



Trabajo de diploma para optar por el título de Ingeniería en
Informática.

Título: "Sitio web para la gestión de la información del
Departamento de Marxismo e Historia de la UNISS."

Autor: Carlos Orlando Hernández Fernández

Tutor: Ing. Yunior Rafael Cabrera Hernández

Consultante: Ing. Yainel García Alfonso

AGRADECIMIENTOS:

- ✚ A mis abuelos Carlos y Gladis que a pesar de estar tan lejos siempre los llevo en el corazón.
- ✚ A mi abuela Rita, a mi mujer Rosalí y a su madre por estar a mi lado en las buenas y en las malas además de creer en mí.
- ✚ A mi mamá y mis hermanas por formar parte de mi vida.
- ✚ A mi abuelo Ramón y a Mery por sus sabios consejos.
- ✚ A mi papá y a mis hermanos por apoyarme.
- ✚ A mis amigos en especial a Ernesto que me dedicó tiempo y conocimientos, a Pito que fue una tortura sentarme al lado del aunque faltaba mucho al aula, a Orelvito, al WAKA, a Yuri, a Willi, Andi, al Perrón y a su novia Yemeisy, al Biko, a Yunion la rata, a Ciroco Cocote, al Juanka rollete, a todos los que han sido fundamentales para el éxito de mi carrera y de este trabajo.
- ✚ A mi tutor Yunion y a Nane.
- ✚ A todos los profesores y profesionales por el ejemplo y los conocimientos brindados.
- ✚ A todos aquellos que de una manera u otra han contribuido con su apoyo.

DEDICATORIA:

*A mis abuelos Carlos y Ramón y a mi tía Elena
que aunque ya no pueda verlos y no se encuentren
entre nosotros siempre los llevaré en el corazón.*

Resumen:

El Departamento de Marxismo e Historia de la Uniss (Universidad de Sancti Spíritus) es una entidad prestadora de servicios y como parte de su trabajo muestra información relacionada con las tareas y eventos que este lleva a cabo, con los servicios que se brindan a estudiantes y profesores y con los acuerdos de la organización. La presente investigación tiene como objetivo desarrollar una Aplicación Web que contribuya a la inserción Departamento de Marxismo e Historia de la Uniss. Además describe todas las etapas que son necesarias para la automatización de la aplicación.

Para dar cumplimiento a estos objetivos y como parte del proceso de informatización surge la necesidad de contar con una aplicación web que permitirá solucionar las limitaciones que presenta el Departamento de Marxismo e Historia de la Uniss.

Quedan descritos en la investigación los elementos que conforman los flujos de trabajo, de acuerdo a lo que establece el Proceso Unificado de Desarrollo de Software (RUP); y como lenguaje de modelado, el UML (Lenguaje Unificado de Modelado); lo cual permitió la adecuada documentación del análisis, diseño e implementación de la solución. Se implementó utilizando el framework Symfony 2.2, lenguaje PHP para la lógica del negocio y JavaScript para la interfaz de usuario, lo que permitió un código eficiente y un agradable diseño, adecuado a las especificaciones del cliente. Se utilizó además como herramienta de modelado, el Rational Rose 7.

Abstract

The Marxism and History Department of the Uniss, sign is a lending the services and part entity of its work related information with the tasks and events that this he implies to end, with the services that offer students themselves and professors and with the agreements of the organization.

This letter he has investigation as objective to develop an application Web that contribute to the insert the Marxism and story department of her Uniss, besides he describes all the stages that are necessary for the automatization of the application.

To give fulfillment to these objectives and the need to count with an application Web that will be allowed to solve the limitations happens as part of the process of computerization that shows the Marxism and story department of her Uniss.

They are left described in the investigation the elements that conform the work flows, that he establishes the united the development process of software according, and as language of modeling, the UML, it as he made possible the adequate documentation of the analysis, design and implementation of the solution.

It took effect utilizing the framework Symfony 2.2, language PHP for the logic of the business and JavaScript for the interface of user, it that he permitted an efficient code and a pleasant design, made suitable to the specifications of the customer, it was utilized besides as tool of modeling, the Rational Rose 7.

INTRODUCCIÓN	9
CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA.....	14
Introducción.....	14
1.1 Tecnología de la Información.....	14
1.2 Breve reseña sobre los orígenes de la Web	15
1.3 Departamento de Marxismo e Historia de la UNISS.....	16
1.3.1 Descripciones Generales.	16
1.3.2 Situación problemática.....	17
1.4 Fundamentación de la metodología utilizada.....	18
1.4.1 Lenguaje Unificado de Modelado (UML).....	18
1.4.2 Proceso Unificado Racional (RUP).....	18
1.5 Tendencias y tecnologías actuales	19
1.5.1 Arquitectura Cliente/Servidor. Arquitectura en N capas	21
1.6 Lenguajes de programación	23
1.6.1 Lenguajes de programación web.....	23
1.6.2 PHP	23
1.6.3 HTML	24
1.6.4 JavaScript.....	24
1.7 Servidores Web	26
1.7.1 Apache.....	26
1.8 Herramientas de desarrollo.....	27
1.8.1 Rational Rose.....	27
1.9 Framework.....	27
1.9.1 Symfony	28
1.9.2 Doctrine.....	29

1.10 Sistemas Gestores de Base de Datos (SGBD)	30
1.10.1 SQL (StructureQueryLanguage).....	31
1.10.2 SQL Server	31
1.10.3 PostgreSQL	32
1.10.4 MySQL	32
1.11 ¿Por qué utilizar Apache-PHP-MySQL?.....	34
CONCLUSIONES.....	35
CAPITULO 2: DESCRIPCIÓN DE LA SOLUCIÓN PROPUESTA.....	36
2.1 Reglas del Negocio a Considerar.....	36
2.2 Descripción del modelo del negocio.....	37
2.2.1 Descripción de los procesos del negocio	37
2.2.2 Modelo de casos de uso del negocio.....	38
2.2.3 Actores del negocio.	38
2.2.4 Diagrama de casos de uso del negocio	39
2.2.5 Trabajadores del negocio.....	39
2.2.6 Descripción de los casos de uso del negocio.....	40
2.2.7 Diagrama de actividades del negocio.	41
2.2.8 Diagrama de clases del modelo de objetos.....	42
2.3 Requerimientos funcionales.....	42
2.4 Requerimientos no funcionales.....	44
2.5 Descripción del sistema propuesto.....	48
2.5.1 Concepción general del sistema.....	48
CONCLUSIONES.....	54
CAPÍTULO III. IMPLEMENTACIÓN.....	55
3.1 Introducción.....	55
3.2 Diagrama de clases del diseño.....	55
3.2.1 Paquete de Seguridad	56

3.2.2 Paquete de Gestión	57
3.2.3 Paquete de Reportes	64
3.3 Diseño de la base de datos	72
3.3.1 Modelo lógico de datos	72
3.3.2 Modelo físico de datos	73
3.4 Principios de diseño	74
3.4.1 Estándares en la interfaz de la aplicación	74
3.4.2 Formatos de reportes.....	75
3.4.3 Concepción general de la ayuda	75
3.4.4 Tratamiento de errores	76
3.4.5 Seguridad.....	76
3.5 Modelo de implementación.....	76
3.5.1 Modelo de despliegue.....	76
3.5.2 Modelo de Componentes	77
CONCLUSIONES.....	78
CONCLUSIONES GENERALES.....	78
RECOMENDACIONES:.....	80
BIBLIOGRAFÍA:	81
GLOSARIO DE TÉRMINOS	86
ANEXOS:.....	88

INTRODUCCIÓN

La automatización de la gestión de la información se ha convertido en una herramienta imprescindible para empresas e instituciones. Las actuales generaciones transitan en una época matizada por la información, donde se habla comúnmente de las TIC (Tecnología en la Información y las Comunicaciones), de ahí que se afirme que vivimos en “la era de la información”. Muchas son las definiciones que se han referido a las TIC, una de ellas sostiene que son “aquellas herramientas y métodos empleados para recabar, retener, manipular o distribuir información” (Bologna&Walsh, 1997). De modo que generalmente se asocian con las computadoras y las tecnologías afines.

Para muchas instituciones, la información y la tecnología, representan sus activos más valiosos. Es por ello que las mismas han incrementado sus expectativas relacionadas con la entrega de servicios de las Tecnologías de la Información (TI). Necesitándose que dichos servicios presenten incrementos en calidad, funcionalidad y facilidades de uso, así como una mejora continua y disminución de los tiempos de entrega.

En la actualidad es indispensable disponer de las herramientas necesarias que cumplan con las mejores prácticas para lograr una correcta Gestión de Servicios de las TI en especial aquellos relacionados con el soporte y provisión. Nuestro país se ha ido insertando en el desarrollo y utilización de las Tecnologías de Información en la sociedad y la economía, potenciando sobre todo la utilización de la informática como herramienta fundamental en la gestión empresarial, debido a la eficiencia que las mismas ofrecen en los procesos de las organizaciones.

Generalmente, en estos departamentos se utilizan aplicaciones que no cumplen con los estándares internacionales para la Gestión de Servicios de las TI y por tanto el trabajo de los departamentos de las TI, responsables de brindar los servicios de soporte y provisión en especial los relacionados con la gestión de incidencias y problemas de las TI, carece de la calidad requerida.

Dentro de la realidad práctica en la Educación Superior en Cuba en la actualidad el trabajo político ideológico destaca por su importancia, los acuerdos emanados del sexto Congreso de Partido y la primera Conferencia del Partido señalan la importancia conseguida a la formación de un profesional altamente preparado y comprometido con

la Revolución, a estos efectos los Departamentos de Marxismo o equivalentes que funcionan o que existen en las universidades del país cumplen una importante misión al sectorizar y asesorar la materialización del trabajo político ideológico y la docencia vinculada o asociada a la enseñanza del marxismo y la historia de Cuba.

En la Universidad José Martí Pérez de la provincia de Sancti Spíritus al igual que en el resto de las universidades la producción científica es una prioridad para los docentes, este proceso genera una imperiosa necesidad, el intercambio científico y la divulgación de los resultados, tal situación no se realiza con facilidad pues la localización de sitios que lo permitan no es accesible a todos.

El Departamento de Marxismo e Historia de la UNISS brinda su servicio a todas las carreras de la Universidad, con un total de 30 profesores que atienden alrededor de 1500 docentes tanto en las FUM (Filial Universitaria Municipal) como en la Sede Central. Si logramos cumplir los objetivos propuestos se puede pensar en el cumplimiento de los acuerdos tomados en el sexto Congreso en la enseñanza superior, donde los estudiantes tienen su tiempo de máquina y estas están habilitadas para el soporte de sus actividades y la prestación de los servicios que se les brindan, no obstante, tampoco han estado favorecidos de contar con un sistema informático basado en las mejores prácticas para la gestión de los servicios de las tecnologías de la información, dificultando así la normalización de los procesos de soporte y provisión de información a dichos docentes, tampoco ha sido posible la uniformidad en la generación de informaciones sobre la prestación de estos servicios, provocando inconsistencia y falta de veracidad en las informaciones ofrecida a los diferentes niveles.

Por otra parte no es posible realizar una atención personalizada que necesitan los usuarios del departamento, lo que implica que se realicen muchos reportes que pudieran ser solucionados inmediatamente con dicha atención y que actualmente distorsionan las estadísticas de la gestión de servicios.

En el presente el papel de las nuevas tecnologías principalmente el uso de las redes y las comunicaciones cobra una vital importancia con espacio de socialización cultural por tanto es inconcebible una plena efectividad de la labor político ideológica sin la presencia en el espacio virtual.

La Misión del Departamento consiste en contribuir a la consolidación de la comunidad universitaria con un claustro altamente categorizado, comprometido con la Revolución y con los valores que comparte. Satisfacer las necesidades de superación de los CUM (Centros Universitarios Municipales) logrando el perfeccionamiento consolidación de sus claustros. Contribuir con calidad a la formación integral de estudiantes y profesores con énfasis en la preparación político ideológica. La gestión de posgrado satisface las necesidades del territorio y se amplían las relaciones internacionales y la colaboración con universidades extranjeras. Se optimizan las estructuras y los métodos de dirección. Se incrementa el uso de la informatización y el empleo de las TIC, asumiendo la prevención y el control de ilegalidades dentro de la cultura organizacional.

En la actualidad el departamento de marxismo no cuenta con un sistema informático para la difusión de la información sobre los actividades que se realizan, lo cual constituye una limitante para un mayor impacto o visibilidad de la labor realizada por el departamento, por lo tanto es difícil una plena efectividad de la labor político ideológica sin la presencia en el espacio virtual de la intranet.

Por lo tanto se plantean las siguientes Manifestaciones del problema:

- Errores en la planificación de actividades y eventos científico-estudiantiles.
- Imprecisiones en las informaciones ofrecidas a los diferentes niveles.
- La información no siempre llega a todas las personas implicadas.

Luego de analizar la información referente al Departamento de Marxismo e Historia de la Uniss surge el siguiente **problema**: Limitaciones en la gestión de la información del Departamento de Marxismo e Historia de la Uniss.

Se determinaron como posibles **causas de los problemas**:

- La no uniformidad en la generación de informaciones operativas sobre la prestación de estos servicios, provoca la inconsistencia
- No disponen en el departamento de las herramientas necesarias que cumplan con las mejores prácticas para lograr una correcta Gestión de la información.
- Las malas condiciones con que llega la información, proporciona una baja calidad de la formación integral de estudiantes con énfasis en la preparación político ideológica.

-No existe la atención personalizada que necesitan los estudiantes, lo que implica la distorsión de la gestión de servicios.

Se propone como **objetivo general**: Desarrollar un Sitio Web para la gestión de la información sobre el departamento de Marxismo con la aplicación de tecnologías informáticas en este proceso.

Para guiar el curso de la investigación, se plantean las siguientes **preguntas de investigación**:

- ¿Cuáles son los fundamentos teóricos y metodológicos que sustentan el uso de un sitio web para contribuir a la gestión de la información?
- ¿Cómo diseñar un sitio web para contribuir a la gestión de la información del Departamento de Marxismo e Historia de la Uniss?
- ¿Cómo implementar un sitio web para contribuir a la gestión de la información del Departamento de Marxismo e Historia de la Uniss?

Para darle cumplimiento a las preguntas de investigación se realizaron las siguientes **tareas de investigación**:

- Determinación de los fundamentos teóricos y metodológicos que sustentan el uso de un sitio web para contribuir a la gestión de la información.
- Diseño de un sitio web para contribuir a la gestión de la información del Departamento de Marxismo e Historia de la Uniss.
- Implementación de un sitio web para contribuir a la gestión de la información del Departamento de Marxismo e Historia de la Uniss.

El documento está compuesto por tres capítulos, que abarcan todo lo relacionado con el trabajo investigativo realizado, además del análisis y diseño del sistema que se propone. Contando cada capítulo con introducción y conclusiones parciales, además de introducción y conclusiones generales. También cuenta con un apéndice para anexos en el cual podrán encontrar imágenes, gráficas y tablas que le ayudarán a un mejor entendimiento del problema.

Capítulo 1: Fundamentación teórica: En este capítulo se exponen los fundamentos teóricos del presente trabajo. Además, se revisan las tecnologías, lenguajes y herramientas existentes; se describen sus características y el por qué se seleccionaron.

Capítulo 2: Descripción de la solución propuesta: Describe el modelo del negocio identificando los procesos involucrados en él, las reglas que lo caracterizan, se identifican actores y trabajadores que en él intervienen, el diagrama de casos de uso y los diagramas de actividades correspondientes; también se muestra el diagrama de clases del modelo de objetos. Se definen los requerimientos funcionales, los no funcionales y los casos de uso del sistema así como los diagramas asociados a cada uno de ellos.

Capítulo 3: Construcción de la solución propuesta: Se describen los casos de uso del sistema; se define el diseño mediante los diagramas de clases de diseño; se representa el diseño de la base de datos a través del diagrama de clases persistentes y el modelo de datos, que incluye la descripción de cada una de las tablas; además se describe la implementación a través del diagrama de despliegue y del diagrama de componentes.

CAPÍTULO I: FUNDAMENTACIÓN TEÓRICA.

Introducción

Este capítulo contiene los fundamentos teóricos y los principales problemas que fundamentan la propuesta de solución. Además de brindar un enfoque general de aplicaciones web existentes vinculadas al campo de acción y el análisis comparativo de las soluciones existentes con la propuesta dada en este trabajo. Se describen además las tecnologías actuales de desarrollo utilizadas para el análisis, diseño e implementación del sistema sobre las cuales se apoya la propuesta.

1.1 Tecnología de la Información.

Las tecnologías de la información y la comunicación (TIC) agrupan los elementos y las técnicas usadas en el tratamiento y la transmisión de las informaciones, principalmente vinculadas a la informática, internet y telecomunicaciones.

Se denominan tecnologías de la información y la comunicación al conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz, imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética [1]. Las TIC incluyen la electrónica como tecnología base que soporta el desarrollo de las telecomunicaciones, la informática y el audiovisual.

La tecnología de información (TI), según lo definido por la asociación de la tecnología de información de América (ITAA) es “el estudio, diseño, desarrollo, puesta en práctica, ayuda o gerencia de los sistemas de información computarizados, particularmente usos del software y hardware.” La TI ha llegado a ser recientemente popular para ensanchar el término explícitamente para incluir el campo de la comunicación electrónica de modo que la gente entienda para utilizar la abreviatura TIC (tecnología de información y de comunicación). (Bologna&Walsh, 1997)

La información es probablemente la fuente principal de negocio en el primer mundo y ese negocio a su vez genera ingentes cantidades de información. Su correcta gestión es de importancia estratégica y no debe considerarse como una herramienta más entre muchas otras.

1.2 Breve reseña sobre los orígenes de la Web.

La Web es una idea que se construyó sobre la Internet. Las conexiones físicas son sobre la Internet, pero introduce una serie de ideas nuevas, heredando las ya existentes.

Empezó a principios de 1990, en Suiza en el centro de investigación CERN (centro de Estudios para la Investigación Nuclear) y la idea fue de Tim Berners-Lee, que se gestó observando una libreta que él usaba para añadir y mantener referencias de cómo funcionaban los ordenadores en el CERN.

Antes de la Web, la manera de obtener los datos por la Internet era caótica: había un sinnúmero de maneras posibles y con ello había que conocer múltiples programas y sistemas operativos. La Web introduce un concepto fundamental: la posibilidad de lectura universal, que consiste en que una vez que la información esté disponible, se pueda acceder a ella desde cualquier ordenador, desde cualquier país, por cualquier persona autorizada, usando un único y simple programa. Para que esto fuese posible, se utilizan una serie de conceptos, el más conocido es el hipertexto.

Con Web los usuarios novatos podrían tener un tremendo poder para hallar y tener acceso a la riqueza de información localizada en sistemas de cómputos en todo el mundo.

Este solo hecho llevó un avance tremendo de Internet, un ímpetu tan grande que en 1993 World Wide Web (WWW) creció un sorprendente 341000%, tres años después, en 1996, todavía se está duplicando cada 50 días.

1.3 Departamento de Marxismo e Historia de la UNISS.

El Departamento de Marxismo de la UNISS, fue fundado en 1997 a partir del núcleo inicial de profesores del ciclo existente en la filial de la UCLV. Este grupo compartía como departamento la atención a profesores de inglés y otros de las áreas humanísticas y sociales.

Con la aparición de la Sede Universitaria el colectivo, integrado por instructores y asistentes, se independiza y comienza un proceso de despegue en su actividad que permitió su crecimiento desde el punto de vista académica científico e investigativo a través del impulso a la formación doctoral y el perfeccionamiento de la pirámides categorial, lo que permitió inscribir 3 de sus profesores en doctorados curriculares y su conclusión en 2004, la ascensión de categorías docentes principales hasta alcanzar la satisfacción de las necesidades del claustro para cumplir el encargo social y la misión del Departamento, la búsqueda de relaciones internacionales y el trabajo sostenido con la cátedra martiana.

De igual forma el Departamento buscó alianzas con otras universidades del país como la Universidad Central Marta Abreu de Las Villas, La Universidad de Matanzas Camilo Cienfuegos y la Universidad de La Habana, centros que han contribuido a potenciar la calificación científica del Departamento.

1.3.1 Descripciones Generales.

El Departamento abre sus puertas con el fin de formar un profesional que se caracterice por una sólida preparación política e ideológica basado en los principios de la ideología de la Revolución Cubana: martiana, marxista leninista para dar respuesta a las necesidades de preparación de la sociedad cubana y en especial de los adolescentes y jóvenes como continuadores del proceso de construcción del socialismo.

Existen actualmente en la Sede Central un total de 20 profesores, donde solo 1 es profesor titular, 7 son asistentes, 11 son auxiliares y 1 es instructor, de ellos 3 obtuvieron el título de Doctor y 9 de Máster. Donde 2 profesores auxiliares se encuentran en estos momentos en proceso para la obtención de la Categoría docente principal de Profesor Titular, 2 profesores deben obtener el grado científico de DrC (Doctor en Ciencias) este año, y 1 profesor debe obtener este año el grado académico de Máster en Ciencias Históricas. En la FUM de Cabaiguán, al igual que Yaguajay hay presente 1 profesor asistente que presenta los grados de Máster. A diferencia de Fomento que presenta a 1 profesor auxiliar con un título de Máster.

La Misión del Departamento consiste en contribuir a la consolidación de la comunidad universitaria con un claustro altamente categorizado, comprometido con la Revolución y con los valores que comparte. Satisfacer las necesidades de superación de los CUM logrando el perfeccionamiento y consolidación de sus claustros. Contribuir con calidad a la formación integral de estudiantes y profesores con énfasis en la preparación político ideológica. La gestión de posgrado satisface las necesidades del territorio y se amplían las relaciones internacionales y la colaboración con universidades extranjeras. Se optimizan las estructuras y los métodos de dirección. Se incrementa el uso de la informatización y el empleo de las TIC, asumiendo la prevención y el control de ilegalidades dentro de la cultura organizacional.

La Visión es que el Departamento de Marxismo Historia de la UNISS forma parte de la universidad territorializada, sustentable y forma continuamente profesionales revolucionarios, comprometidos con la Revolución y la solución de necesidades del territorio y la nación.

1.3.2 Situación problemática.

Los Departamentos de Marxismo o equivalentes que funcionan o que existen en las universidades del país cumplen una importante misión al sectorizar y asesorar la materialización del trabajo político ideológico y la docencia vinculada o asociada a la enseñanza del marxismo y la historia de Cuba, no obstante, tampoco han estado favorecidos de contar con un sistema informático basado en las mejores prácticas para

la gestión de los servicios de las tecnologías de la información, dificultando así la normalización de los procesos de soporte y provisión de los servicios informáticos en dichos Departamentos, tampoco ha sido posible la uniformidad en la generación de informaciones operativas y estadísticas sobre la prestación de estos servicios, provocando inconsistencia y falta de veracidad en las informaciones ofrecida a los diferentes niveles. Por otra parte no es posible realizar una atención personalizada que necesitan los usuarios de las tecnologías de la información, lo que implica que se realicen muchos reportes que pudieran ser solucionados inmediatamente con dicha atención y que actualmente distorsionan las estadísticas de la gestión de servicios. En la actualidad la no inserción en la red del departamento de marxismo de la UNISS constituye una limitante para un mayor impacto o visibilidad de la labor desplegada por la institución.

1.4 Fundamentación de la metodología utilizada

1.4.1 Lenguaje Unificado de Modelado (UML)

Es una herramienta de lenguaje visual para modelación de sistemas (comprendiendo hardware y software) que permite generar diagramas organizados que capten todas las ideas de un proceso, siendo su principal finalidad que sea fácil de comprender para todas las personas involucradas en el desarrollo del sistema. Esto tiene la ventaja que al realizar un plan de diseño organizado a través de un cuidadoso análisis de las necesidades del cliente, se puede contar con un diseño sólido, el cual, puede ser modificado sin mayores problemas. Por consiguiente el objetivo del Proceso Unificado es guiar a los desarrolladores en la implementación y distribución eficiente de sistemas que se ajusten a las necesidades de los clientes. (R., 2008) UML es ante todo un lenguaje. Y como todo lenguaje proporciona vocabulario y reglas para permitir la comunicación. En este caso, el lenguaje se centra en la representación gráfica de un sistema. (R., 2008)

1.4.2 Proceso Unificado Racional (RUP)

RUP (Rational UnifiedProcess). Proceso Unificado Racional. Es una metodología cuyo fin es entregar un producto de software. Se estructura todos los procesos y se mide la eficiencia de la organización. Es un proceso de desarrollo de software el cual utiliza el lenguaje unificado de modelado UML, puede ser utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. El RUP es un conjunto de metodologías adaptables al contexto y necesidades de cada organización. Describe como aplicar enfoques para el desarrollo del software, llevando a cabo unos pasos para su realización. Se centra en la producción y mantenimiento de modelos del sistema. (Garzón, 2008) El modelo RUP le ofrece la mitigación temprana de posibles riesgos, el progreso visible en las etapas tempranas, así como la aplicación de iteración a iteración del conocimiento adquirido en una iteración. (Jacobson, Booch, &Rumbaugh, 2006)

1.5 Tendencias y tecnologías actuales

El estudio de las tecnologías actuales se ha convertido en uno de los factores claves en el desarrollo de cualquier sistema informático por muy sencillo que este resulte. Representa uno de los aspectos determinantes en el funcionamiento de las empresas e instituciones. En la medida en que el concepto calidad tecnológica sea mayor, se logrará alcanzar los objetivos trazados de una forma eficiente y efectiva.

Se necesita realizar una correcta selección de las tecnologías a emplear, en dependencia de las necesidades y recursos propios con los que se cuenta. Una correcta selección permitirá lograr un producto final con calidad y eficiencia. (Febles, 2005) En los epígrafes que vienen a continuación se pretende ofrecer un camino informativo a través de las tecnologías que se encuentran entorno de las requeridas para el desarrollo del sistema propuesto, analizando las ventajas y desventajas que las caracterizan, con el fin de poder justificar el por qué del lenguaje, y otros software utilizados.

1.1.1 Software Libre.

Entre los años 60 y 70 del Siglo XX, era común que los programadores y desarrolladores de software compartieran libremente sus programas unos con otros. Esto se debía en gran medida a que, por ese entonces, el software no era visto como un producto en sí mismo sino como un adjunto que los vendedores de las grandes computadoras de la época (las mainframes) aportaban a sus clientes para que éstos pudieran usarlas. A finales de los 70, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia. Todo esto fue empeorando con la llegada de las computadoras modernas que necesitaban de un sistema operativo (Free Software Foundation, 2008). 16

Dada esta situación, en 1985, Richard Stallman fundó la Free Software Foundation (FSF). Con el fin de otorgar libertad a los usuarios y restringir las posibilidades de apropiación del software, Stallman introdujo la definición de free software (software libre) que según él se refería a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software; de modo más preciso, se refiere a cuatro libertades de los usuarios del software: la libertad de usar el programa, con cualquier propósito; de estudiar el funcionamiento del programa, y adaptarlo a las necesidades; de distribuir copias, con lo cual se puede ayudar a otros y de mejorar el programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie (para la segunda y última libertad mencionadas, el acceso al código fuente es un requisito previo). (Free Software Foundation, 2008)

En la actualidad existen varios países en los cuales sus administraciones públicas muestran apoyo al software libre, tal es el caso de Alemania, Argentina, Brasil, Cuba, Chile, China, Ecuador, España, Francia, México, República Dominicana y Venezuela.

Principalmente para los países subdesarrollados, la aparición del software libre ha tenido un gran significado, pues en ocasiones el pago constante de software no libre es casi intolerable. Para Cuba, por sus características especiales de ser un país bloqueado, en ocasiones el pago de estas licencias no es tomado en cuenta y, por lo general, existe una tendencia a la piratería informática. Con el fin de ir extinguiendo esta tendencia y de apoyarse a los beneficios que trae consigo el software libre, nuestro país lleva a cabo un amplio programa de apoyo al software de fuentes abiertas y ha ido paulatinamente migrando total o parcialmente sus servidores y sistemas de escritorio.

1.5.1 Arquitectura Cliente/Servidor. Arquitectura en N capas

La arquitectura Cliente/Servidor es un modelo para desarrollar sistemas de información en el que las transacciones se dividen en elementos independientes que tienen la facilidad de cooperar entre ellos para intercambiar información, recursos o servicios. Esta arquitectura consiste básicamente en un cliente que realiza peticiones a otro programa (el servidor) que le da respuesta (Molina & Castro, 2009).

La arquitectura cliente/servidor genérica tiene dos tipos de nodos en la red: clientes y servidores. Consecuentemente, se refieren a veces como arquitecturas de dos niveles o dos capas. Algunas redes disponen de tres tipos de nodos (Hurtado Jara, 2008).

La arquitectura de n capas se enfoca en la distribución de roles y responsabilidades de forma jerárquica proveyendo una forma muy efectiva de separación de responsabilidades. El rol indica el modo y tipo de interacción con otras capas, y la responsabilidad indica la funcionalidad que está siendo desarrollada (Pérez Sandoval, 2009).

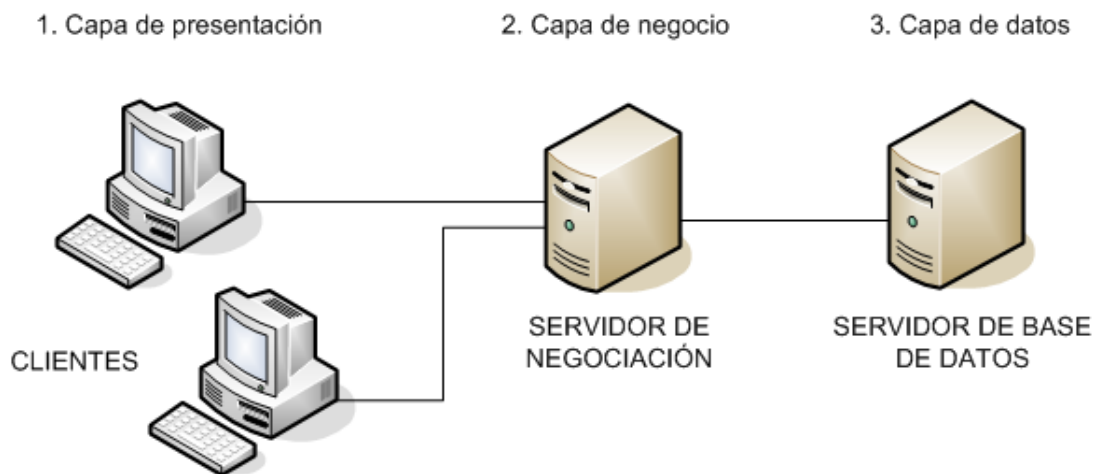


Figura1: Arquitectura en tres capas.

Arquitectura de 3 Capas

Una buena arquitectura de software debe facilitar los requerimientos de mantenimiento, reusabilidad, escalabilidad, y robustez del mismo. Al concertar la solución de un

problema como una serie de capas, cada capa debe ocuparse de un subconjunto de responsabilidades fuertemente acopladas y tener poca cohesión con las demás (Peláez, 2009).

La primera capa se denomina capa de presentación y normalmente consiste en una interfaz gráfica de usuario de algún tipo. La capa intermedia, consiste en la aplicación o lógica del negocio, y la tercera capa, la capa de datos, contiene los datos necesarios para la aplicación (bases de datos y otros). Esta separación entre la lógica de aplicación de la interfaz de usuario añade una enorme flexibilidad al diseño de la aplicación (Chappell& Kirk, 2006).

Dentro de la arquitectura de n Capas se encuentra el patrón Modelo-Vista-Controlador (MVC), el cual es usado en este trabajo.

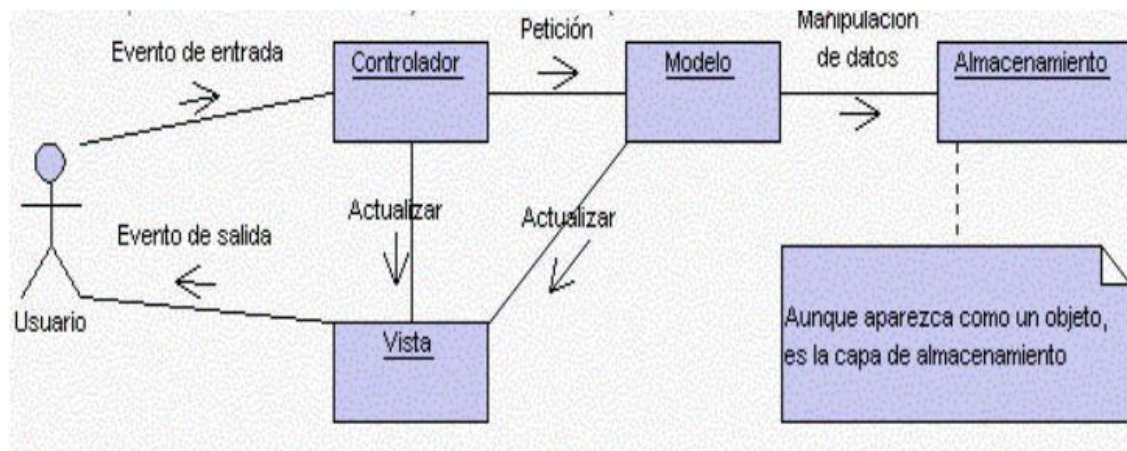


Figura 2: Esquema Modelo-Vista-Controlador

Fue diseñada para reducir el esfuerzo de programación necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos. Sus características principales son que el Modelo, las Vistas y los Controladores se tratan como entidades separadas; esto hace que cualquier cambio producido en el Modelo se refleje automáticamente en cada una de las Vistas. (Viktor, 2006).

Definición de las partes

El Modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos (Catalani, 2007).

La Vista es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario (Catalani, 2007).

El Controlador es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo. Sirve como un *intermediario* entre el Modelo y la Vista (Catalani, 2007).

Se trata de realizar un diseño que desacople la vista del modelo, con la finalidad de mejorar la reusabilidad. De esta forma las modificaciones en las vistas impactan en menor medida en la lógica de negocio o de datos.

Ventajas de este patrón según (Martínez, 2007):

- ✓ Hay una clara separación entre los componentes de un programa; lo cual nos permite implementarlos por separado.
- ✓ Debido a que cada parte es independiente de la otra los cambios o actualizaciones pueden ser realizados sin afectar la aplicación como un todo.
- ✓ La conexión entre el Modelo y sus Vistas es dinámica; se produce en tiempo de ejecución, no en tiempo de compilación.

1.6 Lenguajes de programación

1.6.1 Lenguajes de programación web

Actualmente existen diferentes lenguajes de programación para desarrollar en la web, estos han ido surgiendo debido a las tendencias y necesidades de las plataformas. Algunos de estos son JSP (Java Server Page), Python, Perl, Ruby, HTML (HyperTextMarkupLanguage), JavaScript, PHP (Personal Home Page), ASP (Active Server Page), ASP.Net (Active Server Page.Net). A continuación se dará una introducción a algunos de los mencionados anteriormente.

1.6.2 PHP

Es un lenguaje de programación utilizado para la creación de sitio web. PHP es un acrónimo que significa “PHP Hypertext Pre-processor”, (inicialmente se llamó Personal

Home Page). PHP es un lenguaje de script interpretado en el lado del servidor utilizado para la generación de páginas web dinámicas, embebidas en páginas HTML y ejecutadas en el servidor. Surgió en 1995, desarrollado por PHP Group. PHP no necesita ser compilado para ejecutarse. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas. (Pérez Valdés, 2007)

1.6.3 HTML

Desde el surgimiento de internet se han publicado sitios web gracias al lenguaje HTML. Es un lenguaje estático para el desarrollo de sitios web (acrónimo en inglés de HyperTextMarkupLanguage, en español Lenguaje de Marcas de Hipertexto). Desarrollado por el World Wide Web Consortium(W3C). Este lenguaje se caracteriza por ser un lenguaje de fácil aprendizaje, estático, lo admiten todos los exploradores, pero la interpretación de cada navegador puede ser diferente.

1.6.4 JavaScript

Este es un lenguaje interpretado, no requiere compilación. Fue creado por BrendanEich en la empresa Netscape Communications. Utilizado principalmente en páginas web. Es similar a Java, aunque no es un lenguaje orientado a objetos, el mismo no dispone de herencias. La mayoría de los navegadores en sus últimas versiones interpretan código Java Script. El código JavaScript puede ser integrado dentro de las páginas web. Los script tienen capacidades limitadas, por razones de seguridad, el código visible por cualquier usuario y el código deben descargarse completamente (Álvarez, 2001).

1.6.5 CSS

CSS (Cascade Style Sheets) es una tecnología que permite controlar la presentación de los documentos en la Web. Las hojas de estilo en cascada o CSS constituyen un lenguaje sencillo que complementa el de HTML, suponiendo un apoyo fundamental a la hora de diseñar páginas Web, porque permiten una mayor precisión en el ajuste de los elementos de diseño. (Vedora, 2006)

Esta técnica consiste en separar el diseño del contenido, de manera que las indicaciones para conformar el diseño se agrupan en una hoja de estilo o archivo fuera del contenido del documento de la página HTML. Lo que hace fundamentalmente el código de las hojas de estilos es transformar las etiquetas del lenguaje HTML y

conformarlas a las características que se quiera darle; pero también, con este código se pueden crear etiquetas nuevas, que se introducen dentro del documento. Una de las ventajas de las hojas de estilos es que se puede modificar algunas características de todos los documentos de un sitio Web desde un archivo, sin tener que modificarlas en cada uno de los documentos. (Vedora, 2006)

Ventajas: (Cadenas, 2010)

- ✓ Es la mejor forma de separar los contenidos y su presentación, es imprescindible para crear páginas Web complejas.
- ✓ Reduce la complejidad de su mantenimiento.
- ✓ Permite visualizar el mismo documento en infinidad de dispositivos diferentes.
- ✓ Permite especificar la apariencia de todo el sitio Web en un único lugar, los estilos para fuentes, tamaños de la fuente, espaciado de letras, colores, y márgenes. Un cambio en un fichero genera cambios en todos los ficheros.

Para la realización del Portal Web se decidió utilizar como lenguaje de programación para la lógica del negocio PHP, y para la interfaz de usuario JavaScript debido a sus ventajas. Pérez Valdés (2007) enumera las siguientes ventajas que tiene el uso de PHP:

- ✓ Se caracteriza por ser un lenguaje muy rápido.
- ✓ Soporta en cierta medida la orientación a objeto. Clases y herencia.
- ✓ Es un lenguaje multiplataforma: Linux, Windows, entre otros.
- ✓ Tiene capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- ✓ Capacidad de expandir su potencial utilizando módulos.
- ✓ Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- ✓ Incluye gran cantidad de funciones.
- ✓ No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

Pérez Valdés (2007) enuncia las siguientes ventajas que tiene el uso de JavaScript:

- ✓ Tiene muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos, con funciones, estructuras de datos complejas.
- ✓ Es un lenguaje interpretado (que no es compilado) por el cliente.
- ✓ Las referencias a objetos se comprueban en tiempo de ejecución.
- ✓ Permite crear páginas más dinámicas, haciéndolas más atractivas para el usuario.

1.7 Servidores Web

Un servidor Web es un programa que implementa el protocolo HTTP (HyperText Transfer Protocol). Este protocolo está diseñado para transferir lo que se llama hipertextos, páginas Web o páginas HTML, textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos. Entre los servidores más utilizados en Internet se encuentran Apache e Internet Information Services(IIS) (Cuenca, 2006).

Entre los servidores Web más utilizados se encuentran Microsoft Internet Information Server, Microsoft Personal Web Server, Apache HTTP Server, Netscape Enterprise Server y Sun ONE Web Server.

1.7.1 Apache

El servidor Apache es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras. Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración (Una introducción a Apache, 2006).

Ventajas de su uso son (Una introducción a Apache,2006):

- ✓ Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- ✓ Apache es una tecnología gratuita de código fuente libre.
- ✓ Apache trabaja con gran cantidad de Perl, PHP y otros lenguajes de script.

- ✓ Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- ✓ Es una tecnología gratuita de código fuente abierta.
- ✓ Es un servidor altamente configurable de diseño modular.
- ✓ Trabaja con gran cantidad de lenguajes como Perl, PHP, Java y páginas jsp.
- ✓ Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.

1.8 Herramientas de desarrollo

1.8.1 Rational Rose

Para apoyar el trabajo de la metodología RUP, ha sido desarrollada por la Compañía norteamericana Rational Corporation la herramienta CASE (*Computer Assisted Software Engineering*) Rational Rose desde el año 2000. Esta herramienta integra todos los elementos que propone la metodología para cubrir el ciclo de vida de un proyecto. Esta herramienta CASE propone la utilización de cuatro tipos de modelos para realizar un diseño del sistema, utilizando una vista estática y otra dinámica de los modelos del sistema, uno lógico y otro físico. Permite crear y refinar estas vistas creando de esta forma un modelo completo que representa el dominio del problema y el sistema de software. (Rodríguez Febles, 2005)

Rational Rose proporciona también mecanismos para realizar la denominada Ingeniería Inversa, es decir, a partir del código de un programa, se puede obtener información sobre su diseño. (Rodríguez Febles, 2005)

1.9 Framework

El concepto framework se emplea en muchos ámbitos del desarrollo de sistemas software, no solo en el ámbito de aplicaciones Web. En general, con el término framework, se está refiriendo a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y

configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta. (Gutiérrez, 2006)

Algunas de las características presentes en casi todos son: abstracción de URL y sesiones (no es necesario manipular directamente las URLs ni las sesiones, el framework se encarga de esto); acceso datos (incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en bases de datos, etc.); controladores (la mayoría de los frameworks que existen implementa una serie de controladores para gestionar eventos). (Gutiérrez, 2006)

Son diseñados con la intención de facilitar 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 de proveer un sistema funcional.

En el mundo del software es común observar como la mayoría de los productos utilizan de alguna u otra forma una base de datos que permita el almacenamiento de la información. En este proceso de creación de software, el empleo de frameworks se ha convertido para muchos programadores en una potente herramienta.

Actualmente existen en el mundo una gran variedad de frameworks para PHP y JavaScript. Algunos de estos son Symfony, Akelos Framework, PHP Framework, Biscuit, Cake, Codelgniter, PhpMVC, Prado, Seagull, Zend Framework, etc.

1.9.1 Symfony

Symfony es un framework diseñado para optimizar el desarrollo de las aplicaciones web mediante algunas de sus principales características. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web.

Symfony ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel. Es compatible con la mayoría de gestores de

bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server. Se puede ejecutar tanto en plataformas *nix (Unix, Linux, etc.) como en plataformas Windows.

Las características más comunes para el desarrollo de proyectos web están automatizadas en Symfony, tales como:

- ✓ Permite la internacionalización para la traducción del texto de la interfaz, los datos y el contenido de localización.
- ✓ La presentación usa templates y layouts que pueden ser construidos por diseñadores de HTML que no posean conocimientos del framework.
- ✓ Los formularios soportan la validación automática, lo cual asegura mejor calidad de los datos en las base de datos y una mejor experiencia para el usuario.
- ✓ El manejo de caché reduce el uso de banda ancha y la carga del servidor.
- ✓ La facilidad de soportar autenticación y credenciales facilita la creación de áreas restringidas y manejo de seguridad de los usuarios.
- ✓ El enrutamiento y las URLs inteligentes hacen amigable las direcciones de las páginas de la aplicación.
- ✓ Las listas son más amigables, ya que permite la paginación, clasificación y filtraje automáticos.
- ✓ Los plugins proveen un alto nivel de extensibilidad.

1.9.2 Doctrine

Doctrine es un mapeador de objetos-relacional (ORM) escrito en PHP que proporciona una capa de persistencia para objetos PHP. Es una capa de abstracción que se sitúa justo encima de un Sistema Gestor de Bases de Datos.

Doctrine tiene influencias de docenas de proyectos de personas muy diferentes. Las mayores influencias son de Hibernate (el ORM de Java) y de ActiveRecord (de Ruby

onRails). Ambos tienen una implementación completa tanto en Java como en Ruby. El propósito de Doctrine es construir una solución igual de potente para PHP.

Una característica de Doctrine es el bajo nivel de configuración que necesita para empezar un proyecto. Doctrine puede generar clases a partir de una base de datos existente y después el programador puede especificar relaciones y añadir funcionalidad extra a las clases autogeneradas. No es necesario generar o mantener complejos esquemas XML de base de datos como en otros frameworks.

Otra característica importante de Doctrine es la posibilidad de escribir consultas de base de datos utilizando un dialecto de SQL denominado **DQL** (Doctrine QueryLanguage) que está inspirado en Hibernate (Java).

Otras características notables de Doctrine son:

- ✓ Soporte para datos jerárquicos;
- ✓ Soporte para hooks (métodos que pueden validar o modificar las escrituras y lecturas de la base de datos) y eventos para manejar la lógica de negocio relacionada;
- ✓ Herencia;
- ✓ Un framework de caché que utiliza diversos motores como memcached, SQLite o APC;
- ✓ Transacciones ACID;
- ✓ Diversos comportamientos del modelo (conjuntos anidados, internacionalización, log, índice de búsqueda);
- ✓ Una función "compilar" que combina varios archivos PHP del framework en uno solo para evitar el descenso de rendimiento que provoca incluir varios archivos PHP.

1.10 Sistemas Gestores de Base de Datos (SGBD)

Los sistemas de gestión de base de datos (SGBD); (en inglés: Databasemanagementsystem, abreviado DBMS) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las

aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. Su propósito general es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante, para un buen manejo de datos (Sandoval Méndez, 2009).

El objetivo fundamental de un SGBD consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, o sea, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado (Mato García, 2006)

En el mundo existen varios tipos de SGBD clasificados en licenciamiento comercial como Oracle, SQL SERVER, etc. y los de licenciamiento libre entre los que se encuentra MySQL, PostgreSQL, FireBird, SQLite, etc.

1.10.1 SQL (StructureQueryLanguage)

SQL (StructureQueryLanguage) ó Lenguaje de Consulta Estructurado es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas. Posibilita lanzar consultas con el fin de recuperar información de interés de una base de datos de una forma sencilla.

SQL permite la concesión y denegación de permisos, la implementación de restricciones de integridad y controles de transacción, y la alteración de esquemas. Debido a que es un lenguaje declarativo, especifica qué es lo que se quiere y no como conseguirlo, por lo que una sentencia no establece explícitamente un orden de ejecución.

1.10.2 SQL Server

Diseñada desde su inicio para trabajar en entornos Internet e Intranet, Microsoft SQL Server es capaz de integrar los nuevos desarrollos para estos entornos específicos con los desarrollos heredados de aplicaciones tradicionales. (Sandoval, 2009)

SQL Server soporta la configuración automática y la auto-optimización, además tiene una administración multiservidor para un gran número de servidores, pero es mayormente utilizada para trabajar en Plataformas de Windows. (Sandoval, 2009)

1.10.3 PostgreSQL

Es un sistema de gestión de base de datos relacional orientada a objetos y libre. PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma sin necesidad de bloqueos. Además de una amplia variedad de tipos nativos, provee soporte para, números de precisión arbitraria, texto de largo ilimitado, figuras geométricas, y arrays. Soporta transacciones, tiene mejor soporte para subselects, triggers, vistas y procedimientos almacenados en el servidor, además de ciertas características orientadas a objetos. (Johnson, 2007)

1.10.4 MySQL

MySQL es el sistema gestor de bases de datos Open Source más popular en la comunidad de programadores. Este puede ser descargado de Internet y usado de forma gratuita. Su código se puede estudiar y adecuar a cualquier requerimiento establecido. Es muy rápido, fiable, fácil de usar y surge para manipular bases de datos muy grandes con gran eficiencia.

Es un sistema multiplataforma de base de datos, aspecto que le aporta la característica de ser fiable y veloz. Cuenta con un sistema de privilegios y contraseñas muy seguro que permite la autenticación básica para el acceso al servidor; elemento muy importante que contribuye a garantizar en gran medida la seguridad de la aplicación. El lenguaje PHP posee un amplio conjunto de funciones definidas para el tratamiento de este gestor. Este SGBD es considerado (en su propia documentación así lo reseña) como la más rápida y robusta herramienta para la administración y gestión de bases de datos, tanto para volúmenes de datos grandes como pequeños.

Sus principales características son: (Vedora, 2006)

- ✓ Cuenta con tecnología InnoDBEngine, que proporciona transacciones, claves externas, actualización y borrado en cascada y bloqueo a nivel de fila, más rápido, con caché de consultas, mejoras en inserciones, búsqueda en índices compuestos y creación de índices sobre texto completo, un servidor embebido y compatibilidad con otras bases de datos.
- ✓ Es posible definir diversos tipos de columnas como enteros de 1, 2, 3, 4, y 8 bytes, coma flotante, doble precisión, carácter, fechas, enumerados, etc.

- ✓ El servidor soporta mensajes de error en distintas lenguas.
- ✓ Todas las columnas pueden tener valores por defecto.
- ✓ Admite registros de longitud fija y variable.
- ✓ Todas las claves viajan cifradas en la red.
- ✓ Cuenta con un sistema de contraseñas y privilegios muy flexibles y seguros.
- ✓ Puede trabajar en distintas plataformas y sistemas operativos.
- ✓ Implementa el multiprocesamiento, es decir, puede usar varias CPU si estas están disponibles.

Ventajas: (Cadenas, 2010)

- ✓ Está disponible para la mayoría de las plataformas de sistemas operativos.
- ✓ Su bajo consumo lo hace apto para ser ejecutado en una máquina con escasos recursos sin ningún problema.
- ✓ Es software libre. Cualquier persona puede utilizar y modificar el código ajustándolo a sus necesidades sin pagar por hacerlo.
- ✓ Gran velocidad y flexibilidad. Es muy rápido.
- ✓ Alto rendimiento. (Servidores de Base de Datos citado en Raphie Morgan, 2006)
- ✓ Elevada velocidad tanto al conectar con el servidor como al servir consultas y demás funciones.
- ✓ Excelentes utilidades de administración (backup, recuperación de errores, etc).
- ✓ Aunque se bloquee, no suele perder información ni corromper los datos.
- ✓ Excelente integración con PHP.
- ✓ No tiene límites en el tamaño de los registros.
- ✓ Significativo control de acceso, es decir, qué usuarios tienen acceso a qué tablas y con qué permisos.
- ✓ MySQL se comporta mejor que otros motores de Base de Datos como SQL Server y Oracle a la hora de modificar ó añadir campos a una tabla.

- ✓ Es multi-hilo, o sea, maneja muchas conexiones al mismo tiempo, donde cada una tiene su propio hilo, de modo que ningún hilo tiene que esperar por otro, a menos que uno esté modificando una tabla que otro quiera acceder.

1.11 ¿Por qué utilizar Apache-PHP-MySQL?

La selección de estas tecnologías se basó principalmente en los criterios de rapidez, sencillez y rendimiento, unidos a otros dos mucho más importantes: la tecnología Open Source y la característica de integrarse para funcionar en cualquier plataforma. Estas características aportan un gran beneficio gracias al entorno de software libre de la actualidad y cumplen con una política trazada por el país de proyectarse con visión al futuro y comenzar a desarrollar este tipo de aplicaciones.

Como Sistema Gestor de Base de Datos se seleccionó MySQL, por su velocidad y sencillez. La selección estuvo motivada además por la utilización de PHP, ya que esta tecnología exhibe un magnífico soporte para el trabajo con MySQL. Finalmente como servidor Web se determinó que la mejor opción era el Apache, por mostrar un excelente comportamiento y estabilidad. Este servidor es el más utilizado en el mundo por su magnífico desempeño, seguridad y las posibilidades de gestionar de una forma eficiente muchos de los aspectos que definen en gran medida la calidad de los servidores Web.

La selección de Apache añade una integración perfecta con las tecnologías antes mencionadas. Apache-PHP-MySQL forman un en su conjunto una vía muy eficaz a la hora de tener en cuenta la implementación de sitios web de corto, mediano o gran alcance.

CONCLUSIONES

El vasto estudio teórico y metodológico efectuado, evidenció la necesidad de la creación de una aplicación web para contribuir a la inserción del Departamento de Marxismo e Historia en la red. La automatización del proceso de mostrar información es una importante contribución a los trabajadores de la entidad con el objetivo de ahorrar recursos materiales y humanos, pues permitirá mejorar el control de la información relacionada con los servicios, los estudiantes, los eventos y con los acuerdos, además de disminuir el tiempo de búsqueda de la información solicitada. Se consideró a RUP la metodología más apropiada para el desarrollo del proyecto y a UML como lenguaje de modelado. La aplicación estará basada en la arquitectura de programación de tres capas y dos niveles, lo que permitirá el reuso y mantenimiento del código. Además, se propone utilizar la programación orientada a objetos. En este capítulo se analizaron las características de las distintas metodologías y herramientas de desarrollo, así como las ventajas de las mismas, La selección de estas tecnologías se basó principalmente en la tecnología código libre y la característica de integrarse para funcionar en cualquier plataforma.

CAPITULO 2: DESCRIPCIÓN DE LA SOLUCIÓN

PROPUESTA.

Introducción

En este capítulo se describe el proceso de mostrar información en la en la red del Departamento de Marxismo e Historia de la UNISS, se exponen las reglas del negocio a tener en cuenta para el diseño del sistema, una caracterización de los actores y trabajadores del mismo, sus casos de uso, conformándose el diagrama de casos de uso, la descripción de estos en formato expandido, y se muestran los diagramas de actividades y del modelo objetos. Además se describe de modo general el funcionamiento del sistema diseñado. Se definen los requerimientos funcionales y no funcionales, se realiza la descripción del modelo de casos de uso del sistema, basándose en los actores y los casos de uso.

2.1 Reglas del Negocio a Considerar.

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio. (Jacobson, Booch, &Rumbaugh, El Proceso Unificado de Desarrollo de Software, 2006) Partiendo de lo planteado anteriormente fueron identificadas las siguientes reglas:

- Solo el administrador del sitio web deberá gestionar toda la información.
- El administrador del sistema deberá mantenerlo actualizado con la información disponible.
- El sistema deberá mostrar todos los reportes requeridos.

- El administrador y el usuario no pueden tener los mismos privilegios.

2.2 Descripción del modelo del negocio.

Los procesos de negocio son el conjunto total de actividades necesarias para producir un resultado de valor percibido y medible para un cliente individual de un negocio. (Jacobson, Booch, & Rumbaugh, El Proceso Unificado de Desarrollo de Software, 2006) Se puede concluir que los procesos de negocio son un grupo de tareas relacionadas lógicamente que se llevan a cabo en una determinada secuencia y forma, y que emplean los recursos de la organización para dar resultados que apoyen sus objetivos.

Los propósitos que se persiguen al realizarse el modelado del negocio, son:

- Entender la estructura y la dinámica de la organización.
- Entender los problemas actuales e identificar mejoras potenciales.
- Asegurarse de que los clientes, usuarios finales y desarrolladores tienen una idea común de la organización.
- Derivar los requerimientos del sistema a partir del modelo de negocio que se obtenga.

(Hernández A., 2005)

Los principales artefactos que se obtienen como resultado del modelamiento del negocio son: reglas del negocio, modelo de casos de uso del negocio y el modelo de objetos del negocio.

2.2.1 Descripción de los procesos del negocio

Un proceso de negocio es un conjunto de tareas relacionadas lógicamente, llevadas a cabo para lograr un resultado de negocio definido. Lowenthal (2004). A partir del planteamiento anterior se realizó un estudio de la organización en cuestión y el proceso identificado para ser modelado fue mostrar la información que es manipulada en el Departamento de Marxismo de la UNISS. La información que se manipula en la entidad está relacionada con los servicios que ofrece la misma a estudiantes y profesores.

Cuando un cliente de la entidad solicita alguna información, el especialista es el encargado de buscarla y mostrarla al solicitante. Para buscar la información el especialista debe dirigirse al lugar donde se encuentre la misma, es decir, si la información solicitada por el cliente está relacionada con los servicios, el jefe de Departamento se dirige a los distintos soportes que se tienen en el Departamento, pueden ser en una flash, en los propios archivos digitalizados, o en soportes duros. Es de interés para el Departamento que se ofrezca una buena cantidad de información científica que podrá ser consultada por los usuarios.

2.2.2 Modelo de casos de uso del negocio.

El modelo de Casos de Uso del Negocio (CUN) describe los procesos de una empresa en términos de casos de uso y actores del negocio en correspondencia con los procesos del negocio y los clientes, respectivamente. El modelo de casos de uso presenta un sistema desde la perspectiva de su uso y esquematiza cómo proporciona valor a sus usuarios. (Jacobson, Booch, & Rumbaugh, El Proceso Unificado de Desarrollo de Software, 2006)

2.2.3 Actores del negocio.

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados. (Álvarez, 2005) Basado en esta idea se definió el actor del negocio:

Actor	Descripción
Usuario	Es un docente o un profesor de la Uniss que se beneficia de la información que brinda el Jefe del Departamento.

Tabla: 1 Descripción del Actor del Negocio

2.2.4 Diagrama de casos de uso del negocio

Los diagramas de casos de uso sirven para especificar las funcionalidades y el comportamiento de un sistema mediante su interacción con los usuarios y/o otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema (Jacobson et al., 2006b). Un diagrama de casos de uso del negocio representa gráficamente la interacción de los procesos del negocio y los actores y/o trabajadores del negocio. Permitiendo mostrar los límites y el entorno de la organización bajo estudio (Jacobson et al., 2006b).

Para comprender los procesos de negocio se construye el diagrama de casos de uso del negocio en el que aparece cada proceso del negocio relacionado con su actor.

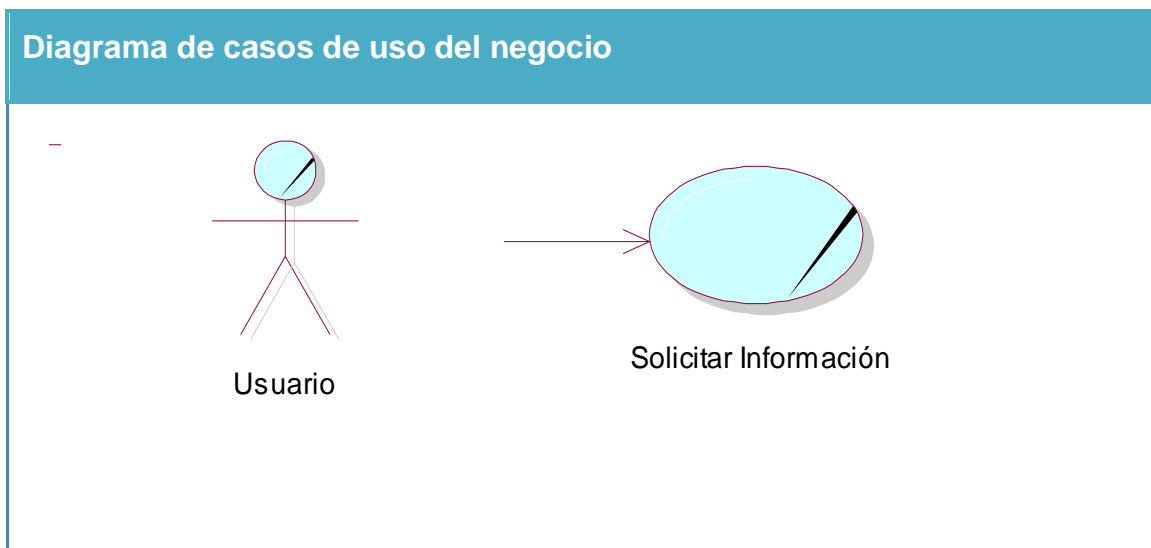


Figura 3: Diagrama de Caso de Uso del Negocio.

2.2.5 Trabajadores del negocio.

Es una abstracción de una persona o grupo de personas, una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio, manipulando entidades del mismo y representando un rol. (Álvarez, 2005) A continuación se muestra el trabajador del negocio.

Trabajador del negocio	Descripción
Jefe del Departamento	Es la persona que atiende a los docentes y profesores, y se encarga de brindarle la información que estos soliciten.

Tabla: 2 Trabajador del Negocio

2.2.6 Descripción de los casos de uso del negocio

Luego de identificar los procesos del negocio y realizar el diagrama de caso de usos del negocio, se hace necesario describirlos en detalle. A continuación tiene lugar dicha descripción representadas en forma de tabla:

Caso de Uso:	Solicitar Información.	
Actores:	Usuario.	
Propósito:	Obtener información sobre los servicios que brinda la entidad, sobre las actividades que atiende el Departamento.	
Resumen:	El caso de uso se inicia cuando el usuario le solicita al jefe del Departamento la información que necesite, la cual puede ser sobre los servicios que brinda la entidad, ó sobre los profesores que brindan servicios en estos Departamentos. El Jefe de Departamento busca la información solicitada en la Plantilla de trabajo en dependencia de la necesidad del usuario. El jefe del Departamento le muestra la información al usuario, este recibe la información y finaliza el caso de uso.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Negocio	
1. El estudiante o el profesor le solicita al Jefe del Departamento la información. 4. El estudiante o el profesor recibe la información y finaliza el caso de uso.	2. El Jefe Del Departamento busca la información que desea obtener el estudiante o el profesor. 3. El Jefe del Departamento le muestra la información al estudiante o al profesor.	

Prioridad	Media
-----------	-------

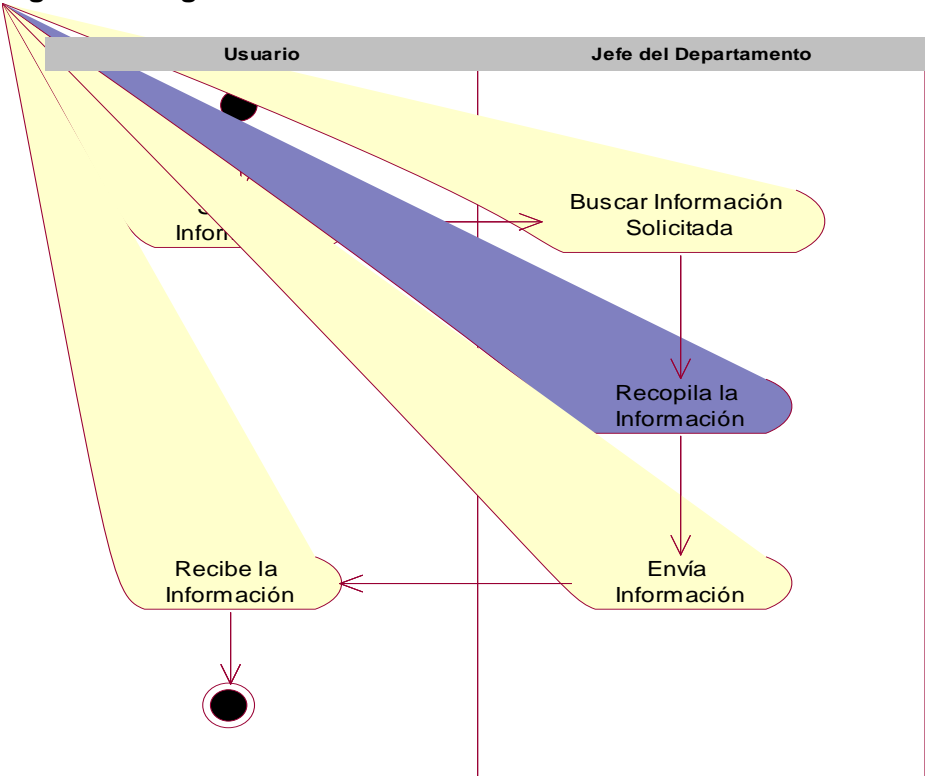
Tabla 3: Descripción del Caso de Uso del Negocio Solicitar Información.

2.2.7 Diagrama de actividades del negocio.

El diagrama de actividad es un grafo que contiene los estados en que puede hallarse la actividad a analizar. Cada estado de la actividad representa la ejecución de una sentencia de un procedimiento, o el funcionamiento de una actividad en un flujo de trabajo. En resumen describe un proceso que explora el orden de las actividades que logran los objetivos del negocio. (Jacobson, Booch, & Rumbaugh, 2006).

Para un mejor entendimiento del proceso en cuestión se realiza el Diagrama de Actividad correspondiente al Caso de Uso del Negocio Solicitar Información.

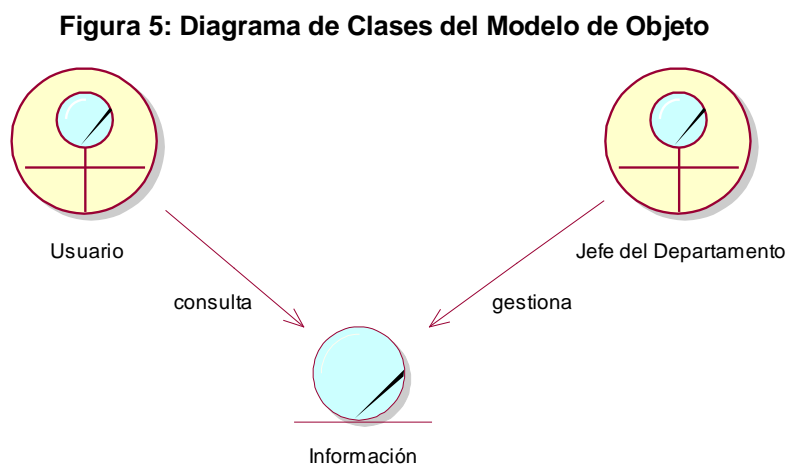
Figura 4: Diagrama de actividad del Caso de Uso Solicitar Información.



2.2.8 Diagrama de clases del modelo de objetos

Un modelo de objetos del negocio es un modelo interno a un negocio. Describe como cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo. (Jacobson, Booch, & Rumbaugh, 2006) Una entidad del negocio representa algo, que los trabajadores toman, inspeccionan, manipulan, producen o utilizan en un caso de uso del negocio. (Jacobson, Booch, & Rumbaugh, 2006)

Después de analizar los conceptos anteriores se realiza el Diagrama de Clases del Modelo de Objetos que se muestra:



2.3 Requerimientos funcionales.

Los requerimientos funcionales permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara, lo que debe hacer el mismo. Todo esto basándose en las necesidades de los usuarios y clientes. Los requerimientos funcionales del sistema propuesto son los siguientes:

1. Autenticarse.

2. Registrarse.

3. Cambiar contraseña.

4. Gestionar Taller Martiano

- 4.1 Insertar Taller Martiano
- 4.2 Modificar Taller Martiano
- 4.3 Eliminar Taller Martiano

5. Gestionar Fórum de Historia de la UNISS

- 5.1 Insertar Fórum de Historia de la UNISS
- 5.2 Modificar Fórum de Historia de la UNISS
- 5.3 Eliminar Fórum de Historia de la UNISS

6. Gestionar Otros Eventos

- 6.1 Insertar Otros Eventos
- 6.2 Modificar Otros Eventos
- 6.3 Eliminar Otros Eventos

7. Gestionar Cambio de Categorías

- 7.1 Insertar Cambio de Categorías
- 7.2 Modificar Cambio de Categorías
- 7.3 Eliminar Cambio de Categorías

8. Gestionar Cursos

- 8.1 Insertar Curso
- 8.2 Modificar Curso
- 8.3 Eliminar Curso

9 Gestionar Otras Convocatorias

- 9.1 Insertar Otras Convocatorias
- 9.2 Modificar Otras Convocatorias
- 9.3 Eliminar Otras Convocatorias

10 Gestionar Bibliografía

- 10.1 Insertar Bibliografía
- 10.2 Modificar Bibliografía
- 10.3 Eliminar Bibliografía

11 Gestionar Postgrados

- 11.1 Insertar Postgrados
- 11.2 Modificar Postgrados
- 11.3 Eliminar Postgrados

12 Gestionar Publicación

- 12.1 Insertar Publicación

- 12.2 Modificar Publicación
- 12.3 Eliminar Publicación

13 Gestionar Departamento

- 13.1 Insertar Departamento
- 13.2 Modificar Departamento
- 13.3 Eliminar Departamento

14 Gestionar Asignatura

- 14.1 Insertar Asignatura
- 14.2 Modificar Asignatura
- 14.3 Eliminar Asignatura

15 Gestionar Profesor

- 15.1 Insertar Profesor
- 15.2 Modificar Profesor
- 15.3 Eliminar Profesor

16 Reportes:

- Listar Departamento
- Listar Profesores
- Listar Publicaciones
- Listar Asignaturas
- Listar Fóruns
- Listar Talleres Martianos
- Listar Cursos
- Listar Cambio de Categorías
- Listar Otros Eventos
- Listar Otras Convocatorias

17 Cerrar Sesión.

2.4 Requerimientos no funcionales.

Los requisitos no funcionales especifican propiedades del sistema, como restricciones del entorno o de la implementación, rendimiento, dependencias de la plataforma, factibilidad de mantenimiento, extensibilidad y fiabilidad; con las cuales debe cumplir.

Requerimientos de apariencia o interfaz externa.

- ❖ La interfaz del sistema debe ser amigable, sencilla, agradable, legible, de fácil uso, aunque debe ser personalizada de acuerdo al tipo de usuario que acceda a ella.

- ❖ La interfaz estará diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad.
- ❖ La información debe tener claridad y una adecuada organización, de manera que permita una interpretación correcta de la misma.
- ❖ Se definen pautas, y aplicación de normas de diseño que garanticen armonía visual entre los elementos de la interfaz.

Requerimientos de Usabilidad.

- ❖ El sistema solo podrá ser accedido por los usuarios definidos por la Oficina de Seguridad para las Redes Informáticas.
- ❖ El sistema debe garantizar un acceso fácil y rápido, siendo los trabajadores de la Oficina de Seguridad para las Redes Informáticas de Sancti Spíritus quienes más lo utilicen.

Requisitos de Rendimientos.

- ❖ La eficiencia del sistema estará determinada por el aprovechamiento de los recursos que se disponen en la arquitectura 3 capas, la velocidad de respuesta de las consultas realizadas a la Base de Datos, además se seguirá una lógica de negocios en la comunicación entre el cliente y el servidor web de modo tal que cada cual procesará lo que le corresponde, puesto que es posible mencionar que existen funciones que son más óptimas ejecutarlas en el cliente, otras por cuestiones de seguridad, o de acceso a los datos, deben realizarse en el servidor.
- ❖ La herramienta propuesta debe ser rápida y el tiempo de respuesta debe ser el mínimo posible, adecuado a la rapidez con que el cliente requiere la respuesta a su acción..

Requisitos de Soporte

- ❖ Se requiere un servidor de bases de datos con soporte para grandes volúmenes de información, velocidad de procesamiento y tiempo de respuesta rápido en accesos concurrentes. Se documentará la aplicación para garantizar su soporte.

- ❖ Los servicios de instalación y mantenimiento del sistema serán responsabilidad del administrador.

- ❖ El sistema será realizado con un enfoque flexible, escalable y robusto para propiciar su futuro mejoramiento y la incorporación de otras funcionalidades, módulos u opciones.

Portabilidad.

- ❖ Para desarrollar la aplicación se seleccionó la plataforma Windows, pero podrá ejecutarse desde otras plataformas, siempre que soporten el lenguaje PHP y PostgreSQL, por ejemplo Linux.

Requisitos de seguridad.

- ❖ Debe garantizar la conectividad e integridad de los datos almacenados a través de la red. Esto está garantizado por el Sistema Operativo.

- ❖ Debe garantizar la confidencialidad para proteger la información de acceso no autorizado. Esto estará garantizado por el Sistema Gestor de Base de Datos.

- ❖ El sistema impondrá un estricto control de acceso que permitirá a cada usuario tener disponible solamente las opciones relacionadas con su actividad.

- ❖ La información deberá estar disponible a los usuarios en todo momento, limitada solamente por las restricciones que estos tengan de acuerdo con la política de seguridad del sistema.

Ayuda y documentación en línea.

- ❖ El sistema contará con una ayuda que le explicará de manera clara y sencilla al usuario todas las funcionalidades del sistema.

- ❖ La ayuda quedará conformada por un menú general que le ofrecerá al usuario una forma fácil para navegar por toda la ayuda, mediante el menú el usuario podrá ir de un lugar a otro sin perderse.

Software.

❖ La aplicación debe poderse ejecutar en entornos Windows o Linux (Multiplataforma).

❖ Del lado del servidor se utilizará Apache como servidor Web, PHP 5.2.6 o superior como lenguaje de programación, como gestor de base de datos PostgreSQL versión 8.3 o superior para garantizar la integridad y consistencia de la información. Los servidores Apache y PostgreSQL pueden estar instalados en la misma máquina o en máquinas diferentes.

❖ Del lado del cliente se requiere un navegador que interprete las funciones básicas de Java Script, se recomienda Mozilla Firefox 3.0 o superior.

Hardware. Para el desarrollo y puesta en práctica del proyecto se requieren máquinas con los siguientes requisitos:

❖ Se requiere de una máquina que funcione como servidor, la cual debe tener como mínimo:

✓ Un Procesador Pentium III con 1 GHz de frecuencia o superior.

✓ 512 Mb de Memoria RAM.

✓ 20 GB de Disco Duro.

❖ Las computadoras clientes deben estar conectadas en red con el servidor a través de una tarjeta de red de 100 Mbps además deben cumplir los requisitos mínimos para poder ejecutar los navegadores Web.

Restricciones en el diseño y la implementación.

❖ Se definen pautas para la arquitectura de información, de manera que el diseño del sistema sea homogéneo en todas sus interfaces de usuario.

❖ Se definen además, estándares de diseño y codificación por los cuales debe regirse tanto la implementación como los modelos generados como parte del proceso de desarrollo.

2.5 Descripción del sistema propuesto.

2.5.1 Concepción general del sistema

El resultado que se pretende alcanzar con esta propuesta es la obtención de una aplicación propia que gestione la información en el Departamento de Marxismo e Historia de la Uniss, facilitando el acceso y manejo de la información, es decir posibilitando al jefe de este Departamento a la divulgación de sus actividades y logros así como posibilitando a los alumnos maestrantes el acceso a la bibliografía lo que repercute en mayor calidad de trabajo, la aplicación cuenta con una interfaz web. La aplicación brinda la posibilidad de hacer algún procesamiento básico así como acceder a la información contenida en la base de datos mediante una interfaz web.

2.5.1.1 Actores del sistema.

Un actor es un rol que un usuario juega con respecto al sistema. Es importante destacar el uso de la palabra rol, pues con esto se especifica que un Actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema.

A continuación se definen los actores del sistema propuesto:

Actor del Sistema	Descripción
Usuario	Es una persona que puede registrarse, cambiar contraseña, autenticarse y además de ver la información almacenada en el sistema. Requisitos funcionales asociados: RF1, RF2, RF3, RF16, RF17.
Administrador	Es una persona que labora directamente con la información de servicio, se encarga de crear los usuarios del sistema, así como de todo el proceso de soporte y mantenimiento del mismo. Requisitos funcionales asociados: RF1, RF3, RF4, RF5, RF6, RF7, RF8,

	RF9, RF10, RF11, RF12, RF13, RF14, RF15, RF16, RF17.
--	--

Tabla 4 Actores del Sistema

2.5.1.2 Paquetes y sus Relaciones

“Subdividir los casos de uso en paquetes resulta de mucha ayuda en la modelación de cualquier sistema informático. Los paquetes son un mecanismo de organización de elementos que subdividen el modelo en otros más pequeños que colaboran entre sí. Este particionamiento debe hacerse sobre la base de los requerimientos funcionales y el dominio del problema y debe ser reconocible por las personas con conocimiento del dominio.” (Cáceres, 2005, p. 50) Para ello se propone asignar la mayor parte de un cierto número de casos de uso a un paquete concreto. A partir de los criterios expuestos anteriormente se propone el siguiente diagrama de paquetes:

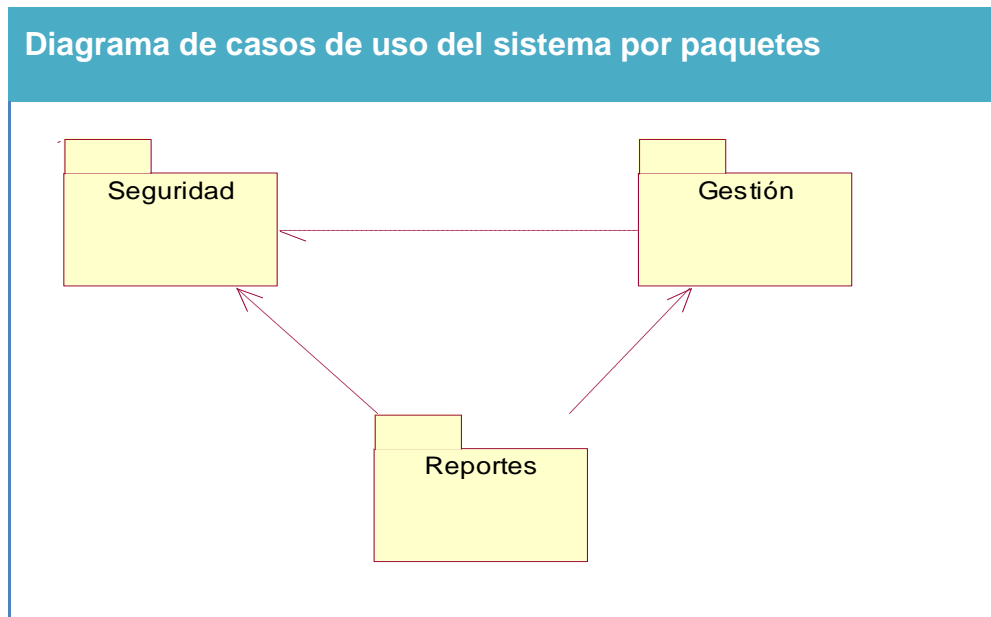


Figura 6: Diagrama de casos de uso del sistema por paquetes.

Paquete Seguridad: En este paquete se agrupan las funcionalidades relacionadas con la registración y cambio de contraseña de los usuarios del sistema. La descripción de los casos de del sistema del paquete seguridad se encuentran en el Anexo1.

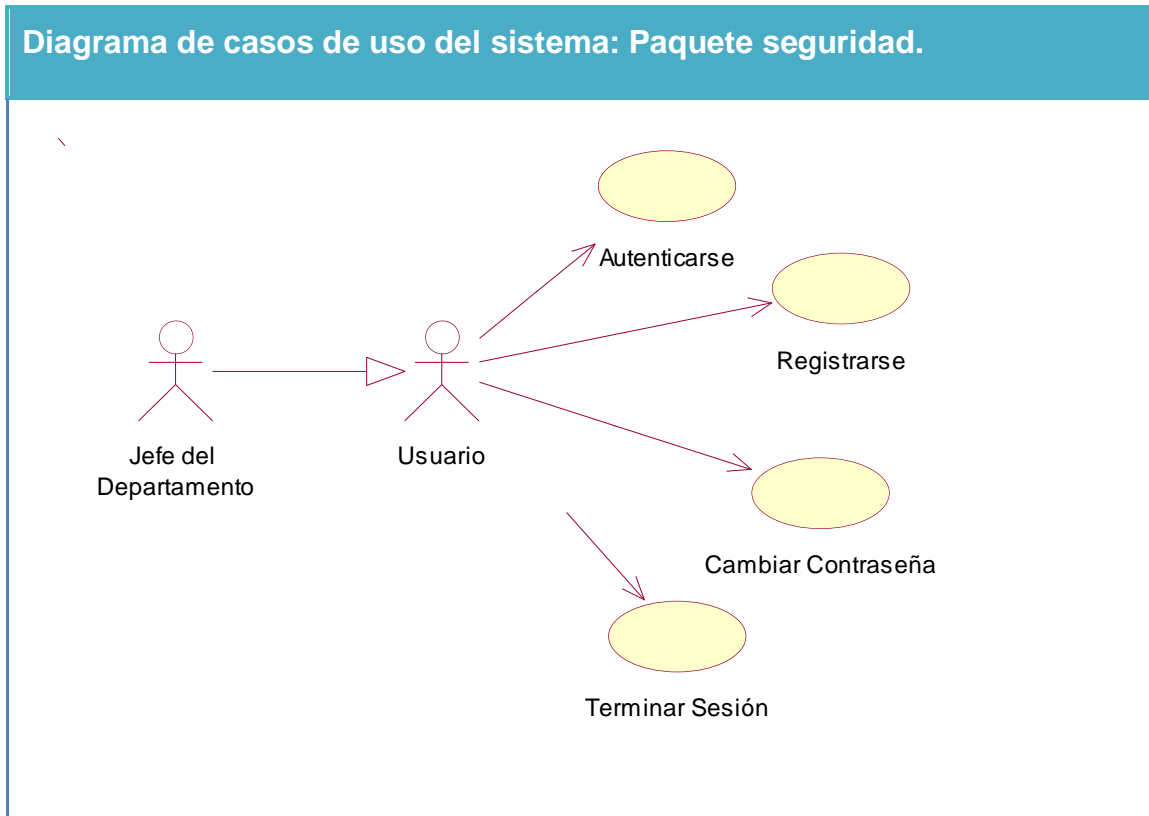


Figura 7: Diagrama de casos de uso del sistema. Paquete de Seguridad

Paquete Gestión Administrativa: En este paquete se agrupan las funcionalidades relacionadas con la actualización de los datos. La descripción de los casos de del sistema del paquete gestión se encuentran en el Anexo2.

Diagrama de casos de uso del sistema: Paquete gestión.

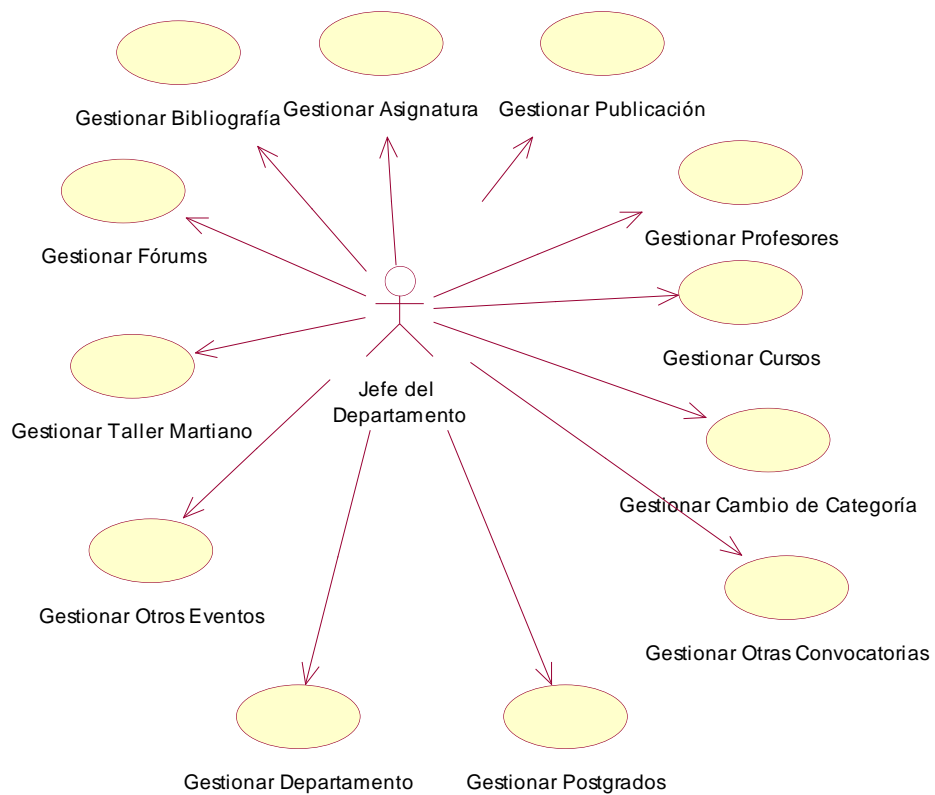


Figura 8: Diagrama de casos de uso del sistema. Paquete de Gestión.

Paquete Visualización de Reportes: En este paquete se agrupan las funcionalidades relacionadas con, la visualización de los datos, filtrados según diferentes criterios de búsqueda; la obtención de reportes. La descripción de los casos de del sistema del paquete reportes se encuentran en el Anexo3.

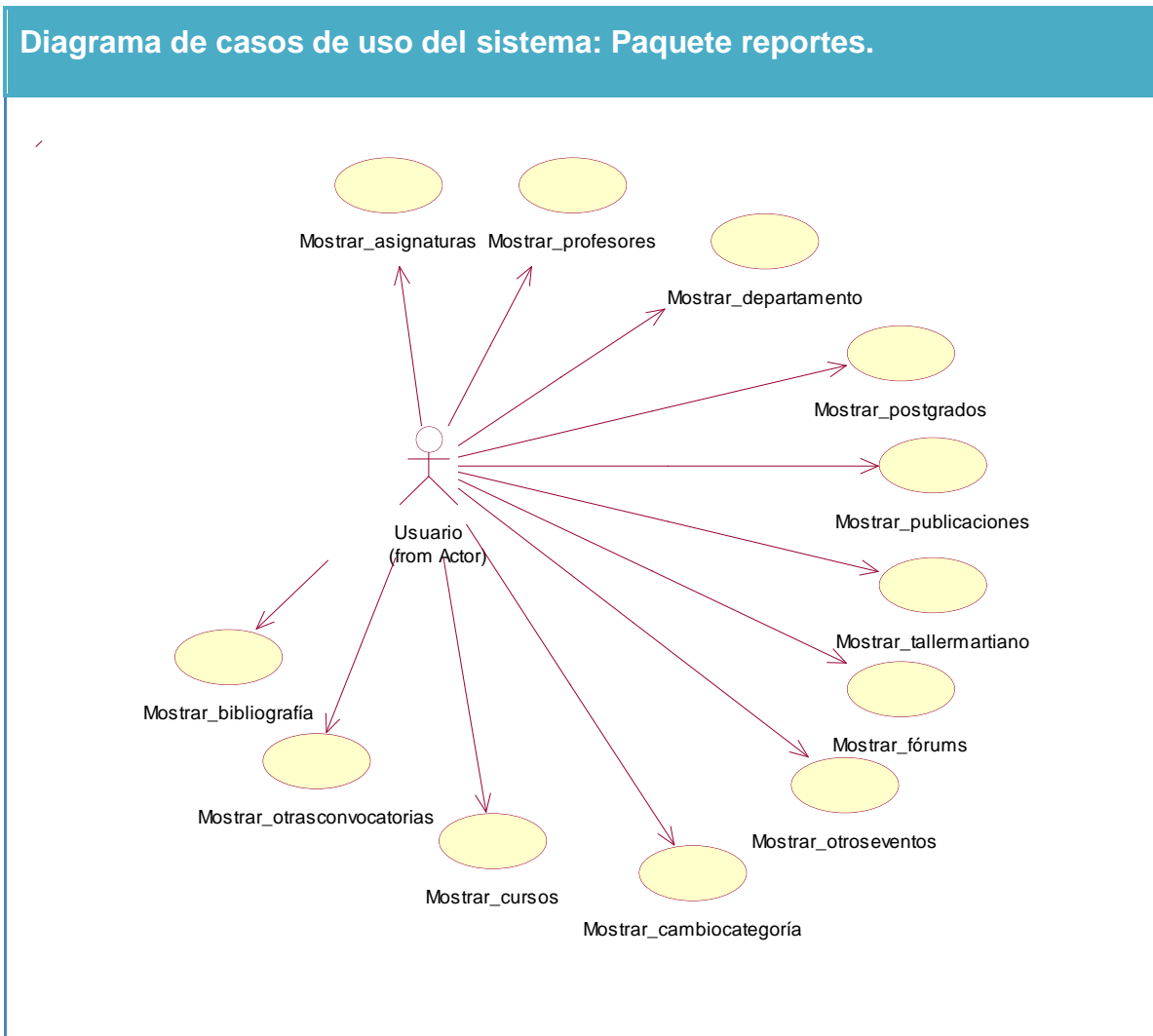


Figura 9: Diagrama de casos de uso del sistema: Paquete reportes.

Paquete: Seguridad

1. Autenticarse
2. Registrarse
3. Cambiar Contraseña
4. Cerrar Sesión

Paquete: Gestión de Información

5. Gestionar Taller Martiano
6. Gestionar Fórum de Historia de la UNISS
7. Gestionar Cambio de Categorías
8. Gestionar Cursos
9. Gestionar Bibliografía
10. Gestionar Postgrados
11. Gestionar Publicación
12. Gestionar Departamento
13. Gestionar Asignatura
14. Gestionar Profesor
15. Gestionar Otras Convocatorias
16. Gestionar Otros Eventos

Paquete: Salida Reportes

17. Reportes:
 - Listar Departamento
 - Listar Profesores
 - Listar Publicaciones
 - Listar Asignaturas
 - Listar Fóruns
 - Listar Talleres Martianos
 - Listar Cursos
 - Listar Cambio de Categorías
 - Listar Otras Convocatorias
 - Listar Otros Eventos

2.5.1.3 Diagramas de casos de uso del sistema.

El Diagrama de Caso de Uso es muy útil en la determinación de los requerimientos, en la comunicación con los clientes y en generar los casos de prueba. Los diagramas de casos de uso proporcionan una visión externa del sistema. Un caso de uso se modela para todos los procesos que el sistema debe llevar a cabo. Los procesos se describen dentro del caso de uso por una descripción textual o una secuencia de pasos ejecutados. De manera más precisa, un Caso de Uso especifica una secuencia de

acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia.

2.5.1.4 Descripción de los casos de uso del sistema.

En este epígrafe se realiza una descripción textual detallada de todos los casos de uso del sistema. Por cada caso de uso se exponen aspectos tales como: los actores que intervienen en ellos, su propósito general, un resumen de cómo se realizan, las referencias a los requisitos que cubren, entre otros. Las descripciones de los casos de uso aparecen en el anexo 1, 2 y 3.

Conclusiones

En este capítulo se comenzó a desarrollar la propuesta de solución, obteniéndose a partir del análisis de los procesos del negocio, un listado con las principales funcionalidades que debe tener el sistema y los requisitos adicionales, se representaron los Diagramas de Casos de Uso del Sistema, y finalmente se describieron las acciones de los actores del sistema con los casos de uso con los que interactúan.

CAPÍTULO III. IMPLEMENTACIÓN.

3.1 Introducción

En este capítulo se describen los diagramas de las clases de la aplicación, el diseño de la base de datos, el tratamiento de errores, así como el modelo de implementación al describir los componentes a construir y su organización y dependencia entre nodos físicos en los que funcionará a aplicación, es decir se define la organización del sistema en subsistemas y se describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración.

3.2 Diagrama de clases del diseño

El Diagrama de Clases es el diagrama principal de diseño y análisis para un sistema. Presenta las clases, junto con sus atributos, operaciones, interfaces y relaciones. Se convierte en el diagrama central del análisis del diseño orientado a objetos, y el que muestra la estructura estática del sistema. Durante el análisis del sistema, el diagrama se desarrolla buscando una solución ideal. Durante el diseño, se usa el mismo diagrama, y se modifica para satisfacer los detalles de las implementaciones. Puede ser dividido en capas: aplicación, y datos, las cuales muestran las clases que intervienen con la interfaz de usuario, la lógica del software de la aplicación, y el almacenamiento de datos respectivamente. Presenta las clases del sistema con sus relaciones (estructurales y de herencia). En el caso de las aplicaciones web, el diagrama de clases representa las colaboraciones entre las páginas, donde cada página lógica puede ser representada como una clase.

Los diagramas de clases web, fueron realizados a partir de los diferentes casos de uso del sistema y empleando las extensiones de UML para web. Seguidamente se muestran los diagramas realizados.

3.2.1 Paquete de Seguridad

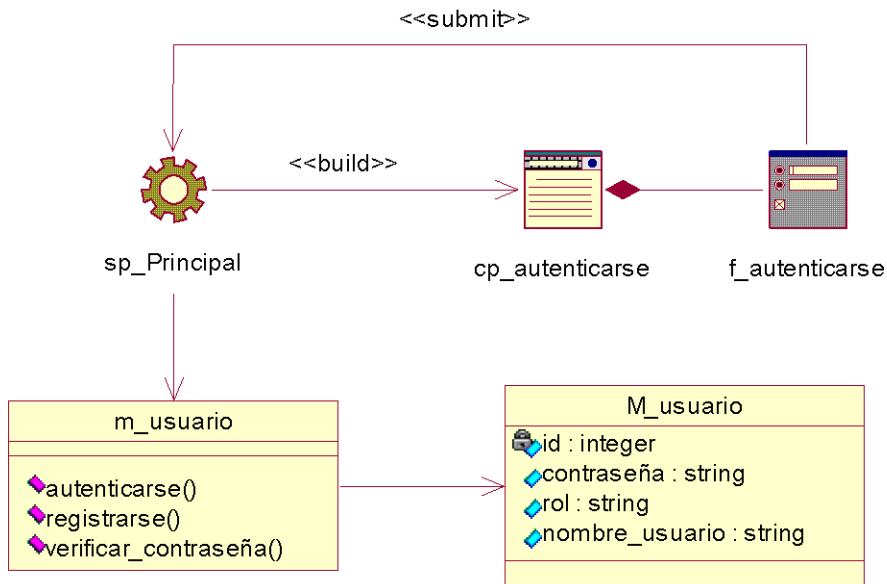


Figura 10: Diagrama de clases web Caso de Uso Autenticar usuario

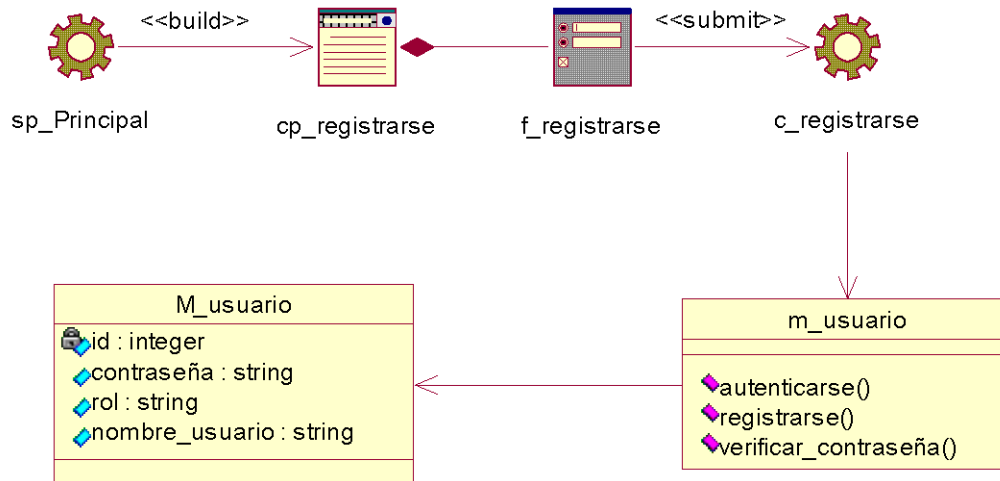


Figura 11: Diagrama de clases web Caso de Uso Registrar usuario

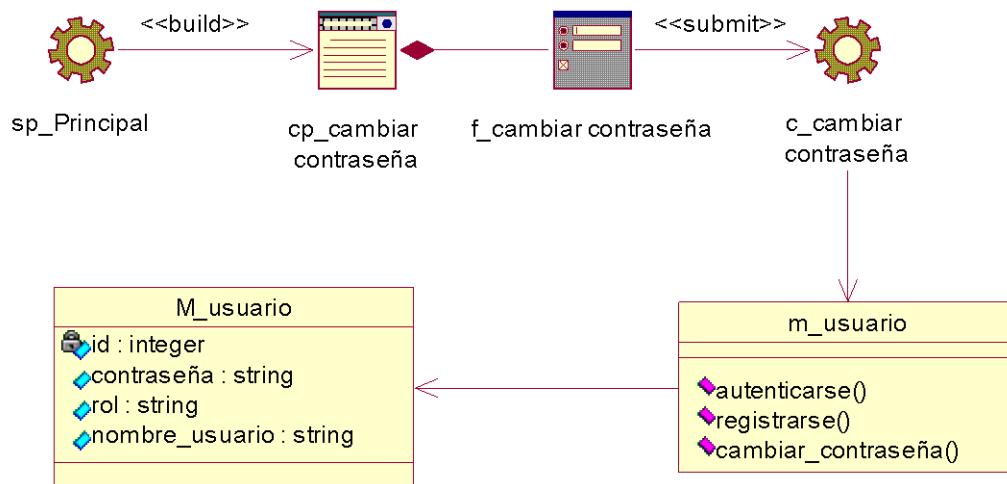


Figura 12: Diagrama de clases web Caso de Uso Cambiar Contraseña.

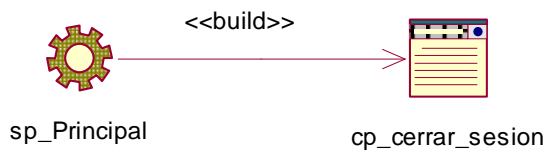


Figura 13: Diagrama de clases web Caso de Uso Cerrar Sesión

3.2.2 Paquete de Gestión

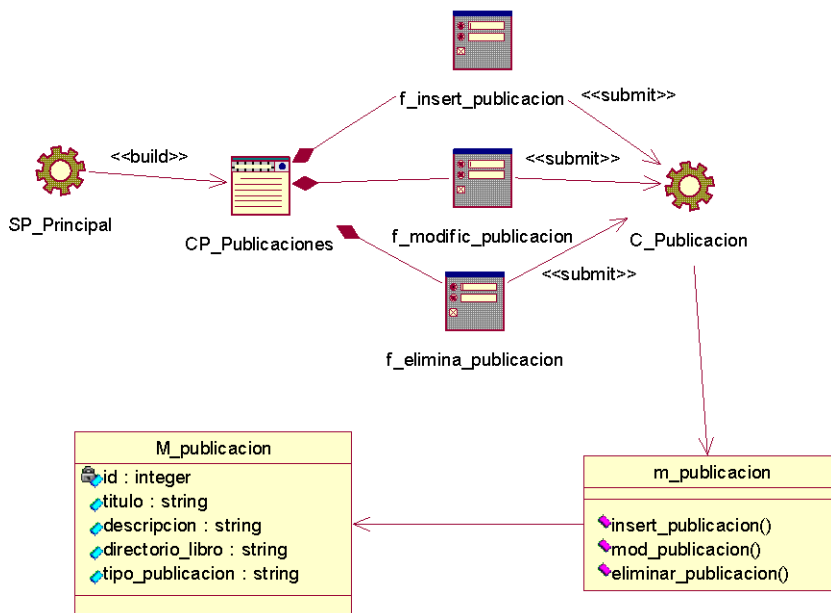


Figura 14: Diagrama de clases web Caso de Uso Gestionar Publicación.

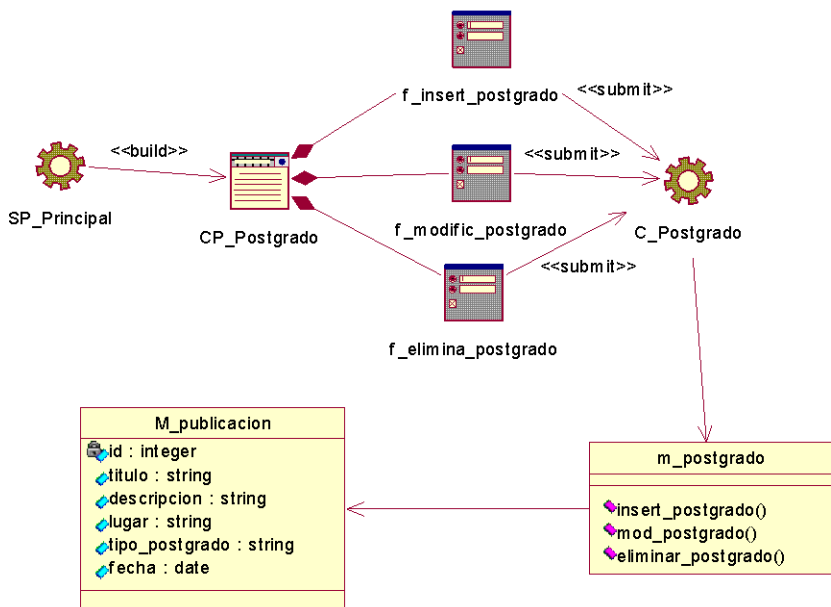


Figura 15: Diagrama de clases web Caso se Uso Gestionar Postgrado

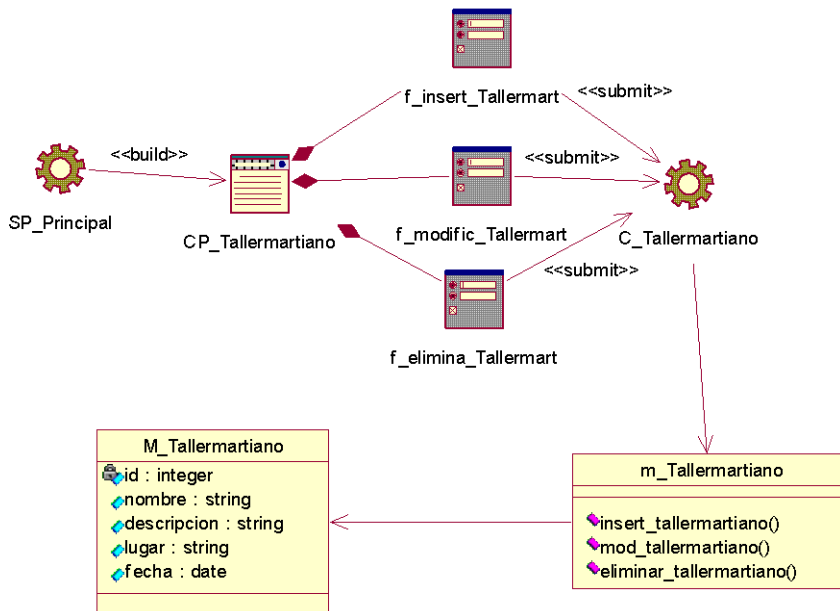


Figura 16 :Diagrama de clases web Caso de Uso Gestionar Taller Martiano

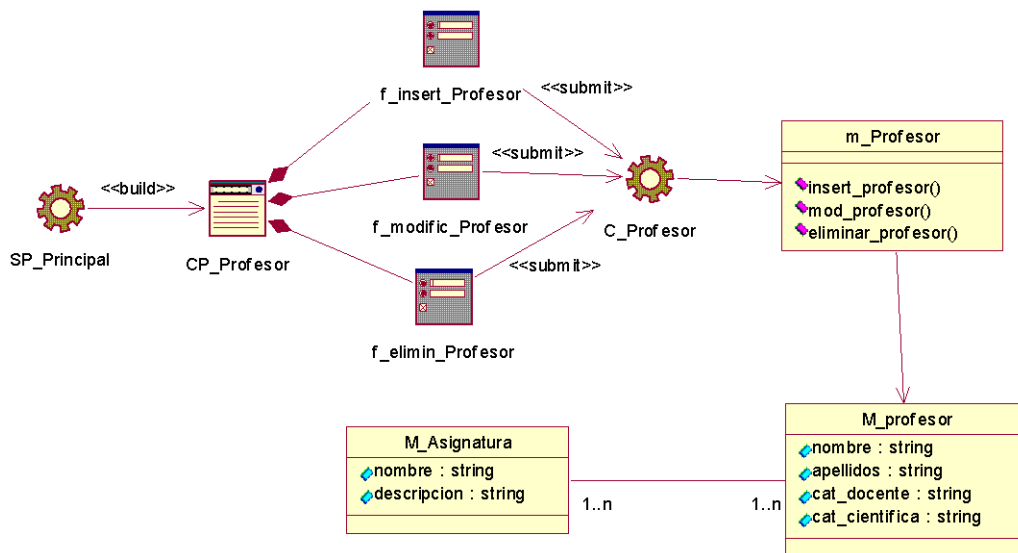


Figura 17: Diagrama de clases web Caso de Uso Gestionar Profesor

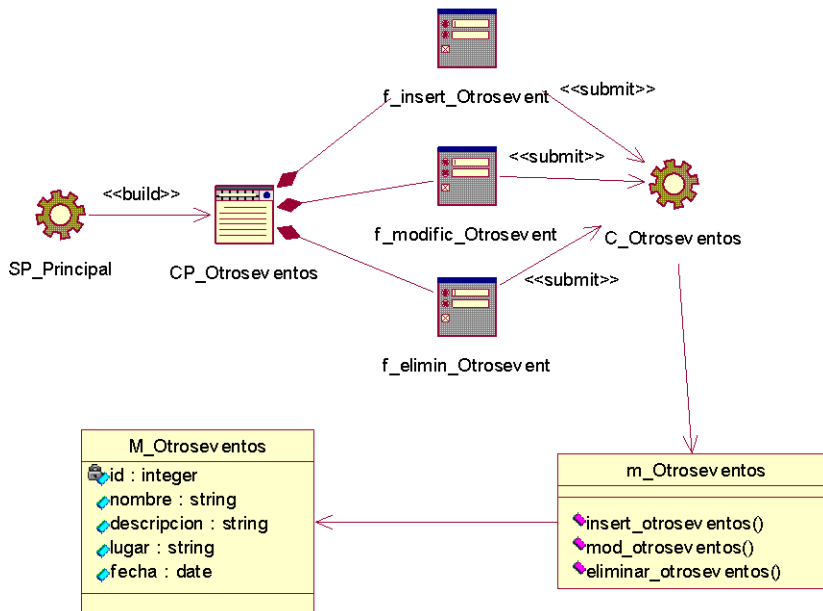


Figura 18: Diagrama de clases web Caso de Uso Gestionar Otros Eventos

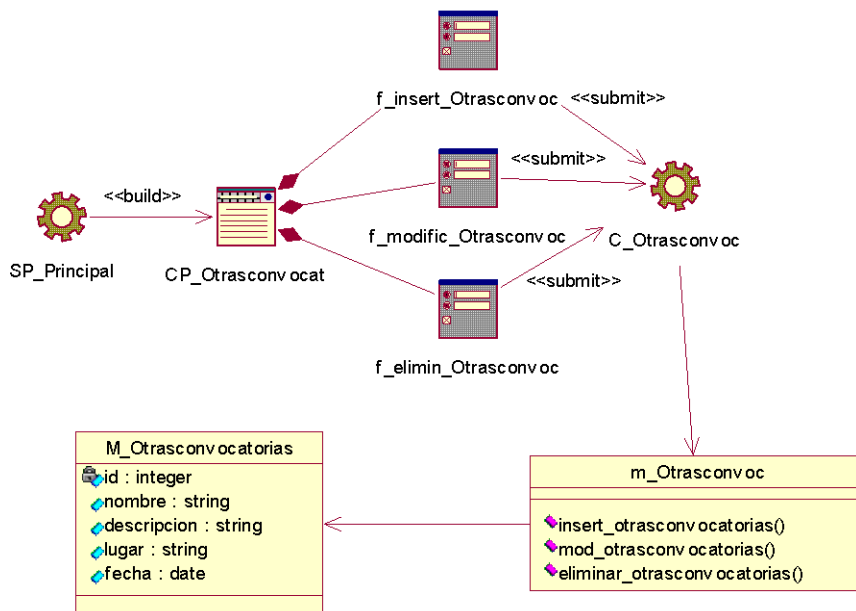


Figura 19: Diagrama de clases web Caso de Uso Gestionar Otras Convocatorias

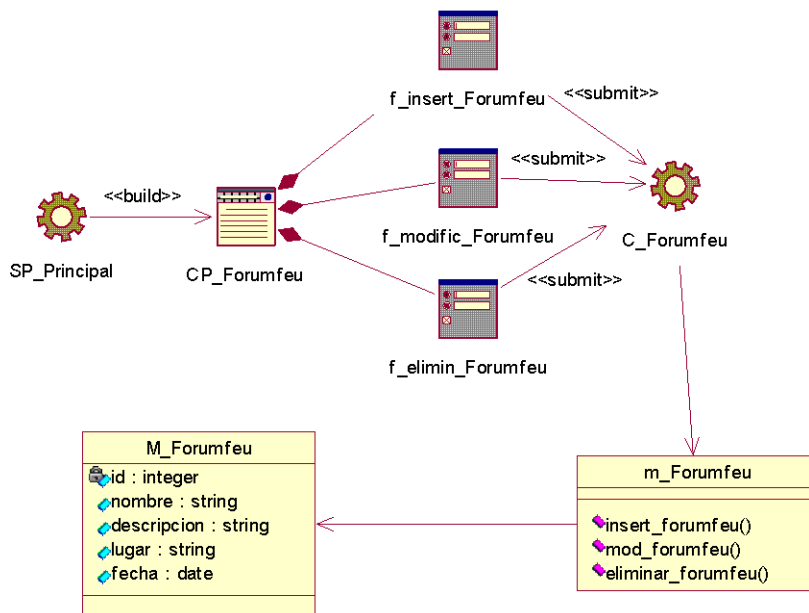


Figura 20: Diagrama de clases web Caso de Uso Gestionar Forum Feu

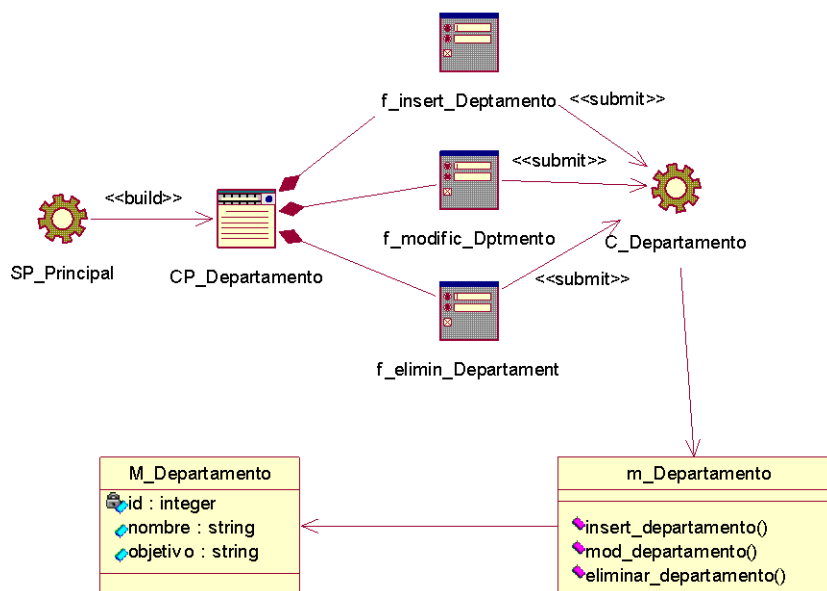


Figura 21: Diagrama de clases web Caso de Uso Gestionar Departamento

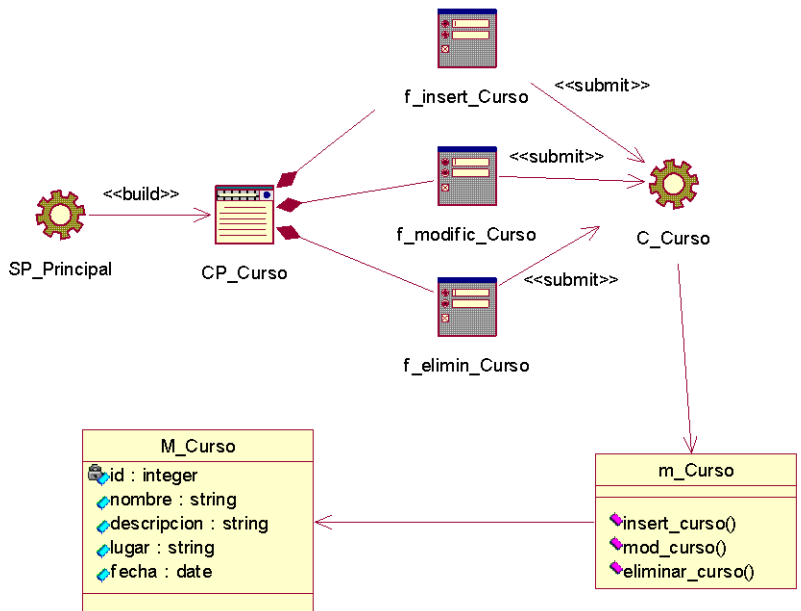


Figura 22: Diagrama de clases web Caso de Uso Gestionar Curso

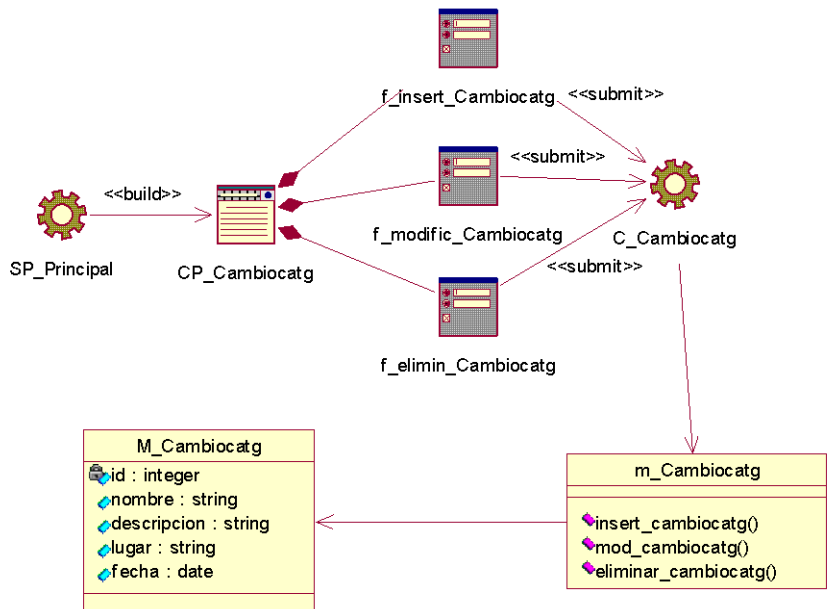


Figura 23: Diagrama de clases web Caso de Uso Gestionar Cambio de Categoría

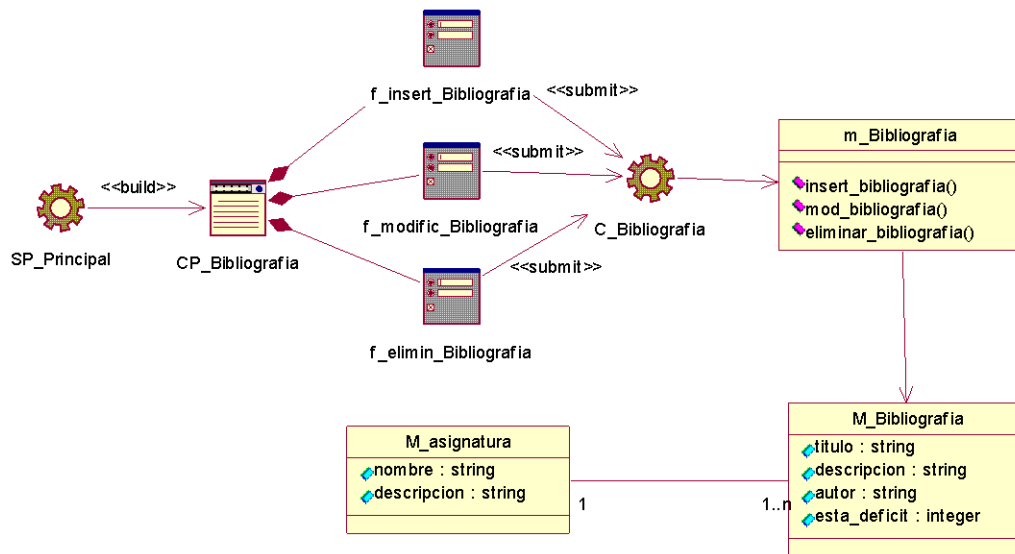


Figura 24: Diagrama de clases web Caso de Uso Gestionar Bibliografía

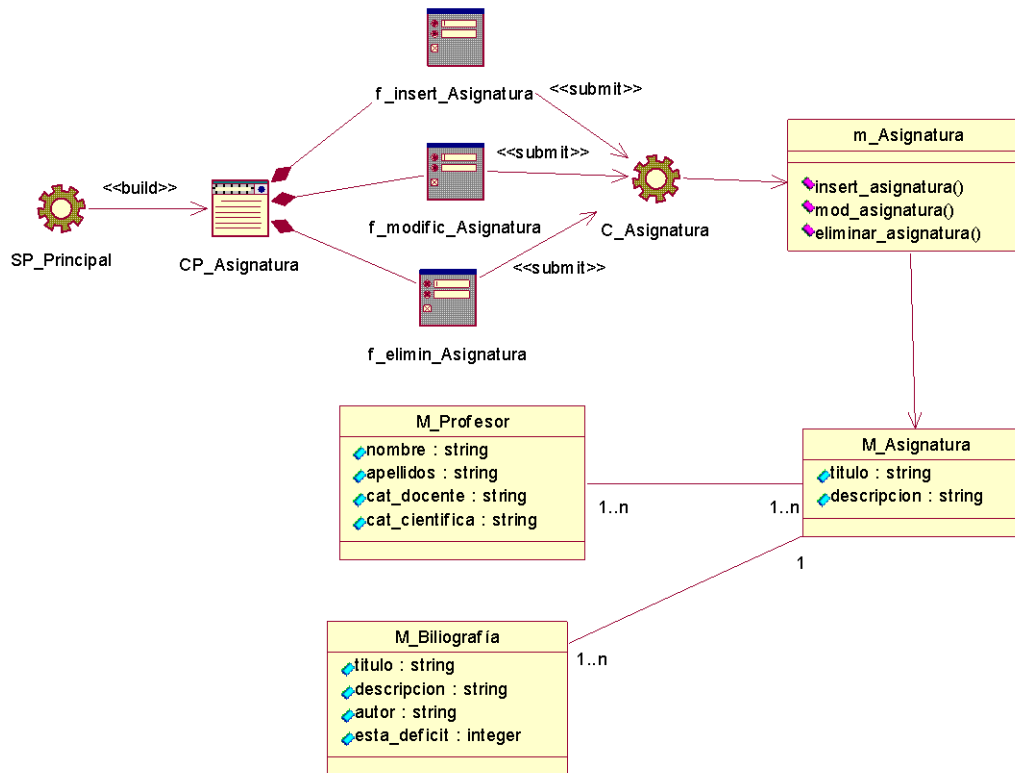


Figura 25: Diagrama de clases web Caso de Uso Gestionar Asignatura

3.2.3 Paquete de Reportes

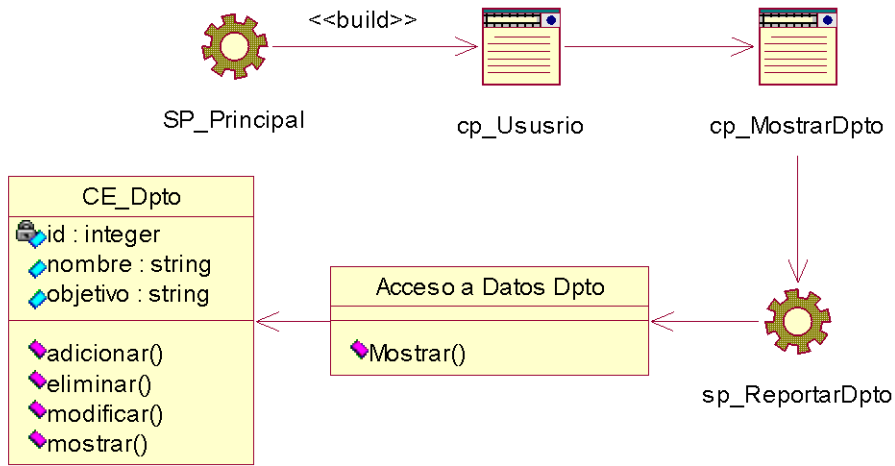


Figura 26: Diagrama de clases web Caso de Uso Mostrar Departamento

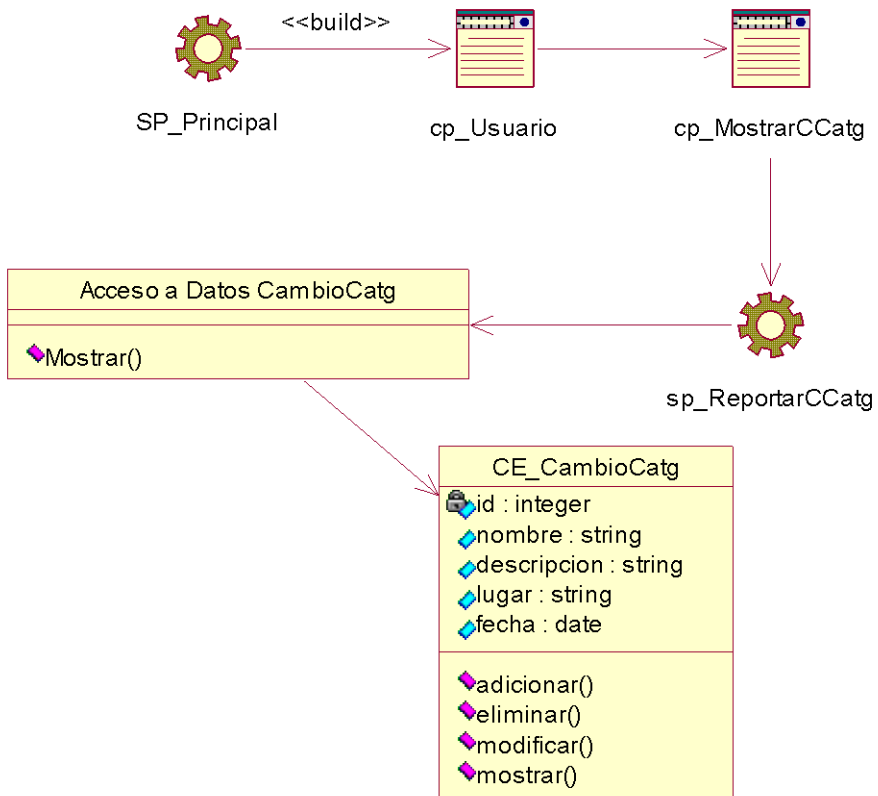


Figura 27: Diagrama de clases web Caso de Uso Mostrar Cambio de Categoría

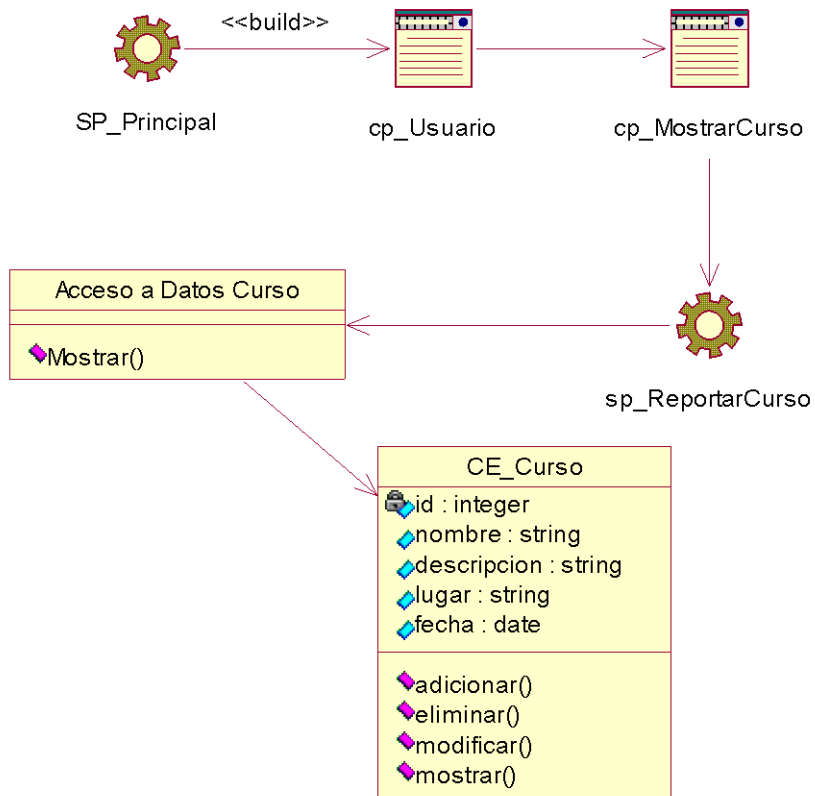


Figura 28: Diagrama de clases web Caso de Uso Mostrar Curso

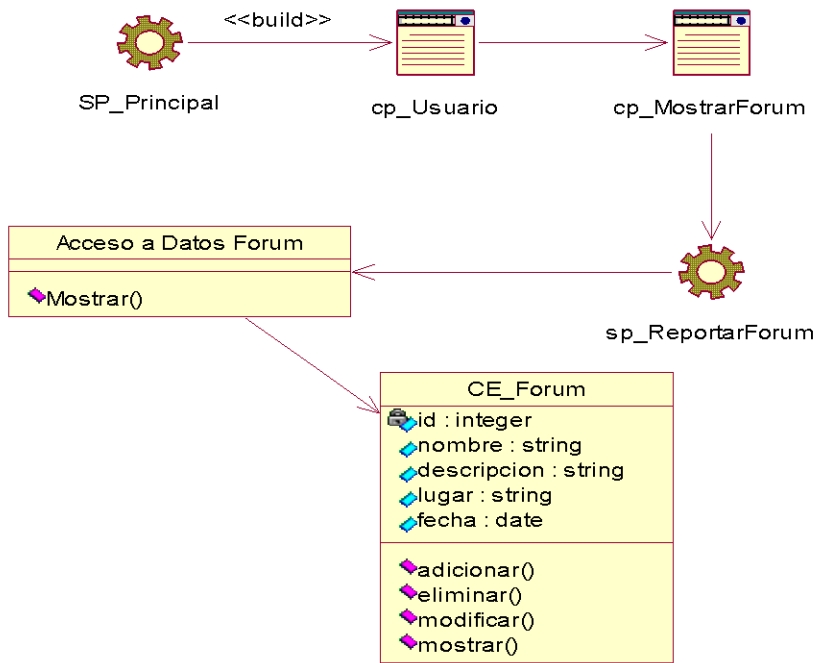


Figura 29: Diagrama de clases web Caso de Uso Mostrar Fóruns de Historia de la FEU

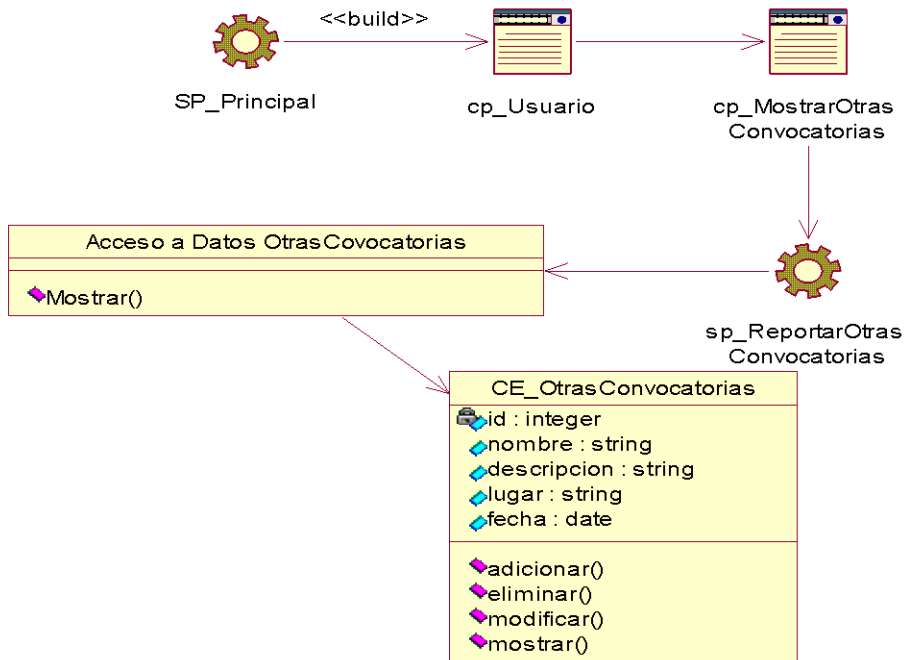


Figura 30: Diagrama de clases web Caso de Uso Mostrar Otras Convocatorias

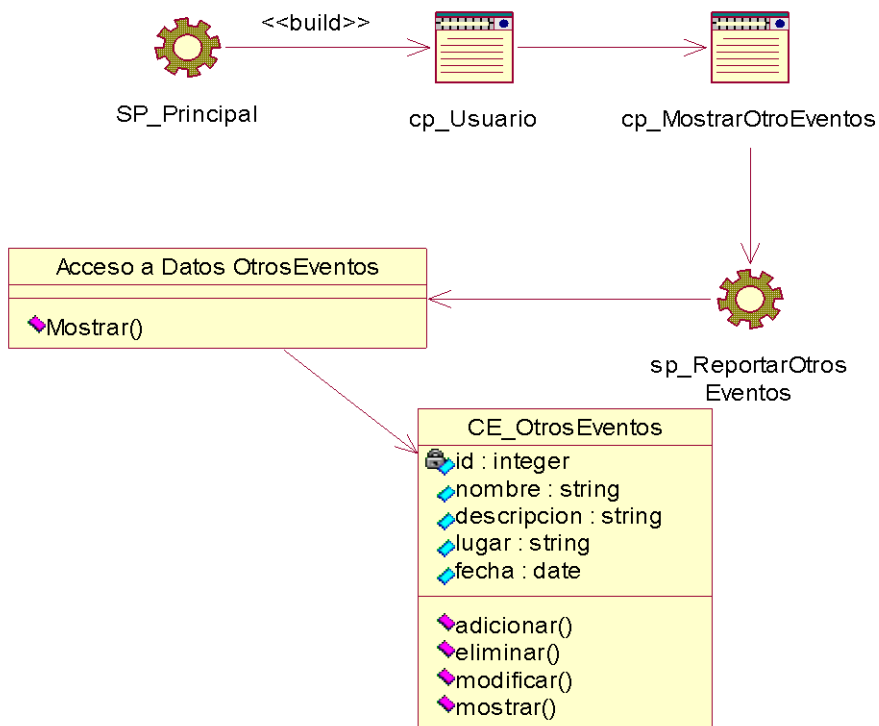


Figura 31: Diagrama de clases web Caso de Uso Mostrar Otros Eventos

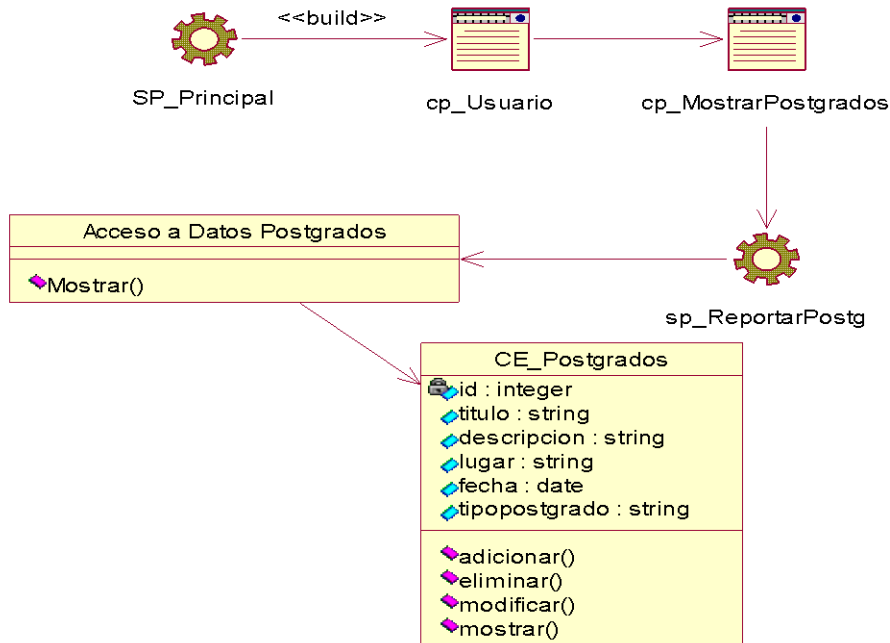


Figura 32: Diagrama de clases web Caso de Uso Mostrar Postgrados

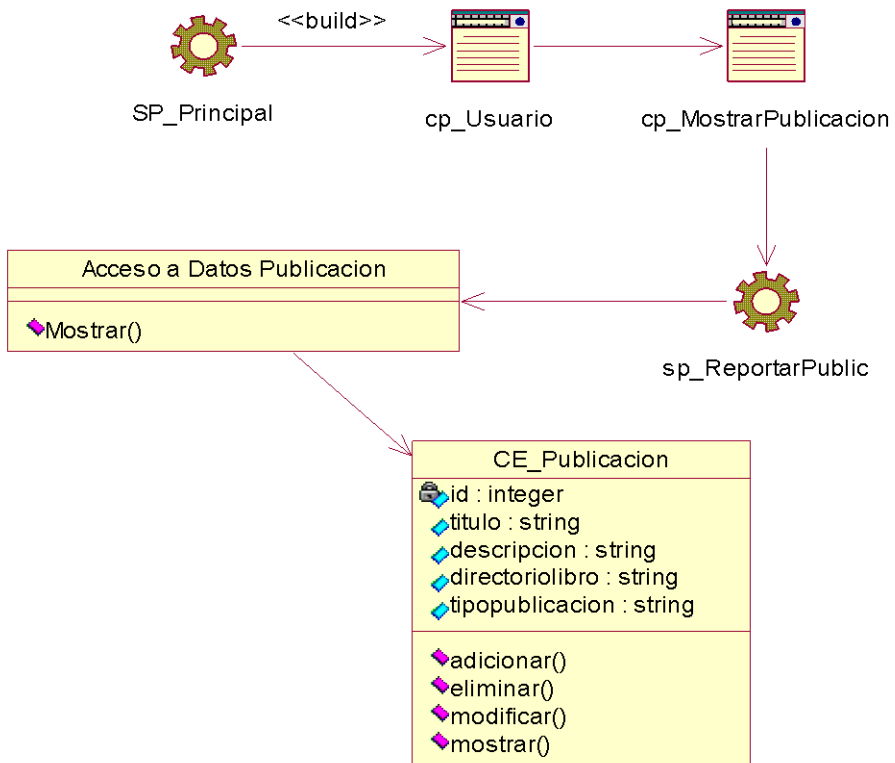


Figura 33: Diagrama de clases web Caso de Uso Mostrar Publicación

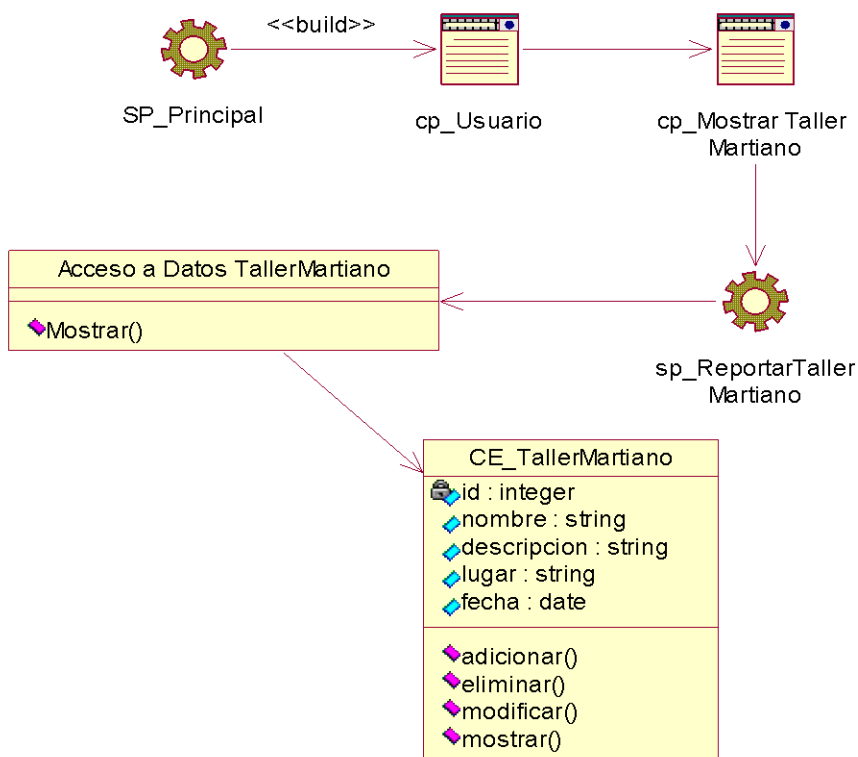


Figura 34: Diagrama de clases web Caso de Uso Mostrar Taller Martiano

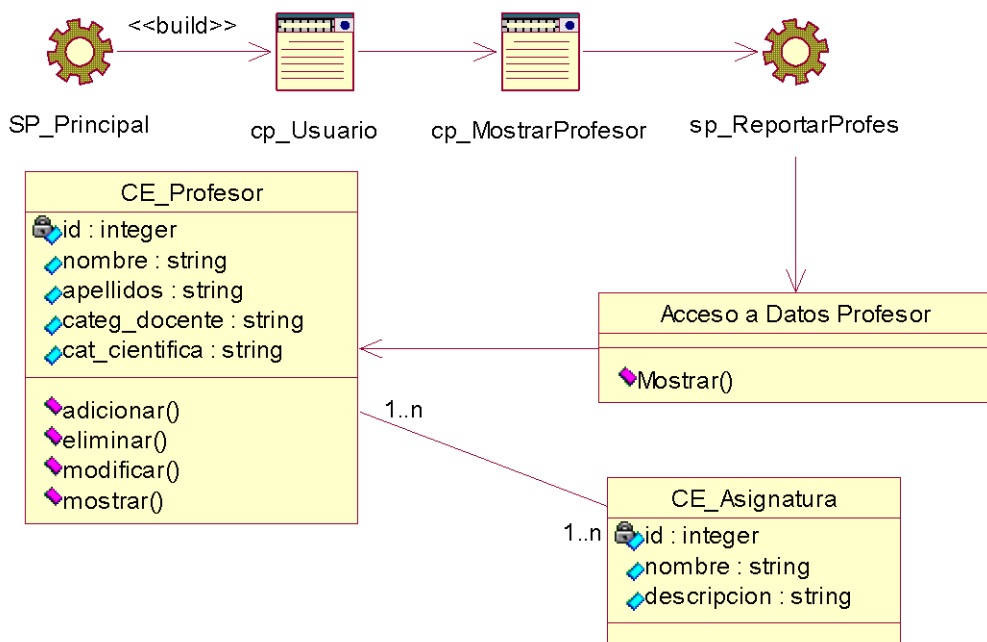


Figura 35: Diagrama de clases web Caso de Uso Mostrar Profesor

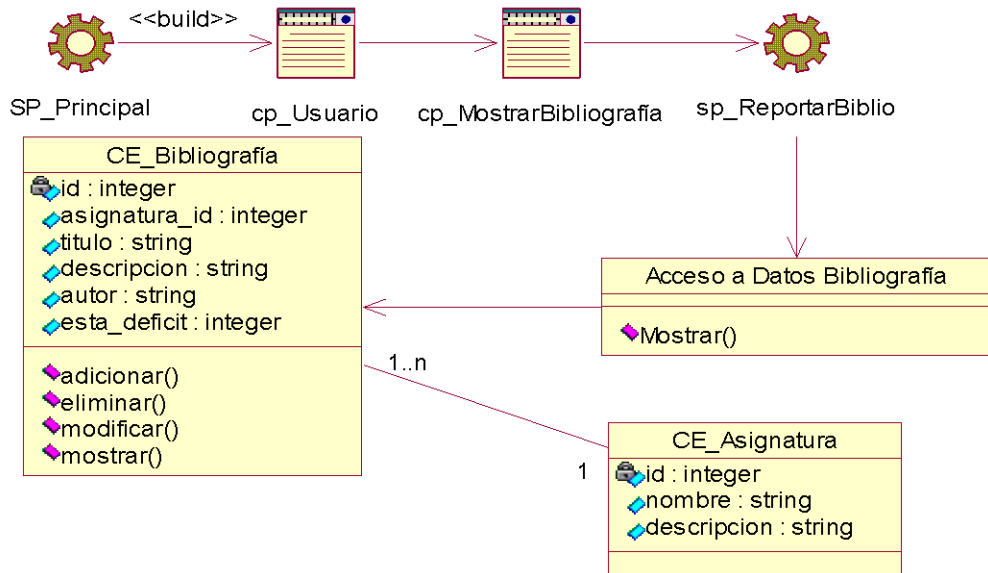


Figura 36: Diagrama de clases web Caso de Uso Mostrar Bibliografía

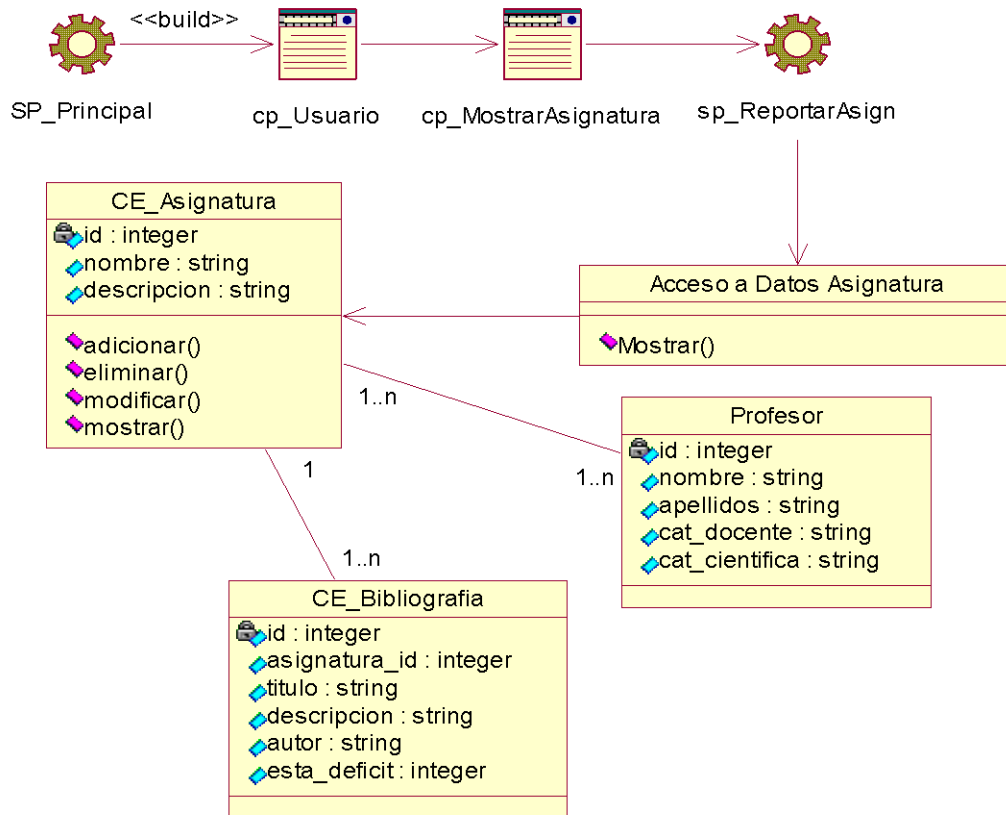


Figura 37: Diagrama de clases web Caso de Uso Mostrar Asignatura

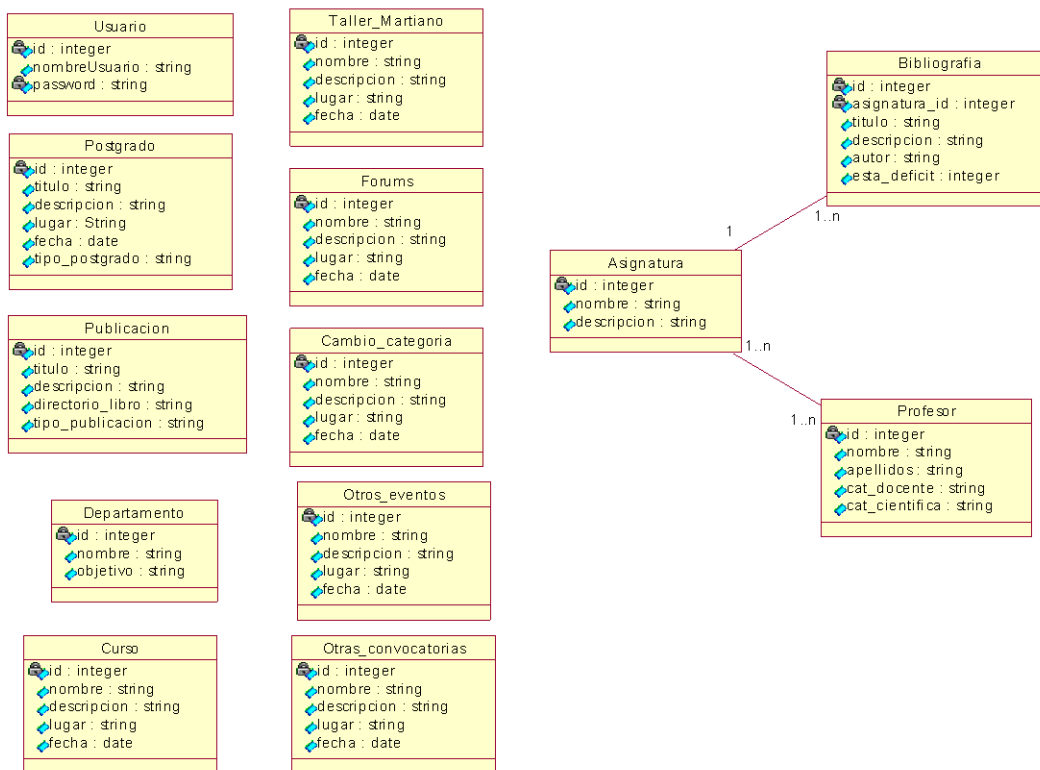
3.3 Diseño de la base de datos

Una base de datos o banco de datos (del inglés Database) es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. El diseño de la base de datos es una de las tareas más importantes en la construcción de un sistema, ya que una base de datos bien diseñada contendrá información correcta, almacenará los datos más eficientemente y será más fácil de gestionar y de mantener. (Wesley, Introducción a los Sistemas de Bases de Datos, 2005)

3.3.1 Modelo lógico de datos

El diagrama del modelo lógico de datos o diagrama de clases persistentes, muestra las clases capaces de mantener su valor en el espacio y en el tiempo. (Cáceres, 2005) A partir de este planteamiento se definieron las clases que participan en el modelo lógico de datos.

Figura 38: Diagrama de Clases Persistentes.



3.3.2 Modelo físico de datos

Cuando se define correctamente el modelo lógico, se hace mucho menos engorroso llegar al modelo de datos o modelo físico como también se le denomina en la metodología RUP de la siguiente forma: *“el modelo de datos representa la estructura o descripción física de las tablas de la base de datos y es obtenido a partir del diagrama de clases persistentes”*. (Cáceres, 2005, p. 91) En el Anexo 4 aparece el diagrama clases del modelo físico para el sistema propuesto.

3.4 Principios de diseño

El diseño de la interfaz de una aplicación, el formato de los reportes, la concepción de la ayuda y el tratamiento de excepciones tiene gran influencia en el éxito o fracaso de una aplicación. A continuación se describen los principios de diseño que deben tenerse en cuenta para el desarrollo del sistema.

3.4.1 Estándares en la interfaz de la aplicación

“La interfaz es en realidad un modelo mental permanente, es decir una representación cognitiva o conceptualización que el usuario hace del sistema. A fin de que este modelo se mantenga a lo largo del programa ha de tener una consistencia, es decir mantener su coherencia de principio a fin.” (Machado, 2006, p. 103) Por ello se han de mantener las reglas, los criterios en la operatividad, la imagen parcial o total, etc.; pues una incoherencia de diseño puede aportar pérdidas de eficacia del propio contenido que se quiera transmitir. Para lograr un adecuado diseño de la interfaz de la aplicación web, se tuvieron en cuenta las ideas siguientes:

- El uso adecuado de iconos e imágenes relativamente pequeñas que facilite la comprensión de las funcionalidades del sistema.
- Un menú acorde en gran medida a los requerimientos funcionales, no funcionales y a la temática en cuestión.
- Las etiquetas de los formularios deben contener un lenguaje claro y evitarse la carga visual de los mismos.

- La interfaz diseñada para la aplicación web dinámica debe estar concebida para la resolución 800x600 píxel.
- La navegabilidad por las páginas debe ser consistente y evitando la ruptura de hipervínculos.
- Es aconsejable utilizar colores claros y relajantes a la vista del usuario para que este se sienta cómodo mientras interactúa con el sistema.

3.4.2 Formatos de reportes

- Los reportes se obtendrán de una sola forma. La información será mostrada en forma de listado.
- Los reportes serán concebidos sobre páginas web, utilizando un formato de letra clara, legible y con colores claros para no recargar y hacer engorroso su diseño.
- El sistema tiene dentro de sus principales funcionalidades, además de mostrar toda la información referente a la divulgación, gestionar todo su contenido, mostrar una serie de reportes y que el usuario pueda realizar solicitudes para hacer publicaciones científicas y para matricular en cursos.

3.4.3 Concepción general de la ayuda

Dentro del mundo de los sitios web en general, la ayuda constituye una parte importante del sistema. Las tendencias actuales apuntan a que estas no deben ser muy detallistas o extensas, sino simplemente explicaciones sencillas y aclaraciones del producto y de las operaciones que puede realizar el usuario sobre el mismo. Se concibió una ayuda amigable y práctica para el módulo del sistema relacionado con la entrada y actualización de los datos. La ayuda de la aplicación quedó formada por un menú interno que contiene todas las funcionalidades del sistema. El usuario podrá navegar por cada uno de ellos y consultar su funcionalidad, descrita de un modo fácil de entender.

3.4.4 Tratamiento de errores

El sistema ha sido implementado de manera tal que se minimicen los errores aplicando técnicas de validación de datos y mediante la cuidadosa confección de la interfaz de entrada salida. Los mensajes de error que emite el sistema se mostrarán en un lenguaje claro y de fácil comprensión a personas sin conocimientos avanzados de computación. Además de esto se pedirá la confirmación por parte del usuario de operaciones que impliquen riesgos de pérdida de datos, tales como eliminación.

3.4.5 Seguridad

Para lograr la seguridad y protección de la información almacenada, el sistema exige a los usuarios un login y una contraseña para poder acceder al mismo, controlándose de esta forma los niveles de acceso a la información. Solo podrán actualizar información presente en la base de datos del sistema, los usuarios que cuenten con dichos privilegios. La consistencia de los datos es otro aspecto que se toma en consideración, y para ello el sistema cuenta con formularios validados, con funciones del lenguaje PHP que garantizan que la información que se registre en la base de datos sea totalmente consistente e íntegra.

3.5 Modelo de implementación

El modelo de implementación “describe cómo los elementos del modelo de diseño se implementan en términos de componentes” (Jacobson, Booch, &Rumbauch, 2006), cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y cómo dependen unos de otros.

3.5.1 Modelo de despliegue

El diagrama de despliegue contiene los nodos que conforman la topología de hardware sobre la cual se ejecuta el sistema software implementado (aplicación web). En la siguiente figura se muestra el Diagrama de despliegue de la aplicación desarrollada. El mismo cuenta con tres nodos.

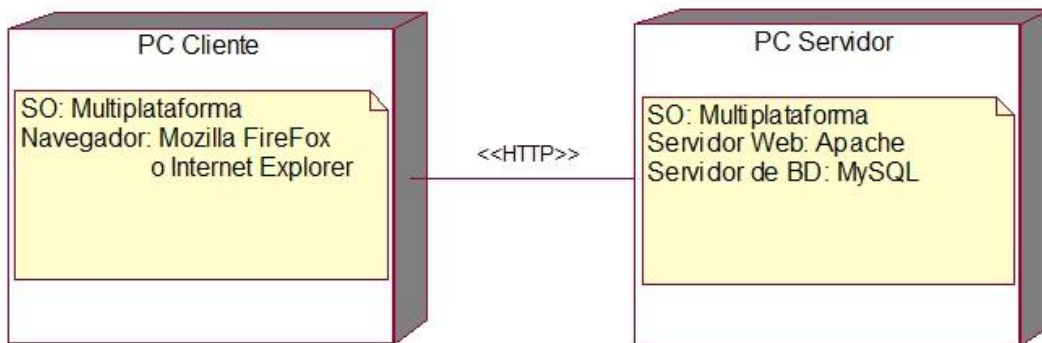


Figura 39: Diagrama de Despliegue.

3.5.2 Modelo de Componentes

Los diagramas de componentes se utilizan para modelar la vista de implementación estática de un sistema. Muestran tanto los componentes de software (código fuente, binario y ejecutable) como las relaciones lógicas entre ellos en un sistema. Y como todos los diagramas, también pueden contener paquetes utilizados para agrupar elementos del modelo.

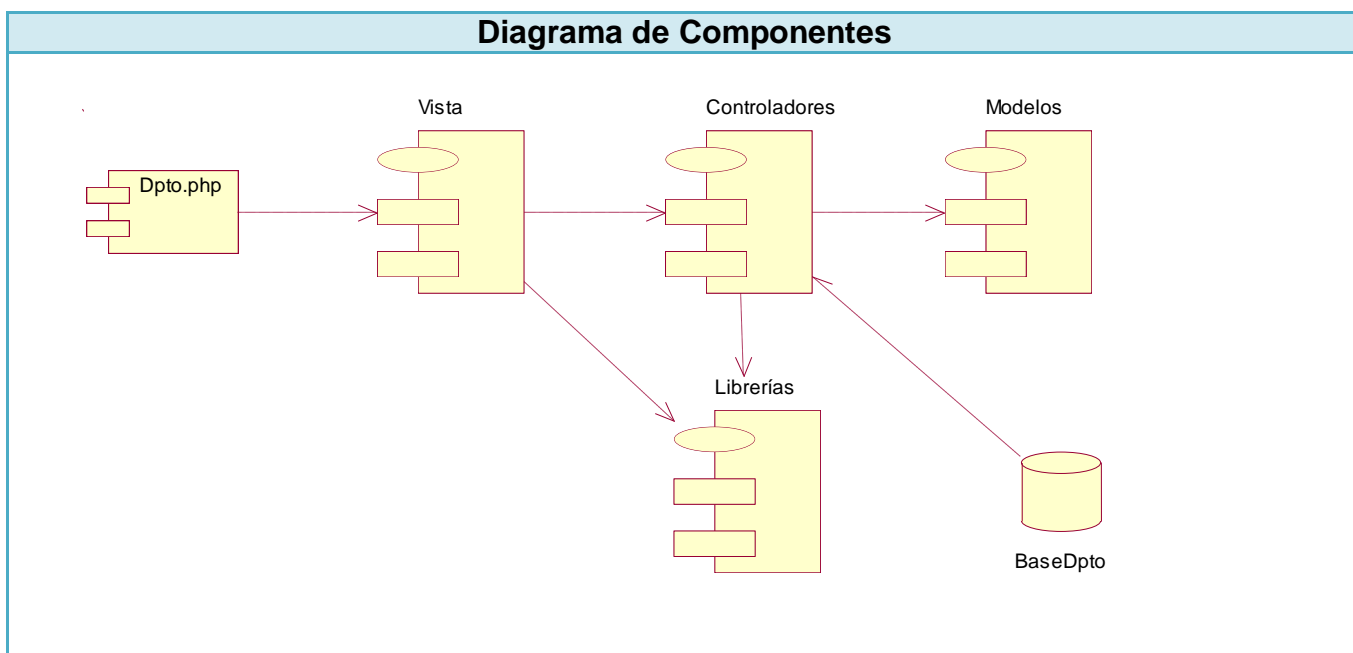


Figura 40: Diagrama de Componentes.

Descripción del diagrama de componentes

Paquete vista: Agrupa los ficheros de las clases de interfaz.

Paquete controlador: Agrupa las clases controladoras, y las de Acceso a Datos que actúan como intermediarias entre las clases del paquete modelo y el paquete vista.

Paquete modelo: Contiene las clases de acceso a los datos.

BaseDpto: Representa al Sistema Gestor de Base de Datos donde se guarda la información de la aplicación, este caso se utilizó MySQL.

Paquete librería: Este paquete contiene las librerías para la validación, generación de tablas, disponibles en el marco de trabajo (framework) Symfony, además de las librerías JavaScript utilizadas para las interfaces que se muestran.

Conclusiones

En este capítulo se han descrito los diferentes pasos que componen el diseño y la implementación del sistema. Se desarrollaron los diagramas de clases de aplicaciones web, el diseño de la base de datos, el modelo de implementación y se definieron además una serie de políticas que tienen en cuenta el diseño de la interfaz, el tratamiento de los posibles errores, entre otras. Se describió la distribución del sistema en nodos, especificados en el diagrama de despliegue y se mostró el diagrama de componentes. Todo ello con el objetivo de lograr una herramienta automatizada que cumpla con las expectativas del cliente.

Conclusiones Generales

➤ Los antecedentes teóricos metodológicos identificados exponen la utilidad de la aplicación del sitio web como medio y herramienta de trabajo para la gestión de la información en el Departamento de Marxismo e Historia de la Uniss.

- Se diseñó un sitio web partiendo de la descripción de los procesos del negocio y de la modelación de los casos de uso correspondientes al sistema, así como los requisitos funcionales y no funcionales definidos por el usuario. Permitted determinar el empleo de la metodología RUP con el lenguaje UML para realizar el diseño y modelado de la aplicación.
- Se implementó un sitio web que siguió los principios del diseño, los estándares de codificación y se ajustó a los requerimientos funcionales del sistema. Para su construcción se empleó el Framework Symfony, también se utilizaron los lenguajes PHP, Java Script, HTML y CSS. Como gestor de base de datos se utilizó MySQL, el cual cuenta con un buen desempeño y es de código abierto. El sistema se desplegó a través de dos nodos en los cuales se distribuyen los componentes de la aplicación, la cual está basada en una arquitectura cliente - servidor con tres capas que proporciona escalabilidad y posibilidades de mantenimiento del código. Dicho sitio está sustentado en los preceptos del software libre y el patrón Modelo Vista Controlador (MVC), lo que brinda flexibilidad y posibilidades para futuras modificaciones y mejoras.

RECOMENDACIONES:

Para el enriquecimiento de este trabajo se recomienda:

- Añadir un foro de discusión para el intercambio de conocimientos alumno-profesor e ideas a través del sitio web.
- Añadir una biblioteca virtual para que los usuarios tengan otras opciones de adquisición de conocimientos.

BIBLIOGRAFÍA:

Álvarez, M. A., Monteiro Lázaro, J., & Méndez, N. (2009). *DesarrolloWeb.com*. Recuperado el 24 de Febrero de 2013, de Introducción a los lenguajes del web: <http://www.desarrolloweb.com/manuales/27/>

Álvarez, M. A., Monteiro Lázaro, J., Álvarez, R., Matías Sebely, M., González Seco, J. A., Masip, D., y otros. (2009). *DesarrolloWeb.com*. Recuperado el 24 de Febrero de 2013, de Qué es cada tecnología: <http://www.desarrolloweb.com/manuales/15/>

Bologna, J., & Walsh, A. M. (1997). *"The Accountant's Handbook of Information Technology"* (Vol.I). John Wiley and Sons.

Date, C. J. (2001). *Introducción a los sistemas de bases de datos* (7ma ed.). México: Pearson Education.

Elmasri, R., & Navathe, S. B. (2004). *Fundamentals of Database Systems* (4ta ed.). USA: Pearson Education.

Free Software Foundation. (2008). *Free software definition*. Recuperado el 25 de Febrero del 2013, de <http://www.gnu.org/philosophy/free-sw.html>

Jacobson, I., Booch, G., & Rumbaugh, J. (2006). *El Proceso Unificado de Desarrollo de Software*. La Habana: Editorial Félix Varela.

Lago, R. (Abril de 2007). *Patrones de diseño software*. Recuperado el 25 de febrero del 2013, de Patrones de diseño software: <http://www.proactiva-calidad.com/java/patrones/mvc.html>

Microsoft SQL Server. (2004). *Libros en pantalla de SQL Server* (8va ed.). Microsoft Corporation.

Muñoz, S. (31 de Enero de 2006). *Tecnologías web/ PHP*. Recuperado el 26 de febrero del 2013, de Tecnologías web/ PHP:

<http://es.selfhtml.org/introduccion/tecnologias/php.htm>

Navarro, M. S. (2007). *TEMAS DE INFORMÁTICA BÁSICA*. Ciudad de La Habana: Pueblo y Educación.

Núñez Camallea, N. L., & Coutin Abalo, R. (2005a). Aplicaciones con tres capas. *Diccionario de informática*. Cuba: Editorial Científico-Técnica.

Núñez Camallea, N. L., & Coutin Abalo, R. (2005b). CGI. *Diccionario de informática*. Cuba: Editorial Científico-Técnica.

O'Donahue, J. (2002). *Java Database Programming Bible*. USA: John Wiley & Sons.

Pérez Valdés, D. (7 de noviembre de 2007). *Los diferentes lenguajes de programación para la web*. Recuperado el 28 de marzo del 2013, de <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/> **Bibliografía 73**

Díaz Antón, M. G. (s.f.). Propuesta de una metodología de desarrollo de software educativo bajo un enfoque de calidad sistemática. Recuperado el 28 de marzo del 2013, a partir de <http://www.academia-interactiva.com/ise.pdf>

Febles, R. (2005). Información y tecnologías pilares gestión y conocimiento. Recuperado el 28 de marzo del 2013 a partir de www.arenotech.org/.../informacion_tecnologia.htm

Fermín Gómez, W. (s.f.). Para Aprendemas.com. Recuperado a partir de http://www.aprendemas.com/Noticias/html/N358_F20012004.HTML

R., V. T. (10 de 2008). *APLICACIÓN DEL LENGUAJE UNIFICADO DE MODELADO (UML)*. Recuperado el 30 de marzo del 2013, de APLICACIÓN DEL LENGUAJE UNIFICADO DE MODELADO (UML) <http://www.roso-control.com/Espanol/EDU/MICRO/.../Monografia.pdf>

Hernández Quevedo, M. L., & Pérez Suzarte, J. C. (s. f.). La educación de posgrado explorando una discusión CTS. Recuperado el 2 de abril del 2013, a partir de www.monografias.com

Menacho Chiok, L. A. (s. f.). Historia de la educación superior y de postgrado. Recuperado el 2 de abril del 2013, a partir de www.GestioPolis.com

Méndez Cáceres. (2005). Generador de Patrones de Navegación Aplicando Web Log Mining en Cliente/Servidor. Recuperado febrero 7 de abril del 2013, a partir de www.tamps.cinvestav.mx/~vjsosa/thesis.html

Santos, V. R. (2008, de 3 de). Curso JavaScript. Recuperado 7 de abril del 2013, a partir de http://geneura.ugr.es/~victor/cursillos/javascript/js_intro.html

Verano, Catoní, M. L. (2009, enero 25). Software y Educación. Recuperado 7 de abril del 2013, a partir de <http://blogs.rimed.cu/infoedu/category/software-educativo/>

Acosta, H. Á. (2005). *Desarrollo de una Intranet para un Departamento Docente de un Centro de Educación Superior (CES)*. Ciudad de la Habana: Trabajo de Diploma para optar por el título de Ingeniero Informático, Superior Politécnico “José Antonio Echeverría”.

Pelaez, J. C. (25 de Mayo de 2009). *Blog de Juan Peláez en Geeks.ms*. Recuperado el 10 de abril del 2013, de Blog de Juan Peláez en Geeks.ms: <http://www.juanpelaez.com/>

R., V. T. (Octubre de 2008). *APLICACIÓN DEL LENGUAJE UNIFICADO DE MODELADO (UML)*. Recuperado el 10 de abril del 2013, de APLICACIÓN DEL

LENGUAJE UNIFICADO DE MODELADO (UML): www.ros-control.com/Espanol/EDU/MICRO/.../Monografia.pdf

Garzón, D. J. (6 de Junio de 2008). *Ingeniería de Software II*. Recuperado el 13 de abril del 2013, de Ingeniería de Software II: www.codeticainge.googlepages.com/guiaing.pdf

Ruiz, M. H. (2006). *Programación WEB avanzada Soluciones rápidas y efectivas para desarrolladores de sitios*. La Habana: Felix Varela.

Cuenca, C. L. (2006). *Arquitectura del servidor Apache*. Recuperado el 13 de abril del 2013, de <http://www.desarrolloweb.com/articulos/1112.php>

Gutiérrez, J. J. (2006). *¿Qué es un framework web?* Recuperado el 15 de abril del 2013, de <http://jordisan.net/blog/2006/que-es-un-framework/> ¿Qué es un framework?

Hurtado Jara, O. (2008). *Sistemas distribuidos*. Recuperado el 16 de abril del 2013, de Monografías: <http://www.monografias.com/trabajos16/sistemas-distribuidos/sistemas-distribuidos.shtml>

Johnson, A. (2007). *MySQL*. Recuperado el 16 de abril del 2013, de Software Libre: http://softwarelibre.net63.net/?page_id=69

Johnson, A. (2007). *PostgreSQL*. Recuperado el 16 de abril del 2013, de Software Libre: http://softwarelibre.net63.net/?page_id=69

Molina, M., & Castro, A. (2009). *Cliente - servidor*. Recuperado el 16 de abril del 2013, de <http://deymar-clienteservidor.blogspot.com/2009/04/si-bien-la-clasica-arquitectura-cs.html>

Martínez, P. (2007). *Patrones*. Recuperado el 18 de abril del 2013, de Patrón "Modelo-Vista-Controlador": <http://www.proactiva-calidad.com/java/patrones/mvc.html>

Catalani, E. (2007). *ARQUITECTURA Modelo/Vista/Controlador* . Recuperado el 18 de abril del 2013, de <http://exequielc.wordpress.com/2007/08/20/arquitectura-modelovistacontrolador/>

Chappell, D., & Kirk, S. (2006). *De N-capas a .NET. Desarrollo de aplicaciones*. Recuperado el 18 de abril del 2013, de <http://www.microsoft.com/spanish/msdn/articulos/archivo/081102/voices/dncapas.asp>

Pérez Sandoval, H. (2009). *Programación por capas*. Recuperado el 18 de abril del 2013, de MTY.Coders. Comunidad de Desarrolladores de Monterrey: <http://mtycoders.com/programacion-por-capas/>

Sandoval Mendez, S. (2009). *Gestores de Bases de Datos*. Recuperado el 18 de abril del 2013, de <http://www.chacharaselnido.com/ITVG/GBD%20EXPO.pptx>

Una introducción a Apache. (2006). *Una introducción a Apache*. Recuperado el 18 de abril del 2013, de http://linux.ciberaula.com/articulo/linux_apache_intro/

Machado, O. J. (2006). *Portal de Psicología de la Universidad de Cienfuegos*

Wesley, A. (2005). *Introducción a los Sistemas de Bases de Datos*.

Cadenas, Y. (2010). *Aplicación Web para la gestión de la información en las Oficinas de Seguridad para las Redes Informáticas*. Sancti-Spíritus. Sancti-Spíritus, Cuba.

Vedora Willock, V. (2006). *Sistema de gestión de información estudiantes becarios extranjeros*. Cienfuegos, Cuba.

GLOSARIO DE TÉRMINOS

Administrador: es la persona que tiene privilegios para determinadas funcionalidades del sistema.

APACHE: es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras, que implementa el protocolo HTTP/1.1.

Arquitectura Cliente/Servidor: es un modelo para el desarrollo de sistemas de información, en el que las transacciones se dividen en elementos independientes que cooperan entre sí para intercambiar información, servicios o recursos.

CASE: Computer Aided Software Engineering.

DMHU: Departamento de Marxismo e Historia de la Uniss

HTML: HyperText Markup Language. Lenguaje usado para escribir documentos para servidores World Wide Web. Es una aplicación de la ISO Standard 8879:1986. Es un lenguaje de marcas. Los lenguajes de marcas no son equivalentes a los lenguajes de programación aunque se definan igualmente como "lenguajes". Son sistemas complejos de descripción de información, normalmente documentos, que se pueden controlar desde cualquier editor ASCII.

HTTP: HyperText Transfer Protocol. Protocolo de Transferencia de Hipertextos. Modo de comunicación para solicitar páginas web.

Hardware: Componentes electrónicos, tarjetas, periféricos y equipo que conforman un sistema de computación; se distinguen de los programas (software) porque son tangibles.

Internet: Sistema de redes de computación ligadas entre sí, con alcance mundial, que facilita servicios de comunicación de datos como registro remoto, transferencia de archivos, correo electrónico y grupos de noticias. Internet es una forma de conectar las redes de computación existentes que amplía en gran medida el alcance de cada sistema participante.

MySQL: Es un sistema de gestión de bases de datos relacional que cuentan con todas las características de un motor de BD comercial: transacciones atómicas, triggers, replicación, llaves foráneas entre otras. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar.

PHP: Hypertext Preprocessor. Es un ambiente script del lado del servidor que permite crear y ejecutar aplicaciones web dinámicas e interactivas. Con PHP se pueden combinar páginas HTML y scripts. Con el objetivo de crear aplicaciones potentes.

Perl: Practical Extraction and Report Language. Es un lenguaje de programación desarrollado por Larry Wall inspirado en otras herramientas de UNIX como son: sed, grep, awk, c-shell.

RUP: Rational Unified Process (Proceso Unificado de desarrollo). Metodología para el desarrollo de Software.

Software: Programas de sistema, utilerías o aplicaciones expresados en un lenguaje de máquina.

SQL: Structured Query Language. Es un lenguaje declarativo de acceso a bases de datos que permite especificar diversos tipos de operaciones sobre las mismas. Aúna características del álgebra y el cálculo relacional permitiendo lanzar consultas con el fin de recuperar información de interés de una base de datos.

Sitio Web: Es un conjunto de páginas web, típicamente comunes a un dominio de Internet o subdominio en la World Wide Web en Internet.

SGBD: Sistema de Gestión de Bases de Datos. Es el software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez.

UML: Unified Modeling Language. Es una notación estándar para modelar objetos del mundo real como primer paso en el desarrollo de programas orientados a objetos. Es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema de software.

WEB (WWW): Red de documentos HTML intercomunicados y distribuidos entre servidores del mundo entero. XML: eXtensible Markup Language, lenguaje de marcado ampliable o extensible.

ANEXOS:

Anexo 1: Descripción de los Casos de Uso del Sistema. Paquete Seguridad

Caso de Uso:1		Autenticarse
Actores:		Usuario, Administrador.
Propósito:		Iniciar la sesión
Resumen:		El caso de uso inicia cuando el actor introduce su nombre de actor y su contraseña y culmina cuando el sistema lo verifica permitiéndole la entrada al mismo según los permisos que tenga asignados.
Referencia	RF-R1	
Prototipo		Ver Anexo 5.1

Tabla 5 Descripción del caso de uso de sistema Autenticarse.

Caso de Uso:2	Registrarse.
----------------------	---------------------

Actores:	Usuario.
Propósito:	Permite al usuario crear una cuenta en el sistema.
Resumen:	El caso de uso inicia cuando el usuario decide registrarse en el sistema introduciendo los datos requeridos y culmina cuando el sistema comprueba que no hay errores de entrada de datos y entra a la nueva sesión creada.
Referencia	RF-R2
Prototipo	Ver Anexo 5.2

Tabla 6 Descripción del caso de uso del sistema Registrarse.

Caso de Uso:3	Cambiar Contraseña.
Actores:	Usuario, Administrador
Propósito:	Permite al usuario o al administrador cambiar su contraseña de acceso al sistema.

Resumen:	<p>El caso de uso se inicia cuando el usuario o el administrador luego de haber iniciado sesión, ya descrito necesita cambiar contraseña de acceso al sistema, el sistema muestra formulario donde el usuario escribe la nueva contraseña y la confirma, el sistema actualiza la base datos. Culmina el caso de uso.</p>
Referencia	RF-R3
Prototipo	Ver Anexo 5.3

Tabla 6 Descripción del caso de uso del sistema Cambiar Contraseña.

Caso de Uso:4	Cerrar Sesión.
Actores:	Usuario, Administrador.
Propósito:	Salir del Sistema

Resumen:		El caso de uso inicia cuando el actor selecciona la opción de cerrar sesión. El caso de uso finaliza cuando el sistema vuelve a la página inicial.
Referencia	RF-R17	
Prototipo		

Tabla 7 Descripción del caso de uso de sistema Cerrar Sesión.

Anexo 2: Descripción de los Casos de Uso del Sistema. Paquete Gestión.

Caso de Uso:5	Gestionar Taller Martiano.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con los Talleres Martianos, la creación de nuevos Talleres Martianos, modificación, eliminación de los mismos.

Resumen:	<p>El Caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a los Talleres Martianos. Para registrar un Taller Martiano, deben registrarse su nombre, descripción, lugar y fecha del taller. Para eliminar se debe seleccionar el estudiante que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>
Referencia	RF-R4
Prototipo	Ver Anexo 5.4

Tabla 8 Descripción del caso de uso de sistema Gestionar Taller Martiano.

Caso de Uso:6	Gestionar Fórum.
Actores:	Administrador.
Propósito:	<p>Permite al administrador el trabajo con los diferentes Fórum, creación de nuevos Fóruns, inserción, modificación y eliminación de los mismos.</p>

Resumen:	<p>El Caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a los Fórum. Para registrar un Fórum, deben insertarse su nombre, descripción, lugar y fecha. Para eliminar se debe seleccionar el Fórum que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>
Referencia	RF-R5
Prototipo	Ver Anexo 5.5

Tabla 9 Descripción del caso de uso de sistema Gestionar Fóruns.

Caso de Uso:7	Gestionar Cambio de Categorías.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con los diferentes Cambios de Categorías creación de nuevos Cambios de Categorías, modificación, eliminación de las mismas.

Resumen:	<p>El Caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a los Cambios de Categorías. Para registrar un Cambio de Categoría, deben registrarse su nombre, descripción, lugar y fecha. Para eliminar se debe seleccionar el Cambio de Categorías que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>
Referencia	RF-R7
Prototipo	Ver Anexo 5.7

Tabla 10 Descripción del caso de uso de sistema Gestionar Cambios de Categorías.

Caso de Uso:8	Gestionar Curso.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con los diferentes Cursos creación de nuevos Cursos, modificación, eliminación de los mismos.

Resumen:	<p>El Caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a los Cursos. Para registrar un Cursos, deben registrarse su nombre, descripción, lugar y fecha. Para eliminar se debe seleccionar el Curso que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>
Referencia	RF-R8
Prototipo	Ver Anexo 5.8

Tabla 11 Descripción del caso de uso de sistema Gestionar Cursos.

Caso de Uso:9	Gestionar Bibliografía.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con las diferentes bibliografías creación de nuevas bibliografías, modificación, eliminación de las mismas.

Resumen:	<p>El Caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a las bibliografías. Para registrar una bibliografía, deben registrarse su título, descripción, autor y si esta en déficit. Para eliminar se debe seleccionar la bibliografía que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>
Referencia	RF-R10
Prototipo	Ver Anexo 5.10

Tabla 12 Descripción del caso de uso de sistema Gestionar Bibliografía.

Caso de Uso:10	Gestionar Postgrado.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con los diferentes postgrados, creación de nuevos postgrados, modificación, eliminación de los mismos.

Resumen:	<p>El caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a los postgrados. Para registrar un postgrado, deben registrarse su nombre, descripción, fecha y horas. Para eliminar se debe seleccionar el postgrado que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>
Referencia	RF-R11
Prototipo	Ver Anexo 5.11

Tabla 12 Descripción del caso de uso de sistema Gestionar Postgrados.

Caso de Uso:11	Gestionar Publicación.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con las diferentes publicaciones creación de nuevas publicaciones, modificación, eliminación de las mismas.

Resumen:	<p>El caso de uso se inicia cuando el administrador entra en el sistema con su contraseña e inserta, o elimina los datos relacionados con las publicaciones, para insertar una publicación es necesario que no exista ninguna con ese título, y para eliminar los datos si es necesario que exista la publicación registrada. El caso de uso finaliza cuando el actor mencionado anteriormente termina la acción que comenzó.</p>
Referencia	RF-R12
Prototipo	Ver Anexo 5.12

Tabla 13 Descripción del caso de uso de sistema Gestionar Publicación.

Caso de Uso:12	Gestionar Departamento.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con el departamento, la creación de un nuevo departamento, modificación, eliminación de del mismo.

Resumen:	El caso de uso se inicia cuando el administrador necesita insertar, modificar o eliminar al departamento, la aplicación muestra al departamento existente y opciones, el administrador selecciona alguna de estas opciones y completa el formulario, la aplicación registra la información. Culmina el caso de uso.
Referencia	RF-R13
Prototipo	Ver Anexo 5.13

Tabla 14 Descripción del caso de uso de sistema Gestionar Departamento.

Caso de Uso:13	Gestionar Asignatura.
Actores:	Administrador.
Propósito:	Que el administrador pueda gestionar la información relacionada con las asignaturas satisfactoriamente.

Resumen:	<p>El caso de uso se inicia cuando el administrador entra en el sistema con su contraseña e inserta, modifica o elimina los datos relacionados con las asignaturas, para insertar una asignatura es necesario que no exista ninguna con ese nombre, y para modificar los datos si es necesario que exista la asignatura registrada, para eliminar la asignatura no puede pertenecer a ningún profesor y para el caso de listar tiene que existir al menos una asignatura. El caso de uso finaliza cuando el actor mencionado anteriormente termina la acción que comenzó.</p>
Referencia	RF-R14
Prototipo	Ver Anexo 5.14

Tabla 15: Descripción del caso de uso de sistema Gestionar Asignatura.

Caso de Uso:14	Gestionar Profesor.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con los diferentes profesores creación de nuevos profesores, modificación, eliminación de los

	mismos.
Resumen:	El Caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a los profesores. Para registrar un profesor, deben registrarse su nombre, categoría docente, grado científico, facultad. Para eliminar se debe seleccionar el profesor que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.
Referencia	RF-R15
Prototipo	Ver Anexo 5.15

Tabla 16 Descripción del caso de uso de sistema Gestionar Profesor.

Caso de Uso:15	Gestionar Otros Eventos.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con el campo de otros eventos, la creación de otros eventos, modificación, eliminación de los mismos.

Resumen:	<p>El Caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a otros eventos. Para registrar otro evento, deben registrarse su nombre, descripción, lugar y fecha. Para eliminar se debe seleccionar el otro evento que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>
Referencia	RF-R6
Prototipo	Ver Anexo 5.6

Tabla 17 Descripción del caso de uso de sistema Gestionar Otros Eventos.

Caso de Uso:16	Gestionar Otras Convocatorias.
Actores:	Administrador.
Propósito:	Permite al administrador el trabajo con el campo de otras convocatorias, la creación de otras convocatorias, modificación, eliminación de los mismos.

<p>Resumen:</p>	<p>El Caso de uso se inicia cuando el administrador accede al sistema para registrar, modificar o eliminar la información correspondiente a otras convocatorias. Para registrar otra convocatoria, deben registrarse su nombre, descripción, lugar y fecha. Para eliminar se debe seleccionar la otra convocatoria que se desea borrar. El caso de uso finaliza cuando se ejecuta una de las operaciones antes descrita.</p>
<p>Referencia</p>	<p>RF-R9</p>
<p>Prototipo</p>	<p>Ver Anexo 5.9</p>

Tabla 18 Descripción del caso de uso de sistema Gestionar Otras Convocatorias.

Anexo 3: Descripción de los Casos de Uso del Sistema. Paquete Reportes.

Caso de Uso:17	Gestionar Reporte listado de Departamentos.
Actores:	Usuario y Administrador.
Propósito:	Que el Usuario y el Administrador del Sistema puedan visualizar el Departamento.
Resumen:	El caso de uso se inicia cuando el visitante o el administrador entran en el sistema seleccionan el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que esté actualizado el Departamento en la base de datos. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.

Referencia	RF-R16	
Prototipo	Ver Anexo 5.16.1	

Tabla 19: Descripción del caso de uso de sistema Gestionar Reporte listado de Departamentos.

Caso de Uso:18	Generar Reporte Listado de Profesores.
Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuál es el claustro del Departamento.

Resumen:	<p>El caso de uso se inicia cuando el visitante o el administrador entran en el sistema, seleccionan el reporte, y mismo es mostrado. Para mostrar el reporte es necesario que esté actualizado el Claustro en la base de datos. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.</p>
Referencia	RF-R16
Prototipo	Ver Anexo 5.16.2

Tabla 20: Descripción del caso de uso de sistema Gestionar Reporte listado de Profesores.

Caso de Uso:19	Generar Reporte Listado de las Publicaciones.
Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuales son

	las Publicaciones del Departamento.
Resumen:	El caso de uso se inicia cuando el administrador entra en el sistema selecciona el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que estén actualizadas las publicaciones en la carpeta. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.
Referencia	RF-R16
Prototipo	Ver Anexo 5.16.3

Tabla 21: Descripción del caso de uso de sistema Gestionar Reporte listado de las Publicaciones.

Caso de Uso:20	Generar Reporte Listado de las Asignaturas.
-----------------------	--

Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuales son las Asignaturas del Departamento.
Resumen:	El caso de uso se inicia cuando el usuario o el administrador entran en el sistema seleccionan el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que estén actualizadas las asignaturas en la base de datos. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.
Referencia	RF-R16
Prototipo	Ver Anexo 5.16.4

Tabla 22: Descripción del caso de uso de sistema Gestionar Reporte listado de las Asignaturas.

Caso de Uso:21	Generar Reporte Listado de los Fóruns
Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuales son los Fóruns que ha dado el Departamento.
Resumen:	El caso de uso se inicia cuando el usuario o el administrador entran en el sistema seleccionan el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que estén actualizados los Fóruns en la base de datos. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.

Referencia	RF-R16
Prototipo	Ver Anexo 5.16.5

Tabla 23: Descripción del caso de uso de sistema Gestionar Reporte listado de los Fóruns.

Caso de Uso:22	Generar Reporte Listado de los Talleres Martianos.
Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuales son los Talleres Martianos que ha dado el Departamento.
Resumen:	El caso de uso se inicia cuando el usuario o el administrador entran en el sistema seleccionan el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que estén actualizados los Talleres Martianos en la base de datos. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.

Referencia	RF-R16	
Prototipo	Ver Anexo 5.16.6	

Tabla 24: Descripción del caso de uso de sistema Gestionar Reporte listado de los Talleres Martianos.

Caso de Uso:23	Generar Reporte Listado de los Cursos.
Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuales son los Curso que ha dado el Departamento.

Resumen:	<p>El caso de uso se inicia cuando el usuario o el administrador entran en el sistema seleccionan el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que estén actualizados los Cursos en la base de datos. Se permite descargar. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.</p>
Referencia	RF-R16
Prototipo	Ver Anexo 5.16.7

Tabla 25: Descripción del caso de uso de sistema Gestionar Reporte listado de los Cursos.

Caso de Uso:24	Generar Reporte Listado de los Cambios de Categorías.
Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuales son

	los Cambios de Categorías que ha dado el Departamento.
Resumen:	El caso de uso se inicia cuando el usuario o el administrador entran en el sistema seleccionan el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que estén actualizados los Cambios de Categorías en la base de datos. Se permite descargar. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.
Referencia	RF-R16
Prototipo	Ver Anexo 5.16.8

Tabla 26: Descripción del caso de uso de sistema Gestionar Reporte listado de los Cambios de Categoría.

Caso de Uso:25	Generar Reporte Listado de los Otros Eventos.
-----------------------	--

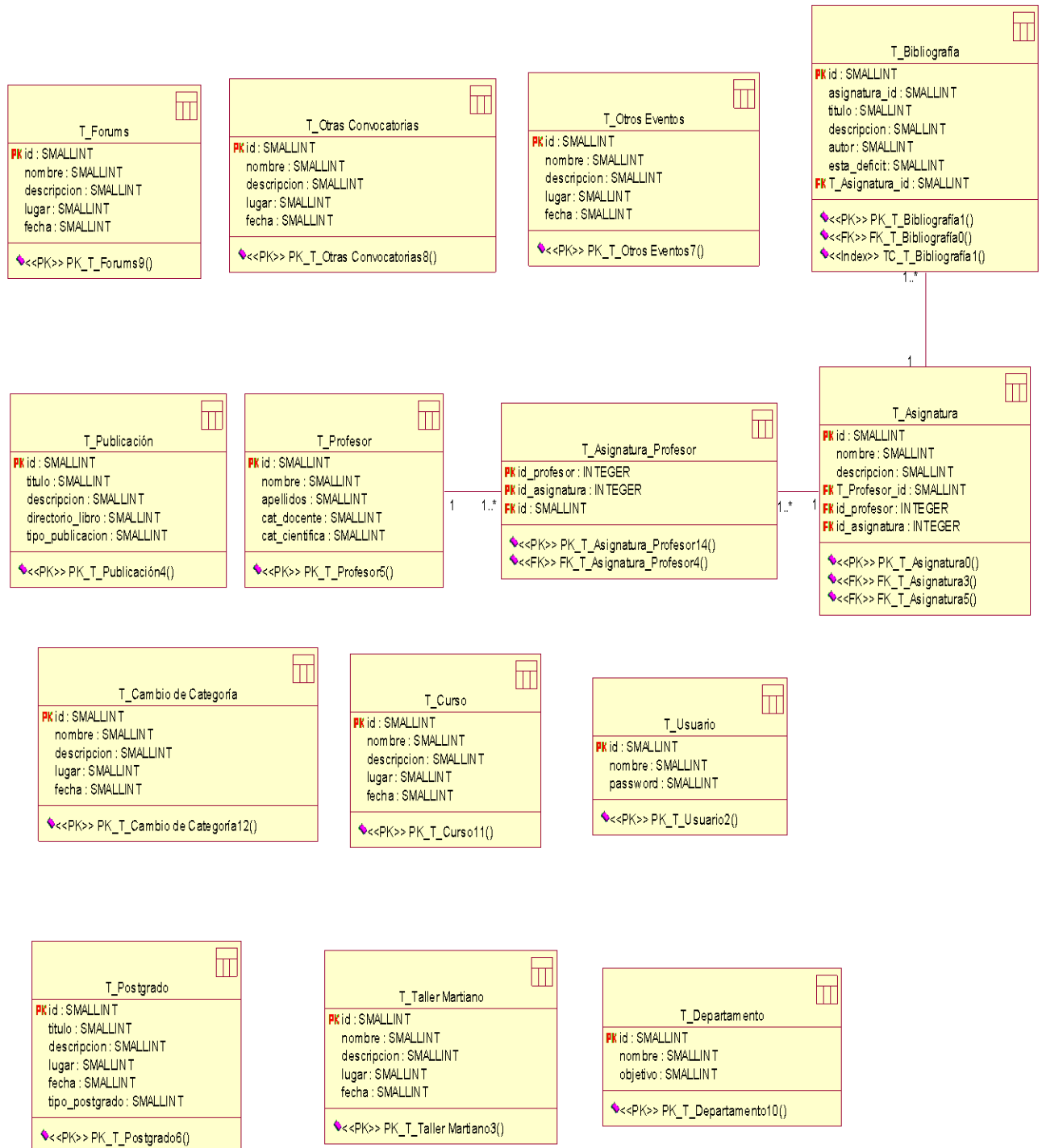
Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuales son los Otros Eventos que ha dado el Departamento.
Resumen:	El caso de uso se inicia cuando el usuario o el administrador entran en el sistema seleccionan el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que estén actualizados los Otros Eventos en la base de datos. Se permite descargar. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.
Referencia	RF-R16
Prototipo	Ver Anexo 5.16.9

Tabla 27: Descripción del caso de uso de sistema Gestionar Reporte listado de los Otros Eventos.

Caso de Uso:25	Generar Reporte Listado de los Otras Convocatorias.
Actores:	Usuario y Administrador.
Propósito:	Que el usuario y el administrador del sistema puedan visualizar cuales son las Otras Convocatorias que ha dado el Departamento.
Resumen:	El caso de uso se inicia cuando el usuario o el administrador entran en el sistema seleccionan el reporte, y el mismo es mostrado. Para mostrar el reporte es necesario que estén actualizados los Otras Convocatorias en la base de datos. El caso de uso finaliza cuando los actores mencionados anteriormente terminan la acción que comenzaron.
Referencia	RF-R16
Prototipo	Ver Anexo 5.16.10

Tabla 28: Descripción del caso de uso de sistema Gestionar Reporte listado de las Otras Convocatorias.

Anexo 4: Diagrama clases del modelo físico de Datos.



Anexo 5: Prototipos

Ventana de Inicio



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingenieria
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores



¿Qué significa hoy ser de izquierda?

Empecemos por el asunto más espinoso. El pasado mes de abril, la condena a muerte de tres secuestradores y las elevadas penas de cárcel impuestas a "disidentes" acusados de operar a las órdenes o en favor de la Oficina de Negocios de EEUU en Cuba, fue utilizada -como era de esperar- para justificar o aumentar las presiones imperialistas sobre la isla

[Read more >](#)



Vigencia del pensamiento de Karl Marx

Quiero iniciar esta disertación planteando las siguientes preguntas: ¿Por qué hablar de Karl Marx, pensador y revolucionario del siglo XIX en el siglo XXI? ¿Se debe insistir en el estudio y vigencia de su obra? ¿Se puede argumentar a su favor si los países del llamado campo socialista en Europa del Este se fueron a pique y se constituyeron en sociedades capitalistas?.

[Read more >](#)



Fragmento de El príncipe

El príncipe, de Nicolás Maquiavelo, es uno de los más influyentes tratados de ciencia política, publicado en 1532. El fragmento siguiente reproduce su capítulo XV, donde el autor italiano enuncia los comportamientos que debe seguir un gobernante, siempre conducentes al mantenimiento del poder sobre sus territorios.

[Read more >](#)

INFORMACIÓN

- › Portafolio
- › Visión y Misión
- › Acerca de nosotros
- ›

SERVICIOS

- › Asignaturas
- › Publicaciones
- › Eventos
- › Convocatorias
- › Postgrados
- › Nuestros Profesores
- › Bibliografías
- › Departamento
- › Atrás

SESIÓN

- administrador
- › Cerrar Sesión
- › Cambiar contraseña

En e Footer de arriba se ve que alguien está logueado, ahora veremos cuando todavía no se ha logueado nadie.

INFORMACIÓN	SERVICIOS	SESIÓN
› Portafolio	› Asignaturas	› Iniciar Sesión
› Visión y Misión	› Publicaciones	› Registrate
› Acerca de nosotros	› Eventos	
› 	› Convocatorias	
	› Postgrados	
	› Nuestros Profesores	
	› Bibliografías	
	› Departamento	
	› Atrás	

Anexo 5.1: Prototipo Autenticación.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Usuario:

administrador

Contraseña:

...



Estimado Usuario

Si usted ha accedido a este sitio, es necesario que se cree una cuenta de usuario, si es que no tiene una, o puede seguir como usuario anónimo.

Anexo 5.2: Prototipo Registrarse.

[Login](#)

Username:

Email:

Password:

Verification:

Anexo 5.3: Prototipo Cambiar Contraseña.

Logged in as administrador | [Logout](#)

Current password:

New password:

Verification:

Ventana de Eventos



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros Ingenieros

Facultad de
Contabilidad
Futuros Contadores

 **TALLER**

Aquí se encuentran los diferentes Talleres Martianos que darán los profesores de nuestro Departamento.



 **FÓRUMS**

Aquí se encuentran los diferentes Fórum de Historia de la Feu en que participarán nuestros docentes.



 **OTROS**

Aquí se encuentran otros tipos de Eventos dentro del ámbito universitario



Anexo 5.4: Prototipo Gestionar Taller Martiano.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Taller martiano

Nombre:

Descripción:

Lugar:

Fecha

2013 Oct

12

Editar

[Regresar](#)

Anexo 5.5: Prototipo Gestionar Fóruns de Historia de la FEU.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Forumfeu

Nombre:

Descripción:

Lugar:

Fecha

2013 May

11

Editar

[Regresar](#)

Anexo 5.6: Prototipo Gestionar Fóruns Otros Eventos.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Otros eventos

Nombre:

Descripción:

Lugar:

Fecha

2013 Sep

1

Editar

[Regresar](#)

Ventana de Convocatorias



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

CURSOS

Aqui se encuentran los diferentes Cursos que darán los profesores de nuestro Departamento.

CAMBIOS

Aqui se encuentran los diferentes Cambios de Categorías que darán nuestros profesores.

OTRAS

Aqui se encuentran otros tipos de Convocatorias dentro del ámbito universitario

Anexo 5.7: Prototipo Gestionar Cambio de Categoría.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Cambiocategoria

Nombre: <input type="text" value="Titulares"/>	Descripción: <input type="text" value="dfsdf h lllhuf thuf uyhfosa sd dthsoh"/>
Lugar: <input type="text" value="Docente 1"/>	Fecha <input type="text" value="2014"/> <input type="text" value="Oct"/> <input type="text" value="15"/>

[Regresar](#)

Anexo 5.8: Prototipo Gestionar Cursos.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Curso

Nombre: <input type="text" value="Curso de Superacion"/>	Descripción: <input type="text" value="dhfgsdf dsf shf wyh thuh sdl"/>
Lugar: <input type="text" value="Las Villas"/>	Fecha <input type="text" value="2013"/> <input type="text" value="May"/> <input type="text" value="20"/>

[Regresar](#)

Anexo 5.9: Prototipo Gestionar Otras Convocatorias.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Otras convocatorias

Nombre:

Activida del departamento

Descripción:

cvbhk vh fhsadlf fsdhsahf fdhas

Lugar:

Docente 1

Fecha

2013 Jul

7

Editar

• [Regresar](#)

Anexo 5.10: Prototipo Gestionar Bibliografía.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Bibliografía

Título:

El Capital

Descripción:

dsfns fhbsld fh fiutr ty

Autor:

marx

Esta deficit

Asignatura

Editar

• [Regresar](#)

Ventana de Postgrados



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores



DE LA CIENCIA

Aqui se encuentran los Problemas de la Ciencia y la Tecnologia que se imparten a los profesores de nuestro Departamento.



CURSOS

Aqui se encuentran los Cursos de Superacion Politico Ideologica que se imparten a los profesores de nuestro Departamento.



OTROS

Aqui se encuentran otros Postgrados que le pueden ser de interes

Anexo 5.11.1: Prototipo Gestionar Problemas de la Ciencia y la Tecnología.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Problemas Sociales de la Ciencia y la Tecnologia

Titulo	Descripcion	Lugar	Tipo_postgrado	Fecha	Editar	Eliminar
Tecnologia y Sociedad	dsfs sdfh sadhf sadfhsa	Docente 2	problemas	2013-05-08		
Problemas de Medio Ambiente	df hf s dfhasdfh fashdfaisdhfa ijdfd	Docente 1	problemas	2013-05-01		

- [Crear un nuevo Problema](#)

Anexo 5.11.2: Prototipo Gestionar Cursos de Superación Política Ideológica.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Curso de Superacion Politico Ideologica

Titulo	Descripcion	Lugar	Tipo_postgrado	Fecha	Editar	Eliminar
Curso de Marxismo	dfsdnt fyhf aseuy afuf fehfa sadfu	Las Villas	cursos	2013-09-01		
Curso Politico Ideologico	dfhdshs adlf sdfhaslf sadfhlas	Biblioteca	cursos	2008-01-01		

- [Crear un nuevo Curso](#)

Anexo 5.11.3: Prototipo Gestionar Otros Postgrados.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar postgrado

Título:

Descripción:

Lugar:

Fecha:

Tipo de Postgrado:

[Regresar](#)

Ventana de Publicación



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

LIBRO

Aquí se encuentran los diferentes Libros que han publicado los profesores de nuestro Departamento.

ARTÍCULO CTFCO

Aquí se encuentran los diferentes Artículos Científicos que han publicado los profesores de nuestro Departamento.

OTROS

Aquí se encuentran otros tipos de publicaciones que han publicado los profesores de nuestro Departamento.ej: revistas

Anexo 5.12: Prototipo Gestionar Publicación.

Anexo 5.12.1: Prototipo Gestionar Libro.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar libro

Título:

Descripción:

Directorio libro:

[Regresar](#)

Anexo 5.12.2: Prototipo Gestionar Artículo Científico.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar articulo

Titulo
De la Ciencia

Descripcion
dshfhjisd sdfjs
sdfjas;n fo;ijf;s idjf;a

Directorio libro
Ciencia y Tecnología

Editar

• [Regresar](#)

Anexo 5.12.3: Prototipo Gestionar Otras Publicaciones.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar publicacion

Titulo:
Revista del Departamento

Descripción:
dfisda fdlids dsfhla
dhf sdh

De cual libro:
de ninguno

Tipo de Publicación: revista

Editar

• [Regresar](#)

Anexo 5.13: Prototipo Gestionar Departamento.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Departamento

Nombre:
Departamento de Marxismo

Objetivo:
dasdfh asdfhsal dthhf fhasdf
fsadthsadf sdf sfjjsfjasj fsadffjasñ
fdja asdfñjañsdfj sadffjas

Editar

• [Regresar](#)

Anexo 5.14: Prototipo Gestionar Asignatura.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Asignatura

Asignatura:
Filosofia

Descripción:
dfugugfs hdfhdf blduhfiauhf fdhslfh
nmf,hdf f,sdjf dth,odfj ,djf;sdj

Profesores:
Frank
Edelso

Editar

• [Regresar](#)

Anexo 5.15: Prototipo Gestionar Profesor.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Editar Profesor

Nombre:

Apellidos:

Categoría Docente:

Categoría Científica:

Asignaturas

[Regresar](#)

Anexo 5.16.1: Prototipo Reportes Listar Departamento.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Departamento

Nombre	Objetivo	Editar	Eliminar
Departamento de Marxismo e Historia de la Uniss	dasdfh asdfhsal dhhf fhadsf fsadthsadf sdf sfdfjsfaj fsadfasñ fdja asdfñafsdfj sadfjas		

- [Crear](#)



Anexo 5.16.2: Prototipo Reportes Listar Profesores.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Profesores

Nombre	Apellidos	Categoria_docente	Categoria_cientifica	Editar	Eliminar
Frank	Quesada	Profesor	master		
Edelso	Hernandez	titular	licenciado		

- [Crear un nuevo Profesor](#)

Anexo 5.16.3: Prototipo Reportes Listar Publicaciones. Anexo 5.16.3.1: Prototipo Reportes Listar Libros.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Libros

Título	Descripción	Directorio del Libro	Tipo de Publicación	Editar	Eliminar
Aprendiendo de Martí	dflhsjldkfvnxcn dsfj;sd dijfo;sd dijf;sod djf,	Historias de Martí	libro		
Jose Varona	sjfh wkf.m jh wkje sdji wej; djf.s jf,	Vida y Obra de Varona	libro		

- [Crear un nuevo libro](#)

Anexo 5.16.3.2: Prototipo Reportes Listar Artículos Científicos.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Articulos Cientificos

Título	Descripción	Directorio del Libro	Tipo de Publicación	Editar	Eliminar
De la Ciencia	dslhfvjlsd sdfjs sdfjas;n fo;ijf;s idjf,a	Ciencia y Tecnologia	articulo		
Historia de la Uniss	dsfhsd ds.fn fsdfhsdhf dfijs;f dhfs;dhf djf,oa	Historia de SS	articulo		

- [Crear un nuevo Artículo](#)

Anexo 5.16.3.2: Prototipo Reportes Listar Otras Publicaciones.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Otras publicaciones

Titulo	Descripcion	Tipo de Publicación	Directorio del Libro	Editar	Eliminar
Revista del Departamento	dfdsda fdlfds dsfhla dhf sdh	de ninguno	revista		
Documento de la FEU	dfhu;h dfhn sfh dfjsd; sjfj	Historia de la FEU	Documento		

- [Crear una nueva publicación](#)

Anexo 5.16.4: Prototipo Reportes Listar Asignatura.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Asignaturas

- [Filosofía](#)
- [Ética](#)
- [Estética](#)
- [Economía política](#)



Los Profesores de nuestro Departamento imparten las asignaturas que usted puede ver, brinda su servicio a todas las carreras de la Universidad, con un total de 30 profesores que atienden a alrededor de 1500 docentes tanto en las FUM (Filial Universitaria Municipal) como en la Sede Central.

Donde asesoran la materialización del trabajo político ideológico y la docencia vinculada o asociada a la enseñanza del marxismo y la historia de Cuba.

Con la aparición de la Sede Universitaria el colectivo, integrado por instructores y asistentes, se independiza y comienza un proceso de despegue en su actividad que posibilitó su crecimiento desde el punto de vista académica científico e investigativo a través del impulso a la formación doctoral y el perfeccionamiento de la pirámides categorial, lo que posibilitó inscribir 3 de sus profesores en doctorados curriculares y su conclusión en 2004, la ascensión de categorías docentes principales hasta alcanzar la

Lista de Asignaturas

Nombre	Descripción	Editar	Eliminar
Filosofía	dfugugfs hdfndf blduhfliauhf fdhslfh nmf,hdf f,sdjf dfh;o,dj ;djfl,sdj		
Ética	dfhsf fjds sdfdsf skdfh; sdhf dfhl fsfdnlwuyw fdhl		
Estética	dhljsh djhdh dij;oij dio;os		
Economía política	dhsfk dskhls sdjs; sdijd;d		

[Crear una nueva Asignatura](#)

Anexo 5.16.5: Prototipo Reportes Listar Fóruns.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Lista de Fóruns

Nombre	Descripcion	Lugar	Fecha	Editar	Eliminar
Carlos Baliño	dshguf d,fnsa fudhbf sdhf dsfh	Residencia Estudiantil	2013-05-11		
Antonio Guiteras	sdg dfbsad fs,fs fuhqwlue wuh	Docente 2	2013-06-01		

- [Crear un nuevo Fórum](#)

Anexo 5.16.6: Prototipo Reportes Listar Talleres Marianos.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Lista de Talleres Marianos

Nombre	Descripcion	Lugar	Fecha	Editar	Eliminar
Vidas de nuestros Heroes	ldushflh dsfh dfi; df;	Docente2	2013-02-01		
Nuestro Apostol	d;fhsd dfh fs dfh dfjsd; dfj ngf	Docente1	2013-10-12		

- [Crear un nuevo Taller](#)

Anexo 5.16.7: Prototipo Reportes Listar Cursos.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Lista de Cursos

Nombre	Descripcion	Lugar	Fecha	Editar	Eliminar
Curso de Superacion	dhfgsdf dsf shf wyh fthuh sdl	Las Villas	2013-05-20		
Curso de maestros	dsfds ds fh fwqew wjhefqwhf	Docente 2	2013-03-01		

- [Crear un nuevo Curso](#)

Anexo 5.16.8: Prototipo Reportes Listar Cambio Categoría.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Cambios de categorías

Nombre	Descripcion	Lugar	Fecha	Editar	Eliminar
Par auxiliar	sdfhu fsd fsd uf ufsdfsñ sdfusdf fjsa djf	Docente 1	2013-05-10		
Titulares	dfsdf fh lfhhuf fhuf uyhfosa sd dhsoh	Docente 1	2014-10-15		

- [Crear un nuevo Cambio de Categoría](#)

Anexo 5.16.9: Prototipo Reportes Listar Otros Eventos.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Otros eventos

Nombre	Descripcion	Lugar	Fecha	Editar	Eliminar
Forum de la Patria	fds fv rtgergrt ghrt g	Docente 1	2013-07-01		
Forum de Departamento	sdfgsad dyf wey weyh ey	Docente 1	2013-09-01		

- [Crear un nuevo Evento](#)

Anexo 5.16.10: Prototipo Reportes Listar Otras Convocatorias.



Home
pagina de inicio

Acerca
de nuestra
universidad

Facultad de
Humanidades
Con dos Carreras

Facultad de
Ingeniería
futuros ingenieros

Facultad de
Contabilidad
Futuros Contadores

Otras convocatorias

Nombre	Descripcion	Lugar	Fecha	Editar	Eliminar
Activida del departamento	cvbhk vh fhasdlf fs dhfsahf fdhas	Docente 1	2013-07-07		
Evento de Departament	fgdf g dsggd g dfgdsfgd g dsgd gd	Docente1	2013-04-01		

- [Crear una nueva Convocatoria](#)

