

UNIVERSIDAD CENTRAL "MARTA ABREU" DE LAS VILLAS
FACULTAD DE MATEMATICA, FISICA Y COMPUTACION
LICENCIATURA EN CIENCIAS DE LA COMPUTACION



TRABAJO DE DIPLOMA

Aplicaciones y servicios para la Web
de la Dirección de Economía

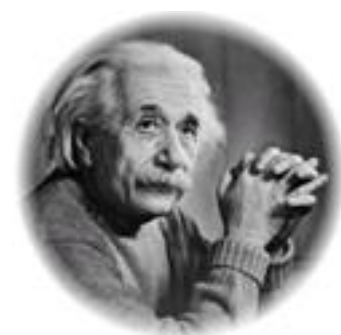


Autor: Yuniesky Alemán Caballero

Tutora: MSc. Carmen Elena Ruiz Espino



Curso 2004-2005



**... Nunca consideres el estudio como una obligación sino como una oportunidad para penetrar
en el bello y maravilloso mundo del saber...**

Albert Einstein

Dedicatoria.

- A mis padres y hermana por el apoyo brindado todo el tiempo.
- A mis abuelos por su preocupación constante.
- A mi novia por su tierna compañía.
- A mi familia en general.
- A la Revolución Cubana.

Agradecimientos.

- Al colectivo de profesores por brindarme todo su conocimiento a lo largo de mi formación.
- A la Dra. María Matilde García y Dr. Ramiro Pérez por su ayuda incondicional.
- A mi tutora MSc. Carmen Elena Ruiz.
- Al MSc. Armando Lauchy.
- A Irene Alfonso Alba por brindarme la información necesaria.
- A mis compañeros de la Dirección de Economía.
- A mi familia por guiarme e impulsarme a lograr esta meta.
- A los que de una u otra forma colaboraron para que este trabajo se realizara satisfactoriamente.

A todos... con sinceridad...

Muchas gracias

Resumen.

El trabajo consiste en el diseño y desarrollo del sitio Web de la Dirección de Economía de la Universidad Central “Marta Abreu” de Las Villas y sus servicios, así como la implementación del sistema para el control del pago del estipendio estudiantil. Se utilizó tecnología Open Source, se programaron módulos que logran vincularse con el Sistema Contable Integral (AssetsNS), obteniendo reportes de interés para los dirigentes de las distintas áreas. Se estudiaron las resoluciones vigentes para establecer el pago del estipendio estudiantil y se desarrolló un servicio que cumple con lo establecido en las mismas. El resultado obtenido permite que la Dirección de Economía brinde algunos servicios a la Intranet UCLV posibilitando una mayor eficiencia en su gestión al disponer de una herramienta que permita publicar información y tener automatizado el control del pago del estipendio estudiantil.

Abstract.

The project consists in design and development of the Web site for Direction of Economy of the Central University "Marta Abreu" of Las Villas and something services, as well as the implementation of the system "Control of the Payment to Students". Was used the Open Souce technology, programmed modules that manage with Integral the Countable System (AssetsNS), obtaining reports of interest for the leaders of the different areas. The effective resolutions studied to establish the payment of students and a service was developed that fulfills the established thing in the same. The obtained result allows that the Direction of Economy offers to some services to Intranet UCLV making possible a greater efficiency in its management when having a tool that allows publishing information automating the control of the payment to students.

Índice.

Introducción:	1
Problema científico:	9
Hipótesis de investigación:	9
Objetivos generales:	9
Objetivos específicos:	9
Valor práctico:	10
Principales aspectos que condujeron a la realización de los servicios:	10
Viabilidad:	11
Capítulo 1: Uso de la tecnología Web para desarrollar aplicaciones y servicios	12
1.1. La tecnología Web.	12
1.1.1. La necesidad y ventajas de la tecnología Web actual.	12
1.2. Avances en la tecnología Web.	13
1.2.1. El Open Source.	14
1.2.2. ¿Qué es el Software libre?	15
1.3. Diseño de sitios Web.	15
1.3.1. Modelado de un sitio Web.	16
1.3.2. Usabilidad y accesibilidad Web.	17
1.4. Programación de sitios Web.	18
1.4.1. El HTML.	18
1.4.1.1. Algunas etiquetas básicas del HTML:	18
1.4.2. Las hojas de estilo CSS.	19
1.4.2.1. Ventajas de utilizar CSS:	19
1.4.3. El JavaScript.	20
1.4.4. El lenguaje PHP.	20
1.4.4.1. ¿Qué es PHP?	21
1.4.4.2. Integración del PHP con HTML.	22
1.5. Uso de Bases de Datos en los sitios Web.	24
1.5.1. El MySQL.	24
1.6. La seguridad en la tecnología Web.	25
1.6.1. Variables globales.	25
1.6.2. Seguridad en la programación.	25
1.6.2.1. Inyección SQL:	26
1.6.3. Seguridad del servidor Web (SSL).	27
1.7. Uso de la Intranet UCLV.	27

1.7.1. Ejemplo de aplicaciones desarrolladas en la UCLV utilizando tecnología Web:	28
Capítulo 2: Diseño y desarrollo de la Web de la Dirección de Economía y sus servicios.....	30
2.1. Introducción.....	30
2.2. <i>Etapas de análisis y diseño</i>	31
2.2.1. <i>Diagrama conceptual del sitio Web</i>	32
2.2.2. Diseño de la base de datos:	33
2.2.3. <i>Boceto estructural de los espacios en el diseño gráfico</i> :.....	33
2.3. <i>Etapas de implementación</i>	34
2.3.1. Diseño y construcción de la Web.....	34
2.3.2. Módulo para la publicación de archivos.	35
Capítulo 3: Diseño y desarrollo del Sistema de Control del Pago del Estipendio Estudiantil.....	39
3.1. Estado actual del proceso de estipendio y servicios Web de la Dirección de Economía.....	39
3.2. Etapas de análisis.	41
3.2.1. Actores del sistema y sus funciones.....	42
3.2.2. Diagramas de Flujo de Datos.	45
3.2.3. Diseño de la base de datos.	46
3.2.3.1. Estructura.....	46
3.2.3.2. Normalización de tablas.	49
3.2.3.3. Elementos de datos.....	50
3.2.3.4. Diagrama modelo de la base de datos.	56
3.3. Etapas de diseño.	57
3.3.3. Diseño de la interfaz.....	62
3.3.4. Actualización automática de estudiantes.	63
3.3.4.1. Implementación del servicio:	63
Capítulo 4: Manual de usuario del Sistema de Control del Pago del Estipendio Estudiantil.	65
4.1. Características del software.	65
4.1.1. Requerimientos mínimos.....	65
4.2. Instalación.....	66
4.3. Descripción del módulo administrativo.	66
4.4. Descripción del módulo para secretarías.....	70
Conclusiones:	73
Recomendaciones:	74

Introducción:

El desarrollo de la computación en los últimos años ha permitido un crecimiento en el uso de la misma. Los usuarios cada vez van siendo más expertos y esto implica que se le pueda dar un mejor uso a las herramientas informáticas. La tendencia actual es automatizar la mayor cantidad de procesos de tal forma que sea una máquina quien efectúe las labores que de otra forma han resultado complejas así como desarrollar tareas de cálculo que antes no se concebían debido a lo complejas que resultaban.

En la actualidad las redes de informatización y en general el crecimiento de Internet han desarrollado servicios múltiples a nivel mundial que van desde una página personal hasta servicios como tiendas virtuales que tienen o tuvieron su contraparte en las tiendas clásicas y supermercados. Se debe mencionar que el uso o no de Internet para un negocio es sólo una decisión a tomar pero muchos aseguran que “si no se está en Internet, se está perdiendo clientes y por lo tanto se pierde dinero”. Por lo tanto, para muchos hoy en día es imprescindible disponer de una computadora con acceso a Internet.

En nuestra universidad se dispone de una Intranet muy estable y de gran ancho de banda que permite brindar servicios Web para la Comunidad Universitaria. Nuestros estudiantes y profesores se van relacionando cada vez más con el proceso automatizado y el uso de la Web. Se han creado grupos de investigación (generalmente formados por estudiantes) que en cada área han desarrollado servicios para la red.

En la Dirección de Economía de la Universidad Central “Marta Abreu” de Las Villas no se cuenta con un sitio Web para brindar información. Se ha estudiado el estado actual de los sistemas que se utilizan en el desempeño de la colosal tarea que realizan (llevar el control de la Economía de cada área de la Universidad) y se ha observado que cuentan con un sistema, el AssetsNS (Sistema Económico Contable), como herramienta contable que no brinda ninguna información a la red debido a la falta de un sitio Web y de gestiones necesarias para implementar un servicio como ese.

También se puede ver que en la Dirección de Economía mensualmente se realiza el pago por estipendio a todos los estudiantes de la Universidad como está establecido a nivel nacional. Es necesario automatizar los procesos referentes al pago y vincular esto con el sistema existente en la UCLV para el control de la matrícula.

Los aspectos señalados anteriormente permiten plantear el siguiente:

Problema científico:

1. La dirección de Economía de la UCLV no cuenta con un sitio Web que le permita brindar información a las áreas universitarias.
2. En nuestro país no existe un sistema automatizado para el pago y control del estipendio estudiantil por parte de la Dirección de Economía de cada CES.
3. En la Universidad Central de Las Villas se utiliza un software desarrollado en Access que no cumple con las resoluciones para el pago del estipendio.

Dado dicho problema científico se plantean las siguientes hipótesis de investigación.

Hipótesis de investigación:

1. Se pueden obtener datos desde el Sistema Contable Integral (AssetsNS) que se explota en el mismo para generar reportes y publicarlos en la Web de la Dirección de Economía.
2. Todos los procesos que actualmente se ejecutan en el pago del estipendio estudiantil pueden automatizarse sin dificultad utilizando tecnología Web.

Para darle cumplimiento a las Hipótesis de investigación se proponen los objetivos del trabajo.

Objetivos generales:

1. Desarrollar la Web de la Dirección de Economía (DE) y servicios que brinden información al centro.
2. Automatizar el control del pago del estipendio de los estudiantes de la universidad como un servicio Web de la Dirección de Economía de la Universidad Central “Marta Abreu” de Las Villas.

Objetivos específicos:

1. Desarrollar la Web de la Dirección de Economía y servicios tales como mostrar a los estudiantes la información relacionada con el pago de su estipendio y brindar información sobre los útiles y herramientas y los medios básicos obtenidos del sistema AssetsNS.
2. Analizar y diseñar los procesos para el control del pago del estipendio estudiantil.
3. Diseñar la base de datos para el control del pago del estipendio estudiantil de tal forma que garantice la integridad de los datos.
4. Obtener los datos requeridos directamente del sistema de Control Estudiantil que actualmente es utilizado a fin de permitir actualizaciones automáticas en la base de datos e impedir trabajar con información falsa en el pago del estipendio estudiantil.
5. Implementar los módulos para el control del pago del estipendio.
6. Evaluar y probar del sistema de control del pago del estipendio.

Valor práctico:

Al disponer de un servicio mediante Web se logra automatizar el flujo de información desde las facultades hacia la Dirección de Economía con el fin de agilizar y facilitar los procesos del pago a estudiantes.

Permitirá a la persona responsable del pago del estipendio tener acceso a la administración del sistema, sección que le posibilitará llevar un control del estudiantado, gestionar los pedidos de reintegro y de pago retroactivo, hacer búsquedas, obtener los reportes necesarios y calcular las nóminas.

La administración de cada facultad tendrá la posibilidad de que un responsable mantenga actualizado el monto de pago a cada estudiante por los diferentes conceptos: si es alumno ayudante, orden 18, si tiene préstamo, etc. así como hacer peticiones de reintegros y pagos retroactivos sin necesidad de recorrer las largas distancias que existen entre las áreas y la Dirección de Economía.

El estudiante podrá mantenerse informado del día de cobro, ver cuando y como hacer pedidos de reintegros, informarse sobre cuánto cobra y porque, etc.

Al desarrollar servicios de reportes del sistema AssetsNS, los responsables de útiles y herramientas y activos fijos desde su puesto de trabajo podrán tener acceso a los listados y 10% sin necesidad de dirigirse al área económica lo que facilita el control de éstos recursos tanto a las áreas como a los auditores en futuras inspecciones. También la directiva del centro podrá tener acceso a la disponibilidad económica y su ejecución, aspectos de gran importancia para la toma de decisiones.

Principales aspectos que condujeron a la realización de los servicios:

Se ha hecho un estudio previo para valorar los servicios existentes en otras universidades del país y se pudo observar que no existen aplicaciones encargadas de llevar todo el control necesario en cuanto al pago de estipendio.

En estos momentos la persona encargada del pago de estipendio estudiantil en la UCLV labora con una pequeña aplicación desarrollada en Access la cual sólo tiene implementado el cálculo de nóminas para imprimirlas, no cuenta con herramientas que brinden la posibilidad de la comunicación con las secretarías de cada facultad.

La Dirección de Economía de la UCLV no contaba con sitio Web por lo que no brindaba información referente al centro.

Actualmente en todas las universidades del país utilizan el sistema AssetsNS para llevar control de los procesos contables de cada centro, no se han desarrollado herramientas que brinden información o reportes como por ejemplo: los útiles y herramientas de cada área.

Viabilidad:

Para realizar el proyecto se cuenta con un servidor WEB debidamente instalado y capacitado para la realización de esta tarea. El personal del centro que realiza el control del pago de estipendio está estrechamente vinculado a este proyecto por lo que su utilización práctica es inminente.

Actualmente la UCLV cuenta con una red local cuya conexión es estable y rápida. Esto facilita el trabajo en la obtención de la información necesaria en el desempeño de dicha tarea así, una vez terminado el proyecto, permitirá a los estudiantes y personal de las facultades relacionados con el estipendio estudiantil, una fácil conexión con la Dirección de Economía y la información confiable referente a las operaciones relacionadas con el mismo.

La programación en Web permite desarrollar un diseño orientado a objetos y/o una programación estructural con funciones específicas. Se cuenta con módulos para conexión a bases de datos, envío de correo, manejo de fechas. También permite el uso de plantillas las que hacen independiente el diseño y la programación. Haciendo uso correcto de estos elementos en el control del proceso del pago estudiantil se puede lograr flexibilidad en los módulos ante posibles cambios futuros.

El lenguaje que se utilizará (PHP) permite la creación de variables que sólo se encuentran en la memoria del servidor mientras el visitante está activo en las páginas. El uso de ellas permite llevar control de quien está accediendo y en cada momento restringir el acceso a la zona no autorizada.

Para la autenticación se hará uso de los servicios que el Grupo de Redes tiene implementado y que permite identificar el usuario por su nombre y clave del dominio UCLV, por lo que no habrá que tener una lista de usuarios con claves en la base de datos del sistema, sólo un identificador a cada uno de ellos con el rol que ocupan dentro del sistema.

Capítulo 1: Uso de la tecnología Web para desarrollar aplicaciones y servicios.

En este capítulo se verán algunos conceptos básicos sobre la tecnología Web, su necesidad, ventajas y sus avances. Se explica algunas de sus funcionalidades, la forma de trabajar y herramientas a emplear a la hora del desarrollo de una aplicación Web, la seguridad, uso de bases de datos, etc.

También se nombran algunas aplicaciones desarrolladas en la UCLV que hoy forman parte de la Comunidad Universitaria y sus servicios para la Intranet.

1.1. La tecnología Web.

La Web es una red mundial de información que logra integrar los documentos que están contenidos en Internet. Esta integración se traduce en la accesibilidad universal a los documentos que ella contiene y se potencia mediante la utilización de hipertexto. Los hipertextos están soportados por estándares de servicio y codificación de textos, denominados HTTP (Hyper Text Transfer Protocoly, Protocolo de Transferencia de Hipertexto) y HTML (Hyper Text Markup Language, Lenguaje de Marcación de Hipertexto) respectivamente. La tecnología Web son las herramientas que se utilizan para desarrollar aplicaciones y servicios para la Web.

1.1.1. La necesidad y ventajas de la tecnología Web actual.

En la actualidad existen muchos lenguajes de programación, cada uno tiene sus características individuales, sus ventajas y desventajas lo que hace que existan distintos criterios al seleccionar uno u otro para desarrollar alguna aplicación. Esta elección depende principalmente del lugar donde se laborará con la aplicación; si es para usarla desde pocas computadoras, si estas no tienen conexión con redes ni acceso a servidores, es más recomendable desarrollar una aplicación de escritorio, si por el contrario, sería utilizada por muchas personas en una misma institución, mediante redes locales o redes más grandes como Internet es más recomendable desarrollar una aplicación mediante Web.

Actualmente la tendencia a disponer de redes de información, la conectividad entre puntos distantes nos hacen pensar en plataformas que nos permita desarrollar aplicaciones más allá de las que se puede lograr con Delphi, Visual Basic, entre otros, más nos interesamos entonces por desarrollar aplicaciones que puedan ser utilizadas por distintos usuarios incluso sin tener necesidad de instalar un software en cada estación de trabajo.

1.1.2. La tecnología Web frente a otras tecnologías existentes.

No se puede afirmar que la tecnología Web remplazará en el futuro a otras tecnologías, actualmente estamos en presencia de dos tendencias en la programación, su uso tiene mucha dependencia con la finalidad que tendrá la aplicación que se desarrolle. Hasta el momento las ventajas de cada cual son muy semejantes con la diferencia de lo anterior, no obstante se comienzan a dar soluciones oportunas a otras tecnologías como el .NET de Microsoft que brinda posibilidades de acceder remotamente a través de Web y hacer uso de todas sus funcionalidades mediante estaciones remotas.

Se entiende que el lenguaje C es muy potente para desarrollar programas que corren localmente y por su parte el lenguaje Perl o el Action Script vienen con el Linux y son muy utilizados para implementar pequeñas herramientas que faciliten el trabajo a los administradores del UNIX; cuando se habla de *tecnología Web frente a otras tecnologías* no se refiere a la competencia que pueda haber con estos lenguajes, sino se ve como una comparación entre la tecnología utilizada para aplicaciones de escritorio y la tecnología Web a la hora de hacer una aplicación con interfaz de usuario, comparando las ventajas que trae el uso de la red en dependencia del producto final que se desee obtener.

1.2. Avances en la tecnología Web.

Muchas de las tecnologías han avanzado exponencialmente. Este es el caso de la famosa Ley de Moore que dice que la capacidad de los microprocesadores se dobla cada 18 meses. Esta predicción fue realizada en 1965 [1]. Recientemente, un crecimiento similar se puede apreciar en Internet donde el número de computadoras conectadas se doblaba cada 15 meses [2] (que en la realidad no ocurre debido a que ya son cifras muy elevadas). El crecimiento de la Web es también impresionante; desde 1993, el número de servidores se dobla cada 3 meses, sobrepasando hoy los 38 millones [3]. Todo esto hizo inspirar a desarrollar plataformas para el uso eficiente de Internet a compañías como Yahoo, Microsoft, Google, EBay, Amazon y AOL, por nombrar algunas, cuyos servicios hoy son los más utilizados en la Web según [16].

En un comienzo sólo existía la posibilidad de desarrollar páginas estáticas con contenido que no cambiaba al no ser que fuera modificado por su realizador o alguna persona encargada. El visitante sólo era un observador que navegaba por las páginas Web las que eran cargadas desde un servidor remoto mediante el protocolo de transmisión de datos HTTP (Hyper Text Transfer Protocol, Protocolo de Transferencia de Hipertexto). Era necesario entonces interactuar con el visitante, pedirle sus datos personales para identificarlo, conocer las páginas que visitaba, leer información de bases de datos, guardar datos del visitante mediante formularios, etc. entonces aparece por primera vez las llamadas Páginas Dinámicas. Mediante estas pequeñas aplicaciones los desarrolladores Web pudieron permitir

que los visitantes interactuaran con el sitio Web, ésto también les sirvió para desarrollar pequeñas aplicaciones que le permitían actualizar sus propias páginas mediante la Web.

Las páginas dinámicas son una vía fiable para lograr una mayor utilidad de las redes y sus servicios, la integración con bases de datos hacen posible el desarrollo no sólo de páginas Web sino de sitios o sistemas más complejos que pueden ir desde un simple contador de visitas a una página, hasta sistemas tan profesionales como el manejo de las transferencias de un banco.

1.2.1. El Open Source.

La tecnología *Open Source* (fuente abierta) hoy en día está en punta en el mundo principalmente por las ventajas que trae:

- Los productos *Open Source* con licencia GPL (General Public License, Licencia Pública General) tienen un costo muy reducido en comparación con productos equivalentes e incluso muchos son completamente gratuitos.
- Alta fiabilidad, escalabilidad y rendimiento.
- Uso de estándares, que posibilitan la integración con otros sistemas y la evolución de los existentes.
- Control absoluto sobre los procesos y la información que se trata.
- Invulnerable frente a cambios y versiones que se desarrollen.
- Optimización en el uso de la plataforma existente.

Ya existen muchos algoritmos o plataformas (phpmailer, pear, wact) correctamente desarrollados que se pueden obtener de Internet y utilizar en nuestros programas **y** así hacer mucho más rápido el desarrollo de cualquier sistema que utiliza la tecnología *Open Source* que no por ser “libre” estamos ante la presencia de desarrollos de poca calidad. El *Open Source* ha llegado hasta tal punto que hoy en día existen innumerables productos que se han convertido en un estándar y populares como está el llamado PHPMyAdmin que es un gestor de bases de datos MySQL muy completo desarrollado en PHP y se puede descargar gratis desde su sitio en Internet. (phpmyadmin.net).

Ahora, ¿por qué estamos en presencia de una tecnología gratis?, respondamos con un ejemplo: Si queremos desarrollar una aplicación en PHP que lleve accesos a Bases de Datos y envíos de correos (por indicar sólo dos funcionalidades), podemos pensar en el uso de tecnología *Open Source* antes de comenzar a ver funciones primitivas en el lenguaje. Entonces nos encontraremos con decenas de clases o algoritmos programados por diferentes desarrolladores en todo el mundo. Como se trata de software gratis, este código se encuentra listo para descargar y utilizar. Todo esto se va enriqueciendo con la llegada de nuevas ideas, muchos programadores primero se preguntan: ¿qué no se ha

desarrollado? o ¿de lo desarrollado, hay algo que se puede mejorar? y si obtienen una respuesta es muy probable que intente “batir ese hueco” porque es para el bien de todos, entonces al tener su producto terminado lo aprovechan como propaganda para mejorar su currículo y encontrar mejores puestos de trabajo en empresas importantes o para dar promoción a su empresa.

1.2.2. ¿Qué es el Software libre?.

Software libre es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente. El software libre suele estar disponible gratuitamente en Internet, o a precio del costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente. Análogamente, el software gratuito (denominado usualmente Freeware) incluye en algunas ocasiones el código fuente; sin embargo, este tipo de software no es libre en el mismo sentido que el software libre, al menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

Según se señala en [15], en 1984, Richard Stallman comenzó a trabajar en el proyecto GNU (General Public License, Licencia Pública General) fundando la FSF (Free Software Foundation, Fundación de Software Libre) un año más tarde. Stallman introdujo una definición para free software y el concepto de "copyleft", el cual desarrolló para dar a los usuarios libertad y para restringir las posibilidades de apropiación del software.

De acuerdo con tal definición, si el software es libre se garantiza:

- La libertad para ejecutar el programa con cualquier propósito (llamada "libertad 0")
- La libertad para estudiar y modificar el programa ("libertad 1")
- La libertad de copiar el programa de manera que puedas ayudar a tu vecino ("libertad 2")
- La libertad de mejorar el programa, y hacer públicas tus mejoras, de forma que se beneficie toda la comunidad ("libertad 3")

1.3. Diseño de sitios Web.

Antes de desarrollar un sitio Web se debe tener bien en cuenta un conjunto de elementos que definirán la finalidad del sitio y en base a eso definir que tipo de diseño y estructura llevará. Primeramente se hacen dos preguntas claves: ¿qué tipo de información se publicará en el sitio? y ¿a quienes va dirigido, en base a las respuestas de esas preguntas se implementa el diseño y la estructura.

1.3.1. Modelado de un sitio Web.

El diseño y modelado de un sistema computacional muchas veces es olvidado por los programadores y analistas. Es importante disponer de un diseño antes su desarrollo si se desea obtener un sistema fiable en el futuro y que no lleve demasiados errores de programación a la hora de ser implementado el producto final, cosa que es muy molesto y puede traer muchos problemas para la entidad y las personas que lo utilizan.

Los modelos se crean para lograr un mejor entendimiento de un ente a representar. En fases iniciales del desarrollo de sistemas, específicamente en la fase de Análisis, se deben crear los modelos conceptuales del sistema. Según [4], estos modelos deben describir el problema y los requerimientos de los usuarios, considerando que en una etapa posterior de diseño se construirán los modelos que entreguen la solución que satisfaga los requerimientos de los usuarios y el problema planteado. Durante el análisis, el analista debe poner especial atención en el dominio del conocimiento, objetivos, requerimientos y entorno del sistema, es decir, debe “orientarse al problema” y no limitarse a modelar la solución al problema. De esta forma se facilita la validación por parte de los usuarios.

Los modelos creados hacen uso de notación gráfica que representa principalmente la información, los procesos y el comportamiento del sistema los que pueden complementarse con texto. El rol de los modelos se puede sintetizar en:

- Ayudar al analista a entender la información, la función y el comportamiento del sistema, haciendo por tanto más fácil y sistemática la tarea de analizar requerimientos.
- Convertirse en el punto de comparación entre lo logrado y lo planificado.
- Fundamentar el diseño, proporcionando al diseñador del sistema (no necesariamente tiene que ser propio el analista) una representación lógica, o esencial, de la implementación.

En [5] destaca la importancia de modelar un sitio Web para ayudar a manejar la variedad de un sistema y para que las aplicaciones en Web no se transformen en aplicaciones muy complejas. Si se considera además, que un sistema se debe representar por distintos modelos consistentes entre sí, cada uno con un propósito específico y con un nivel distinto de abstracción, la modelación se hace imprescindible.

Un elemento básico de un sitio Web es la página Web, por lo que será un elemento que aparecerá modelado en todo sitio Web. En [5] se propone la utilización del UML para representar una página como un objeto, con las propiedades de tal objeto: fuentes, tablas, texto, etc.; y los scripts (código fuente) de la página se representarían como métodos de un objeto página.

Un "sistema" lo podemos ilustrar, desde el punto de vista conceptual, como una caja, a la cual se introducen determinadas entradas, y salen determinados resultados [10]. El modelado del mismo debe reflejar tal funcionalidad deteniéndose en la especificación de los estados por el cual pasa la información que se procesa. Se debe lograr también modularizar el máximo las partes componentes del sistema de tal forma que cada modulo quede como una "caja negra", o sea, que dado un conjunto de entradas brinde salidas consistentes y que para una reutilización del mismo no sea necesario conocer las funciones implementadas ni los datos internos y mucho menos tener que modificarlos. Según [10], un sistema es un conjunto de módulos organizados en jerarquías que cooperan y se comunican para realizar una tarea.

1.3.2. Usabilidad y accesibilidad Web.

La usabilidad y accesibilidad Web es un tema algo olvidado o que no toman en cuenta muchos desarrolladores. Cuando se habla en estos términos se refiere al hecho de que una Web usable y accesible es aquella que tiene la estructura y organización de textos, uso de colores, elementos multimedia, etc. que con lo cual se logre una interactividad aceptable y una navegación intuitiva con el usuario menos experto o con incapacidades físicas que permita que sus contenidos sean altamente provechosos para la mayoría de los usuarios de Internet.

También se debe tener en cuenta que la incapacidad para navegar por Internet puede estar dada por la velocidad de conexión a Internet, si se quiere que una Web sea accesible se debe lograr que los elementos utilizados como imágenes, animaciones flash, etc. tengan el menor tamaño posible, si es preciso se recomienda minimizar la calidad para hacer más pequeño los archivos o hacer dos versiones para los visitantes de mejor conexión o visitantes que se conectan a velocidades menores de 56Kb/s. Muchas veces visitamos sitios Web donde en su página principal o "Splash Page" viene la opción de acceder por la versión Flash o sólo HTML, precisamente se está teniendo como referencia que el usuario de menor conexión puede navegar más libremente por los contenidos en HTML que por los contenidos en Flash donde seguramente se puedan encontrar con imágenes animadas y sonido que hacen más lento el proceso de carga y muchas veces son elementos decorativos sin mucha importancia.

Cada vez que un usuario llega a un sitio Web nuevo, muchas veces le cuesta trabajo identificarse y simplemente encontrar lo que quiere. En [7] se explica la importancia de capturar al visitante en los primeros 8 segundos, si en ese período de tiempo el visitante no piensa que en el sitio puede encontrar lo que busca, simplemente terminará cerrando el navegador o minimizar la ventana para buscar otros sitios semejantes lo que muchas veces se queda como una página olvidada y en desuso aunque esté proporcionando el mejor contenido sobre lo que el visitante estuvo buscando.

El uso de Scripts y Applets deben ser directamente compatibles con tecnologías accesibles [8]. Los navegadores muchas veces no están actualizados y se corre el riesgo de que el contenido no sea visible si se utilizan funciones no soportadas por los mismos.

En [6] se plantea la importancia de cuidar el uso de elementos parpadeantes en los sitios Web y que en innumerables ocasiones lo vemos en las animaciones flash o gif animados muchas veces porque queremos dar mejor imagen y viveza a nuestro sitio. Estos elementos pueden ser muy perjudiciales a personas que son fotosensiblemente epilépticas.

Proporcionar mecanismos claros y consistentes de navegación así como usar el lenguaje apropiado más claro y simple como sea posible al contenido de las páginas sin olvidar lo expuesto en párrafos anteriores, es una buena alternativa para lograr un uso eficiente de la Internet y hacer llegar la información a la mayoría de los usuarios.

1.4. Programación de sitios Web.

En este epígrafe aparecen algunos aspectos básicos para desarrollar sitios Web. Primeramente se debe conocer que una página Web no es más que un código fuente interpretado por el navegador de Internet y mostrado en pantalla como texto, tablas, colores. Este código fuente generalmente se almacena en el servidor Web y es cargado al navegador al solicitar la página.

Una Web también puede tener imágenes, animaciones y sonido, estos son elementos que se cargan independientemente al código fuente, el HTML es quien los solicita y los hace funcionar debidamente cuando la página es cargada.

1.4.1. El HTML.

El HTML es un lenguaje de marcas diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas Web. Gracias a Internet y a los navegadores del tipo Explorer o Netscape, el HTML se ha convertido en uno de los formatos más populares que existen para la construcción de documentos.

1.4.1.1. Algunas etiquetas básicas del HTML:

Las etiquetas básicas de HTML, de obligada presencia en todo documento son:

<HTML>: Es la etiqueta que define el inicio del documento HTML, le indica al navegador que todo lo que viene a continuación debe tratarlo como una serie de códigos HTML.

<HEAD>: Define la cabecera del documento HTML, esta cabecera suele contener información sobre el documento que no se muestra directamente en el navegador. Como por ejemplo el título de la ventana de su navegador. Dentro de esta etiqueta podemos encontrar:

<TITLE>: Define el título de la página. Por lo general, el título aparece en la barra de título encima de la ventana.

LINK: Para definir algunas características avanzadas, como por ejemplo los estilos usados para la visualización de la página, ejemplo :<link rel="stylesheet" href="/style.css" type="text/css">.

<BODY>: Define el contenido principal o cuerpo del documento, ésta es la parte del documento HTML que se muestra en el navegador, dentro de esta etiqueta pueden definirse propiedades comunes a toda la página, como color de fondo y márgenes. Dentro del cuerpo <BODY> podemos encontrar numerosísimas etiquetas. A continuación se indican algunas a modo de ejemplo:

<H1>, <H2>, ..., <H6>: Encabezados o títulos del documento en diferentes tamaños de fuente.

<P>: Párrafo nuevo.

**
**: Salto de línea forzado.

<TABLE>: Comienzo de una tabla (las filas se identifican con <TR> y las celdas dentro de las filas con <TD>).

<a>: Indica la existencia de un hipervínculo o enlace, dentro o fuera la página Web. Debe definirse el parámetro *href* que indica a donde apunta en enlace.

<DIV>: Comienzo de un área especial en la página.

1.4.2. Las hojas de estilo CSS.

Las hojas de estilo en cascada (Cascading Style Sheets, CSS) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML. La idea del desarrollo utilizando CSS es separar la estructura de un documento de su presentación o diseño.

La información de estilo puede ser adjuntada tanto como un documento separado o en el mismo documento HTML. En este último podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "style".

1.4.2.1. Ventajas de utilizar CSS:

- Control centralizado de la presentación de un sitio Web con lo que se agiliza de forma considerable la actualización del mismo.
- Algunos navegadores permiten a los usuarios especificar su propia hoja de estilo local que será aplicada a un sitio Web remoto, con lo que aumenta considerablemente la accesibilidad. Por ejemplo, personas con deficiencias visuales pueden configurar su propia hoja de estilo para aumentar el tamaño del texto o remarcar más los enlaces.

- El documento HTML en sí mismo es más claro de entender y se consigue reducir considerablemente su tamaño.

1.4.3. El JavaScript.

JavaScript es un lenguaje interpretado orientado a las páginas Web, con una sintaxis semejante a la del lenguaje Java, se utiliza en páginas Web HTML para realizar tareas y operaciones en el marco de la aplicación cliente.

El JavaScript es utilizado principalmente para mejorar la interactividad del usuario con el navegador ya que es la herramienta que interpreta el código y quien finalmente tiene relación con el visitante.

Como todos los lenguajes de programación orientados a los objetos, el JavaScript establece una jerarquía de objetos, que permite definir propiedades. Los principales objetos son: window, document, history, location y navigator. Dichos objetos están comunicados entre sí por relaciones estructurales del tipo: *Objeto1.Objeto2.Propiedad2* donde *Objeto1* está en el nivel más alto; *Objeto2* es una propiedad de *Objeto1* y *Propiedad2* es una propiedad de *Objeto2*.

JavaScript se compone de elementos de programación como: argumentos, gestores de eventos, funciones, literales, expresiones, métodos, objetos, operadores, propiedades, instrucciones, valores y variables.

Entre otros, los usos más frecuentes del JavaScript son:

- Mostrar mensajes en la barra de estado del navegador.
- Crear botones activos mediante imágenes.
- Gestionar Cookies (fragmento de información que se almacena en el disco duro del visitante a través de su navegador).
- Mostrar anuncios (banners) de forma rotativa en una misma posición de la página.
- Redireccionar según el tipo de navegador.
- Crear ventanas independientes.
- Controlar datos introducidos en un formulario.
- Crear menús desplegables dentro de una página Web.

1.4.4. El lenguaje PHP.

Al igual que el ASP (Active Server Pages, Páginas Activas del Servidor) de Microsoft, el PHP se ha convertido en un lenguaje de programación muy utilizado en el desarrollo de páginas dinámicas.

Se puede ver en [17] que el PHP fue considerado el lenguaje de programación del 2004 según cálculos hechos por la TIOBE (Programming Community Index, Índice de Programación de la Comunidad) para el cual se utiliza información recogida de los motores de búsqueda populares, se basa en la disponibilidad mundial de ingenieros expertos y de cursos disponibles.

1.4.4.1. ¿Qué es PHP?.

El PHP (Hypertext Preprocessor) es un lenguaje OPEN SOURCE interpretado de alto nivel. El código PHP es ejecutado en el servidor Web quien devuelve su salida (en caso de que la tenga) al cliente o navegador Web.

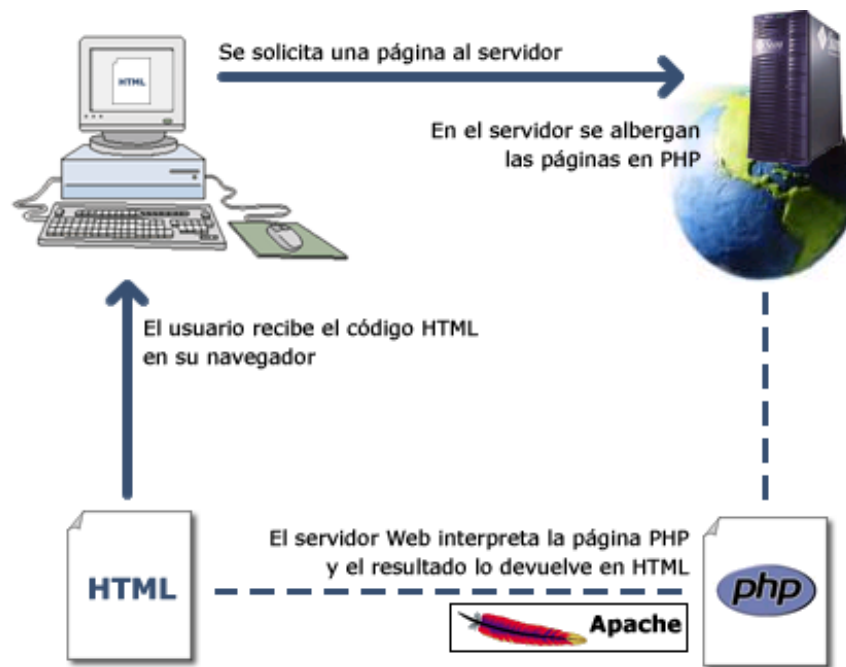


Figura 1. Solicitud de página PHP al servidor Web.

Cuenta con funciones muy específicas, para el trabajo con matrices, matemáticas, para el trabajo con la mayoría de los servidores de bases de datos existentes, funciones para el manejo de ficheros, entre muchísimas otras.

En estos momentos se ha dividido en dos, el PHP4 y el PHP5, los desarrolladores implementaron el PHP5 con una nueva metodología que mejora sustancialmente el rendimiento y el trabajo con objetos principalmente, pero no han dejado olvidado el PHP4 producto de que la mayoría de los sitios Web programados en PHP no van a emigrar al PHP5 tan rápidamente, aunque en teoría la versión 5 es compatible con la versión 4. Por eso se entiende que se trata de un “abrazo mortal” porque la mayoría de los servidores no emigran a PHP5 por temor de dejar algunos sitios sin funcionar y los programadores no hacen su versión en PHP5 debido a que no hay servidores donde ejecutar su

código. En todo esto hay que tener presente que muchos programadores contratan sus servidores de Hosting en Internet que comparten con muchos otros programadores o Webmasters, por lo tanto no depende sólo de uno sino de la compañía proveedora del servicio.

1.4.4.2. Integración del PHP con HTML.

El PHP ha sido implementado con el fin de poderse incluir embebido en páginas HTML de tal forma que indicando sus bloques con etiquetas `<?php` y `?>` pueda ser interpretado solo el PHP y el HTML se envía limpiamente al navegador.

La mayor ventaja que lleva esta inclusión del código PHP en páginas HTML es utilizar salidas, que pueden ir acompañadas con estructuras repetitivas o no, mediante funciones como `echo()`, `print()`, etc. que genere y permita enviar código HTML al navegador el cual interpreta y muestra finalmente al visitante.

Pongamos un ejemplo: Supongamos que queremos enviar al navegador alguna información que se encuentra almacenada en un fichero, el procedimiento seria de la siguiente forma:

1. Abrir el fichero.
2. Leer el fichero en una variable.
3. Imprimir la variable conjuntamente con código HTML.

Entonces este tercer paso puede ser de la siguiente forma, suponiendo que la variable utilizada es `$texto`:

```
<?php
echo ("<div align='center'><font size='12'><b>". $texto. "</b></font></div>");
?>
```

En ese ejemplo la salida sería:

Este es el texto

Suponiendo que esa cadena sea la que contenga el fichero. Obtendríamos el texto en negrita de tamaño 12 y centrado debido a las etiquetas `<div>` y `` con sus respectivas propiedades `align='center'` y `size='12'`.

De igual manera podemos ver un ejemplo para cuando se trata de un ciclo, si queremos que la cadena se repita muchas veces y se inserte en una tabla podemos poner el siguiente código:

```

<table width="100%" border="1" cellspacing="1" cellpadding="1">
<?php for($i=1; $i<=10; $i++){ ?>
    <tr>
        <td>
<?php
echo("<div align='center'><font size='12'><b>Este es el texto</b></font></div>");
?>
        </td>
    </tr>
<?php } ?>
</table>

```

El anterior código tendría como salida lo siguiente pero con 10 filas:

Este es el texto
Este es el texto

Podemos apreciar el gran valor práctico que tiene poder incluir códigos PHP dentro de páginas HTML, todo esto permite brindar información útil que sólo se encuentra en el servidor, también puede ser utilizado para hacer operaciones sin depender del navegador que se este utilizando por parte del visitante y olvidar las posibles incompatibilidades con los mismos.

1.4.4.3. Integración con JavaScript.

Al igual que la integración con el HTML, el PHP puede incluirse dentro de las páginas Web que contengan código JavaScript. La ventaja principal de ello es que se puede generar código JavaScript desde PHP que próximamente será ejecutado por el navegador Web. El poder hacer esto constituye una ventaja para los programadores ya que la inclusión de JavaScript en las páginas Web puede ser muy efectivo en cuanto a la interactividad con el visitante.

Por ejemplo, un código muy simple en JavaScript de uso muy frecuente que es utilizado para informar algo al visitante es el siguiente:

```

<script language="javascript">
    alert("Este es el texto");
</script>

```

El mismo tendría como salida una alerta que en el Internet Explorer se vería así:



Ahora bien, “dentro” de este código JavaScript puede incluirse PHP que dé cómo salida el texto a mostrar:

```
<script language="javascript">  
    alert("<?php echo('Este es el texto'); ?>");  
</script>
```

De igual manera puede utilizarse para lo que se desee, desde estructuras repetitivas en PHP puede generarse código JavaScript con información leyéndose de bases de datos. Esta funcionalidad es muy utilizada en aplicaciones de comunicación online como chat y centro de ayuda a clientes.

1.5. Uso de Bases de Datos en los sitios Web.

El PHP tiene implementadas funciones que permiten acceder a los más populares servidores de bases de datos (Microsoft SQL, My SQL, Oracle, Postgre SQL, etc).

Si vemos que añadido a la posibilidad que el PHP brinda para el trabajo con la Web le sumamos que pueda interactuar con servidores de bases de datos mediante el potente lenguaje estándar SQL (Structured Query Language) podemos pensar que la programación en la Web se ha convertido en un nuevo método de programación cuyas herramientas pueden ser utilizadas para crear sistemas potentes que utilicen bases de datos y a su vez que sean accesibles por todos los internautas.

Muchos programadores, aprovechando esta funcionalidad, comenzaron a implementar entonces sus sitios Web de tal forma que otras personas pudieran adicionar contenido y modificarlo sin necesidad de entender el HTML lo que marcó una ventaja que hoy en día esta siendo utilizado en nuestro país en los sitios Web de periódicos y revistas permitiendo a sus Webmasters adicionar cada número con mucha facilidad.

1.5.1. El MySQL.

Según [11], MySQL es una de las bases de datos más populares desarrolladas bajo la filosofía de código abierto. Tiene una bien merecida reputación de ser un servidor de bases de datos muy rápido que también es muy fácil de configurar y usar, además de que en los últimos años su popularidad ha

crecido notablemente debido a que se utiliza en infinidad de sitios Web que requieren hacer uso de una base de datos.

La versión 4.0 del MySQL soporta cinco tipos de tablas: MyISAM, ISAM, HEAP, BDB (Base de datos Berkeley), e InnoDB. BDB e InnoDB son ambos tipos de tablas transaccionales [12]. Además de poder trabajar con transacciones en MySQL, las tablas del tipo InnoDB también tienen soporte para la definición de claves foráneas, por lo que se nos permite definir reglas o restricciones que garanticen la integridad referencial de los registros.

1.6. La seguridad en la tecnología Web.

En ningún tipo de aplicación se puede olvidar garantizar la seguridad de los datos, la Web tiene sus propias herramientas que mediante su uso permiten a los programadores desarrollar páginas seguras para sus clientes o usuarios.

1.6.1. Variables globales.

En PHP es muy importante tener bien configurado el servidor Web para mayor seguridad. A partir de PHP 4.2.0, el valor por defecto de la directiva PHP `register_globals` es off (desactivada) [9]. Esto afecta el conjunto de variables predefinidas, por lo tanto no podremos utilizar directamente una variable como `$nombre` si estamos pasando como parámetro por URL un valor como sigue: <http://www.dominio.com/index.php?nombre=juan> sino que tendremos que utilizar la respectiva matriz definida para variables por el método GET, o sea que el resultado lo obtenemos en `$_GET['nombre']`, análoga a la antigua matriz `$HTTP_GET_VARS` de poco uso. De la misma forma pero con `$_POST['nombre']` podemos capturar los valores enviados si lo hacemos por el método POST.

Lo anterior es importante porque cuando utilicemos variables POST no permitimos que cualquier usuario envíe datos por medio de variables HTTP (GET, COOKIE, FILES, entre otras), con el uso de variables simples con `register_globals` activado existe la vulnerabilidad de entender los distintos tipos de envío de información con el uso de una sola variable lo que puede ser muy cómodo para el programador menos avanzado pero menos inseguro para la propia aplicación en desarrollo.

1.6.2. Seguridad en la programación.

Muchas veces los fallos de seguridad vienen producidos por el propio servidor Web y sus configuraciones, pero tenemos que tomar en cuenta errores comunes que vienen a darse por parte del programador Web que puede conllevar en el futuro a catástrofes producidas por personas experimentadas que desean hacer el mal.

1.6.2.1. Inyección SQL:

Es poco conocido por los programadores un simple problema de seguridad que afecta a aplicaciones dinámicas desarrolladas en Web que utilicen servidores de bases de datos como Microsoft SQL, Postgre SQL, MySQL, etc. debido al uso del lenguaje SQL como un estándar común para todos estos servidores. Se está hablando de inyección SQL que se basa en el envío de sentencias SQL al servidor mediante variables HTTP.

Por ejemplo: tenemos una aplicación que mediante un formulario envíe datos al servidor que serán utilizados para ejecutar una sentencia SQL a la base de datos. Este formulario tiene un campo llamado nombre y al llenar y dar “enviar” se enviará esa información a otra página quien se encargaría de ejecutar la sentencia SQL:

```
<form name="enviar" method="post" action="recibe.php">
  <input name="nombre" type="text" id="nombre">
  <input type="submit" name="Submit" value="Enviar">
</form>
```

La página que recibe la información (recibe.php) enviada por el formulario tendría la siguiente sentencia SQL suponiendo que la tabla se llame “estudiantes” con un campo “nombre”.

```
<?php $sql = "DELETE FROM estudiantes WHERE nombre = '$_POST['nombre']'; ?>
```

Esto eliminaría todos los registros en la base de dato que contenga el valor indicado en el campo “nombre” de la tabla “estudiantes”. Hasta ahora bien, pero pensemos que el envío de la variable “nombre” por HTTP POST puede ser desde cualquier sitio en Internet o desde una página cualquiera creada por un visitante malicioso, sólo necesitaría tener “cargada” en su navegador una página que contenga el mismo formulario pero con la propiedad action=”http://www.dominio.com/recibe.php”, estaríamos enviando la información deseada a la misma página y ésta será interpretada como válida si no se tienen en cuenta otras medidas de seguridad, como ver de que dirección es que viene la información mediante la variable \$_SERVER["HTTP_REFERER"]. El código enviado dentro del campo texto puede ser: “juan; DELETE FROM estudiantes” lo que producirá que la sentencia SQL que se ejecute sería:

```
DELETE FROM estudiantes WHERE nombre = juan; DELETE FROM estudiantes
```

Como se ve, produce dos sentencias SQL. La primera eliminaría todos los registros donde el campo nombre de la tabla estudiantes es *juan*; esto puede ser, por ejemplo, las participaciones en un evento por dicho estudiante. La segunda sentencia producida eliminaría a todos los registros de la tabla ya que no esta acotado la cláusula "DELETE" por un "WHERE".

Existe un método muy censillo de impedir la inyección SQL y es aplicándole a cada dato de entrada (variable nombre en el ejemplo) una función que lleve todo los caracteres punto y coma (;) y comilla simple (') (que igualmente pueden utilizarse para alargar sentencias mediante WHERE) a comillas dobles ("), el código a continuación soluciona el problema del ejemplo:

```
<?php
$tmp_text = str_replace(";", "\'", $_POST['nombre']);
$tmp_text = str_replace("'", "\"", $tmp_text);
$sql = "DELETE FROM estudiantes WHERE nombre = ".$tmp_text;
?>
```

Lo anterior produce una salida SQL inválida al encontrarse comillas dobles (") que hace cerrar la sentencia inesperadamente y produce un error sin ejecutarse el *DELETE* malicioso.

1.6.3. Seguridad del servidor Web (SSL).

SSL (Secure Sockets Layer) es un protocolo diseñado por la empresa Netscape Communications, que permite cifrar la conexión, incluso garantiza la autenticación. Se basa en la criptografía asimétrica y en el concepto de los certificados. La versión se conoce como TLS (Transport Layer Security, Capa de Transporte Segura), estandarizada por el IETF (Internet Engineering Task Force, Grupo de Trabajo en Ingeniería de Internet) que es una organización internacional abierta de normalización. [14]

Su mayor ventaja es que funciona entre la capa TCP (Transmission Control Protocol, Protocolo de Control de Transmisión) y la capa de aplicación, por esto es muy fácil usarlo para proteger los protocolos de la capa de aplicación (por ejemplo: FTP, Gopher, HTTP, etc.) sin tener que realizar cambios importantes en los mismos.

En el servidor Web Apache viene esta funcionalidad sin costo alguno protegido por la licencia de software libre GNU.

1.7. Uso de la Intranet UCLV.

La interconexión de las computadoras de toda la Universidad nos ha permitido pensar en desarrollar sistemas basados en Web. La principal ventaja del uso de la Web es precisamente que desde cualquier computadora conectada en red puede hacer uso de los servicios desarrollados, el otro

requisito sería disponer de un navegador y todas las versiones del Windows a partir del Windows 98 tienen incluido el Internet Explorer en su instalación.

Permitir el uso desde cualquier computadora es una preocupación por problemas de seguridad pero no un motivo para descartar la posibilidad del uso de la red. Además si la tecnología a utilizar es Open Source se puede garantizar buena seguridad incluso mejor que una aplicación de escritorio.

Actualmente en nuestra universidad se pretende llegar a tener una Comunidad Virtual Universitaria que ayude a efectuar todos o la mayoría de los procesos que actualmente se efectúan a mano por lo que hay que recorrer largas distancias entre las áreas para llevar información en papel. Alguno de estos problemas se resuelve mediante el uso del correo electrónico lo que constituye una herramienta poco segura, menos estable y funcional que las aplicaciones Web.

1.7.1. Ejemplo de aplicaciones desarrolladas en la UCLV utilizando tecnología Web:

Actualmente en nuestra universidad existen servicios en la red dirigidos a todos los usuarios que de una forma u otra han ayudado al proceso docente educativo.

Las facultades cuentan con un sitio propio bajo el dominio correspondiente y alojado en servidores locales en cada una. Esto ha sido posible gracias a que grupos de investigación conjuntamente con los administradores se han visto integrados en estas tareas de Webmaster.

A continuación se mencionan algunas de las aplicaciones que han sido desarrolladas con un fin específico, su funcionalidad no hubiese podido ser posible sin el uso de la tecnología Web como herramienta a utilizar:

- Sitio Internacional UCLV: Sitio Web oficial Internacional de la Universidad Central de Las Villas.
<http://www.uclv.edu.cu>
- Intranet2 UCLV: Intranet de la Universidad Central de Las Villas
<http://intranet.uclv.edu.cu>
- SEPAD: Sistema de Enseñanza Personalizada A Distancia.
<http://sepad.uclv.edu.cu>
- CDICT: Centro de Documentación e Información Científico - Técnica.
<http://www.cdict.uclv.edu.cu>
- Quipus Virtual: Biblioteca Virtual CDICT-UCLV.
<http://qvirtual.cdict.uclv.edu.cu>

- ADSI.FIE: La interfaz Web de administración de Usuarios en la Red UCLV.
<http://adsi.fie.uclv.edu.cu>
- Centro Agrícola: Sitio Oficial de la Revista.
<http://rca.agronet.uclv.edu.cu>
- Media.FIE: Sitio multimedia de la Facultad de Ing. Eléctrica.
<http://mp3.fie.uclv.edu.cu>
- Portal Colaborativo VRIP: Portal de postgrado de la UCLV.
<http://posgrado.uclv.edu.cu>
- VOIP.UCLV: Servicios de Voz sobre IP de la Red UCLV.
<http://voip.uclv.edu.cu>

Actualmente en la universidad se pretende mejorar los servicios actuales y añadir nuevos servicios para lograr una Comunidad Virtual en la Intranet Universitaria colaborando de esta forma con la informatización de la sociedad.

El grupo de redes de la UCLV <http://net.uclv.edu.cu> tiene como meta lograr que todos los servicios sean basados en tecnología Open Source debido a las ventajas que tiene en comparación con otras herramientas no libres.

Capítulo 2: Diseño y desarrollo de la Web de la Dirección de Economía y sus servicios.

2.1. Introducción.

La Dirección de Economía fue conectada a la Red UCLV mediante el nodo del Rectorado. Desde hace años contaba con la posibilidad del uso del correo electrónico, herramienta que hizo interesar a los trabajadores del área, el empleo de la computación en la comunicación de las operaciones registradas allí, que resultan de interés al resto de la UCLV.

Con el paso del tiempo se fueron formando grupos de investigación que trazaron líneas de trabajo para desarrollar proyectos Web que comenzaron a aparecer como servicios para la red.

Muchas de estas ideas fueron tareas asignadas por la dirección de la Universidad a cada facultad y a cada área en general. Gracias a eso ahora contamos con una Red que tiene su infraestructura Web bien implantada; mediante la Web <http://intranet.uclv.edu.cu> se conoce cuales son los servicios ofrecidos a la Intranet, los servicios Internacionales son accedidos desde <http://intranet.uclv.edu.cu>.

En la Dirección de Economía no existe un sitio Web donde brindar información referente a la gestión económica, financiera y contable, así como los reglamentos y las normas. También, por su parte, el AssetsNS solo realiza operaciones contables y no ofrece ninguna información que se pueda acceder desde la red si el sistema no está instalado.

Situación sin proyecto:

- No se puede conocer fácilmente cual es el consejo de dirección de la Dirección de Economía.
- Cada Jefe de Grupo en la Dirección de Economía es el encargado de brindarle información a las personas que la requieran, ya sea reglamentos, normas, etc.
- La información de los activos fijos, útiles y herramientas, etc. son informados mediante email lo que no garantiza seguridad, muchas veces son impresos para hacerle llegar a los directivos de cada área.
- Se le informa de cualquier modificación mediante email por lo que muchas veces las personas tienen información desactualizada.

Situación optimizada sin proyecto:

- Habilitar una página en el sitio de la Intranet donde se brinde información acerca de la dirección de economía.
- Habilitar un servicio FTP donde un responsable pueda actualizar los documentos como reglamentos, normas, etc.
- Asignar un responsable que se encargue de obtener resúmenes de los reportes de activos fijos, útiles y herramientas, etc. y publicarlos en documentos Excel o Word mediante FTP.

Situación con proyecto:

- Se dispone de un sitio Web centralizado con dirección única que permite publicar información referente a la dirección de Economía de manera amena e intuitiva.
- Se dispone de un directorio creado con una jerarquía de carpetas con acceso a los responsables a almacenar archivos que son publicados automáticamente en la Web.
- Existe la posibilidad de autenticarse con usuario y clave de del dominio UCLV para tener acceso automáticamente y dentro de la misma Web a los reportes tomados de la base de datos utilizada por el sistema AssetsNS.
- Se reduce el gasto de papel, cinta y tonel. Se aumenta la vida útil de las impresoras.

Por tales motivos se concibió una Web para la Dirección de Economía (DIRECO) donde se brinda información a los usuarios de la Intranet UCLV. Actualmente puede ser accedida mediante el enlace: <http://direco.uclv.edu.cu>.

2.2. Etapa de análisis y diseño.

Primeramente se pensó en el uso de herramientas Web de la tecnología de fuente abierta (Open Source) ya que es la más recomendada, además la meta de hoy en la Intranet UCLV es llegar a conseguir que la mayor cantidad de aplicaciones sean basadas en esta tecnología.

Dado lo anterior se llegó al acuerdo de utilizar el PHP como lenguaje de programación que es soportado por versiones libres del servidor Web Apache, pues junto al servidor de bases de datos MySQL logran mayor estabilidad, eficiencia y seguridad que las versiones del *Internet Information Services* (IIS) de Microsoft que era el servidor más utilizado en la UCLV. Hoy la mayoría de los servidores Web de la UCLV son Apache.

2.2.1. Diagrama conceptual del sitio Web.

A continuación se brinda el diagrama conceptual de la forma más general. Como se muestra, en los apartados Gestión, Reglamento y Control Interno aparecen representados los archivos que son de publicación automática en la Web por los responsables de cada área en la Dirección de Economía.

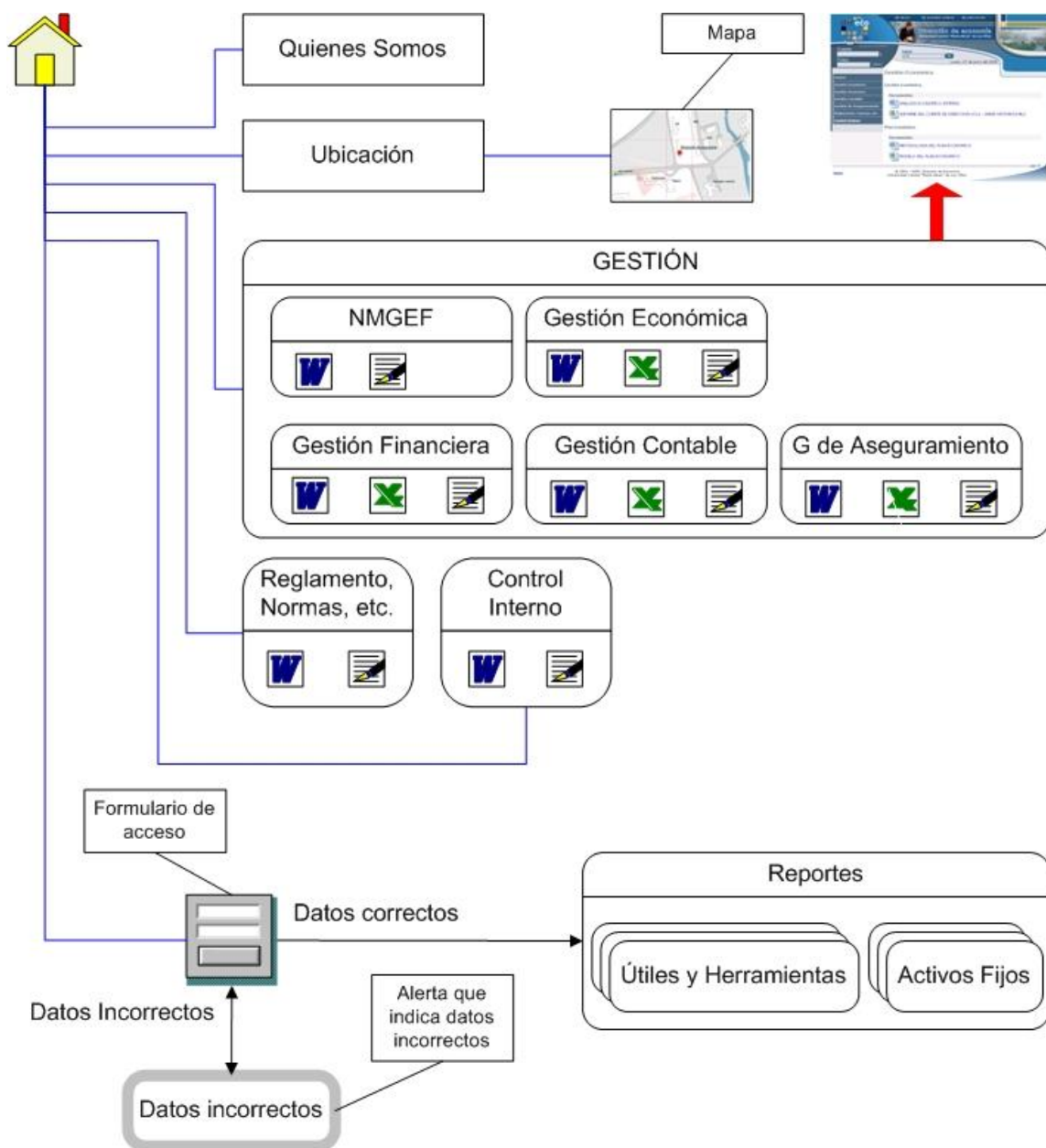


Figura 2. Diagrama conceptual del sitio Web.

2.2.2. Diseño de la base de datos:

Es necesario restringir los accesos a la información según el cargo pues sólo el responsable del área puede acceder a los listados de útiles y activos. Sólo el Consejo de Dirección puede acceder a cierta información clasificada por lo que es necesario aplicar métodos para restringir los accesos.

Para esto fue necesario el uso de una pequeña base de datos de tres tablas, conteniendo una de ellas el listado de los usuarios con permiso a los reportes.

[Usuario](

[Id_Usuario] [int] NOT NULL,
[Id_Grupo] [int] NOT NULL,
[Id_Area] [int] NOT NULL

)

[Grupo](

[Id_Grupo] [int] (1) NOT NULL,
[Nombre] [varchar] (255) NOT NULL,
[Reporte_ utiles] [int] (1) NOT NULL,
[Reporte_activos] [int] (1) NOT NULL,
[Reporte_dies_porcentaje] [int] (1) NOT NULL

)

[Area](

[Id_Area] [int] NOT NULL,
[Nombre] [varchar] (255) NOT NULL

)

En la tabla “Usuario” se almacena el nombre de usuario de la red UCLV y las llaves extrañas de “Grupo” y “Area”. La tabla “Grupo” permite almacenar qué reportes son a los que podrán tener acceso los usuarios pertenecientes a ese grupo. La tabla “Area” almacena el nombre del área universitaria donde pertenece el usuario.

2.2.3. Boceto estructural de los espacios en el diseño gráfico:

Como gran cantidad de los sitios Web actuales, la estructuración que se ha diseñado para la plantilla de la Web es la que se muestra en la figura 3.

Esta plantilla con logotipo, menú de presentación, formulario de acceso en la parte superior, etc. con el menú de contenidos a la izquierda y el pie de página permite mantener la misma estructura en cada

página visualizando en el centro el contenido a mostrar y permitiendo tener acceso a cada enlace principal desde todas las páginas.



Figura 3. Boceto estructural de los espacios en el diseño gráfico

En la sección donde aparece el formulario de acceso se muestra la información del usuario cuando este ha sido autenticado, también el enlace “Salir” el cual debe ser visible en todas las páginas mientras navega.

Este diseño ofrece la posibilidad de tomar todo el ancho de la página cuando sea necesario tomando el espacio del menú de contenidos, esto es importante ya que no se pierde la estructura y diseño una vez que se necesite introducir tablas grandes lográndose mayor espacio para colocar los contenidos.

2.3. Etapa de implementación.

2.3.1. Diseño y construcción de la Web.

El diseño de la Web (figura 4) se ha hecho utilizando el Macromedia Fireworks que permite hacer diseños profesionales de forma relativamente fácil si comparamos el trabajo con otros software profesionales de diseño como el Photoshop, Corel Draw, etc.

Una vez concluido el diseño se procede a “picarlo”, es decir, se divide el diseño de la Web en imágenes más pequeñas siguiendo el boceto estructural, el Foreworks da esta posibilidad. Las imágenes quedan en ficheros separados y almacenados en una carpeta llamada “images”.

El proceso de armado de la plantilla se hace ya dando uso al HTML, el Macromedia Dreamweaver es el software utilizado para esta tarea. Este proceso consiste en unir cada imagen para lograr una estructura de tablas en HTML que es finalmente lo que será publicado.



Figura 4. Diseño del sitio Web.

2.3.2. Módulo para la publicación de archivos.

Se ha desarrollado en PHP un módulo que permite leer una estructura de directorios en forma de árbol y publicar cada archivo que en ellas se encuentra automáticamente en la Web de la Dirección de Economía.

Los documentos pueden ser actualizados mediante el acceso a FTP o mediante una carpeta compartida de tal forma que un responsable por área pueda añadir, remplazar o eliminar archivos, estos se actualizan directamente y pueden ser accedidos, al instante, por interfaz Web. Los archivos quedan publicados en la Web como se muestra en la figura 5, o sea, se toma la extensión y se coloca la imagen correspondiente delante del enlace y el nombre del enlace es el mismo nombre del archivo sin la extensión.



Los documentos pueden ser abiertos directamente haciendo click encima de los enlaces o el icono, también pueden descargarse haciendo click derecho y seleccionar “salvar como”.

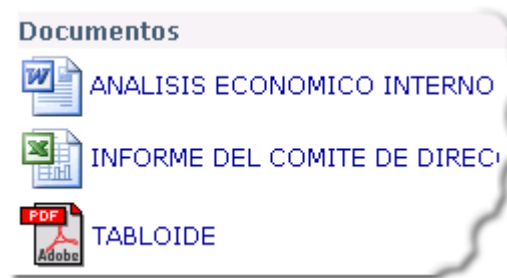


Figura 5. Publicación de artículos en la Web.

Código fuente:

El siguiente fragmento de código fuente es el encargado de leer los archivos contenidos en un directorio y generar el HTML que se mostrará en la sección de contenido dentro de la Web.

```
<table width="95%" border="0" align="center" cellpadding="2" cellspacing="0">
  <tr>
    <td colspan="2" bgcolor="#F0F0F0"><strong class="text_al2">Documentos</strong></td>
  </tr>
<?php
if ($handle = opendir($directorio)) {
  while (false != ($file = readdir($handle))) {
    if ($file != "." && $file != "..") {
      $ext = split('[.]', $file);
      ?>

      <tr>
        <td width="5%">
          <a href="<?php echo $directorio.$file ?>" target="_blank">
            
          </a>
        </td>
        <td width="95%">
          <a href="<?php echo $directorio.$file ?>" class="link_al2" target="_blank">
            <?php echo $ext[0] ?>
          </a>
        </td>
      </tr>

      <?
    }
  }
  closedir($handle);
}
?>
</table>
```

2.3.3. Reportes del AssetsNS.

El sistema AssetsNS registra los datos primarios de las operaciones contables y financieras. Es un sistema implementado en Visual Basic y con plataforma Microsoft SQL Server. Este sistema permite el control de los Útiles y Herramientas y los Activos Fijos que resultan de gran interés a los trabajadores encargados del cuidado de los mismos. Actualmente para realizar el conteo del 10% de los mismos es necesario imprimir los documentos para entregarlos personalmente a los responsables por lo que se impone la necesidad de visualizar las salidas del sistema a las áreas que lo necesiten.

El PHP tiene funciones primitivas para conectarse a servidores Microsoft SQL Server por lo tanto es posible acceder y obtener datos desde el PHP conectándose debidamente con un usuario con privilegios.

Existe un proyecto Open Source que se llama Eclipse, el mismo brinda un objeto (MSDatabase) tanto para establecer la conexión como para ejecutar sentencias SQL y obtener el resultado directamente en una variable de PHP la que más tarde puede generar el HTML y mostrarlo finalmente al visitante.

Para darle uso al Eclipse es necesario disponer de todos sus ficheros e incluir los archivos “MSDatabase.php” y “QueryIterator.php” dentro de la página PHP. El “MSDatabase.php” es requerido para establecer la conexión con el SQL Server y el “QueryIterator.php” se incluye para poder utilizar el objeto QueryIterator quien es el que va recorriendo la variable donde se almacena el registro completo de la consulta una vez se haga un *SELECT* a la base de datos.

Activos Fijos

No de inventario	Descripcion	Valor	Depreciac. Acumulada	Valor Actual	Resp
302216	ESTANTE	50	0	50	
195945	ARCHIVO METALICO	150	0	150	
196325	ESCAPARATE METALICO	110	0	110	
198684	BURO DE MADERA UNA TORRE	71.4	0	71.4	
198700	SILLA ESCOLAR DE MADERA	12	0	12	
198708	SILLA APILABLE ESCOLAR	12	0	12	
198709	SILLA APILABLE ESCOLAR	12	0	12	
198710	SILLA APILABLE ESCOLAR	12	0	12	
198712	SILLA APILABLE ESCOLAR	12	0	12	
198714	SILLA APILABLE ESCOLAR	12	0	12	
198746	SILLA APILABLE ESCOLAR	12	0	12	
198783	SILLA APILABLE ESCOLAR	12	0	12	

Figura 6. Ejemplo de reporte de los Activos Fijos

Los Activos Fijos, Útiles y Herramientas son los reportes que se muestran a los Decanos, Vice-decanos, Rector, Vice-rector y Administradores de las distintas áreas que hoy no tienen esa posibilidad. En la figura 6 se muestra un ejemplo de la tabla obtenida en el reporte Activos Fijos para lo cual fue necesario hacer una consulta uniendo 6 tablas mediante INNER JOIN.

Capítulo 3: Diseño y desarrollo del Sistema de Control del Pago del Estipendio Estudiantil.

En este capítulo se expone el estado actual del proceso del pago del estipendio estudiantil y como ha sido el desarrollo del Sistema de Control del Pago del Estipendio Estudiantil (SCPEE) así como lo referente a las etapas de análisis y de diseño.

3.1. Estado actual del proceso de estipendio y servicios Web de la Dirección de Economía.

En la Dirección de Economía de la UCLV se llevan a cabo todas las tareas para el control de los recursos económicos del centro. El pago del Estipendio Estudiantil es una de las labores que se realiza mensualmente por el personal que labora en dicha entidad. Este comprende el control del estudiantado autorizado a recibir un importe por concepto de estipendio a estudiante que varía según el año cursado por el mismo. El importe a devengar por concepto de Orden 18, aplicado a aquellos estudiantes que se acogieron a dicha Orden para hacer su entrada a la Universidad desde el Servicio Militar. Así como tener la relación de aquellos que tienen préstamos asignados y perciben un importe por este concepto.

El estipendio estudiantil en la Educación Superior es la cantidad de dinero en efectivo que se le entrega al alumno con carácter no reembolsable. A partir del curso 1993-94 los estudiantes becarios que recibían el servicio de albergue y comedor, junto a los estudiantes no becarios recibieron el estipendio para cubrir otras necesidades. Para ello se establecen normas que regulan dicho otorgamiento que quedan reflejadas en las resoluciones y cartas circulares vigentes hasta la fecha.

En las Facultades existen Comisiones de Subvenciones a Estudiantes integradas por el Decano o Vicedecano, el Administrador, un representante del área económica, uno de la Federación Estudiantil Universitaria y otros de la Unión de Jóvenes Comunistas.

Dicha comisión es la encargada de tramitar las solicitudes de estipendios de otras subvenciones y préstamos a estudiantes diurnos, normando los períodos que limitan su confección y entrega y manteniendo la constancia de recepción a cada solicitante. Deben, además, durante el año, recibir, analizar, tramitar, comunicar, informar y controlar las solicitudes de:

- el estipendio a alumnos externos matriculados en el curso 1993-94
- la ayuda especial no reintegrable por una sola vez a cada estudiante que haya estado matriculado
- la subvención a estudiantes sin amparo familiar

- la prolongación del tiempo reglamentario de estipendio a estudiantes que por razones docentes justificadas se vean precisados a permanecer en el centro de estudios, siempre y cuando no se encuentren vinculados a un centro laboral
- los préstamos estudiantiles reintegrables
- así como otras solicitudes relacionadas con estas cuestiones a la facultad

Esta comisión recibe, resuelve y comunica las reclamaciones, modificaciones y ajustes al estipendio de cada estudiante. Deben informar en tiempo y con los requisitos fijados, las bajas de estudiantes que perciben estipendio por algún concepto antes mencionado. Es el Administrador de la Facultad quien se encarga de tramitar todos los asuntos relacionados con el estipendio en la Dirección de Economía y a su vez mantener informados a los estudiantes.

Las decisiones de la Comisión de Subvención o de otras instancias del Centro sobre estipendios y becas son apelables ante el Rector mediante solicitud por escrito debidamente fundamentada del estudiante.

Según lo establecido por la Resolución 173 de 1994, tienen derecho a recibir el estipendio los becarios extranjeros, los estudiantes beneficiados con la Orden 18 de 1979 del Ministerio de las Fuerzas Armadas Revolucionarias, los estudiantes externos matriculados en el curso 1993-94, hasta la terminación de sus estudios, cuyo núcleo familiar perciba un ingreso per cápita mensual de \$50.00 o menos y los demás becarios de los centros de Educación Superior que hayan estado matriculados en el curso escolar 1993-94 hasta la terminación de sus estudios de pregrado o hasta el momento que causen baja por cualquier motivo.

Los estudiantes sin amparo familiar pueden ser beneficiados con una retribución adicional. En cambio, los trabajadores matriculados en los cursos diurnos que perciben salario o estipendio del Ministerio de Trabajo y Seguridad Social, no tienen derecho a recibir el estipendio estudiantil; así como tampoco, los estudiantes que reciben estipendio de otro organismo estatal o institución a menos que renuncien al estipendio antes asignado.

Relacionada con esta actividad está la definición de reintegros, la solicitud de los mismos por parte de los estudiantes para ser cobrado en el siguiente mes y el control del derecho a recibir el estipendio pues se penaliza con la pérdida del mismo a aquellos que dejan de cobrarlo por dos meses consecutivos o tres meses en el año. Según lo establece el artículo 21 de la Resolución 173, los

estudiantes están obligados a cobrar el importe de su estipendio o subvención mensual o ambos en las fechas establecidas.

Cuando un estudiante por razones debidamente justificadas deja de cobrar su estipendio ha de solicitar por escrito el abono del efectivo reintegrado que solo podrá tramitarse y pagarse en la fecha establecida para el siguiente mes.

En el curso 2002-2003 se extiende la Universalización de la Educación Superior a todos los municipios del país y a numerosas localidades, iniciándose nuevas carreras y otras ya abiertas. Los alumnos que en condiciones de profesores imparten docencia en las Sedes Universitarias Municipales (SUM) tienen una preparación adecuada en virtud de la cual fueron seleccionados y preparados para esta tarea. En la Resolución No. 104 de 2003 se regula el pago a estos estudiantes.

Actualmente en el Sistema Económico Contable AssetsNS, el pago a estudiantes se registra meramente como actividad contable debido a que no se tiene implementado módulo alguno que lo automatice, aunque en el caso del control del pago a trabajadores se utiliza un sistema implementado en Access que genera comprobante automáticos para ser introducidos en AssetsNS. Dicho sistema han sido certificados por el Ministerio de Educación Superior como lo establece la Resolución 340 del 2004.

Para realizar las nóminas del Estipendio Estudiantil se emplea un sistema en Access que no ha sido certificado y no permite el control real del pago a estudiantes pues sólo realiza el cálculo de las nóminas y solicitudes de fondo para el pago sin hacer tratamiento alguno a los reintegros, las reclamaciones de éstos, la pérdida del derecho al estipendio, en fin, todas las restantes operaciones.

Es de vital importancia para el buen desempeño del control del pago a estudiantes automatizar todas estas operaciones pues manualmente resultan difíciles y tediosas. Por ello se plantea la necesidad de este proyecto.

3.2. Etapa de análisis.

En esta etapa del desarrollo se estudia bien el proyecto, se analizan quienes son los actores que interactúan con el sistema y sus funciones, se especifican cuales son los procesos y módulos a implementar y se detallan con diagramas, se realiza el diseño de la base de datos, etc. Se puede afirmar que esta etapa es una de las más importantes en el desarrollo de un proyecto ya que todo esto forma la base para la implementación y es guía en cualquier momento.

Si el diseño de la base de datos se elabora sin tener presente todos los elementos, muy probablemente tendría que haber cambios cuando se idee algún proceso que no se había tenido en cuenta, esto implica que el código fuente deje de funcionar y por lo tanto se requiere de cambios también en la programación de los módulos.

3.2.1. Actores del sistema y sus funciones.

En el sistema tenemos los siguientes actores y sus tareas a cumplir en cada caso:

1. Administrador.

- Administrar el pago por facultades
 - Poner fecha de cobro y de reintegros por facultades.
 - Listar informes por año, especialidad y por cada mes (nóminas).
 - Revisar y marcar para aprobar el reintegro a los estudiantes que lo solicitaron mediante la secretaria de su facultad.
- Añadir estudiantes sin cobrar
- Cambiar el monto a pagar por cada concepto.
 - Modificar el pago por cada vía de ingreso del estudiante a la universidad.
 - Modificar el pago por cada año en curso del estudiante.
 - Modificar el pago por etapa de ayudantía.
 - Modificar cuanto es que se le paga a los estudiantes extranjeros.
- Actualizar los estudiantes bajas y altas.
- Administrar usuarios del sistema.

2. Secretarias.

- Administrar la ayudantía y el préstamo.
 - Listar estudiantes de la facultad a la que pertenece.
 - Insertar, modificar y quitar estudiantes con ayudantía y en que etapa se encuentra cada alumno ayudante.
 - Añadir, modificar y quitar estudiantes con préstamo y cuanto cobra cada uno.
- Solicitar cobros retroactivos.
- Solicitar reintegros.
 - Listar estudiantes que piden reintegro.
 - Solicitar reintegro.

3. Estudiantes.

- Listar fecha de cobro por facultades.

4. Sistema.

- Enviar correo a la secretaria cuando ésta ha sido dada de alta para utilizar el sistema.
- Enviar correos a los estudiantes recordándole que debe pedir reintegro si no ha cobrado en algún mes.
- Deshabilitar a los estudiantes que llevan dos meses consecutivos sin cobrar y/o tres meses no consecutivos.
- Enviar correo a la secretaria cuando el administrador ha declarado los estudiantes sin cobrar.
- Enviar correo a la secretaria cuando los pedidos de reintegro han sido cerrados.
- Activar y permitir que cobre el próximo mes el estudiante que ha pedido reintegro.
- Enviar correo a la secretaria cuando ésta ha sido dada de baja para utilizar el sistema.

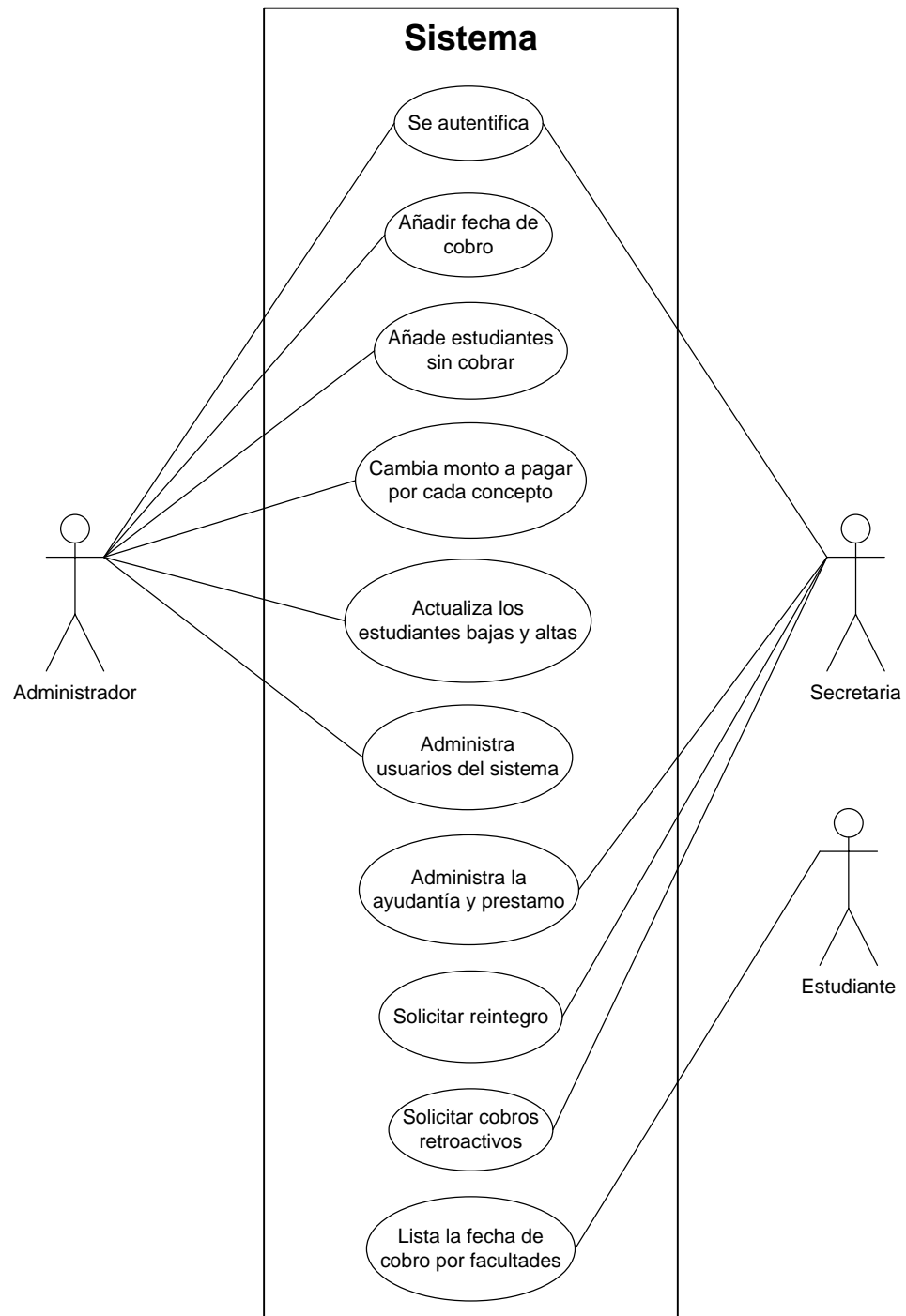


Figura 7. Casos de uso del sistema

3.2.2. Diagramas de Flujo de Datos.

Un Diagrama de Flujo de Datos (DFD) expresa el problema en forma de procesos, muestra los componentes activos del sistema y la interfaz entre ellos. Se utiliza para particionar un sistema por niveles de abstracción, es decir, usando un refinamiento sucesivo o enfoque “top-down” donde cada nivel es más detallado que el anterior [10]. A continuación se muestran dos DFD referentes al caso de uso “Añadir fecha de cobro”.

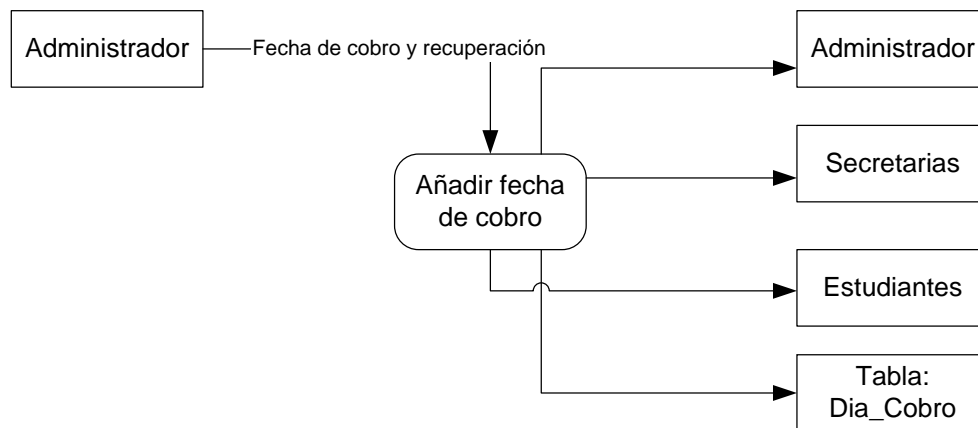


Figura 8. DFD de añadir fecha de cobro (nivel 1).

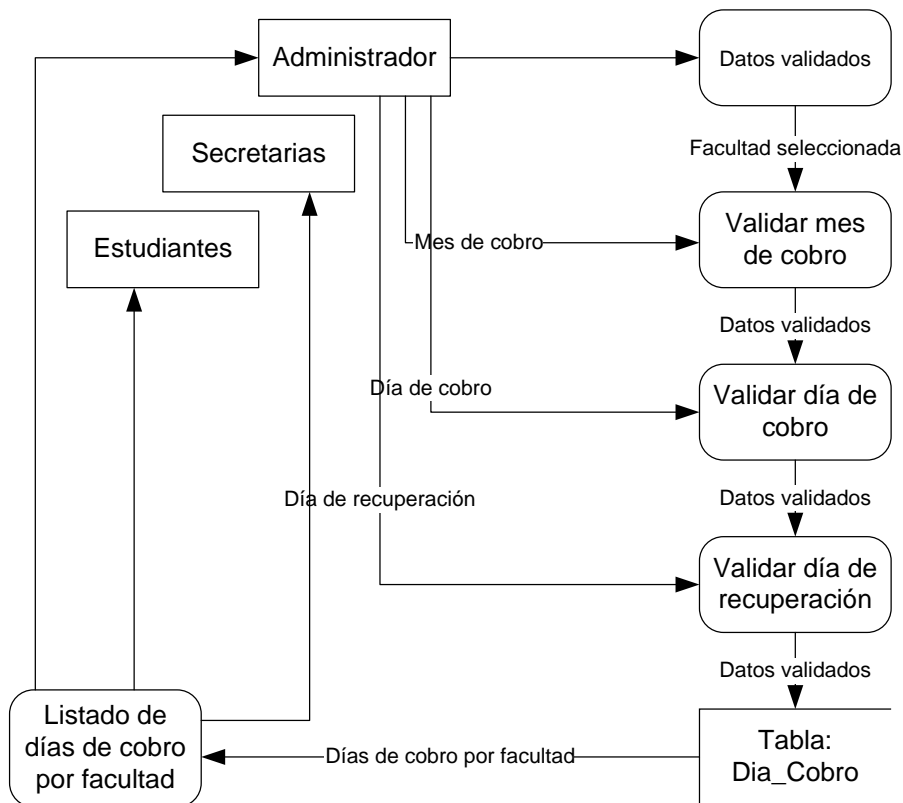


Figura 9. DFD de añadir fecha de cobro (nivel 2).

3.2.3. Diseño de la base de datos.

Para el diseño de la base de datos se ha tomado en cuenta los siguientes requisitos puntuales:

- Almacenar sólo la información necesaria
- Solicitar sólo lo necesario y ser explícito
- Normalizar las estructuras de tablas
- Seleccionar el tipo de dato apropiado
- Utilizar índices apropiadamente

Se ha adoptado también determinadas conversiones de nombres a tener en cuenta a la hora de crear las tablas y campos, de esa forma se gana en uniformidad y entendimiento. A continuación se especifican algunas de ellas:

- Utilizar caracteres alfanuméricos.
- Limitar los nombres a menos de 64 caracteres (es una restricción de MySQL).
- Utilizar el guión bajo (_) para separar palabras.
- Utilizar palabras en minúsculas para los nombres de las tablas (es una restricción de MySQL).
- Utilizar palabras que comiencen con mayúsculas para los nombres de los campos de las tablas.
- Utilizar las letras Id al comienzo de un campo que sea llave tanto en la tabla como llave extranjera.
- En una tabla, colocar primero la clave primaria seguida de las claves extranjeras.
- Los nombres de los campos deben ser descriptivos de su contenido.
- Los nombres de los campos deben ser unívocos entre tablas, en excepción de las claves.

3.2.3.1. Estructura.

En el modelado del sistema es necesario tener en cuenta como van a estar estructurado los datos; para ello se piensa en cuales datos leer automáticamente desde la base de datos del estudiantes administrada por el sistema de matrículas de las secretarías docentes de las facultades, cuáles serán introducidos por la persona responsable del estipendio y cuáles serán introducidos por las secretarías directamente en el sistema.

Haciendo un estudio profundo de la base de datos de estudiante, se ha podido ver que datos como: Carnet de Identidad, año, nombre, apellidos, facultad, especialidad, vía de ingreso, vicedecano, entre otros, pueden ser obtenidos automáticamente y que los restantes datos son introducidos “a mano” mediante el sistema de estipendio.

A continuación se ofrece la estructura de la base de datos:

[Cargo](

[Id_Cargo] [int] (1) NOT NULL,
[Descripcion] [varchar] (30) NOT NULL

)

[Cobro_Ano](

[Ano] [int] (1) NOT NULL,
[Base] [float] NOT NULL,
[Orden_18] [float] NOT NULL

)

[Cobro_Ayudantia](

[Etapas] [varchar] (1) NOT NULL,
[Cantidad] [float] NOT NULL

)

[Cobro_Estudiante_Extranjero](

[Cantidad] [float] NOT NULL

)

[Cobro_Retroactivo](

[Id_Estudiante] [varchar] (6) NOT NULL,
[Concepto] [text] NOT NULL,
[Cantidad] [float] NOT NULL
[Id_Dia_Cobro] [varchar] (6) NOT NULL,
[Estado] [int] (1) NULL

Llave extranjera de Estudiante

)

[Dia_Cobro](

[Id_Dia_Cobro] [int] NOT NULL,
[Curso] [varchar] (8) NOT NULL,
[Mes] [int] (2) NOT NULL,
[Fecha_Cobro] [date] NOT NULL,
[Fecha_Recuperacion] [date] NULL,
[Id_Facultad] [varchar] (2) NOT NULL,
[Estado_Pago] [int] (1) NOT NULL

Llave extranjera de Facultad

)

[Especialidad](

[Id_Especialidad] [int] (6) NOT NULL,
[Id_Facultad] [varchar] (2) NOT NULL,
[Nombre_Especialidad] [varchar] (50) NOT NULL

Llave extranjera de Facultad

)

[Estudiante](

[Id_Estudiante] [varchar] (6) NOT NULL,
[CI] (varchar) (11) NOT NULL
[Ano] [int] (1) NOT NULL,
[Nombre] [varchar] (25) NOT NULL,
[Apellido1] [varchar] (25) NOT NULL,
[Apellido2] [varchar] (25) NOT NULL,
[Id_Facultad] [varchar] (2) NOT NULL,
[Id_Especialidad] [varchar] (6) NOT NULL,
[Id_Via_Ingreso] [int] NULL ,
[Id_Pais] [int] (3) NOT NULL,
[Activo] [int] (1) NOT NULL

Llave extranjera de Facultad
Llave extranjera de Especialidad
Llave extranjera de Via_Ingreso

)

[Estudiante_Ayudante](

[Id_Estudiante] [varchar] (6) NOT NULL,
[Etapa] [char] (2) NOT NULL

Llave extranjera de Estudiante

)

[Estudiante_Prestamo](

[Id_Estudiante] [varchar] (6) NOT NULL,
[Cantidad] [float] NOT NULL

Llave extranjera de Estudiante

)

[Estudiante_Tmp](

[Id_Estudiante] [varchar] (6) NOT NULL,
[CI] (varchar) (11) NOT NULL
[Ano] [int] (1) NOT NULL,
[Nombre] [varchar] (25) NOT NULL,
[Apellido1] [varchar] (25) NOT NULL,
[Apellido2] [varchar] (25) NOT NULL,
[Id_Facultad] [varchar] (2) NOT NULL,
[Id_Especialidad] [varchar] (6) NOT NULL,
[Id_Via_Ingreso] [int] NOT NULL,
[Id_Pais] [int] NOT NULL,
[Activo] [int] (1) NOT NULL

Llave extranjera de Facultad
Llave extranjera de Especialidad
Llave extranjera de Via_Ingreso

)

[Facultad](

[Id_Facultad] [varchar] (2) NOT NULL,
[Nombre] [varchar] (25) NOT NULL,
[Decano] [varchar] (25) NULL,
[Vice_Decano] [varchar] (25) NULL,

```

[Sec_Tec] [varchar] (25) NULL
)
[No_Cobro](
    [Id_Dia_Cobro] [int] NOT NULL,                Llave extranjera de Dia_Cobro
    [Id_Estudiante] [varchar] (6) [int] NOT NULL,    Llave extranjera de Estudiante
    [Fecha_Estudiantes_No_Cobro] [date] NULL,
    [Fecha_Estudiantes_Reintegro] [date] NULL,
    [Reintegro] [int] (1) NULL
)
[Usuario](
    [Id_Usuarios] [varchar] (30) NOT NULL,
    [Id_Cargo] [int] (1) NOT NULL,                Llave extranjera de Cargo
    [Id_Facultad] [varchar] (2) NULL,              Llave extranjera de Facultad
    [Nombre] [varchar] (255) NOT NULL
)
[Via_Ingreso](
    [Id_Via_Ingreso] [int] (2) NOT NULL,
    [Descripcion] [varchar] (30) NOT NULL,
    [Cantidad] [int] NOT NULL
)

```

3.2.3.2. Normalización de tablas.

La base de datos anteriormente planteada ha sido diseñada con cuidado de tal forma que se logre consistencia en la información, o sea, integridad entre los datos, objetivo que se plantea para valorar o no si la base de datos cumple con las formas normales establecidas, o lo que es lo mismo, si la base de datos se encuentra normalizada. Los principales objetivos de la normalización, que según [13] estos son:

- Controlar la redundancia de la información.
- Evitar pérdidas de información.
- Capacidad para representar toda la información.
- Mantener la consistencia de los datos.

Cada una de las tablas cumple con la primera forma normal, debido a que cuentan con un atributo o más que forman la clave que relacionarán datos que son únicos entre sí. Se le ha dado solución, por ejemplo, a la tabla “*Estudiante*” que unido a la tabla “*Estudiante_Ayudante*” mediante una llave extranjera (atributo “*Id_Estudiante*”) a la tabla “*Estudiante*” se mantienen los cobros en la tabla

correspondiente sin necesidad de repetir toda la información del estudiante (“Nombre”, “Apellido1”, “Apellido2”, etc.) si el estudiante es alumno ayudante.

Las tablas se encuentran también en segunda forma normal ya que cada campo depende por sí sólo de la llave o clave de la tabla. En el caso de la tabla “Usuario” (donde se representa el identificador del usuario dentro del sistema) y “Cargo” (donde se representa el cargo que ocupa la persona) se ha tenido que hacer en dos tablas aunque se obtengan ambas con dos atributos solamente, debido a que el campo “Descripción” no depende de su totalidad de la llave “Id_Usuario” porque se puede dar el caso que exista dos o más usuarios con el mismo cargo dentro del sistema.

También podemos decir que se encuentra en tercera forma normal debido a que cada atributo de las tablas depende únicamente de la llave y no de otros atributos dentro de la misma tabla. Por ejemplo: la información del estudiante y de la facultad se han separado en dos tablas distintas y existe una llave común que indica a que facultad pertenece el estudiante, al igual ocurre con la especialidad y la vía de ingreso en el caso del ejemplo.

Se cumple igualmente la cuarta forma normal porque para cualquier combinación llave – campo podemos encontrar que no existirán valores duplicados debido a que cada llave es única y no existen dos valores distintos para el atributo llave. Podemos poner por ejemplo que nunca se dará el caso de que aparezca el estudiante repetido porque un estudiante sólo pertenece a una facultad, también podemos ver que en una fecha determinada el estudiante cobra una sola vez, el estudiante ha ingresado una sola vez a la universidad (al menos se toma la última vez ingresada), etc. se cumple para todas las tablas.

3.2.3.3. Elementos de datos.

A continuación se especifica cada elemento de dato utilizado en la base de datos, se encuentran organizados por orden alfabético del nombre de las tablas.

Nombre del Elemento de Dato: Id_Cargo

Tabla donde aparece: Usuario; Cargo.

Valores y Significado: Número consecutivo. Llave de la tabla Cargo.

Nombre del Elemento de Dato: Descripción

Tabla donde aparece: Cargo

Valores y Significado: Texto que describe cual es el cargo que ocupa del usuario en su puesto de trabajo común.

Nombre del Elemento de Dato: Ano

Tabla donde aparece: Cobro_Ano; Cobro_Orgen_18; Estudiante.

Valores y Significado: [1 | 2 | 3 | 4 | 5 | 6]. Año que el estudiante esta cursando en estos momentos.

Nombre del Elemento de Dato: Base

Tabla donde aparece: Cobro_Ano

Valores y Significado: Cantidad de dinero que cobra el estudiante por cada año cuando la vía de ingreso es la normal.

Nombre del Elemento de Dato: Orden_18

Tabla donde aparece: Cobro_Ano

Valores y Significado: Cantidad de dinero que cobra el estudiante por cada año cuando la vía de ingreso es Orden 18.

Nombre del Elemento de Dato: Etapa

Tabla donde aparece: Cobro_Ayudantia; Estudiante_Ayudante

Valores y Significado: Etapa de ayudantía existentes.

Nombre del Elemento de Dato: Cantidad

Tabla donde aparece: Cobro_Ayudantia, Cobro_Estudiante_Extranjero, Cobro_Retroactivo; Estudiante_Prestamo; Via_Ingreso

Valores y Significado: Cantidad de dinero que cobra el estudiante.

Nombre del Elemento de Dato: Id_Estudiante

Tabla donde aparece: Cobro_Retroactivo; Estudiante; Estudiante_Ayudante; Estudiante_Prestamo; No_Cobro; Estudiante_Tmp, No_Cobro.

Valores y Significado: Código que se le asigna a cada estudiante cuando es matriculado. Tiene 6 caracteres que pueden ser números y letras mayúsculas, siendo siempre el primer carácter una inicial que significa históricamente la facultad. A para Agropecuaria, C Construcciones, E Eléctrica, I Empresariales, M mecánica, Q Química, F MFC, H Humanidades, P Psicología, S Sociales, D Derecho.

Nombre del Elemento de Dato: Concepto

Tabla donde aparece: Cobro_Retroactivo.

Valores y Significado: Concepto por el cual el estudiante cobra la retroactividad.

Nombre del Elemento de Dato: Id_Dia_Cobro

Tabla donde aparece: Cobro_ Retroactivo; Dia_Cobro; No_Cobro.

Valores y Significado: Identifica el día del cobro en el sistema.

Nombre del Elemento de Dato: Estado

Tabla donde aparece: Cobro_ Retroactivo.

Valores y Significado: [0 | 1]. Estado en que se encuentra la retroactividad, si es "0" esta inactivo y si es "1" esta activo o aprobado por el administrador para que se sume en el cálculo del cobro final.

Nombre del Elemento de Dato: Curso

Tabla donde aparece: Dia_Cobro.

Valores y Significado: [20042005 | 20052006 | ...] Curso actual del día de cobro con formato "aaaabbbb" donde "aaaa" es el año en que comienza el curso y "bbbb" es el año en que culmina el curso escolar.

Nombre del Elemento de Dato: Mes

Tabla donde aparece: Dia_Cobro.

Valores y Significado: Mes en que será el cobro.

Nombre del Elemento de Dato: Fecha_Cobro

Tabla donde aparece: Dia_Cobro.

Valores y Significado: Fecha en que será el cobro.

Nombre del Elemento de Dato: Fecha_Recuperacion

Tabla donde aparece: Dia_Cobro.

Valores y Significado: Fecha en que será la recuperación.

Nombre del Elemento de Dato: Estado_Pago

Tabla donde aparece: Dia_Cobro.

Valores y Significado: [0 | 1 | 2]. Cuando es "0" indica que el pago no ha sido efectuado, "1" indica que el pago ha sido cerrado y cuando es "2" indica que los pedidos de reintegro van a ser mostrados en el próximo cobro para ser aprobados o no.

Nombre del Elemento de Dato: Id_Especialidad

Tabla donde aparece: Estudiante; Especialidad.

Valores y Significado: Código que identifica la especialidad dentro del sistema.

Nombre del Elemento de Dato: Id_Facultad

Tabla donde aparece: Dia_Cobro; Especialidad; Estudiante; Facultad; Usuario.

Valores y Significado: [101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 | 111 | 112]. Código que identifica la facultad establecido en la UCLV.

Nombre del Elemento de Dato: Nombre_Especialidad

Tabla donde aparece: Especialidad.

Valores y Significado: Nombre de la especialidad.

Nombre del Elemento de Dato: CI

Tabla donde aparece: Estudiante.

Valores y Significado: Número de carné de identidad del estudiante.

Nombre del Elemento de Dato: Nombre

Tabla donde aparece: Estudiante.

Valores y Significado: Nombre del estudiante.

Nombre del Elemento de Dato: Apellido1

Tabla donde aparece: Estudiante.

Valores y Significado: Primer apellido del estudiante.

Nombre del Elemento de Dato: Apellido2

Tabla donde aparece: Estudiante.

Valores y Significado: Segundo apellido del estudiante.

Nombre del Elemento de Dato: Id_Via_Ingreso

Tabla donde aparece: Estudiante; Via_Ingreso.

Valores y Significado: Código que identifica la vía de ingreso dentro del sistema.

Nombre del Elemento de Dato: Id_Pais

Tabla donde aparece: Estudiante.

Valores y Significado: Código que identifica el país dentro del sistema, el número “192” es el correspondiente con Cuba.

Nombre del Elemento de Dato: Activo

Tabla donde aparece: Estudiante.

Valores y Significado: [0 | 1]. Dice si el estudiante está activo o no en el sistema para poder cobrar.

Nombre del Elemento de Dato: Nombre

Tabla donde aparece: Facultad.

Valores y Significado: Nombre de la facultad.

Nombre del Elemento de Dato: Decano

Tabla donde aparece: Facultad.

Valores y Significado: Nombre(s) y apellidos del decano o decana de la facultad.

Nombre del Elemento de Dato: Vice_Decano

Tabla donde aparece: Facultad.

Valores y Significado: Nombre(s) y apellidos del vice-decano o vice-decana de la facultad.

Nombre del Elemento de Dato: Sec_Tec

Tabla donde aparece: Facultad.

Valores y Significado: Nombre(s) y apellidos del secretario técnico o secretaria técnica de la facultad.

Nombre del Elemento de Dato: Fecha_Estudiantes_No_Cobro

Tabla donde aparece: No_Cobro.

Valores y Significado: Fecha en que el administrador ha declarado qué estudiantes no cobraron.

Nombre del Elemento de Dato: Fecha_Estudiantes_Reintegro

Tabla donde aparece: No_Cobro.

Valores y Significado: Fecha en que el estudiante ha pedido el reintegro.

Nombre del Elemento de Dato: Reintegro

Tabla donde aparece: No_Cobro.

Valores y Significado: [0 | 1]. Indica si el reintegro ha sido aprobado (1) o no (0).

Nombre del Elemento de Dato: Id_Cargo

Tabla donde aparece: Usuario.

Valores y Significado: Cargo que ocupa el usuario en su puesto de trabajo.

Nombre del Elemento de Dato: Nombre

Tabla donde aparece: Usuario.

Valores y Significado: Nombre del usuario.

Nombre del Elemento de Dato: Descripcion

Tabla donde aparece: Via_Ingreso.

Valores y Significado: Descripción de la vía de ingreso.

3.2.3.4. Diagrama modelo de la base de datos.

En el siguiente diagrama se muestran las tablas y sus llaves indicadas por “PK”, las llaves foráneas o extranjeras son indicadas por “FK”. Los elementos obligatorios o no nulos se encuentran en negrita. Las flechas indican la relación de una tabla con otra mediante su llave foránea.

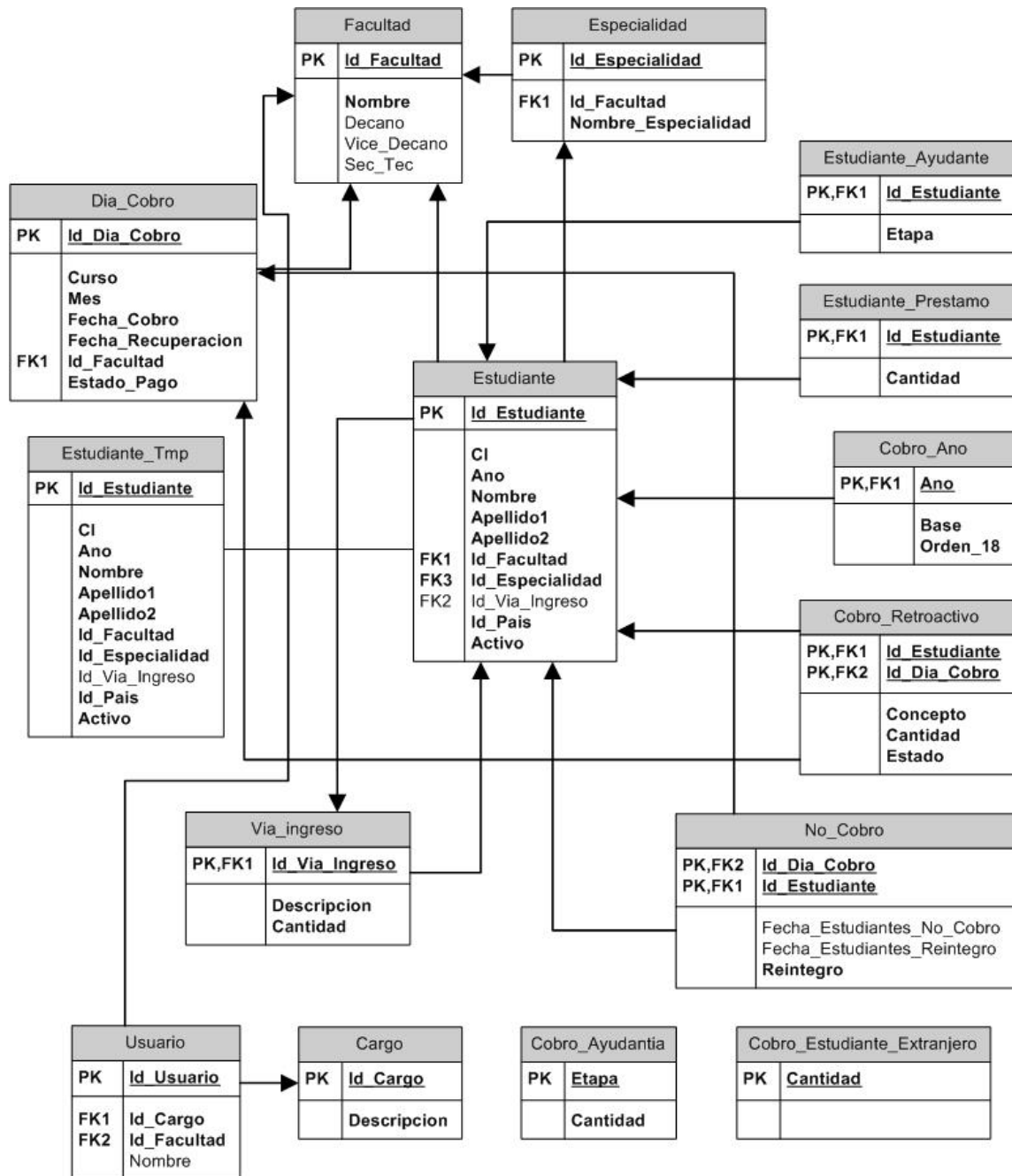


Figura 10. Diagrama modelo de la base de datos

3.3. Etapa de diseño.

Basado en la etapa de análisis se concibe el desarrollo del proyecto. En este epígrafe se explica a grandes rasgos cómo ha sido la implementación, así como las herramientas utilizadas.

Se explica la solución que se brinda para obtener mayor seguridad y accesibilidad en el sistema, puntos que se entienden como muy necesarios a tenerlos en cuenta debido a que la aplicación es mediante Web y las herramientas que brinda el HTML y son muy escasas.

3.3.1. Garantizar seguridad en el acceso mediante usuario y clave.

Sólamente el área económica es quien puede tener acceso al módulo de administración y de la misma forma, las secretarías de cada facultad son las únicas personas que pueden tener acceso al módulo de secretarías. También hay que tener en cuenta que cada secretaria tiene acceso sólo a controlar los estudiantes de la facultad a la que pertenece.

Para garantizar el correcto acceso mediante usuario y clave se ha llegado a la conclusión de que la manera más factible es que el sistema acepte los respectivos usuarios del dominio UCLV. Para tomar dicha decisión se ha tenido en cuenta que los usuarios que laborarían con el mismo son personas de media experiencia en el uso de la computación, por lo tanto el uso de la propia clave de acceso diario a su computadora de trabajo le ayudaría en gran medida. Con ello se resuelve el problema de las claves olvidadas, lo que puede traer consigo la posibilidad del uso indebido del sistema por parte de intrusos, utilizando la Ingeniería Social u otros medios para apoderarse de las claves. Si el usuario cambia la clave ya se vería reflejado en el acceso al sistema con la nueva clave, también si por algún motivo el usuario es bloqueado por los administradores, éste no podría entrar al sistema tampoco, se puede ver como un medio de autenticación en tiempo real.

3.3.1.1. Uso de número de Turing:

Se prevé el uso del número de Turing para garantizar que en cada acceso lo que “esté” del lado del cliente sea un humano y no un algoritmo computacional (una máquina) intentando hacer pruebas para lograr el acceso, lo que finalmente se lograría tras una cantidad determinada de intentos aleatorios o consecutivos en la clave y mucho más si es conocido el nombre del usuario.

Para lograr ésto se ha utilizado funciones del PHP como `imagecreate()`, `imagecopyresized()`, `imageline()`, entre otras, que permiten generar imágenes en tiempo de ejecución y cargarlas directamente en el HTML mediante la etiqueta ``. Con el uso de 2 líneas horizontales y 6 verticales en posiciones aleatorias, también aleatoriamente se obtiene la posición de los números cada

vez que se carga la página (teniendo en cuenta un rango para que siempre sean visibles y no se superpongan ni se “corten” ocultándose por los extremos). De la misma forma aparecen los números sin importar el orden y se le aplica una función que obtiene un número aleatorio con una distribución normal. El resultado es una imagen como la que se muestra en la figura 11 en el campo “número de seguridad”.

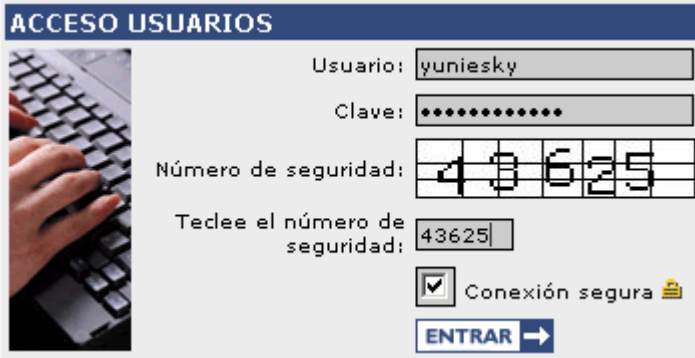


Figura 11. Formulario de acceso al sistema

Dicho “número de seguridad” es visto por el visitante y debe ser tecleado en la casilla donde se indica, para así verificar que quien tenemos al lado del cliente, es un humano. Al cargar la página se genera una variable sesión en el servidor que contiene un número encriptado, el mismo que es visible al usuario mediante la imagen “sucia” y que se le pide para luego encriptar y comparar con el número guardado.

Una vez validado el número de seguridad es cuando se prosigue a verificar los datos de usuarios y clave en el dominio UCLV del “Active Directory” donde el uso de funciones para el trabajo con el Protocolo de Acceso a Directorios Ligeros (LDAP) juegan un papel importante ya que desde la misma aplicación podríamos acceder con usuario y clave del dominio UCLV.

3.3.1.2. Uso del SSL:

Como se describe en el Capítulo 1, el SSL es un protocolo de conexión que garantiza la seguridad de los datos a la hora de ser enviados al servidor Web. En el caso del SCPEE se le da uso a la hora de hacer acceder con usuario y clave en el sistema, dichos datos son enviados mediante este protocolo cuya certificación de seguridad la garantiza el servidor Web del servicio ADSI.FIE.

3.3.2 Accesibilidad.

En el capítulo 1 se describe la importancia que tiene disponer de un producto que sea accesible por la mayoría de los usuarios. En el caso del desarrollo del SCPEE se ha tomado la accesibilidad como algo

a tener en cuenta a la hora de disponer de un sistema que sea rápido, intuitivo y, lo más importante, que sea de fácil manejo, ya que estamos hablando de algo que sería utilizado a diario por una misma persona y muy frecuentemente por un grupo de personas. No se trata de una Web donde se tendrá que ver la posibilidad de que la utilice un discapacitado o personas con diferentes configuraciones del navegador ni personas que tienen navegadores obsoletos. Estamos hablando de obtener un producto final donde pueda ser utilizado por personas de experiencia media y con configuraciones estándares de conexión y configuraciones del Software a utilizar, que en este caso puede ser el navegador Internet Explorer versión 6 o superior, de tal forma que se pueda utilizar funciones de JavaScript con las cuales se logre mayor interactividad usuario – navegador, y por lo que es lo mismo, usuario - sistema.

3.3.2.1. Uso del JPSpan:

El JPSpan es un proyecto Open Source que se basa en brindar herramientas que permitan de una manera relativamente fácil ejecutar una función de PHP desde el JavaScript y obtener el resultado en una variable de JavaScript. No utiliza la técnica del IFRAME (fragmentos de páginas dentro de un mismo escenario o vista Web divididas mediante código HTML) oculto para recargar JavaScript desde el servidor, sino que utiliza el objeto XMLHttpRequest, que es una manera nativa de acceder a páginas Web como objetos desde JavaScript.

Esta tecnología es utilizada por el servicio de correo del Google (Gmail.com) donde se obtienen resultados sorprendentes. Debido al fácil manejo y rapidez, permitiendo mejor accesibilidad y usabilidad, ha tenido gran popularidad.

En el caso de nuestro trabajo, le hemos dado utilidad a esta herramienta para hacer búsquedas en la base de datos y encontrar los Carné de Identidad de los estudiantes a medida que se vaya escribiendo en el campo edit, al encontrar resultados se listan sugerencias a las cuales se le hace click y se auto-completa en el edit.

Buscar estudiantes sin cobrar:

Carnet de Identidad:

Sugerencia 1: [81111211529](#) (Yoany González Sosa)
 Sugerencia 2: [81112612825](#) (Jorge Coello Mena)
 Sugerencia 3: [81110828789](#) (Alejandro Hernández Nápoles)
 Sugerencia 4: [81110710081](#) (Mario Alejandro Moreira Martínez)
 Sugerencia 5: [81110617942](#) (Sain Salazar Martínez)
 Sugerencia 6: [81111516149](#) (Mijail And Saralain Figueredo)
 Sugerencia 7: [81110212012](#) (Marian Cepero Martínez)
 Sugerencia 8: [81110312417](#) (Click para completar con 81111516149)

Esto es una forma de validar el carné de identidad introducido antes de que se le de click en “Añadir” por lo tanto el usuario puede ver si existe un estudiante con el carné de identidad tecleado y estar seguro de que es quien ha tecleado debido a que también se muestra su nombre en cada sugerencia.

La misma técnica es utilizada al dar click en “Añadir”, los dígitos tecleados se envían sin que se refresque la página y el resultado (“El estudiante se ha añadido” o “El estudiante había sido añadido”) es mostrado directamente en el mismo lugar donde se muestran las sugerencias ya que ambos son el resultado del contenido modificable de la etiqueta “<div id=“results”></div>” a la cual mediante JavaScript se le pueden asignar etiquetas HTML que más tarde el navegador las interpretará y mostrará dinámicamente al usuario.

Añadir retroactividad:

Carnet de identidad del estudiante:

Sugerencia 1: [81090916286](#) (Juan Manuel Medero Martínez)
 Sugerencia 2: [81092211022](#) (Libernys Valdés Vera)
 Sugerencia 3: [81090314280](#) (Yoicel Rodríguez Martínez)

Concepto:
 ¿A qué se debe este pago retroactivo?

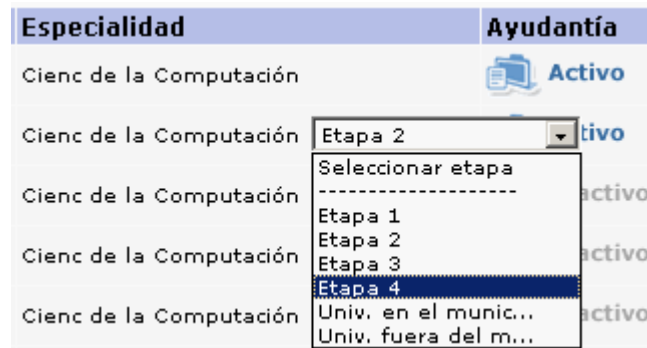
Todo esto es posible ya que de esta manera se puede ejecutar, desde el navegador, funciones PHP en páginas remotas que se encuentran en el servidor y el resultado obtenerlo en variables de JavaScript.

3.3.2.2. Uso etiquetas DIV:

Mediante el uso de la etiqueta “<div></div>” y asignándole la propiedad “style=‘position: absolute;’” se obtiene una “capa flotante” en la página Web, esto combinando con propiedades de visibilidad “style=‘visibility: hidden;’” y modificando su posición con respecto a las coordenadas del puntero se

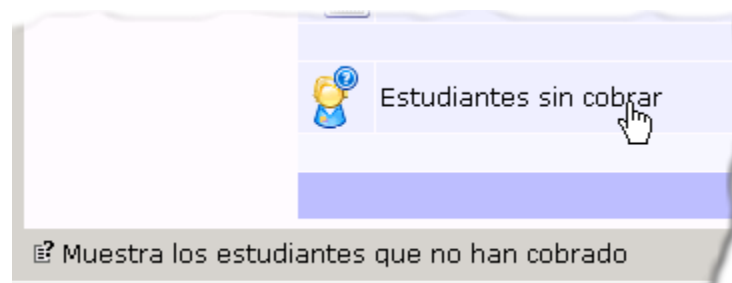
obtuvo un resultado sorprendente. Al añadir o modificar un estudiante con ayudantía se muestra en la posición del cursor un campo con la etapa actual, el cual permite modificarla.

Esta funcionalidad mejora sustancialmente la accesibilidad y hace más cómodo e intuitivo el uso del sistema.



3.3.2.3. Mensajes en la barra de estado:

Se ha ideado una barra de estado mediante marcos que permite mostrar un mensaje en cualquier momento tanto cuando la página sea muy larga y aparezca la barra de desplazamiento vertical como cuando se muestra en pantalla completa.



Se muestra un mensaje cuando se posiciona el puntero encima de cualquier elemento que al darle click ejerce una acción en el sistema. Esta funcionalidad ayuda al actor ya que es informado de lo que ocurrirá si hace click mejorando la accesibilidad, uso del sistema y por lo tanto se aprovecha al máximo la potencialidad del mismo.

Esto puede ser posible con el uso del Javascript, se crea una zona *div* en el marco inferior cuyo código HTML es remplazado por un texto que se le pasa como parámetro a la función encargada (estado) de modificarle la propiedad *innerHTML*, proceso que es efectuado en tiempo de ejecución y mostrado activamente en el navegador Web.

```

<script>
function estado(texto){
    parent.frame_estado.document.getElementById("div_estado").innerHTML = texto;
}
</script>

```

3.3.3. Diseño de la interfaz.

Para el diseño de la interfaz se utilizó el color azul por ser un color refrescante a la vista, además un diseño sencillo y agradable con pocos elementos pesados como son imágenes “grandes”.

La plantilla está dividida por zonas cuadradas con una estructura estándar para todas las páginas.

Los enlaces en el menú están acompañados por íconos que identifican la acción, ícono que aparece junto al nombre de las páginas correspondiente.

Fue creada una hoja de estilos con 33 estilos utilizados en todo el proyecto para dar colores a las letras, hipervínculos, formulario, bordes, fondos, etc. que posibilita una fácil actualización en caso de que se desee cambiar algún tipo de letra, tamaños o colores al diseño actual.

Figura 12. Pantalla capturada del Sistema de Control del Pago de Estipendio

3.3.4. Actualización automática de estudiantes.

En el SCPEE se utiliza una tabla en la base de datos que es la que almacena el listado de todos los estudiantes que actualmente están estudiando de la universidad.

En el sistema no se implementó una herramienta para eliminar a los estudiantes bajas ni para adicionar estudiantes de nuevo ingreso, en su lugar se desarrolló un servicio que permite obtener automáticamente los datos de los estudiantes, desde la base de datos del Sistema de Control de Estudiantes utilizado actualmente por todas las secretarías y se mantiene actualizado constantemente.

Lo anterior pudo ser posible ya que (al igual que el AssetsNS) este Sistema de Control de Estudiantes utiliza un servidor Microsoft SQL Server para almacenar los datos. Utilizando el módulo Eclipse podemos acceder al SQL Server y ejecutar consultas a la base de datos.

Quien tiene esta funcionalidad en el Panel de Control es el administrador del sistema, o sea la persona encargada de llevar el control del pago del estipendio en la Dirección de Economía decidiendo en cada momento qué estudiantes actualizar, tanto los bajas como los estudiantes de alta.

3.3.4.1. Implementación del servicio:

Para la implementación de este servicio se ha creado una tabla temporal en la base de datos del SCPEE que tiene la misma estructura que la tabla de estudiantes de forma tal que se pueda recuperar todos los datos necesarios en esa tabla y luego listar los estudiantes bajas y altas en páginas distintas para que el administrador seleccione los que desea actualizar.

Actualización de los estudiantes



[Leer estudiantes actuales](#) | [Estudiantes de alta](#) | [Estudiantes de baja](#)


Esta funcionalidad ha sido necesaria ya que en la primera versión del sistema no se almacenan los nombres de los estudiantes dados de baja y si algunos de ellos tienen el derecho de cobrar reintegro o por algún motivo tenga derecho de cobro retroactivo, ya no saldría en la nómina.

Recuperando la tabla de estudiantes:

Al hacer click en el enlace “Leer estudiantes actuales” el sistema le hace un *TRUNCATE* a la tabla “Estudiante_Tmp”, lee los datos requeridos desde la base de datos del Sistema de Control de Estudiantes y los almacena en esa tabla. Se notifica al usuario que han sido leídos los estudiantes y los lista.

Actualizando los estudiantes de alta:

Al hacer click en “Estudiantes de alta” aparece una página con los estudiantes nuevos que no existen en la base de datos local (figura 13), los estudiantes pueden ser seleccionados y añadidos a la base de datos del SCPEE automáticamente.

 **Estudiantes de alta:**

CI	Nombre	Facultad	Año	
84082312572	Lucrines Azcuy Aguilera	Ciencias Sociales	3 ^{ro}	<input type="checkbox"/>
87120512134	William Andres Martinez Herrera	Ciencias Sociales	1 ^{er}	<input type="checkbox"/>

Marcar: [Todos](#) | [Ninguno](#)

Añadir los marcados

Figura 13. Ejemplo del listado de estudiantes de alta

Actualizando los estudiantes de baja:

Ocurre lo mismo que con los estudiantes de alta, sólo que cuando se trata de los estudiantes dados de baja éstos aparecen en color rojo en lugar del verde y al seleccionarlos son eliminados de la tabla de estudiantes automáticamente.

Capítulo 4: Manual de usuario del Sistema de Control del Pago del Estipendio Estudiantil.

4.1. Características del software.

El sistema es programado utilizando tecnología Web, para lo cual fue necesario el uso de los lenguajes HTML, JavaScript y PHP. Cuenta con dos módulos principales, uno utilizado por la persona encargada en la Dirección de Economía y el otro por la persona asignada por cada facultad (generalmente la Secretaria Técnica) para informar a la Dirección de Economía el estado de cada estudiante; ya sea el cobro por ayudantía, reintegros, pagos retroactivos, etc.

Requiere una base de datos de 15 tablas, la cual es utilizada para almacenar los datos necesarios para controlar el proceso de forma estable.

La información de cada estudiante como: nombre y apellidos, facultad, especialidad, año, vía de ingreso, etc. son datos obtenidos o exportados del Sistema de Control de Estudiantes de la Facultad cuya base de datos se encuentra ubicada en los servidores centrales del Nodo Puerta de la Intranet UCLV.

4.1.1. Requerimientos mínimos.

El sistema debe ser instalado en un servidor Web Apache por lo tanto es necesario disponer de una de sus versiones, existen versiones para Windows. Igualmente existen versiones para Windows del MySQL, servidor de bases de datos también requerido.

Para poder instalar correctamente el sistema, es necesario tener versiones del PHP4 o superior (incluido las últimas versiones del Apache) y cualquier versión del MySQL (aunque se recomienda la última para ganar en estabilidad y eficiencia). Estos servidores pueden ser descargados de los respectivos sitios en Internet de forma gratuita ya que forman parte del proyecto de fuente abierta.

Como se trata de una aplicación Web, los clientes pueden ser estaciones conectadas a red con cualquier versión de Windows y con el Internet Explorer 6 o superior ya que se utilizan funcionalidades que otras versiones de navegadores no las implementa.

4.2. Instalación.

Para llevar a cabo la instalación del sistema se deben seguir los siguientes pasos en cualquier orden:

- Copiar los archivos para un directorio que pueda ser accedido mediante HTTP de tal forma que sea interpretado por el PHP.
- Crear una base de datos y ejecutar el código SQL contenido en el fichero “scpee.sql” que es el encargado de generar las tablas y datos para que el sistema pueda funcionar.
- Abrir el archivo “tconnection.php” ubicado en el directorio “_SCPEE/modulos/” y modificar las líneas 13, 14, 15 y 16. Indicar el nombre o IP servidor de base de datos MySQL (localhost si es el mismo del servidor Web), indicar el usuario, contraseña y nombre de la base de datos a utilizar.

Al realizar los pasos anteriores y solicitar la página mediante HTTP en el navegador Web deber mostrarse una vista como la de la figura 12 del capítulo 3.

4.3. Descripción del módulo administrativo.

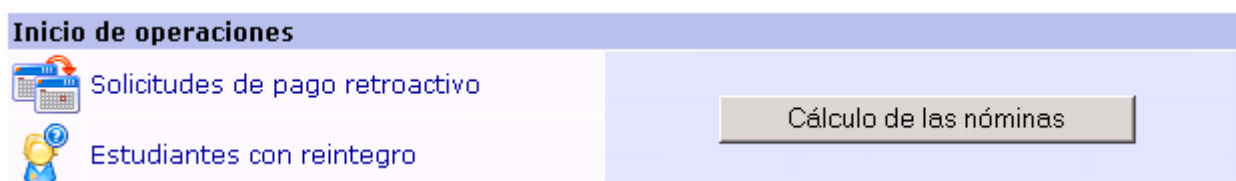
El módulo administrativo brinda la posibilidad de controlar el proceso referente al pago estudiantil: permite asignar fecha de cobro por facultad – mes, imprimir las nóminas ya calculadas, ver y aprobar solicitudes de pago retroactivo, estudiantes con reintegro, entre muchas otras funcionalidades.

Añadir fecha de cobro:

Para añadir la fecha de cobro a una facultad vaya a “*Días de cobro y nóminas*” y seleccione el mes de cobro que se quiera añadir, se le presentará el formulario para seleccionar la fecha de cobro y la fecha de recuperación. Al seleccionar las fechas de cobro y recuperación se muestra la tabla con los meses del curso con el mes que se le ha adicionado la fecha de cobro en negrita.

Cálculo de nóminas:

Para obtener las nóminas de un mes determinado vaya a “*Días de cobro y nóminas*” y seleccione el mes de cobro al que se le quiera calcular las nóminas, el mes tiene que estar en negrita porque de lo contrario tendría que añadir fecha de cobro.



Hay que verificar los procesos correspondientes a “Inicio de Operaciones”:

- Vaya a “Solicitud de pago retroactivo”, si aparece algún estudiante debe aprobarlo o no.
- Vaya a “Estudiantes con Reintegro” y le aparecerá los estudiantes que han solicitado reintegro.



Reintegro cobro anterior

No aprobados [Listar los aprobados](#)

Cantidad: 4

CI	Año		Aprobar reintegro
82061622647	4 ^{to}	José Jorge Lorenzo Vila	No solicitado
82071012640	4 ^{to}	Damny Magdaleno Guevara	<input checked="" type="checkbox"/>
82051515644	4 ^{to}	William Abel Alvarez Martínez de la Coter	No solicitado
82031215622	4 ^{to}	Ernesto Sarduy Alonso	No solicitado

Marcar: [Todos](#) | [Ninguno](#)

Aprobar los marcados

[« Ir al inicio de la página de este pago](#)

Nótese que estas dos operaciones no son necesarias cuando se trata del mes de septiembre.

Luego de haber verificado estos dos puntos haga click en “Cálculo de las nóminas” y se mostrarán las nóminas por año.

Cerrar pago:

Cuando se hayan calculado las nóminas se muestra un botón en la parte inferior de la página del día de cobro. Al presionar este botón se activa el mes de cobro como pagado y se muestra una sección denominada “Estudiantes sin cobrar”.

Añadir estudiantes sin cobrar:

Cuando el pago ha sido cerrado se muestra la sección “Estudiantes sin cobrar” en la página del día de cobro. En el apartado “Estudiantes sin cobrar”, se debe teclear el carné de identidad en la casilla correspondiente.

Buscar estudiantes sin cobrar:

Carnet de Identidad:

Sugerencia 1: [81111211529](#) (Yoany González Sosa)
 Sugerencia 2: [81112612825](#) (Jorge Coello Mena)
 Sugerencia 3: [81110828789](#) (Alejandro Hernández Nápoles)
 Sugerencia 4: [81110710081](#) (Mario Alejandro Moreira Martínez)
 Sugerencia 5: [81110617942](#) (Sain Salazar Martínez)
 Sugerencia 6: [81111516149](#) (Mijail And Saralain Figueredo)
 Sugerencia 7: [81110212072](#) (Marian Cepero Martínez)
 Sugerencia 8: [811103124](#)


Nótese que al teclear 4 números serán buscados estudiantes que coincidan, si se le da click en el carné de identidad de algún estudiante que haya aparecido, se completará en la casilla y no necesariamente hay que escribirlo completo.

Declarar estudiantes sin cobrar y cierre de petición de reintegros:

Los estudiantes que han sido añadidos como estudiantes sin cobrar sólo son listados a las secretarías una vez hayan sido declarados.

Para declarar estudiantes por cobrar: haga click en el botón “Declarar estudiantes sin cobrar” del apartado “Sincronización con las secretarías”.

Para cerrar peticiones de reintegros por las secretarías: haga click en el botón “Cerrar petición de reintegros” del apartado “Sincronización con las secretarías”. Las secretarías no podrán solicitar más reintegros.

 **Sincronización con las secretarías**

Al presionar el siguiente botón la secretaria de la facultad verá los estudiantes que no han cobrado:

Al presionar el siguiente botón la secretaria de la facultad no podrá solicitar más reintegros:

Importante: Esto último es importante hacerlo antes de imprimir las nóminas del siguiente mes de cobro.

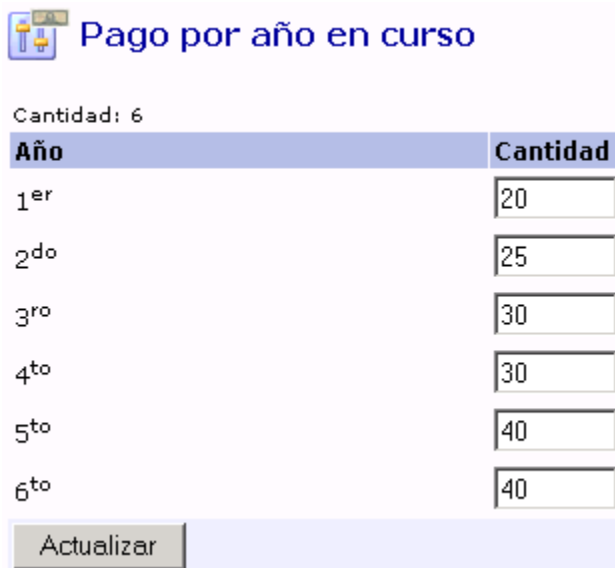
Listar y eliminar estudiantes sin cobrar:

Esta funcionalidad es necesaria para verificar los estudiantes añadidos como estudiantes sin cobrar. En la página del día de cobro, en el apartado “Estudiantes sin cobrar”, haga click en “Listar estudiantes

sin cobrar”, se mostrará un listado de estudiantes sin cobrar, para eliminar uno de ellos haga click en el enlace “Quitar” a la derecha de cada estudiante.

Cambiar el importe a pagar por concepto:

En el Panel de control en sección de enlace “Pago por:” seleccione el concepto de pago que desee modificar. Se le mostrará el formulario correspondiente al concepto de pago seleccionado, modifique los datos y haga click en “Actualizar”.



Pago por año en curso

Cantidad: 6

Año	Cantidad
1 ^{er}	<input type="text" value="20"/>
2 ^{do}	<input type="text" value="25"/>
3 ^{ro}	<input type="text" value="30"/>
4 ^{to}	<input type="text" value="30"/>
5 ^{to}	<input type="text" value="40"/>
6 ^{to}	<input type="text" value="40"/>

Listar usuarios del sistema:

En el Panel de control haga click en el enlace “Usuarios del sistema”, se mostrará el listado de todos los usuarios del sistema tanto el (los) administrador (es) como las secretarias especificando la facultad a la que pertenece y el cargo que ocupa en su puesto de trabajo.

Actualizar estudiantes:

Para actualizar los estudiantes en el Panel de control haga click en “Actualizar la base de datos”, se mostrará una página con tres enlaces como se muestra a continuación.

Actualización de los estudiantes

 [Leer estudiantes actuales](#) | [Estudiantes de alta](#) | [Estudiantes de baja](#)

Antes que todo haga click en “Leer estudiantes actuales”, cuando se muestren los estudiantes haga click en “Regresar” y luego en “Estudiantes de alta”, se mostrará una página como se indica más adelante con los estudiantes que han sido dados de alta desde la última vez que el sistema se ha actualizado.

Estudiantes de alta:

CI	Nombre	Facultad	Año	
84082312572	Lucrines Azcuy Aguilera	Ciencias Sociales	3 ^{ro}	<input type="checkbox"/>
87120512134	William Andres Martinez Herrera	Ciencias Sociales	1 ^{er}	<input type="checkbox"/>

Marcar: [Todos](#) | [Ninguno](#)

Añadir los marcados

Para añadir algún estudiante debe seleccionarlo y hacer click en “Añadir los marcados”.











Para actualizar los estudiantes dados de baja se hace la misma operación pero esta vez seleccionando la opción “Estudiantes de baja”.

4.4. Descripción del módulo para secretarias.

Añadir estudiantes con ayudantía y préstamo:

Para añadir estudiantes con ayudantía haga click en “Listar todos los estudiantes” desde el Panel de control, seleccione el enlace “Mostrar estudiantes” del año correspondiente al año en que se encuentra el estudiante, se muestra el listado de estudiantes del año.

En la columna “Ayudantía” aparece el texto “Activo” iluminado en caso que el estudiante ya tenga ayudantía asignada, si aparece el texto “Inactivo” es porque el estudiante no tiene asignada ayudantía. Para añadir o modificar la ayudantía haga click en dicho texto, se mostrará un campo para seleccionar la etapa, seleccione la etapa y luego haga click en “Aceptar”.

Nombre	Especialidad	Ayudantía	Préstamo
Yuniesky Abreu Collazo	Cienc de la Computación	 Inactivo	 Inactivo
Yania Alemán Rivas	Cienc de la Computación	 Inactivo	 Inactivo
Michel Alonso Gutiérrez	Cienc de la Computación	 Inactivo	 Inactivo
Arian Alonso González-Abreu	Cienc de la Computación	 Activo	 Inactivo
Frey Eduardo Alvarez González	Cienc de la Computación	 Inactivo	 Inactivo

Para añadir préstamo haga la misma operación sólo que esta vez en la columna de “Préstamo”.

Añadir y quitar estudiantes con ayudantía y con préstamo:

En el Panel de control, sección “Estudiantes con:” haga click en “Con ayudantía”, seleccione el enlace “Mostrar estudiantes” del año correspondiente al año en que se encuentra el estudiante, se muestra el listado de estudiantes del año.



Estudiantes con ayudantía:

Pago por ayudantía: 5 año

Cantidad: 10

Buscador interno:

CI	Nombre	Especialidad	Etapas	
79091812340	Yuniesky Alemán Caballero	Cienc de la Computación	<input type="text" value="Etapa 2"/>	Quitar
81032412448	Tomy Bolaños De la Coba	Cienc de la Computación	<input type="text" value="Etapa 3"/>	Quitar
81013013104	Yusmaikel Cabrera Marin	Cienc de la Computación	<input type="text" value="Etapa 2"/>	Quitar
81112612825	Jorge Coello Mena	Cienc de la Computación	<input type="text" value="Etapa 2"/>	Quitar
81030415301	Yaser Couce Sardiñas	Cienc de la Computación	<input type="text" value="Etapa 3"/>	Quitar
<input type="button" value="Actualizar etapas"/>				

Cada estudiante tiene asignada una etapa de ayudantía, esta puede ser modificada y haciendo click en “Actualizar etapas” y son modificados en la base de datos.

Lo mismo ocurre con préstamo, sólo que seleccionando en el Panel de control el enlace “Con préstamo”.

Listar peticiones de pago retroactivo:

En el Panel de control, sección “Cobros retroactivos” haga click en “Listar todos”, seleccione el enlace “Mostrar estudiantes” del año correspondiente al año en que se encuentra el estudiante, se muestra el listado de estudiantes del año.

Se muestran las peticiones pendientes y aprobadas.

Añadir petición de pago retroactivo:

En el Panel de control, sección “Cobros retroactivos” haga click en “Añadir”, se mostrará un formulario “Añadir solicitud de pago retroactivo”. Se teclean los datos requeridos y se presiona en Añadir.



Cobro retroactivo

Añadir solicitud de pago retroactivo:

Carnet de identidad del estudiante:

Concepto:

¿A qué se debe este pago con carácter retroactivo?

Cantidad:

Monto total a pagar.

pesos

Mes a pagar:

Noviembre (próximo pago)

Añadir

Cancelar

Observe que el mes a pagar seleccionado por defecto es el mes de próximo cobro, en caso de no haber sido asignado todavía una fecha para el mes del próximo cobro no es posible adicionar retroactividad ya que la secretaria debe conocer exactamente que mes será pagado y hacer la petición.

Solicitud de reintegro:

En el Panel de control haga click en “Estudiantes sin cobrar”, seleccione el enlace “Mostrar estudiantes” del año correspondiente al año en que se encuentra el estudiante, se muestra el listado de estudiantes del año que no ha cobrado durante el curso.



Estudiantes sin cobrar:

Curso 2004-2005

Estudiantes: 2^{do} año

Cantidad: 2

CI	Nombre	Especialidad	Estado	Reintegro
Noviembre				
84051513307	Yunier Rojas Garcia	Cienc de la Computación	Pedido	En espera de aprobación...
84090512186	Fernando Ibargollin Fajardo	Cienc de la Computación	Pedir reintegro	Ignorado

Para hacer una solicitud de reintegro haga click en el enlace “Pedir reintegro” correspondiente al estudiante que lo solicita.

Los estados de reintegro pueden ser: Ignorado (cuando el reintegro no ha sido solicitado), En espera de aprobación, o Aprobado.

Conclusiones:

- La Dirección de Economía cuenta con un sitio Web visible desde la Intranet UCLV que le permite a las dependencias publicar información de interés a la Comunidad Universitaria.
- Se obtuvo un software que cumple con lo establecido en las resoluciones vigentes (173/94, 10/2000 y 104/2003) que rigen el control del estipendio estudiantil.
- El sistema obtenido regula las bajas y los nuevos ingresos automáticamente.
- El Sistema de Control del Estipendio Estudiantil agiliza el proceso de envío y recepción de información entre las áreas y la Dirección de Economía.
- Las tecnologías de fuente abierta pueden utilizarse en actividades propias de la Universidad mejorando los servicios de la Intranet Universitaria.

Recomendaciones:

- Vincular el Sistema de Control del Pago de Estipendio al Sistema AssetsNS automatizando el proceso de contabilización.
- Hacer cumplir las normas establecidas para que el sistema pueda ser certificado.
- Obtener reportes de la base de datos del sistema AssetsNS con el fin de publicar los útiles y herramientas y activos fijos solo a los directivos institucionales.
- Las actualizaciones de la tabla estudiante debe hacerse automáticamente en el inicio de las operaciones antes de calcular las nóminas.