la actualidad, constituye un reto. La informatización como parte del avance tecnológico que precisa nuestro país, demanda una elevada capacidad tecnológica y operativa, lo que trae consigo a gran escala la utilización de modernos medios de procesamiento de la información, entre los que se encuentra el Kettle.

1.6.2 Consideraciones importantes del Kettle

Pentaho Data Integration es un motor de transformación, y desde el principio se observa que ha sido diseñado para cubrir las necesidades en la integración de datos. La plataforma *Open Source Pentaho Business Intelligence* cubre amplias necesidades

de análisis de los datos y de los informes empresariales. Las soluciones de Pentaho están escritas en Java y tienen un ambiente de implementación también basado en este lenguaje. Eso hace que sea una solución muy flexible para cubrir una amplia gama de necesidades empresariales.

Kettle es una herramienta de PDI que permite implementar los procesos extracción, transformación y carga de datos (ETL), la misma es de código abierto compuesta por cuatro componentes fundamentales: SPOON para el diseño gráfico de las transformaciones, *PAN* para la ejecución de los trabajos y las transformaciones, *CHEF* para el diseño de la carga de datos y *KITCHEN* para la ejecución de los trabajos *Batch* diseñados con *CHEF*.

Con esta herramienta se construyen las transformaciones (mínimo nivel de diseño) utilizando los pasos (steps). En un nivel superior se encuentran los trabajos (Jobs), que permiten ejecutar las transformaciones y otros componentes, y orquestar los procesos. PDI no es un generador de código, es un motor de transformación, donde los datos y sus transformaciones están separados. Las transformaciones y trabajos son almacenadas en formato *XML*, donde se especifican las acciones a realizar en los datos. Para construir las transformaciones, se utilizan los pasos o componentes, que se enlazan entre sí mediante saltos (steps), que determinan el flujo de datos entre los diferentes componentes. Para los trabajos, existe otro grupo de pasos, en los que se pueden realizar diferentes acciones (o ejecutar transformaciones). Los pasos o saltos en este caso determinan el orden de ejecución o la ejecución condicional.

Este *software* incluye herramientas para realizar consultas, generación de informes y reportes, análisis interactivo, tableros de mando, extracción, transformación y carga de datos (ETL) / integración de datos, minería de datos (data mining), y un servidor para la plataforma de BI (Business Intelligence) que lo ha convertido en la suite de software libre más popular.

Pentaho Data Integration ofrece amplia conectividad a una variedad de diversos datos, incluyendo todas las fuentes de datos no estructurados y semi-estructurados estructurados populares. Algunos ejemplos incluyen:

- Bases de datos relacionales estándar (por ejemplo, Oracle, DB2, MySQL, SQL Server y PostgreSQL).
- Hadoop (por ejemplo Apache Hadoop, Cloudera, Hortonworks, MapR).
- Bases de datos NoSQL (por ejemplo, MongoDB, Cassandra, HBase).
- Bases de datos analíticas (por ejemplo, Vertica, Greenplum, Teradata).
- Paquetes de aplicaciones empresariales (como SAP).
- Basado en la nube y aplicaciones SaaS (por ejemplo, Salesforce, Amazon Web Services).
- Archivos (por ejemplo, XML, Excel, archivos planos) y el servicio web APIs.

Para aumentar el rendimiento de los procesos de extracción, carga y entrega de datos, Pentaho ofrece las siguientes capacidades:

- Conectividad nativa y volumen de carga a la mayoría de las fuentes de datos comunes.
- La entrega de datos en un formato multi-dimensional para aplicaciones analíticas.
- La entrega de datos a través de los servicios de datos en tiempo real para aplicaciones de otros fabricantes tercero operacionales.

Kettle es una potente herramienta capaz de manipular y transformar información en todos aquellos proyectos dónde sea necesario trabajar con datos. Su solución para cubrir las necesidades de extracción, manipulación, validación y carga de datos desde múltiples fuentes de origen y en diferentes entornos puede resolver muchos problemas(Casters, Bouman et al. 2010).

1.7 Lenguajes de modelado de objetos

El lenguaje de modelado de objetos es un conjunto estandarizado de símbolos y de modos de disponerlos para modelar un diseño de software orientado a objetos.

Algunas organizaciones los usan extensivamente en combinación con una metodología de desarrollo de software para avanzar de una especificación inicial a un plan de implementación y para comunicar dicho plan a todo un equipo de desarrolladores. El uso de un lenguaje de modelado es más sencillo que la auténtica

programación, pues existen menos medios para verificar efectivamente el funcionamiento adecuado del modelo. Esto puede suponer también que las interacciones entre partes del programa den lugar a sorpresas cuando el modelo ha sido convertido en un software funcionante.

Algunos metodólogos del software orientado a objetos distinguen tres grandes "generaciones" cronológicas de técnicas de modelado de objetos:

- En la primera generación, tecnólogos aislados y grupos pequeños desarrollaban técnicas que resolvían problemas que se encontraban de primera mano en los proyectos de desarrollo orientado a objetos. En esta generación se incluye a autores y técnicas como Rumbaugh, Jacobson, Booch, los métodos formales, Shlaer-Mellor y Yoda.
- En la segunda generación se reconoció que muchas de las mejores prácticas pertenecían a diferentes métodos del fragmentado terreno de la metodología orientada a objetos. Se realizaron múltiples intentos para integrar dichas técnicas en marcos coherentes tales como fusión. En cualquier caso, la comunidad del software orientado a objetos empezaba a reconocer los beneficios que la standarización de las técnicas conllevaría: abandono de las "buenas" formas de hacer las cosas en favor de "la" manera adecuada, que permitiría un lenguaje y unas prácticas comunes entre los diferentes desarrolladores.
- La tercera generación consiste en intentos creíbles de crear dicho lenguaje unificado por la industria, cuyo mejor ejemplo es UML.

1.7.1 UML como lenguaje de modelación

El UML es un lenguaje gráfico que sirve para modelar, diseñar, estructurar, visualizar, especificar, construir y documentar. El UML proporciona un vocabulario común para toda la cadena de producción, desde quien recaba los requisitos de los usuarios, hasta el último programador responsable del mantenimiento.

Es un sistema de notaciones (que, entre otras cosas, incluye el significado de sus notaciones) destinado a los sistemas de modelación que utilizan conceptos orientados a objetos. UML combina notaciones provenientes desde:

- ✓ Modelación Orientado a Objetos
- ✓ Modelación de Datos
- ✓ Modelación de Componentes
- ✓ Modelación de Flujos de Trabajo (*Workflows*).

El UML es un estándar de la industria para construir modelos orientados a objetos. Nació en 1994 por iniciativa de *Grady Booch* y *Jim Rumbaugh* para combinar sus dos famosos métodos: el de *Booch* y la Técnica de Modelación de Objetos (OMT por sus siglas del inglés (*Object Modeling Technique*)). Más tarde se les unió *Ivar Jacobson*, creador del método Ingeniería de Software Orientada a Objetos (OOSE, acrónimo del inglés (*Object-Oriented Software Engineering*)). En respuesta a una petición de OMG (*Object Management Group*) asociación para fijar los estándares de la industria) para definir un lenguaje y una notación estándar del lenguaje de construcción de modelos, en 1997 propusieron el UML (Simon, Steve et al. 2010).

1.7.2 Conceptos básicos de UML

Para comprender UML basta con analizar cada una de las palabras que lo componen por separado.

Lenguaje: el UML es, precisamente, un lenguaje. Lo que implica que este cuente con una sintaxis y una semántica. Por lo tanto, al modelar un concepto en UML, existen reglas sobre cómo debe agruparse los elementos del lenguaje y el significado de esta agrupación.

Modelado: el UML es visual. Mediante su sintaxis se modelan diferentes aspectos del mundo real que permiten una mayor interpretación y entendimiento de éste.

Unificado: Por que unifica varias técnicas de modelado en una única. Por provenir el UML de técnicas orientadas a objetos, el UML se crea con la fuerte intención de que este permita un correcto modelado orientado a objetos.

UML está consolidado como un lenguaje estándar en el análisis y diseño de sistemas de cómputo. Mediante UML es posible establecer la serie de requerimientos y estructuras necesarias para plasmar un sistema de software previo al proceso intensivo de escribir código.

1.7.3 Inconvenientes de UML

A pesar de su status de estándar ampliamente reconocido y utilizado, UML siempre ha sido muy criticado por su carencia de una semántica precisa, lo que ha dado lugar a que la interpretación de un modelo UML no pueda ser objetiva.

Otro problema de UML es que no se presta con facilidad al diseño de sistemas distribuidos. En tales sistemas cobran importancia factores como transmisión, serialización, persistencia y no cuenta con maneras de describir tales factores. No se puede, por ejemplo, usar para señalar que un objeto es persistente o remoto, o que existe en un servidor que corre continuamente y es compartido entre varias instancias de ejecución del sistema analizado.

1.7.4 Perspectivas de UML

UML será el lenguaje de modelación orientado a objetos estándar predominante en los próximos años. Las razones para ello son las siguientes:

- ✓ Participación de metodólogos influyentes.
- ✓ Participación de importantes empresas.
- ✓ Aceptación del OMG como notación estándar.
- ✓ Herramientas que proveen la notación UML
- ✓ Edición de libros, Congresos, Cursos

En UML hay 13 tipos diferentes de diagramas y son los siguientes:

- Diagramas estructurales que enfatizan en los elementos que deben existir en el sistema modelado:
 - ✓ Diagrama de clases (incluyendo Diagrama de Objetos).
 - ✓ Diagrama de componentes.
 - ✓ Diagrama de objetos.
 - ✓ Diagrama de estructura compuesta.
 - ✓ Diagrama de despliegue.
 - ✓ Diagrama de paquetes.

- Los diagramas de comportamiento que enfatizan en lo que debe suceder en el sistema modelado.
 - ✓ Diagrama de actividades.
 - ✓ Diagrama de casos de uso.
 - ✓ Diagrama de estados.
 - ✓ Diagrama de secuencias.
 - ✓ Diagrama de colaboración o diagrama de comunicación.
 - ✓ Diagrama global de interacciones.
 - ✓ Diagrama de tiempos.

1.7.5 Utilidad de UML

UML es un lenguaje para modelación de propósito general evolutivo, ampliamente aplicable y que puede ser soportado por herramientas e industrialmente estandarizado. Se aplica a una multitud de diferentes tipos de sistemas, dominios, y métodos o procesos.

- Como lenguaje de *propósito general*, se enfoca en el corazón de un conjunto de conceptos para la adquisición, compartimiento y utilización de conocimientos emparejados con mecanismos de extensión.
- Como un lenguaje para modelación *ampliamente aplicable*, puede ser aplicado a diferentes tipos de sistemas (software y no - software), dominios (negocios versus software) y métodos o procesos.
- Como un lenguaje para modelación *soportable por herramientas*, las herramientas ya están disponibles para soportar la aplicación del lenguaje para especificar, visualizar, construir y documentar sistemas.
- Como un lenguaje para modelación *industrialmente estandarizado*, no es un lenguaje cerrado, propiedad de alguien, sino más bien, un lenguaje abierto y totalmente extensible y reconocido por la industria.

UML posibilita la captura, comunicación y nivelación de conocimiento estratégico, táctico y operacional para facilitar el incremento de valor, aumentando la calidad, reduciendo costos y reduciendo el tiempo de presentación al mercado; manejando

riesgos y siendo proactivo para el posible aumento de complejidad o cambio (Rumbaugh, Booch et al. 1997).

Debido a que UML evolucionó primeramente de varios métodos orientados al objeto de segunda generación (en cuanto a nivel de notación), la mayoría de aplicadores de UML creen que sólo es relativo a sistemas de software orientados a objeto, cuando actualmente UML no es simplemente un lenguaje para modelación orientado a objeto de tercera generación, sino un "lenguaje para modelación unificado" relativo a sistemas en general.

El éxito de UML es medido por su uso apropiado en proyectos exitosos. UML no garantiza el éxito, sino que permite a los aplicadores enfocarse en la distribución de valor, usando un lenguaje para modelación consistente, estandarizado y soportable por herramientas (Schmuller 2000).

A pesar de que UML es un lenguaje preciso que utiliza las mejores técnicas, se le puede realizar una extensión, así mismo mejoras en los conceptos de modelación, y muchas técnicas avanzadas pueden ser definidas usando UML como base. El UML ha integrado muchas ideas dispares, de manera que dicha integración acelerará el uso de Orientado a Objetos (OO). Hay dos aspectos de “unificado” que UML logra: elimina efectivamente muchas de las diferencias entre los lenguajes de modelación y métodos previos y unifica las perspectivas entre muchas diferentes clases de sistemas, fases de desarrollo y conceptos internos. Cada vez más usuarios adoptan UML debido a sus características similares en cuanto a semántica y anotación a los métodos *Booch*, *OMT*, *OOSE*, entre otros, la contribución de *UML Partners*, la incorporación de la información de la comunidad general (Maliaños 2006).

1.7.6 UML vs otros lenguajes de modelación

El UML, al ser el sucesor legítimo de los métodos *Booch*, *OMT* y *OOSE*, permite adaptarse fácilmente a sus usuarios, así como a otros usuarios de otros métodos. El UML resulta más expresivo y uniforme que los métodos *Booch*, *OMT*, *OOSE*, entre otros, a la vez que permite en los proyectos modelar nuevas cosas. Provee una expresividad e integridad holística mejorada, respecto a otros lenguajes de modelación visual. Es fácil de aprender y usar, ya sea respecto a las técnicas más

avanzadas, es decir, estereotipos y propiedades, así como algunos cambios en la anotación y semánticas(Maliaños 2006).

Consideraciones finales del capítulo

En función de la implementación de una aplicación web que respondiera a las necesidades de gestión de la información de la Dirección de Recursos Humanos en la UCLV en este capítulo se determinaron las herramientas a utilizar para el logro de los objetivos propuestos. Entre estas se pueden citar: PHP como lenguaje de programación, UML como lenguaje de modelado para el sistema y como motor de base de datos PostgreSQL.

Capítulo 2: Diseño e implementación del Sistema.

En el presente capítulo se documenta el diseño de la base de la base de datos (BD) para la visualización de la información del claustro universitario en la UCLV. Se detallan aspectos de interés del diagrama entidad relación y se exponen esquemas del modelo relacional. Además se realizará la descripción de los avances en la solución de la problemática que da lugar a este trabajo. También se muestran los principales resultados del análisis diseño mediante diagramas UML.

2.1 Requisitos del Sistema

Después de un análisis detallado en el proceso de diseño del sistema, se han definido varios requisitos funcionales y no funcionales, que deben ser tenidos en cuenta para la implementación y correcto funcionamiento del sistema.

Un requisito es una necesidad documentada sobre el contenido, forma o funcionalidad de un producto o servicio y, en su conjunto, establecen qué debe hacer o qué cualidad debe poseer el sistema.

Los requisitos deben satisfacer varias características:

Necesidad: Deben ser algo necesario para el producto.

No ambigüedad: Su descripción debe ser clara, precisa y tener una única interpretación posible.

Concisión: Debe redactarse en un lenguaje comprensible por los usuarios del sistema en lugar de uno de tipo técnico y especializado, aunque aun así debe referenciar los aspectos importantes.

Consistencia: Ningún requisito debe entrar en conflicto con otro diferente, ni con parte de otro.

Compleitud: Deben contener en sí mismos toda la información necesaria, y no remitir a otras fuentes externas que los expliquen con más detalle.

Alcanzabilidad: Deben ser objetivos realistas, posibles de ser alcanzados con el dinero, el tiempo y los recursos disponibles.

Verificabilidad: Se debe poder verificar con absoluta certeza, si fueron satisfechos o no.

2.1.1 Requisitos funcionales

Los requerimientos funcionales permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara, lo que debe hacer el mismo. Como se muestran en la Tabla 2.1

Requisito	Nombre
1	Autenticar usuario
2	Visualizar información general de empleados por sexo
3	Visualizar información general de empleados por sexo y color de la piel
4	Visualizar información de empleados docentes por sexo
5	Visualizar información de empleados docentes por sexo y categoría
6	Visualizar información de empleados investigadores por sexo
7	Visualizar información de empleados investigadores por sexo y categoría
8	Visualizar combustible de empleados por sexo
9	Visualizar reporte de nómina incorrecta de docentes e investigadores
10	Visualizar reporte de nómina incorrecta de profesores a tiempo parcial (PTP) por categoría.

Tabla 2. 1 Requisitos funcionales

2.1.2 Requisitos no funcionales

Registrar los requisitos no funcionales de un sistema, como son el caso de las restricciones del entorno, de la implementación, dependencia de la plataforma,

rendimiento, facilidad de realizar mantenimiento e incrementación de este, es un aspecto básico a tener en cuenta para garantizar un funcionamiento óptimo de este, dándole a los usuarios la garantía de un producto seguro, usable, agradable y conveniente; por lo que debe tener una alta tendencia a ser visitado y usado. A continuación se detallan los requisitos no funcionales de este sistema:

Interfaz del sistema: Este entorno web propuesto garantiza una interfaz gráfica grata para los usuarios finales, además proporciona una fácil familiarización el sistema ya que no está destinada específicamente para expertos en el uso de las computadoras, por lo que debe mantener una interacción usuario-sistema bastante sencillo y entendible.

Usabilidad: La aplicación se desplegará en la Dirección de Recursos Humanos de la UCLV con el objetivo de tener una visualización de la información referente al claustro universitario.

Rendimiento: El sistema garantiza la disponibilidad para los usuarios las 24h, asegurando así su máxima explotación.

Soporte: El sistema cuenta con una aplicación web que se sirve de una base de datos donde se almacena totalmente todo lo que sucede en él. La asistencia y mantenimiento del producto es responsabilidad de un especialista, el producto a la interacción usuario-sistema o a las nuevas necesidades de la aplicación.

Portabilidad: Este sistema podrá utilizar tanto para el sistema operativo Windows como para sistemas operativos Linux (Ubuntu) ya que utiliza un servidor web apache y un servidor de base de datos PostgreSQL que son comunes en ambos sistemas operativos y constan del mismo principio de funcionamiento.

Seguridad: No se establece un nivel de acceso al sistema, para garantizar un control sobre la información, definiendo una política de usuario con un rol con los mismos privilegios.

Rol del sistema:

Se maneja un tipo de rol como se muestra en la Tabla 2.2

Rol	Descripción
Técnico de RH	Este rol puede acceder al sistema y visualizar toda la información del claustro universitario en la UCLV.

Tabla 2. 2 Rol del sistema

Requerimientos de Software:

Cliente: Navegador Mozilla Firefox (versión 6.0 en adelante), Internet Explorer (versión 7.0 en adelante), Google Chrome (cualquiera de sus versiones) y Opera.

Servidor: Sistema de Gestión de Base de Datos (SGBD) PostgreSQL, Servidor de aplicaciones Apache que soporte tecnología PHP5 (XAMP o WAMP).

Requerimientos de hardware:

Cliente: Pentium, con 256 MB de RAM (mínimo) con conexión con la red de la Dirección de Recursos Humanos de la UCLV.

Servidor: servidor con 256 MB de memoria RAM (mínimo), conexión con la red del entorno donde se desplegará.

Requerimientos Legales: El sistema pertenece a la UCLV, basándose en un estándar que se rige por normas internacionales y cumple con las normas y leyes establecidas en nuestro país.

Restricciones en el diseño y la implementación: El proyecto se acomodará a los estándares aplicados en la UCLV para la creación de todos los sitios que esta presenta. Se decidió utilizar PHP como lenguaje de programación, como herramienta de desarrollo el Joomla y para el diseño se emplean algunos artefactos que propone RUP apoyado en el estándar notacional UML.

2.2 Modelación conceptual de datos

La modelación conceptual es el proceso de creación de representaciones abstractas de un dominio de aplicación en términos de conceptos familiares a los actores de ese dominio y no en términos técnicos. Esta requiere de notaciones, herramientas y técnicas para representar datos y procesos. Las investigaciones actuales tratan de

estrechar la barrera entre los conceptos del mundo real y su representación en modelos conceptuales.

Se puede dividir el proceso de modelación conceptual en dos grupos de actividades relacionadas entre sí: análisis de entidades y análisis funcional. El análisis de entidades está relacionado con el análisis de un problema a resolver para determinar los tipos de entidades, sus atributos, y las interrelaciones entre ellos. El análisis funcional, por otra parte, se corresponde con la determinación de las funciones fundamentales con las cuales se relaciona el problema modelado. Los resultados de estos dos pasos deben hacerse en paralelo para obtener una mejor comprensión sobre cuáles funciones se corresponden con tales entidades.

Los modelos semánticos de datos son beneficiosos en las actividades de diseño conceptual debido a la economía de expresión, ya que las operaciones se definen explícitamente en las interrelaciones, y el modelo de datos incluye la semántica en sí; el mantenimiento de la integridad se da a través de la capacidad de definición de restricciones de integridad y a su vez permite un nivel más alto de abstracción, dotado de capacidades para modelar situaciones del mundo real; con estos modelos los diseñadores usan operaciones y restricciones predefinidas sin tener que implementarlas a bajo nivel.

Es ampliamente aceptado el uso del modelo semántico de datos conocido como modelo Entidad-Relación (ER) introducido por Chen y sus extensiones para el diseño del Esquema Conceptual Global (ECG). El modelo ER extendido (ERE) usa los conceptos de entidad, propiedad, interrelación y subtipo para incorporar información semántica importante acerca del mundo. La mayoría de la información semántica de estos modelos está basada en las interrelaciones entre entidades.

El modelo ERE a veces es impreciso para transformar un diagrama ERE a los formalismos de un manejador de BD específico y deja algunos detalles sin concretar (por ejemplo la creación de índices), y es incapaz de tratar las restricciones de integridad relativas al problema (del inglés business rules), excepto en muy pocos casos especiales. No obstante, este modelo ha trascendido como la técnica de modelación conceptual reconocida como ayuda al diseño de BD relacionales.

Cualquier herramienta de ayuda a la caracterización del ECG debe ofrecer una amplia variedad de construcciones del modelo ERE que permitan capturar mejor la semántica del universo de discurso y realicen validaciones a los diagramas para que el resultado sea la obtención de esquemas libres de errores.

Existen varias herramientas de ayuda a la modelación conceptual de datos mediante la creación de diagramas ERE, entre las que se puede citar ERECASE y ER/Studio.

La modelación conceptual es el proceso de creación de representaciones abstractas de un dominio de aplicación en términos de conceptos familiares a los actores de ese dominio y no en términos técnicos. Esta requiere de notaciones, herramientas y técnicas para representar datos y procesos. Las investigaciones actuales tratan de estrechar la barrera entre los conceptos del mundo real y su representación en modelos conceptuales. Se puede dividir el proceso de modelación conceptual en dos grupos de actividades relacionadas entre sí: análisis de entidades y análisis funcional. El análisis de entidades está relacionado con el análisis de un problema a resolver para determinar los tipos de entidades, sus atributos, y las interrelaciones entre ellos. El análisis funcional, por otra parte, se corresponde con la determinación de las funciones fundamentales con las cuales se relaciona el problema modelado. Los resultados de estos dos pasos deben hacerse en paralelo para obtener una mejor comprensión sobre cuáles funciones se corresponden con tales entidades.

2.2.1 Esquema conceptual de la BD visualizar información del claustro universitario.

En este trabajo se ha diseñado una base de datos para la visualización de la información del claustro universitario de la UCLV donde se almacenaran los datos referentes al personal.

Instituyendo una metodología en el diseño del modelo relacional se han integrado herramientas que desde el punto de vista teórico y práctico solucionan esta problemática.

Para el modelado de la base de datos se utilizó la herramienta ERECASE la cual ofrece entre sus funcionalidades: una fuerte capacidad para el diseño lógico; sincronización bidireccional del diseño lógico y físico; construye automáticamente el

modelo físico de la base de datos y permite la ingeniería inversa de la misma; genera documentación en HTML; cuenta con facilidades de reportes y un repositorio para la cooperación entre equipos. Una de las principales ventajas de la herramienta es que permite la creación de submodelos, logrando así una mayor organización en el diseño.

El conjunto de identidades identificadas se relacionan como se muestra en la Figura 2.1

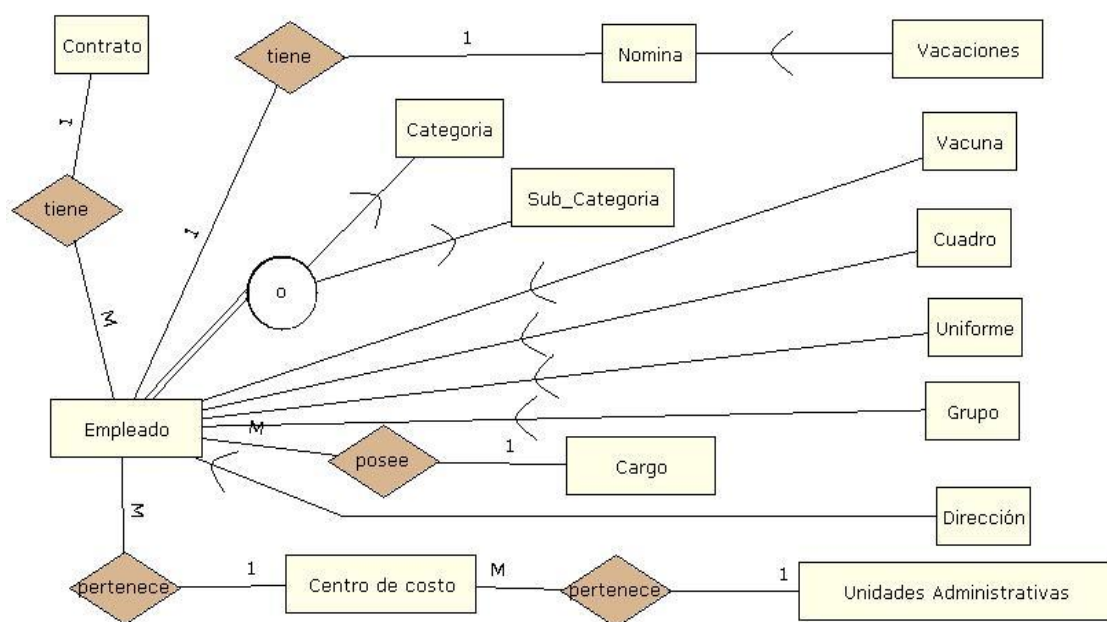


Figura 2.1 Modelo entidad relación de la base de datos

2.1.2 Modelo para la visualización de la información del claustro universitario en la UCLV.

Para la creación de un nuevo modelo para la visualización de la información se realizó un modelo de las clases involucradas. Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.

Para la obtención de la información del claustro universitario se parte de una tabla *empleado general* que se desglosa en otras como: vacuna, cuadro, uniforme, grupo y dirección. También el modelo contendrá las tablas contrato, categoría, subcategoría,

cargo, centro de costo y unidades administrativas. El diseño relacional de la base de datos se muestra en la Figura 2.2

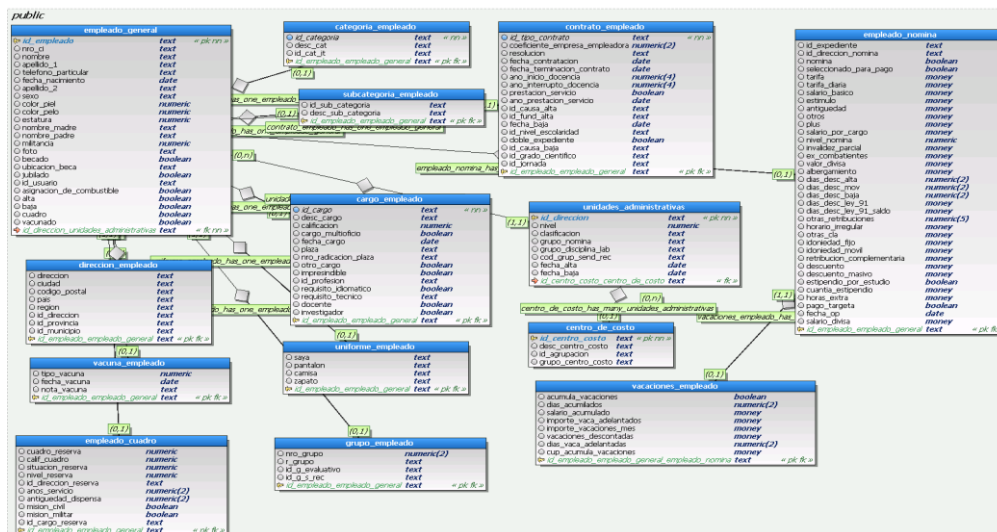


Figura 2. 2 Modelo para la visualización de la información del claustro en la UCLV

2.3 Actor del sistema

El sistema “Claustro UCLV” interactúa con un solo actor con el fin de la visualizar la información del claustro universitario de forma general, de acuerdo a si son docentes e investigadores o también profesores a tiempo parcial como se muestra en la Tabla 2.3

Usuario	Descripción
Técnico de RH	Este rol puede acceder al sistema y visualizar toda la información del claustro universitario en la UCLV y actualizar el sistema.
Usuario	Este rol puede acceder al sistema y visualizar toda la información del claustro universitario en la UCLV.

Tabla 2.3 Actor del sistema

2.4 Modelo de casos de uso del sistema

Un caso de uso es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso, representando gráficamente uno o más escenarios que indican cómo debería interactuar el sistema con el usuario o con otro sistema para lograr un objetivo específico. Esto resulta una técnica muy efectiva para obtener los requerimientos de un nuevo sistema o la actualización de un software. Un modelo de casos de uso muestra, por tanto, los distintos requisitos funcionales que se esperan de una aplicación o sistema y cómo se relacionan con su entorno (usuarios u otras aplicaciones).

2.4.1 Formatos de los casos de uso

En la práctica, los casos de uso pueden expresarse con diverso grado de detalle y de aceptación de las decisiones concernientes al diseño. En otras palabras, un mismo caso de uso puede escribirse en diferentes formatos y con diversos niveles de detalle.

Existe una división fundamental: casos con formato de alto nivel y expandido.

Formato de alto nivel

Un caso de uso de alto nivel describe un proceso muy brevemente, casi siempre en dos o tres enunciados. Conviene servirse de este tipo de caso durante el examen inicial de los requerimientos y del proyecto, a fin de entender rápidamente el grado de complejidad y de funcionalidad del sistema. Estos casos son muy sucintos y vagos en las decisiones de diseño.

Formato expandido

Un caso de uso expandido describe un proceso más a fondo que el de alto nivel. La diferencia básica con el caso de uso de alto nivel consiste en que tiene una sección destinada al curso normal de los eventos, que los describe paso por paso. Durante la fase de especificación de requerimientos, conviene escribir en el formato expandido los casos más importantes y de mayor influencia; en cambio, los menos importantes pueden posponerse hasta el ciclo de desarrollo en el cual van a ser abordados.

2.4.2 Diagrama de actor y casos de uso del sistema

Cuando se accede al sitio se ingresa al usuario empleado el cual se presenta como forma de portal web para la interacción entre el usuario y el contenido de la información del claustro universitario, tales como obtener un listado de los empleados por sexo, color de la piel, categoría o subcategoría, etc. Como se muestra en el caso de uso de la Figura 2.3, en la aplicación el usuario puede acceder y hacer uso de las actividades que presenta.

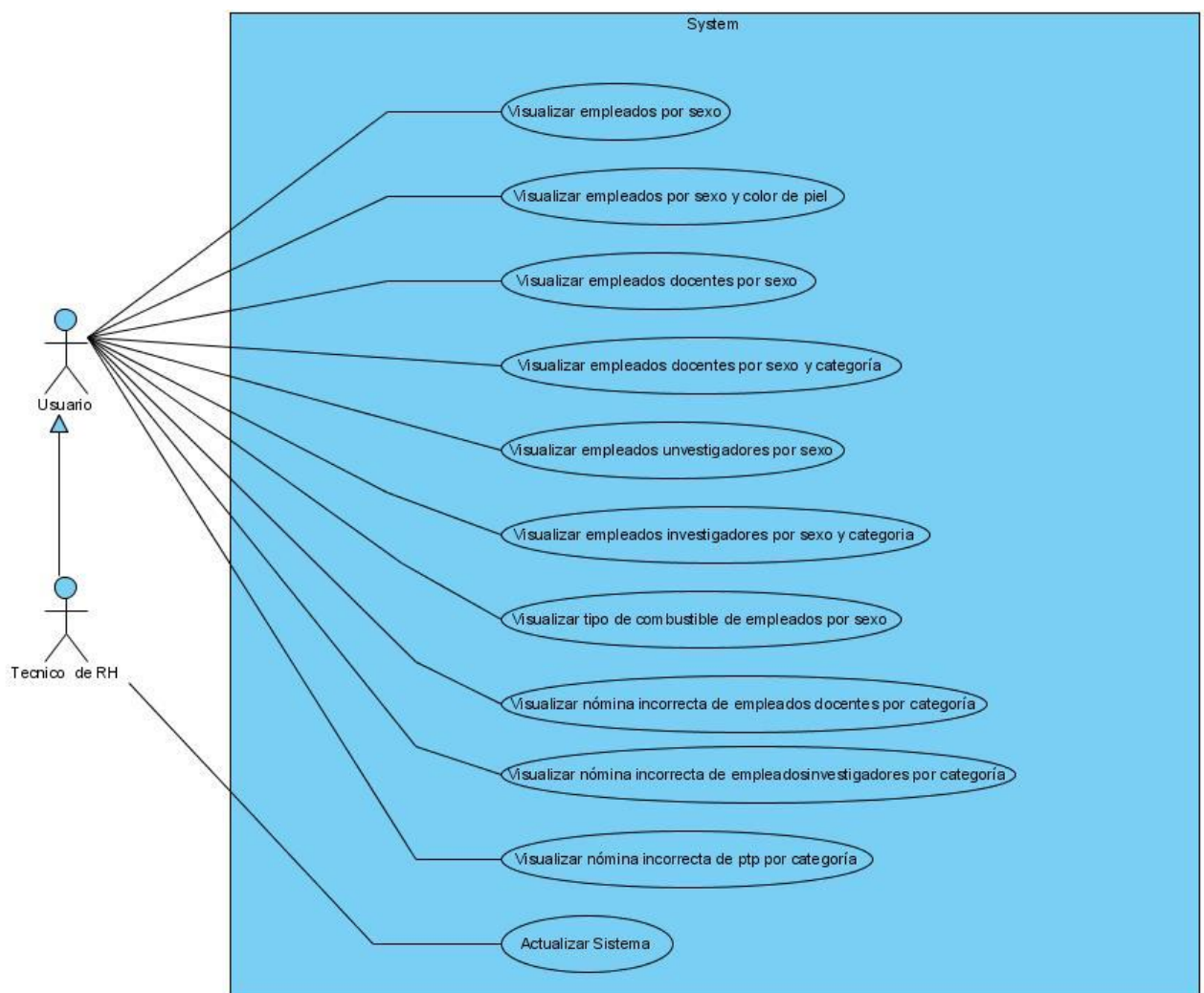


Figura 2.3 Caso de uso para el empleado

Para mayor comprensión del actor empleado se analizan los siguientes casos de uso que realiza el actor Empleado y se realiza una explicación de las funciones que le corresponden dentro del sistema en la Tabla 2.4

Caso de uso	Descripción
Visualizar empleados por sexo	Obtiene la visualización de un listado de los empleados del claustro universitario distinguiéndolos por sexo.
Visualizar empleados por sexo y color de piel	Obtiene la visualización de la información de los empleados de acuerdo al sexo y además del color de piel si son blancos, de color, mestizos o amarillos en caso de no ser ninguna de las anteriores.
Visualizar empleados docentes por sexo	Obtiene una visualización de la información de los empleados docentes por sexo.
Visualizar empleados docentes por sexo y categoría	Obtiene una visualización de la información de los empleados docentes por sexo y categoría docente en este caso son: titular, auxiliar, asistente e instructor.
Visualizar empleados investigadores por sexo	Obtiene una visualización de la información de los empleados investigadores de acuerdo a si son hombres o mujeres
Visualizar empleados investigadores por sexo y categoría	Obtiene una visualización de la información de los empleados investigadores por sexo y categoría investigativa en este caso son: titular, auxiliar, agregado o aspirante a investigador.
Visualizar tipo de combustible de empleados por sexo	Obtiene la visualización de la información de los empleados que reciben combustible por sexo.
Visualizar nómina de empleados docentes por categoría	Obtiene la visualización del salario incorrecto de los empleados docentes atendiendo a su categoría en este caso son: titular, auxiliar, asistente e instructor.
Visualizar nómina de	Obtiene la visualización del salario incorrecto de los

empleados investigadores por categoría	empleaos instructores docentes atendiendo a su categoría en este caso son: titular, auxiliar, agregado y aspirante a investigador.
Visualizar nómina incorrecta de PTP por categoría	Obtiene una visualización del salario incorrecto de los profesores a tiempo parcial atendiendo a su categoría en este caso son: titular, auxiliar, asistente e instructor.
Actualización de BD	Actualiza la base de datos del Sistema.

Tabla 2. 4 Descripción de los casos de uso para el empleado

2.5 Diagrama de navegación

Los mapas de navegación proporcionan una representación esquemática de la estructura del hipertexto, indicando los principales conceptos incluidos en el espacio de la información y las interrelaciones que existen entre ellos. Un mapa es, por ejemplo, una representación completa (o resumida) del sitio web para orientar al usuario durante el recorrido o facilitarle el acceso directo al lugar que le interese. Refleja la estructura de la *web* por medio de enlaces a los nodos principales, y estos también pueden desarrollarse para mostrar sub-nodos. Se puede concluir entonces que el diagrama de navegación es el encargado de modelar como se efectúa la navegación por el sitio.

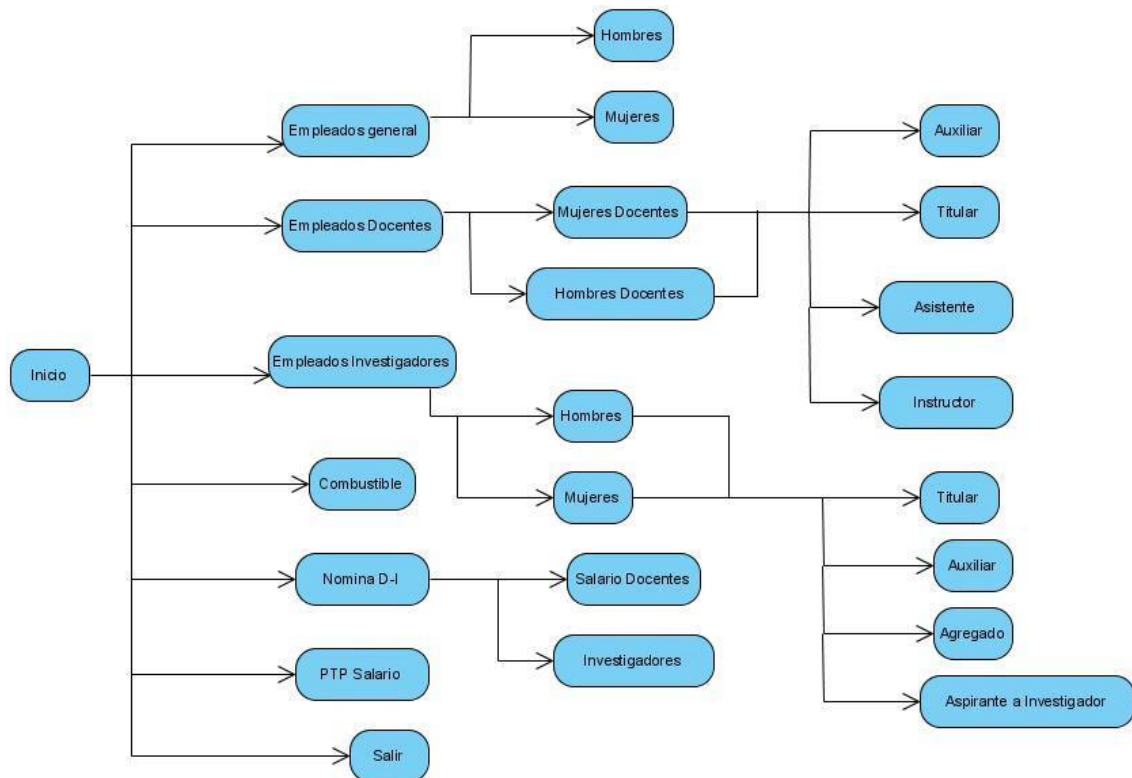


Figura 2. 4 Diagrama de navegación del usuario

2.6 Arquitectura del sistema Cliente-Servidor

El sistema cliente servidor puede ser definido como una configuración que incluye estaciones de trabajo, a las que denominaremos “Clientes”, conectadas a repositorios de programas (Servidores) a través de una red. El servidor es un elemento que generalmente reside en una computadora central, que recibe, procesa y responde una petición de información. El servidor tiene la capacidad de atender en forma simultánea múltiples conexiones provenientes de varios orígenes (diferentes clientes). El cliente está constituido por una computadora o aplicación software que facilita a un usuario enviar una solicitud de información y obtener una respuesta esperada.

2.6.1 Diagrama de despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria.

Este diagrama muestra la topología del sistema Claustro UCLV. La Figura 2.5 describe explícitamente la división cliente-servidor mediante los paquetes denominados clientes y servidores:

- En el cliente se utiliza algún navegador para ver las páginas Web que permiten al usuario interactuar con el sistema.
- En el servidor de aplicación (Páginas Web) se publican las páginas.

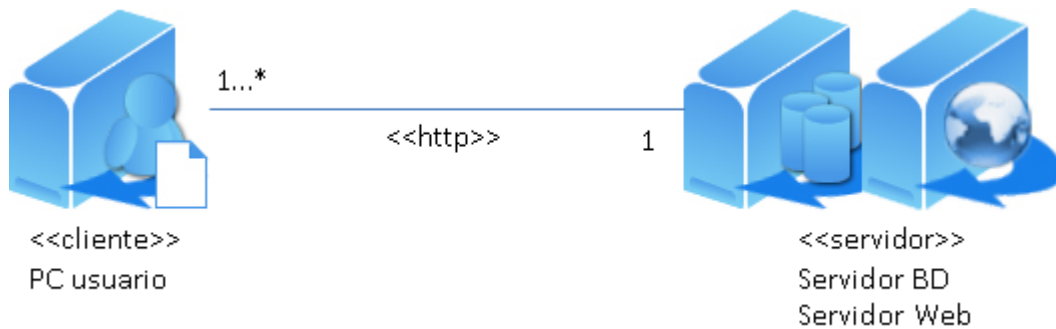


Figura 2.5 Diagrama de despliegue

2.7 Diagrama de transición de estados

Un diagrama de estados muestra una máquina de estados, destacando el flujo de control entre estados. Una máquina de estados es un comportamiento que especifica las secuencias de estados por las que pasa un objeto a lo largo de su vida en respuesta a eventos, junto con sus respuestas a esos eventos.

El diagrama que se muestra en la Figura 2.6 muestra la obtención de un reporte de los empleados investigadores hombres los cuales son aspirante a investigador.

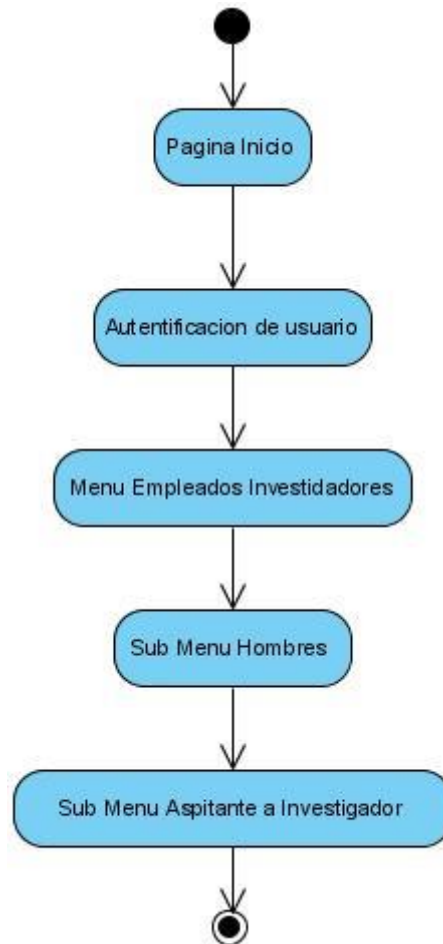


Figura 2. 6 Diagrama de transición de estados para aspirantes a investigador

2.8 SQL como lenguaje de consultas

El SQL (Structured Query Language), o Lenguaje de Consulta Estructurado es un lenguaje de acceso a bases de datos que soporta los conceptos fundamentales del modelo relacional: relaciones, dominios, tuplas y atributos, lo que permite obtener los beneficios que brinda este tipo de modelo. Los principales Sistemas Gestores de Bases de Datos Relacionales (SGBDR) como Oracle, Sybase, DB/2, Informix, Ingres o SQL Server utilizan este lenguaje que se ha transformado en el estándar para definición y manipulación de datos.

Diversos son los beneficios que puede otorgar: es un lenguaje de alto nivel, de alta portabilidad entre plataformas, posee un fundamento relacional, uso interactivo y programado, una arquitectura cliente servidor, entre otras ventajas.

2.8.1 Reportes de empleados implementados mediante consultas SQL

Para la visualización de la información del claustro universitario en la UCLV se emplearon consultas como las que se muestran a continuación:

La siguiente consulta es para seleccionar del claustro a los profesores titulares mujeres que sean investigadores.

SELECT

public.Empleado_general.nombre,
public.Empleado_general.apellido_1,
public.Empleado_general.apellido_2,
public.Empleado_general.id_centro_costo,
public.Cargo_Empleado.id_cargo

FROM

public.Empleado_general

INNER JOIN public.Cargo_Empleado **ON**

(public.Empleado_general.id_Empleado=public.Cargo_Empleado.id_Empleado_Empleado_general)

INNER JOIN public.Empleado_nomina **ON**

(public.Empleado_general.id_Empleado=public.Empleado_nomina.id_Empleado_Empleado_general)

WHERE

(public.Empleado_general.baja = 'N') **AND**

(public.Empleado_general.sexo = 'F') **AND**

(public.Empleado_general.id_Empleado =
public.Cargo_Empleado.id_Empleado_Empleado_general) **AND**

(public.Cargo_Empleado.investigador = 'Y') **AND**

(public.Empleado_nomina.salario_basico = '7350000')

ORDER BY

public.empleado_general.id_centro_costo

La siguiente consulta es para seleccionar del claustro a los profesores instructores hombres que sean investigadores.

SELECT

public.empleado_general.nombre,
public.empleado_general.apellido_1,
public.empleado_general.apellido_2,
public.empleado_general.id_centro_costo,
public.cargo_empleado.id_cargo

FROM

public.empleado_general

INNER JOIN public.cargo_empleado **ON**
(public.empleado_general.id_empleado=public.cargo_empleado.id_empleado_empleado_general)

INNER JOIN public.empleado_nomina **ON**
(public.empleado_general.id_empleado=public.empleado_nomina.id_empleado_empleado_general)

WHERE

(public.empleado_general.baja = 'N') **AND**

(public.empleado_general.sexo = 'M') **AND**

(public.empleado_general.id_empleado =
public.cargo_empleado.id_empleado_empleado_general) **AND**

(public.cargo_empleado.docente = 'Y') **AND**

(public.empleado_nomina.salario_basico = '5550000')

ORDER BY

public.empleado_general.id_centro_costo

La siguiente consulta es para seleccionar del claustro a los profesores docentes que sean titulares que posean un salario básico incorrecto.

SELECT

```
public.Empleado_general.nombre,  
public.Empleado_general.apellido_1,  
public.Empleado_general.apellido_2,  
public.Empleado_nomina.salario_basico
```

FROM

```
public.Empleado_general
```

```
INNER JOIN public.Empleado_nomina ON  
(public.Empleado_general.id_Empleado=public.Empleado_nomina.id_Empleado_Empleado_general)
```

```
INNER JOIN public.Cargo_Empleado ON  
(public.Empleado_general.id_Empleado=public.Cargo_Empleado.id_Empleado_Empleado_general)
```

WHERE

```
(public.Empleado_general.baja = 'N') AND  
(public.Cargo_Empleado.docente = 'Y') AND  
(public.Cargo_Empleado.id_Cargo = '4010') AND  
(public.Empleado_nomina.salario_basico <> '7350000')
```

ORDER BY

```
public.Empleado_general.id_Centro_costo
```

2.9 Sistemas basados en Microsoft Access

Una base de datos suele definirse como un conjunto de información organizada sistemáticamente. En la terminología propia de las bases de datos hay tres conceptos claves dentro de las tablas: campo, registro y dato.

Un campo es cada uno de los tipos de datos que se van a usar. Se hace referencia a los campos por su nombre. Un registro está formado por el conjunto de información en particular. Un dato es la intersección entre un campo y un registro.

2.9.1 Limitaciones del sistema ASSETSS de la UCLV

El sistema ASSETS se encuentra en explotación en el Departamento de Recursos Humanos de la UCLV el cual hace uso de Microsoft ACCESS como sistema de gestor de base de datos.

Este sistema para el manejo del claustro universitario en la UCLV presenta limitaciones en las que podemos señalar las siguientes:

- No presenta una relación entre las tablas que compone el sistema para el manejo de la información de los empleados como se muestra en la Figura 2.5
- Presenta limitaciones con la concurrencia debido a que solamente se pueden estar conectados cuatro usuarios de forma simultánea y trae consigo demoras en la entrada y actualizaciones de datos, así como la obtención de informes actualizados.
- Presenta información acerca de los empleados en muchas tablas teniendo información duplicada en el sistema.
- El sistema maneja a los empleados de forma los empleos de forma general y se desea un sistema que maneje la información de docentes, investigadores y profesores a tiempo parcial del claustro de la UCLV.

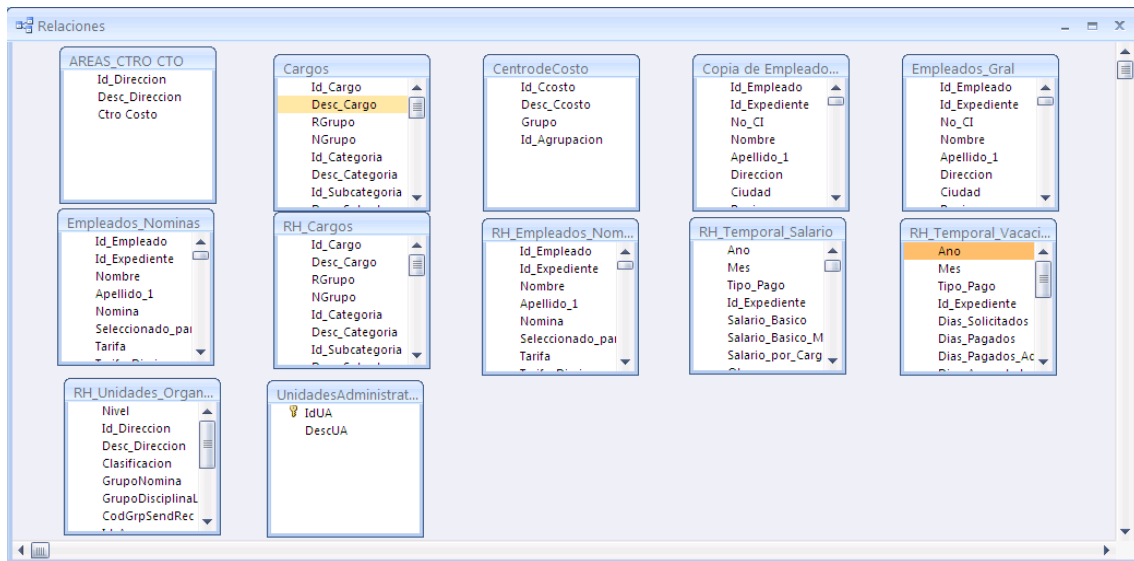


Figura 2. 5 Sistema ASSETS

Consideraciones finales del capítulo

Las Nuevas Tecnologías de la Información y las Comunicaciones y, más puntualmente, sistemas para el manejo de información, juegan un importante papel en la sociedad actual. Con su aparición, la información adquirió el nuevo valor de recurso indispensable para cualquier entidad.

Para ello se ha realizado un profundo estudio teórico, en el que se describieron los sistemas computarizados para la gestión de la información existentes en la Dirección de Recursos Humanos en la UCLV, estos se limitaban al uso de Microsoft Access. Además, se determinaron las principales necesidades relacionadas con la gestión de la información. Estas consistían en el diseño de una base de datos que permitiera al técnico de recursos humanos el manejo de la información del claustro universitario comprendido para docentes, investigadores y profesores a tiempo parcial; necesarios para un adecuado desempeño de la actividad laboral de la entidad.

En función de la implementación de una aplicación web que respondiera a las necesidades de gestión de la información de la empresa se determinaron las herramientas a utilizar para el logro de los objetivos propuestos. Entre estas se pueden citar: PHP como lenguaje de programación, UML como lenguaje de modelación del sistema y como motor de base de datos PostgreSQL.

Capítulo 3: Descripción del ambiente del sistema

El sistema permite mediante una interfaz web amigable poder visualizar la información del claustro universitario en la UCLV en el Departamento de Recursos Humanos. Todo esto con el objetivo de que los especialistas de esta área encuentren un lugar en la red donde comunicarse y gestionar la información que se maneja. Al sentarse en la máquina el usuario podrá acceder a este sistema mediante la web. En los posteriores epígrafes se hace un resumen visual de la navegación por el sitio y su forma de actualización.

3.1 Actualización del sistema

Para la actualización del sistema Claustro UCLV se hará uso de la herramienta Pentaho Data Integration (Kettle) que posibilita la actualización a partir de un backup del sistema ASSETS en que se encuentra sobre Microsoft Access.

3.1.1 Configuración de variables de entorno

Para la configuración de las variables de entorno se debe cumplir el siguiente requisito:

- Tener instalada la máquina virtual de JAVA

En la Figura 3.1 y en la Figura 3.2 se muestra la configuración de las variables de entorno para la actualización del sistema con el jdk de la biblioteca virtual 1.6.0_03.

La configuración de la variable JAVA_HOME se describe a continuación:

- Nombre de la variable: JAVA_HOME
- Valor de la variable: C:\Archivos de programa\Java\jdk1.6.0_03

La configuración de la variable path se describe a continuación:

- Nombre de la variable: path
- Valor de la variable: C:\Archivos de programa\Java\jdk1.6.0_03\bin

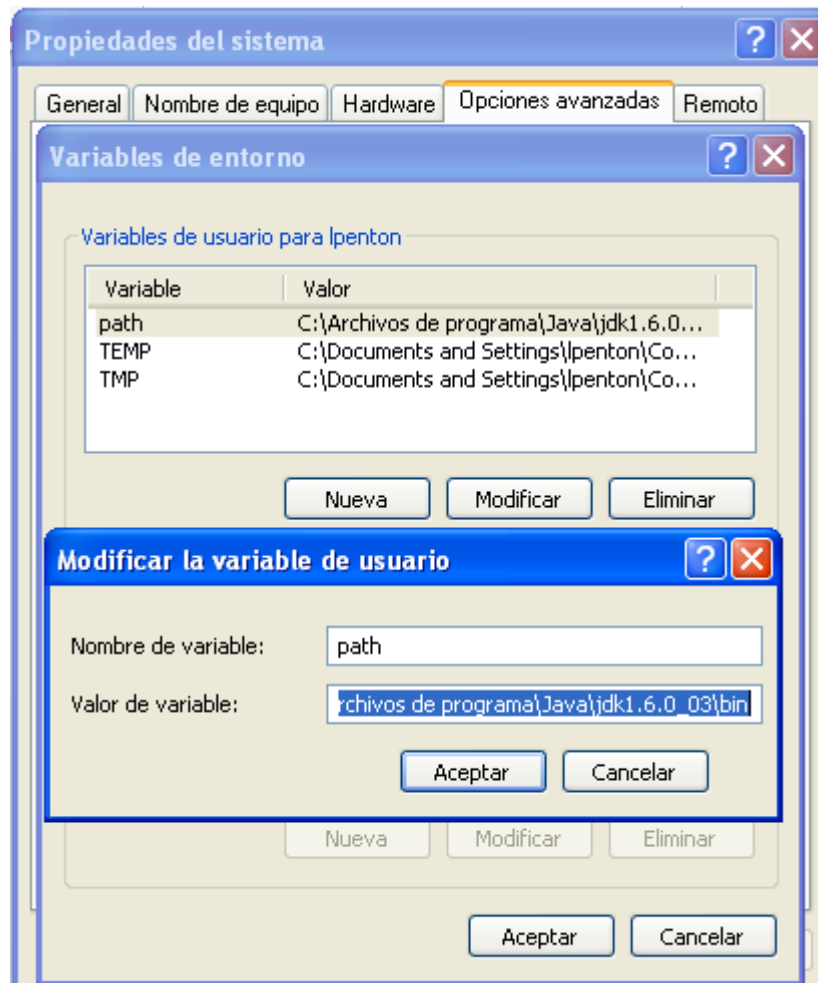


Figura 3.1 Configuración de la variable path

En la Figura 3.1 se muestra dentro de las propiedades del sistema la vía de creación de una variable de entorno para el trabajo con la biblioteca virtual de JAVA en este caso la creación de la variable *path*.

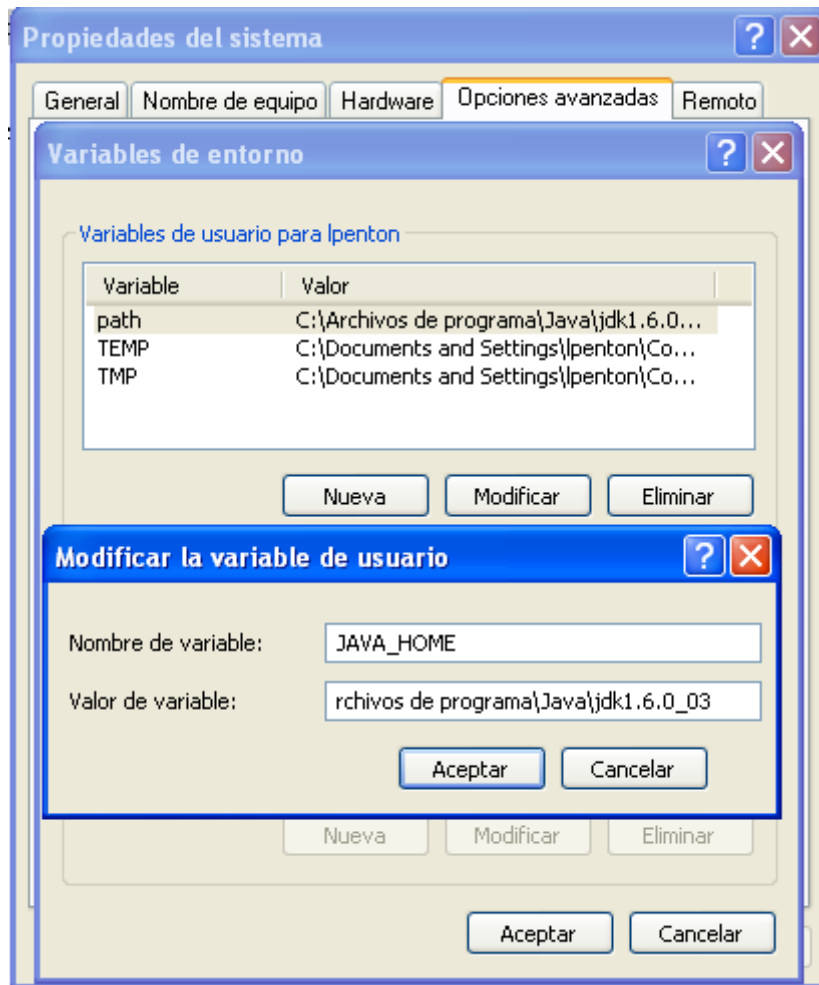


Figura 3. 2 Configuración de la variable JAVA_HOME

En la Figura 3.2 se muestra dentro de las propiedades del sistema la vía de creación de una variable de entorno para el trabajo con la biblioteca virtual de JAVA en este caso la creación de la variable *path*.

3.1.2 Carga y ejecución de las transformaciones mediante el Spoon

En la Figura 3.3 se muestra la pantalla principal de la herramienta con la cual se realizara la actualización del sistema en el cual se cargaran las transformaciones ya previamente definidas.

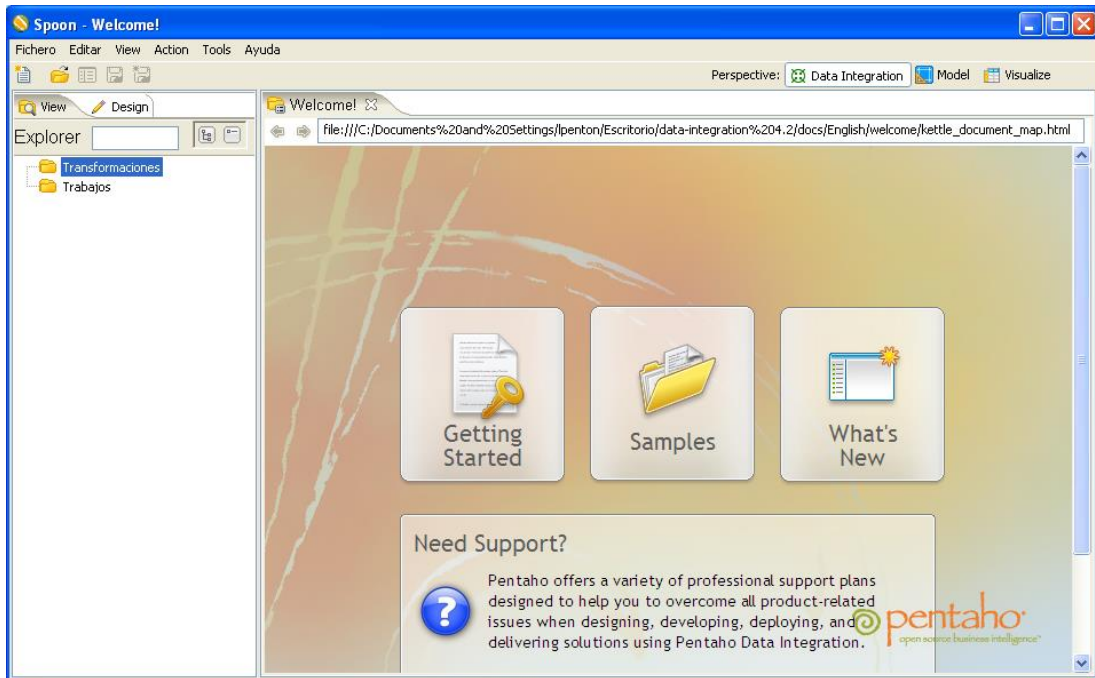


Figura 3. 3 Herramienta Spoon

En la Figura 3.4 se muestra la carga de la transformación centro de costo para la actualización del sistema.

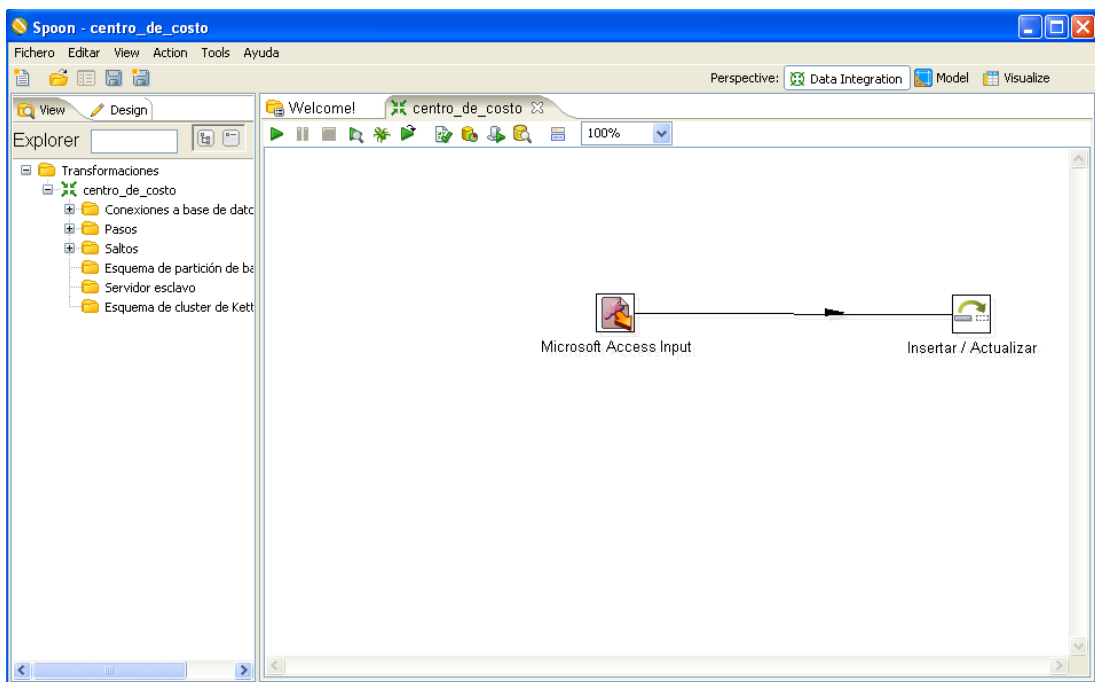


Figura 3. 4 Carga de la transformación centro de costo

En la Figura 3.5 se muestra la forma de editar el camino donde se encuentra el backup del sistema ASSETS para la realizar la actualización.

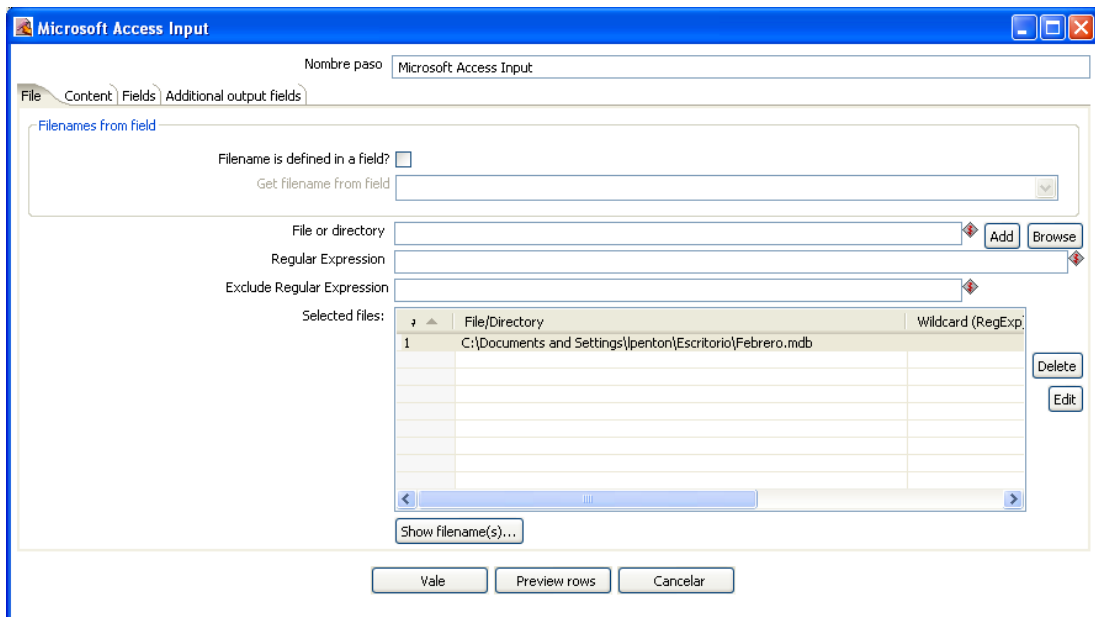


Figura 3. 5 Editar camino de backup

En la Figura 3.6 se muestra la corrida de la trasformación centro de costo y la correcta actualización de la tabla centro de costo en el sistema.

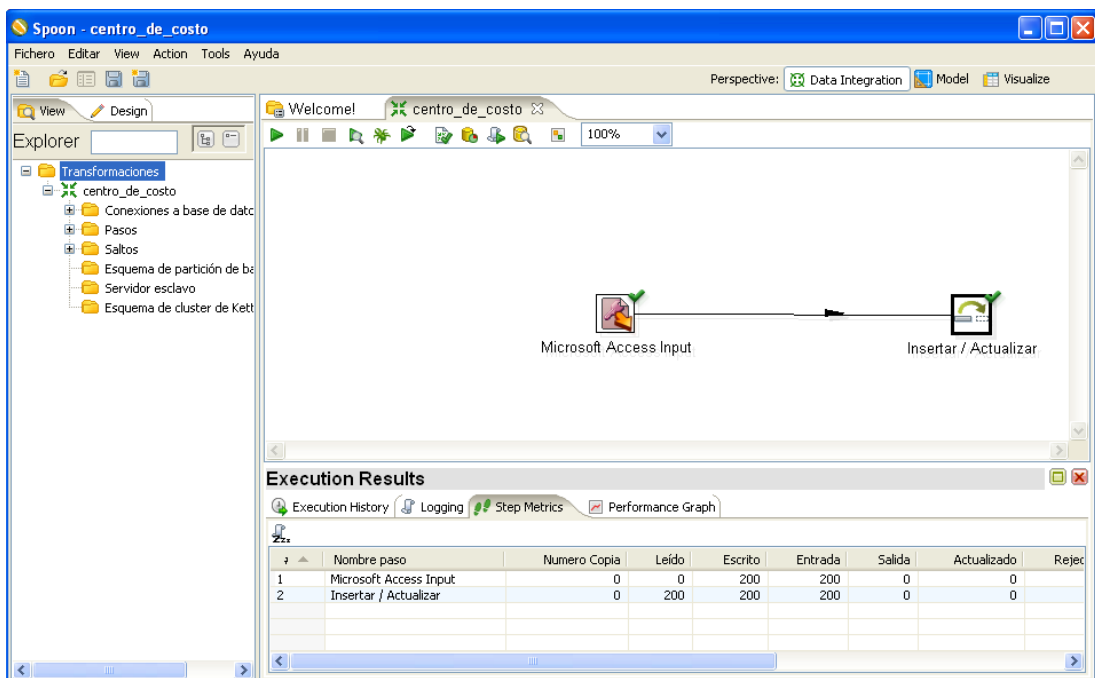


Figura 3. 6 Actualización de centro de costo

Como se muestra anteriormente se cargaran cada una de las trasformaciones especificando donde se encuentra el archivo de actualización; posteriormente se ejecutaran las trasformaciones y se tendrá el sistema Claustro UCLV totalmente actualizado.

3.2 Ambiente de usuario del Sistema

El sistema dispone de un actor para el cual hay una única vía para la navegación del sitio. En la Figura 3.7 se muestra donde aparecen los elementos fundamentales de la portada.



Figura 3. 7 Portada del sistema

En la Figura 3.8 se muestra la sección de autenticación de usuario la cual posibilita el acceso al sistema.

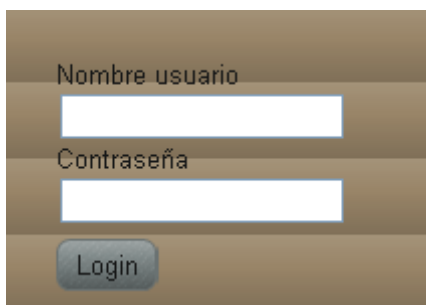


Figura 3. 8 Autenticación de usuario

Capítulo 3

En la Figura 3.9 se muestra la ventana principal del sistema que cuenta con un conjunto de menús donde en técnico de RH podrá visualizar la información del claustro universitario.

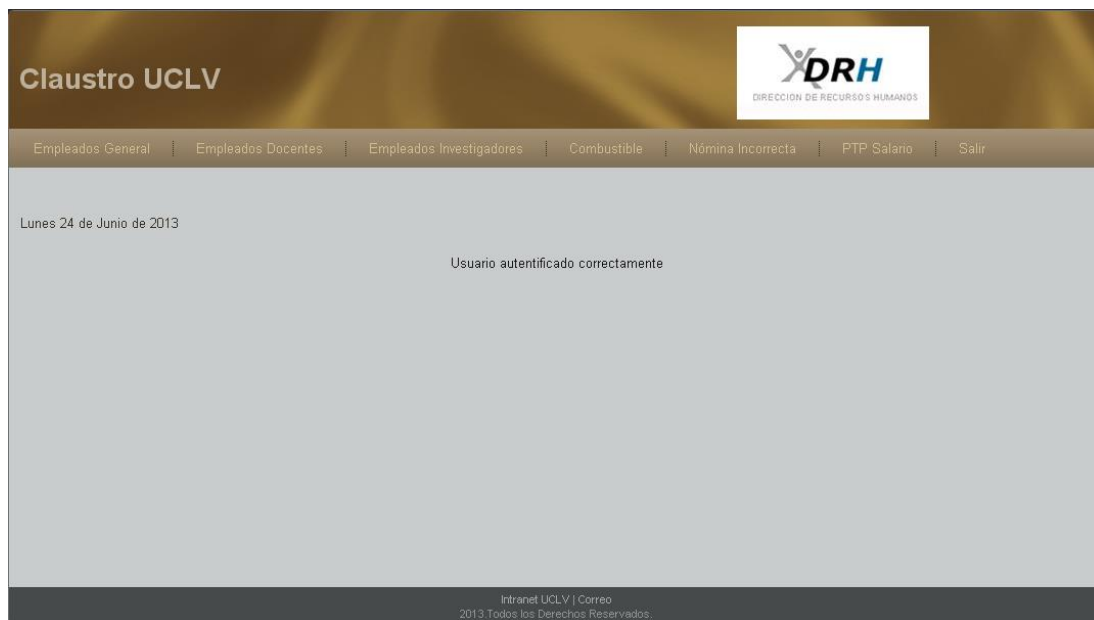


Figura 3. 9 Ventana principal del sistema.

En la Figura 3.10 se muestra el despliegue del menú empleado general donde están clasificados por sexo y dentro de ello por el color de la piel.

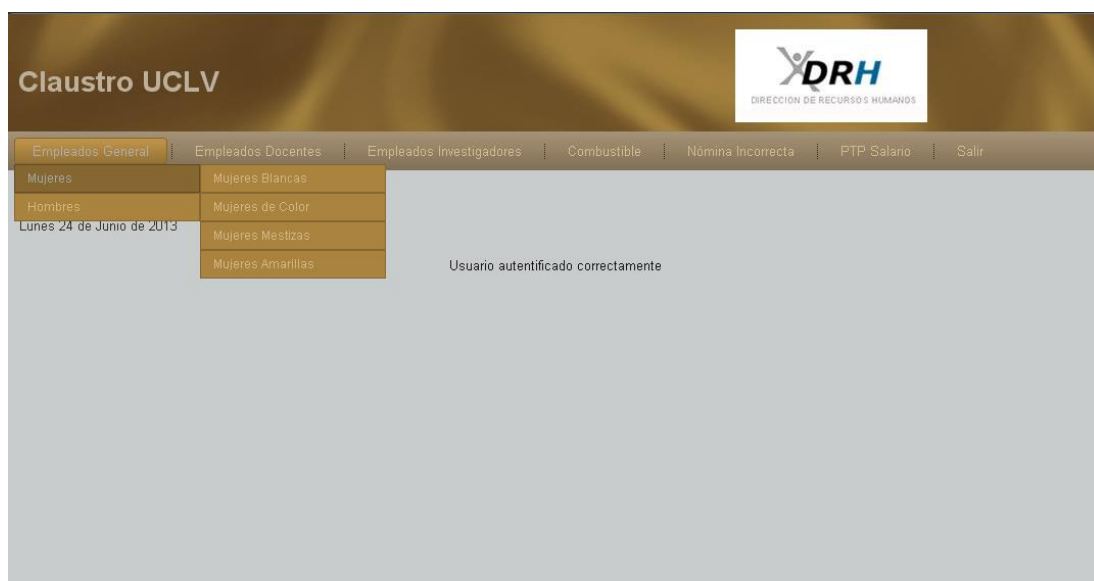
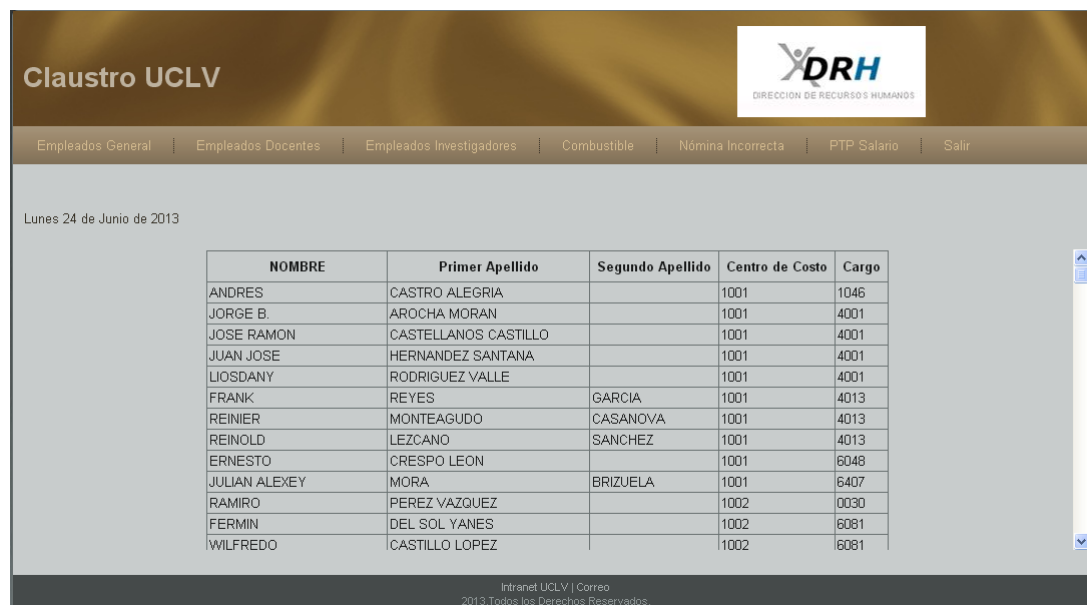


Figura 3. 10 Menú empleado general

En la Figura 3.11 se muestra una relación de los empleados hombres de acuerdo a su centro de costo y cargo.



NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
ANDRES	CASTRO ALEGRIA		1001	1046
JORGE B.	AROCHA MORAN		1001	4001
JOSE RAMON	CASTELLANOS CASTILLO		1001	4001
JUAN JOSE	HERNANDEZ SANTANA		1001	4001
LIOSDANY	RODRIGUEZ VALLE		1001	4001
FRANK	REYES	GARCIA	1001	4013
REINIER	MONTEAGUDO	CASANOVA	1001	4013
REINOLD	LEZCANO	SANCHEZ	1001	4013
ERNESTO	CRESPO LEON		1001	6048
JULIAN ALEXEY	MORA	BRIZUELA	1001	6407
RAMIRO	PEREZ VAZQUEZ		1002	0030
FERMIN	DEL SOL YANES		1002	6081
WILFREDO	CASTILLO LOPEZ		1002	6081

Figura 3. 11 Empleados hombres

En la Figura 3.12 se muestra una relación de los empleados hombres mestizos de acuerdo a su centro de costo y cargo.



NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
JEISIL	AGUILAR SANTOS		2003	4002
DANIEL	GUEMECHU	ALFONSO	2003	4004
MAIDIEL	OCAÑA	MORENO	2005	6067
ALEXIS	GUTIERREZ FRANCISCO		2007	0002
PABLO JOSE	PRIETO ENTENZA		2008	4002
ORESTES RAFAEL	LEON	CUELLAR	2014	6804
YURIEL	LEON	SILVERIO	2021	4004
ARCADIO	ESCUDELOS	GUTIERREZ	2021	4004
ANDREY	VINAJERA ZAMORA		2033	4002
MICHAERLYS	MARRERO	OVIEDO	2033	4002
HUMBERTO NICOLAS	MORA VILLEGAS		2035	4008
MARIO PEDRO	GONZALEZ	RODRIGUEZ	4005	9042
NORBERTO ANDRES	VARGAS	ARTEAGA	4005	9042

Figura 3. 12 Empleados hombres mestizos

En la Figura 3.13 se muestra el despliegue del menú empleados docentes donde están clasificados por sexo y dentro de esto por su categoría.

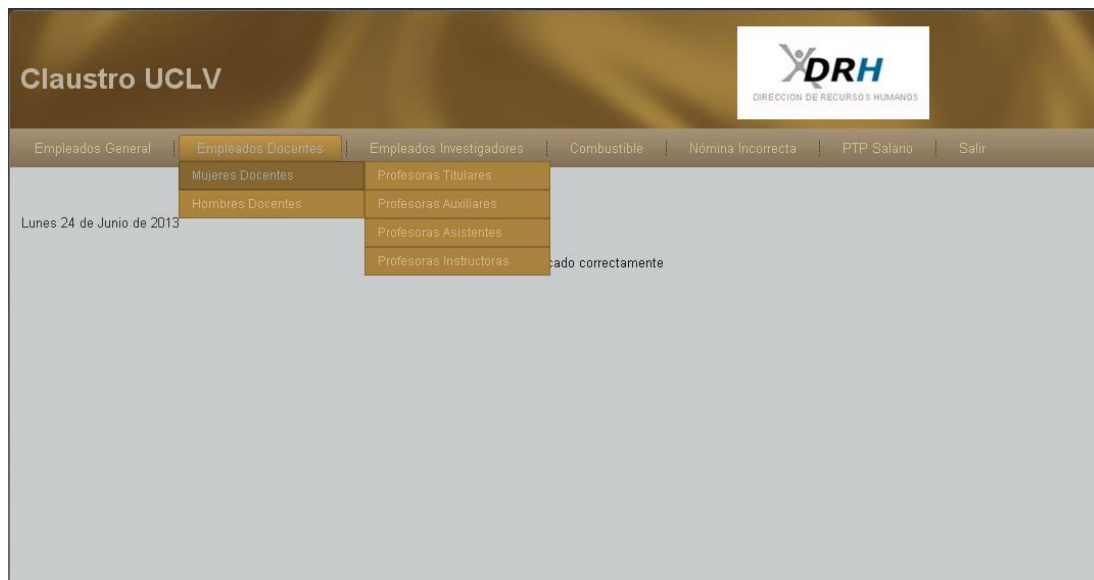


Figura 3. 13 Menú empleados docentes

En la Figura 3.14 se muestra una relación de empleados docentes del sexo masculino con categoría de profesores instructores.

Claustro UCLV				
DIRECCION DE RECURSOS HUMANOS				
Empleados General	Empleados Docentes	Empleados Investigadores	Combustible	Nómina Incorrecta
PTP Salario	Salir			
Lunes 24 de Junio de 2013				
NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
YULIER	CAMPOS	PEREZ	2002	1051
DANIEL	GUEMECHU	ALFONSO	2003	4004
OSMEL	PEREZ	RODRIGUEZ	2004	4004
RAUL	RODRIGUEZ	DEL SOL	2004	4004
RENIEL	ESTRADA YANES		2005	4004
JAVIER ENRIQUE	CABEZA	FERREIRA	2005	4004
VITALIY DANILO	SUAREZ CHERNOV		2005	4004
YANY	PORTAL	CARVAJAL	2007	4004
ALBERTO	DIAZ BARATA		2007	4004
GONZALO	GONZALEZ	CAMACHO	2007	4004
RUBEN EDUARDO	CARLES BARRERO		2008	4004
DELVIS	GARCIA	GARCIA	2008	4004
MARIO ALBERTO	GONZALEZ CARTAS		2008	4004

Figura 3. 14 Empleados docentes hombres con categoría de instructores

En la Figura 3.15 se muestra una relación de las docentes mujeres que poseen el cargo de profesoras auxiliares.



Claustro UCLV

Empleados General | Empleados Docentes | Empleados Investigadores | Combustible | Nómina Incorrecta | PTP Salario | Salir

Lunes 24 de Junio de 2013

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
EBIR	GONZALEZ CRUZ		1006	4006
YANELYS	DELGADO TRIANA		2002	4008
MARITZA VICTORIA	MARTINEZ LIMA		2003	4008
MERCEDES	RIVAS PEREZ		2003	4008
GERTRUDIS L	TOLEDO CABRERA		2003	4008
ANAYANSI	CASTELLON JIMENEZ		2003	4008
LENA ESTHER	HERNANDEZ HERNANDEZ		2003	4008
IVIS	GONZALEZ MORENO		2003	4008
MAYLEN	VILLAMAÑAN ALBA		2003	4008
LLANELYS	PERALTA CASTELLON		2003	4008
IRIS	LAUREIRO RAMIREZ		2003	1051
MARIA A	CASTILLO DAVILA		2003	4008
GEORGINA	CASTRO ACEVEDO		2003	4008

Intranet UCLV | Correo
2013 Todos los Derechos Reservados.

Figura 3. 15 Docentes mujeres profesoras auxiliares

En la Figura 3.16 se muestra el despliegue del menú empleados investigadores donde están clasificados por sexo y dentro de esto por su categoría.

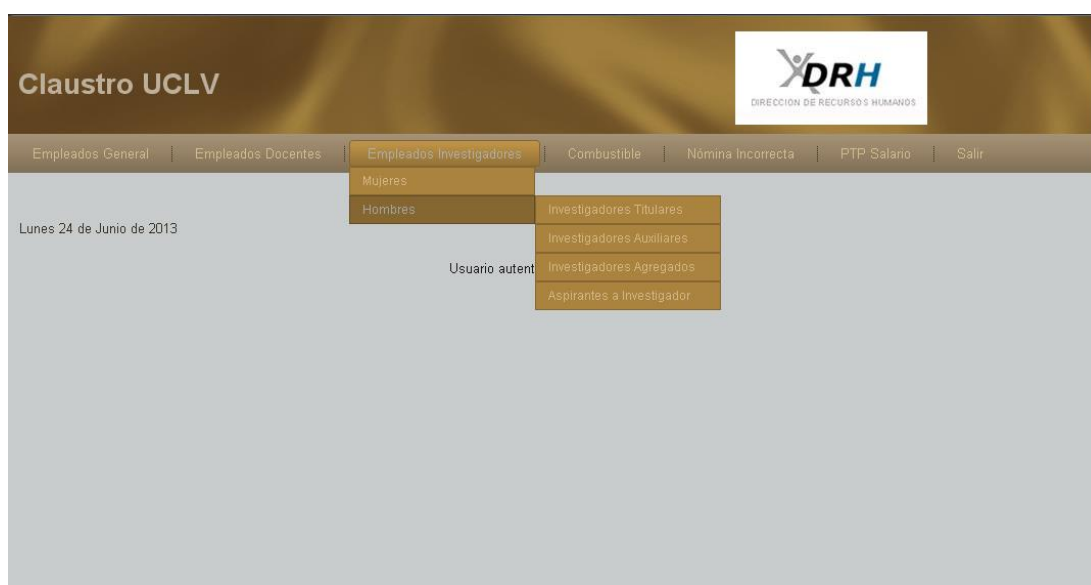


Figura 3. 16 Menú empleados investigadores

En la Figura 3.17 se muestra una relación de los empleados investigadores los cuales son mujeres.

Claustro UCLV

DRH DIRECCION DE RECURSOS HUMANOS

Empleados General | Empleados Docentes | Empleados Investigadores | Combustible | Nómina Incorrecta | PTP Salario | Salir

Lunes 24 de Junio de 2013

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
IDALIA	AGUILA HERNANDEZ		2034	5002
YAILET	ALBERNAS CARVAJAL		2034	5001
LEYANIS	MESA GARRIGA		2034	5002
MEILYN	GONZALEZ CORTES		2034	5002
ALIANNY	RODRIGUEZ URRUTIA		3020	5001
LIZET	RODRIGUEZ MACHIN		3023	5002
PASTORA DE LA C	MARTINEZ NODAL		3026	5002
LIZ MABEL	RIOS HIDALGO		3026	5002
MARGIE	ZORRILLA VELAZCO		3026	5002
MARIA MAIRA	PEREZ VILLAR		3026	5002
ISMAY	PEREZ SANCHEZ		3027	5001
BELKIS	SAROZA HORTA		3027	5002
IDISLEISY	TORRES	RODRIGUEZ	3030	5006

Intranet UCLV | Correo
2013 Todos los Derechos Reservados

Figura 3. 17 Mujeres investigadoras

En la Figura 3.18 se muestra una relación de los empleados investigadores los cuales son mujeres.

Claustro UCLV

DRH DIRECCION DE RECURSOS HUMANOS

Empleados General | Empleados Docentes | Empleados Investigadores | Combustible | Nómina Incorrecta | PTP Salario | Salir

Lunes 24 de Junio de 2013

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
YAMEL	JIMENEZ PRIETO		2034	5001
EDILBERTO	POZO VELAZQUEZ		3020	5002
VICENTE HORACIO	GRILLO RABELO		3020	5003
EDILIO	QUINTERO FERNANDEZ		3020	5003
LEANDRO	MARRERO SUAREZ		3020	5004
ROBERTO	VALDEZ HERRERA		3020	5002
VICTOR DANIEL	GIL DIAZ		3020	5003
RAUL	OLALDE FONT		3020	5004
ORLANDO M	SAUCEDO CASTILLO		3020	5004
PEDRO	CAIRO CAIRO		3020	5004
RAFAEL	QUINTANA PUCHOL		3021	5004
LORENZO	PERDOMO GONZALEZ		3021	5004
DANEL	SANCHEZ TARRAGO		3022	5001

Intranet UCLV | Correo
2013 Todos los Derechos Reservados

Figura 3. 18 Hombres Investigadores

En la Figura 3.19 se muestra una relación de las mujeres que son investigadoras que presentan el cargo de investigadoras agregadas.

Claustro UCLV				
				
Empleados General Empleados Docentes Empleados Investigadores Combustible Nómina Incorrecta PTP Salario Salir				
Lunes 24 de Junio de 2013				
NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
IDALIA	AGUILA HERNANDEZ		2034	5002
MEILYN	GONZALEZ CORTES		2034	5002
LEYANIS	MESA GARRIGA		2034	5002
LIZET	RODRIGUEZ MACHIN		3023	5002
PASTORA DE LA C	MARTINEZ NODAL		3026	5002
LIZ MABEL	RIOS HIDALGO		3026	5002
MARGIE	ZORRILLA VELAZCO		3026	5002
MARIA MAIRA	PEREZ VILLAR		3026	5002
BELKIS	SAROZA HORTA		3027	5002
YUSELY	RUIZ GONZALEZ		3030	5002
ALINA	MONTERO TORRES		4010	1019

Figura 3. 19 Mujeres investigadoras agregadas

En la Figura 3.20 se muestra una relación de las mujeres que son investigadoras que presentan el cargo de aspirante a investigador.



Claustro UCLV				
				
Empleados General Empleados Docentes Empleados Investigadores Combustible Nómina Incorrecta PTP Salario Salir				
Lunes 24 de Junio de 2013				
NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
YAILET	ALBERNAS CARVAJAL		2034	5001
ALIANNY	RODRIGUEZ URRUTIA		3020	5001
ISMAY	PEREZ SANCHEZ		3027	5001

Figura 3. 20 Mujeres investigadoras con cargo aspirante a investigador

En la Figura 3.21 se muestra una relación de los investigadores hombres con el cargo de investigadores titulares.

Claustro UCLV



DRH

DIRECCIÓN DE RECURSOS HUMANOS

Empleados General

Empleados Docentes

Empleados Investigadores

Combustible

Nómina Incorrecta

PTP Salario

Salir

Lunes 24 de Junio de 2013

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
ORLANDO M	SAUCEDO CASTILLO		3020	5004
LEANDRO	MARRERO SUAREZ		3020	5004
PEDRO	CAIRO CAIRO		3020	5004
RAUL	OLALDE FONT		3020	5004
RAFAEL	QUINTANA PUCHOL		3021	5004
LORENZO	PERDOMO GONZALEZ		3021	5004


Intranet UCLV | Correo

2013.Todos los Derechos Reservados.

Figura 3. 21 Investigadores hombres titulares

En la Figura 3.22 se muestra una relación de los investigadores hombres con el cargo de investigadores auxiliares.

Claustro UCLV



DRH

DIRECCION DE RECURSOS HUMANOS

Empleados General

Empleados Docentes

Empleados Investigadores

Combustible

Nómina Incorrecta

PTP Salario

Salir

Lunes 24 de Junio de 2013

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
VICTOR DANIEL	GIL DIAZ		3020	5003
VICENTE HORACIO	GRILLO RABELO		3020	5003
EDILIO	QUINTERO FERNANDEZ		3020	5003
JOSE F.	MARTIRENA HERNANDEZ		3027	1003
CARLOS	RECAREY MORFA		3027	5003

Intranet UCLV | Correo

2013 Todos los Derechos Reservados.

Figura 3. 22 Investigadores hombres auxiliares

En la Figura 3.23 se muestra una relación de los investigadores hombres con el cargo de investigadores agregados.

Claustro UCLV

Empleados General

Empleados Docentes

Empleados Investigadores

Combustible

Nómina Incorrecta

PTP Salario

Salir

Lunes 24 de Junio de 2013

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
EDILBERTO	POZO VELAZQUEZ		3020	5002
ROBERTO	VALDEZ HERRERA		3020	5002
ELADIO	CUELLAR VEGA		3022	5002
OSCAR MIGUEL	RIVERA BORROTO		3022	5002
ERNESTO Y.	FARINAS WONG		3023	5002

Intranet UCLV | Correo

2013 Todos los Derechos Reservados

Figura 3. 23 Investigadores hombres agregados

En la Figura 3.24 se muestra una relación de los investigadores hombres con el cargo de aspirante a investigador.

Claustro UCLV



DIRECCION DE RECURSOS HUMANOS

Empleados General

Empleados Docentes

Empleados Investigadores

Combustible

Nómina Incorrecta

PTP Salario

Salir

Lunes 24 de Junio de 2013

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo
YAMEL	JIMENEZ PRIETO		2034	5001
DANEL	SANCHEZ TARRAGO		3022	5001
IRVIN PABLO	PEREZ MORALES		3027	5001
ROBERTO LUIS	ROSELLO VALERA		3027	5001
HEIKEL	YERVILLA HERRERA		3027	5001
YAIDEL	REYES LOPEZ		3027	5001

Intranet UCLV | Correo

2013 Todos los Derechos Reservados.

Figura 3. 24 Investigadores hombres aspirantes a investigador

En la Figura 3.25 se realiza el despliegue del menú Combustible el cual posibilita tener una relación de los empleados que reciben combustible de acuerdo al sexo y el tipo de combustible asignado.

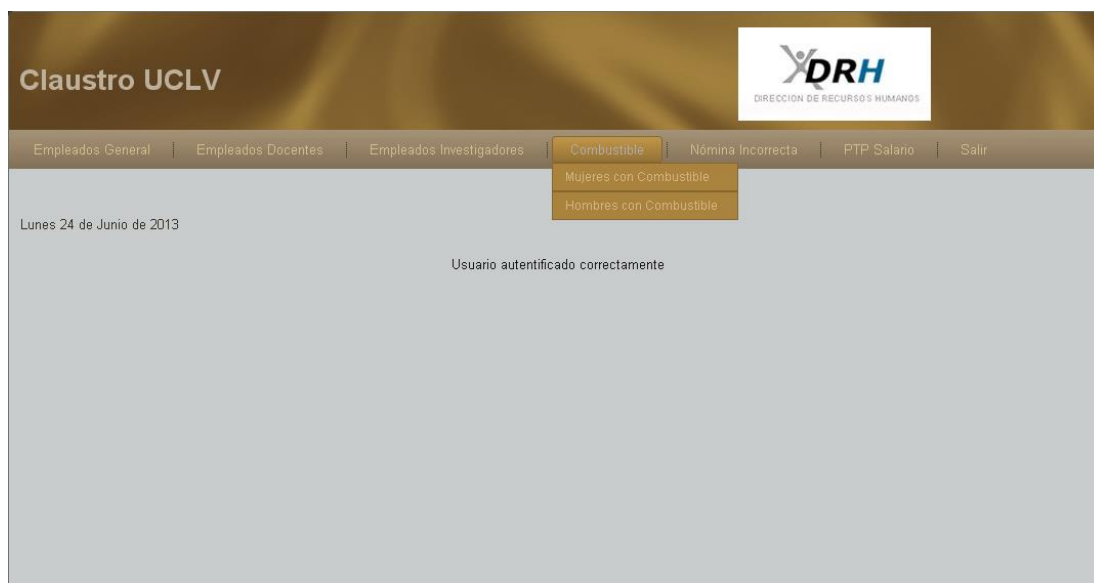


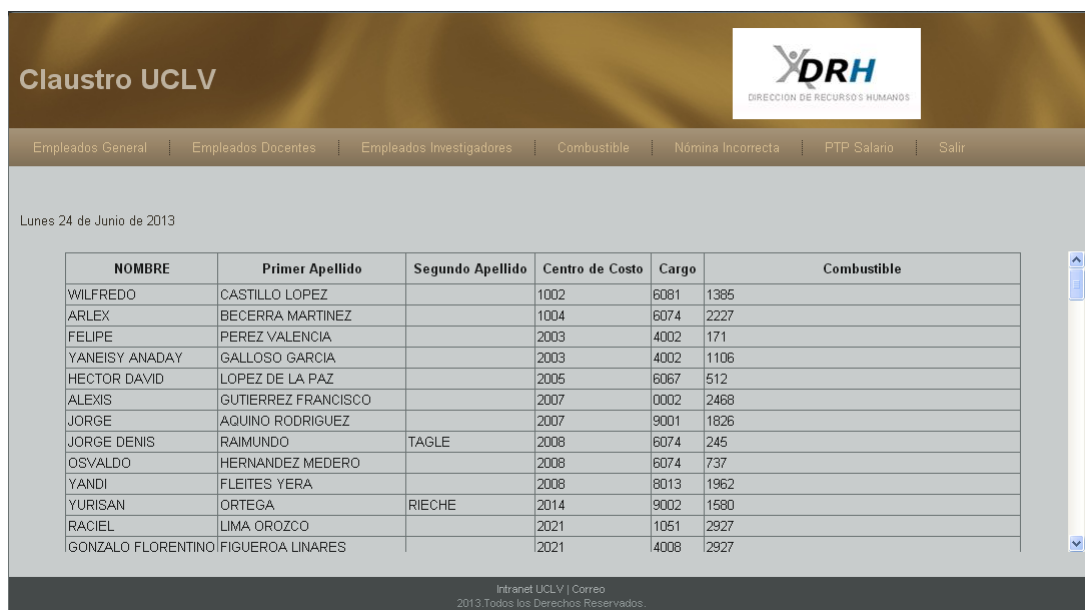
Figura 3. 25 Menú combustible

En la Figura 3.26 se muestra una relación de empleados mujeres que reciben combustible y el tipo de combustible asignado.

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo	Combustible
ANA ROSA	AMADOR RODRÍGUEZ		1001	6900	2693
YAIMA	ALONSO DIAZ		1002	6081	2088
YANISLEIDY	OROZCO ROQUE		1006	6031	2799
LISBETH	RUIZ HERNANDEZ		2002	6011	1333
ARIANNY	NAPOLES CORDOBA		2002	6011	2556
GLADYS	RAMIREZ RAMOS		2002	6061	748
GEIDY	ALEA HERNANDEZ		2002	6062	37
DIOSMARA	FROMETA	ALAYO	2002	9002	191
MATILDE	RODRIGUEZ BECERRA		2003	9002	986
COSETTE	RODRIGUEZ AGUILA		2004	6062	646
NELVI	GARCES GONZALEZ		2004	6067	3798
ROSA MARIA	MENDEZ CAMPOS		2004	9002	1214
MIRIAM MABEL	FLEITES DITA		2004	9002	2556

Figura 3. 26 Relación de mujeres con combustible

En la Figura 3.27 se muestra una relación de empleados hombres que reciben combustible y el tipo de combustible asignado.



Claustro UCLV

Empleados General | Empleados Docentes | Empleados Investigadores | Combustible | Nómina Incorrecta | PTP Salario | Salir

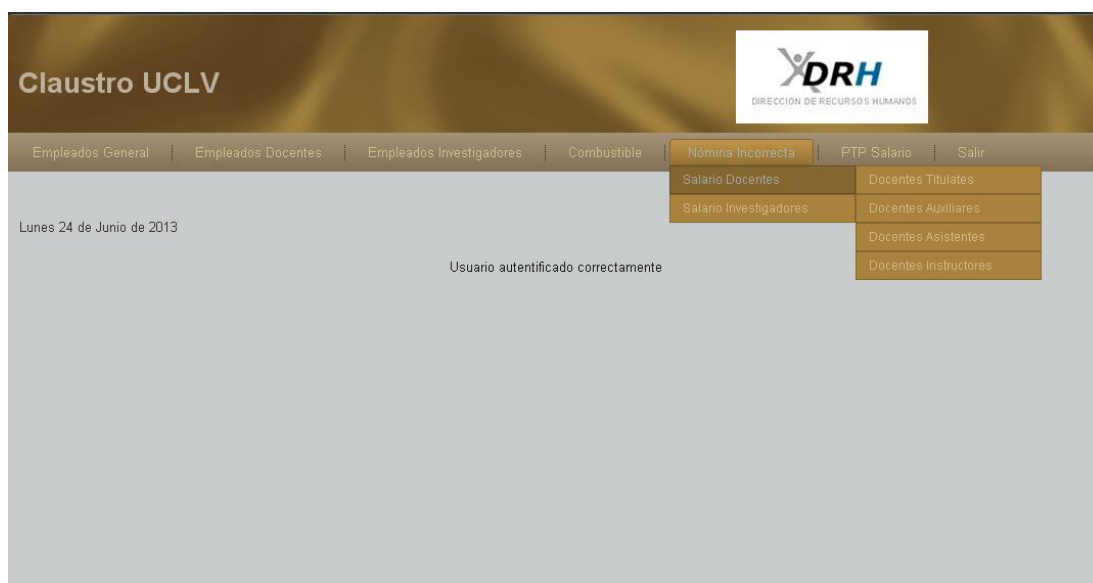
Lunes 24 de Junio de 2013

NOMBRE	Primer Apellido	Segundo Apellido	Centro de Costo	Cargo	Combustible
WILFREDO	CASTILLO LOPEZ		1002	6081	1385
ARLEX	BECERRA MARTINEZ		1004	6074	2227
FELIPE	PEREZ VALENCIA		2003	4002	171
YANEISY ANADAY	GALLOSO GARCIA		2003	4002	1106
HECTOR DAVID	LOPEZ DE LA PAZ		2005	6067	512
ALEXIS	GUTIERREZ FRANCISCO		2007	0002	2468
JORGE	AQUINO RODRIGUEZ		2007	9001	1826
JORGE DENIS	RAIMUNDO	TAGLE	2008	6074	245
OSVALDO	HERNANDEZ MEDERO		2008	6074	737
YANDI	FLEITES YERA		2008	8013	1962
YURISAN	ORTEGA	RIECHE	2014	9002	1580
RACIEL	LIMA OROZCO		2021	1051	2927
GONZALO FLORENTINO	FIGUEROA LINARES		2021	4008	2927

Intranet UCLV | Correo
2013 Todos los Derechos Reservados.

Figura 3. 27 Relación de hombres con combustible

En la Figura 3.28 se muestra el despliegue del menú Nómina Incorrecta el cual contiene un submenú para el salario de los docentes agrupados por categoría docente el cual posibilita obtener una relación de empleados que su salario básico sea incorrecto.



Claustro UCLV

Empleados General | Empleados Docentes | Empleados Investigadores | Combustible | Nómina Incorrecta | PTP Salario | Salir

Lunes 24 de Junio de 2013

Usuario autenticado correctamente

Nómina Incorrecta	
Salario Docentes	Docentes Titulares
Salario Investigadores	Docentes Auxiliares
	Docentes Asistentes
	Docentes Instructores

Figura 3. 28 Menú nomina incorrecta con submenú salario docentes

En la Figura 3.29 se muestra el despliegue del menú Nómina Incorrecta el cual contiene un submenú para el salario de los docentes agrupados por categoría docente el cual posibilita obtener una relación de empleados que su salario básico sea incorrecto.

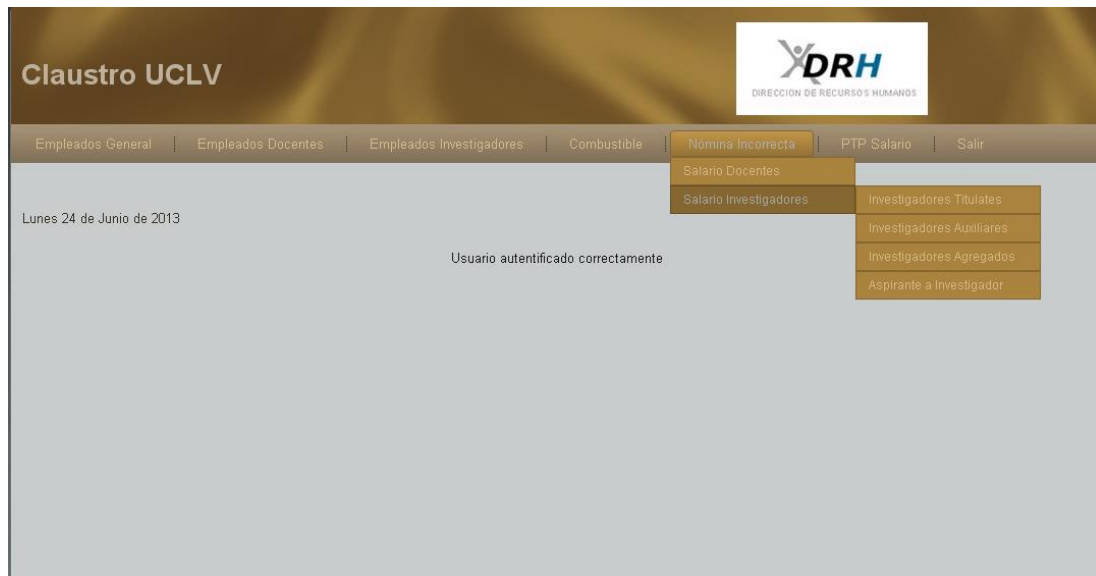


Figura 3. 29 Menú nomina incorrecta con submenú salario investigadores

En la Figura 3.30 se muestra el despliegue del menú PTP Salario el cual posibilita que se tenga una relación del salario básico de los profesores a tiempo parcial y así se evita un salario incorrecto para el empleado.

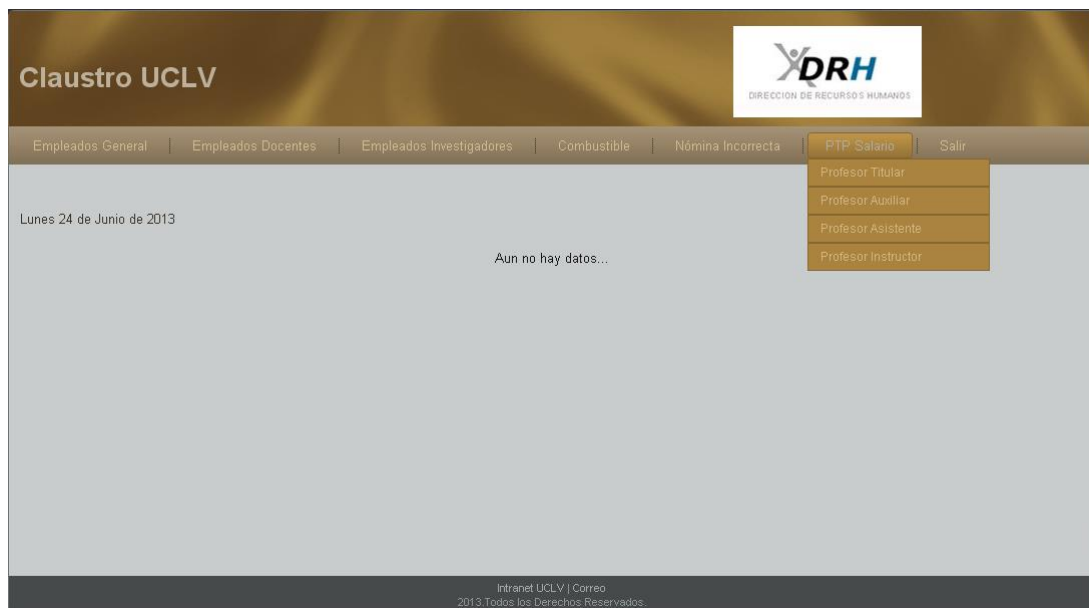


Figura 3. 30 Menú PTP salario

Consideraciones finales del capítulo

En este capítulo se exponen las potencialidades que presentan las herramientas que han sido seleccionadas para realizar la aplicación, además del diseño del sistema Claustro UCLV que permite hacer una visualización de la información referente al mismo.

Capítulo 3

Este sitio Web cuenta con un manual de usuario en el que se explican los principales servicios a los que pueden acceder los técnicos del Departamento de Recursos Humanos. Con ello se facilitan la navegación y el mantenimiento del mismo.

Conclusiones

En este trabajo se logró implementar un entorno Web, que permite acceder de forma rápida y sencilla para la visualización de la información del claustro universitario en la UCLV comprendido para docentes, investigadores y profesores a tiempo parcial, poniendo a la disposición del Departamento de Recursos Humanos el sistema implementado, garantizando la seguridad e integridad de los datos manejados en el sitio. Para lograrlo se dio cumplimiento a los objetivos del mismo, pues se

- Desarrolló la modelación del sistema a implementar.
- Diseñó la BD que contiene la información del claustro de la UCLV.
- Implementó un sistema que permite la visualización de la información del claustro universitario en la UCLV.

Recomendaciones

- Añadir opciones de filtrado y ordenamiento de información al Sistema Claustro UCLV.
- Mejorar el diseño de la interfaz de usuario según las recomendaciones de los usuarios luego de su uso inicial.
- Incluir la posibilidad de actualización automática al Sistema Claustro UCLV.

Referencias Bibliográficas

Referencias Bibliográficas

Aliaga Ibarra, A. and M. A. Miani Flores (2008) PostgreSQL.

Apaza, Á., C. Méndez, et al. (2010). Administración de Recursos Humanos I.

BEÁTI, H. (2012). El Gran Libro de PHP: Creación de páginas Web dinámicas Marcombo.

Casters, M., R. Bouman, et al. (2010). Building Open Source ETL Solutions with Pentaho Data Integration. Indianapolis, Indiana, Wiley Publishing, Inc.

Cernuda, O. (2001). Kitópolis abre la primera campaña por el uso del software libre en la Administración.

Gallego Vásquez, J. A. (2003). Desarrollo Web con PHP y MySQL. J. I. Luca de Tena. España, ANAYA MULTIMEDIA (GRUPO ANAYA) S. A.

González Sabín, R. (2005). Nuevas tecnologías aplicadas a la gestión de RRHH. Las TIC'S como herramienta permanente del capital humano. España, IdeasPropias Editorial.

Lopez, J. (2008). Domine PHP y MySQL - Programacion Dinamica en el Lado del Servidor Alfaomega - Ra-ma

Maliaños, L. (2006) Resumen de UML

Martin, A. (2011). PostgreSQL: Una poderosa base de datos libre E. A. Española.

Morales Guedes, A. (2012). Política de Desarrollo de Software y Sistemas Informáticos. Villa Clara.

Rumbaugh, J., G. Booch, et al. (1997). Lenguaje de modelado de sistemas de software.

Schmuller, J. (2000). Aprendiendo UML en 24 Horas. O. MADRIGAL MUNIZ. Mexico, Pearson Educación.

Referencias Bibliográficas

Simon, B., M. Steve, et al. (2010). Análisis y diseño en sistemas orientados a objetos con UML. M.-H. I. d. E. S.L.

Welling, L. and L. Thomson (2004). Programación Web con PHP y MySQL.