

**ADMINISTRACIÓN, ANÁLISIS DISEÑO, DESARROLLO E IMPLEMENTACIÓN
DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE
INGENIERÍA INDUSTRIAL, INGENIERÍA DE PETRÓLEOS Y DISEÑO
INDUSTRIAL**

**HERNAN DARIO CARREÑO LAGUADO
JORGE ANDRES LOPEZ FIGUEROA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA**

2013

**ADMINISTRACIÓN, ANÁLISIS DISEÑO, DESARROLLO E IMPLEMENTACIÓN
DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE
INGENIERÍA INDUSTRIAL, INGENIERÍA DE PETRÓLEOS Y DISEÑO
INDUSTRIAL**

**HERNAN DARIO CARREÑO LAGUADO
JORGE ANDRES LOPEZ FIGUEROA**

Trabajo de grado para optar el título de Ingeniero de Sistemas

**Director
Msc. LUIS IGNACIO GONZÁLEZ RAMÍREZ
Magíster en Informática**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2013**

DEDICATORIA

A nuestros padres,

Por darnos la vida, su cariño diario, comprensión, confianza y amor.

A nuestras familias por todo su apoyo incondicional y sabios consejos.

A nuestros amigos, por sus voces de aliento cuando las necesitamos.

También dedicamos este logro a nuestros compañeros de Calumet,

Por su amistad, colaboración y gran ambiente de trabajo.

Hernan y Jorge

AGRADECIMIENTOS

*Agradecemos nuestros padres que siempre nos han brindado su apoyo
y confianza en todos nuestros proyectos,
por la educacion que nos han dado
formándonos en las personas que somos hoy en día.*

*A todos nuestros amigos y familiares que nos han ayudado de una u
otra forma en las diversas pruebas que nos ha puesto la vida.
Y en general a todas aquellas personas que nos han colaborado de cualquier
manera, les ofrecemos nuestros más sinceros agradecimientos.*

Hernan y Jorge.

CONTENIDO

INTRODUCCIÓN	22
1 PRESENTACIÓN DEL PROYECTO	24
1.1 ORIENTACIÓN SOBRE EL CONTENIDO DEL INFORME	24
1.2 ANTECEDENTES.....	25
1.3 DEFINICIÓN DEL PROBLEMA	26
1.4 OBJETIVOS.....	27
1.4.1 Objetivo General	27
1.4.2 Objetivos Específicos	27
1.5 JUSTIFICACIÓN	31
1.6 ALCANCES Y LIMITACIONES.....	32
2 MARCO TEÓRICO	34
2.1 ARQUITECTURA CLIENTE/SERVIDOR	34
2.1.1 Características de la arquitectura Cliente/Servidor.....	35
2.1.2 Clasificación de las arquitecturas Cliente/Servidor.	36
2.1.3 Arquitectura Cliente/Servidor aplicada.....	39
2.1.4 Ventajas del esquema Cliente/Servidor	39
2.1.5 Desventajas del esquema Cliente/Servidor	40
2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS.....	40
2.2.1 Código del Lado del Cliente (Client Side Scripts).	41
2.2.2 Código del Lado del Servidor (Server Side Scripts).	41
2.2.3 Tecnología aplicada.	42
2.3 BASES DE DATOS.....	44
2.3.1 Modelos de Bases De Datos.....	45
2.3.2 Acceso a Base de Datos.	46
2.3.3 Conectores más utilizados.	46
2.3.4 Manejadores o Gestores de Bases de Datos.	48
2.3.5 MySQL.....	49
2.3.6 Ventajas de MySQL.	50
2.4 NETBEANS.....	50
2.5 SISTEMA DE CONTROL DE VERSIONES	51
2.5.1 Subversión.	52
2.6 PROGRAMACIÓN UTILIZADA	52
2.6.1 Clases.....	53
2.6.2 Objetos.	53

2.6.3	Atributos.	53
2.6.4	Métodos.	53
2.6.5	Herencia.	53
2.6.6	Beneficios de la Programación Orientada a Objetos.	54
2.6.7	Java y JDK (Java Development Kit).....	54
2.7	SERVIDORES WEB	55
2.7.1	Servidor Jakarta Tomcat	55
3	MARCO METODOLÓGICO	56
3.1	PROTOTIPO EVOLUTIVO	56
3.2	LENGUAJE DE MODELADO UNIFICADO	58
3.2.1	Diagramas de UML.	59
3.2.2	Diagramas de casos de uso.....	59
3.2.3	Diagramas de secuencias.	61
3.3	ESTÁNDARES DE PROGRAMACIÓN	63
3.3.1	Modelo de datos.....	63
3.3.2	Nombres de las tablas.....	63
3.3.3	Clases.....	64
3.3.4	Páginas JSP.....	64
3.3.5	Organización de Directorios.....	65
4	DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO	66
4.1	PROTOTIPO ESPERADO	66
4.1.1	Diagramas de Casos de Uso.	74
4.1.2	Documentación de Casos de Uso del Sistema.....	78
4.1.3	Diseño y Análisis.	83
4.1.4	Modelo de Procesos del Sistema.....	94
4.1.5	Implementación, Implantación y Pruebas Generales.	101
4.2	MANTENIMIENTO Y ADMINISTRACIÓN	103
4.2.1	Actividades de Mantenimiento.	103
4.2.2	Actividades de Soporte a Usuarios.	112
4.2.3	Actividades de Administración.	112
5	PRUEBAS DEL SISTEMA.....	114
5.1	PRUEBAS DE VERIFICACIÓN	114
5.1.1	Pruebas por componente.	114
5.2	PRUEBAS DE INTEGRACIÓN.....	119
5.3	PRUEBAS DE VALIDACIÓN.....	119

6	CONCLUSIONES.....	120
7	RECOMENDACIONES.....	122
	BIBLIOGRAFÍA	123
	ANEXOS	126

LISTA DE TABLAS

Tabla 1. Casos de uso: Limpiar aula.	78
Tabla 2. Casos de uso: Depurar tablas.	78
Tabla 3. Casos de uso: Ver cartelera.	79
Tabla 4. Casos de Uso: Publicar cartelera.	79
Tabla 5. Caso de Uso: Foros.	80
Tabla 6. Casos de uso: Enviar correos.	81
Tabla 7. Casos de uso: Mantenimiento y Administración de los portales EEIWeb, EIPWeb y EDIWeb.	82
Tabla 8. Descripción de las Entidades.	89
Tabla 9. Pruebas Realizadas: Limpiar aula.	115
Tabla 10. Pruebas realizadas: Depurar tablas.	115
Tabla 11. Pruebas realizadas: Ver cartelera.	116
Tabla 12. Pruebas realizadas: Publicar cartelera.	117
Tabla 13. Pruebas realizadas: Foros.	118
Tabla 14. Pruebas realizadas: Correos.	119

LISTA DE FIGURAS

Figura 1. Modelo Cliente/Servidor	35
Figura 2. Arquitectura Cliente/Servidor de Dos Capas	37
Figura 3. Arquitectura Cliente/Servidor de Tres Capas	38
Figura 4. Modelo de acceso a JSP	48
Figura 5. Función del DBMS.	53
Figura 6. Prototipo Evolutivo.	56
Figura 7. Diagramas de Casos de Uso	60
Figura 8. Diagrama de Secuencias.....	62
Figura 9. Diagrama de Casos de Uso: Limpiar aula de clase.	74
Figura 10. Diagrama de Casos de Uso: Depurar tablas	75
Figura 11. Diagrama de Casos de Uso: Ver cartelera.	75
Figura 12. Diagrama de Casos de Uso: Publicar cartelera.	76
Figura 13. Diagrama de Casos de Uso: Foros.	76
Figura 14. Diagrama de Casos de Uso: Enviar correo.	77
Figura 15. Diagrama de Casos de Uso: Subsistema de mantenimiento y administración.....	77
Figura 16. Diagrama E/R: Limpiar Aula	84
Figura 17. Diagrama E/R: Continuación limpiar aula.....	85
Figura 18. Diagrama E/R: Depurar tablas.....	86
Figura 19. Diagrama E/R: Continuación depurar tablas.....	87
Figura 20. Diagrama E/R: Portal del decano.	88
Figura 21. Diagrama de secuencia: Limpiar aula	94
Figura 22. Diagrama de secuencia: Depurar tablas.	95
Figura 23. Diagrama de secuencia: Ver cartelera.	96
Figura 24. Diagrama de secuencia: Publicar cartelera.....	97
Figura 25. Diagrama de secuencia: Correos.	98
Figura 26. Diagrama de secuencia: Foros.....	100

LISTA DE ANEXOS

ANEXO A. MANUAL DE USUARIO.....	125
---------------------------------	-----

GLOSARIO

ACTOR: En el Lenguaje Unificado de Modelado (UML), un actor detalla un rol que juega un usuario u otro sistema que interactúa con el sujeto es decir el intercambio de signos y datos, no necesariamente un actor representa una persona en particular y además es externo a dicho sujeto.

ARQUITECTURA CLIENTE/SERVIDOR: Modelo para el desarrollo de sistemas de información, de procesamiento cooperativo en el que las transacciones se dividen en procesos independientes para intercambiar información, servicios o recursos. Se denomina Cliente al proceso que inicia el diálogo o solicita recursos, y Servidor al proceso que responde a las solicitudes.

COOKIE: Es un pequeño documento de texto grabado en el disco duro de la computadora del usuario utilizado para mantener el estado de una aplicación o seguir la trayectoria de un usuario dentro del sitio.

ESCALABILIDAD: Propiedad deseada que aumenta la capacidad de clientes y servidores por separado. Cualquier elemento puede ser aumentado o mejorado en algún momento, o añadir nuevos nodos a la red (clientes y/o servidores).

HIPERTEXTO: Es todo aquel documento utilizable en el Internet que contenga vínculos con otros documentos como imágenes, textos, sonidos o videos relacionados entre si por medio de enlaces (Links).

HTML: (*HiperText Markup Language*, Lenguaje de Marcas de Hipertexto). Es un lenguaje de programación utilizado para crear textos y páginas web. Está compuesto por etiquetas que definen la estructura y el formato del documento que verá el usuario en la web. Esas etiquetas son leídas por el navegador y finalmente se ejecutan las funciones creadas en HTML para poder ser vistas.

HTTP: (*HiperText Transfer Protocol*, Protocolo de Transferencia de Hipertexto). Lenguaje empleado para describir el envío de documentos HTML por Internet. HTTP proporciona las normas para que los navegadores hagan peticiones y los servidores entreguen respuestas.

INTERNET: Red global de comunicaciones que interconecta computadores y bases de datos distribuidas por todo el planeta.

IP: (*Internet Protocol*). Protocolo que provee funciones básicas de direccionamiento en Internet y en cualquier red TCP/IP (software de comunicación), se encarga de poner una etiqueta con la dirección adecuada a cada paquete, ya que cada computador conectado a la red tiene una dirección de Internet única que lo distingue de cualquier otro computador en el mundo.

JAVA: Es un lenguaje de programación orientado a objetos desarrollado por SUN Microsystems. Está diseñado para usarse en entorno distribuido de Internet.

JAVASCRIPT: Lenguaje de programación interpretado (no requiere compilación), es un lenguaje basado en objetos, permite el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas.

JDBC: (*Java Database Connectivity*, Conectividad de Base de Datos Java). Es una interfaz de programación de aplicaciones (API), que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java, sin importar el sistema operativo donde se ejecute o de la base de datos a la cual se accede, utilizando el dialecto SQL del modelo de base de datos que se utilice.

JSP: (*Java Server Pages*). Tecnología orientada a crear páginas web con programación Java, con este lenguaje podemos crear aplicaciones web que se

ejecuten en múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las paginas JSP están compuestas de código HTML mezclado con etiquetas especiales para programas scripts de servidor en sintaxis Java.

LINUX: *Sistema operativo.* Es una implementación de libre distribución UNIX para computadores personales, servidores y estaciones de trabajo. Consta de componentes GNU y el kernel desarrollado por Linux Torvalds.

MAINFRAME: Es una maquina grande, poderosa y costosa (Computador central), utilizada principalmente en empresas que necesitan procesar gran cantidad de datos o soportar gran cantidad de usuarios.

Puede funcionar años sin problemas ni interrupciones, incluso puede repararse mientras funciona, además también puede simular el funcionamiento de cientos de computadores personales (terminadores virtuales), dentro de una empresa.

MOTOR DE SERVLETS: administra la carga y descarga del servlet, y trabaja con el servidor Web para dirigir peticiones a los servlets y enviar la respuesta a los clientes.

OVA: (*Objetos Virtuales de Aprendizaje*). Archivos o unidades digitales de información, dispuestos para ser utilizados en diferentes contextos pedagógicos. Estos elementos tienen cierto de interactividad e independencia, que podrán ser utilizados o ensamblados en diferentes situaciones de enseñanza-aprendizaje

PÁGINA WEB: Es un documento que forma parte de un sitio web y suele contar con un hipervínculo o link, para facilitar la navegación entre los contenidos, estas páginas están desarrolladas con lenguajes como HTML y pueden presentar información en diferentes formatos como texto, imágenes, sonidos, videos, etc.

PÁGINA WEB DINÁMICA: Es aquella que permite crear aplicaciones dentro de la propia web, dando una mayor interactividad con el navegante, algunas aplicaciones dinámicas son encuestas y votaciones, foros de soporte, libros de visita, envío de e-mails, reserva de productos, pedidos on-line.

PÁGINA WEB ESTÁTICA: Son aquellos sitios enfocados principalmente a mostrar una información permanente, donde el navegante se limita a obtener dicha información, sin que puedan interactuar con la pagina visitada, este tipo de webs son incapaces de soportar aplicaciones como gestores de bases de datos, foros, consultas online, e-mails.

PORTABLE: la portabilidad de un software se define como su grado de dependencia de la plataforma en la que corre. La portabilidad es mayor cuanto menor es su dependencia del software de plataforma.

PORTAL WEB: Es un sitio web cuya característica fundamental es la de servir de puerta de entrada (única) para ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de recursos y de servicios relacionados a un mismo tema.

SCRIPT: programa escrito en un lenguaje específico de programación que tiene una serie de instrucciones y normalmente funciona sobre otras aplicaciones que ya están en funcionamiento.

SERVLET: son clases Java que amplían la funcionalidad de un servidor Web, mediante la generación dinámica de páginas Web.

SERVIDOR WEB: Servidor que almacena las páginas de un sitio Web y envía páginas web en respuesta a la peticiones HTTP hechas desde los navegadores de los clientes.

UML: (*Unified Modeling Language*, Lenguaje de Modelamiento Unificado). Es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de cosas concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

URL: (*Uniform Resource Locator*, Localizador Uniforme de Recursos). Cadena de caracteres que definen la localización y el acceso a documentos de hipertexto o programas en Internet. Un URL está formado de la siguiente manera: Esquema: //máquina/ruta.

WWW: (*World Wide Web*), Es el sistema de documentos de hipertexto que se encuentran enlazados entre si y que son accesibles a través de internet, mediante un software conocido como navegador además los usuarios pueden visualizar las páginas web con texto, imágenes, videos, y otros archivos de multimedia navegando a través de hipervínculos.

RESUMEN

TÍTULO: ADMINISTRACIÓN, ANÁLISIS DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE INGENIERÍA INDUSTRIAL, INGENIERÍA DE PETRÓLEOS Y DISEÑO INDUSTRIAL ⁱ.

AUTORESⁱⁱ: Hernan Dario Carreño Laguado
Jorge Andrés López Figueroa

PALABRAS CLAVE: Sitio Web, Portal Web, Módulo, Servicio, EEIE (Escuela de Estudios Industriales y Empresariales), EIP (Escuela de Ingeniería de Petróleos), EDI (Diseño Industrial), EIEEWeb, EIPWeb, EDIWeb.

DESCRIPCIÓN

Los portales web de la EEIE, EIP y EDI se han posicionado como el medio de comunicación e información más importante para la comunidad dentro del ámbito académico y administrativo en sus respectivas escuelas, por esta razón las necesidades de la comunidad son un factor importante para la continua mejora e innovación en los procesos y servicios que ofrecen los portales, de esta manera se ha logrado formar un sitio web donde se fomenta la participación activa de sus miembros. Por estos motivos, el grupo de desarrollo de software Calumet tiene como objetivo la implementación de nuevos servicios, así como el mantenimiento y mejora de los mismos.

En el ámbito académico el soporte a la labor docente en el portal se realiza a partir del concepto de aula virtual. Este se entiende como el espacio creado para complementar el proceso de aprendizaje de manera no presencial como una forma que garantiza el cumplimiento de las horas de trabajo independiente (T.I.) por parte del estudiante, se mejoró este espacio permitiéndole a los docentes eliminar archivos, respuestas, foros y demás actividades realizadas por estudiantes del semestre anterior para así poder reutilizar guiones de clase existentes en semestres posteriores.

Se creó un servicio el cual permite depurar las bases de datos del portal eliminando registros y datos que han perdido vigencia de manera rápida y ágil, además se creó el portal web del decano, el cual le dará soporte a la función de la decanatura de la facultad de ingenierías fisicomecánicas, con este portal los administrativos de la facultad, pueden interactuar constantemente con la comunidad universitaria, comunicándose con las escuelas pertenecientes a la facultad de una manera transparente por medio del nuevo portal.

ⁱ Trabajo de grado Modalidad Practica Empresarial.

ⁱⁱ Facultad de Ingenierías Físico Mecánicas. Escuela de Ingeniería de Sistemas e Informática.
Director: Luis Ignacio González Ramírez.

ABSTRACT

TITLE: MANAGEMENT, USER SUPPORT, MAINTENANCE OF THE WEB PORTAL, ANALYSIS, DESIGN, DEVELOPMENT AND IMPLEMENTATION OF NEW SERVICES FOR THE WEB PORTAL OF THE METALLURGICAL AND MATERIALS SCIENCE ENGINEERING SCHOOL, AND THE SCHOOL OF INDUSTRIAL AND BUSINESS STUDIESⁱ.

AUTHORSⁱⁱ: Hernan Dario Carreño Laguado
Jorge Andrés López Figueroa

KEYWORDS: Website, Web Portal, Module, Service, EEIE (Industrial and Business Studies School), EIP (Petroleum Engineering School), EDI (Industrial Design School), EIEWeb, EIPWeb, EDIWeb.

DESCRIPTION

Web portals EEIE, EIP and EDI are positioned as the medium of communication and information for the community most important within the academic and administrative staff at their respective schools, which is why the community needs are an important factor in the continuous improvement and innovation in processes and services offered by portals, so has managed to form a website which encourages the active participation of their members. For these reasons, the software development group aims Calumet implementation of new services and the maintenance and improvement of the same.

In the academic scope to support teaching in the portal is done using the concept of virtual classroom. This is understood as the space created to complement the learning process in a non-face as a way to ensure compliance with working hours independent (TI) by the student, this space was improved allowing teachers delete files, answers, forums and other activities carried out by students from the previous semester in order to reuse existing scripts in class semesters.

Created a service which allows you to debug the portal database and deleting data records are obsolete quickly and agile, and they believe the dean's website, which will give support to the role of the Dean of the physicomechanical engineering faculty with this portal administrative faculty, constantly interact with the university community, communicating with schools belonging to the faculty in a transparent manner through new portal.

ⁱ Work Degree: Business Practice.

ⁱⁱ Industrial and Business Studies School, School of Systems Engineering and Computing Science.
Director: M.Sc. Luis Ignacio González Ramírez

INTRODUCCIÓN

Los portales EIEEWeb, EIPWeb y EDIWeb pertenecientes respectivamente a la Escuela de Estudios Industriales y Empresariales, Escuela de Ingeniería de petróleos y la Escuela de Diseño Industrial, han logrado ser el principal canal de comunicación e integración entre los miembros de las escuelas; gracias a la gran aceptación y utilidad que demuestran donde surge el objetivo fundamental de fortalecer y mejorar los servicios que ofrecen y además la creación de módulos para suplir las necesidades crecientes de los miembros de las escuelas.

El grupo de desarrollo software Calumet se encarga de desarrollar, administrar y mantener los portales EIEEWeb, EIPWeb y EDIWeb, los cuales están en continuo cambio para hacerlos un instrumento cada vez más útil, agradable y de fácil uso para los miembros. Con el objeto de llevar a cabo esta labor y proporcionar páginas con contenido dinámico se ha contado con herramientas software de libre distribución como JSP, Java, JavaScript, JQuery y MySQL.

El continuo soporte ofrecido a los portales EIEEWeb, EIPWeb y EDIWeb, por el grupo calumet se hace de forma integral permitiendo el mejoramiento continuo de los servicios ofrecidos, en esta ocasión el grupo vio la necesidad de complementar el servicio de aula virtual específicamente permitiéndole a los profesores reutilizar guiones de clase existentes en semestres posteriores. También surge la necesidad de crear un nuevo servicio orientado a depurar las bases de datos de los portales, se buscó que este servicio les facilitara a los administradores eliminar de forma ágil registros de las bases de datos que pierden vigencia en los portales.

De la misma forma y a pesar de la gran cantidad de servicios ofrecidos por los diferentes portales del grupo Calumet, nace la necesidad de ampliar el campo a los que se les ofrece estos servicios, específicamente para suplir la necesidad de la Facultad de Ingenierías Fisicomecánicas de interactuar con los otros portales del grupo, para así poder comunicarse fácilmente con las comunidad de las escuelas pertenecientes a la facultad a través de un nuevo portal independiente.

En este documento se presenta el soporte teórico, metodológico y técnico del desarrollo web de los módulos que se han mencionado, además de las mejoras a servicios actualmente en funcionamiento, labores de administración, mantenimiento y soporte a usuarios desempeñadas durante el desarrollo del proyecto.

1 PRESENTACIÓN DEL PROYECTO

1.1 ORIENTACIÓN SOBRE EL CONTENIDO DEL INFORME

Este documento presenta en detalle las etapas llevadas en el desarrollo del proyecto: ADMINISTRACIÓN, ANÁLISIS DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE INGENIERÍA INDUSTRIAL, INGENIERÍA DE PETRÓLEOS Y DISEÑO INDUSTRIAL; la información está distribuida de la siguiente forma:

CAPITULO 1. Presentación del Proyecto: Se hace un análisis de los antecedentes del proyecto, definición del problema, objetivos generales, específicos, justificación, alcances y limitaciones del proyecto.

CAPITULO 2. Marco Teórico: Se presentan los conceptos utilizados en el desarrollo técnico del proyecto.

CAPITULO 3. Marco Metodológico: Se menciona el procedimiento metodológico que se siguió para la elaboración del proyecto y las razones por la que fue elegida dicha metodología.

CAPITULO 4. Desarrollo de la herramienta: Se presenta el análisis de requisitos y el diseño de la herramienta.

CAPITULO 5. Documento de pruebas del sistema: se presenta un informe de las pruebas realizadas a la herramienta desarrollada y los resultados obtenidos.

CAPITULO 6. Conclusiones del trabajo realizado.

CAPITULO 7. Recomendaciones y sugerencias a tener en cuenta en la elaboración de futuros proyectos.

1.2 ANTECEDENTES

Teniendo en cuenta el crecimiento de la comunidad Universitaria y en este caso particular de la comunidad de la Escuela de Ingeniería de Sistemas e Informática, se hace más valioso el manejo de información y comunicación entre los miembros de la comunidad. Ante esta necesidad en el año 2004 nació la idea de crear un portal web que facilitara la comunicación y la realización de diferentes actividades dentro de la escuela, labor que se llevó a cabo por los integrantes del grupo de desarrollo software Calumet y que se está mejorando con el paso de generaciones a través del desarrollo de sus respectivos proyectos de grado, para obtener lo que hoy se conoce como el portal web de la Escuela de Ingeniería de Sistemas e Informática EISIWeb, Cabe destacar la implementación de este portal en muchas de las escuelas de la Universidad Industrial de Santander, entre las que se encuentran la Escuela de Estudios Industriales y Empresariales, Ingeniería de Petróleos y Diseño Industrial en las versiones EIEEWeb, EIPWeb y EDIWeb respectivamente, entre otras escuelas.

Gracias a la gran acogida del portal EISIWeb, surge la iniciativa de extender los servicios a usuarios nuevos y la creación de otros servicios para satisfacer las necesidades de la comunidad. Hoy la Escuela de Estudios Industriales y Empresariales, Ingeniería de Petróleos y Diseño Industrial, cuentan con un portal web dinámico que presta sus servicios a la comunidad mediante el continuo mantenimiento y evolución de la herramienta a cargo del Grupo de Desarrollo de Software Calumet y sus integrantes.

1.3 DEFINICIÓN DEL PROBLEMA

Los portales EIEEWeb, EIPWeb y EDIWeb, son sitios en crecimiento continuo, sujetos a cambios e inconvenientes diarios, razón por la cual es necesario delegar personas para ejecutar labores de administración, mantenimiento y soporte a usuarios de los portales.

Para dar una buena continuidad y soporte a los usuarios, se hace necesario la incorporación de nuevos miembros al grupo Calumet a los cuales serán heredadas la responsabilidades adyacentes al pertenecer al grupo, por esta razón, se realiza el proceso de empalme donde se brindan capacitaciones en conocimientos básicos en cuanto a la metodología de la programación y proceso para el mantenimiento y administración de los servidores.

Los servicios de los portales de las Escuelas deben mejorar constantemente y adaptarse a los cambios que se presenten en su entorno, a su vez debe dar solución a los problemas y necesidades que surjan por parte de los usuarios del sistema para incrementar su tiempo de vida útil y no llegar a convertirse en un software obsoleto, razón por la cual las labores de mantenimiento y actualización se hacen indispensables.

Motivados por brindarle a los profesores un mejor servicio y un sistema robusto capaz de suplir todas sus necesidades se complementó el servicio de aula virtual permitiéndole reutilizar guiones de clase actualmente llamados aulas de clase.

De la misma manera se observó la necesidad de brindarle a los administradores de los portales del grupo Calumet un servicio que les facilitara la depuración de las tablas en las bases de datos de los portales, labor que era muy engorrosa y tediosa, con este nuevo servicio se logran depurar registros que pierden vigencia en los portales de manera ágil.

Observando las necesidades nacientes en la comunidad y buscando complementar el actual servicio del portal Web ofrecido a las escuelas, surge la necesidad de desarrollar un nuevo portal independiente para la Facultad de Ingenierías Fisicomecánicas, para así dar soporte a la labor de la decanatura y permitirle a la facultad comunicarse con la comunidad de sus escuelas a través del nuevo portal.

1.4 OBJETIVOS

1.4.1 Objetivo General

Cumplir las funciones de soporte a los usuarios, administración y mantenimiento del portal web de la Escuela de Estudios Industriales y Empresariales, de la Escuela de Ingeniería de Petróleos, y de la Escuela de Diseño Industrial. Además analizar, diseñar desarrollar e implementar nuevos servicios que les permitan a los miembros de dichas Escuelas una interacción más fácil y eficiente.

1.4.2 Objetivos Específicos

1.4.2.1 Efectuar labores de administración del portal web de la Escuela de Estudios Industriales y Empresariales, de la Escuela de Ingeniería de Petróleos, y de la Escuela de Diseño Industrial teniendo en cuenta principalmente:

- Generar Backups (copias de respaldo) diariamente de la Base de Datos.

- Salvar la información de los Sitios Web una vez por semana por medio de copias de respaldo y mantener un histórico de los mismos en caso de alguna falla.
- Hacer seguimiento del uso que hacen los usuarios a los servicios de Foros, Eventos Propuestos, Cartelera, Archivos y Mi perfil dentro de los portales EEIEWEB, EIPWEB y EDIWEB para detectar usos indebidos o incorrectos por parte de éstos.
- Actualizar periódicamente las Bases de Datos con el objeto de mantener al día la información referente a matrículas, horarios, estados y categorías de los usuarios.
- Atender las consultas y sugerencias que los usuarios hagan para proponerlas como mejoramiento y ofrecimiento de nuevos servicios en la próxima versión del portal web.
- Realizar una revisión constante de los archivos que se suben al sitio, eliminando los que no son necesarios para evitar saturación del portal.

1.4.2.2 Llevar a cabo labores de mantenimiento en los portales EEIEWEB, EIPWEB y EDIWEB en los que podemos encontrar:

- Implementar los nuevos servicios o mejoras realizadas por otros desarrolladores del grupo CALUMET dentro de los portales EEIEWEB, EIPWEB y EDIWEB, ajustando el nuevo código, modificando la Base de Datos y realizando las pruebas necesarias.

- Hacer el seguimiento del funcionamiento de los portales para corregir posibles defectos generados por errores en el código fuente que se puedan presentar.
- Revisar y depurar la estructura de directorios y archivos del Portal Web.

1.4.2.3 Análisis, Diseño, Desarrollo e Implementación de nuevos servicios o reingeniería de servicios ya existentes, tales como:

- Depuración de tablas de las bases de datos. Dentro de las funciones de administración de las bases de datos relacionadas con el portal se encuentra la revisión y depuración de tablas que por la dinámica de los portales constantemente es necesario limpiar de datos que van perdiendo vigencia. Ejemplos de estas tablas son las que soportan los servicios de cartelera, archivos, control de actualizaciones, foros y otras. Además, se debe tener en cuenta que la eliminación de datos implica la revisión de otras tablas como las que guardan contenidos de servicios y permisos.
- Limpiar Aula de Clase: al comienzo de cada semestre es necesario que los docentes eliminen los archivos, las respuestas, los foros, y demás actividades realizadas por los estudiantes en el semestre anterior, para dar inicio al nuevo. Este servicio debe de habilitarse al profesor de la asignatura, como a los administradores del sistema
- Desarrollo del portal del decano: debe permitir en primera fase publicar en cartelera, crear foros, y enviar correos.

1.4.2.4 Desempeñar labores de soporte a los usuarios de los portales EEIEWEB, EIPWEB y EDIWEB, brindando así solución a sus diferentes necesidades o situaciones que se puedan presentar. Entre estas labores se destacan.

- Capacitar usuarios y estudiantes de primer nivel en el uso de servicios dentro de los portales EEIEWEB, EIPWEB y EDIWEB promoviendo así su divulgación y utilización.
- Atender usuarios por olvido de la contraseña, creación de grupos, solicitudes de propuestas de eventos o de cartelera en el índice y creación de agendas con eventos y foros con sus respectivas conversaciones.
- Crear usuarios de forma manual, para personas no pertenecientes a la escuela y que por algún motivo necesitan registrarse en el sitio.
- Modificar los estados de los usuarios de acuerdo a la relación con la Escuela (Activo, Inactivo, Suspendido).

1.4.2.5 Capacitar a los estudiantes que recibirán las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro de los portales EEIEWEB, EIPWEB y EDIWEB en cuanto a:

- Implantación del sitio local para la creación de nuevos servicios y realización de pruebas.

- Realizar inducción en cuanto al manejo y utilización de los JSP, Beans y Base de Datos.
- Llevar a cabo la familiarización con el entorno del portal web.

1.5 JUSTIFICACIÓN

En la actualidad las Escuelas de Estudios Industriales y Empresariales, Ingeniería de Petróleos y Diseño Industrial, cuentan con un sistema de información orientado a la Web que se encarga de la administración y control de las diferentes actividades que se realizan dentro de las Escuelas, así como el control de usuarios y servicios que se les proporciona.

Los servicios del portal de las Escuelas deben mejorar constantemente y adaptarse a los cambios que se presenten en su entorno, a su vez debe dar solución a los problemas y necesidades que surjan por parte de los usuarios del sistema para incrementar su tiempo de vida útil y no llegar a convertirse en un software obsoleto, razón por la cual las labores de mantenimiento y actualización se hacen indispensables.

Por otra parte, el soporte a la labor docente desde el Portal se realiza a partir del concepto de Aula Virtual. Este se entiende como el espacio creado para complementar el proceso de aprendizaje de manera no presencial como una forma que garantiza el cumplimiento de las horas de trabajo independiente (T.I.) por parte del estudiante, al inicio de cada semestre es necesario que los docentes puedan eliminar archivos, respuestas, foros y demás actividades realizadas por estudiantes del semestre anterior para que así los profesores puedan reutilizar un mismo guion de clase en semestres posteriores, este servicio es de vital

importancia especialmente para los profesores ya que les permite reutilizar los guiones de clase.

Actualmente la depuración de las bases de datos de los portales se hace manualmente, ingresando a cada portal y eliminando los registros tabla por tabla, labor lenta, dispendiosa y engorrosa, esta tarea es muy importante ya que por la dinámica de los portales es necesario eliminar datos que van perdiendo vigencia, por lo cual se desarrolló un servicio que permite realizar esta labor de manera rápida y ágil.

Finalmente, cabe resaltar que la Facultad de Ingenierías Fisicomecánicas no contaba con un medio de comunicación con la comunidad, razón por la cual estaba prácticamente aislada de sus escuelas, el desarrollo e implementación del portal web del decano fortalece el ámbito de la comunicación, expresión y exposición de sus logros, proyectos, labores entre otros, a la comunidad académica.

1.6 ALCANCES Y LIMITACIONES

La administración y mantenimiento en los portales EEIWeb, EIPWeb y EDIWeb, permite y garantiza su correcto funcionamiento, de esta forma responder a la demanda diaria de los usuarios, quienes contarán con un soporte para realizar diferentes actividades y resolver algunos inconvenientes que se presentan a menudo en el manejo y utilización del portal. Por otra parte, el complemento de los servicios enfocados al Aula Virtual fortalece las funcionalidades de cada portal, con lo que se constituyen, para los usuarios, en una herramienta que facilita el acceso a la información y el manejo de esta. El Aula Virtual se promueve como un espacio interactivo y dinámico que complementa el proceso de aprendizaje de una manera no presencial a partir de una metodología de trabajo en la que se integran

la teoría y la práctica. De la misma manera el nuevo portal del decano con los nuevos servicios desarrollados permitirá en esta primera fase del portal permitirá a los administrativos de la facultad de ingenierías fisicomecánicas comunicarse y acercarse a la comunidad de sus escuelas de una forma transparente mediante correos, publicaciones en cartelera y conversaciones en foros.

2 MARCO TEÓRICO

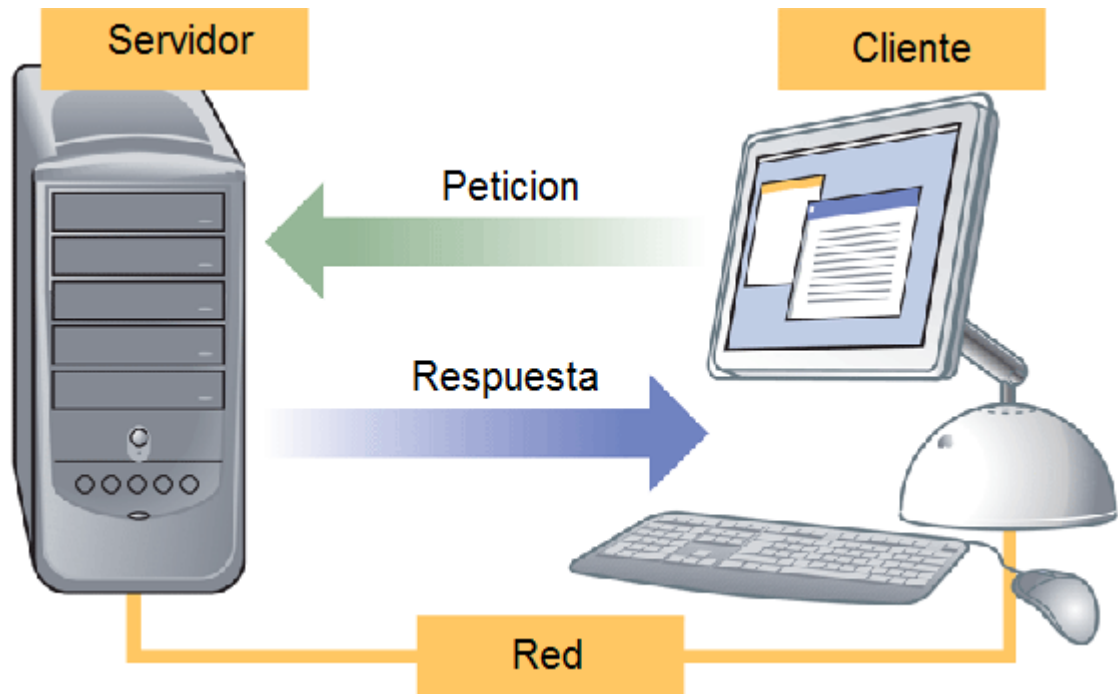
2.1 ARQUITECTURA CLIENTE/SERVIDOR

Se define como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma.

En el modelo C/S, el cliente envía un mensaje solicitando un servicio a un servidor (hacer una petición), y este envía uno o varios mensajes con la respuesta (provee el servicio) (ver figura 1). En un sistema distribuido cada máquina puede cumplir el rol de servidor para algunas tareas y el rol de cliente para otras.

La idea es tratar un computador como un instrumento, que por sí solo pueda realizar muchas tareas, pero con la consideración de realizar aquellas que son más adecuadas a sus características. Si esto se aplica tanto a clientes como servidores, se entiende que la forma estándar de aplicación y uso de sistemas C/S, es mediante la explotación de los computadores a través de interfaces gráficas de usuario; mientras que la administración de datos y seguridad e integridad que deja a cargo de computadores centrales tipo mainframe. Usualmente la mayoría del trabajo pesado se hace en el proceso llamado servidor y los procesos cliente sólo se ocupan de la interacción con el usuario (aunque esto puede variar). La arquitectura C/S es una extensión de programación modular en la que la base fundamental es separar una gran pieza de software en módulos con el fin de hacer más fácil el desarrollo y mejorar su mantenimiento.

Figura 1. Modelo Cliente/Servidor



Fuente: Internet. http://www.ecured.cu/index.php/Arquitectura_Cliente_Servidor

2.1.1 Características de la arquitectura Cliente/Servidor.

Las características básicas de una arquitectura Cliente / Servidor son:

- El proceso del cliente da la interface entre usuarios y el resto del sistema (Interacción con usuarios y el proceso del servidor), maneja recursos compartidos tales como bases de datos, impresoras, módems, etc.
- El cliente y el servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades independientes.

- Las tareas del cliente y el servidor tienen diferentes requerimientos como: velocidad del procesador, memoria o capacidad del disco, por tanto la plataforma de hardware y el sistema operativo del cliente y del servidor no son siempre la misma y eso se conoce como ambiente heterogéneo.
- La escalabilidad horizontal permite agregar más estaciones de trabajo activas sin afectar el rendimiento y la escalabilidad vertical permite mejorar las características del servidor o agregar múltiples servidores. Se puede realizar independientemente cambios en las plataformas de los clientes o de los servidores, ya sea actualización o reemplazo tecnológico, de manera transparente para el usuario final.

2.1.2 Clasificación de las arquitecturas Cliente/Servidor.

Los sistemas Cliente / Servidor se clasifican de acuerdo al nivel de abstracción del servicio que se ofrece. Se distinguen tres componentes básicos de software:

- *Presentación*: Exposición al usuario de un conjunto de objetos visuales procesados y producidos devueltos por el servidor.
- *Lógica de aplicación*: Esta capa es la responsable del procesamiento de la información que tiene lugar en la aplicación.
- *Base de datos*: Está compuesta por los archivos que contienen los datos de la aplicación.

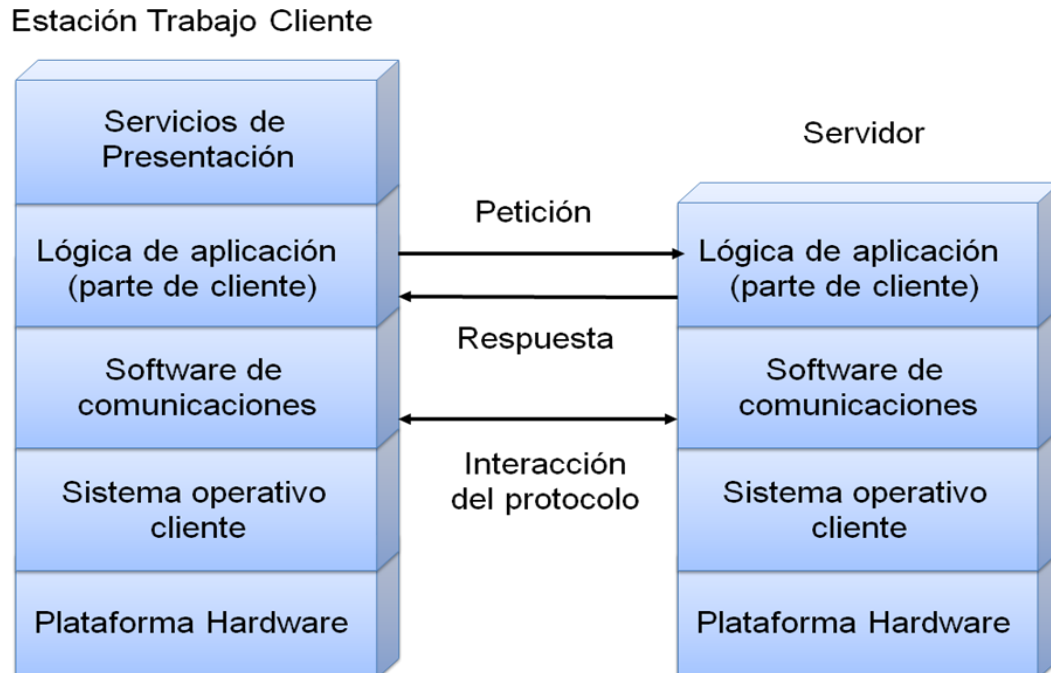
A continuación se muestra la clasificación de los sistemas Cliente / Servidor:

2.1.2.1 Arquitectura Cliente/Servidor de dos capas.

Consiste en una capa de presentación y lógica de la aplicación; y otra de la base de datos, cuando el cliente solicita recursos entonces el servidor responde directamente a la solicitud con sus propios recursos.

- Normalmente esta arquitectura es utilizada en las siguientes situaciones:
 - Cuando se requiera poco procesamiento de datos en la organización.
 - Cuando se tiene una base de datos centralizada en un solo servidor.
 - Cuando la base de datos es relativamente estática.
 - Cuando se requiere un mantenimiento mínimo.

Figura 2. Arquitectura Cliente/Servidor de Dos Capas



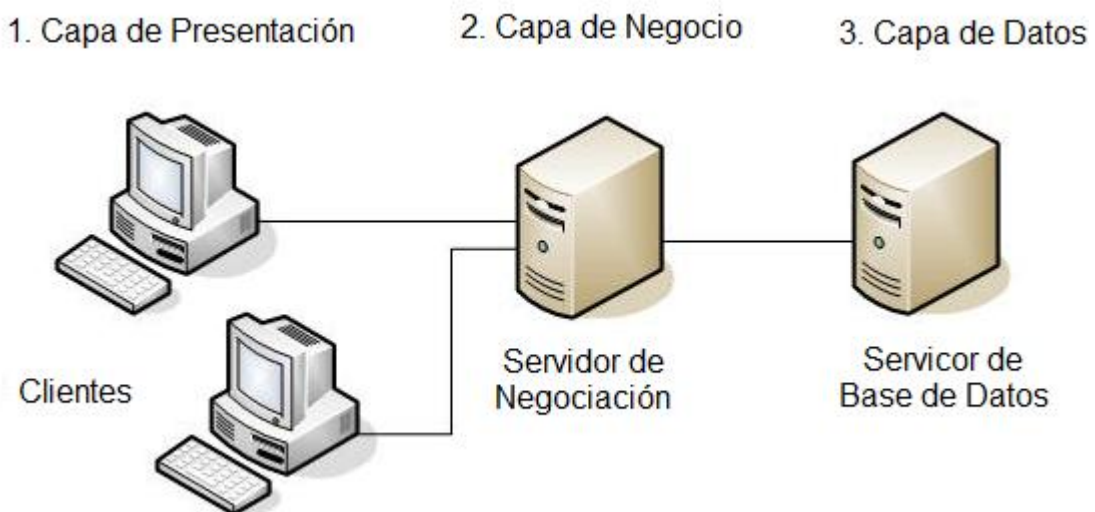
Fuente: Autores.

2.1.2.2 Arquitectura Cliente/Servidor de tres capas.

Define como organizar el modelo de diseño en capas, que pueden estar físicamente distribuidas, es decir que los componentes de una capa solo pueden hacer referencia a componentes en capas inferiores. Este patrón es importante porque simplifica la comprensión y la organización del desarrollo de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no conscientes de ningún detalle o interfaz de las superiores, está compuesta de:

- Un equipo cliente con una interfaz de usuario (habitualmente se utiliza un navegador web), que solicita los recursos.
- El servidor de aplicaciones (o software intermedio), cuya tarea es prestar los recursos solicitados, pero que requiere de otro servidor para hacerlo.
- El servidor de datos que almacena y proporciona al servidor de aplicaciones los datos que requiere.

Figura 3. Arquitectura Cliente/Servidor de Tres Capas



Fuente: Internet. http://www.ecured.cu/index.php/Archivo:Tres_capas.JPG

2.1.3 Arquitectura Cliente/Servidor aplicada.

En el desarrollo de este proyecto se recurre a arquitectura de tres capas, debido a las ventajas ofrecidas como: Escalabilidad, fácil mantenimiento y el manejo de un mayor número de usuarios que la ofrecida por la arquitectura C/S de dos capas. La arquitectura es aplicada de la siguiente forma:

- Capa de Cliente: Interfaz con el usuario, se usa un navegador web.
- Capa Intermedia: Para los servicios del negocio se utiliza un computador configurado como servidor web, el cual almacena el portal web conformado por páginas JSP y JavaBeans. Allí se realizan los procesos complejos, y se solicitan los servicios del servidor de datos cuando es necesario acceder a la información almacenada en la base de datos.
- Capa de Servidor: Se utiliza el motor de bases de datos MySQL, el cual se encuentra en el mismo servidor web.

2.1.4 Ventajas del esquema Cliente/Servidor

- La existencia de plataformas de software y hardware de varios fabricantes y cada vez más a económicas contribuye a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones.
- Este esquema facilita la integración entre sistemas heterogéneos y comparte información permitiendo que las maquinas existentes puedan ser utilizadas con interfaces amigables al usuario, de esta forma integrar los computadores con sistemas medianos y grandes, sin necesidad de que todos tengan que utilizar el mismo sistema operacional.

- Facilita a los diferentes departamentos de una organización soluciones locales, permitiendo la integración de la información principal totalmente.

2.1.5 Desventajas del esquema Cliente/Servidor

- El mantenimiento de los sistemas es complejo pues implica la interacción de diferentes partes hardware y software de diferentes proveedores, lo cual dificulta el diagnostico de fallas.
- Se cuenta con escasas herramientas para la administración y ajuste del desempeño de los sistemas, además se deben tener estrategias para el manejo de errores y para salvaguardar la consistencia de los datos.
- La seguridad del esquema C/S es preocupante, un ejemplo: las validaciones y verificaciones que se deben hacer tanto en el cliente como en el servidor.
- El desempeño es un aspecto a tener en cuenta en el esquema C/S, problemas de este estilo pueden presentarse por congestión en la red.

2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS

Las páginas dinámicas aportan grandes beneficios porque permiten entrar a bases de datos para extraer información que pueda presentarse al usuario, dependiendo de algunos permisos y de la misma forma para almacenar información.

Existen diferentes tecnologías para el desarrollo de páginas dinámicas entre ellas están:

2.2.1 Código del Lado del Cliente (Client Side Scripts).

Código ejecutado por los navegadores, el cual los computadores clientes tienen instalados. Las tecnologías más comunes de este tipo son:

- *JavaScript*: Lenguaje de programación interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Permite, crear ventanas, mostrar texto en movimiento y verificar las entradas a un formulario.
- *Controles Activos*: tecnología Microsoft que permite la creación de aplicaciones Windows, como pueden ser Visual Basic Script o Visual C. Es la respuesta de Microsoft a los Applets de Java.
- *Java Applets*: Programas escritos en lenguaje de programación Java, se incrustan en HTML y se ejecutan en el navegador gracias a la Máquina Virtual de Java (JVM) que lleva éste incorporado.

2.2.2 Código del Lado del Servidor (Server Side Scripts).

Código que se ejecuta en el servidor. Para su actividad el programa ejecuta y procesa los datos o peticiones que el usuario envía desde su navegador, para luego enviar los resultados del programa en una página HTML que el usuario verá normalmente en su navegador. Los más usados son:

- ASP (Active Server Pages): Permite crear dinámicamente páginas Web mediante HTML, scripts, y componentes de servidor ActiveX reutilizables, requiere de un computador configurado como Servidor Web de Microsoft (Microsoft Web Server), el navegador del cliente es indiferente pues el trabajo se realiza del lado del servidor. De gran uso en la gestión de Bases de Datos ya que puede conectarse a SQL, Access, Oracle u otras.
- PHP (PHP Hypertext Pre-processor): Lenguaje de programación interpretado, diseñado para la creación de páginas web dinámicas. Es un lenguaje de código abierto (Open Source) y gratuito. Su gran potencia se encuentra en la interacción con los motores de bases de datos como Oracle y MySQL.
- JSP (Java Server Pages): tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. Permiten la utilización de código Java mediante scripts.

2.2.3 Tecnología aplicada.

La tecnología aplicada para la creación del portal web es JSP, por lo tanto los nuevos servicios son desarrollados con esta misma tecnología, ya que permite producir aplicaciones independientes de la plataforma y portables a otros sistemas operativos y servidores web.

Las paginas JSP y servlets se ejecutan en la Máquina Virtual de Java, lo cual permite que se puedan usar en cualquier tipo de computador, siempre y cuando este instalada la Máquina Virtual de Java. Cada JSP se ejecuta en su propio contexto (llamado también hilo o hebra); pero no se comienza a ejecutar cada vez que recibe una petición, sino que persiste de una petición a la siguiente, de forma

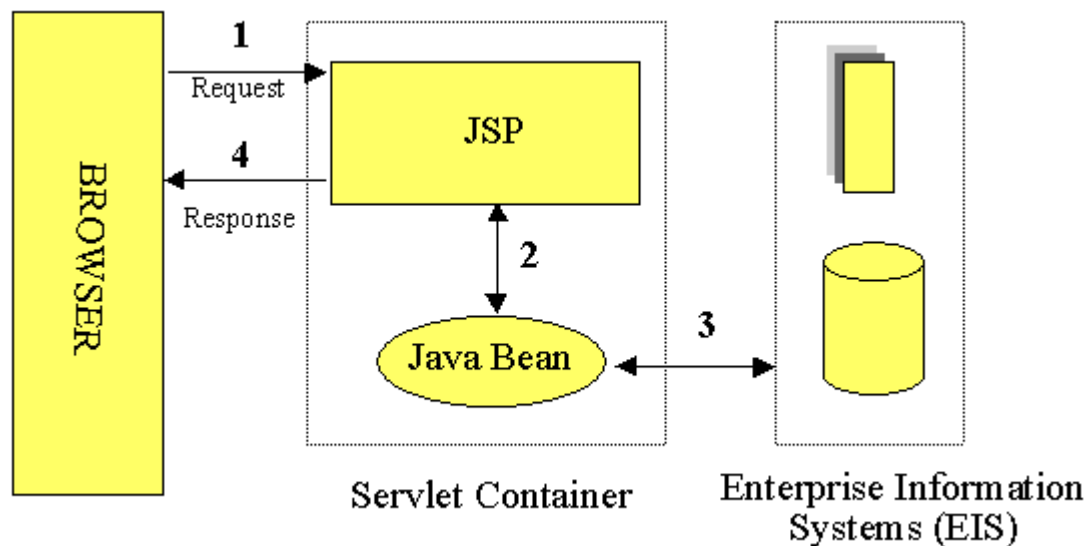
que no se pierde tiempo en invocarlo. Su persistencia permite hacer cosas de forma más eficiente como la conexión a bases de datos y manejo de sesiones.

Una página JSP se compila a una aplicación Java la primera vez que se invoca, y de esta aplicación Java se crea una clase que empieza a ejecutarse en el servidor como un servlet. Un JSP es una página web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa que recibe peticiones y genera a partir de ellas una página web.

2.2.3.1 Modelo de acceso a JSP.

1. Un usuario en su navegador web cliente hace una petición que es enviada a un archivo JSP. Este archivo accede a componentes del servidor que generan contenido dinámico y lo presentan en el navegador.
2. Después de recibir la petición del cliente, el archivo JSP pide información de un JavaBean si es necesario.
3. El JavaBean en turnos puede pedir información de otro JavaBean o de una base de datos.
4. Una vez el JavaBean genera el contenido, el archivo JSP puede consultar y presentar el contenido del JavaBean al navegador.

Figura 4. Modelo de acceso a JSP



Fuente: Internet: <http://geneura.ugr.es/~jmerelo/JSP/modelo1.gif>

La primera vez que un archivo JSP es invocado, este es compilado en un objeto, la respuesta del objeto es HTML estándar, el cual es interpretado por el navegador para ser presentado al usuario. Después de la compilación, el objeto de la página es almacenado en la memoria del servidor. En las peticiones posteriores a esta página, el servidor revisa si el archivo JSP ha cambiado. Si no ha cambiado, el servidor utiliza el objeto de la página compilada guardado en memoria para generar la respuesta al cliente, en caso contrario el servidor automáticamente compila el archivo de la página y reemplaza el objeto en la memoria.

2.3 BASES DE DATOS

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso, con una redundancia controlada y una estructura que refleja las interrelaciones y restricciones

existentes en el mundo real. En la base de datos se almacena información considerada necesaria para una determinada organización o negocio.

2.3.1 Modelos de Bases De Datos.

Este modelo describe la estructura de una base de datos, donde se incluye conceptos como: entidades, Atributos y relaciones, la mayoría de los modelos de datos poseen un conjunto de operaciones básicas para especificar consultar y actualizaciones. Las bases de datos se pueden clasificar de acuerdo a su modelo de administración de datos. Algunos modelos utilizados con frecuencia son:

2.3.1.1 Base de Datos Jerárquica.

Estas bases de datos almacenan su información en una estructura escalonada, organizando los datos en forma similar a un árbol (visto al revés), en donde un nodo padre de información puede tener varios hijos, el nodo que no tiene padres es llamado raíz, y a los nodos que no tienen hijos se les conoce como hojas. Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento. Esta limitado por su incapacidad de representar eficientemente la redundancia de datos.

2.3.1.2 Base de Datos de Red.

En este modelo se permite que un mismo nodo tenga varios padres. Ofrece una solución eficiente al problema de redundancia de datos; sin embargo, la dificultad para administrar los datos en una base de datos de red ha conllevado a que sea un modelo usado más por programadores que por usuarios finales.

2.3.1.3 Base de Datos Relacional.

Es el más utilizado para modelar problemas reales y administrar datos dinámicamente. Su fundamento es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos, también llamados tuplas. Cada relación es una tabla que está compuesta por registros (las filas de una tabla), que representan las tuplas, y campos (las columnas de una tabla). Los datos pueden ser recuperados o almacenados mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información. El lenguaje más habitual para construir las consultas a bases de datos relacionales es el Lenguaje Estructurado de Consultas (Structured Query Language, SQL), un estándar implementado por los principales manejadores de bases de datos relacionales.

2.3.2 Acceso a Base de Datos.

Para desarrollar aplicaciones que conecten bases de datos, se utilizan interfaces y programas estándar que envían demandas escritas en SQL, y procesan los resultados. Para conectarse a un motor de bases de datos determinado, se necesita un driver o una interfaz estándar que medie entre la aplicación y la base de datos.

2.3.3 Conectores más utilizados.

- ODBC (Open DataBase Connectivity): Estándar de acceso a Bases de datos, su objetivo es hacer posible el acceder a cualquier dato desde cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos (DBMS por sus siglas en inglés) almacene los datos, logra esto al insertar una capa intermedia llamada manejador de Bases de Datos, entre la aplicación y el DBMS, el propósito de esta capa es traducir las consultas de datos de la aplicación en comandos que el DBMS entienda. Para que esto

funcione tanto la aplicación como el DBMS deben ser compatibles con ODBC.

- MDB: Esta aplicación permite trabajar con tablas de base de datos creadas en Access 97/2000. Es posible abrir tablas en SQL, visualizarlas, navegar, crear y borrar índices, fijar relaciones, copiar, etc.
- JDBC (Java Database Connectivity): Interfaz de programación de aplicaciones que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java, independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede, utilizando el dialecto SQL del modelo de base de datos que se utilice.

Para el desarrollo de los portales y cada uno de sus módulos se empleó el conector JDBC. Uno de sus mayores beneficios es su capacidad para crear aplicaciones cuya programación sea independiente de la base de datos, es decir, pueden ser migradas a otro servidor de bases de datos sin complicaciones. Sin embargo, dos elementos siguen estando ligados a una base de datos en particular, el nombre de la clase que se usa para cargar el controlador JDBC y la dirección URL (Universal Resource Locator) para acceder a la base de datos.

Los servlets y las páginas JSP usan JDBC habitualmente de la misma manera que cualquier otra aplicación en Java, típicamente los datos del controlador JDBC, la cadena de conexión, el nombre de usuario y contraseña para conectarse a la base de datos son codificados dentro del programa.

Las operaciones primordiales realizadas durante la ejecución del JDBC son:

- Cargar un controlador JDBC.
- Utilizar el controlador para abrir una conexión con la base de datos.
- Formular instrucciones SQL a través de la conexión.

- Procesar los conjuntos de resultados devueltos por las operaciones SQL.

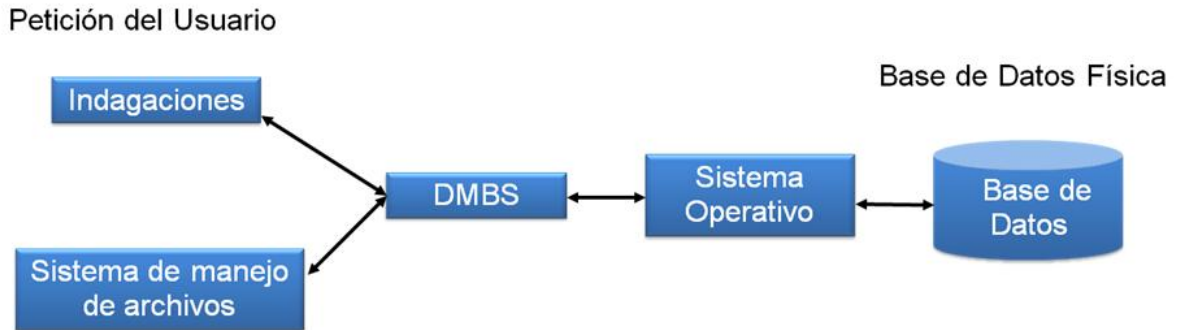
2.3.4 Manejadores o Gestores de Bases de Datos.

El sistema manejador de bases de datos es la porción más importante del software de un sistema de base de datos. Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica.

Las funciones principales de un DBMS son:

- Crear y organizar la Base de Datos.
- Establecer y mantener las trayectorias de acceso a la base de datos de tal forma que los datos puedan ser capturados rápidamente.
- Manejar los datos de acuerdo a las peticiones de los usuarios.
- Registrar el uso de las bases de datos.
- Interacción con el manejador de archivos a través de las sentencias en Lenguaje Manipulador de Datos (Data Manipulation Language, DML) al comando del sistema de archivos.
- Respaldo y recuperación: Consiste en contar con mecanismos implantados que permitan la recuperación fácilmente de los datos en caso de ocurrir fallas en el sistema de base de datos.
- Control de concurrencia: consiste en controlar la interacción entre los usuarios concurrentes para preservar la consistencia de los datos.
- Seguridad e Integridad: consiste en contar con mecanismos que permitan el control de la consistencia de los datos evitando que estos se vean perjudicados por cambios no autorizados o previstos.

Figura 5. Función del DBMS.



Fuente: Autores.

La figura muestra el DBMS como interface entre la base de datos física y las peticiones del usuario. El DBMS interpreta las peticiones de entrada/salida del usuario y las manda al sistema operativo para la transferencia de datos entre la unidad de memoria secundaria y la memoria principal. Es decir el DBMS es el corazón de la base de datos ya que se encarga del control total de los posibles aspectos que la puedan afectar. Existen diferentes manejadores de bases de datos como MySQL, ORACLE, FoxPro, Microsoft Access o PowerBuilder, en el desarrollo de los portales EEIEWeb, EIPWeb, EDIWeb y EISIWeb se utiliza MySQL.

2.3.5 MySQL.

Es un sistema de base de datos operacional considerado uno de los más importantes y utilizados por usuarios del medio para el diseño y programación de base de datos de tipo relacional. MySQL se usa como servidor a través del cual pueden conectarse múltiples usuarios y utilizarlo al mismo tiempo. La característica más interesante de MySQL es que permite recurrir a las bases de datos multiusuario a través de la web y en diferentes lenguajes de programación y diferentes plataformas que se adaptan a diferentes necesidades y requerimientos,

además MySQL es conocida por desarrollar alta velocidad de búsqueda de datos e información, a diferencia de sistemas anteriores.

2.3.6 Ventajas de MySQL.

- El MySQL es un Open Source, o sea código abierto que puede ser usado y modificado.
- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consume puede ser ejecutado en una maquina con escasos recursos sin ningún problema.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Su conectividad, velocidad, y seguridad hacen de MySQL altamente apropiado para acceder a bases de datos en internet.

2.4 NETBEANS

Es un entorno de desarrollo integrado (IDE), siendo una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas escritos en JAVA, pero puede servir para cualquier otro tipo lenguaje de programación. Netbeans es un producto libre y gratuito sin restricciones de uso.

- El Netbeans es un entorno de desarrollo integrado de código abierto escrito completamente en Java usando la plataforma Netbeans, soporta desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles).
- La versión actual es NetBeans IDE 7.1.2. Desde NetBeans IDE 6.5 se extienden las características existentes del Java EE (incluyendo Soporte a Persistencia, EEJB 3 y JAX-WS). Adicionalmente, el Netbeans Enterprise

Pack soporta el desarrollo de Aplicaciones empresariales java EE 5, incluyendo herramientas de desarrollo visuales de SOA, herramientas de esquemas XML, orientación a web servicios (for BPEL), y modelado UML. El NetBeans C/C++ Pack soporta proyectos de C/C++.

- Modularidad. Todas las funciones del IDE son provistas por módulos. Cada Módulo provee una función bien definida, tales como el soporte de Java, edición, o soporte para el sistema de control de versiones. Netbeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, permitiéndole al usuario comenzar a trabajar inmediatamente.

2.5 SISTEMA DE CONTROL DE VERSIONES

- Un sistema de control de versiones es un software que administra el acceso a un conjunto de ficheros, y mantiene un historial de cambios realizados. El control de versiones es útil para guardar cualquier documento que cambie con frecuencia, o el código fuente de un programa.
- Normalmente consiste en una copia maestra en un repositorio central, y un programa cliente con el que cada usuario sincroniza su copia local. Además, el repositorio guarda registro de los cambios realizados por cada usuario, y permite volver a un estado anterior en caso de necesidad.
- Existen multitud de sistemas de control de versiones, pero sin duda, el más popular es CVS (Concurrent Versions System). CVS tuvo el mérito de ser el primer sistema usado por el movimiento de código abierto para que los programadores colaboran remotamente mediante el envío de parches. Es de uso gratuito, código abierto, y emplea fusión de cambios.
- Subversión se creó para igualar y mejorar la funcionalidad de CVS, preservando su filosofía de desarrollo.

2.5.1 Subversión.

Sistema de control de versiones iniciando por CollabNet Inc. Emplea licencia Apache/BSD. Se usa para mantener versiones actuales e históricas y los cambios de archivos tales como los de código fuente, páginas web y/o documentación. Esto permite recuperar versiones antiguas de los datos o examinar cómo han ido evolucionando esto. Su objetivo es ser un sucesor prácticamente compatible del ampliamente usado Concurrent Version system (CVS).

Subversión puede trabajar a través de redes, lo que permite que las personas que estén en diferentes computadores puedan usarlo, con la posibilidad de que varias personas modifiquen y gestionen el mismo conjunto de datos desde sus sitios promueve la colaboración, y como el trabajo está versionado, ya que si se produce algún cambio incorrecto de los datos, sólo hace falta deshacerlo.

2.6 PROGRAMACIÓN UTILIZADA

Para el desarrollo de este proyecto se usó la Programación Orientada a Objetos (P.O.O.). La P.O.O. es una de las formas más populares de programas que usa objetos y sus interacciones para diseñar aplicaciones y programas de computador, intenta simular el mundo real a través del significado de objetos que contienen características y funciones; abstrae algunas características de sistemas naturales complejos como son:

- Atributos: estado del objeto.
- Métodos: comportamiento del objeto.
- Herencia: comportamientos comunes entre objetos relacionados para hallar relaciones de especialización y generalización de comportamientos.

2.6.1 Clases.

Definición de todos los elementos de que esta hecho un objeto. Cuando se programa un objeto y se definen sus características y funcionalidades, realmente se programa una clase. Por lo tanto para realizar la abstracción de sistemas naturales, observamos y analizamos un grupo de cosas con características comunes, el resultado de esta abstracción será válido para todas estas cosas.

2.6.2 Objetos.

Cualquier cosa real o abstracta, que posee atributos y un conjunto de operaciones que manipulan esos atributos que da un comportamiento particular. Un objeto es una instancia de una clase, el estado del objeto se determina por el estado (valor) de sus propiedades o características (atributos).

2.6.3 Atributos.

Características de un objeto siendo un conjunto de datos (valores) y calificadores para aquellos datos. Estos atributos pueden ser desde tipos de datos simples (enteros, caracteres, cadenas de texto) hasta otros objetos.

2.6.4 Métodos.

Son funciones o procedimientos propios de la clase que pueden tener acceso a los atributos de la misma para realizar las operaciones para los que son programados.

2.6.5 Herencia.

Se fundamenta en usar una clase ya creada para tomar sus características en clases más especializadas o derivadas de ésta para reutilizar el código que sea

común con la clase base, y solamente definir nuevos métodos o redefinir algunos de los existentes para ajustarse al comportamiento particular de esta subclase.

2.6.6 Beneficios de la Programación Orientada a Objetos.

- Permite obtener aplicaciones modificables y fácilmente extensibles a partir de componentes reutilizables.
- Disminución en el tiempo de desarrollo gracias a la reutilización del código.
- El desarrollo del software es más intuitivo porque las personas piensan naturalmente en términos de objetos más que en términos de algoritmos de software.

A continuación se presenta una breve descripción de Java, el lenguaje de programación orientado a objetos que se usó en el desarrollo de este proyecto:

2.6.7 Java y JDK (Java Development Kit)

Java es un lenguaje desarrollado por Sun Microsystems que permite escribir aplicaciones que puedan ejecutarse en casi cualquier plataforma. El lenguaje toma parte de la sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. Además, cuenta con una característica denominada “recolección de basura”, que examina la memoria y libera cualquier variable u objeto que no esté siendo usado. El JDK es un software que provee herramientas de desarrollo para la creación de programas en java.

Para trabajar con Java se necesita un kit de desarrollo que proporciona:

- Un compilador: *javac*
- Un intérprete: *java*

- Un generador de documentación: *javadoc*
- Un visor de applet para generar sus vistas previas, ya que un applet carece de método main y no se puede ejecutar con el programa java: Appletviewer.

2.7 SERVIDORES WEB

Es un tipo de software que se encuentra a la espera de una petición hecha por una aplicación cliente y da respuesta a dicha petición a través de una página web. Para cada transacción el servidor debe realizar dos acciones básicas: integrar todos los componentes de la página (texto, imágenes, vídeo, scripts, etc.) y enviarla rápidamente al usuario. A continuación se describe el servidor Web que se ajusta a la tecnología escogida para el proyecto.

2.7.1 Servidor Jakarta Tomcat

Servidor de aplicaciones Java basado en los estándares definidos por Sun Microsystems. Tomcat es desarrollado como parte del proyecto de código abierto Jakarta de la fundación de software Apache y es uno de los servidores de aplicaciones Java más utilizados, en especial porque es liviano, cumple con todos los estándares, sencillo de instalar, tiene muy buena documentación y es gratuito, además por ser escrito en Java funciona en cualquier sistema operativo que disponga de la Máquina Virtual de Java (JVM).

Es posible ejecutarlo desde la línea de comandos (consola o terminal), después de configurar algunas variables de entorno, sin embargo configurar cada variable de entorno y seguir los parámetros de la líneas de comando usados por Tomcat es tedioso y expuesto a errores, en su lugar se proporciona código existente para arrancar y detener el servicio.

3 MARCO METODOLÓGICO

3.1 PROTOTIPO EVOLUTIVO

Para realizar los nuevos servicios para el Portal de la Escuela de Estudios Industriales y Empresariales y de la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander se propone como metodología de desarrollo el prototipo evolutivo.

Figura 6. Prototipo Evolutivo.



Fuente: Autores.

La elección de esta metodología se debe a las siguientes razones:

- Los portales EEIWeb, EIPWeb y EDIWeb presentan constantemente necesidades por parte de los usuarios, por tanto los portales requieren nuevos servicios o mejorar existentes en el menor tiempo posible, debido a que el sistema no es un producto final y por lo tanto está en constante reconstrucción.
- Es importante el desarrollo de los aspectos visibles del sistema para poder agregar sugerencias de cambio por parte de los usuarios de los portales EIMTWeb, EIPWeb y EDIWeb en etapas tempranas durante el desarrollo.
- Durante la etapa temprana del nuevo servicio es esencial estar totalmente seguros de la interpretación apropiada de las especificaciones dadas por los usuarios y Escuelas.
- La mayoría de veces los usuarios no tienen una idea clara de que desean, por lo tanto se deben suponer lo que el usuario desea. Por esto la construcción de prototipos brinda la posibilidad de efectuar refinamientos de los requerimientos en forma sucesiva a fin de acercarse al producto deseado.
- Una ventaja es poder realizar cambios en etapas tempranas y la posibilidad de emisión de varios prototipos evaluables durante el desarrollo, obteniéndose de este modo una metodología integral para el proceso de evaluación del programa.
- Esta metodología favorece la autocrítica al sistema, lo que conlleva a que se produzca muchas pruebas antes de dar un nuevo prototipo, así como mejoras rápidas a problemas que puedan surgir durante su uso.

Procedimiento a seguir para la metodología planteada:

- Se toma nota de los requisitos para la construcción de los prototipos.
- Se especifica en detalle los objetivos globales del software a realizar, mediante una reunión entre el desarrollador y el usuario, para identificar los requisitos conocidos y concluir las áreas donde se necesita mayor definición.
- Posteriormente la presentación del diseño de un prototipo que se enfoca en la presentación de los aspectos del software visible al usuario por ejemplo, métodos de entrada y formatos de salida, para proceder a la construcción.
- El prototipo es evaluado por el usuario y se utiliza para filtrar los requisitos del software a desarrollar.
- Se produce un proceso interactivo en el que el prototipo es depurado para satisfacer necesidades del usuario, de igual forma el desarrollador obtiene una mejor comprensión de lo que hay que hacer para la entrega del producto final de ingeniería requerido por el usuario.

3.2 LENGUAJE DE MODELADO UNIFICADO

El Lenguaje de Modelado Unificado o Unified Modeling Language (UML), es el más utilizado en la actualidad. Es un lenguaje grafico estándar para visualizar, especificar, construir y documentar un sistema para describir un modelo del sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

UML no es un método de desarrollo porque no indica los pasos a seguir para llegar al código, es decir, no especifica como pasar del análisis al diseño y de este al código. Al no ser un método de desarrollo resulta ser independiente del ciclo de

desarrollo que se siga, puede encajar en un ciclo en cascada, evolutivo, espiral o en métodos ágiles de desarrollo.

3.2.1 Diagramas de UML.

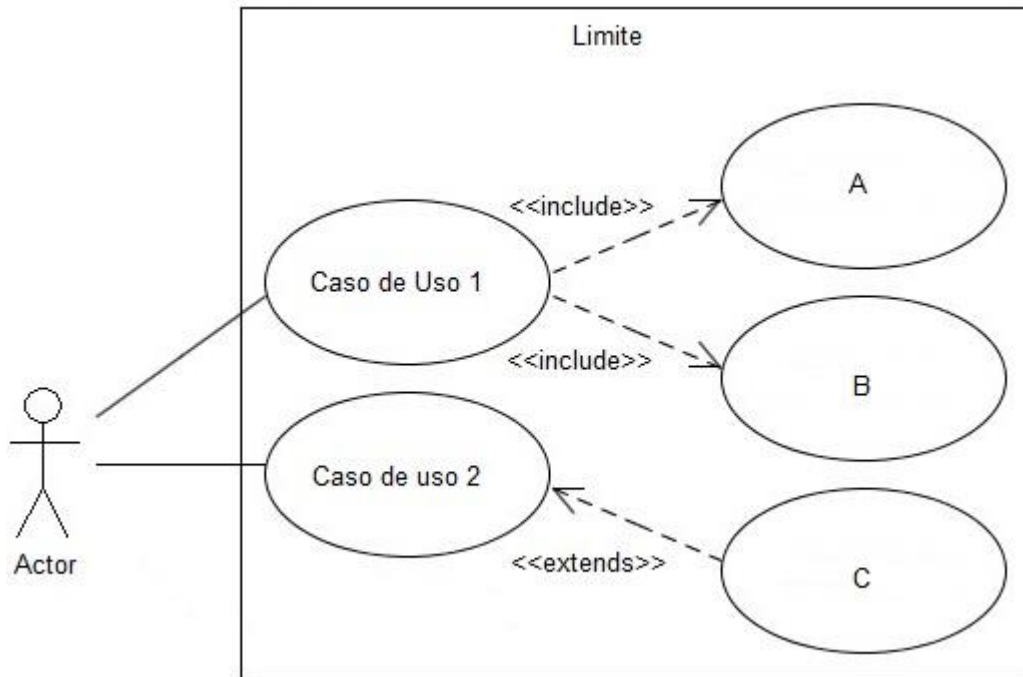
Los diagramas de UML utilizados en el desarrollo de este proyecto fueron: diagramas de casos de uso y diagramas de secuencias. Las principales razones por las cuales se prefirió UML como el lenguaje de modelado son:

- UML tiene una notación gráfica muy expresiva que permite representar todas las fases de un proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos, etc.
- UML facilita el entendimiento de la información, la función y el comportamiento de un sistema, haciendo fácil y sistemático el análisis de los requerimientos, ya que sirve de apoyo en los procesos de análisis de un problema.
- UML permite a los creadores de sistemas realizar diseños que faciliten la comunicación a otras personas de manera convencional.
- UML permite generar un punto de comparación entre lo logrado y lo planificado.

3.2.2 Diagramas de casos de uso.

Representación gráfica del entorno del sistema (actores) y su funcionalidad principal. Describe lo que hace un sistema desde el punto de vista de un observador externo, concentrándose en expresar lo que hace el sistema y no en dar respuesta de cómo lograr su comportamiento.

Figura 7. Diagramas de Casos de Uso



Fuente: Autores.

Actores: Un actor en un caso de uso representa un rol, que alguien o algo puede desempeñar dentro un sistema y no un alguien o algo específico.

En este proyecto se destacan tres clases de actores:

- **Administradores:** Son usuarios que además de pertenecer a la categoría de usuarios tienen un perfil de administrador, con el cual pueden desempeñar ciertas labores diferentes a un usuario normal dentro del sitio. Estos son: los Auxiliares de administración del portal, profesores, secretaria con ciertos privilegios. Dentro de esta categoría se incluye también el súper administrador.

- **Súper Administrador:** Es el tipo de usuario que puede administrar, controlar y modificar el sitio EEIEWeb, EIPWeb y EDIWeb, sus parámetros y sus usuarios.
- **Usuario EEIE, EIP, EDI:** Es el tipo de usuario común del sitio EEIEWeb, EIPWeb y EDIWeb y a quien van dirigidos los servicios. Este usuario solo tiene el control sobre sus privilegios.

Inclusión (include–uses): Es una forma de interacción, un caso de uso dado puede "incluir" otro. Una inclusión es utilizada para indicar que un caso de uso depende de otro, es decir, la funcionalidad de determinado caso de uso se requiere para realizar las tareas de otro. En la figura 7 el caso de uso “D” depende de los casos de uso “A”, “B” y “C”.

Extensión (extend): Es otra forma de interacción, una extensión representa una variación de un caso de uso a otro, es decir, una dependencia específica entre los casos de uso, a través de la cual un caso de uso puede extender a otro.

3.2.3 Diagramas de secuencias.

Es aquel que muestra la forma en que los objetos interactúan entre sí al transcurrir el tiempo. Consta de objetos que se representan del modo usual: rectángulos con nombre (subrayado), mensajes representados por líneas continuas con una punta de flecha y el tiempo representado como una progresión vertical.

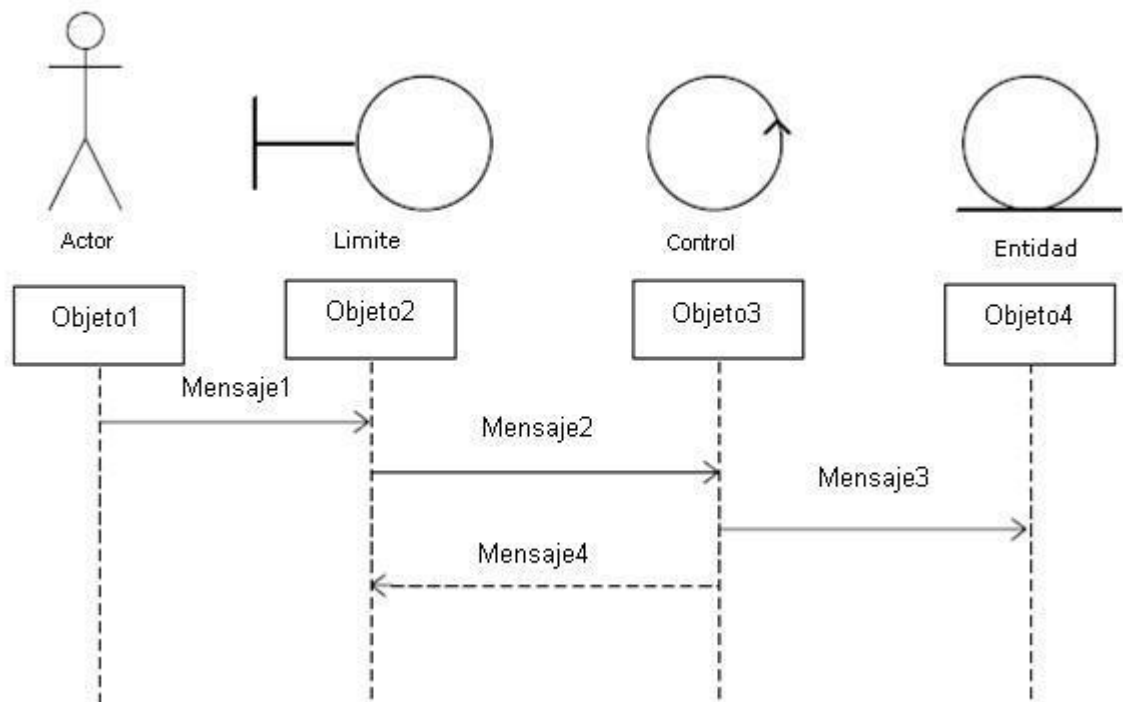
Objetos: Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de cada objeto será una línea discontinua conocida como la *línea de vida* de un objeto. Con la línea de vida se encuentra un pequeño rectángulo conocido como *activación*, el cual representa la ejecución de una operación que realiza el objeto.

Mensaje: Un mensaje que va de un objeto a otro pasa la línea de vida de un objeto a otro. Un objeto puede enviarse un mensaje a sí mismo. Un mensaje puede ser simple, sincrónico o asincrónico.

Tiempo: El diagrama representa al tiempo en dirección vertical. Inicia en la parte superior y avanza hacia la parte inferior. Un mensaje que esté más cerca de la parte superior ocurrirá antes que uno que esté cerca de la parte inferior.

GUI: (Siglas en Ingles) La interfaz gráfica de usuario; es la interfaz de interacción del usuario y en la que más interactividades se presentan con otros objetos.

Figura 8. Diagrama de Secuencias



Fuente: Autores.

3.3 ESTÁNDARES DE PROGRAMACIÓN

3.3.1 Modelo de datos.

Es un lenguaje utilizado para la descripción de una base de datos, por lo general permite describir estructuras de datos de la base de datos (el tipo de datos que incluye la base y la forma en que se relacionan), las restricciones de integridad (las condiciones que los datos deben cumplir para reflejar correctamente la realidad deseada) y las operaciones de manipulación de los datos (agregar, borrar, modificar).

3.3.2 Nombres de las tablas.

Los nombres de los campos, así como de las tablas de la base de datos, se escriben en minúsculas, exceptuando la primera letra de cada palabra que conforme su nombre; si es un nombre compuesto por dos o más palabras, los nombres tendrán en mayúscula la primera letra de cada palabra que la forma.

Se han definido tres categorías para las diferentes tablas que conforman la base de datos. Dada la categoría de la tabla, se antepondrá un prefijo a su nombre que permita conocer la categoría a la que pertenece. Las categorías son:

- *Tabla básica:* Aquella cuyos registros son necesarios para el correcto funcionamiento de la base de datos. Estas tablas no experimentan muchos cambios en los datos. El prefijo a anteponer a los nombres de estas tablas es “TB_”, es decir la que almacena las Materias de Grado es llamada “TB_MateriasGrado”.

- *Tabla de Relación:* Surge de la relación muchos a muchos de una o dos tablas cualquiera. Los nombres de las tablas de relación deberán ser descriptivos para cada relación. El prefijo a anteponer a los nombres de estas tablas es “TR_”, es decir la tabla “Directores” es conocida como “TR_Directores”.
- *Tabla Principal:* Aquella cuya población de registros tiende a crecer en gran cantidad y que además no es posible clasificar como tabla básica o de relación. Un ejemplo de tabla principal es la tabla que almacena los usuarios del portal EISIWeb. El prefijo a anteponer a los nombres de estas tablas es “TP_”, es decir la tabla “Usuarios”, es conocida como “TP_Usuarios”.

3.3.3 Clases.

Los nombres de las clases deben ser sustantivos en plural, la primera letra de cada palabra debe ser mayúscula. Estos deben ser simples, descriptivos como por ejemplo: EncuestasAdministra.java, CambioHorarios.java.

3.3.4 Páginas JSP.

Los nombres de las paginas JSP que componen el portal EEIEWeb, EIPWeb y EDIWeb serán escritos inicializando en mayúscula precedido de letras minúsculas, en caso de que el nombre del JSP compuesto por dos o más palabras, entonces la primera de cada palabra interna debe ir en mayúscula , por ejemplo:

CorreosPorPoseidon.jsp, BorrarTablas.jsp

3.3.5 Organización de Directorios.

Los Directorios del sitio están organizados de tal manera que los archivos que se almacenen en ellos correspondan a lo que describe el nombre del directorio. Por ejemplo:

- El sitio cuenta con un Directorio llamado "images"; en éste se encuentran almacenadas todos los archivos .jpg, .gif, .png.
- Si nos referimos a los archivos compilados de java (.class), estos se guardarán en un directorio llamado WEB-INF que por defecto es para esta extensión de archivos.

4 DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO

Como se mencionó inicialmente para el desarrollo de los servicios de: Depuración de tablas de bases de datos, limpieza de guion de clase ahora llamado aula de clase, desarrollo e implementación del portal web del decano, se siguió la metodología de prototipo evolutivo.

Al inicio del proyecto se elaboró un primer prototipo basado en los requerimientos primarios, luego este prototipo fue mejorado con la inclusión de nuevos requerimientos surgidos en el desarrollo, a medida que se generaba un prototipo este era sometido a pruebas de funcionamiento y se realizaban las correcciones pertinentes a partir del resultado de dichas pruebas.

4.1 PROTOTIPO ESPERADO

Iniciado el proyecto no se tenía bien definido cuál sería el producto final, pero a medida que se evoluciono en los prototipos entonces se realizaron pruebas y análisis del sistema para comprobar que se estaba acercando a los requerimientos iniciales con el fin de enfocar exitosamente a la solución de todas las necesidades de los usuarios, todo este desarrollo se resume a continuación:

Análisis de Requisitos.

El objetivo específico inicial y los requisitos que surgieron se dieron gracias a la presentación de prototipos y la realimentación con el cliente. Para cada objetivo se listan los requerimientos detallados de este, los cuales se cumplieron para el prototipo final.

- 1. Depuración de tablas de las bases de datos.** Dentro de las funciones de administración de las bases de datos relacionadas con el portal se encuentra la revisión y depuración de tablas que por la dinámica de los portales constantemente es necesario limpiar de datos que van perdiendo vigencia. Ejemplos de estas tablas son las que soportan los servicios de cartelera, archivos, control de actualizaciones, foros y otras. Además, se debe tener en cuenta que la eliminación de datos implica la revisión de otras tablas como las que guardan contenidos de servicios y permisos.

Objetivo Inicial:

- Crear un servicio que para cada tabla a depurar despliegue el listado de registros que han perdido vigencia en los portales, ya sea por estado o fecha, junto a un botón que al oprimirlo permita eliminar los registros seleccionados.

Requisitos finales del objetivo:

- Sólo tienen acceso a este servicio los usuarios con perfil de administradores.
 - Este servicio permite eliminar registro por registro o una selección de registros.
 - Lista registros que han perdido vigencia por estado.
 - Lista registros que han perdido vigencia por fecha.
 - Se podrá ordenar los registros de forma ascendente y descendente por el nombre del campo que se requiera.
-
- 2. Limpiar Aula de Clase:** al comienzo de cada semestre es necesario que los docentes eliminen los archivos, las respuestas, los foros, y demás actividades realizadas por los estudiantes en el semestre anterior, para

dar inicio al nuevo. Este servicio debe de habilitarse al profesor de la asignatura, como a los administradores del sistema.

Objetivo inicial:

- Permitirle al profesor de la asignatura eliminar los archivos, respuestas, foros y demás actividades realizadas por los estudiantes del semestre anterior.

Requisitos finales del objetivo:

- El profesor puede eliminar los archivos, respuestas foros, y demás actividades realizadas por los estudiantes.
- El profesor puede reutilizar el guion de clase ahora llamado aula virtual en semestres posteriores.

3. Desarrollo del portal del decano: debe permitir en primera fase publicar en cartelera, foros, y enviar correos.

Objetivo inicial:

- Instalación e implantación de un nuevo portal para la facultad de ingenierías fisicomecánicas el cual le permita tanto al decano como a los administrativos de la facultad comunicarse con sus escuelas mediante cartelera, foros y correos.

Requisitos finales del objetivo:

- Instalar servidor web especializado en JSP apache-tomcat sobre fedora 15.

- Realizar configuraciones de seguridad mediante ssh e iptables para bloquear acceso no autorizado al servidor.
- Instalación de servidor de base de datos MySQL realizando configuraciones necesarias para restringir el acceso a información de las bases de datos.
- Configurar la máquina para arranque automático en caso de fallos de electricidad.
- Permitir conexiones a base de datos y conexiones scp mediante open-ssh para garantizar conexiones seguras.
- Implantar el sitio web en el servidor verificando su perfecto funcionamiento.
- Subir información (imágenes, directivos e información en general) actualizada de la facultad a la plataforma.
- Brindar soporte a la entrada en funcionamiento de la plataforma para filtrar posibles fallos no vistos en pruebas iniciales.
- Permitirá publicar en cartelera general de los portales de las escuelas.
- Permitirá crear foros entre las escuelas pertenecientes a la facultad.
- Permitirá enviar correos a las escuelas pertenecientes a la facultad.

Para todos los servicios se debe incluir la rutina de logueo de usuario y verificación de sesión, requerimiento de seguridad propio de los sitios web a los que presta sus servicios el grupo CALUMET.

Administración.

Objetivos iniciales:

- Efectuar labores de administración de los portales web de la Escuela de Estudios Industriales y Empresariales, de la Escuela de Ingeniería de Petróleos y la Escuela de Diseño Industrial; teniendo en cuenta entre otras:

- Generar Backups (copias de respaldo) diariamente de la Base de Datos.
- Salvar la información de los sitios web semanalmente, por medio de copias de respaldo y mantener un histórico de los mismos en caso de alguna falla.
- Hacer seguimiento del uso que hacen los usuarios en cuanto a los servicios de Foros, Eventos Propuestos, Cartelera, Archivos y Mi perfil dentro de los portales EEIEWeb. EIPWeb y EDIWeb para detectar usos indebidos o incorrectos por parte de éstos.
- Actualizar periódicamente las Bases de Datos con el objeto de mantener al día la información referente a matrículas, horarios, estados y categorías de los usuarios.
- Atender consultas y sugerencias que los usuarios hagan para proponerlas como mejoramiento y ofrecimiento de nuevos servicios en la próxima versión del portal web.
- Realizar una revisión constante de los archivos que se suben al sitio, eliminando los que no son necesarios para evitar saturación del portal.

Requisitos finales del objetivo:

Para usuarios con perfil de administrador:

- Corregir cada uno de los fallos que a diario se presentan en los portales web.
- Mantenimiento a la bases de datos para eliminar tablas que ya no se usan o crear nuevas que se necesitan para que los nuevos servicios funcionen.
- Actualizar el diagrama Entidad/Relación de la base de datos Diamante y subirlo al portal EISIWeb, en el espacio del grupo Calumet; para realizar futuras actualizaciones sobre estos últimos.

- Revisar la estructura de directorios del portal, borrar los archivos que ya no se usan y agregar nuevos servicios implantados en el portal.
- Realizar limpiezas en cuanto a conversaciones y datos que ya no sean necesarios dentro de las Bases de datos.
- Mantener los JavaBeans actualizados y subirlos al portal EISIWeb, en el espacio del grupo Calumet; para realizar futuras modificaciones sobre estos últimos.

Cada una de las actividades mencionadas en el objetivo inicial se llevó a cabo durante la práctica y desarrollo del proyecto.

Mantenimiento.

Objetivos iniciales:

- ✓ Llevar a cabo labores de mantenimiento a los portales EEIEWeb, EIPWeb y EDIWeb, en los que podemos encontrar:
 - Implementar los nuevos servicios o mejoras realizadas por otros desarrolladores del grupo CALUMET, dentro de los portales EEIEWeb, EIPWeb y EDIWeb, ajustando el nuevo código, modificando la Base de Datos y realizando las pruebas necesarias.
 - Hacer el seguimiento del funcionamiento de los portales para corregir posibles defectos generados por errores en el código fuente que se puedan presentar y dañar el funcionamiento del sistema.
 - Revisar y depurar la estructura de directorios y archivos del portal web.
 - Realizar el mantenimiento de un archivo clasificado de las solicitudes de cambios y correcciones que hagan los usuarios.

Requisitos finales del objetivo:

Para usuarios con perfil de administrador:

- Revisar regularmente el historial de cambios para supervisar el uso correcto que los usuarios le dan a los servicios.
- Realizar copias periódicas de la base de datos Diamante y de los directorios de los sitios EEIEWeb, EIPWeb y EDIWeb.
- Eliminar de la base de datos diamante información que no se usa, ya sean archivos o conversaciones.
- Hacer la actualización periódica de la base de datos Diamante.

Cada una de las actividades mencionadas en el objetivo inicial se llevó a cabo durante la práctica.

Soporte a Usuarios.

Objetivo inicial:

- Desempeñar labores de soporte a los usuarios de los portales EEIEWeb, EIPWeb y EDIWeb, brindando así solución a sus diferentes necesidades, conflictos o situaciones que se puedan presentar. Entre estas labores se destacan:
 - Capacitar usuarios y estudiantes del primer nivel en el uso de servicios dentro del portal EEIEWeb, EIPWeb y EDIWeb, promoviendo así su utilización.
 - Atender usuarios por olvido de la contraseña, creación de grupos, solicitudes de propuestas de eventos, solicitudes de cartelera y creación de agendas con eventos y foros.

- Crear usuarios de forma manual para personas no pertenecientes a EEIEWeb, EIPWeb o EDIWeb y que por algún motivo necesitan registrarse en el sitio web correspondiente.
- Modificar los estados de los usuarios de acuerdo a la relación con la escuela (activo, inactivo, suspendido).

Requisitos finales del objetivo:

Para usuarios con perfil de administrador:

- Conocer el total funcionamiento de los portales EEIEWeb, EIPWeb y EDIWeb, a fin de dar solución a las posibles dificultades que se presenten.
- Investigar las soluciones a nuevas dificultades que se presenten.

Cada una de las actividades mencionadas en el objetivo anterior se llevó a cabo durante la práctica en repetidas ocasiones. Además, se dio orientación a los usuarios sobre el uso de determinados servicios.

Capacitación a nuevos integrantes del grupo Calumet.

Objetivo inicial:

- Capacitar a los estudiantes que relevaran las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro de los portales EEIEWeb, EIPWeb y EDIWeb en cuanto a:
 - Implantación del sitio local para la creación de nuevos servicios y realización de pruebas.
 - Realizar inducción en cuanto al manejo y utilización de los JSP, JavaBeans y Base de Datos.

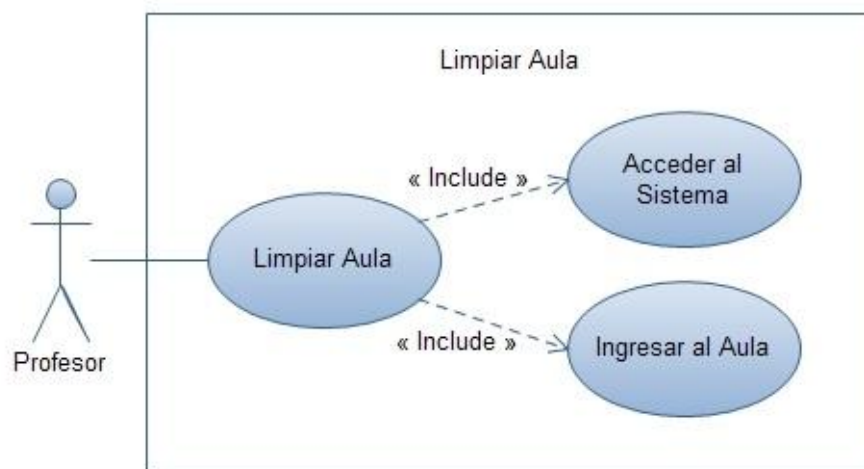
- Llevar a cabo la familiarización con el entorno de los portales web.

Requisitos finales del objetivo:

- Para usuarios con perfil de administrador.
- Dar orientación a los nuevos integrantes del grupo sobre posibles errores en la instalación del sitio local.
- Dar orientación básica a los nuevos integrantes del grupo sobre la programación en JSP, uso de los JavaBeans y funcionalidad de cada tabla dentro de la base de datos.
- Dar capacitación a los nuevos integrantes del grupo sobre los estándares que se siguen para la programación en cuanto a nombres y estilos.
- Orientar a los nuevos integrantes sobre el procedimiento para crear y habilitar nuevos servicios en el sitio.

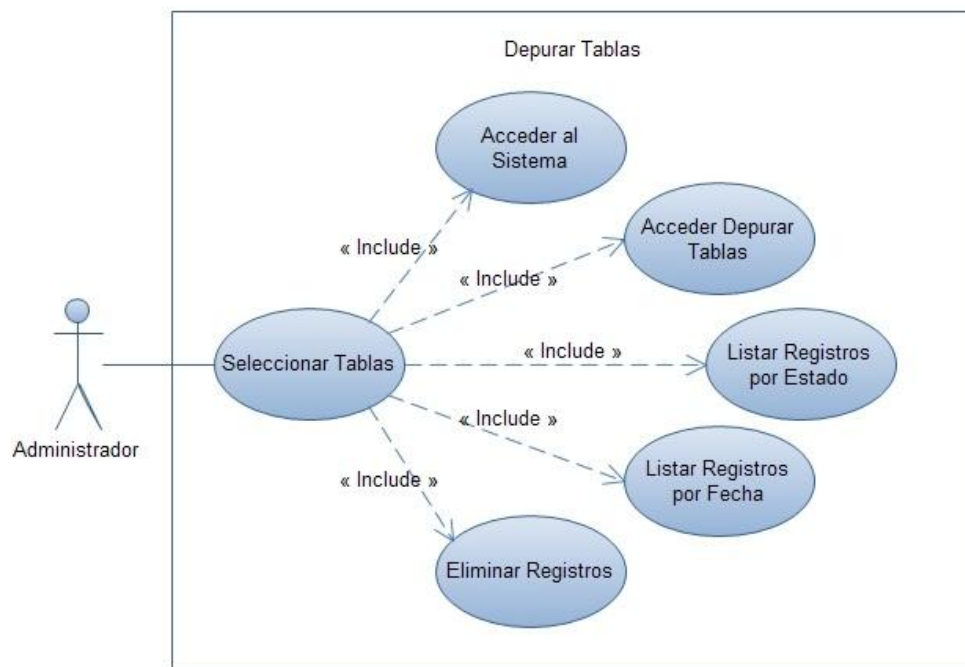
4.1.1 Diagramas de Casos de Uso.

Figura 9. Diagrama de Casos de Uso: Limpiar Aula de Clase.



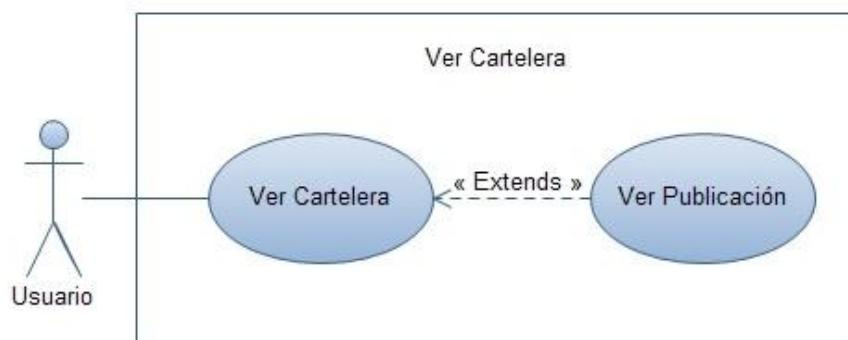
Fuente: Autores.

Figura 10. Diagrama de Casos de Uso: Depurar Tablas.



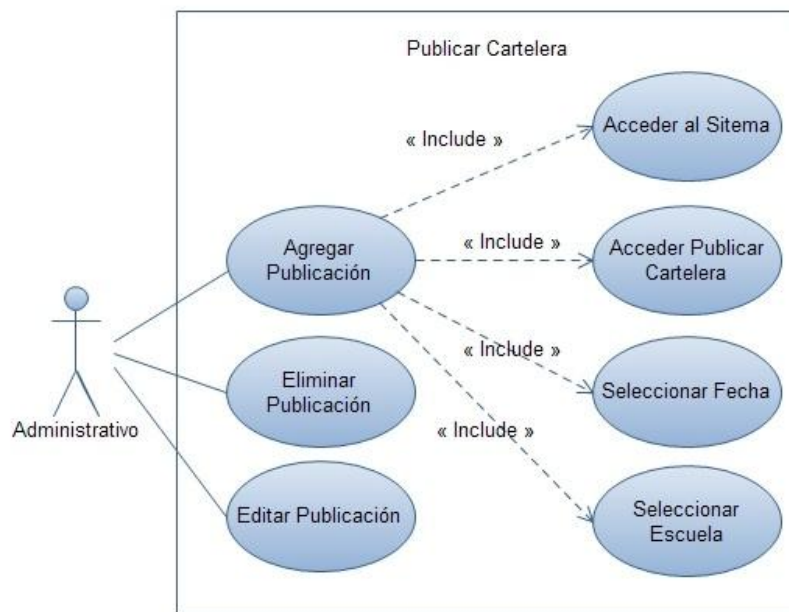
Fuente: Autores.

Figura 11. Diagrama de Casos de Uso: Ver Cartelera.



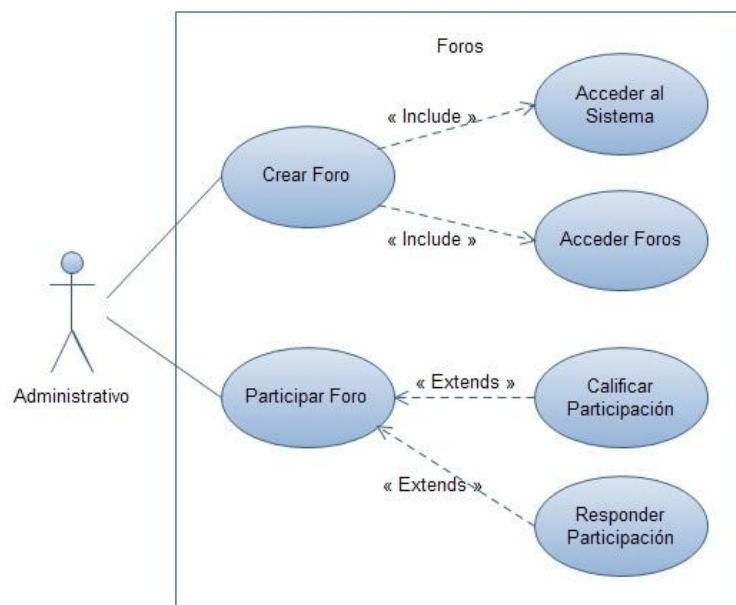
Fuente: Autores.

Figura 12. Diagrama de Casos de Uso: Publicar Cartelera.



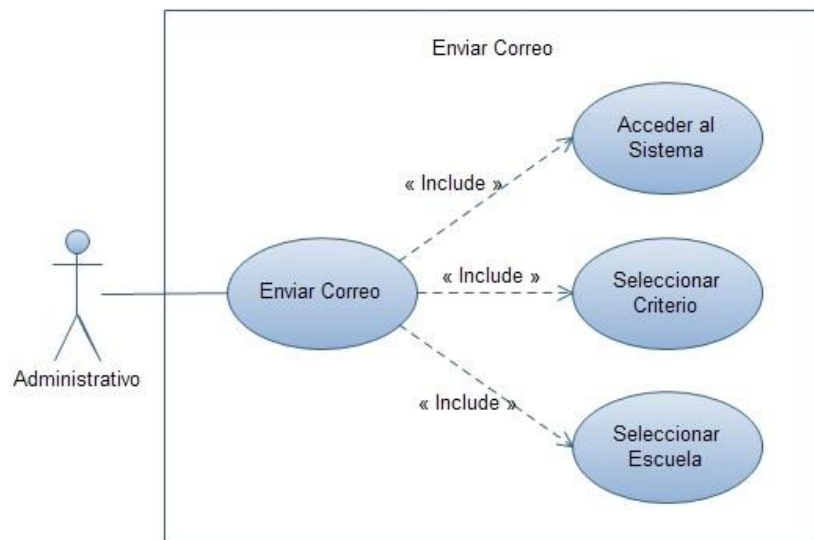
Fuente: Autores.

Figura 13. Diagrama de Casos de Uso: Foros.



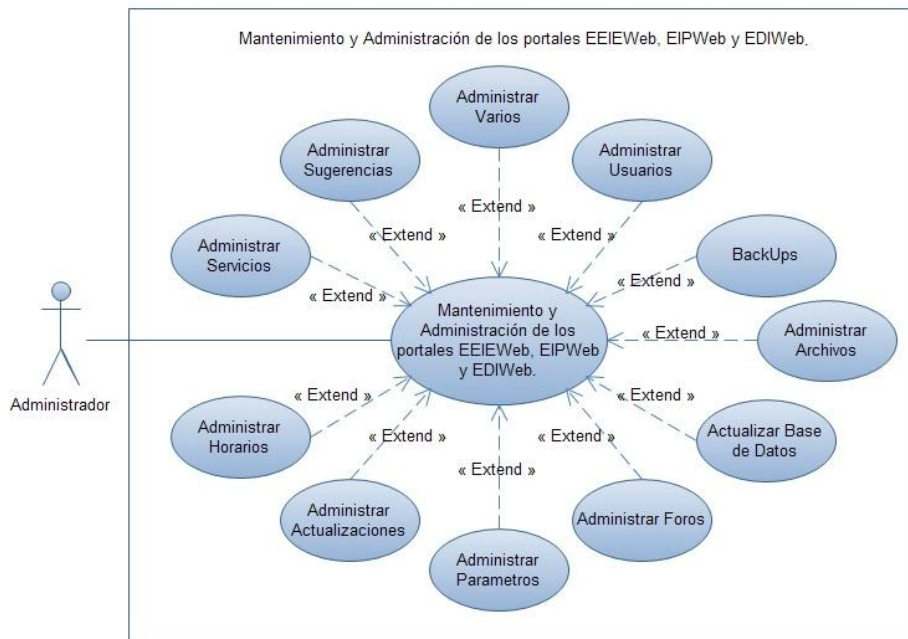
Fuente: Autores.

Figura 14. Diagrama de Casos de Uso: Enviar Correo.



Fuente: Autores.

Figura 155. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.



Fuente: Autores.

4.1.2 Documentación de Casos de Uso del Sistema.

4.1.2.1 Servicio para limpiar aula de clase.

Tabla 1. Casos de uso: Limpiar Aula.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Ingresar al Aula	Acceder al servicio de aula virtual, seleccionar un guion de clases ahora llamado aula virtual e ingresar por dirigir asignatura.
Limpiar Aula	Se eliminan los archivos, respuestas y demás actividades realizadas por los estudiantes

Fuente: Autores.

4.1.2.2 Servicio para depurar tablas.

Tabla 2. Casos de uso: Depurar Tablas.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder Depurar Tablas	Entrar al servicio del portal depurar tablas en el cual se puede elegir la tabla a depurar.
Seleccionar Tabla	Selecciona una tabla a depurar, ya sea por estado o fecha.
Listar Registros por Estado	Lista los registros de la tabla seleccionada según estado.

Listar Registros por Fecha	Lista los registros de la tabla seleccionada según fecha.
Eliminar Registros	Permite eliminar de la base de datos los registros seleccionados, ya sea por estado o por fecha.

Fuente: Autores.

4.1.2.3 Servicio Portal Decano.

Tabla 3. Casos de uso: Ver Cartelera.

TITULO	FUNCIONES PRIMARIAS
Ver Cartelera	Listar una breve información de todas las publicaciones vigentes de acuerdo a la fecha.
Ver Publicación	Ver la información detallada de cada publicación, incluyendo el autor y la fecha de publicación y escuela de origen.

Fuente: Autores.

Tabla 4. Casos de Uso: Publicar Cartelera.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder Publicar Cartelera	Ingresar al servicio Publicar cartelera en el cual se puede agregar, eliminar o editar una publicación en cartelera.

Agregar Publicación	Permite agregar una publicación a cartelera general desde el portal del decano a los portales de las escuelas pertenecientes a la facultad y a otras escuelas.
Seleccionar Fecha	Para poder agregar una publicación se tiene que seleccionar una fecha mayor a la fecha actual.
Seleccionar Escuela	Para poder agregar una publicación se tiene que seleccionar al menos una escuela a la que se desee publicar.
Eliminar Publicación	Permite eliminar una publicación de cartelera general al autor de la misma.
Editar Publicación	Permite editar una publicación de cartelera general al autor dela misma.

Fuente: Autores.

Tabla 5. Caso de Uso: Foros.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder Foros	Acceder al servicio de foros, seleccionar foros UIS.

Crear Foro	Permite crear un nuevo foro, para esto se debe colocar un título, escoger tanto las escuelas como las categorías a las que va dirigida el foro y un mensaje acorde con el nuevo foro.
Participar Foro	Agregar participación del usuario a una conversación pre-visualizando sus participaciones anteriores.
Calificar	Dar una calificación a una participación o a una respuesta de participación por medio de un puntaje visualizado por medio de estrellas.
Responder Participación	Responder a una participación de un usuario en el foro.

Fuente: Autores.

Tabla 6. Casos de uso: Enviar Correos.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Seleccionar Escuela	Permite seleccionar una o más escuelas a las que se desean enviar el correo.
Seleccionar Criterio	Permite seleccionar criterio de envío de correo según categoría.
Enviar Correo	Envío de correo a las escuelas y usuarios de la categoría seleccionada.

Fuente: Autores.

4.1.2.4 Mantenimiento y Administración de los portales EEIWeb, EIPWeb y EDIWeb.

Tabla 7. Casos de uso: Mantenimiento y Administración de los portales EEIWeb, EIPWeb y EDIWeb.

TITULO	FUNCIONES PRIMARIAS
Administrar Servicios	Crear, modificar y eliminar servicios del portal. Establecer parámetros.
Administrar Sugerencias	Revisar todas las sugerencias para atender o eliminar.
Administrar Varios	Ejecutar labores de mantenimiento que no clasifican en otras categorías. Llevar el contador de visitas a cero. Insertar, cambiar y eliminar banner.
Administrar Usuarios	Crear usuarios manualmente. Ver información de usuarios. Cambiar parámetros. Cambiar clave. Autorizar servicios a categorías y perfiles. Crear, modificar, eliminar categorías y perfiles. Enviar Correos.
Backups	Generar copias de la base de datos División y Diamante.
Administrar Archivos	Ver información, descargar y eliminar archivos cargados por los usuarios.

Actualizar Base de Datos	Actualizar tablas de datos de División y Diamante. Ver contadores de registros.
Administrar Foros	Ver información y eliminar conversaciones de los usuarios en foros.
Administrar Parámetros	Crear, modificar y eliminar los diferentes parámetros del portal.
Administrar Actualizaciones	Ver historial y estadísticas de cambios del portal web.
Administrar Horarios	Cambiar, restaurar, ver incoherencias y cambios de horarios. Crear grupo de materia.

Fuente: Autores.

4.1.3 Diseño y Análisis.

4.1.3.1 Diagrama Entidad/Relación de los servicios desarrollados.

- Servicio Limpiar Aula:

Figura 166. Diagrama E/R: Limpiar Aula:

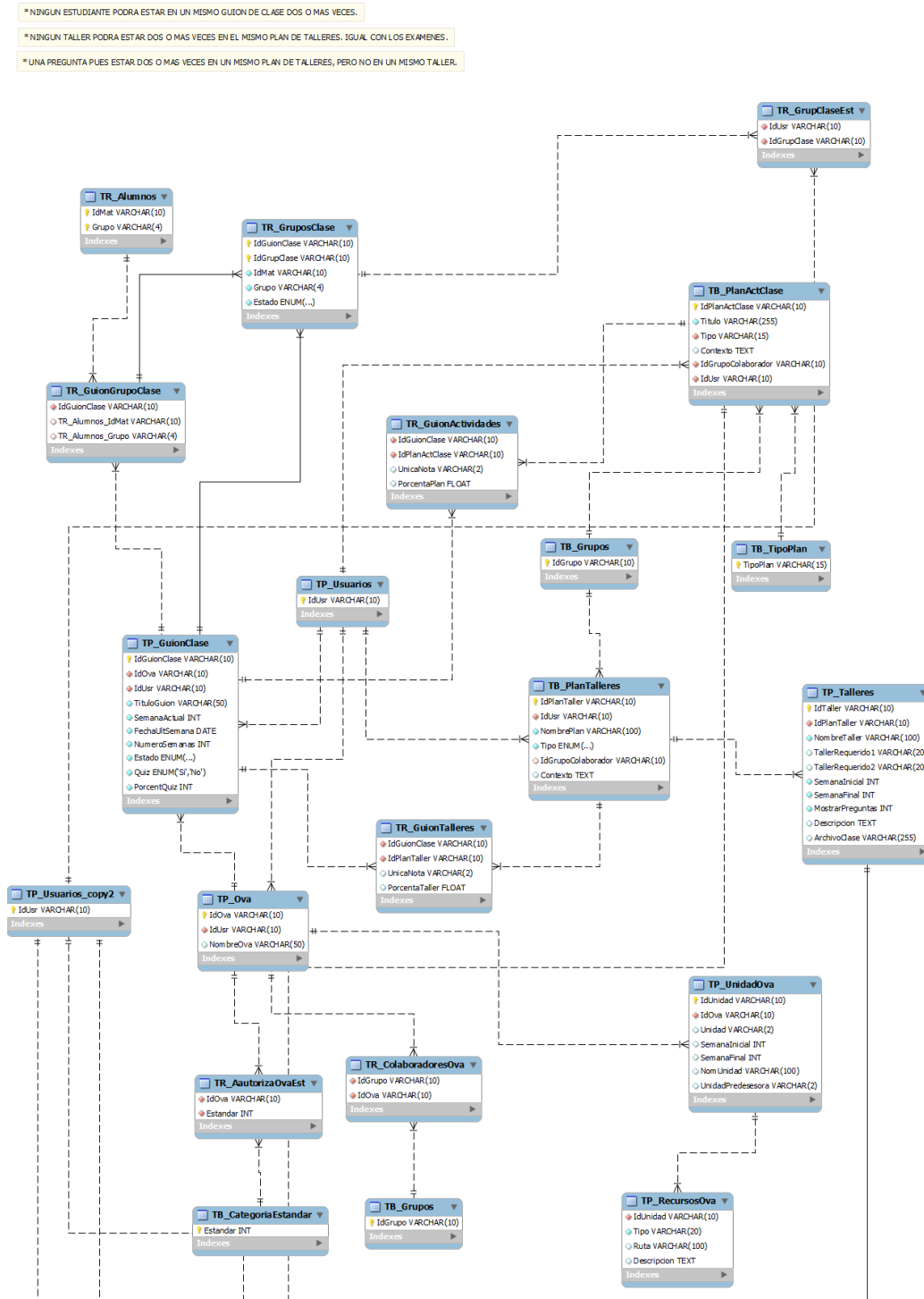
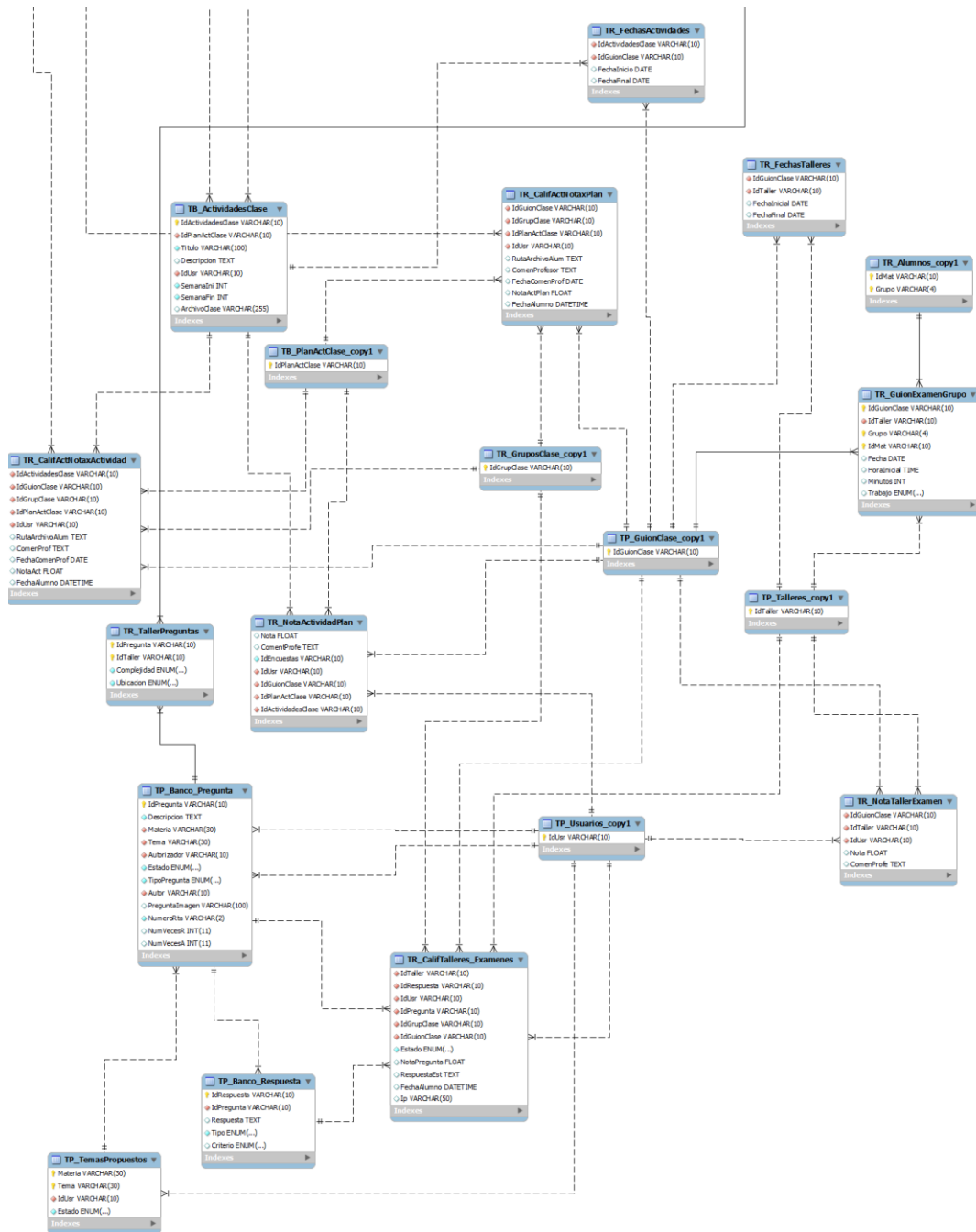


Figura 177. Continuación Diagrama E/R: Limpiar Aula:



- Servicio para depurar tablas:

Figura 18. Diagrama E/R: Depurar Tablas:

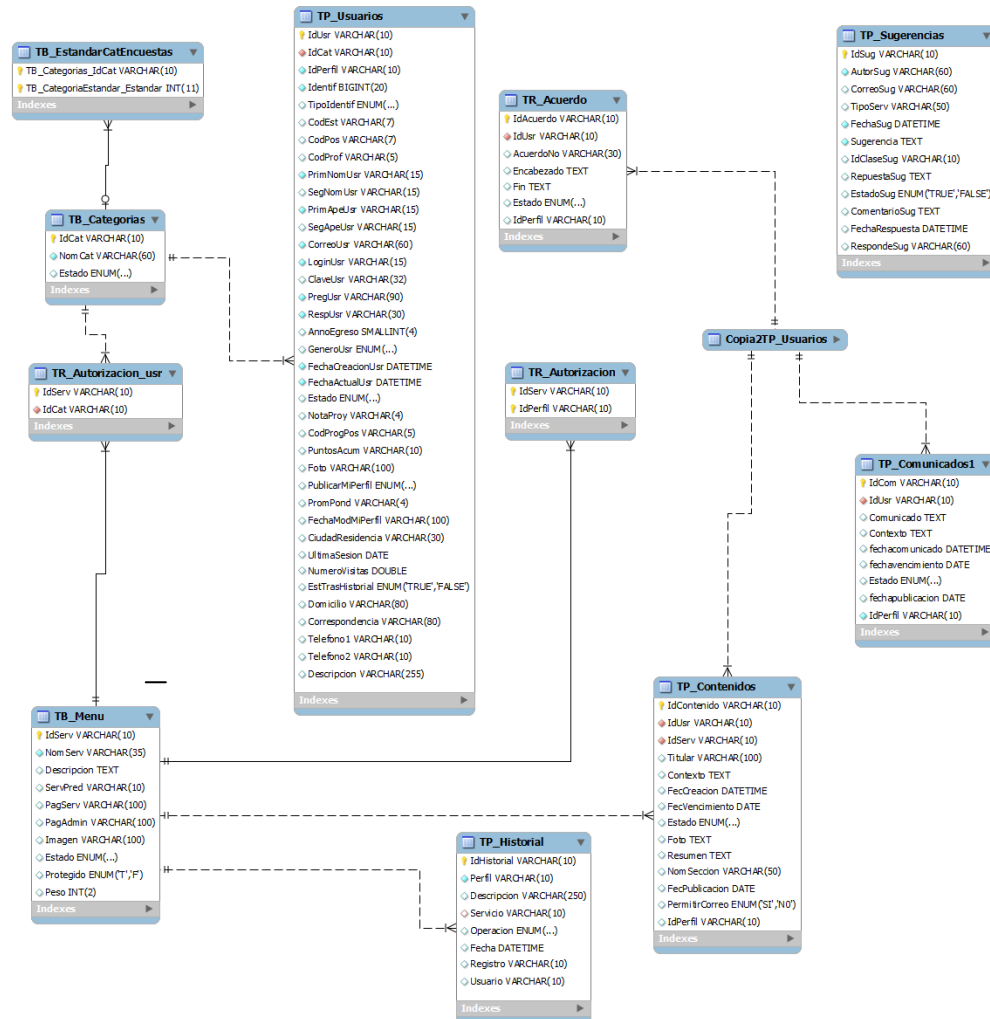
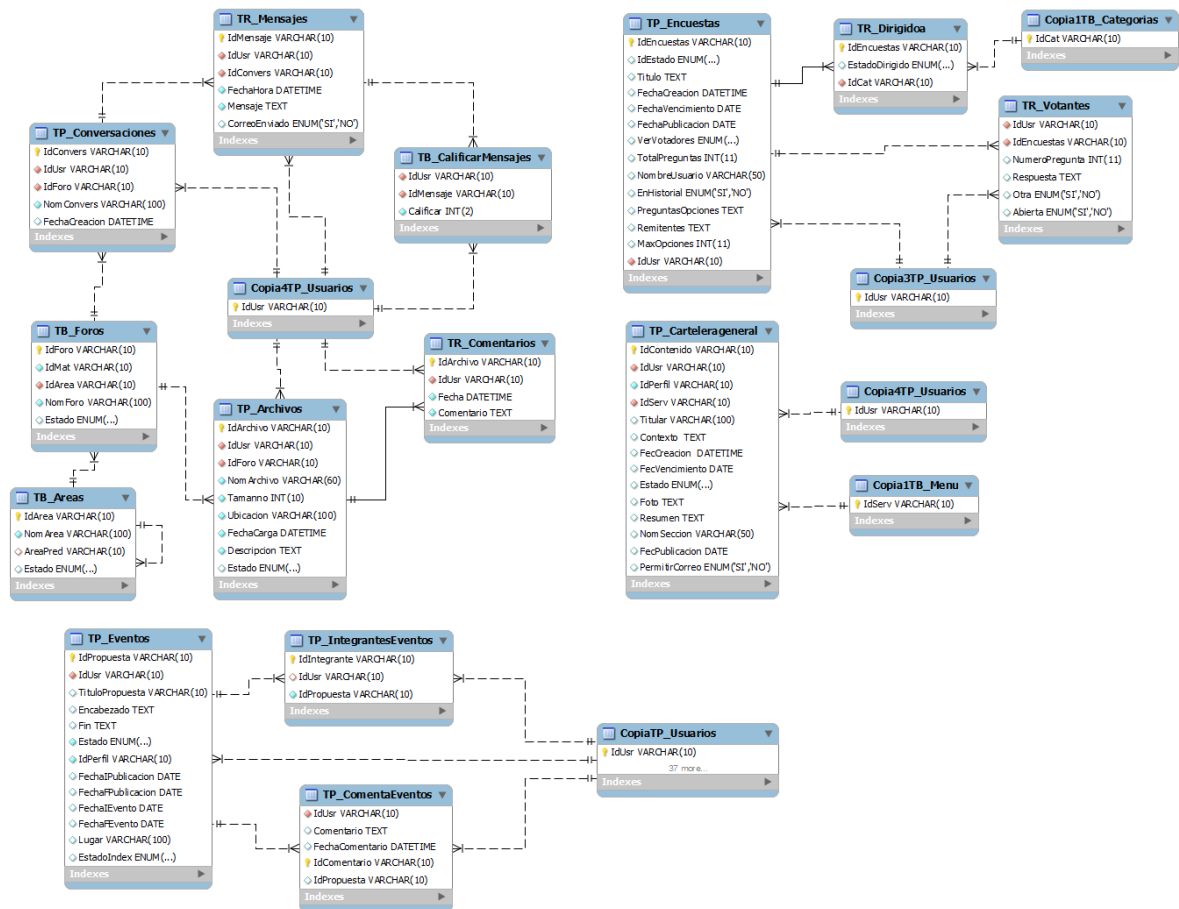


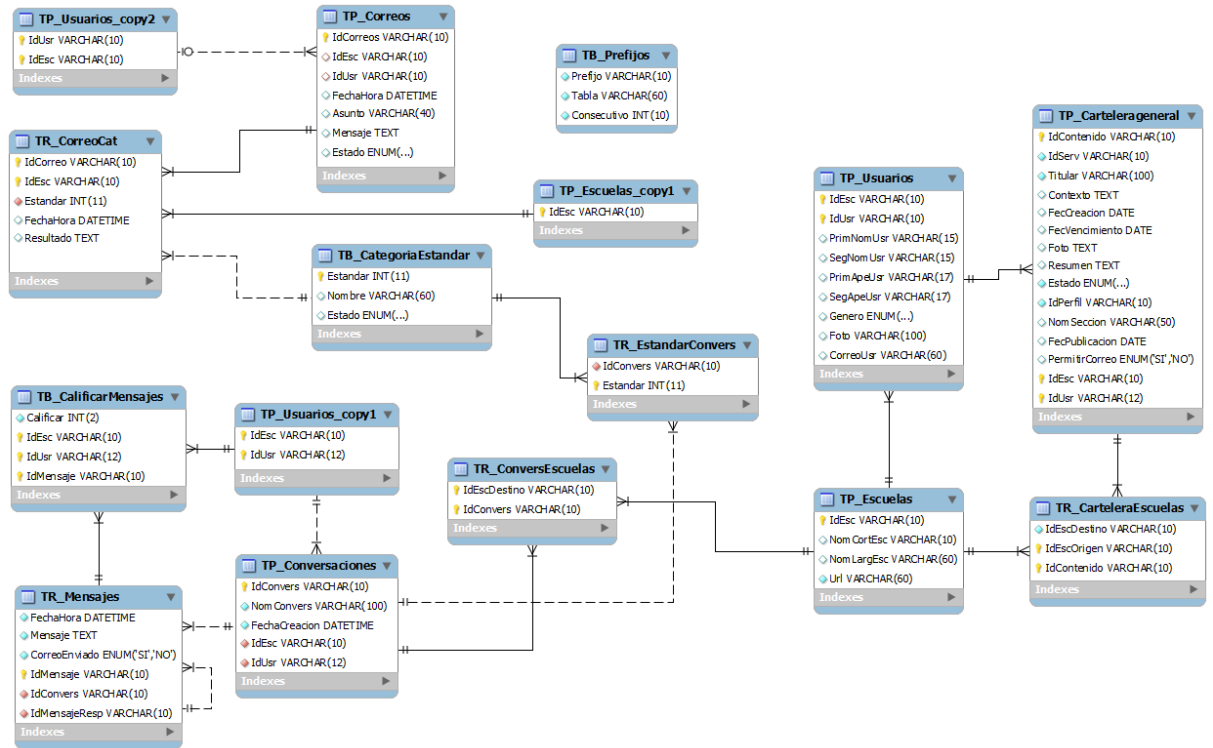
Figura 189. Continuación Diagrama E/R: Depurar Tablas:



Fuente: Autores.

- **Servicio Portal del Decano.**

Figura 19. Diagrama E/R: Portal del Decano:



Fuente: Autores.

4.1.3.2 Descripción de las Entidades.

Definición de las entidades utilizadas y creadas en la base de datos Poseidon Ubicada en el servidor de la escuela de Ingeniería Civil, EICWeb y de la base de datos Diamante ubicada en el servidor de los portales de las escuelas EEIEWeb, EIPWeb, EDIWeb.

Tabla 7. Descripción de las Entidades

ENTIDAD	DESCRIPCIÓN
TB_ActividadesClase	Contiene las actividades de clase con su título, descripción, fechas de inicio y final.
TB_TipoPlan	Contiene los tipos de actividad trabajos, laboratorios, exposiciones, foros, talleres y exámenes.
TB_PlanActClase	Especifica si es un trabajo, foro, laboratorio, exposición asociados con el plan de actividades.
TB_PlanTalleres	Tabla que contiene el nombre del plan y especifica si es un taller o un examen, y una descripción.
TB_Grupos	Contiene los grupos de investigación, extensión, culturales y académicos de las escuelas.
TB_CategoriaEstandar	Se encuentra el número de identificación, el nombre de las categorías estándar de los usuarios y el estado en que se encuentra dicha categoría ya sea activo, anulado e inactivo.
TP_Talleres	Aquí se guarda el identificador, nombre, las semanas, la descripción y el archivo de todas las actividades de tipo taller y examen asociados a un plan.
TP_GuionClase	Esta tabla contiene los guiones de clase con su respectivo nombre, dueño, el objeto virtual de aprendizaje asociado, la fecha de la última semana activa y el estado.
TP_Banco_Preguntas	Esta tabla guarda todas las preguntas que los

	profesores suben al sistema.
TP_Banco_Respuesta	Tabla que contiene todas las respuestas de las preguntas que se han subido al sistema.
TP_Ova	Esta tabla guarda los objetos virtuales de aprendizaje que han creado los profesores.
TP_UnidadOva	Tabla que contiene las unidades y los temas que crean los profesores para sus asignaturas.
TP_RecurosOva	Aquí se guarda la ruta de los archivos que se suben al servidor para cada tema de las asignaturas.
TR_CalifTalleres_Examenes	Acá se guardan todas las respuestas de los estudiantes para las actividades de talleres y exámenes y la respectiva nota que el profesor pone.
TR_GuionExamenGrupo	Contiene la hora, el grupo, la materia, el examen, los minutos a dedicar, el tipo de trabajo (individual o grupal) y el guion de clase.
TR_GuionTalleres	Esta tabla guarda los porcentajes de talleres y exámenes.
TR_TalleresPreguntas	Contiene la las preguntas que se han asociado a los talleres y exámenes de un guion.
TR_CalifNotaxActividad	La tabla contiene la nota de las actividades de un plan.
TR_NotaTallerExamen	Guarda la nota que el sistema construye para los talleres y exámenes y la que el profesor asigna.
TR_FechasTalleres	Contiene la fecha que el profesor le pone a los talleres y exámenes.
TR_Alumnos	Relaciona los estudiantes con la materia y el

	grupo de clase al que pertenecen.
TR_FechaActividades	Contiene la fecha que el profesor le pone a las actividades.
TR_CalifActNotaxPlan	Guarda la nota que el profesor le pone a los planes que tienen única nota.
TR_AutorizaOvaEst	Relaciona las categorías estándar que tienen acceso a los objetos virtuales de aprendizaje
TR_ColaboradoresOva	Relaciona los grupos de las escuelas que tienen permiso de editar objetos virtuales de aprendizaje.
TP_TemasPropuestos	Contiene los temas propuestos para cada materia.
TP_Usuarios	Contienen todos los usuarios que han sido cargados a la base de datos Poseidón desde las diferentes bases de datos Diamante.
TP_Usuarios	Contiene la información de todos los usuarios en la base de datos Diamante de cada escuela.
TR_GrupClaseEst	Relaciona a los usuarios con los grupos de clase.
TR_GruposClase	Guarda el registro de los subgrupos de clase para cada aula virtual.
TB_Areas	Se guardan las áreas por las cuales se clasifican los foros.
TB_CalificarMensajes	Guarda la calificación que asigna un usuario a un mensaje.
TB_Categorias	Contiene las categorías de los usuarios en el portal y el estado de las mismas.
TB_EstandarCatEncuestas	
TB_Foros	Contiene el nombre de los foros y las áreas a

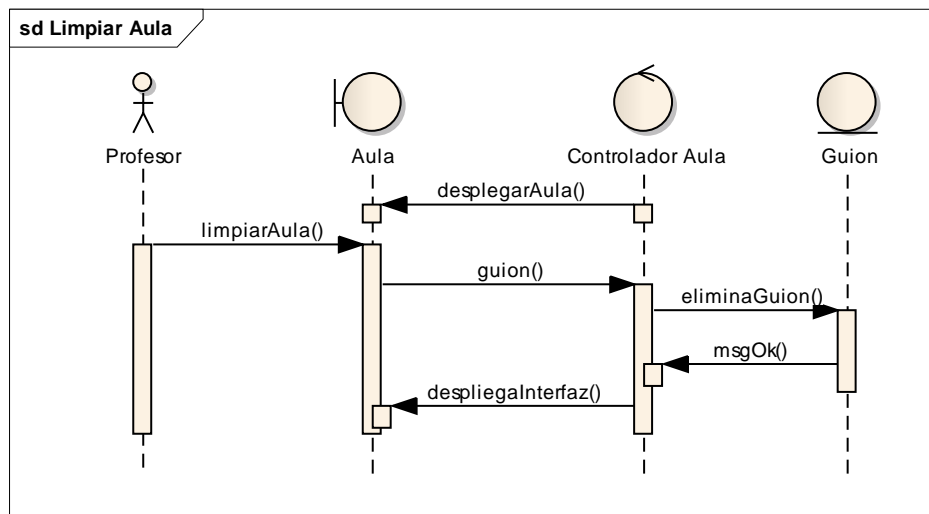
	las que va dirigida.
TB_Menu	Contiene toda la información de los menus de los portales.
TP_Archivos	Contiene la ubicación de los archivos que son subidos a los portales.
TP_CarteleraGeneral	Contiene las publicaciones que son mostradas en cartelera general de los portales de las escuelas.
TP_ComentaEventos	Contiene los comentarios a las propuestas de eventos.
TP_Comunicados1	Contiene los comunicados a la comunidad.
TP_Contenidos	Se almacena toda la información estática del portal.
TP_Conversaciones	Contiene las conversaciones de los foros.
TP_Encuestas	Contiene las encuestas realizadas en los portales.
TP_Eventos	Contiene la información de los eventos programados en los portales
TP_Historial	Mantiene el registro de todo el historial de las acciones llevadas a cabo en los portales.
TP_IntegrantesEventos	Personas que se han unido a un evento.
TP_Sugerencias	Contiene las sugerencias que los usuarios envían a través de los portales del grupo.
TR_Acuerdo	Contiene Información de los calendarios académicos y demás calendarios creados.
TR_Autorizacion	Contiene la relación de los servicios y los perfiles autorizados para esos servicios
TR_Autorizacion_Usr	Contiene la relación de los servicios y las

	categorías autorizadas para esos servicios.
TR_Comentarios	Contiene los comentarios que los usuarios agregan a los archivos
TR_DirigidoA	Relaciona las encuestas con las categorías a las que va dirigidas las encuestas.
TR_Mensajes	Relaciona los mensajes con los usuarios y la conversación donde se originó el mensaje.
TR_Votantes	Relaciona las respuestas de los usuarios a las encuestas realizadas
TB_CalificarMensajes	Guarda la calificación de los mensajes.
TB_CategoriaEstandar	Contiene las categorías estandarizadas en los portales.
TB_Prefijos	Contiene los prefijos de las tablas junto con un consecutivo para cada tabla.
TP_Escuelas	Contiene la información de los portales de las escuelas.
TR_ConversEscuelas	Relaciona la escuela destinataria con la conversación respectiva.
TR_EstandarConvers	Relaciona las conversaciones con las categorías estándar a las que fue dirigida la conversación.
TR_Mensajes	Relaciona los foros con los mensajes y participaciones de usuarios.
TP_Correos	Contiene la información de los correos a enviar.
TP_CarteleraEscuelas	Relaciona la escuela de origen y destino con el contenido de la publicación.
TR_CorreoCat	Relaciona el estado del correo con las escuelas y categorías a las que va dirigido.

Fuente: Autores.

4.1.4 Modelo de Procesos del Sistema

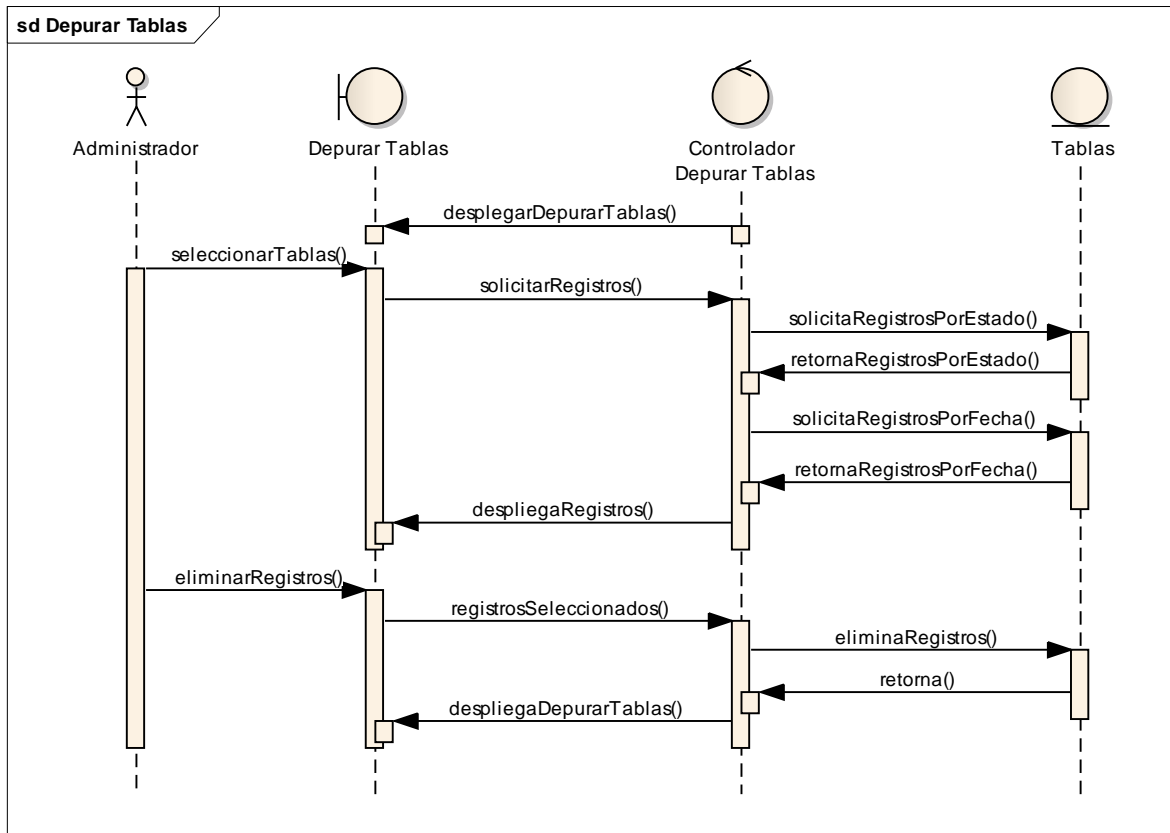
Figura 20. Diagrama de secuencia: Limpiar Aula.



Fuente: autores.

1. Una vez el Profesor ingresa al servicio de Aula Virtual el controlador aula despliega el aula.
2. El profesor selecciona el guion (ahora llamado aula) a eliminar, el controlador aula recibe la solicitud, elimina los archivos, respuestas y demás actividades realizadas por estudiantes en el aula.
3. Una vez que son eliminados los datos el controlador aula despliega nuevamente la interfaz.

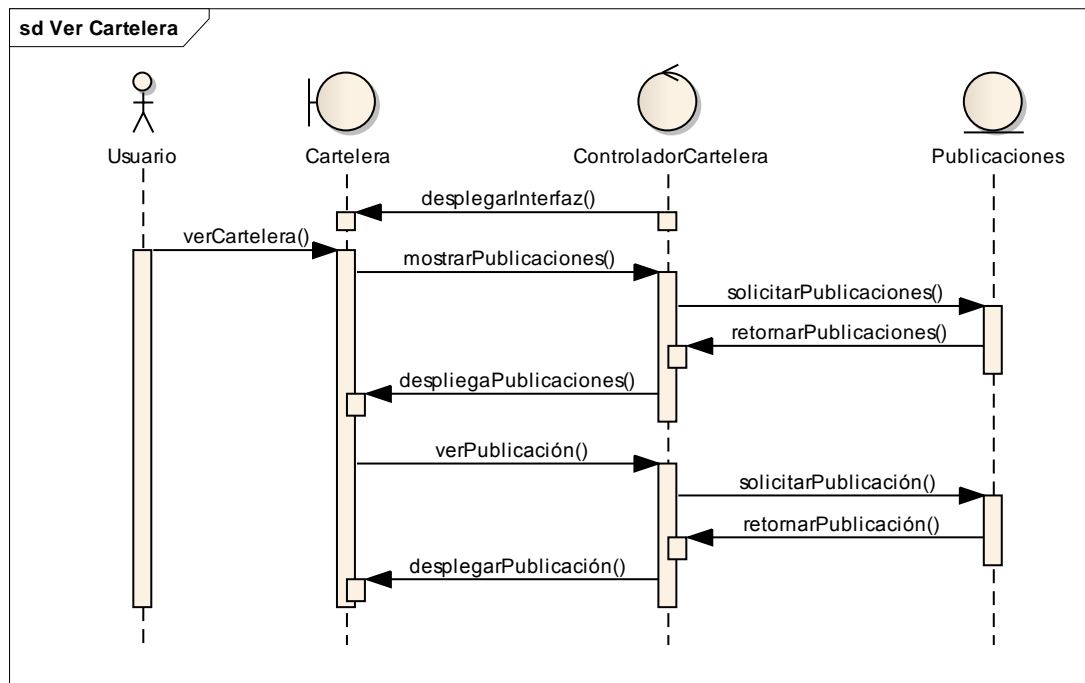
Figura 22. Diagrama de secuencia: Depurar Tablas.



Fuente: autores.

1. Una vez el administrador ingresa al servicio el controlador depurar tablas despliega la interfaz de depurar tablas.
2. El administrador elige la tabla que desee depurar, y el controlador recibe los registros de las tablas y despliega la interfaz con la lista de registros ya sea por fecha o estado según sea solicitado por el administrador.
3. Una vez que los registros son listados el administrador puede proceder a eliminarlos seleccionando la cantidad de registros que desee eliminar.
4. El controlador recibe los registros seleccionados, elimina los registros de la base de datos y despliega nuevamente la interfaz.

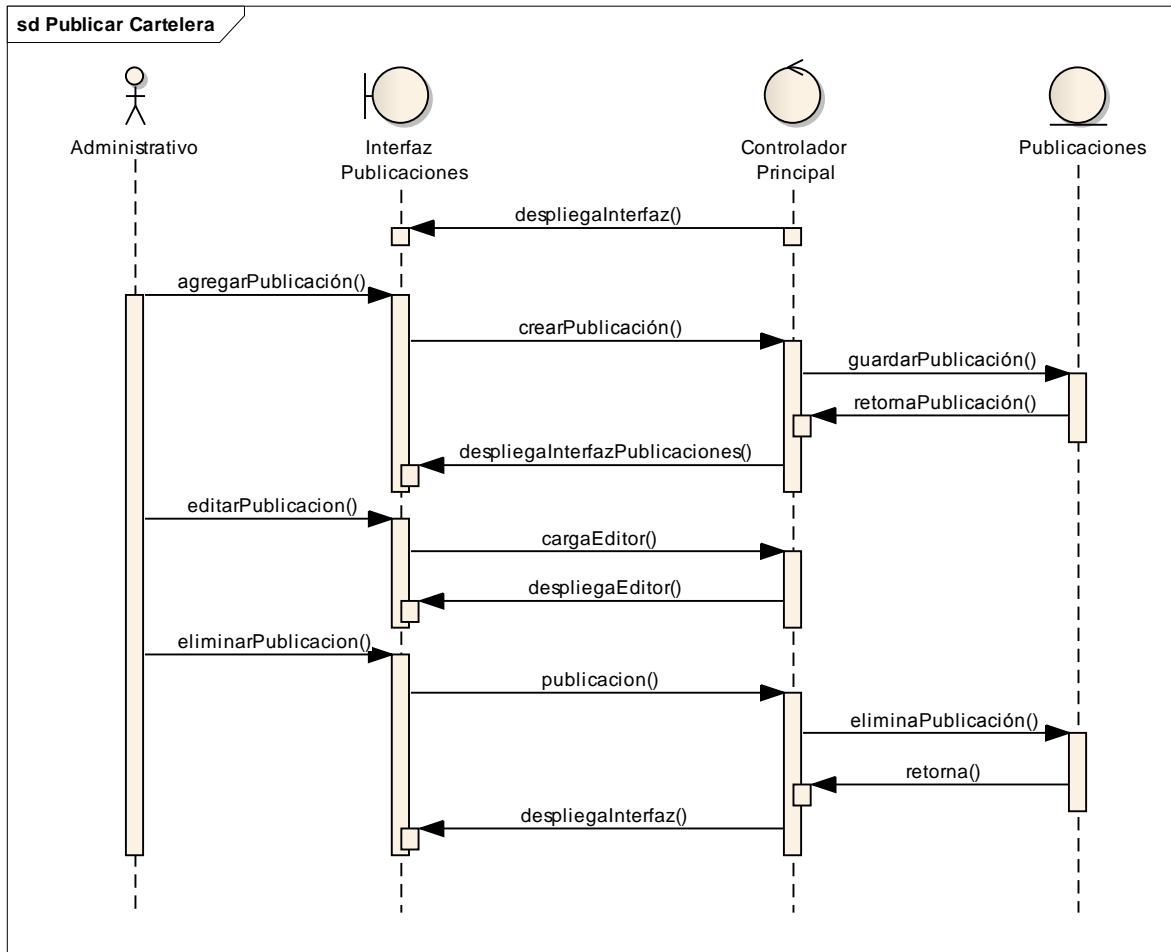
Figura 21. Diagrama de secuencia: Ver Cartelera.



Fuente: Autores

1. Una vez el usuario ingresa a ver cartelera el controlador de cartelera despliega una interfaz con una lista de las publicaciones en línea.
2. En esta interfaz el usuario puede ver una publicación, para esto el controlador de cartelera solicita la información de la publicación seleccionada y la despliega en una nueva interfaz.

Figura 22. Diagrama de secuencia: Publicar Cartelera.

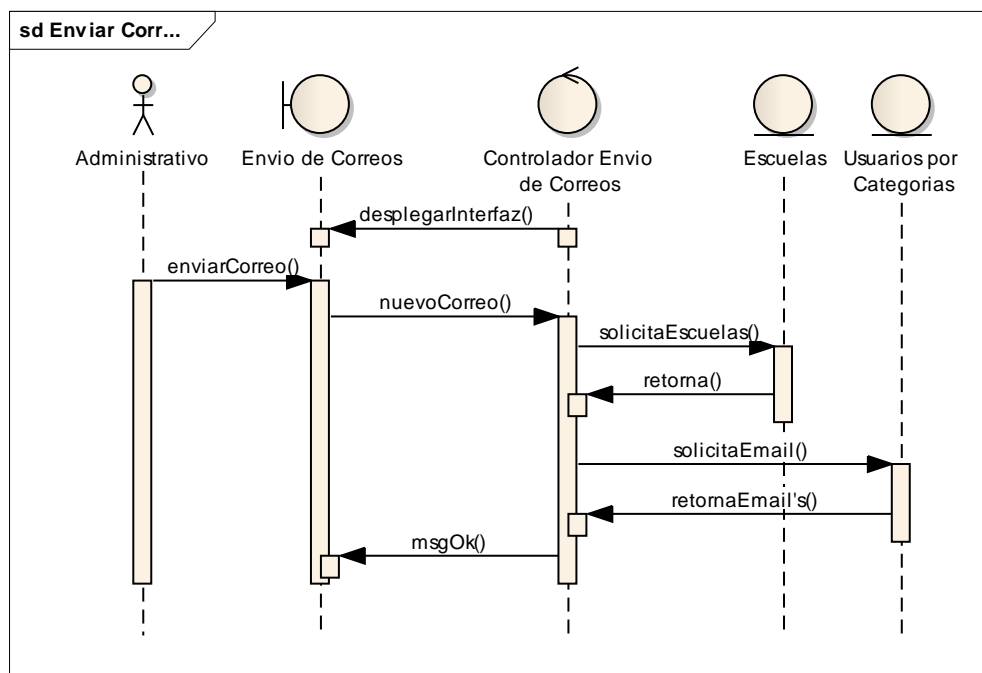


Fuente: Autores.

1. Cuando un administrativo de la facultad ingresa por publicar cartelera el controlador principal despliega una lista con las publicaciones en el portal actual así como las publicaciones en otras escuelas en línea o no en línea.
2. Una vez en esta interfaz el administrativo puede agregar una nueva publicación, para esto debe seleccionar la fecha de vencimiento de la publicación y las escuelas a las que va dirigida la publicación, la información de esta publicación será recibida por el controlador principal que solicitara guardarla en base de datos y retorna la publicación.
3. El usuario en esta interfaz podrá ver todas las publicaciones, en este caso el controlador solicitara una nueva interfaz y desplegara las publicaciones.

4. En esta interfaz el usuario tendrá la opción de editar o eliminar publicaciones de su autoría.
5. Si se decide editar una publicación el controlador del programa recibirá la información modificada y solicita actualizar la base de datos, a su vez retornara todas las publicaciones acompañado de un mensaje de confirmación.
6. Por el contrario si se decide eliminar una publicación, el controlador recibe la publicación a eliminar y solicita borrar los registros de las bases de datos, retornando un mensaje de confirmación.

Figura 23. Diagrama de secuencia: Correos.

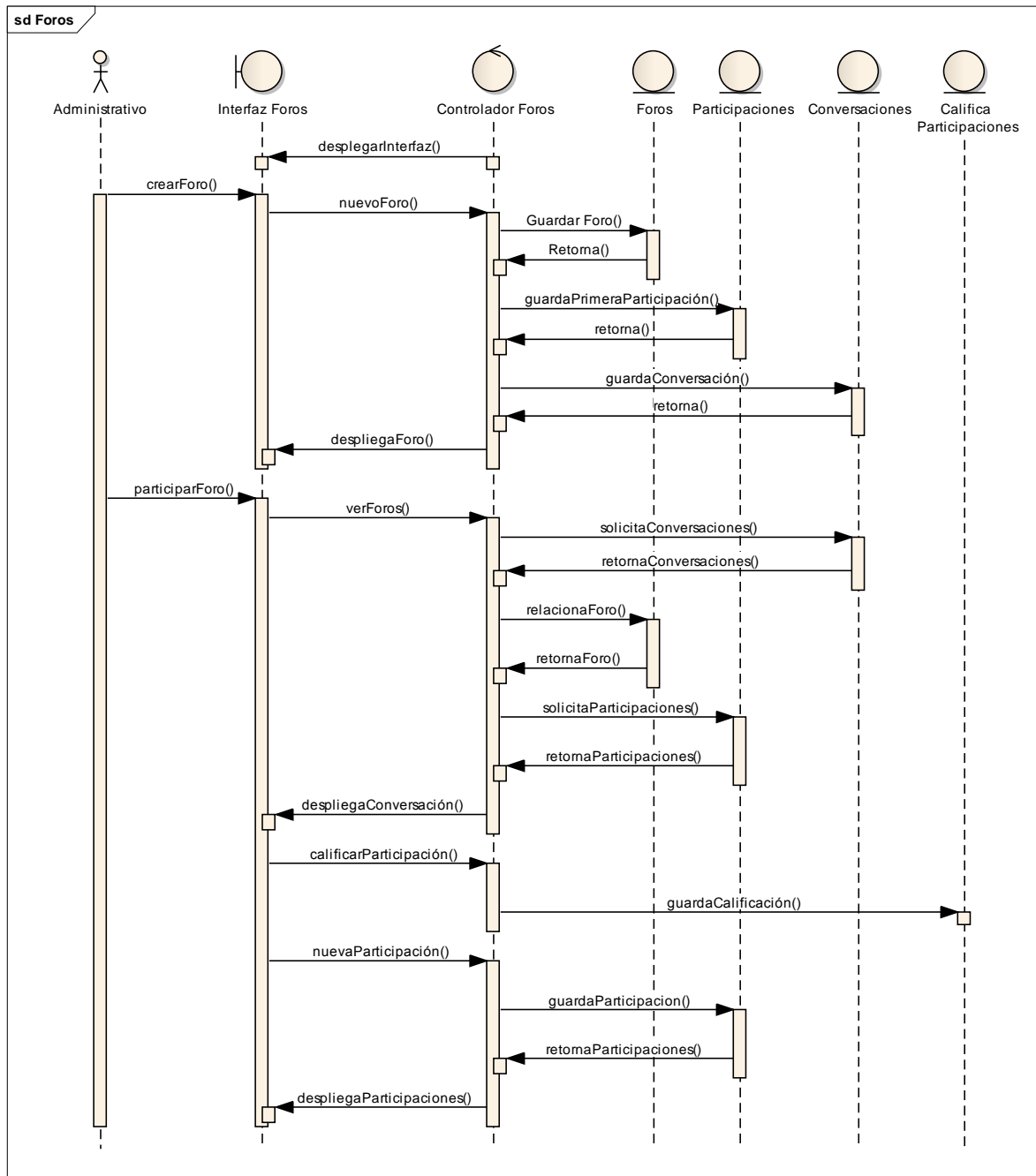


Fuente: Autores.

1. Cuando un usuario administrativo del portal del decano decide enviar un correo el controlador envió de correo despliega la interfaz para enviar correos junto con las escuelas y las categorías a las que puede enviar correo.

2. En esta interfaz el usuario selecciona las escuelas y la categoría a la que va dirigido el correo junto con el respectivo asunto y mensaje, lo cual recibe el controlador de envío de correos, el cual solicita las escuelas y los email's de los usuarios según la categoría, se guarda en base de datos y se despliega un mensaje al usuario de confirmación de correo programado y luego se procede a enviar el correo.

Figura 26. Diagrama de secuencia: Foros.



Fuente: Autores.

1. Una vez el administrativo ingresa al servicio de foros comunidad uis el controlador despliega la interfaz con los foros comunidad uis y foros pertenecientes al usuario.
2. En esta interfaz el usuario puede crear un nuevo foro, para lo cual el controlador recibe los datos y solicita guardar en base de datos la conversación inicial, el foro la participación, las escuelas y la categoría, después de esto se despliega una nueva interfaz con los foros vigentes y el foro creado.
3. El usuario puede participar tanto en los foros vigentes así como en los propios, para esto el controlador foros solicita la conversación con las participaciones y las despliega en una interfaz.
4. En esta interfaz el usuario puede calificar una participación, en este caso el controlador foros recibe el puntaje dado y pide guardar la calificación en base de datos, este a su vez retorna un promedio de las calificaciones dadas a dicha participación y la despliega en la interfaz.
5. Además el usuario, al participar en el foro, llena los datos que son recibidos por el controlador de foros que solicita guardar la nueva participación en la base de datos.

4.1.5 Implementación, Implantación y Pruebas Generales.

Para la implementación de los prototipos se utilizaron las siguientes herramientas:

- Lenguaje de programación orientado a la web, JSP.
- Hojas de estilo en cascada, CSS.
- Lenguaje Java.
- NetBeans, IDE para desarrollar las clases de Java y los archivos JSP y CSS.
- Servidor Jakarta Tomcat.

- Bloc de notas para desarrollar con HTML.
- Manejador de Base de datos, MySQL 5.0.

Con estas herramientas de programación y el sistema gestor de base de datos, se codificaron las páginas y se estructuraron los datos que se habían definido para el primer prototipo. Se tomaron en cuenta las sugerencias de los usuarios encargados de hacer seguimiento y que utilizaban las interfaces (Director del proyecto, Grupo Calumet, desarrolladores) y posteriormente se efectuó refinamiento de las interfaces.

Para el diseño realizado, se trabajó con la base de datos “Diamante” que ya existía y estaba implantada en el Servidor Cormorán. Según el diseño realizado para el primer prototipo, se modificaron algunas tablas y se crearon otras; también se trabajó siguiendo la estructura de directorios mencionada anteriormente. Actualmente el tamaño del portal es de 490 MB.

Las pruebas se llevaron a cabo en cada subsistema propuesto, verificando que las validaciones realizadas respondieran a lo dispuesto, de esta manera, se observó que la captura de datos, selección de ítems, almacenamiento de información y los contenidos de los datos eran validados correctamente, evitando que se incluyera información incorrecta en la base de datos.

Los servicios se implantaron uno a uno y se puso a disposición de los usuarios en el transcurso del desarrollo del proyecto, todos los usuarios del portal EISIWeb han hecho uso de todos los servicios nuevos; en este periodo se han corregido algunos detalles superficiales; pero en su totalidad el portal ha funcionado como se propuso. En el capítulo 6 se presentan las pruebas realizadas del sistema. Los servicios desarrollados para la Escuela de Ingeniería de Sistemas e Informática, luego de pasar ciertas pruebas serán implantados en los portales de las demás

escuelas donde el grupo Calumet también ejerce labores de desarrollo, mantenimiento, administración y soporte.

4.2 MANTENIMIENTO Y ADMINISTRACIÓN

4.2.1 Actividades de Mantenimiento.

Una vez empiece el funcionamiento del sistema como es de esperarse se presenta fallas en el sistema (incidencias) en algunas ocasiones, debido a errores en el código fuente o falta de pruebas exhaustivas. En consecuencia, una de las actividades del mantenimiento del sistema es corregir errores; los usuarios del sistema informan sobre la necesidad de nuevas funcionalidades y sobre los errores encontrados durante el uso. La función del administrador es mejorar el sistema y corregir las diferentes fallas que se puedan presentar.

- Colaborar a integrantes del grupo Calumet en la parametrización del sitio EISIWeb para que la implantación en otras escuelas se realizara de una manera más fácil.
- Colaborar constantemente en la adaptación de los servicios de EISIWeb al portal web de la Escuela de Estudios Industriales y Empresariales y de las demás Escuelas a las que presta sus servicios el grupo CALUMET.
- Dar solución a los problemas de código fuente que se puedan presentar, impidiendo el funcionamiento de los servicios debido a diferentes factores (parametrización, migración del sitio a versiones nuevas de Tomcat y Java), estos problemas deben ser solucionados inmediatamente luego de ser descubiertos.

- Implantar el sitio en otras escuelas. Modificar JavaBeans para filtrar la información descargada desde la División de Servicios de Información de acuerdo a la escuela.
- Realizar en el portal EISIWeb los enlaces a portales de otras escuelas que actualmente soporta el grupo Calumet.
- Revisar archivos de los portales web y borrar los que ya no eran utilizados.

INCIDENCIAS:

- **Correo al usuario al momento del registro con login y contraseña.**

Funcionalidad:

Al registrarse un usuario en el portal debe de enviarse un correo con su nombre de usuario y contraseña al correo inscrito por él.

Desarrollo del Problema:

El Servicio de registro permite a nuevos usuarios registrarse en el Portal web, que se acompaña de un correo electrónico con los datos del Usuario para verificar que se hizo un Ingreso satisfactorio y, además, como punto de ayuda para recordar los datos de Acceso al usuario. Por lo anterior, se gestionó que para todos los tipos de usuarios se envíe una copia de la Información del Registro.

Problemática:

No a todos los usuarios está llegando este correo, se debe de investigar la razón, y corregir si es necesario.

Descripción:

Se mejoró y eliminó líneas innecesarias en el archivo /eisi/Registro/registroChequeo.jsp para certificar que se envíe el Correo electrónico a todos los Usuarios independientemente de la Categoría.

- **Solicitud de datos al registro no parametrizado.**

Funcionalidad:

Al momento de registro se solicita a los usuarios una serie de datos que permiten identificar al usuario. Estas solicitudes se realizan acompañadas de textos descriptivos.

Desarrollo del Problema:

En el servicio de registro del portal se procedió a observar el problema planteado, pero al entrar a revisar el código, el nombre corto de la escuela ya se encontraba parametrizado.

Problemática:

En la página de registro dice: "Llene el mayor número de campos que conozca para ser identificado como usuario activo de la EISI.". Debe de parametrizarse EISI.

Descripción:

Se verifico que el parámetro encontrado correspondiera con el nombre corto de la escuela parámetro [3] y no se realizó ningún cambio.

- **Nombre de usuario no puede contener palabras reservadas.**

Funcionalidad:

Al momento de ingreso del nombre de usuario, y de textos en general, se debe comprobar que este (el nombre de usuario) no contenga palabras reservadas.

Desarrollo del Problema:

Se revisaron los jsp's de registro de usuario, se usó el bean que contiene las palabras reservadas y se restringió la creación de nombres de usuario con las palabras reservadas contenidas en el bean.

Problemática:

Revisar el nombre de usuario al realizar el registro para que evitar se digiten palabras reservadas, existe un beans que contiene estas palabras. Muchos usuarios se registran y después no pueden entrar por tener palabras reservadas. Además revisar que no se permitan nombres de usuario repetidos.

Descripción:

Se agregaron las líneas necesarias en /eisi/Registro/RegistroValidacionS.jsp y en /eisi/Registro/registroCrearCuentaS.jsp para restringir la creación de usuarios con palabras reservadas.

- Impresión del acta por parte del admin de trabajos de grado es necesario revisar.

Funcionalidad:

El administrador del sistema cuenta con la opción de imprimir cualquier acta del comité de trabajos de grado.

Desarrollo del Problema:

Cuando se daba clic en vista para impresión solo se imprimía el título anterior y el objetivo anterior, se modificó para que imprimiera el título y objetivos nuevos en las actas.

Problemática:

Se debe revisar esta opción ya que no ha sido actualizada después de los últimos cambios al acta en la parte de solicitudes, donde debe salir el dato viejo y el nuevo del cambio aprobado.

Descripción:

Se agregaron las líneas necesarias al archivo `/eisi/TrabajoGrado/Consultas/ConsultarActasVistaImpresion.jsp` para imprimir correctamente el acta.

- **Fecha de creación de usuarios por parte del admin (manuales).**

Funcionalidad:

En el Portal es constante la necesidad de crear usuarios que no son profesores, ni estudiantes, ni egresados. Normalmente ellos son administrativos que requieren de los servicios. El portal cuenta con estos servicios para los administradores.

Desarrollo del Problema:

Se hicieron pruebas creando usuarios manuales de diferentes categorías, se revisó el código de los archivos `adminCrearUsuarios.jsp` y del bean `Usuarios.java` la función `adminCrearUsuarioManual` y se verificó que estaba agregando la fecha de creación correctamente.

Problemática:

Al crear un usuario manual no se está llenando el campo de fecha de creación de usuario, este está quedando con ceros. Es necesario que este dato quede registrado.

Descripción:

Se verifico que al crear los usuarios manuales guardara en base de datos la fecha de creación correctamente, no se realizó ningún cambio.

- **Cambio de parámetros en Mi Perfil.**

Funcionalidad:

Mi Perfil permite la actualización de información más relevante de cada usuario como por ejemplo experiencia laboral, campos de desempeño y demás.

Desarrollo del Problema:

Se hicieron los cambios necesarios, se dejó parametrizado para que se pueda usar en cada escuela,

Problemática:

Mi Perfil cambiar mis aplicaciones por mis proyectos y logros. Y cambiar textos para dejarlos genéricos a todas las escuelas.

Descripción:

Tan solo se cambiaron algunas palabras en los jsp que se necesitaban y se dejaron de manera genérica para que se pueda usar en cualquier escuela.

- **Textos en evaluación de temas por parte del comité de trab de grado.**

Funcionalidad:

Los textos que acompañan la evaluación de temas hacen más sencilla la labor del comité de trabajos de grado.

Desarrollo del Problema:

Se hicieron los cambios respectivos y se dejó parametrizado el nombre largo de la escuela para que se pueda usar en cualquier escuela sin ningún inconveniente.

Problemática:

En Comité "Evaluación Temas" aparece esta frase: "En el siguiente listado se muestran... Esc de Ing. de Sistemas". Se debe parametrizar y corregir el error de ortografía.

Descripción:

Se hicieron los cambios respectivos y se dejó parametrizado el nombre largo de la escuela para que se pueda usar en cualquier escuela sin ningún inconveniente.

- **Textos explicativos en creación de Mi Perfil.**

Funcionalidad:

Al crear la hoja de vida, por parte de los usuarios, parecen una serie de textos explicativos que hacen la labor más sencilla.

Desarrollo del Problema:

Ya se agregaron los textos explicativos en donde se hacían necesario.

Problemática:

Actualmente los textos están orientados a la carrera de Ing. de sistema, se deben cambiar los textos para que sean genéricos a todas las escuelas.

Descripción:

Se agregaron los textos explicativos en donde se hacían necesarios para mejorar el entendimiento del usuario y se dejaron genéricos para todas las escuelas.

- **Correo del autor que avala la participación, al autor que digitó el tema.**

Funcionalidad:

En el sistema de trabajos de grado es necesario que cada autor propuesto, por el autor que digita la propuesta, exprese de manera explícita su deseo de ingresar a este proyecto. Mientras la totalidad de los autores no den este aval la propuesta no pasa al director propuesto.

Desarrollo del Problema:

En el sistema de trabajos de grado es necesario que cada autor propuesto, por el autor que digita la propuesta, exprese de manera explícita su deseo de ingresar a este proyecto. Mientras la totalidad de los autores no den este aval la propuesta no pasa al director propuesto.

Problemática:

En este momento al digitar la propuesta un correo avisa a los autores propuestos sobre la posibilidad de unirse al proyecto. Se hace necesario ahora que al dar aval un autor, para aceptar la participación, se requiere de correo que avise al autor que digitó la propuesta de esto. De esta forma, el autor que digitó podrá estar al tanto, y podrá saber en qué momento todos los propuestos avalaron, para de esta forma enviarla al director propuesto.

Descripción:

Se realizaron los cambios oportunos dando pronta respuesta a los correos electrónicos que deben llegarle a los interesados (autores) del proyecto. Junto con su respectivo mensaje, los cambios se hicieron en el respectivo jsp agregando el correo de los autores.

- **Claridad de los datos solicitados sobre programas en el registro.**

Funcionalidad:

Al registro se solicitan los datos de programa de posgrado al cual puede pertenecer un usuario. Esta solicitud está acompañada de texto explicativo que hace la labor más sencilla al usuario.

Desarrollo del Problema:

Se hicieron los arreglos necesarios para que las personas puedan llenar mejor el formulario de cuando se está registrando

Problemática:

En el registro de usuarios, en la caja de posgrados debe de quedar claro que el primer código que se pide es de estudiante de posgrado, y el segundo código del programa. En este momento dice Código de Posgrado y Código del programa.

Descripción:

Se hicieron los arreglos necesarios en el jsp para que las personas puedan llenar mejor el formulario de cuando se está registrando.

4.2.2 Actividades de Soporte a Usuarios.

Mantener una comunicación abierta con los usuarios y directivos es de gran importancia para analizar y evaluar las percepciones con respecto al sistema. Esta forma de participación con los usuarios durante el soporte aumenta el grado de confianza y credibilidad. En esta actividad, los usuarios del sistema informan sobre los problemas en la utilización del mismo y los encargados del soporte responden con: cambios en los procedimientos de operación, información adicional y mejoramiento en el servicio. Algunas de las labores realizadas comúnmente son:

- Restablecimiento de contraseña a usuarios.
- Dar capacitación sobre manejo del sitio, creación de servicios, labores de administración a los nuevos integrantes del grupo CALUMET.
- Aclarar dudas a los usuarios sobre la utilización de algunos servicios.
- Dar respuesta y solución a las inquietudes y problemas que tengan los usuarios de los portales EEIEWeb, EIPWeb y EDIWeb.
- Facilitar orientación a los estudiantes nuevos de las escuelas a las cuales presta sus servicios el grupo CALUMET, en cuanto al registro en el portal WEB y el uso de sus principales servicios.

4.2.3 Actividades de Administración.

Es necesario asignar personas encargadas de los portales web, para su administración, debido a que las actividades e información deben ser manejadas de forma segura por el personal autorizado. Por esta razón son asignadas funciones especiales al administrador de cada portal y es el único que tiene

acceso a la información, además de ser el encargado de asignar permisos a los usuarios del sistema.

A los administradores les corresponde tareas como; habilitar nuevos servicios, cambiar servicios, actualizar bases de datos, realizar copias de seguridad con el fin de mantener en funcionamiento un sistema actualizado y seguro.

- Realizar periódicamente copias de las bases de datos Diamante y División.
- Dar aval a las solicitudes de publicación de los usuarios en cartelera web para ser vista por la comunidad.
- Atender las sugerencias hechas por los usuarios del sistema a través del servicio correspondiente.
- Actualizar periódicamente las bases de datos con respecto a la información que ofrece la División de Servicios de Información de la Universidad Industrial de Santander, para que el portal cuente con información actualizada.

5 PRUEBAS DEL SISTEMA

Para garantizar el óptimo desarrollo del sistema realizado, descartando errores de diseño, programación en el producto final por lo tanto a continuación se presentan las pruebas tenidas en cuenta y aplicadas para su comprobación:

5.1 PRUEBAS DE VERIFICACIÓN

Esta prueba es una de las más prácticas, seguras y utilizadas, puesto que esta modalidad se realiza obteniendo información de la Base de Datos y comprobando con las actividades realizadas utilizando los respectivos servicios desarrollados, haciendo cierta clase de verificaciones como por ejemplo verificar que los campos que contienen datos obligatorios no queden vacíos, es decir la respectivas validaciones que dependen del tipo de dato utilizado y la estabilidad del portal al ocurrir algún evento inesperado e incierto.

En las siguientes tablas se relaciona y describe las pruebas de cada caso de uso de los servicios que fueron desarrollados y su estado.

5.1.1 Pruebas por componente.

Esta clase de prueba se realizó para cada caso de uso de cada servicio desarrollado y descrito anteriormente:

Tabla 8. Pruebas Realizadas: Limpiar Aula.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Acceder al Sistema	Ingresar al portal usando un nombre de usuario y contraseña.	✓
Ingresar al Aula	Ingresar al servicio “Acceder al Aula” con perfil de Docente y verificar que se despliegue correctamente la interfaz junto con el botón limpiar aula.	✓
Limpiar Aula	Una vez el profesor seleccione limpiar aula deben mostrarse ventanas emergentes notificándole al profesor que las acciones a realizar son irreversibles, además verificar en base de datos que todo sea borrado correctamente después de la acción del profesor.	✓

Fuente: Autores.

Tabla 9. Pruebas realizadas: Depurar Tablas

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.	✓
Acceder Depurar Tablas	Entrar con perfil de Administrador al servicio del portal depurar tablas en el cual se puede elegir la tabla a depurar.	✓
Seleccionar Tabla	Seleccionar una tabla a depurar, ya sea por estado o fecha y verificar que se listen correctamente los registros en base de datos.	✓

Listar Registros por Estado	Verificar que se listen los registros de la tabla seleccionada según estado anulado o inactivo.	✓
Listar Registros por Fecha	Verificar que se listen correctamente los registros de la tabla seleccionada según el rango de fecha seleccionada.	✓
Eliminar Registros	Verificar que se eliminen en cascada correctamente los registros de la base de datos los registros seleccionados, ya sea por estado o por fecha.	✓

Fuente: Autores.

Tabla 11. Pruebas realizadas: Ver Cartelera.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Ver Cartelera	Verificar que se Liste una breve información de todas las publicaciones vigentes de acuerdo a la fecha.	✓
Ver Publicación	Ver la información detallada de cada publicación, incluyendo el autor y la fecha de publicación y escuela de origen.	✓

Fuente: Autores.

Tabla 10. Pruebas realizadas: Publicar Cartelera.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.	✓
Acceder Publicar Cartelera	Ingresar al servicio Publicar con perfil administrativo de la facultad de fisicomecánicas.	✓
Agregar Publicación	Permite agregar una publicación a cartelera general desde el portal del decano a los portales de las escuelas pertenecientes a la facultad y a otras escuelas.	✓
Seleccionar Fecha	Se restringe la selección de fecha final a una fecha mayor o igual a la fecha actual.	✓
Seleccionar Escuela	Permite seleccionar una o más escuelas a la que se desee publicar, solo se despliegan la lista de escuelas cuando se ha seleccionado una fecha mayor o igual a la fecha actual.	✓
Eliminar Publicación	Permite eliminar una publicación de cartelera general al autor de la misma.	✓
Editar Publicación	Permite editar una publicación de cartelera general al autor dela misma.	✓

Fuente: Autores.

Tabla 13. Pruebas realizadas: Foros.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.	✓
Acceder Foros	Al Acceder al servicio de foros y seleccionar foros UIS se despliega la interfaz con los foros de la escuela, de otras escuelas y foros pertenecientes al usuario que ingresa.	✓
Crear Foro	Permite crear un nuevo foro dirigido a las escuelas y categorías disponibles.	✓
Participar Foro	Se puede participar en los foros de otras escuelas y del usuario pre-visualizando las participaciones anteriores.	✓
Calificar	Se puede asignar una calificación a una participación o a una respuesta de participación por medio de un puntaje visualizado por medio de estrellas.	✓
Responder Participación	Permite responder a una participación de un usuario en el foro.	✓

Fuente: Autores.

Tabla 14. Pruebas realizadas: Correos.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.	✓
Enviar Correo	Verificar en base de datos que se programe el envío de correo a las escuelas y usuarios de la categoría seleccionada correctamente, se verifica con correos de prueba que lleguen los correos a los destinatarios.	✓

Fuente: Autores.

5.2 PRUEBAS DE INTEGRACIÓN

Una vez creados todos los servicios se verificó que estos funcionarán correctamente y que no interfirieran con el buen funcionamiento del portal en general. Se verificó que las consultas a la base de datos se hicieran de manera correcta, que en el sistema se visualizaran correctamente los servicios y que cada tipo de usuario tuviera acceso a los servicios en relación a su perfil y categoría.

5.3 PRUEBAS DE VALIDACIÓN

Todos los servicios desarrollados y descritos anteriormente fueron sometidos a una serie de pruebas y verificaciones minuciosas de validaciones, realizadas con el fin de comprobar que correspondieran a los requerimientos establecidos al inicio del proyecto, de esta forma se observó el almacenamiento de información, la validación de los datos fuera correcta para evitar la inclusión incorrecta de información en la base de datos.

6 CONCLUSIONES

- El servicio complementario del aula virtual, limpiar aula permitió mejorar la experiencia de los profesores al usar el aula virtual y se vio reflejada en la reutilización de guiones de clase (ahora llamado aula de clase), por lo que es de vital importancia mejorar continuamente los servicios ofrecidos a los usuarios en los portales desarrollados por el grupo Calumet.
- El modelo de ciclo de vida del sistema es de vital importancia para realizar proyectos software. Para la realización de este proyecto se aplicó el modelo de prototipito evolutivo debido a que permite diseñar y construir las partes más importantes de la aplicación en un prototipo que constantemente es mostrado al cliente, de esta forma se enriquece el producto hasta obtener el prototipo final a gusto del cliente.
- Es importante hacer un seguimiento continuo del código que escriben los desarrolladores de software, con el fin de evitar mal entendidos con los requerimientos del sistema y poder revisar que cumplan con las políticas de escritura de código de la organización.
- Los controladores de versiones de software son fundamentales para trabajar con un número amplio de desarrolladores ya que podemos recuperar código fácilmente cuando se ha perdido o estropeado, controlar quien ha realizado cambios sobre el código y lo más importante facilita el trabajo en equipo.
- El servicio Depurar Tablas es de vital importancia ya que facilito la labor de los administradores de los portales al permitirles depurar los registros sin vigencia fácilmente ahorrándoles gran cantidad de tiempo y permitiéndoles enfocarse en otras labores de administración.

- El desarrollo del portal del decano al cual se puede acceder desde la siguiente url: fisicomecánicas.uis.edu.co era muy necesario desarrollarlo ya que la facultad de fisicomecánicas no contaba con ninguna forma de comunicarse con la comunidad de sus escuelas estando prácticamente aislado de la comunidad, con el desarrollo del portal la facultad puede comunicarse por diferentes medios con sus escuelas, con servicios desarrollados especialmente para este nuevo portal, tales como: publicar en cartelera, foros y envío de correos. Facilitando así la labor de la decanatura.

7 RECOMENDACIONES

- Extender los servicios desarrollados para el portal del decano a los portales soportados por el grupo calumet, para lograr interacción con otras escuelas.
- Delegar por completo las labores de administración de los portales a los integrantes que heredan dichas labores, actualmente el administrador solo tiene acceso a un usuario con perfil de administrador en el portal, limitando su labor, es necesario permitirle conexión a base de datos y servidor para realmente hacerse cargo del portal encargado.
- Desarrollar un servicio de consultas y sugerencias académicas referidas a los administradores y directivos de las escuelas, ya que el servicio actual de consultas y sugerencias soporta tanto las consultas académicas como las de funcionamiento del portal.
- Implementar el portal del decano en la facultad de ingenierías fisicoquímicas para integrar las escuelas de la facultad con la decanatura a través de los servicios ya desarrollados en el portal de la facultad de ingenierías fisicomecánicas.
- Integrar los portales con las redes sociales, tales como Facebook y Twitter, especialmente las publicaciones en cartelera general. Generando así mayor difusión de las noticias y aumentando el conocimiento y uso de los portales.

8 BIBLIOGRAFÍA

1. CÁRCAMO SEPÚLVEDA, José. Bases de Datos Relacionales: Un enfoque práctico de diseño. Universidad Industrial de Santander. Colombia, 1994. Se encuentra en este libro un guía muy práctica y fácil de entender al momento de trabajar con bases de datos.
2. JOHNSON, James. Bases de datos: Modelos lenguajes y diseño. 1ª ed. Oxford, 2000. Presenta temas de teoría de bases de datos: modelos y métodos de acceso, administración, diseño de aplicaciones.
3. PRESSMAN, R. Ingeniería de Software, un enfoque práctico. Quinta Edición. McGraw-Hill. 2002. En este libro se encuentra información sobre las metodologías de desarrollo software.
4. PIATTINI, Mario, CALVO-MANZANO, José A., CERVERA, Joaquín, FERNANDEZ, Luis. Análisis y diseño detallado de Aplicaciones Informáticas de Gestión. Alfaomega, 2000. Este libro contiene información sobre técnicas para el buen modelado de aplicaciones informáticas.
5. SCHMULLER, JOSEPH. Aprendiendo UML en 24 horas. 1ª ed. México: Alhambra Mexicana S.A, 2000. En este libro se encuentra una guía muy práctica que permite conocer y entender sobre UML.
6. STALLINGS, William. Sistemas Operativos: Aspectos Internos y Principios de Diseño. Madrid: Pearson Prentice Hall, 2005. Este libro se ocupa de los conceptos completos de las características de los sistemas operativos.

7. <http://manuales.dgsca.unam.mx/jsp>. Manuales básicos de JSP.
8. <http://www.aprender-informatica.com/>. (n.d). Tutoriales del lenguaje de programación JavaScript.
9. <http://www.htmlquick.com/es/tutorials.html>, tutoriales de cómo programar en html de gran ayuda en detalles básicos y sencillos.
10. http://www.milestone.com.mx/articulos/casos_a_incluir_casos_a_extender.htm. Artículo relacionado con los diagramas de casos de uso, definición, explicación y ejemplos
11. <http://www.mysql.com>. Sitio oficial de MySQL, que permite participar en capacitaciones en línea y tener las ultimas noticias acerca del producto para su mayor aprovechamiento.
12. <http://www.programacion.com/java/tutorial>. En este sitio se encuentran una gran variedad de tutoriales relacionados con: Los APIS, las herramientas y los servicios que proporciona los Servlets y las Java Server Pages (JSP), los tópicos necesarios para la programación de acceso a bases de datos en Java
13. http://www.programacionfacil.com/programacion:manual_java_jsp, manual de programación para Java Jsp (Java Server Pages).
14. <http://www.youblisher.com/p/155105-Aprendiendo-UML/>. Manual de conceptos básicos sobre UML.

ANEXOS

ANEXO A. MANUAL DE USUARIO

El manual de usuario el cual es un manual guía para que el usuario pueda acceder a los diferentes servicios creados, donde se explica su manejo, paso a paso.

Se puede encontrar en el menú de Ayudas en todos los portales del grupo, por ejemplo, en el nuevo portal del decano al cual se puede acceder desde el siguiente enlace:

<http://fisicomecanicas.uis.edu.co/eisi/Ayudas/index.jsp?IdServicio=S683>

Así como en el canal del grupo en YouTube:

<http://www.youtube.com/feed/UCIIYsDiHVug2FfZHICEEzEg/u>

Se toma como entorno de desarrollo, pruebas y presentación el portal de pruebas del grupo CALUWeb, para luego de rigurosas revisiones ser implementado en los portales EIMTWeb y EEIEWeb, como también en los demás portales que administra el grupo Calumet.