

Universidad de Cienfuegos “Carlos Rafael Rodríguez”

Facultad de Ingeniería

Carrera de Ingeniería Informática

Trabajo de diploma para optar por el título de
Ingeniero en Informática



*Sistema Informático para la gestión de la información
académica en la Cátedra Universitaria del Adulto Mayor en la
Universidad de Cienfuegos*

Daniel Villazón Pérez

Tutores:

Ing. Gabriel Alejandro López López

Ing. Cinthya Rodríguez Hernández

Cienfuegos, Cuba
2012 - 2013

Declaración de autoría

Yo, Daniel Villazón Pérez declaro que soy el único autor del trabajo de diploma titulado: "FÉNIX: Sistema informático para la gestión de la información académica en la Cátedra Universitaria del Adulto Mayor en la Universidad de Cienfuegos." y autorizo a la Facultad de Ingeniería en la Universidad de Cienfuegos "Carlos Rafael Rodríguez", para que hagan el uso que estimen pertinente del presente trabajo de diploma.

Para que así conste firmo (firmamos) la presente a los ____ días del mes de ____ del ____.

Firma Autor

Los abajo firmantes certificamos que el presente trabajo ha sido revisado según acuerdo de la dirección de nuestro centro y el mismo cumple los requisitos que debe tener un trabajo de esta envergadura referente a la temática señalada.

Firma Tutor

Firma Tutor

Firma ICT

Firma Vicedecano

***Aprender sin pensar es inútil. Pensar sin
aprender, peligroso.***

Agradecimientos

A toda mi familia y en especial a mis padres por todo el amor, confianza y ayuda prestada.

A mi novia Laura por amarme, comprenderme en mis largas horas de estudio y por su inmensa paciencia para tolerar mi estrés.

A mi amigo José Carlos, quien me apoyó en los momentos que de verdad lo necesitaba.

A mis tutores: a Cinthia por insistirme en que terminara el documento y ayudarme a confeccionarlo, a Gabriel por prestarme su tiempo y por guiarme por el camino del “código”.

A todo aquel que confió en mí.

A todos aquellos que hicieron posible la confección y elaboración de este trabajo.

Muchas Gracias

Dedicatoria

A toda mi familia...

*...y especialmente a mis padres por ser el pilar fundamental en todo
lo que soy.*

Resumen

La presente investigación lleva como título: “FÉNIX: Sistema informático para gestión de la información académica en la Cátedra Universitaria del Adulto Mayor” y se realizó en la facultad de Ingeniería de la Universidad de Cienfuegos “Carlos Rafael Rodríguez”. El software que se propone está basado en la arquitectura cliente/servidor, mediante el empleo de tecnologías web cumpliendo con las necesidades reales de los clientes y usuarios finales. FÉNIX pretende mejorar la gestión de matrícula de estudiantes, memorias de eventos, talleres y otros tipos de actividades que se desarrollen en las CUAM, contribuyendo a que toda la información esté accesible, reduciendo el tiempo en las búsquedas de información y permitiendo la gestión de reportes de interés. A través del documento de la investigación quedan descritos los elementos que conforman el análisis, diseño e implementación del sistema propuesto, siguiendo lo establecido por la metodología FDD. Para la implementación del mismo se utilizó MySQL como Sistema Gestor de Bases de Datos, Apache como servidor web, PHP como lenguaje de programación y el framework Yii para apoyar el desarrollo.

Abstract

The current research entitled: “FÉNIX: a Computer system to manage the academic information in the University Working Groups for the Elderly People” (CUAM) was carried out in the Faculty of Engineering at the University of Cienfuegos “Carlos Rafael Rodríguez”. The software proposed is based on the client /server architecture through the use of web technologies to meet the real needs of clients and final users. FÉNIX intends to improve the management of information regarding the students’ registration, events, workshops and other kinds of activities developed in the CUAM, and thus to contribute to have all the information always accessible, reducing the time when searching information and allowing the management of reports. Throughout the research paper, the elements that constitute the analysis, design and implementation of the system proposed are described, following the FDD methodology. For its implementation, MySQL as Database Management System, Apache as web server, PHP as programming language and Yii framework to support the development were used.

Índice

Introducción.....	1
Capítulo1: Fundamentación teórica.....	9
1.1 Principales conceptos asociados al dominio del sistema.....	9
1.2 Sistemas existentes vinculados al campo de acción.....	10
1.3 Conceptos, patrones, aplicaciones, tecnologías, lenguajes y bibliotecas utilizados en el desarrollo de software	13
1.4 Lenguajes de Programación utilizados	16
1.5 Descripción del sistema propuesto	23
1.6 Conclusiones.....	23
Capítulo2: Análisis y diseño del sistema informático.....	24
2.1 Desarrollo de un modelo global.....	24
2.2 Flujo actual de los procesos involucrados en el campo de acción.....	24
2.2.1 Modelo de Dominio	25
2.2.2 Construcción de una lista de funcionalidades.....	26
2.3 Planeación por funcionalidad	34
2.4 Diseño por funcionalidad.....	38
2.4.1 Esquema de la Base de Datos	39
2.5 Descripción de la arquitectura propuesta.....	39
2.6 Diagrama de Despliegue.....	41
2.7 Conclusiones.....	41
Capítulo3: Estudio de factibilidad y validación de la solución.....	42
3.1 Estudio de la factibilidad	42
3.1.1 Planificación.....	42
Puntos de Función Ajustados	48
3.1.2 Productividad en el desarrollo de una aplicación	48
3.1.3 Esfuerzo para el desarrollo de una aplicación	49
3.1.4 Costo de la aplicación.....	49
3.2 Beneficios tangibles e intangibles.....	49
3.3 Análisis de costos y beneficios.....	50

3.4 Validación de la solución.....	50
3.4.1 Resultados de la aplicación de la encuesta.....	50
3.5 Conclusiones.....	54
Conclusiones.....	55
Recomendaciones.....	56
Referencias Bibliográficas.....	57
Bibliografía	58
Anexos	61
Anexo A: Encuesta.....	61

Índice de Tablas

Tabla 1 Primer Bloque de funcionalidades	27
Tabla 2 Segundo Bloque de funcionalidades	29
Tabla 3 Tercer Bloque de funcionalidades	29
Tabla 4 Cuarto Bloque de funcionalidades.....	30
Tabla 5 Quinto Bloque de funcionalidades	30
Tabla 6 Bloque de funcionalidades del Módulo de Administración.....	32
Tabla 7 Planeación por funcionalidad.	38
Tabla 8 Archivos lógicos internos.....	43
Tabla 9 Entradas externas	43
Tabla 10 Salidas externas.....	44
Tabla 11 Consultas externas.....	46
Tabla 12 Puntos de función brutos.....	47
Tabla 13 Factor de ajuste.....	48
Tabla 14 Resumen del estudio de factibilidad.	49
Tabla 15 Utilidad como sistema de gestión académica en la CUAM	51
Tabla 16 Utilidad en el apoyo al trabajo administrativo de la CUAM	51
Tabla 17 Existencia de un software similar	52
Tabla 19 En cuanto al uso respecto a otros sistemas	52
Tabla 18 Facilidad de uso con respecto a otros sistemas	52
Tabla 20 En cuanto a la presentación	53
Tabla 21 En que radican las ventajas	53
Tabla 22 Valoración del sistema	53

Índice de Figuras

Figura 1: Fases de Desarrollo de la metodología FDD	15
Figura 2: Diagrama del modelo de objetos.....	25
Figura 3: Diagrama de la Base de Datos	39
Figura 4: Estructura de la capa modelo.....	40
Figura 5: Diagrama de despliegue	41

Introducción

El envejecimiento de la población es uno de los fenómenos demográficos más importantes de finales de siglo XX y será protagonista del siguiente, pues traerá profundas modificaciones en las estructuras sociales, económicas y culturales. Se considera un fenómeno social más, dentro los muchos existentes y posibles, el cual puede ser científica y empíricamente aprendido.

“Las primeras y más influyentes aproximaciones a la vejez como fenómeno social coinciden con el inicio de la década de 1950, bajo el claro dominio de los paradigmas funcionalistas y conductistas, y en respuesta a una serie de repercusiones sociales que venían emergiendo en el marco del desarrollo de los países industrializados, especialmente los Estados Unidos (como la jubilación obligatoria, el predominio de la familia nuclear, el impacto de los procesos de industrialización y urbanización, y la creciente movilidad social y geográfica). Estas aproximaciones resultaron en un debate que sigue siendo fuente de controversias: la cuestión del envejecimiento y la adaptación social.”[1]

En el mundo moderno, el criterio generacional indica la existencia de una permanente lucha o relación de conflicto potencial entre generaciones, que en la actualidad tiende a resolverse a favor de los más jóvenes y en detrimento de las generaciones más viejas, que se ven relegadas a los últimos puestos sociales. Sin embargo, siguiendo la lógica analítica de este enfoque, el proceso de envejecimiento no es inmutable ni está fijo, sino que varía a lo largo y dentro de cada cohorte a medida que la sociedad cambia, que la sociedad establece los límites cronológicos que separan los grupos de edad y que establece valoraciones positivas y negativas por razón de pertenencia a estos grupos, por lo tanto es un error (frecuente en la gerontología) pensar que quienes hoy son jóvenes experimentarán la vejez de la misma forma en que la experimentan los actuales viejos, ya que será otra la sociedad que dicte pautas, norme y valore la vejez.[1]

En la larga historia de vida de los adultos mayores han sido muchos los cambios que han tenido que enfrentar: transitar de una etapa de vida a otra, protagonistas

de interacciones familiares primero, sociales y laborales después; que han ido conformando y consolidando la personalidad de los ancianos que se expresa en modos de pensar y actuar muy enraizados.

Para la comunidad de adultos mayores se hace difícil asimilar los cambios en el arte, la cultura, el pensamiento y la vida social. Esta es una realidad difícil de comprender y no es la etiqueta de “intolerantes” la que se le debe poner sino analizar que es característico de la que están viviendo la resistencia al cambio.

A pesar de esto, los adultos mayores continúan proyectándose al futuro con una esperanza de vida incrementada, tratando no solo de entender el mundo de hoy sino de implicarse en él de manera activa y aplicando su sabiduría en las acciones que realiza.

Estas transformaciones ya son experimentadas por el mundo desarrollado y, si bien Latinoamérica tiene la ventaja de aprender de sus experiencias, algunas circunstancias nos diferencian de estos países, tanto en lo que atañe al proceso de envejecimiento como a la estructura social y económica que enfrentará las consecuencias de este proceso.

El nivel de envejecimiento que Europa logró en dos siglos lo alcanzará América Latina en apenas cincuenta años lo que significa que tendrá menos tiempo para adaptar sus sistemas al nuevo escenario de una población con mayor vejez.[2]

El área del Caribe actualmente es la parte más envejecida de América Latina y el Caribe (ALC) y cuenta con el mayor porcentaje de personas de 75 y más años (3.4% en el 2010). Para el 2050, estos alcanzarán casi el 10% del total poblacional y cerca del 40% del total de adultos mayores. En el área más joven (América Central), se estima que entre el 2025 y 2050, el ritmo de crecimiento de dicho grupo superará al estimado para el resto de la región e incluso al de Norteamérica, mientras que Cuba destaca como el país con mayor discrepancia entre poblaciones jóvenes y adultos mayores, se estima que llegará para el 2050 a la cifra de 5 personas de 75 y más años por cada niño de 0-4 años y 3 personas de 60 y más años por cada niño menor de 15 años.

La necesidad de desarrollo de la política de atención al anciano hace que la dirección del gobierno de nuestro país decidiera acrecentar los esfuerzos y dinamizar las respuestas necesarias para que Cuba cumpliera con los principios y recomendaciones sobre envejecimiento en la Asamblea de Naciones Unidas, efectuada en Nueva York en 1992, y de esta forma brindar la calidad de vida óptima a nuestros ancianos. Destacan dentro de estos esfuerzos la fundación del Centro Iberoamericano para La Tercera Edad y la puesta en marcha de un nuevo Programa de Atención Integral al Anciano Cubano brindando particular atención al estudio e implementación de estrategias en todos los sectores de la sociedad para enfrentar los elevados niveles de envejecimiento de la población.

“...lo más humano que pueda hacer la sociedad es ocuparse de los ancianos...”

Fidel Castro Ruz.

La adultez mayor es otra etapa de la vida en la cual se continúa desarrollando la personalidad, y las capacidades para aprender pues *“la educación comienza en la cuna y termina en la tumba”*, por tanto es un reto de nuestros planes de educación colocar a las personas mayores a la altura de estos tiempos, desde el punto de vista cultural haciéndose necesario que las ciencias de la educación garanticen un adecuado proceso educativo y de aprendizaje del anciano en esta etapa, estimularlo para que viva esta etapa llena de cambios desde una perspectiva positiva y enriquecedora e influir sobre estos factores psicológicos y sociales.

“... hay que darle un lugar de honor en la sociedad a las personas que entregaron su vida al trabajo. Claro hay que expresarles una consideración especial y crearles todas las condiciones para que vivan de la manera más feliz posible...”

Fidel Castro Ruz.

Producto de que la población de adultos mayores ha experimentado un crecimiento notorio como respuesta a numerosos factores que se han conjugado para sustentar una mayor esperanza de vida, Naciones Unidas ha considerado, a través de distintos organismos, elaborar políticas generales que vayan en

beneficio directo de las condiciones de vida de estas personas. Se destacan los planes y programas emprendidos a nivel mundial por la UNESCO en el área de la cultura y la educación, y los desarrollados por la Organización Mundial de la Salud (OMS), que han puesto especial énfasis en la salud preventiva, en beneficio de una mayor calidad de vida.

La educación en el adulto mayor, una educación para aprender a vivir, es una tarea de primer orden para todos aquellos que desde la familia, la comunidad, el centro de salud, centros educativos interactúen con personas de la tercera edad.

La educación en la tercera edad debe partir de que sea ofrecida a los ancianos para conservar su autosuficiencia, la adaptación social, forma de mantener el vínculo con el desarrollo social actual, el sentirse informado como un hombre de su tiempo sobre la evolución del mundo actual. Teniendo en cuenta la profundización en la búsqueda de métodos idóneos para transmitir mensajes que enseñen y eduquen, ajustados a la vejez logrando un mejor estilo de vida, en donde existan proyectos, esperanzas, conocimiento real de sus potencialidades, de sus valores y hasta dónde puede llegar. Permitiendo al anciano llevar una vida más saludable tanto física como psíquicamente. La educación en el adulto mayor permite la inserción del anciano en esa sociedad que un día construyó y de la que se siente apartado, resulta un modo de lucha contra las representaciones que hoy priman de la vejez y que tanto los limitan, constituyendo así una necesidad de orden social.

En el mundo se conoce como Universidades de Adultos Mayores o de la Tercera Edad, en nuestro país se les llaman Cátedras ya que forman parte del sistema nacional de la Educación Superior como programa de extensión universitaria dirigida a la educación de las personas mayores.

La primera Cátedra Universitaria del Adulto Mayor (CUAM) se fundó en febrero del año 2000, en la Universidad de la Habana con el coauspicio de la Central de Trabajadores de Cuba (CTC), y la Asociación de Pedagogos de Cuba (APC). Existen gran cantidad de Cátedras, Filiales y Aulas del Adulto Mayor siendo estas

urbanas o rurales en los municipios cabeceras, en la montaña, en la ciénaga, en los bateyes, en las comunidades agropecuarias, azucareras, o en las zonas pesqueras, e inclusive en los cayos.

Sin dudas, esta extensión a nivel nacional ha sido gracias al empeño en lo fundamental del papel organizativo del Movimiento de Atención a Jubilados y Pensionados de la CTC como auspiciador del programa en cada territorio. De hecho ha sido decisiva la incorporación de los jubilados y jubiladas cubanas, ya sea como alumnos, como gestores de filiales en sus comunidades, e inclusive como profesores o facilitadores, a medida que se han ido graduando como cursantes de la Cátedra.

Por lo que puede afirmarse que este es un proyecto educativo: no para, sino con los adultos mayores.

Este proyecto es atendido a nivel nacional por la Dirección de Extensión Universitaria por lo que pertenece al sistema nacional de educación, y a la vez objeto de análisis del cumplimiento de objetivos de las Universidades.

Las CUAM, sus filiales y aulas son atendidas por los Centros de Educación Superior de cada provincia y por las Sedes Universitarias Municipales (SUM), como actividad extensionista de la nueva universidad en los territorios, conocido como proceso de universalización de la educación superior cubana.

La generación de mayores en Cuba ha sido, y es importante protagonista de nuestra Revolución. Memoria viva de una sociedad que trabaja por un mundo mejor, y que ha logrado un preciado valor: la dignidad humana.

En los últimos años la cátedra de adultos mayores ha experimentado un incremento en el ingreso de nuevos alumnos ofreciéndoles una amplia gama de posibilidades, donde los profesores adjuntos imparten talleres a los adultos mayores, realizan funciones de tutorías a los trabajos de cursos o tesinas de los cursantes mayores, se realizan actividades de intercambio intergeneracional, se abordan temáticas específicas dentro de un área del conocimiento, entre otras. De

todas estas actividades se lleva un riguroso control, utilizando únicamente para esto la ayuda de herramientas ofimáticas como Documentos Word y Excel, donde la gran parte de toda la información se concentra en archivos de secretaria y gabinetes dificultando su resguardo y el acceso a consultar la información y su actualización, propiciando así errores de duplicidad o pérdida de información que pueden afectar los resultados del curso y la gestión académica lo cual trae consigo inconvenientes que indiscutiblemente presentan una **situación problemática**:

Gran cantidad de información a manipular y consultar. La misma se almacena en formato duro, provocando pérdida de tiempo en la búsqueda de información necesaria para elaborar los reportes e informes requeridos. Asimismo se precisa que esta información esté disponible para ser consultada por los coordinadores de la provincia y el municipio.

El proceso se realiza manualmente, constando solo con el apoyo de herramientas de ofimática, abarcando la creación de matrículas, generación de reportes, cantidad de continuantes y nuevos ingresos por cada aula y municipio, la impresión de diplomas, y así un gran número de actividades las cuales en ocasiones sufren cambios durante el curso.

El **problema de la investigación** está dado según lo anteriormente expuesto, en: ¿Cómo contribuir a la gestión de la información académica de la Cátedra Universitaria del Adulto Mayor de la Universidad de Cienfuegos?

Para dar solución a este problema se define como **idea a defender**: la aplicación de un sistema informático para la gestión de la información académica en la CUAM facilitará el trabajo con los datos de estudiantes, actividades y obtención de reportes.

En este trabajo se presenta como **objeto de estudio** la gestión de la información académica en la Universidad de Cienfuegos, más específicamente como **campo de acción** la gestión de la información académica en la Cátedra Universitaria del Adulto Mayor de la Universidad de Cienfuegos.

Se define como **objetivo general** de esta investigación: la realización de un sistema informático para la gestión de la información académica de la CUAM en la UCF.

Se desglosa en los siguientes **objetivos específicos**:

1. Analizar el proceso de gestión de la información académica de la CUAM en la UCF.
2. Diseñar una solución capaz de agilizar el proceso de gestión de la información académica de la CUAM en la UCF.
3. Implementar la solución propuesta mediante el desarrollo de un sistema informático web.
4. Validar el sistema propuesto.

Los anteriores objetivos se cumplirán a través del desarrollo de las siguientes **tareas**:

1. Entrevista a los trabajadores que participan en el proceso.
2. Investigación de la existencia de otros sistemas informáticos con funcionalidades similares a las que se necesitan o relacionados con el proceso de gestión que se necesita.
3. Elaboración de la arquitectura base de FÉNIX.
4. Esquema del modelo de datos que almacenará la información del proceso.
5. Selección de las metodologías, tecnologías, herramientas y lenguajes a utilizar en la realización de FÉNIX.
6. Realización del estudio de factibilidad del proyecto.
7. Aplicación de la encuesta a los usuarios finales del producto para conocer su satisfacción con el mismo.
8. Análisis de los resultados de la encuesta.
9. Confección de la documentación del sistema.

Esta solución ofrecerá un significativo **aporte práctico** a la Universidad de Cienfuegos pues estará provista de un sistema informático que se ajusta a las necesidades reales del manejo de la información de la CUAM, siendo capaz de

mejorar el proceso de gestión de la información académica en la cátedra y ofreciendo reportes de interés para los usuarios.

Esta investigación está estructurada en resumen, introducción, tres capítulos, Conclusiones, Recomendaciones, Referencias Bibliográficas, Bibliografía y Anexos. A continuación se muestra una breve descripción de cada uno de ellos:

Capítulo 1 “Fundamentación Teórica”: Se abordan los aspectos teóricos de la investigación y los conceptos asociados al dominio del problema. Se explican conceptos asociados al objeto de estudio y las diferentes tendencias y metodologías existentes en la actualidad para el desarrollo de aplicaciones informáticas.

Capítulo 2 “Análisis y diseño del sistema informático”: Descripción de los procesos del negocio, actores y trabajadores. Se aborda lo concerniente a la concepción del sistema partiendo de las funcionalidades del mismo y los principios de diseño para la implementación y diseño de la interfaz de la aplicación propuesta.

Capítulo 3 “Estudio de factibilidad y validación de la solución”: Se realiza el estudio de factibilidad del sistema, determinándose los costos, beneficios tangibles e intangibles asociados al sistema. Además la validación del mismo mediante métodos estadísticos, analizándose los resultados a través la realización de entrevistas a los usuarios finales del sistema.

Capítulo 1: Fundamentación teórica

En este capítulo se exponen aspectos teóricos de la investigación realizando un análisis sobre los principales conceptos vinculados al dominio del problema y sobre los sistemas existentes vinculados al campo de acción. Concluye con el estudio de metodologías, tecnologías y herramientas que se utilizarán para el desarrollo del sistema.

1.1 Principales conceptos asociados al dominio del sistema.

Información: Es un conjunto de datos que están organizados y que tienen un significado. De esta manera, si tomamos datos por separado no tendrían un significado mientras que si los agrupamos en forma organizada, sí. La información es un elemento fundamental en el proceso de comunicación, ya que tiene un significado para quien la recibe, que la va a comprender si comparte el mismo código de quien la envía. Esto no sólo ocurre en un proceso social sino también en el mundo de la informática. [3]

Gestión: La gestión, entendida en términos con temporáneos, es el procedimiento de adecuación de recursos de cualquier índole a aquellos fines para los cuales han sido recabados dichos recursos. [3]

Gestión de la información: La gestión de la información es el proceso de analizar y utilizar la información que se ha obtenido y registrado para permitir a los administradores tomar decisiones documentadas. La información es un elemento fundamental para el desarrollo, con el transcurso de los años, la gestión de la información ocupa, cada vez más, un espacio mayor en la economía de los países a escala mundial. Para desarrollar una correcta gestión de la información es necesario tener en cuenta una serie de pasos, entre los que se encuentran los siguientes: [3]

1. Determinar la información que se precisa.
2. Recoger y analizar la información.

3. Registrarla y recuperarla cuando sea necesaria.
4. Utilizarla.
5. Divulgarla.

1.2 Sistemas existentes vinculados al campo de acción

SIGENU

El Sistema de Gestión de la Nueva Universidad (SIGENU) es un sistema que se ha desarrollado para la gestión de toda la información académica vinculada con la educación superior en Cuba. En correspondencia con su carácter nacional y la gran diversidad de sistemas de enseñanza superior con que cuenta la universidad cubana, este sistema ha sido concebido para adaptarse a todos los centros de educación superior del país con sus diversas particularidades y distintas maneras de realizar determinados procedimientos.

El sistema SIGENU está compuesto por cuatro elementos fundamentales:

Base de Datos: Este es lugar en el cual se almacena toda la información registrada.

Servidor de Aplicaciones: Este elemento es el encargado de actuar como intermediario entre la Base de Datos y las aplicaciones clientes, o sea, es quien hace posible que la información registrada en la base sea visualizada y actualizada a través de las aplicaciones que son manipuladas por el usuario. Este elemento es quien permite dar un servicio a través de la red de manera que la información pueda ser accedida remotamente por las aplicaciones ubicadas en varios lugares.

Aplicación Cliente Secretaría: Es la aplicación que constituye el elemento que esencialmente permite la inserción y actualización de toda la información que se registre en el sistema. Además, permite obtener un conjunto importante de reportes muy usados cotidianamente en el mundo de la educación superior. Consta de los siguientes módulos: Codificadores, Matrícula, Control de

estudiantes, Plan de Estudio, Evaluaciones y Reportes. La aplicación cliente se encuentra disponible para escritorio (desktop) y web.

Aplicación Cliente de Administración: Es la aplicación que permite la inserción y actualización de los usuarios y todas las funcionalidades que deban ser ejecutadas por los administradores para monitorear el correcto funcionamiento del sistema y su seguridad. Naturalmente a esta aplicación solo tendrán acceso los encargados de la administración del SIGENU.

Modulo Web de Recuperación de Información (Recuperador): Es un módulo que permite obtener diversos reportes con los que se puede recuperar toda la información necesaria del sistema.[4]

Deficiencia:

No cuenta con un módulo para la gestión de la información académica de la CUAM.

SGE: Sistema de Gestión Educativa

Corresponde a un software orientado a facilitar las actividades de directivos y docentes, permitiendo planificar acciones elementales de la gestión educativa a nivel administrativo, como ingreso de alumnos, dotación docente y administración de establecimientos, entre otras acciones. A nivel pedagógico, el recurso permite el diseño de planificaciones integrales de la Unidad Técnico Pedagógica: planificaciones, evaluaciones, control de horarios y nómina de alumnos, son algunos ejemplos. La aplicación presenta amplias funcionalidades relacionadas con la generación de informes, los cuales son requeridos por directivos, docentes, alumnos y alumnas, administrativos, DAEM y UTP, entre otros.

Del punto de vista técnico, el software opera en ambiente web. Para su correcto funcionamiento, se requiere de conexión a internet con velocidad media – alta, de manera que sea posible ejecutar las diferentes opciones en una unidad de tiempo aceptable.

El software cuenta con un módulo que permite configurar determinadas funcionalidades, lo que otorga mayor flexibilidad para su uso en la institución educativa. Por otra parte, es posible definir distintos perfiles de usuarios, lo cual constituye un aporte en aspectos de seguridad.

Sistema de Gestión Docente SGD

El sistema de Gestión Docente SGD, es una potente herramienta de Gestión Global Académica para centros educativos que incorpora como elemento principal un Software diseñado específicamente para el tratamiento de la Información Académica y las Comunicaciones entre padres, profesores, tutores y alumnos.

SGD es, día a día, un nuevo modo de trabajo de alto rendimiento desarrollado en Java y con un potente motor de base de datos MySQL, que tanto para centros educativos como para docentes; se ha convertido en la aplicación de gestión académica más avanzada del mercado para el sector educativo: segura, de fácil uso y de tecnología compatible e integrable con los principales sistemas oficiales de las Consejerías de Educación.

El Software SGD V.7 es una aplicación multilingüe de total compatibilidad con sistemas operativos Windows y Linux; instalado en cualquier PC, permite que tanto el equipo directivo como profesores o tutores puedan centralizar y acceder simultáneamente a todos sus procesos de trabajo diario.

1. Gestión de Notas y Fichas de Alumnos
2. Introducción de incidencias de Alumnos
3. Seguimiento del Comportamiento del Alumno
4. Notas parciales y seguimiento de deberes
5. Informes y Estadísticas actualizadas
6. Horarios, Actividades, Tareas...
7. Anotaciones de mantenimiento del centro

Deficiencias de **SGE y SGE**

1. Son productos de software privativos
2. No cuentan con un módulo de gestión de información académica de las CUAM.

1.3 Conceptos, patrones, aplicaciones, tecnologías, lenguajes y bibliotecas utilizados en el desarrollo de software

UML

Lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

UML permite la modificación de todos sus miembros mediante estereotipos y restricciones. Un estereotipo nos permite indicar especificaciones del lenguaje al que se refiere el diagrama de UML. Una restricción identifica un comportamiento forzado de una clase o relación, es decir mediante la restricción estamos forzando el comportamiento que debe tener el objeto al que se le aplica. [5]

Es importante resaltar que UML no es un lenguaje de programación sino un lenguaje de propósito general para el modelado orientado a objetos y también puede considerarse como un lenguaje de modelado visual que permite una abstracción del sistema y sus componentes, permitiendo modelar cualquier tipo de proyectos, tanto informáticos como de arquitectura, o de cualquier otra rama.

Metodología de desarrollo

Una metodología de desarrollo de software se refiere a un *framework* que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información.

A lo largo del tiempo, una gran cantidad de métodos han sido desarrollados diferenciándose por su fortaleza y debilidad.

El *framework* para metodología de desarrollo de software consiste en:

1. Una filosofía de desarrollo de software con el enfoque del proceso de desarrollo de software.
2. Herramientas, modelos y métodos para asistir al proceso de desarrollo de software.

Existen dos tipos principales de metodologías de desarrollo de software: las llamadas metodologías pesadas y las metodologías ágiles o livianas. Estas últimas tienen como características fundamentales el promover iteraciones en el desarrollo a lo largo de todo el ciclo de vida del proyecto minimizando los riesgos. El software desarrollado en una unidad de tiempo es llamado una iteración, la cual debe durar de una a cuatro semanas. Cada iteración del ciclo de vida incluye: planificación, análisis de requerimientos, diseño, codificación, revisión y documentación.

Una iteración no debe agregar demasiada funcionalidad para justificar el lanzamiento del producto al mercado, pero la meta es tener un demo (sin errores) al final de cada iteración. Al final de cada iteración el equipo vuelve a evaluar las prioridades del proyecto.

Existen varias metodologías de desarrollo de software calificadas como livianas de la que se pueden mencionar Crystal Clear, Scrum, eXtreme Programming y Feature Driven Development (FDD) que fue la seleccionada para el desarrollo de este proyecto.

FDD

FDD es un proceso diseñado por Peter Coad, Erich Lefebvre y Jeff de Luca y se podría considerar a medio camino entre RUP y XP, aunque al seguir siendo un proceso ligero es más similar a este último.

FDD está pensado para proyectos con tiempo de desarrollo relativamente corto (menos de un año). Se basa en un proceso iterativo de iteraciones cortas que producen un software funcional que el cliente y la dirección de la empresa pueden ver y monitorizar.

Las iteraciones se deciden en base a funcionalidades (de ahí el nombre del proceso) o funcionalidades, que son pequeñas partes del software con significado para el cliente.

Un proyecto que sigue FDD se divide en 5 fases:

1. Desarrollo de un modelo general.
2. Construcción de una lista de funcionalidades.
3. Plan de entregas en base a las funcionalidades a implementar.
4. Diseñar en base a las funcionalidades.
5. Implementar en base a las funcionalidades.

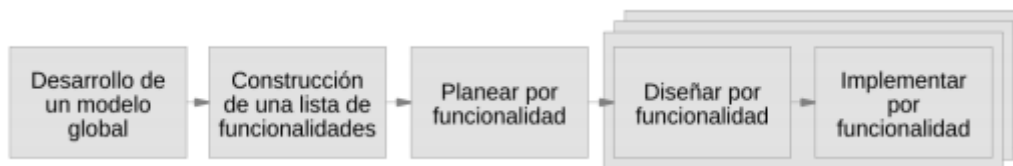


Figura 1: Fases de Desarrollo de la metodología FDD

Patrón Modelo-Vista -Controlador (MVC)

Una aplicación diseñada siguiendo el patrón MVC organiza y separa las funcionalidades en tres capas:

Modelo: encapsula los datos de la aplicación, el flujo de datos y la lógica del negocio.

Vista: extrae datos del modelo y les aplica el formato necesario para la presentación.

Controlador: dirige el flujo de la aplicación y las entradas y salidas de datos entre la capa modelo y la capa controladora.

MVC fue originalmente desarrollado en la década de 1970 en el centro de investigaciones de Xerox en Palo Alto, California: la cuna de las interfaces gráficas de usuario precisamente para resolver los problemas de interacción en las GUI.

Cuando la vista y el controlador están combinados, el resultado es una versión simplificada de MVC llamada arquitectura modelo/vista. Esta aún separa los datos de la presentación pero simplifica la arquitectura al eliminar una capa desde el punto de vista del desarrollador, pero aún existente con el nombre de delegate: otro componente (normalmente ya desarrollado) que lleva las responsabilidades del controlador.

1.4 Lenguajes de Programación utilizados

PHP

PHP es un lenguaje de programación interpretado, originalmente diseñado en Perl, con las bases de la escritura de un grupo de CGI binarios escritos en lenguaje C por el programador danés Rasmus Lerdorf en el año 1994.

Posee un gran parecido con los lenguajes más comunes de programación estructurada posibilitándole a los desarrolladores crear aplicaciones más complejas con un curva de aprendizaje corta.

A pesar de que fue diseñado originalmente para la creación de páginas web dinámicas, permite crear aplicaciones con una interfaz para el usuario.

Otra de sus características es que permite la conexión a diferentes tipos de servidores de bases de datos como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite. Tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos tales como UNIX (y de ese tipo, como Linux), Windows y Mac OS X, y puede interactuar con los servidores Web más populares como Apache. Es gratuito e independiente de plataforma, muy rápido, con una gran biblioteca de funciones y mucha documentación.

Es una alternativa a las tecnologías de Microsoft ASP y ASP.NET (que utiliza C# y Visual Basic .NET como lenguajes), a ColdFusion de la empresa Adobe, a JSP/Java y a CGI/Perl.

En resumen, algunas características que lo hacen uno de los lenguajes del lado del servidor más populares son:[6]

- Es un lenguaje multiplataforma.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite aplicar técnicas de programación orientada a objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.

- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.
- Tiene manejo de excepciones (desde PHP5).

Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar (muchos otros lenguajes tampoco lo hacen), aun estando dirigido a alguna en particular, el programador puede aplicar en su trabajo cualquier técnica de programación y/o desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del Patrón de diseño Modelo Vista Controlador(o MVC), que permiten separar el tratamiento y acceso a los Datos, la Lógica de control y la Interfaz de usuario en tres componentes independientes.

Framework utilizado

Un framework para aplicaciones web es un framework diseñado para apoyar el desarrollo de sitios web dinámicos, aplicaciones web y servicios web. Estos facilitan el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los tediosos detalles de bajo nivel para proveer un sistema funcional. Por ejemplo, muchos frameworks proporcionan bibliotecas para acceder a bases de datos, estructuras para plantillas, gestión de sesiones, y con frecuencia facilitan la reutilización de código.

El framework Yii

El sitio oficial de Yii describe al producto como:

Yii es un framework PHP basado en componentes de alta performance para desarrollar aplicaciones Web de gran escala. El mismo permite la máxima reutilización en la programación web y puede acelerar el proceso de desarrollo. El nombre Yii (pronunciado /i:/) es por *fácil* (en inglés: easy), *eficiente* (en inglés: efficient) y *extensible* (en inglés: extensible).

Yii es un framework genérico de programación Web que puede ser utilizado para todo tipo de aplicaciones Web. Gracias a que es liviano de correr y que está equipado con soluciones de cacheo sofisticadas, es adecuado para desarrollar aplicaciones de gran tráfico como portales, foros, sistemas de administración de contenidos (CMS), Sistemas de comercio electrónico, entre otros.

Como la mayoría de los frameworks PHP, Yii es un framework MVC (modelo-vista-controlador).

Yii sobresale frente a otros frameworks PHP en su eficiencia, su gran cantidad de características y su clara documentación. Yii ha sido diseñado cuidadosamente desde el principio para el desarrollo de aplicaciones de Web. No es ni un subproducto de un proyecto ni un conglomerado de trabajo de terceros. Es el resultado de la vasta experiencia de los autores en desarrollo de aplicaciones Web y de la investigación y la reflexión de los más populares frameworks de programación Web y aplicaciones.

JavaScript

Es un lenguaje interpretado, es decir, que no requiere compilación, orientado a las páginas Web. Dirigido por eventos, por lo que estará listo para actuar en cuanto un evento (un clic en un botón, por ejemplo) sea ejecutado, implementa una sencilla interfaz de objetos/propiedades/métodos. Se integra dentro del código HTML de las páginas Web, se ejecuta en el navegador al mismo tiempo que las sentencias van descargándose junto con el código HTML. Brinda rapidez a la aplicación Web a la hora de las validaciones de los formularios. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado.

jQuery

“jQuery es una biblioteca de JavaScript rápida y concisa que simplifica el recorrido, manejo de eventos, animación, e interacciones Ajax en el documento HTML para

un rápido desarrollo web. jQuery está diseñado para cambiar la manera que tu escribes JavaScript". [7]

jQuery tiene licencias MIT y GPL permitiendo el desarrollo de aplicaciones de Software Libre. La biblioteca se centra en la creación de ricas funcionalidades para el contenido web mediante la encapsulación de la tradicional interfaz de DOM en un mecanismo descrito como encuentra algo, manipúlalo. La selección de elementos DOM se realiza utilizando los tradicionales selectores CSS y posee un gran número de funcionalidades. jQuery soporta animaciones, manejo de eventos, Ajax y extensiones nativas al lenguaje JavaScript. La comunidad de desarrollo y usuarios de jQuery es muy activa por lo que existe una cantidad creciente de plugins.

HTML

El HTML, acrónimo inglés de Hypertext Markup Language (lenguaje de marcas hipertextuales). Lenguaje de marcado predominante para la elaboración de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un script (por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

CSS

El concepto de hojas de estilo apareció por primera vez en 1996 cuando W3C publicó una recomendación nueva titulada "Hojas de estilo en cascada" o CSS su sigla en inglés. El principio de las hojas de estilo consiste en la utilización de un solo documento para almacenar las características de presentación de las páginas asociadas a grupos de elementos. Esto implica nombrar un conjunto de definiciones y características de presentación de las páginas, y activar esos nombres para aplicarlos a una parte del texto. Por ejemplo, se pueden

configurarlos títulos de una sección para que aparezcan en fuente Arial, en color verde y en cursiva.

DOM

El DOM (Modelo de Objetos del Documento) es una interfaz de programación mantenida por el W3C para representar y manipular la estructura de documentos XML y HTML mediante la utilización de lenguajes tipo ECMAScript aunque ya tiene implementaciones para muchos otros lenguajes de programación.

Sistema de Base de Datos

MySQL

Es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.[8]

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.[8]

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.[8]

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius.[8]

Entorno de desarrollo integrado NetBeans

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java, pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extenderlo. NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento, y con cerca de 100 socios en todo el mundo. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio de 2000 y continúa siendo el patrocinador principal de los proyectos.[9]

Herramienta de diseño gráfico GIMP

GIMP es una herramienta de manipulación fotográfica multiplataforma. GIMP es un acrónimo de GNU Image Manipulation Program. En GIMP se pueden realizar todo tipo de tareas de manipulación de imágenes, incluyendo retoque fotográfico, composición de imágenes y creación de imágenes.[10]

GIMP tiene muchas capacidades. Se puede usar como un sencillo programa de pintura, un programa de retoque fotográfico profesional, un sistema en línea de proceso por lotes, un generador de imágenes para producción en masa, un conversor de formatos de imágenes, etc.[10]

GIMP es ampliable y extensible. Está diseñado para ampliarse con complementos y extensiones. La interfaz avanzada de guionado (scripting) permite automatizar desde las tareas más simples hasta los procedimientos más complejos de manipulación de imágenes.[10]

Uno de los fuertes de GIMP es su libre disponibilidad desde varias fuentes para muchos sistemas operativos. Casi todas las distribuciones de GNU/Linux incluyen al GIMP como una aplicación estándar. El GIMP también está disponible para otros sistemas operativos como Microsoft Windows™ o Mac OS X™(Darwin) de Apple. GIMP es una aplicación de Software Libre cubierta por la Licencia Pública General [GPL].[10]

1.5 Descripción del sistema propuesto

El sistema propuesto será implementado como una aplicación web. La aplicación estará orientada a que un usuario pueda gestionar o consultar información asociada a la CUAM sin la necesidad de tener que utilizar diferentes aplicaciones para realizar los procesos requeridos, puesto que todas las herramientas estarán unificadas en una misma aplicación disminuyendo así considerablemente el consumo de recursos que representa el tener varias aplicaciones ejecutándose al mismo tiempo.

1.6 Conclusiones

Después del análisis realizado se concluye que dada la importancia que se le atribuye hoy en día a la CUAM, se hace necesario utilizar las metodologías y tecnologías actuales para crear un sistema informático que facilite la gestión de la información académica relacionada a la CUAM. Para esto se utilizó la metodología FDD y el lenguaje de modelado UML, por las ventajas que proporcionan. Para el almacenamiento de los datos se seleccionó MySQL y como servidor web Apache2. Se escogió NetBeans para la implementación del sistema web con HTML, CSS3, JavaScript, Twitter Bootstrap, JQuery y Ajax del lado del cliente, PHP y el framework Yii del lado del servidor.

Capítulo2: Análisis y diseño del sistema informático

En el presente capítulo se toma como guía para el desarrollo del software la metodología FDD detallando sus procesos a partir de las fases definidas: se definen los requisitos, se planifican y además se hace una descripción del diseño general así como de la arquitectura de la aplicación.

2.1 Desarrollo de un modelo global.

Al iniciar esta fase ya se tiene una idea del contexto y los requerimientos del sistema. Siendo probable que el documento de especificación de requerimientos exista, FDD no hace énfasis en la recolección y administración de estos, por lo que no existe un documento que especifique los mismos.

“El dominio global es dividido en diferentes áreas y se realiza un informe detallado para cada una de dichas áreas por parte de los expertos del dominio. Luego de cada informe, un grupo de desarrolladores realizan un modelo de objetos para el área del dominio. Además, el equipo de desarrollo discute y decide cual es el modelo de objetos más apropiado para cada área del dominio. Simultáneamente, el modelo global del sistema es construido.”[11]

2.2 Flujo actual de los procesos involucrados en el campo de acción.

En la CUAM de la UCF se atiende no solo a los coordinadores municipales sino también de manera más directa a los presidentes de filiales y responsables de módulos quienes garantizan el funcionamiento y la calidad de las actividades desarrolladas por los estudiantes.

Dichos coordinadores deben realizar actividades como:

- Divulgar las actividades que se orienten por el Grupo Provincial y de la Cátedra.

- Coordinar, chequear, y controlar el trabajo de las aulas del territorio.
- Consolidar la información académica de todos los cursantes en las filiales y aulas del municipio.

Los coordinadores deben tener archivada toda esta información para el posterior uso y manejo de datos conformando reportes mensualmente, quedando estos limitados por las habilidades con herramientas ofimáticas con que cuente el personal encargado de esto y el difícil acceso a la información.

Objeto de automatización

Se desea mejorar el proceso de gestión relacionado con las actividades que realizan los coordinadores. Entre los que resalta el proceso de gestión de la información académica.

2.2.1 Modelo de Dominio

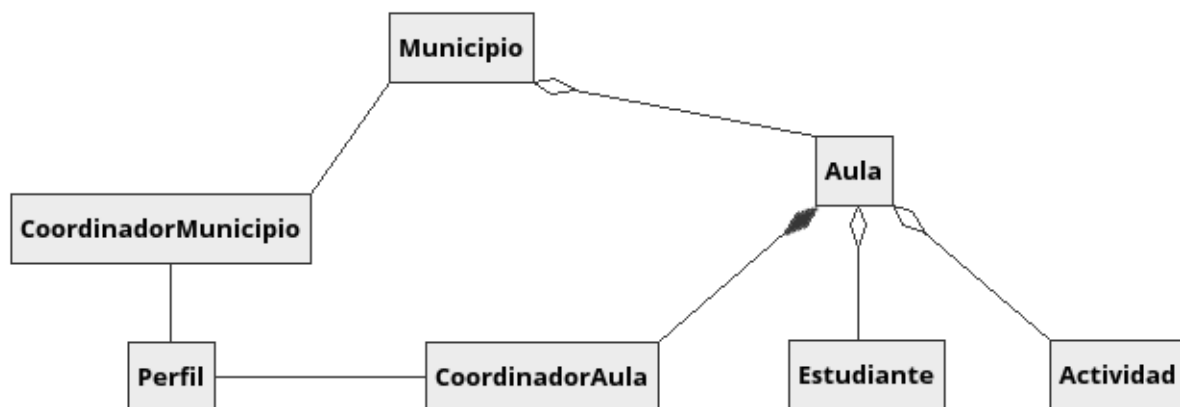


Figura 2: Diagrama del modelo de objetos

Reglas a considerar

El coordinador de municipio es el responsable de recepcionar e incluir los datos del municipio y las actividades que se realicen.

1. El administrador tendrá acceso a todo el contenido.
2. Cada coordinador solo tendrá acceso a sus datos.

3. Estos datos podrán ser editados por sus coordinadores.
4. Las operaciones de impresión e importación solo estarán disponibles para los usuarios que posean los permisos.

2.2.2 Construcción de una lista de funcionalidades.

La lista de funcionalidades resume todas las funciones que debe realizar el sistema así como restricciones de diseño e implementación. Esta lista es dividida en subconjuntos basados en la funcionalidad representando diferentes actividades en un área específica del dominio.

Requerimientos funcionales

Los requerimientos funcionales permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. La idea principal en esta etapa es identificar y documentar lo que se necesita, de forma tal que pueda ser fácilmente transmitido al cliente y al equipo de desarrollo.

Se definieron un total de 98 requerimientos funcionales, los mismos permiten determinar, de una manera clara, lo que el sistema será capaz de realizar, representando las aspiraciones de los usuarios y lo que necesitan, por lo que se puede definir en los siguientes requerimientos funcionales:

Código	Funcionalidad	Prioridad
FUM01	Crear municipio	A
FUM02	Eliminar municipio	A
FUM03	Buscar municipio	A
FUM04	Listar municipio	B
FUM05	Cambiar datos del municipio	A
FUM06	Calcular total de nuevos ingresos femeninos por municipio	A

FUM07	Calcular total de nuevos ingresos masculinos por municipio	A
FUM08	Calcular total continuantes femeninos por municipio	A
FUM09	Calcular total de continuantes masculinos por municipio	A
FUM10	Calcular matricula total por municipio	A
FUM11	Calcular total de aulas por municipio	A
FUM12	Mostrar datos de municipio	B
FUM13	Opción para cambiar vista de datos Tabla/Grafico	C
FUM14	Mostrar estadística de municipio	B
FUM15	Graficar datos de municipio	B
FUM16	Mostrar vista detallada del municipio	C
FUM17	Imprimir datos del municipio	C
FUM18	Imprimir gráfico del municipio	C
FUM19	Exportar datos del municipio a .doc	C
FUM20	Exportar datos del municipio a .odt	C
FUM21	Exportar datos del municipio a .pdf	C
FUM22	Exportar gráfico del municipio	C

Tabla 1 Primer Bloque de funcionalidades

Código	Funcionalidad	Prioridad
FUA01	Crear Aula	A
FUA02	Eliminar Aula	A
FUA03	Buscar Aula	A

FUA04	Cambiar datos de Aula	A
FUA05	Listar Aulas	B
FUA06	Calcular total de nuevos ingresos femeninos por aula	A
FUA07	Calcular total de nuevos ingresos masculinos por aula	A
FUA08	Calcular total de continuantes femeninos por aula	A
FUA09	Calcular total de continuantes masculinos por aula	A
FUA10	Calcular total de nuevos ingreso por aula	A
FUA11	Calcular total de continuantes por aula	A
FUA12	Calcular total estudiantes femeninos por aula	A
FUA13	Calcular total de estudiantes masculinos por aula	A
FUA14	Calcular matricula total por aula	A
FUA15	Mostrar datos de Aula	B
FUA16	Mostrar vista detallada del aula	C
FUA17	Mostrar estadística de aula	B
FUA18	Opción para cambiar vista de datos Tabla/Grafico	C
FUA19	Graficar datos del aula	B
FUA20	Imprimir datos del aula	C
FUA21	Imprimir gráfico del aula	C
FUA22	Exportar datos del municipio a .doc	C
FUA23	Exportar datos del municipio a .odt	C
FUA24	Exportar datos del municipio a .pdf	C

FUA25	Exportar gráfico del aula	C
-------	---------------------------	---

Tabla 2 Segundo Bloque de funcionalidades

Código	Funcionalidad	Prioridad
FUE01	Asignar estudiante a un aula	A
FUE02	Modificar los datos de un estudiante	A
FUE03	Eliminar los datos de un estudiante	A
FUE04	Listar estudiantes	B
FUE05	Permitir ordenar el listado de estudiantes por nombre	C
FUE06	Permitir ordenar el listado de estudiantes por carnet de identidad	C
FUE07	Permitir ordenar el listado de estudiantes por continuantes	C
FUE08	Mostrar datos detallados del estudiante	C

Tabla 3 Tercer Bloque de funcionalidades

Código	Funcionalidad	Prioridad
FUT01	Agregar actividad	A
FUT02	Eliminar actividad	A
FUT03	Buscar actividad	A
FUT04	Listar actividades ordenadas por fecha	B
FUT05	Cambiar datos de actividad	A
FUT06	Mostrar datos de actividad	B
FUT07	Mostrar datos de actividad con la opción de ordenar por	B

	fecha	
FUT08	Mostrar datos de actividad con la opción de ordenar por participantes	B
FUT09	Mostrar los datos completos de las actividades	B

Tabla 4 Cuarto Bloque de funcionalidades

Código	Funcionalidad	Prioridad
FUC01	Agregar coordinador a un municipio	A
FUC02	Agregar coordinador a un aula	A
FUC03	Eliminar coordinador	A
FUC04	Cambiar datos del coordinador	A
FUC05	Crear perfil a un coordinador	A
FUC06	Listar los coordinadores que pertenecen a un aula	B
FUC07	Listar los coordinadores que pertenecen a un municipio	B
FUC08	Mostrar datos del coordinador de municipio	B
FUC09	Mostrar los datos del coordinador de aula	B
FUC10	Mostrar datos detallados del coordinador	C
FUC11	Actualizar perfil del coordinador	A
FUC12	Cambiar contraseña	A
FUC13	Iniciar sesión en el sistema	A
FUC14	Cerrar sesión	B

Tabla 5 Quinto Bloque de funcionalidades

Funcionalidades de administración

Código	Funcionalidad	Prioridad
ADM01	Autenticarse	A
ADM02	Crear nuevo usuario	A
ADM03	Cambiar datos de los usuarios	A
ADM04	Cambiar contraseña	A
ADM05	Eliminar usuarios	A
ADM06	Buscar usuarios por Carne de Identidad	B
ADM07	Buscar usuarios por Nombre	B
ADM08	Buscar usuarios por Nombre de Usuario	B
ADM09	Buscar usuarios por Tipo de Usuario	B
ADM10	Listar usuarios	B
ADM11	Ordenar listado de usuarios por Carne de Identidad	C
ADM12	Ordenar listado de usuarios por Nombre	C
ADM13	Ordenar listados de usuarios por Nombre de Usuario	C
ADM14	Ordenar listado de usuarios por Tipo de Usuario	C
ADM15	Crear nuevo municipio	A
ADM16	Modificar municipio	A
ADM17	Eliminar Municipio	A
ADM18	Buscar municipio por Nombre	B
ADM19	Listar municipio	B

Tabla 6 Bloque de funcionalidades del Módulo de Administración

Requisitos no funcionales

“Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Debe pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable[12].

En muchos casos los requisitos no funcionales son fundamentales en el éxito del producto. Normalmente están vinculados a requisitos funcionales, es decir una vez que se conozca lo que el sistema debe hacer se puede determinar cómo ha de comportarse, qué cualidades debe tener o cuán rápido o grande debe ser.”[12]

Los requerimientos no funcionales del sistema son los siguientes:

Requerimientos de apariencia o interfaz externa

- El sistema debe tener una interfaz fácil de navegar, sencilla, legible, simple de usar, interactiva e informativa.
- La interfaz debe respetar los parámetros de diseño.
- La interfaz debe estar orientada al cumplimiento de los objetivos del usuario.

Requerimientos de usabilidad

- El sistema debe ser fácil de usar por personas con poca experiencia con el trabajo con computadoras.
- Las funcionalidades del sistema deben ser fáciles de ubicar.
- El tamaño y color de la letra debe permitir leer con facilidad.
- El diseño de la interfaz debe evitar la pérdida de tiempo.

- Las opciones se deben localizar rápidamente y ser consistente en la ubicación de las funciones e iconos en la pantalla.

Requerimientos de rendimiento

- El sistema deberá ser rápido ante las solicitudes de los usuarios y en el procesamiento de la información.
- No debe tardarse en el procesamiento de acciones complejas.
- Debe permitir el acceso simultáneo a los datos por diferentes usuarios.

Requerimientos de portabilidad

La aplicación debe ser totalmente independiente de la plataforma, pudiendo ejecutarse en cualquier sistema operativo (Linux o Windows) a través de un servidor Web y servidor de bases de datos.

Requerimientos de seguridad

- Se debe garantizar un control riguroso sobre la seguridad de la información, estableciendo niveles de acceso. Definiendo una política de usuarios con privilegios de acuerdo a su rol, asegura que la información pueda ser consultada de acuerdo al nivel de acceso.
- Se implementará un mecanismo de encriptación para las contraseñas utilizando la función hash de MD5, garantizando la seguridad de las mismas.
- La información almacenada deberá ser consistente, utilizando para su validación funciones que limiten la entrada de datos.

Requerimientos de hardware

- Las máquinas que utilizarán el software tienen que estar conectadas a la red.
- Los requerimientos recomendados para una estación de trabajo deben ser de 128 Mb de RAM, 800 MHz, 2 GB de disco duro.
- Los requerimientos recomendados para el servidor del software deben ser de 1 GB de RAM, 1.4 GHz, 10 GB de disco duro.

Requerimientos de software

- Los requerimientos mínimos para una estación de trabajo deben ser:
- Un navegador web que supere las pruebas de Acid3.
- Se requiere además, una máquina servidor que cuente con:
- La versión 2.2.6 o superior del servidor Web Apache.
- Lenguaje de programación PHP.
- Servidor de base de datos MySQL 5.0 o superior.

Requisitos Políticos

La aplicación debe cumplir con lineamientos, políticas y/o regulaciones del Ministerio de la Informática y las Comunicaciones y del MES.

2.3 Planeación por funcionalidad

“En esta etapa se incluye la creación de un plan de alto nivel, en el cual la lista de funcionalidades es ordenada en base a la prioridad y a la dependencia entre cada funcionalidad. Además, las clases identificadas en la primera etapa son asignadas a cada programador.”[11]

Código	Valor	Fecha Inicial	Fecha Final	Días
FMU01	A	feb 28,2013	mar 1,2013	1
FMU02	A	feb 28,2013	mar 1,2013	1
FMU03	A	feb 28,2013	mar 1,2013	1
FUA01	A	feb 28,2013	mar 1,2013	1
FUA02	A	feb 28,2013	mar 1,2013	1
FUA03	A	feb 28,2013	mar 1,2013	1
FUE01	A	feb 28,2013	mar 1,2013	1
FUE02	A	feb 28,2013	mar 1,2013	1
FUT01	A	feb 28,2013	mar 1,2013	1

FUT02	A	feb 28,2013	mar 1,2013	1
FUC01	A	mar 1,2013	mar 2,2013	1
FUC02	A	mar 1,2013	mar 2,2013	1
FUM05	A	mar 1,2013	mar 2,2013	1
FUA04	A	mar 1,2013	mar 2,2013	1
FUE03	A	mar 1,2013	mar 2,2013	1
FUT03	A	mar 1,2013	mar 2,2013	1
FUT05	A	mar 1,2013	mar 2,2013	1
FUC03	A	mar 1,2013	mar 2,2013	1
FUC04	A	mar 2,2013	mar 3,2013	1
FUC05	A	mar 2,2013	mar 3,2013	1
FUC13	A	mar 2,2013	mar 3,2013	1
FUC11	A	mar 3,2013	mar 6,2013	3
FUC12	A	mar 3,2013	mar 6,2013	3
FUM06	A	mar 6,2013	mar 11,2013	5
FUM07	A	mar 6,2013	mar 11,2013	5
FUM08	A	mar 6,2013	mar 11,2013	5
FUM09	A	mar 6,2013	mar 11,2013	5
FUM10	A	mar 6,2013	mar 11,2013	5
FUM11	A	mar 6,2013	mar 11,2013	5
FUA06	A	mar 11,2013	mar 17,2013	6
FUA07	A	mar 11,2013	mar 17,2013	6
FUA08	A	mar 11,2013	mar 17,2013	6
FUA09	A	mar 11,2013	mar 17,2013	6
FUA10	A	mar 11,2013	mar 17,2013	6
FUA11	A	mar 11,2013	mar 17,2013	6
FUA12	A	mar 11,2013	mar 17,2013	6

FUA13	A	mar 11,2013	mar 17,2013	6
FUA14	A	mar 11,2013	mar 17,2013	6
ADM01	A	mar 18,2013	mar 20,2013	2
ADM02	A	mar 18,2013	mar 20,2013	2
ADM03	A	mar 18,2013	mar 20,2013	2
ADM04	A	mar 18,2013	mar 20,2013	2
ADM05	A	mar 18,2013	mar 20,2013	2
ADM15	A	mar 18,2013	mar 20,2013	2
ADM16	A	mar 18,2013	mar 20,2013	2
ADM17	A	mar 18,2013	mar 20,2013	2
FUC12	B	mar 21,2013	mar 28,2013	7
FUM04	B	mar 21,2013	mar 28,2013	7
FUM12	B	mar 21,2013	mar 28,2013	7
FUM16	C	mar 21,2013	mar 28,2013	7
FUM14	B	mar 21,2013	mar 28,2013	7
FUA05	B	mar 21,2013	mar 28,2013	7
FUA15	B	mar 21,2013	mar 28,2013	7
FUA17	B	mar 21,2013	mar 28,2013	7
FUC06	B	mar 21,2013	mar 28,2013	7
FUC07	B	mar 21,2013	mar 28,2013	7
FUC08	B	mar 21,2013	mar 28,2013	7
FUA16	C	mar 28,2013	abr 1,2013	4
FUE04	B	mar 28,2013	abr 1,2013	4
FUT04	B	mar 28,2013	abr 1,2013	4
FUC09	B	mar 28,2013	abr 1,2013	4
FUT06	B	abr 1,2013	abr 5,2013	4
FUT07	B	abr 1,2013	abr 5,2013	4

FUT08	B	abr 1,2013	abr 5,2013	4
FUT09	B	abr 1,2013	abr 5,2013	4
FUM15	B	abr 5,2013	abr 19,2013	14
FUA19	B	abr 5,2013	abr 19,2013	14
FUM13	C	abr 19,2013	abr 20,2013	1
FUA18	C	abr 19,2013	abr 20,2013	1
ADM06	B	abr 20,2013	abr 22,2013	2
ADM07	B	abr 20,2013	abr 22,2013	2
ADM08	B	abr 20,2013	abr 22,2013	2
ADM09	B	abr 20,2013	abr 22,2013	2
ADM10	B	abr 20,2013	abr 22,2013	2
ADM18	B	abr 20,2013	abr 22,2013	2
ADM19	B	abr 20,2013	abr 22,2013	2
FUE05	C	abr 22,2013	abr 23,2013	1
FUE06	C	abr 22,2013	abr 23,2013	1
FUE07	C	abr 22,2013	abr 23,2013	1
FUE08	C	abr 23,2013	abr 30,2013	7
FUC10	C	abr 23,2013	abr 30,2013	7
FUM17	C	may 1,2013	may 5,2013	4
FUM18	C	may 1,2013	may 5,2013	4
FUM19	C	may 1,2013	may 5,2013	4
FUM20	C	may 1,2013	may 5,2013	4
FUM21	C	may 1,2013	may 5,2013	4
FUM22	C	may 1,2013	may 5,2013	4
FUA20	C	may 1,2013	may 5,2013	4
FUA21	C	may 1,2013	may 5,2013	4
FUA22	C	may 1,2013	may 5,2013	4

FUA23	C	may 1,2013	may 5,2013	4
FUA24	C	may 1,2013	may 5,2013	4
FUA25	C	may 1,2013	may 5,2013	4
ADM11	C	may 5,2013	may 6,2013	1
ADM12	C	may 5,2013	may 6,2013	1
ADM13	C	may 5,2013	may 6,2013	1
ADM14	C	may 5,2013	may 6,2013	1
ADM20	C	may 5,2013	may 6,2013	1

Tabla 7 Planeación por funcionalidad.

2.4 Diseño por funcionalidad

El diseño y construcción de la funcionalidad es un proceso iterativo durante el cual las funcionalidades seleccionadas son producidas. Una iteración puede tardarse desde unos pocos días a un máximo de dos semanas. Este proceso iterativo incluye tareas como inspección del diseño, codificación, prueba unitaria, integración e inspección del código. Luego que la iteración llega a su fin se realiza una construcción de la funcionalidad en el cual esta es integrada.

Principios de diseño

Para el diseño de la interfaz de la aplicación se tuvieron en cuenta algunos tips y axiomas de diseño aplicables al proyecto los cuales se definen en[13]:

- Un cuadro de diálogo es otra habitación. Tenga una buena razón para ir allí.
- Ofrezca directamente suficiente información al usuario para evitar errores.
- No pregunte: el usuario tomó esa decisión.
- No detenga los procesos con información innecesaria.
- Las buenas interfaces de usuario son invisibles: no importa lo bien que esté diseñada, con menos estaría mejor.
- Imagine a los usuarios como muy inteligentes pero muy ocupados: que la aplicación no interfiera en los objetivos del usuario.

- Los usuarios no cometen errores pues siempre tienen la razón: hacer imposible la ejecución de operaciones que impliquen errores.
- Obedecer estándares a menos que se tenga una muy buena razón.
- Optimizar para usuarios intermedios: nadie permanece como usuario principiante por siempre.
- Poner las interacciones primarias en la pantalla principal: el usuario desea sacar provecho de la aplicación ya.
- Las cosas que se comportan de forma diferente se deben ver diferentes: evitar la confusión entre los tipos de operaciones posibles.
- Los cuadros de diálogo deben ser lo más pequeños posibles pero no más.
- Evitar el desplazamiento horizontal de texto.
- Ofrecer opciones de *Aceptar* y *Cancelar* en todos los diálogos.
- Prepárese para el caso más probable.

2.4.1 Esquema de la Base de Datos

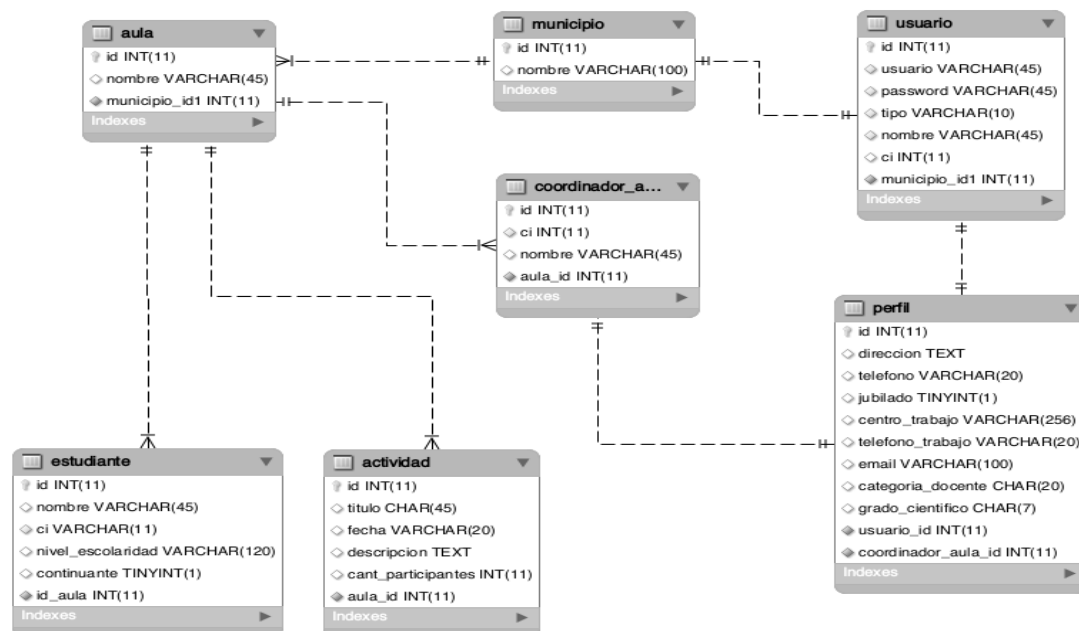


Figura 3: Diagrama de la Base de Datos

2.5 Descripción de la arquitectura propuesta

La arquitectura de la aplicación FÉNIX está basada en el patrón Modelo-Vista-Controlador.

Capa Modelo

El Modelo representa los datos y las operaciones que se realizan sobre ellos. La implementación de esta capa se desarrolló mediante clases de PHP para cada una de las entidades persistentes existentes en el dominio de la aplicación. Cada una de estas clases es una subclase de *CActiveRecord* que a su vez es una subclase de *CModel*.

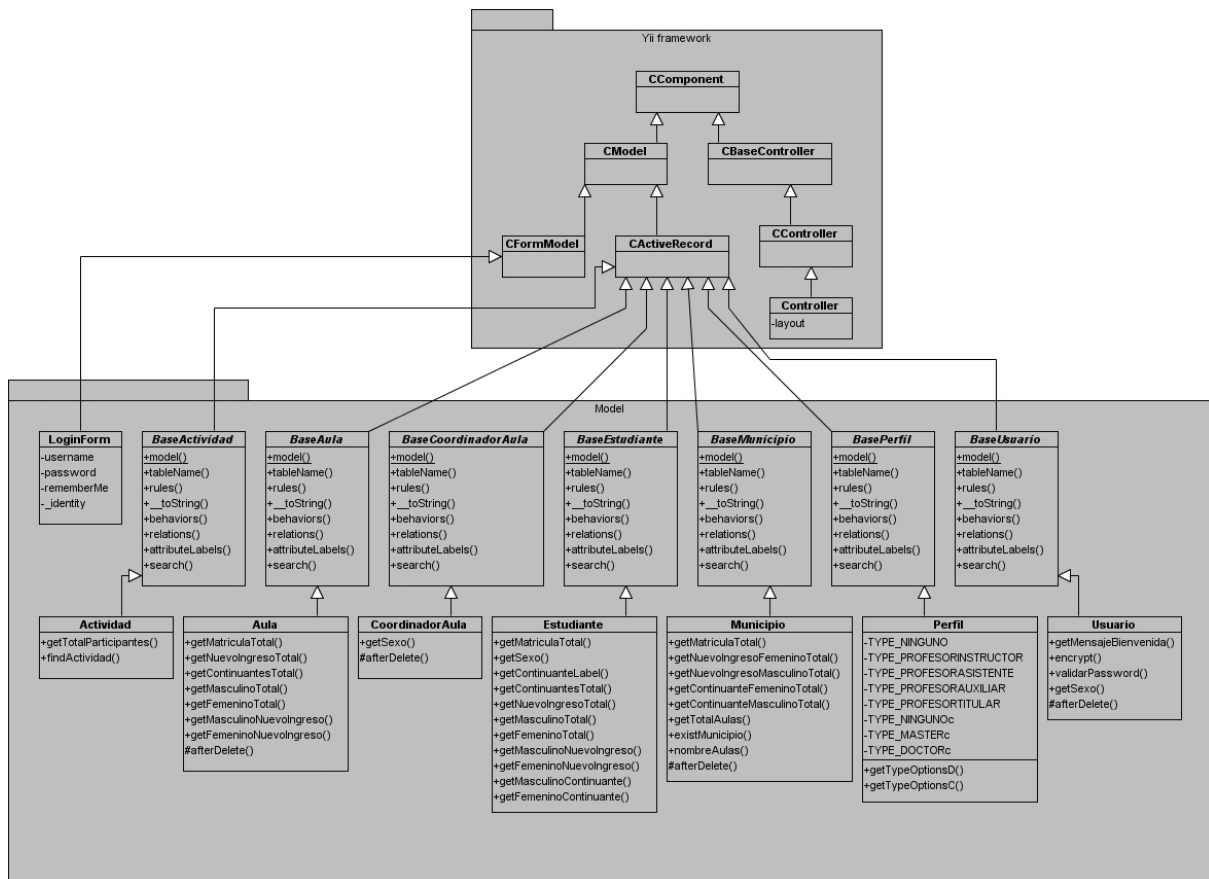


Figura 4: Estructura de la capa modelo

Capa Vista

De acuerdo al rol del usuario en cada momento se pueden presentar dos vistas del mismo modelo.

Vista de invitado

Encargada de mostrar todo el contenido de forma amigable a la navegación.

Vista de usuario

Muestra una interfaz en la que se puede ver el contenido asociado al usuario y a su vez permite editarlo.

Vista de administración

Brinda a un usuario con privilegios una interfaz donde se puede editar todo el contenido y a su vez permite manejar la información referente a los usuarios.

Capa Controlador

La capa Controlador está representada por las ventanas de la interfaz gráfica ya que esta prepara los modelos para su presentación y guía el flujo de la aplicación.

2.6 Diagrama de Despliegue

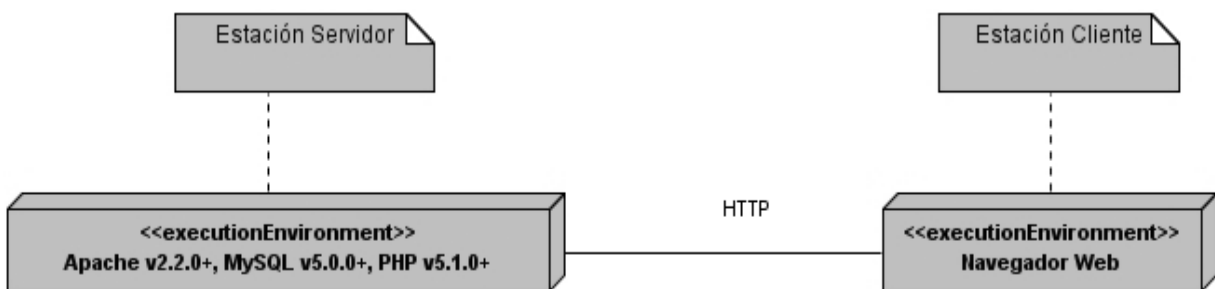


Figura 5: Diagrama de despliegue

2.7 Conclusiones

En este capítulo se describió el proceso de desarrollo de la aplicación según las etapas de la metodología FDD: Se describió el modelo global donde se definieron algunos conceptos utilizados en el modelo de dominio. Se analizó y describió el flujo de los procesos involucrados en la gestión de la información académica que se genera en la CUAM de la UCF.

Capítulo3: Estudio de factibilidad y validación de la solución

En el presente capítulo se aborda el tema asociado al estudio de la factibilidad del sistema, debido a que para la realización de un proyecto es necesario estimar el tiempo de desarrollo que se requiere para la ejecución del mismo, así como los costos asociados, los beneficios tangibles e intangibles que reportaría su elaboración y finalmente el análisis entre los costos y los beneficios para concluir si es o no factible el desarrollo del sistema.

Estas estimaciones pueden realizarse a través del método de puntos de función del modelo de COCOMO II y la validación del producto mediante entrevistas aplicadas a los usuarios a los cuales está destinado el software, procesándose luego en el SPSS para obtener finalmente los resultados.

3.1 Estudio de la factibilidad

3.1.1 Planificación

Se utilizó para el cálculo de la estimación del esfuerzo, el tiempo de desarrollo y el costo del proyecto por el método de puntos de función.

Para realizar el cálculo de los costos de desarrollo del sistema se deben tener primero las instrucciones fuentes. Analizándose para esto las cantidades de entradas, salidas, peticiones, archivos lógicos e interfaces externas preliminares que tiene el sistema.

Después de este estudio se obtuvieron los siguientes resultados:

Funciones de tipos de datos

Nombre	DER	RLR	Valor
municipio	2	1	simple
actividad	5	1	simple

aula	2	1	simple
estudiante	5	1	simple
coordinador_aula	3	1	simple
perfil	9	2	simple

Tabla 8 Archivos lógicos internos

Funciones de transacción

Nombre	DER	RLR	Valor
FUM01	1	1	simple
FUA01	1	1	simple
FUE01	4	1	simple
FUT01	5	1	simple
FUC01	2	2	simple
FUC02	2	2	simple
FUC13	3	1	simple
ADM02	6	2	medio
ADM15	1	1	simple
FUC05	11	2	medio
ADM01	3	1	simple

Tabla 9 Entradas externas

Nombre	DER	RLR	Valor
FUM05	1	1	simple
FUA04	1	1	simple
FUE02	4	1	simple
FUT05	5	1	simple
FUC04	11	2	medio

FUC12	3	1	simple
FUC11	11	2	medio
ADM03	6	2	medio
ADM04	3	1	simple
ADM16	1	1	simple

Tabla 10 Salidas externas

Nombre	DER	RLR	Valor
FUM02	1	5	medio
FUM03	1	1	simple
FUM04	1	3	simple
FUM06	1	3	simple
FUM07	1	3	simple
FUM08	1	3	simple
FUM09	1	3	simple
FUM10	1	3	simple
FUM11	1	2	simple
FUM12	1	5	medio
FUM13	1	1	simple
FUM14	1	3	simple
FUM15	1	3	simple
FUM17	1	1	simple
FUM18	1	1	simple
FUM19	1	1	simple
FUM20	1	1	simple
FUM21	1	1	simple
FUA02	1	4	medio
FUA03	1	1	simple

FUA05	1	3	simple
FUA06	1	2	simple
FUA07	1	2	simple
FUA08	1	1	simple
FUA09	1	1	simple
FUA10	1	1	simple
FUA11	1	1	simple
FUA12	1	1	simple
FUA13	1	1	simple
FUA14	1	1	simple
FUA15	1	5	medio
FUA17	1	2	simple
FUA18	1	1	simple
FUA19	1	2	simple
FUA20	1	1	simple
FUA21	1	1	simple
FUA22	1	1	simple
FUM22	1	1	simple
FUA23	1	1	simple
FUA24	1	1	simple
FUA25	1	1	simple
FUE03	1	1	simple
FUE04	1	1	simple
FUE05	1	1	simple
FUE06	1	1	simple
FUE07	1	1	simple
FUE08	1	1	simple

FUT02	1	1	simple
FUT03	1	1	simple
FUT04	1	1	simple
FUT06	1	1	simple
FUT07	1	1	simple
FUT08	1	1	simple
FUT09	1	1	simple
FUC03	1	2	simple
FUC06	1	2	simple
FUC07	1	2	simple
FUC08	1	2	simple
FUC09	1	2	simple
FUC10	1	3	simple
ADM05	1	2	simple
ADM06	1	1	simple
ADM07	1	1	simple
ADM08	1	1	simple
ADM09	1	1	simple
ADM10	1	1	simple
ADM11	1	1	simple
ADM12	1	1	simple
ADM13	1	1	simple
ADM14	1	1	simple
ADM17	1	7	medio
ADM18	1	1	simple
ADM19	1	1	simple

Tabla 11 Consultas externas

Tipo de Función		Complejidad	Valor del factor		Total por complejidad	Suma
Archivos Lógicos Internos		Simple(s)	7	6		42
		Medio(s)	10	0		
		Complejo(s)	15	0		
Archivos de Interface Externas		Simple(s)	5	0		0
		Medio(s)	7	0		
		Complejo(s)	10	0		
Entradas Externas		Simple(s)	3	5		35
		Medio(s)	4	3		
		Complejo(s)	6	0		
Salidas Externas		Simple(s)	4	5		43
		Medio(s)	5	3		
		Complejo(s)	7	0		
Consultas Externas		Simple(s)	3	26		224
		Medio(s)	4	3		
		Complejo(s)	6	0		
		Total		100		344

Tabla 12 Puntos de función brutos

Características generales del sistema	Nivel de influencia
Comunicación de datos	4
Procesamiento distribuido	1
Desempeño	1
Configuración del equipamiento	0
Volumen de transacciones	0
Entrada de datos on-line	0
Interfase con el usuario	2

Actualización on-line	4
Procesamiento complejo	0
Reusabilidad	1
Facilidad de implementación	0
Facilidad de operación	0
Múltiples locales	3
Facilidad de cambios	3
Nivel de influencia	19

Tabla 13 Factor de ajuste

El factor de ajuste se calcula mediante la fórmula:

$$\text{Factor de ajuste} = (\text{Nivel de influencia} * 0,01) + 0,65$$

$$\text{Factor de ajuste} = 0.81$$

Puntos de Función Ajustados

$$\text{Puntos de función} = \text{Puntos de función brutos} * \text{Factor de Ajuste}$$

$$\text{Puntos de función} = 344 * 0.81$$

$$\text{Puntos de función} = 278.64000000000004$$

3.1.2 Productividad en el desarrollo de una aplicación

La productividad en el desarrollo de una aplicación se traduce como la velocidad con la que la aplicación fue construida, esto es, cuántas unidades de tamaño (PF) fueron construidas en una unidad de tiempo o cuantas unidades de tiempo fueron consumidas para realizar una unidad de software.

$$\text{Productividad} = \text{Tiempo} / \text{Puntos de función}$$

$$\text{Productividad} = 2.0671834625322996$$

3.1.3 Esfuerzo para el desarrollo de una aplicación

El esfuerzo necesario para desarrollar una aplicación puede ser definido como la cantidad de horas de trabajo necesarias para desarrollar una aplicación

$\text{Esfuerzo} = \text{Productividad} * \text{Tamaño del software}$

Esfuerzo = 576.0

3.1.4 Costo de la aplicación

El costo de la aplicación se define como la multiplicación del número de personas involucradas, el salario básico promedio en días laborables y el tiempo establecido.

$\text{Costo} = \text{Hombres} * \text{Salario promedio} * \text{Tiempo}$

Costo = 1440.0

Concepto	Valor
Tiempo de desarrollo	3 meses
Cantidad de hombres	1
Salario medio	\$480.00
Costo asociado al salario básico	\$1440.00

Tabla 14 Resumen del estudio de factibilidad.

3.2 Beneficios tangibles e intangibles.

Entre los beneficios tangibles obtenidos con la realización del software se pueden mencionar un ahorro sustancial de recursos humanos y materiales de oficina, lo que permite una disminución del nivel de errores y los retrasos en el logro de los resultados finales.

Todo lo anteriormente mencionado puede resumirse en la posibilidad de gestionar rápida y eficientemente la información asociada a los procesos ya mencionados, siendo la misma segura y confiable, brindando de manera significativa ayuda a los coordinadores de la CUAM. La aplicación y el ahorro de tiempo constituyen los

beneficios intangibles que además se traducen en un ahorro económico considerable permitiendo que los coordinadores dediquen menos tiempo en este proceso y que un trabajo que antes requería de varias horas sea terminado en poco tiempo.

3.3 Análisis de costos y beneficios

Al desarrollo de todo producto informático va asociado un costo, el justificarlo depende de los beneficios tangibles e intangibles que produce.

La incorporación del sistema propuesto traerá grandes beneficios como se evidenció en el epígrafe anterior. Permitirá entre otros reducir la pérdida de información por deterioro de documentación, mayor rapidez y confiabilidad del proceso, así como la obtención de reportes y gráficos de fácil entendimiento para los usuarios. El desarrollo no supone grandes gastos de recursos y las tecnologías empleadas no requieren pagos de licencias.

3.4 Validación de la solución

Para la validación del software se aplicó una encuesta a los usuarios finales del mismo. El cuestionario fue diseñado cumpliendo los requisitos de presentación, motivación y preguntas claras simples y con secuencia lógica.

La encuesta fue revisada antes de su aplicación por profesores de categoría docente con vistas a revisar la redacción, el enfoque de las preguntas y el cumplimiento del objetivo general que se planteó. Las preguntas se diseñaron tomando en consideración requisitos de presentación, longitud adecuada, secuencia lógica y terminología.

3.4.1 Resultados de la aplicación de la encuesta

Se encuestaron a 40 personas entre coordinadores provinciales, coordinadores de municipio y otros trabajadores de la CUAM, repartiéndose la muestra en 1 coordinador provincial, 8 municipales y 31 trabajadores de la CUAM.

Para procesar los resultados se utilizó el paquete estadístico SPSS v19 obteniéndose los resultados que se muestran a continuación:

Utilidad como sistema de gestión académica en la CUAM

En esta pregunta las respuestas de los usuarios oscilaron entre los valores Bueno y Muy Bueno, con un porcentaje de 45% para el primer caso y de 55% para el segundo.

		Frecuencia	Porcentaje
Válidos	Bueno	18	45,0
	Muy Bueno	22	55,0
	Total	40	100,0

Tabla 15 Utilidad como sistema de gestión académica en la CUAM

Utilidad en el apoyo al trabajo administrativo de la CUAM

La segunda pregunta es sobre la utilidad del software con respecto al trabajo administrativo, siendo las respuestas de Bueno con un valor de 22.5% y las de Muy Bueno con 77.5%.

		Frecuencia	Porcentaje
Válidos	Bueno	9	22,5
	Muy Bueno	31	77,5
	Total	40	100,0

Tabla 16 Utilidad en el apoyo al trabajo administrativo de la CUAM

Existencia de un software similar

Esta pregunta es sobre si los usuarios conocen otro software, con un porcentaje de 12.5% en el primer caso y de 87.5% para el segundo.

		Frecuencia	Porcentaje
Válidos	No	5	12,5

	Si	35	87,5
	Total	40	100,0

Tabla 17 Existencia de un software similar

Facilidad de uso con respecto a otros sistemas

La cuarta pregunta es sobre el Sistema de gestión académica presentado, las respuestas posibles eran: Es más fácil de usar, Es igual a los otros, Es más difícil o Es único, siendo el 85% de las respuestas Es más fácil de usar y el 15% Es único.

		Frecuencia	Porcentaje
Válidos	Es Novedoso	31	77,5
	Tiene Mejoras	9	22,5
	Total	40	100,0

Tabla 18 En cuanto al uso respecto a otros sistemas

En cuanto al uso respecto a otros sistemas

La quinta pregunta es sobre el uso del sistema, las respuestas posibles eran: Es novedoso, Tiene mejoras, Es igual o Es más malo, siendo el 77.5% de las respuestas Es novedoso y el 22.5% Tiene mejoras.

		Frecuencia	Porcentaje
Válidos	Es más fácil de usar que otros	6	15,0
	Es único	34	85,0
	Total	40	100,0

Tabla 19 Facilidad de uso con respecto a otros sistemas

En cuanto a la presentación

La sexta pregunta es sobre la presentación del sistema, las respuestas posibles eran: Muy Bueno, Bueno, Regular o Malo, siendo el 80% de las respuestas Muy bueno y el 20% Bueno.

		Frecuencia	Porcentaje
Válidos	Bueno	8	20,0
	Muy Bueno	32	80,0
	Total	40	100,0

Tabla 20 En cuanto a la presentación

En que radican las ventajas

La séptima pregunta es sobre las ventajas que ofrece el sistema, las respuestas posibles eran: En la entrada de datos, En la facilidad de búsqueda, En la impresión, En la calidad de la aplicación, En todas las anteriores o No tiene ventajas, siendo el 15.0% en la impresión, el 35.0% En la calidad de la aplicación, y el 50% En todas las anteriores.

		Frecuencia	Porcentaje
Válidos	En la Impresión	6	15,0
	En la calidad de la Aplicación	14	35,0
	En todas las anteriores	20	50,0
	Total	40	100,0

Tabla 21 En que radican las ventajas

Valoración del sistema

La octava pregunta es para otorgarle una evaluación al sistema, que puede oscilar entre los valores desde 1 hasta 5, siendo el 16.7% de valor 4 y el 83.3% de valor 5.

		Frecuencia	Porcentaje
Válidos	Bueno	8	20,0
	Muy Bueno	32	80,0
	Total	40	100,0

Tabla 22 Valoración del sistema

La utilización del sistema propuesto ofrece ventajas, las cuales fueron obtenidas mediante la encuesta realizada y se relacionan a continuación:

1. El software presenta gran utilidad para el trabajo con la información académica en la CUAM.
2. Es novedoso.
3. La presentación es buena.
4. La información se presenta de forma legible.
5. El objeto de interés del usuario es fácil de identificar.
6. Navegación fácil.
7. Presenta facilidad en la entrada de datos.
8. Presenta facilidades de búsqueda.
9. Presenta versiones imprimibles para los reportes de interés.
10. Los errores en el procesamiento de la información son mínimos lo que lo convierte en un sistema confiable.

3.5 Conclusiones

El presente capítulo ha analizado la factibilidad de desarrollar el software con los recursos humanos y en el tiempo estimado, resultando factible implementarlo en 3 meses, con una persona a un costo de \$1440.00 por concepto de salario básico.

Además para la validación del sistema se aplicó un sistema de encuesta usando una escala del tipo Likert para el sistema de evaluación y para procesar la información se utilizó el paquete estadístico SPSS v15 con 13 variables involucradas a 19 casos diferentes, arrojando como resultado una puntuación media de 5 puntos de los ítems puestos a consideración.

Conclusiones

En la Universidad de Cienfuegos el proceso de gestión de la información académica en la Cátedra Universitaria del Adulto Mayor genera gran volumen de información, haciendo difícil el trabajo con la misma, y el procesamiento de una gran cantidad de datos e informes a imprimir.

En el desarrollo de este trabajo se plantearon los conceptos asociados al campo de acción, se seleccionaron las metodologías, herramientas, tecnologías y lenguajes más adecuados para el desarrollo del sistema.

Se diseñó una interfaz siguiendo los parámetros de diseño planteados, ofreciéndose un ambiente amigable a los usuarios que interactúan con el sistema.

Se implementó un sistema informático que se adecua a las necesidades propias de la gestión de la información académica en las cátedras universitarias del adulto mayor para dar cumplimiento a los requerimientos funcionales planteados.

Los resultados obtenidos con la encuesta aplicada para la validación del sistema a los principales usuarios asociados al dominio fueron procesados con el paquete estadístico SPSS. Los mismos corroboraron la validez del sistema, que fue referido como muy útil, rápido, confiable y seguro.

Recomendaciones

A pesar de que los objetivos trazados con la realización de este trabajo fueron cumplidos, los desarrolladores del mismo sugieren tomar esta propuesta solo como la primera fase de un proyecto mucho más ambicioso y continuar esta investigación.

Se recomienda entonces que:

1. Continuar con el estudio del flujo de trabajo de la Cátedra Universitaria del Adulto Mayor para incorporar todos los aspectos referidos a esta, así como profundizar en los tratados con mayor detalle.
2. Mantener la concepción base utilizada de los servicios web para garantizar la interoperabilidad con otros módulos de otros sistemas y la incorporación del mismo a la plataforma SIGENU.
3. Ampliar el modelo de dominio del sistema para poder manejar la información de las cátedras de otras provincias.

Referencias Bibliográficas

- [1] P. Aranimar, *Acercamiento conceptual a la situación del adulto mayor en América Latina*. 2001.
- [2] «Crece la población de 75 años y más en América Latina y el Caribe | Portal de Salud y Envejecimiento en las Américas», 22-nov-2012. [Online]. Available: <http://www.mayoressaludables.org/focos/crece-la-poblacion-de-75-anos-y-mas-en-america-latina-y-el-caribe>. [Accessed: 22-nov-2012].
- [3] P. Bartle, «Información para la gestión y gestión de la información», 22-nov-2012. [Online]. Available: <http://www.scn.org/mpfc/modules/mon-miss.htm>.
- [4] *Sistema de soporte para la toma de decisiones*. .
- [5] «Introducción a UML. Programación en Castellano.», 01-abr-2013. [Online]. Available: http://www.programacion.com/articulo/introduccion_a_uml_181. [Accessed: 01-abr-2013].
- [6] «PHP - EcuRed», 01-abr-2013. [Online]. Available: <http://www.ecured.cu/index.php/Php>. [Accessed: 01-abr-2013].
- [7] «jQuery Wiki», 23-feb-2007. [Online]. Available: http://docs.jquery.com/Main_Page. [Accessed: 23-feb-2007].
- [8] «MySQL - EcuRed», 01-abr-2013. [Online]. Available: <http://www.ecured.cu/index.php/MySQL>. [Accessed: 01-abr-2013].
- [9] «NetBeans - EcuRed», 01-abr-2013. [Online]. Available: <http://www.ecured.cu/index.php/NetBeans>. [Accessed: 01-abr-2013].
- [10] «Capítulo 1. Introducción», 01-abr-2013. [Online]. Available: <http://docs.gimp.org/es/introduction.html>. [Accessed: 01-abr-2013].
- [11] A. Molpeceres, *Procesos de desarrollo: RUP, XP y FDD*. .
- [12] G. B. I. Jacobson, y J. Rumbaugh, *El Proceso Unificado de Desarrollo de Software*. Addison Wesley, 1999.
- [13] *About Face: The Essentials of User Interface Design*. .

Bibliografía

- [1] «.:Web del Adulto Mayor.: - El Adulto Mayor», 22-nov-2012. [Online]. Available:
http://www.vcl.jovenclub.cu/prov/adultomayor/index.php?option=com_content&task=view&id=3. [Accessed: 22-nov-2012].
- [2] About Face: The Essentials of User Interface Design. .
- [3] P. Aranimar, Acercamiento conceptual a la situación del adulto mayor en América Latina. 2001.
- [4] «ADULTO MAYOR:», 22-nov-2012. [Online]. Available:
<http://www.ubiobio.cl/vitrina/envejecim.htm>. [Accessed: 22-nov-2012].
- [5] Jeffery Winesett, Agile Web Application Development with Yii 1.1 and PHP5. .
- [6] «Bootstrap», Bootstrap. [Online]. Available: <http://github.com/twitter/bootstrap>.
- [7] «Capítulo 1. Introducción», 01-abr-2013. [Online]. Available:
<http://docs.gimp.org/es/introduction.html>. [Accessed: 01-abr-2013].
- [8] «CEPAL - Acercamiento conceptual a la situación del adulto mayor en América Latina», 22-nov-2012. [Online]. Available: <http://www.eclac.org/cgi-bin/getProd.asp?xml=/publicaciones/xml/0/9260/P9260.xml&xsl=/celade/tpl/p9f.xsl&base=/celade/tpl/top-bottom.xsl>. [Accessed: 22-nov-2012].
- [9] «Crece la población de 75 años y más en América Latina y el Caribe | Portal de Salud y Envejecimiento en las Américas», 22-nov-2012. [Online]. Available: <http://www.mayoressaludables.org/focos/crece-la-poblacion-de-75-anos-y-mas-en-america-latina-y-el-caribe>. [Accessed: 22-nov-2012].
- [10] «CSS - EcuRed», 01-abr-2013. [Online]. Available:
<http://www.ecured.cu/index.php/CSS>. [Accessed: 01-abr-2013].
- [11] «El Adulto Mayor en el Mundo Actual», 22-nov-2012. [Online]. Available:
<http://www.psicologia-online.com/articulos/2012/adulto-mayor-en-el-mundo-actual.html>. [Accessed: 22-nov-2012].
- [12] G. B. I. Jacobson, y J. Rumbaugh, El Proceso Unificado de Desarrollo de Software. Addison Wesley, 1999.

- [13] «Epidata consulting.», Introducción a UML 2.0., 15-nov-2012. [Online]. Available: http://www.epidataconsulting.com/tikiwiki/tiki-read_article.php?articleId=15. [Accessed: 15-nov-2012].
- [14] «FDD Ingenieria de Software», 26-feb-2013. [Online]. Available: http://ingenieriadesoftware.mex.tl/61162_FDD.html. [Accessed: 26-feb-2013].
- [15] P. Bartle, «Información para la gestión y gestión de la información», 22-nov-2012. [Online]. Available: <http://www.scn.org/mpfc/modules/mon-miss.htm>.
- [16] «Introducción a UML. Programación en Castellano.», 01-abr-2013. [Online]. Available: http://www.programacion.com/articulo/introduccion_a_uml_181. [Accessed: 01-abr-2013].
- [17] R. Álvarez, Introducción al HTML. 2010.
- [18] «jQuery Wiki», 23-feb-2007. [Online]. Available: http://docs.jquery.com/Main_Page. [Accessed: 23-feb-2007].
- [19] Manual de PHP. .
- [20] Amaro Calderón y Sarah Dámaris, Metodologías Ágiles. Trujillo-Perú: , 2007.
- [21] «Mundo Tercera Edad», 22-nov-2012. [Online]. Available: <http://www.mundoterceraedad.cl/>. [Accessed: 22-nov-2012].
- [22] «MySQL - EcuRed», 01-abr-2013. [Online]. Available: <http://www.ecured.cu/index.php/MySQL>. [Accessed: 01-abr-2013].
- [23] «NetBeans - EcuRed», 01-abr-2013. [Online]. Available: <http://www.ecured.cu/index.php/NetBeans>. [Accessed: 01-abr-2013].
- [24] «Orientacion al Adulto Mayor - EcuRed», 22-nov-2012. [Online]. Available: http://www.ecured.cu/index.php/Orientacion_al_Adulto_Mayor. [Accessed: 22-nov-2012].
- [25] «PHP - EcuRed», 01-abr-2013. [Online]. Available: <http://www.ecured.cu/index.php/Php>. [Accessed: 01-abr-2013].
- [26] A. Molpeceres, Procesos de desarrollo: RUP,XP y FDD. .
- [27] «Programa de Atención al Adulto Mayor en Cuba», 22-nov-2012. [Online]. Available: <http://www.sld.cu/instituciones/gericuba/paamc/index.htm>. [Accessed: 22-nov-2012].
- [28] «Proyecto de Lineamientos de la Política Económica y Social.» .

- [29] «Qué es el adulto mayor?», 22-nov-2012. [Online]. Available: <http://www.misrespuestas.com/que-es-el-adulto-mayor.html>. [Accessed: 22-nov-2012].
- [30] «Red Latinoamericana de Gerontología», 22-nov-2012. [Online]. Available: <http://www.gerontologia.org/pram.php>. [Accessed: 22-nov-2012].
- [31] Karenny Brito Acuña, «Selección de Metodologías de Desarrollo para Aplicaciones Web en la Facultad de Informática de la Universidad de Cienfuegos.», Universidad de Cienfuegos«Carlos Rafael Rodríguez», Cienfuegos,Cuba, 2009.
- [32] Sistema de soporte para la toma de decisiones. .
- [33] «Tercera edad. Necesidad de educación en el adulto mayor - Monografias.com», 22-nov-2012. [Online]. Available: <http://www.monografias.com/trabajos23/tercera-edad-educacion/tercera-edad-educacion.shtml>. [Accessed: 22-nov-2012].
- [34] Qiang Xue y Xiang Wei Zhuo, The Definitive Guide to Yii 1.1. .
- [35] «Un resumen de la metodología ágil FDD | Javier Garzás», 26-feb-2013. [Online]. Available: <http://www.javiergarzas.com/2012/09/metodologia-gil-fdd-1.html>. [Accessed: 26-feb-2013].
- [36] «Yii Framework Forum», Yii Framework Forum. [Online]. Available: <http://www.yiiframework.com/forum/>.
- [37] Yii software LLC, Yii framework v1.1.13-RC Class Referece. .

Anexos

Anexo A: Encuesta

Estimado Usuario la presente encuesta forma parte de la Validación de un Sistema Informático para un Trabajo de Diploma en la carrera de Ingeniería Informática.

Muchas Gracias por su participación.

Usuario: Profesor: ____ Estudiante: ____ Trabajador Docente: ____

1.- Utilidad del Producto Informático:

a.- Como Sistema de gestión para la información académica de las CUAM:

Muy Buena: ____ Buena: ____ Regular: ____ Mala: ____

b.- Como apoyo al trabajo Administrativo de las CUAM en la Universidad.

Muy Buena: ____ Buena: ____ Regular: ____ Mala: ____

2.- Relacionado con otras Aplicaciones Informáticas para la Gestión de la información académica de las CUAM.

a.- Conoce usted algún otro gestor de información académica: Si: ____ No: ____

b- En cuanto al presente Sistema FÉNIX:

Es más fácil de usar que otros: ____ Es igual a los otros: ____ Es más difícil: ____ Es único: ____

c.- En cuanto al uso:

Es Novedoso: ____ Tiene Mejoras: ____ Es Igual: ____ Es más malo: ____

d.- En cuanto a la presentación:

Muy Buena: ____ Buena: ____ Regular: ____ Mala: ____

3.- En qué radican las ventajas:

En la Entrada de Datos: ____ En la Facilidad de la búsqueda: ____

En la Impresión: ____ En la calidad de la Aplicación: ____

En todas las anteriores. ____ No tiene ventajas. ____

4.- Si usted lo fuera a valorar en una escala de 5 cuántos puntos le daría al Sistema____

5.- Algún comentario al respecto: