

**ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO,
DESARROLLO E IMPLANTACIÓN DE LOS SERVICIOS DE QUICES,
ENCUESTAS Y RECURSOS PARA LOS PORTALES DE LAS
ESCUELAS DE LA UIS.**

JORGE ARMANDO RUEDA ESTUPIÑAN

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2015**

**ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO,
DESARROLLO E IMPLANTACIÓN DE LOS SERVICIOS DE QUICES,
ENCUESTAS Y RECURSOS PARA LOS PORTALES DE LAS
ESCUELAS DE LA UIS.**

JORGE ARMANDO RUEDA ESTUPIÑAN

Trabajo de grado para optar al 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
2015**

AGRADECIMIENTOS

*Agradezco a **Dios** por haberme permitido llegar hasta aquí, por ayudarme en los momentos difíciles de esta etapa de mi vida y por darme la oportunidad de compartirlo con las personas que amo y que siempre han estado ahí para mí.*

*A mi **madre** por su apoyo incondicional, por inculcarme la importancia de estudiar, por su gran ejemplo de honestidad y trabajo, gracias a ella por su amor y ayuda en los momentos difíciles por comprenderme en todo momento, dándome las palabras que en ese momento fueron necesarias para continuar en busca de esta meta. Un sueño que ella siempre tuvo para mi vida, Gracias mamá por siempre creer en mí.*

*A mi **padre** por su ayuda en todo este tiempo, por su amor que siendo incondicional me trajo a este punto en mi vida, lograr un sueño que sé que también compartía conmigo, Gracias papá.*

*A mi **abuela**, ella con su amor incondicional y con sus consejos que me ayudaron en todo este proceso de crecimiento personal e intelectual, gracias a esto nunca desistí con este sueño.*

*A **Liz** porque desde el primer momento en la universidad fuiste mi apoyo, por todas esas mañanas y tardes de estudio, por motivarme a ser cada día mejor y a hacer las cosas con excelencia, por tenerme paciencia en todo momento y siempre tenerme como primera opción, mi eterna compañera de grupo, estaré por siempre agradecido contigo.*

*Al grupo **Calumet** por abrirme las puertas y sentir en él una familia, por enseñarme la importancia del trabajo en equipo y ayudarme a amar aún más mi profesión, por todas las cosas que aprendí y viví gracias.*

CONTENIDO

INTRODUCCIÓN.....	12
1 DEFINICIÓN DEL PROBLEMA	14
2 JUSTIFICACIÓN.....	16
3 OBJETIVOS.....	18
3.1 OBJETIVO GENERAL.....	18
3.2 OBJETIVOS ESPECÍFICOS.....	18
4. MARCO TEÓRICO.....	21
4.1 ARQUITECTURA CLIENTE/SERVIDOR	22
4.1.1 Arquitectura Cliente/Servidor aplicada.....	23
4.1.2 Ventajas del esquema Cliente/Servidor	23
4.1.3 Desventajas del esquema Cliente/Servidor	23
4.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS.	
4.2.1 Tecnología aplicada	24
4.2.2 Bases de datos.....	26
4.3.1 MySQL	26
4.4 NETBEANS.....	27
4.5 SISTEMA DE CONTROL DE VERSIONES	28
4.5.1 Subversión	29
4.6 PROGRAMACIÓN UTILIZADA	30
4.6.1 Clases	30
4.6.2 Objetos	30
4.6.4 Métodos	30
4.6.5 Herencia	31
4.6.6 Beneficios de la Programación Orientada a Objetos	31
4.6.7 Java y JDK (Java Development Kit)	32
4.7 SERVIDORES WEB	32

4.7.1 Servidor Jakarta Tomcat	33
5. DISEÑO METODOLÓGICO	34
5.2 LENGUAJE DE MODELADO UNIFICADO	35
5.2.1 Diagramas de UML	36
5.3 ESTÁNDARES DE PROGRAMACIÓN	37
5.3.1 Modelo de datos	37
5.3.2 Nombres de las tablas	38
5.3.3 Clases	38
5.3.4 Páginas JSP	38
5.3.5 Organización de Directorios.....	39
6 DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y	
MANTENIMIENTO	39
6.1 PROTOTIPO ESPERADO	39
6.1.1 Diagramas de Casos de Uso	48
6.1.2 Documentación de Casos de Uso del Sistema	50
6.1.3 Diseño y Análisis	56
6.1.4 Descripción de entidades.....	59
6.1.5 Modelo de Procesos del Sistema	60
6.1.6 Implementación, Implantación y Pruebas Generales	72
6.2 MANTENIMIENTO Y ADMINISTRACIÓN	73
6.2.1 Actividades de Mantenimiento	73
6.2.3 Actividades de Administración	75
7 PRUEBAS DEL SISTEMA	76
7.1 PRUEBAS DE VERIFICACIÓN	76
7.1.1 PRUEBAS POR COMPONENTE.....	76
7.2 PRUEBAS DE INTEGRACIÓN.....	83

7.3 PRUEBAS DE VALIDACIÓN	83
8 CONCLUSIONES	85
9 RECOMENDACIONES	85
BIBLIOGRAFÍA	87
ANEXOS	90

LISTA DE TABLAS

Tabla 1. Casos de uso: Gestionar Quiz.....	50
Tabla 2. Casos de uso: Gestión Quices Estudiante.....	51
Tabla 3. Caso de Uso: Gestionar Encuestas.....	51
Tabla 4. Casos de uso: Gestión Encuestas Estudiante.....	52
Tabla 5. Casos de uso: Gestión Foros Evaluables.....	52
Tabla 6. Casos de uso: Gestión Foros No Evaluables.....	53
Tabla 7. Caso de Uso: Gestión Foros Estudiantes Evaluables.....	53
Tabla 8. Caso de Uso: Gestión Foros Estudiantes No Evaluables.....	53
Tabla 9. Caso de Uso: Gestionar Recursos.....	54
Tabla 10. Descripción de las Entidades.....	59
Tabla 11. Pruebas Realizadas: Gestionar Quiz.....	76
Tabla 12. Pruebas Realizadas: Gestión Quices Estudiante	77
Tabla 13. Pruebas Realizadas: Gestión Encuestas.....	78
Tabla 14. Pruebas Realizadas: Gestión Encuestas Estudiantes.....	78
Tabla 15. Pruebas Realizadas: Gestión Foros Evaluables.....	79
Tabla 16. Pruebas Realizadas: Gestión Foros No Evaluables.....	80
Tabla 17. Pruebas Realizadas: Gestión Foros Estudiantes.....	81
Tabla 18. Pruebas Realizadas: Gestión Foros Estudiantes No.....	82
Tabla 19. Pruebas Realizadas: Gestión de Recursos.....	82

LISTA DE FIGURAS

Figura 1. Modelo Cliente/Servidor.....	18
Figura 2. Modelo de acceso a JSP.....	23
Figura 3. Prototipo Evolutivo.....	31
Figura 4. Casos de uso : Gestionar Quiz.....	46
Figura 5. Casos de uso : Gestionar Quices Estudiante.....	46
Figura 6. Casos de uso : Gestionar Encuestas.....	47
Figura 7. Casos de uso : Gestionar Encuestas Estudiante.....	48
Figura 8. Casos de uso : Gestionar Foros Evaluables.....	49
Figura 9. Casos de uso : Gestionar Foros No Evaluables.....	49
Figura 10. Casos de uso : Gestionar Foros Evaluables Estudiante.....	50
Figura 11. Casos de uso : Gestionar Foros No Evaluables Estudiante...	50
Figura 12. Casos de uso : Gestionar Recursos.....	51
Figura 13. Diagrama E/R : Quices y encuestas aula virtual.....	56
Figura 14. Diagrama E/R : Diagrama E/R: Foros del aula virtual.....	57
Figura 15. Diagrama E/R : Diagrama E/R:Recursos.....	58
Figura 16. Diagrama de secuencia : Crear Quiz.....	61
Figura 17. Diagrama de secuencia : Editar Quiz.....	62
Figura 18. Diagrama de secuencia : Activar Quiz.....	63
Figura 19. Diagrama de secuencia : Crear Encuesta.....	64
Figura 20. Diagrama de secuencia : Activar Encuesta.....	65
Figura 21. Diagrama de secuencia : Responder Quiz.....	66
Figura 22. Diagrama de secuencia : Crear Conversación.....	67
Figura 23. Diagrama de secuencia : Participar Conversación.....	69
Figura 24. Diagrama de secuencia :Calificar Conversación.....	70
Figura 25. Diagrama de secuencia :Crear Recurso.....	71
Figura 26. Diagrama de secuencia :Editar Recurso.....	72
Figura 27. Arquitectura Cliente/Servidor de Dos Capas.....	94
Figura 28. Arquitectura Cliente/Servidor de Tres Capas.....	95
Figura 29. Función del DBMS.....	102
Figura 30. Diagramas de Casos de Uso.....	103

Figura 31. Diagramas de Secuencia.....106

LISTA DE ANEXOS

Anexo A.	Arquitectura Cliente/Servidor.....	98
Anexo B.	Tecnologías de desarrollo en páginas dinámicas.....	101
Anexo C.	Bases de datos.....	103

RESUMEN

TÍTULO: ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO, DESARROLLO E IMPLANTACIÓN DE LOS SERVICIOS DE QUICES, ENCUESTAS Y RECURSOS PARA LOS PORTALES DE LAS ESCUELAS DE LA UIS.ⁱⁱ

AUTORES : Jorge Armando Rueda Estupiñan ⁱ

PALABRAS CLAVE: Encuestas, Quices, Aula Virtual, Estudiantes, Servicios, Conversaciones, Mensajes, Foros Evaluables, Foros No Evaluables, Recursos.

DESCRIPCIÓN

En la actualidad, las escuelas cuentan con un sistema de información orientado a la Web que se encarga de la administración y control de las diferentes actividades, tanto académicas como administrativas que se realizan dentro de ellas, así como del control de usuarios y servicios que se les proporcionan. Estos servicios ofrecen soluciones a los problemas y necesidades de los usuarios, por esta razón las labores de mantenimiento y actualización son indispensables.

El aula virtual es uno de los servicios que ofrece el portal web, con el objetivo de mejorar el funcionamiento y ofrecer todas las opciones requeridas se implementó el servicio de quices y encuestas, el cual por medio de varias interfaces presenta todas las opciones y suplente todas las necesidades que pueda tener el profesor a la hora de querer realizar algunas de estas dos actividades.

Debido a la importancia de aprovechar las nuevas tecnologías para el manejo de los servicios preexistentes en el aula, se realizó reingeniería al módulo de foros, que cuenta con dos modalidades diferentes: evaluables y no evaluables, mejorando los procesos, interfaces y agregando diferentes opciones para su calificación y un manejo más eficiente.

Las necesidades de la comunidad aumentan cada día y el servicio de recursos, debido a las grandes ventajas que ofrece y a su creciente demanda debió ser reestructurado y mejorado usando las nuevas tecnologías y corrigiendo errores que le daban al mismo una imagen de inseguridad frente a los usuarios, esto condujo a una mejora de los procesos, con el fin de prestar un servicio completamente nuevo y con más opciones para los usuarios.

ⁱ Trabajo de grado: Modalidad Práctica Empresarial.

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

ABSTRACT

TITLE: ADMINISTRATION, USERS SUPPORT, MAINTENANCE, DEVELOPMENT AND IMPLEMENTATION OF THE QUIZZES, INQUESTS AND RESOURCES SERVICES FOR UIS'S SCHOOLS PORTALS.ⁱ

AUTHORS: Jorge Armando Rueda Estupiñanⁱⁱ

KEYWORDS: Surveys, Quizzes , Virtual Classroom , Student Services, Conversations , Messages, Evaluables Forums , Chats not assessable , Resources.

DESCRIPTION

Nowadays the schools have a web oriented information system that handles the administration and control of the different academic and administrative activities that take place inside of them, as well as user control and services that are provided. These services provide solutions to the problems and needs of the users, for this reason maintenances and updates are indispensable.

The virtual classroom is one of the services provided by the web portal, having the goal of improving the performance and offer all of the required options, the quizzes and surveys service was implemented, which shows all of the options through several interfaces and covers every need that the teacher can have if he/she wants to do one of these two activities.

Due to the importance of taking advantage of the new technologies for managing the pre-existing services inside the classroom, the forums module's re engineering was done, this service has two different modalities: qualifying and not qualifying, improving the processes, interfaces and adding different options for its qualification and a more efficient use.

The community needs increase every day and the resource service, due to the big advantages that offers and to its growing demand had to be restructured and improved using the new technologies and fixing up bugs that were giving an unsafe image to the users, this led to an improvement of the processes, with the purpose of giving a completely new service and with more options to the users.

ⁱ Degree Work: Business Practice.

ⁱⁱ Faculty of Physical Mechanical Engineering, School of System Engineering and Computing Science.

Director: M.Sc. Luis Ignacio González Ramírez.

INTRODUCCIÓN

Debido al crecimiento de la comunidad dentro de la Universidad Industrial de Santander e inicialmente la comunidad de la escuela de Ingeniería de Sistemas, se hace necesario un canal de información y control dentro de la misma. Es así, como surge la idea de crear una herramienta que en primer lugar facilita la interacción entre los miembros de la comunidad, desarrollo llevado a cabo en el año 2004 por el grupo de desarrollo de software Calumet, el cual, conformado y liderado por estudiantes de pregrado de Ingeniería de Sistemas, da vida a la primera versión del portal web de la escuela de Ingeniería de Sistemas EISIWeb.

Dado el crecimiento y la acogida del portal web EISIWeb, apoyado en la labor de administración, mantenimiento y actualización que realizan sus integrantes, el grupo de desarrollo de software Calumet ha logrado la implementación de su portal web en las escuelas pertenecientes a las facultades de Ingenierías Fisicomecánicas y Fisicoquímicas, así como en sus respectivas decanaturas.

En el aula virtual, uno de los servicios que ofrece el portal web, con el objetivo de mejorar el funcionamiento y ofrecer todas las opciones requeridas se implementó el servicio de quices y encuestas, el cual por medio de varias interfaces presenta todas las opciones y suple todas las necesidades que pueda tener el profesor a la hora de querer realizar algunas de estas actividades.

Debido a la gran importancia de mantener y aprovechar las nuevas tecnologías para el manejo de los servicios preexistentes en el aula, se realizó reingeniería al módulo de foros, que cuenta con dos modalidades diferentes: evaluables y no evaluables, mejorando los procesos, interfaces y agregando diferentes opciones para su calificación y un manejo más eficiente.

Las necesidades de la comunidad cada día van en aumento y el servicio de recursos debido a las grandes ventajas que ofrece y su creciente demanda debió ser reestructurado y mejorado usando las nuevas tecnologías y corrigiendo errores que le daban al mismo una imagen de inseguridad frente a los usuarios, esto nos condujo a un mejoramiento de los procesos, con el fin de prestar un servicio completamente nuevo, mejorado y con más opciones para los usuarios.

1. DEFINICIÓN DEL PROBLEMA

En la actualidad los portales web de las escuelas cuentan con un sistema de información orientado a la Web el cual se encarga de la administración y control de las diferentes actividades que se realizan dentro de estas, así como un control a los usuarios y los servicios que se proporcionan. Los servicios de los portales están en constante mejoramiento pues deben adaptarse a las necesidades que se van presentando, lo cual aumenta la vida útil del software evitando su obsolescencia por lo cual las labores de actualización y mantenimiento se hacen indispensables.

Actualmente existen los servicios de Quices y Encuestas dentro del Aula Virtual, pero cuentan con una interfaz antigua y presentan muchas limitaciones, razón por la cual surge la necesidad de re-implementar estos servicios haciendo uso de las nuevas tecnologías disponibles, así mismo incorporando nuevas y eficientes funcionalidades de manera que aumente su usabilidad.

El servicio de Foros tanto evaluables como no evaluables del Aula Virtual existe actualmente y es usado por parte de los usuarios del sitio, este servicio cuenta con una interfaz que no es acorde a las necesidades de los usuarios y presenta múltiples problemas debido a las tecnologías con las cuales fue implementado; Debido a la gran importancia de los mismos se plantea la necesidad de una reingeniería, con el fin de mejorar completamente su interfaz y cada uno de sus procesos.

El servicio de recursos el cual se usa actualmente para el préstamo de todo tipo elementos entre estudiantes y el manejo de los diferentes recursos con que cuenta la escuela y se pueden poner en uso de los diferentes usuarios de la comunidad, debido a la manera en que se implementó no ha podido brindar a cabalidad lo esperado del mismo por cual se plantea la posibilidad

de aplicar reingeniería al mismo y de esta manera hacer un mejor manejo de los procesos internos y el cambio de su interfaz para que sea más amigable con el usuario.

2. JUSTIFICACIÓN

Los servicios del portal de las escuelas deben mejorar constantemente y proporcionalmente a los cambios que se presentan en su entorno, a su vez deben dar correcta solución a los problemas y las necesidades que surjan por parte de los usuarios, razón por la cual las labores de mantenimiento y actualización se hacen indispensables.

Enfocados en prestar un mejor servicio a los profesores, estudiantes y comunidad en general supliendo cada una de sus necesidades, se implantaron los servicios de quices y encuestas dentro del aula virtual con el fin de ofrecer al profesor una posibilidad adicional a la hora de evaluar y consultar a sus estudiantes de tal manera que se facilita el trabajo del profesor y brinda a los estudiantes un canal de evaluación más ágil y dinámico, así como la posibilidad de participar activamente en las diferentes consultas que se hagan en el curso.

Debido a la necesidad de establecer canales de comunicación eficientes entre los estudiantes y el profesor se estableció como prioridad re-implementar el servicio de foros tanto evaluables como no evaluables, brindando además funcionalidades adicionales volviendo así el servicio más interactivo y participativo en sí mismo, además de esto establecer canales de calificación más ágiles para los profesores en los foros evaluables.

Un servicio para el préstamo de recursos dentro de la comunidad, se convirtió en una necesidad pues no se contaba con un canal por el cual los diferentes usuarios pudieran poner a disposición sus propios recursos, por medio del servicio ayudando a una administración más controlada y además sirviendo de medio de comunicación entre los interesados en el recurso y su dueño.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Aplicar reingeniería y crear nuevos servicios al Portal Web Comunidad Académica de las Escuelas de la UIS para permitir la interacción de la Comunidad y el acceso a la información de manera más sencilla, ágil, óptima y eficiente.

3.2 OBJETIVOS ESPECÍFICOS

3.2.1 Efectuar labores de Administración del portal web de las escuelas.

- Generar Backups (Copias de Respaldo), todos los días de la Base de Datos.
- Hacer seguimiento del uso que hacen los usuarios a los servicios de Foros, Eventos Propuestos, Cartelera, Archivos y Mi perfil dentro de todos los portales, para detectar usos indebidos o incorrectos por parte de éstos.
- Actualizar periódicamente las Bases de Datos para mantener al día la información referente a matrículas, horarios, estados y categorías de los usuarios.
- Atender las consultas y sugerencias que realicen los usuarios para proponerlas como mejoramiento y ofrecimiento de nuevos servicios en la próxima versión de los portales WEB.

3.2.2 Análisis, diseño e implementación de nuevos servicios para los siguientes módulos:

- Implementar dentro del aula virtual la creación de quices y encuestas, de manera que los profesores puedan aplicarlos en las distintas asignaturas. Las encuestas tendrán una estructura similar a la de quices pero no serán calificables.
- Implementar reingeniería al préstamo colaborativo de recursos entre los usuarios de la comunidad.
- Re-implementación de los foros tanto evaluables como no evaluables en el aula virtual.

3.2.3 Desempeñar labores de soporte a los usuarios de los portales, brindando así solución a sus diferentes necesidades o situaciones que se puedan presentar. Entre estas labores se destacan:

- Capacitar usuarios y estudiantes del primer nivel en el uso de servicios de los servicios implementados en este proyecto.
- 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 o la facultad y que por algún motivo necesitan registrarse en el sitio.

4. MARCO TEÓRICO

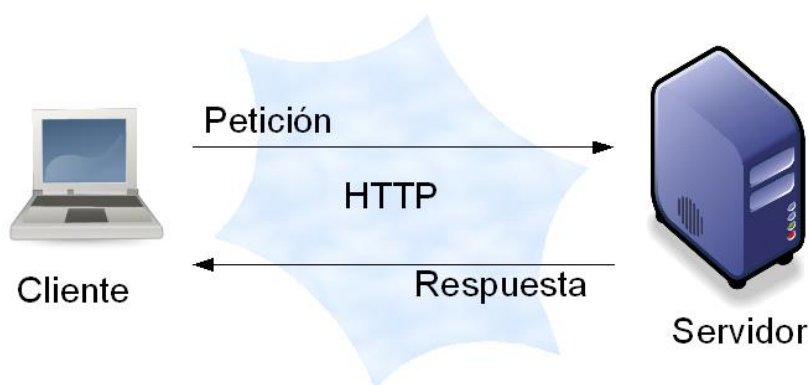
4.1 ARQUITECTURA CLIENTE/SERVIDOR

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 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: Arquitectura cliente servidor. [en línea].

<https://www.codejobs.biz/es/blog/2014/11/05/que-es-el-modelo-cliente-servidor-networking>[citado el 05 Octubre de 2015].

La arquitectura C/S presenta unas características y clasificaciones que permiten la aplicación correcta de este modelo. (Ver Anexo A).

4.1.1 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.
- Capa de Servidor: Se utiliza el motor de bases de datos MySQL, el cual se encuentra en el mismo servidor web.

4.1.2 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 máquinas 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.

4.1.3 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 diagnóstico 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.

4.2 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, las cuales se pueden agrupar en Código del Lado del Cliente (Client Side Scripts) y Código del Lado del Servidor (Server Side Scripts). (Ver Anexo B).

4.2.1 Tecnología aplicada: La tecnología aplicada para la creación del portal web fue 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 páginas 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 esté 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.

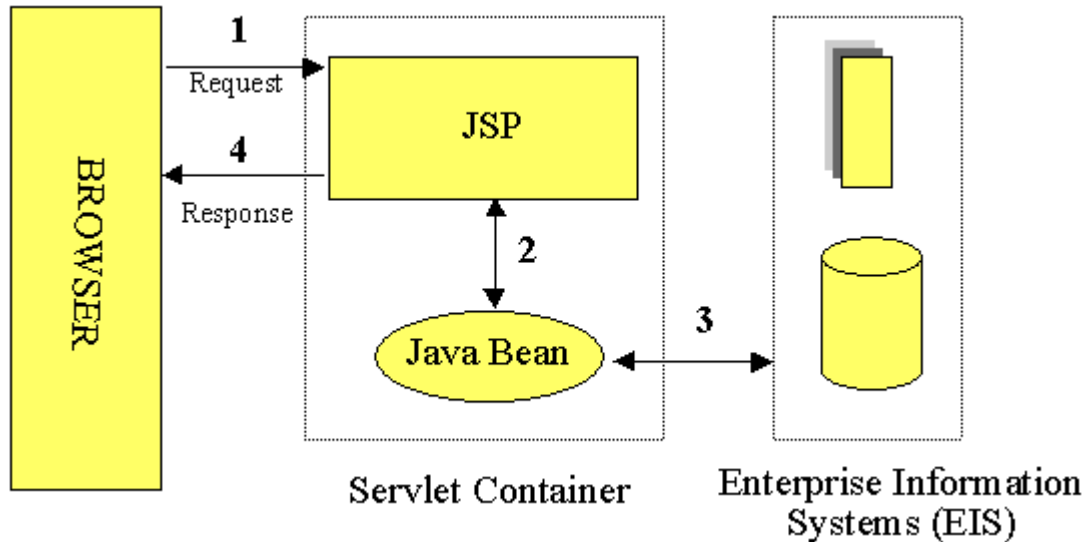
Una página JSP se compila a una aplicación Java la primera vez que se invoca, y de esta aplicación Java se crea una clase que empieza a ejecutarse en el servidor como un servlet. Un JSP es una página web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa que recibe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

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 2. Modelo de acceso a JSP



Fuente:ModeloJSP.[enlínea]<http://geneura.ugr.es/~jmerelo/JSP/modelo1.gif> [citado el 05 Octubre de 2015].

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.

4.3 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.

Dentro de las bases de datos existen diferentes modelos, entre los cuales encontramos bases de datos jerárquicas, de red y relacionales. Así mismo para tener conexión a estos modelos es necesario utilizar conectores los más comunes son ODBC, MDB y JDBC. (Ver Anexo C).

4.3.1 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.

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 consumo puede ser ejecutado en una máquina 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.

4.4 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 services (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.

4.5 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.

4.5.1 Subversión: Sistema de control de versiones iniciado 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.

4.6 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.

4.6.1 Clases: Definición de todos los elementos de que está 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.

4.6.2 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).

4.6.3 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.

4.6.4 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.

4.6.5 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.

4.6.6 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 orientada a objetos que se usó en el desarrollo de este proyecto.

4.6.7 Java y JDK (Java Development Kit): Java es un lenguaje desarrollado por Sun Microsystems, en el año 2009 fue adquirida por la compañía Oracle.

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: Applet Viewer.

4.7 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.

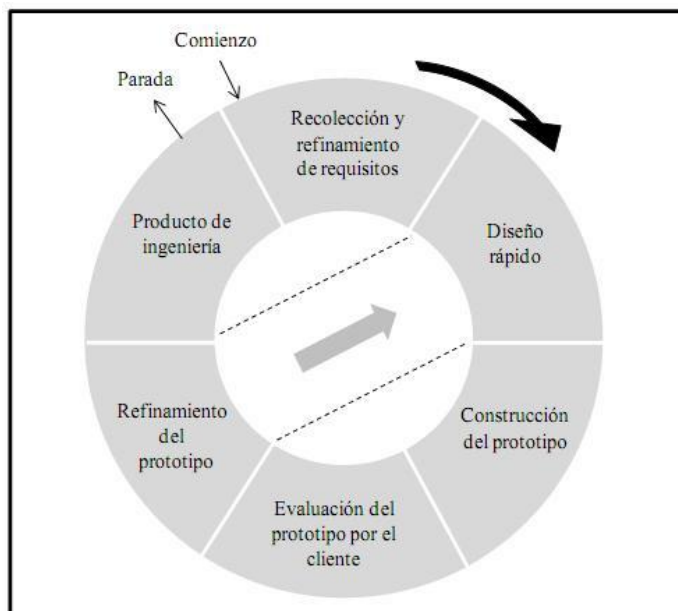
4.7.1 Servidor Jakarta Tomcat: Servidor de aplicaciones Java basado en los estándares definidos por Sun Microsystems. Tomcat es desarrollado como parte del proyecto de código abierto Jakarta de la fundación de software Apache y es uno de los servidores de aplicaciones Java más utilizados, en especial porque es liviano, cumple con todos los estándares, sencillo de instalar.

5. DISEÑO METODOLÓGICO

5.1 PROTOTIPO EVOLUTIVO

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

Figura 3. Prototipo Evolutivo.



Fuente: ModeloJSP. [en línea] <http://cflores334.blogspot.es/img/prototipo.jpg> [citado el 05 octubre de 2015].

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

Los portales presentan constantemente necesidades por parte de los usuarios, por lo tanto, se requiere el desarrollo de nuevos servicios y así mismo la mejora de los ya existentes, debido a que el sistema no es un producto final, el mismo se considera en constante reconstrucción.

El desarrollo de los aspectos visuales del sistema puede estar sujeto a cambios por parte de los usuarios de los portales durante el desarrollo del proyecto. Durante la primera etapa de los nuevos servicios es necesario tener una clara interpretación de la especificaciones dadas por los usuarios y escuelas, pero muchas veces los usuarios no tienen una idea clara de lo que necesitan. 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.

Otra ventaja es tener la posibilidad de realizar cambios en etapas tempranas y crear 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 especifican en detalle los objetivos globales del software a realizar, mediante una reunión entre el desarrollador y el usuario, en la cual se identifican los requisitos básicos y se concluyen las áreas donde se necesita mayor definición.

- Posteriormente se presenta al usuario el diseño de un prototipo que se enfoca en los aspectos visuales del software, 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.

5.2 LENGUAJE DE MODELADO UNIFICADO

El Lenguaje de Modelado Unificado o Unified Modeling Language (UML), es el más utilizado en la actualidad. Es un lenguaje gráfico 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.

5.2.1 Diagramas de UML: Los diagramas 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 casos de uso, el diseño con diagramas de clases, objetos, etc.

- UML facilita el entendimiento de la información, la función y el comportamiento de un sistema, haciendo fácil 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.

Diagramas de casos de uso

Representación gráfica del entorno del sistema (actores) y su funcionalidad principal. Describe lo que hace el 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.

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.

5.3 ESTÁNDARES DE PROGRAMACIÓN

5.3.1 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).

5.3.2 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:

- **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 tabla que almacena los puntos que se tratan en un consejo de escuela es llamada “TB_PuntosConsejo”.
- **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 de “GuionEncuesta” es conocida como “TR_GuionEncuesta”.
- **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 de los portales. El prefijo a anteponer a los nombres de estas tablas es “TP_”, es decir la tabla “Usuarios”, es conocida como “TP_Usuarios”.

5.3.3 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: Encuestas.java, Foros.java.

5.3.4 Páginas JSP: Los nombres de las páginas JSP que componen los portales 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: ActivarQuiz.jsp, CalificarMensajes.jsp.

5.3.5 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 almacenados todos los archivos .jpg, .gif, .png.

El sitio cuenta con un directorio llamado Script, donde están todos los archivos de JavaScript “.js” necesarios para el buen funcionamiento del portal.

6.DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO

Como se mencionó anteriormente para el desarrollo de este proyecto se siguió la metodología de prototipado evolutivo.

Al iniciar el proyecto se elaboró un primer prototipo durante la fase de requerimientos, el cual fue mejorado con la inclusión de nuevos requerimientos surgidos en la fase de desarrollo, a medida que se generaba un prototipo, el mismo era sometido a pruebas de funcionamiento y se le realizaban los refinamientos pertinentes a partir del resultado de dichas pruebas.

6.1 PROTOTIPO ESPERADO

Al inicio el proyecto no se tenía una concepción clara de cómo sería el producto final, sin embargo, durante el desarrollo y evolución de los prototipos, las pruebas y análisis del sistema se pudo comprobar que se estaba acercando a los requerimientos iniciales, esto con el fin de enfocar exitosamente el desarrollo a la solución de las necesidades de los usuarios.

El objetivo específico inicial y los requisitos que surgieron se dieron gracias a la realización 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. Implementar en el Aula virtual la creación de quices y encuestas, de manera que los profesores puedan aplicarlos en las distintas asignaturas y su calificación se realice automáticamente. Las encuestas tendrán una estructura similar a la de quices pero no serán calificables y además de esto contarán con un lugar para ver los resultados de forma estructurada, por medio de gráficos dependiendo el tipo de la pregunta.**

Objetivo inicial:

- Implementar dentro del aula virtual la creación de quices y encuestas, de manera que los profesores puedan aplicarlos en las distintas asignaturas. Las encuestas tendrán una estructura similar a la de quices pero no serán calificables.

Requisitos finales del objetivo:

- El profesor debe poder crear un quiz independientemente si el aula posee un plan de quices, este se debe crear automáticamente con el quiz y quedar disponible para la utilización del profesor y de los estudiantes del aula; De esta manera hacer parte de las calificaciones dentro del aula.
- El quiz debe contar un tema para una búsqueda más fácil por parte del profesor definiéndola no solamente por medio del título del mismo.
- Los quices se deben crear para toda el aula no para un grupo en particular, esto genera que pueda ser ejecutado en cualquier momento para alguno de los grupos que pertenecen a la misma, activándose de manera individual o en la manera en que el profesor así lo considere.
- El profesor podrá ver en tiempo real la ejecución del quiz contando así con un control total de su ejecución.
- El quiz debe contar únicamente con dos tipos de preguntas con número ilimitado de opciones de respuesta (única opción y múltiple opción), de esta manera brindar la posibilidad de una calificación inmediata, por lo cual es estudiante termina el quiz y conoce su nota.
- Al igual que los demás planes dentro del aula el profesor tendrá la posibilidad de calificar (Individualizar la nota) los quices por medio de las diferentes interfaces ya existentes en el aula.
- El quiz debe tener una interfaz en la cual los estudiantes solo visualizarán una pregunta y deberá ser solucionada para pasar a una siguiente de esta manera evitar ayuda entre estudiantes.
- Las encuestas deben tener una opción adicional de respuesta tipo párrafo o abierta con el fin de obtener diferentes puntos de vista sobre una cuestión en particular.

- Estas encuestas deben contar con una visualización ágil y eficaz dependiendo el tipo de preguntas con la cual se realice.
- La encuesta tendrá dos opciones para el manejo de la visualización de las respuestas, con las cuales el profesor decidirá dependiendo de su propia opinión mostrar o no a los estudiantes los resultados.
- El profesor no conocerá en ningún momento de la misma los resultados de un estudiante en cuestión pues se quiere manejar la privacidad de la consulta.
- Las encuestas se podrán activar y desactivar dadas las preferencias del profesor, teniendo en cuenta que la encuesta será activada por rangos de días.

2. Implementar reingeniería al servicio de recursos en el módulo perteneciente a la publicación y mantenimiento de los recursos por parte de los usuarios.

Objetivo inicial:

- Implementar reingeniería al préstamo colaborativo de recursos entre los usuarios de la comunidad.

Requisitos finales del objetivo:

- Mejoramiento de la interfaz de usuario para el módulo, dando una mejor organización y un solo lugar donde se pueda dar manejo a las diferentes opciones que debe tener el usuario a la hora de publicar un recurso.
- El usuario tendrá la posibilidad de asignar a un usuario de la comunidad escuela un recurso y entablar una comunicación por medio de correo electrónico.

- Se deberá poder calificar el estado del recurso y al usuario propietaria para así tener una idea de cómo se da le da manejo al mismo.

2.Re-implementación de los foros tanto evaluables como no evaluables en el aula virtual.

Objetivo inicial:

- Re-implementación de los foros tanto evaluables como no evaluables en el aula virtual.

Requisitos finales del objetivo:

- Los foros deberán ser visualizados por medio de una interfaz acorde a las necesidades de los usuarios.
- Las diferentes participaciones dentro de un foro podrán ser calificadas por cualquier usuario, dando de esta una idea de la importancia de la misma dentro de la conversación.
- Los foros evaluables tendrán dos interfaces para su respectiva calificación, por subgrupo de clase o participación general, por medio de la última dar la posibilidad que el profesor conozca el contexto de la respuesta, brindado así una ayuda adicional a la hora de calificar el mismo, además de visualizar el puntaje obtenido por los votos de los diferentes usuarios participantes.
- La interfaz de individualización de las notas de los foros deberá ser cambiada totalmente presentando una mejor imagen al profesor.

Para aquellos servicios que lo requieran, se debe incluir la rutina de logueo de usuario y verificación de sesión, requerimiento de seguridad intrínseco 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 las escuelas y facultades; teniendo en cuenta entre otras:
- Generar Backups (copias de respaldo) diariamente de la Base de Datos.
- 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 de las escuelas 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 las 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 web, 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 de las escuelas.
- 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.

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 las bases de datos Diamante y de los directorios de los sitios de las escuelas.
- 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 de las escuelas, 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 de los portales, 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 los portales de las escuelas 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 de las escuelas, 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 relevarán las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro de los portales de las escuelas 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:

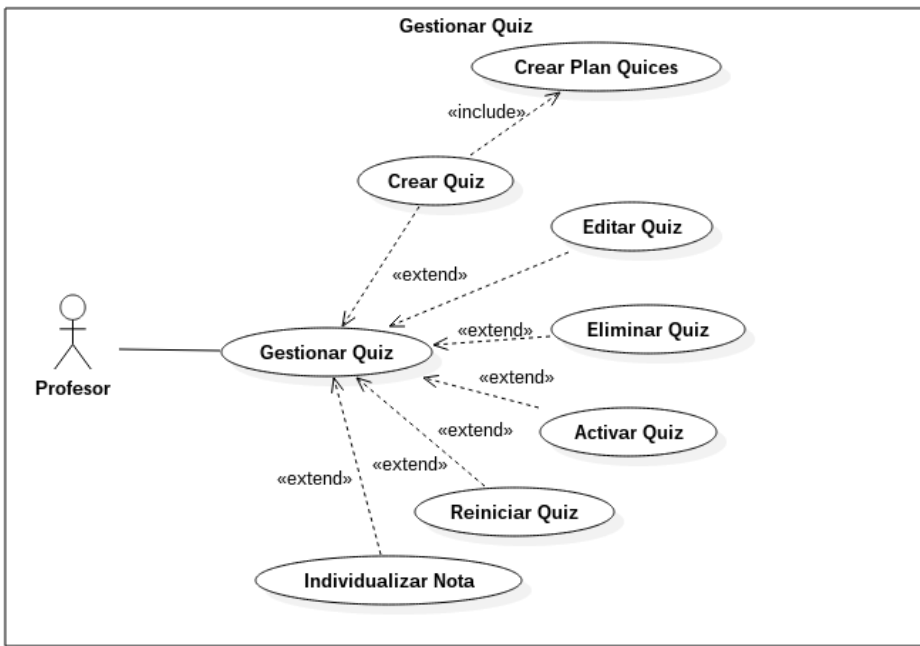
Para usuarios con perfil de administrador.

- 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.
-

6.1.1. Diagramas de Casos de Uso

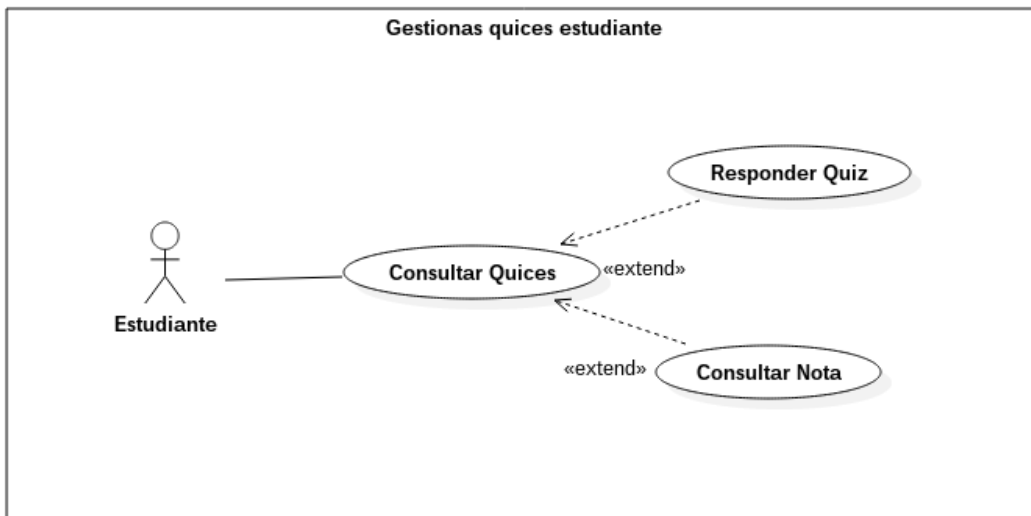
6.1.1.1. Servicio de Quices y Encuestas.

Figura 4. Casos de uso: Gestionar Quiz.



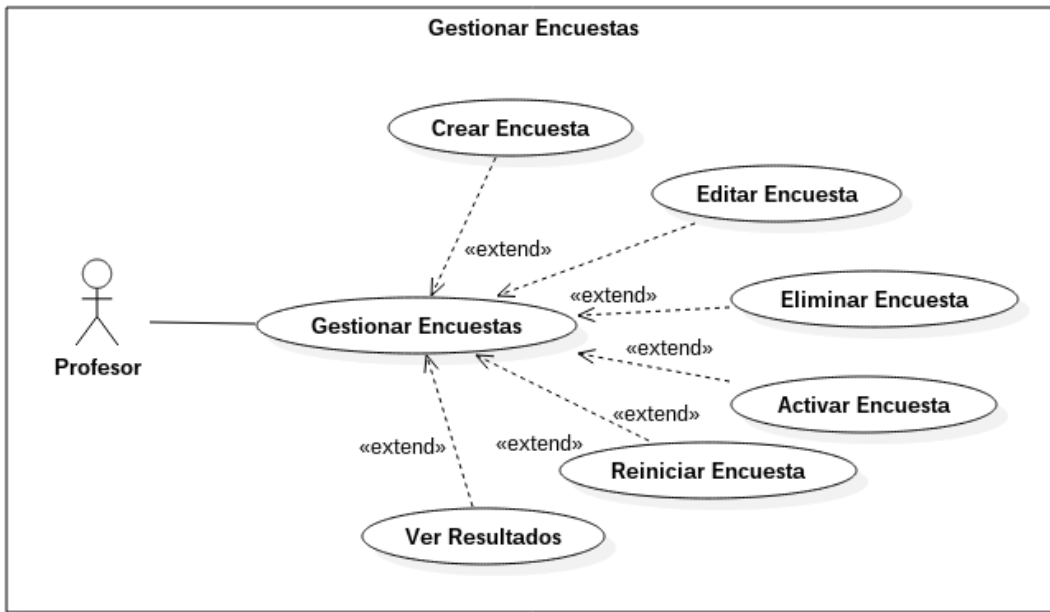
Fuente: Autor.

Figura 5. Casos de uso: Gestionar Quices Estudiante.



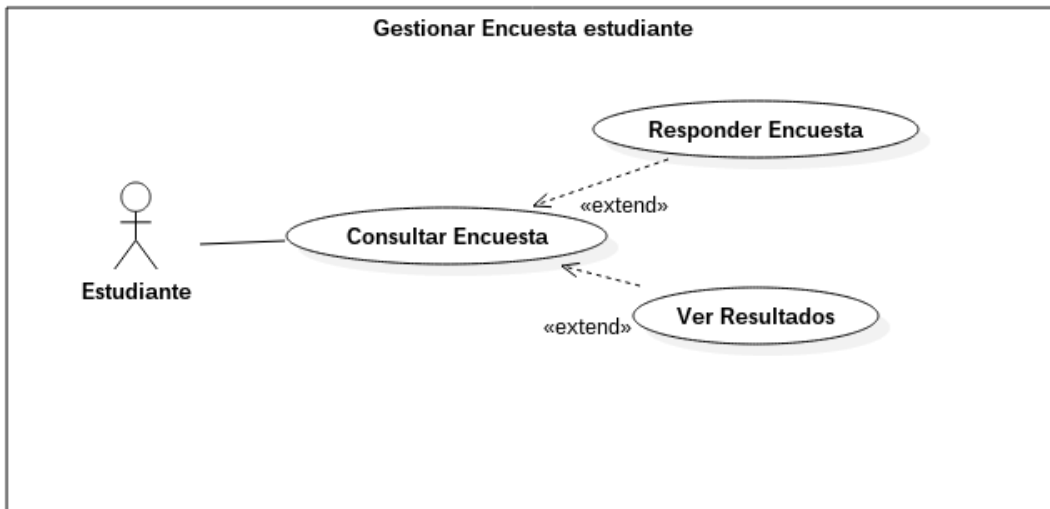
Fuente: Autor.

Figura 6 Casos de uso: Gestionar Encuestas.



Fuente: Autor.

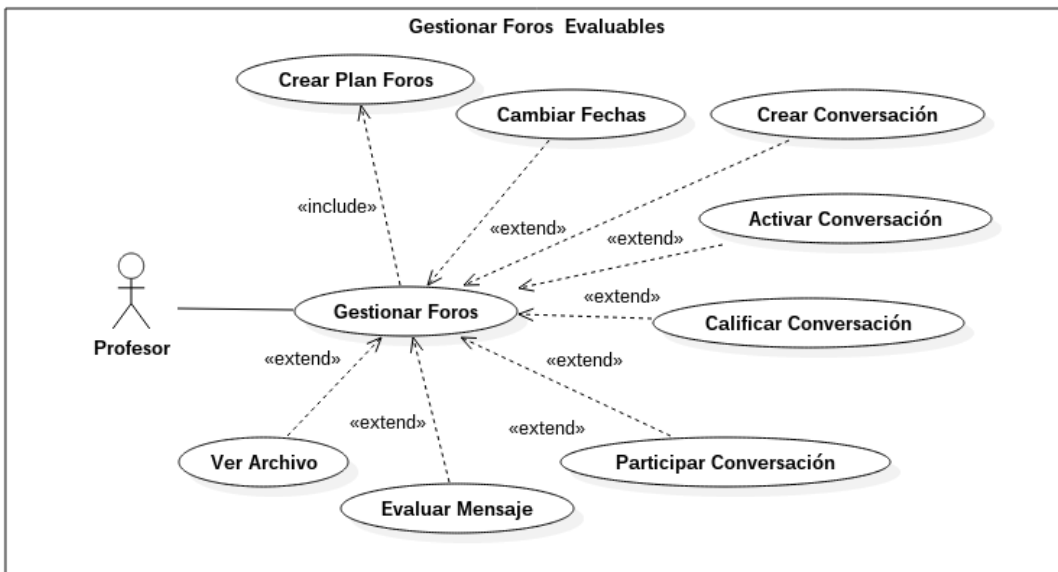
Figura 7 Casos de uso: Gestionar Encuestas Estudiante.



Fuente: Autor.

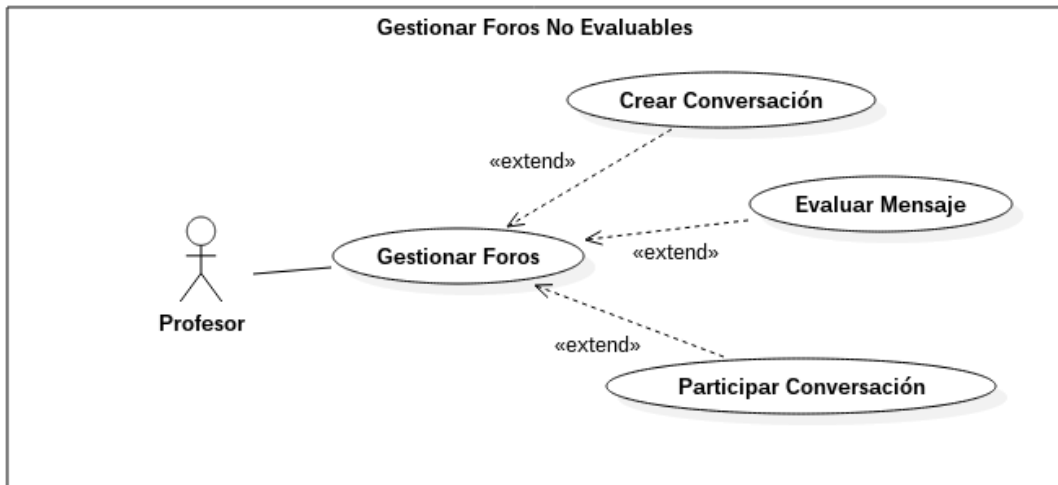
6.1.1.2. Servicio de Foros.

Figura 8 Casos de uso: Gestionar Foros Evaluables.



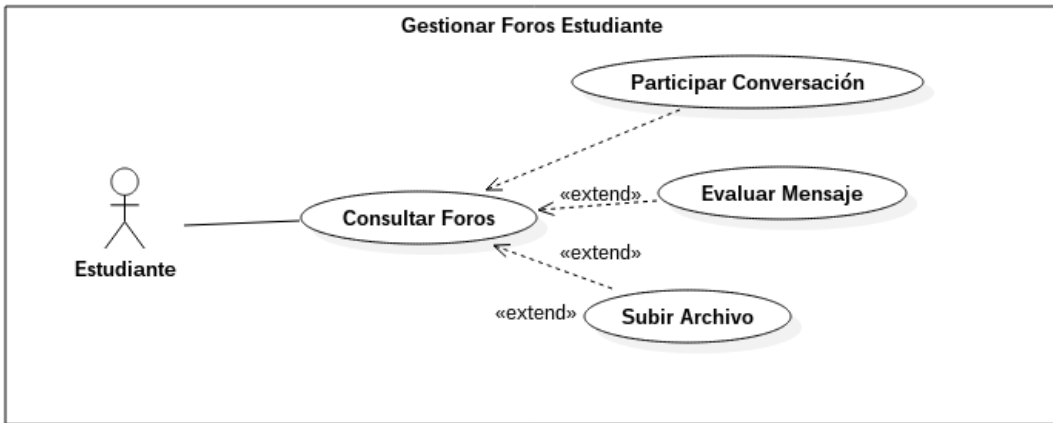
Fuente: Autor.

Figura 9 Casos de uso: Gestionar Foros No Evaluables.



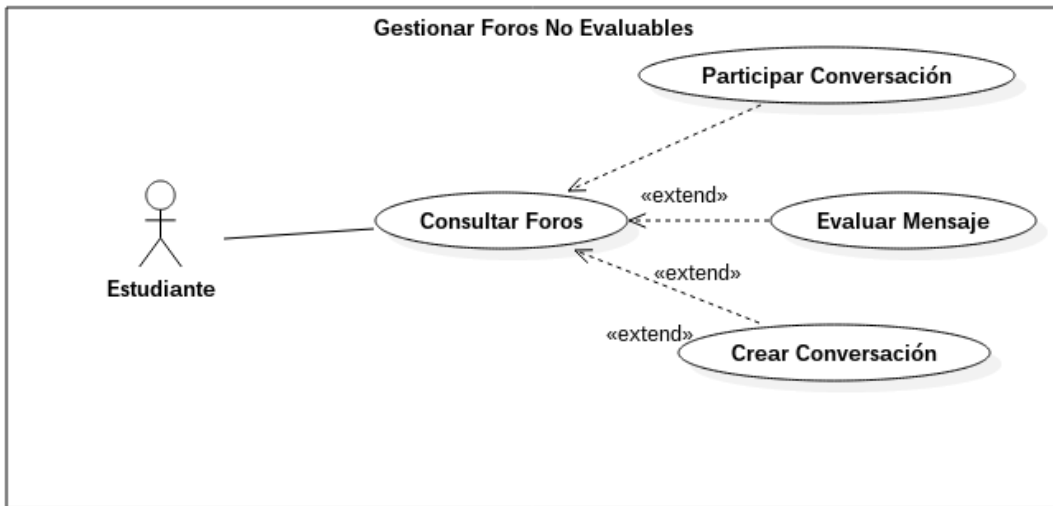
Fuente: Autor.

Figura 10 Casos de uso: Gestionar Foros Evaluables Estudiante.



Foros: Autor.

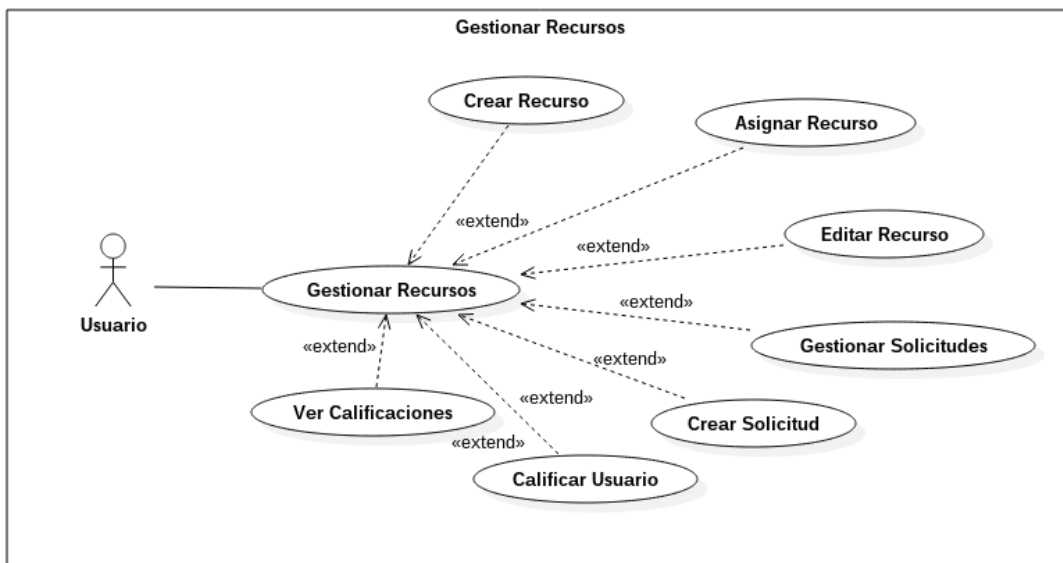
Figura 11 Casos de uso: Gestionar Foros No Evaluables Estudiante.



Fuente: Autor.

6.1.1.3. Servicio de Recursos.

Figura 12 Casos de uso: Gestionar Recursos.



Fuente: Autor.

6.1.2 Documentación de casos de uso del sistema: Una vez identificados los casos de uso, procedemos a documentar sus pasos, describiendo cada una de las acciones que se llevan a cabo en los casos de uso anteriormente mencionados. Los detalles completos se documentan en los anexos de este libro.

Tabla 1. Casos de uso: Gestionar Quiz.

TÍTULO	FUNCIONES PRIMARIAS
Gestión Quiz	Accede al servicio que lista los quices dependiendo su estado dentro del aula.
Crear Quiz	Crea un nuevo quiz para el aula.
Editar Quiz	Edita uno de los quices creados anteriormente.
Eliminar Quiz	Elimina uno de los quices del aula.
Activar Quiz	Activa el quiz para alguno de los grupos del aula, con una duración.
Reiniciar Quiz	Activa nuevamente el quiz

	agregando un tiempo estimado por el profesor.
Individualizar Nota	Las notas de los quices se ponen a la respectiva actividad.

Fuente: Autor.

Tabla 2. Casos de uso: Gestión quices estudiante.

TÍTULO	FUNCIONES PRIMARIAS
Consultar Quices	Accede al menú que lista los quices dependiendo su estado dentro del aula.
Responder Quiz	Responde el quiz.
Consultar Nota	El estudiante consulta las notas de los quices desarrollados.

Fuente: Autor.

Tabla 3. Casos de uso: Gestionar Encuestas.

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Encuestas	Accede al servicio que lista las encuestas dependiendo su estado dentro del aula además de múltiples opciones.
Crear Encuesta	Crea las encuestas en el aula.
Editar Encuesta	Edita las encuestas existentes en el aula.
Eliminar Encuesta	Elimina las encuestas del aula.
Activar Encuesta	Activa la encuesta dentro del aula.
Reiniciar Encuesta	Activa nuevamente la encuesta con las nuevas fechas definidas por el profesor.
Ver Resultados	El profesor revisa los resultados de las encuestas que han finalizado.

Fuente: Autor.

Tabla 4. Casos de uso: Gestión encuestas estudiantes.

TÍTULO	FUNCIONES PRIMARIAS
Consultar Encuesta	Accede al servicio que lista las encuestas dependiendo su estado dentro del aula además de múltiples opciones.
Responder Encuesta	Responde las encuestas disponibles para el usuario.
Ver Resultados	Ver resultados de las encuestas.

Fuente: Autor.

Tabla 5. Casos de uso: Gestión Foros Evaluables.

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Foros	Accede al servicio que lista las foros dependiendo su estado dentro del aula .
Cambiar Fechas	Accede a cambiar las fechas de la actividad (Foro).
Crear Conversación	Crea una conversación en el aula.
Activar Conversación	Activa las conversaciones dentro del aula.
Calificar Conversación	Califica los mensajes en las diferentes conversaciones.
Participar Conversación	Participar en alguna de las conversaciones del aula.
Evaluar Mensaje	Evalúa alguno de las mensajes dentro de las diferentes conversaciones.
Ver Archivo	Mostrar los archivos de los diferentes estudiantes.

Fuente: Autor.

Tabla 6. Casos de uso: Gestión Foros No Evaluables.

TÍTULO	FUNCIONES PRIMARIAS
Consultar Foros	Accede al menú que lista las foros dependiendo su estado y tipo dentro del aula .
Participar Conversación	Participar en alguna de las conversaciones del aula.
Evaluar Mensaje	Evalúa alguno de las mensajes dentro de las diferentes conversaciones.
Crear Conversación	Crea una conversación en el aula.

Fuente: Autor.

Tabla 7. Casos de uso: Gestión Foros estudiantes Evaluables.

TÍTULO	FUNCIONES PRIMARIAS
Consultar Foros	Accede al menú que lista las foros dependiendo su estado y tipo dentro del aula .
Participar Conversación	Participar en alguna de las conversaciones del aula.
Evaluar Mensaje	Evalúa los mensajes dentro de las diferentes conversaciones.
Subir Archivo	Sube el archivo necesario para completar la actividad.

Fuente: Autor.

Tabla 8. Casos de uso: Gestión Foros estudiantes No Evaluables.

TÍTULO	FUNCIONES PRIMARIAS
Consultar Foros	Accede al menú que lista las foros dependiendo su estado y tipo dentro del aula .
Participar Conversación	Participar en alguna de las conversaciones del aula.

Evaluar Mensaje	Evalúa los mensajes dentro de las diferentes conversaciones.
Crear Conversación	Crea una conversación en el aula.

Fuente: Autor.

Tabla 9. Casos de uso: Recursos.

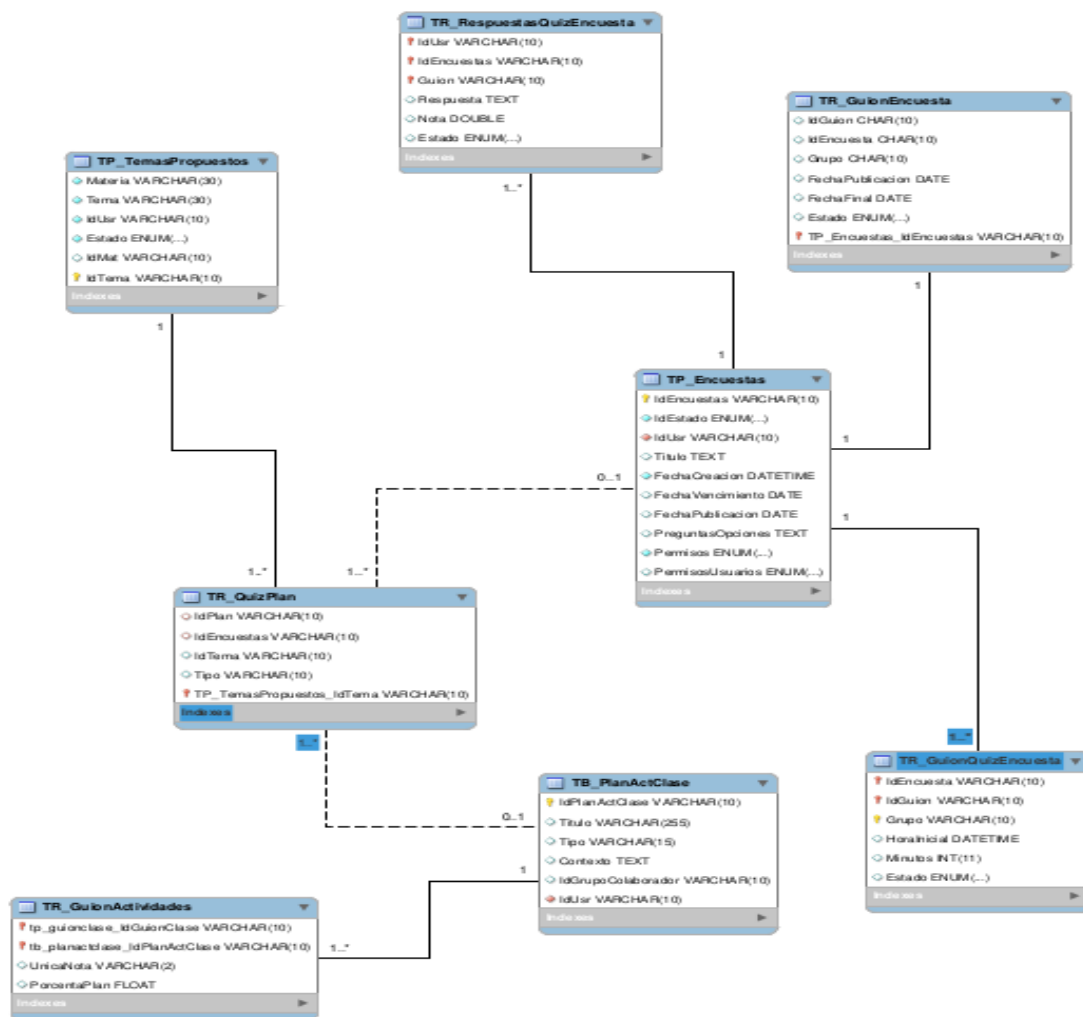
TÍTULO	FUNCIONES PRIMARIAS
Gestionar Recursos	Accede al menú donde se encuentran todos recursos con sus respectivas opciones.
Crear Recurso	Crear un nuevo recurso.
Asignar Recurso	Asigna un recurso a algún solicitante.
Editar Recurso	Edita alguno de los recursos disponibles del usuario.
Gestionar Solicitudes	Gestiona las solicitudes de los diferentes usuarios sobre algún recurso.
Crear Solicitud	Crear una solicitud para algún recurso.
Calificar Usuario	Calificar un usuario dependiendo su comportamiento con alguno de los recursos.
Ver Calificaciones	Visualiza las diferentes calificaciones de los recursos del usuario.

6.1.3 Diseño y Análisis

6.1.3.1. Diagrama Entidad/Relación de los servicios desarrollados

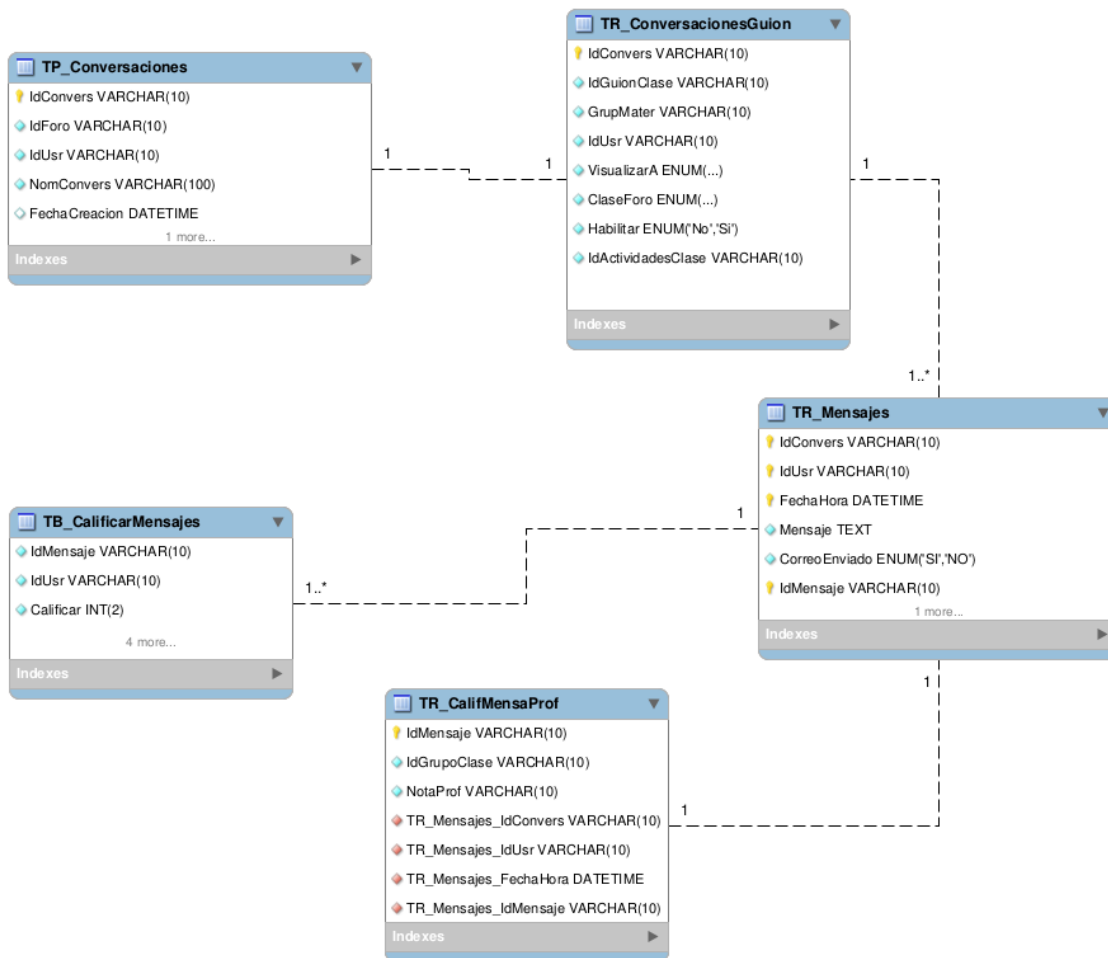
6.1.3.2. Quices y encuestas aula virtual.

Figura 13. Diagrama E/R: Quices y encuestas aula virtual.



6.1.3.3. Foros del aula virtual

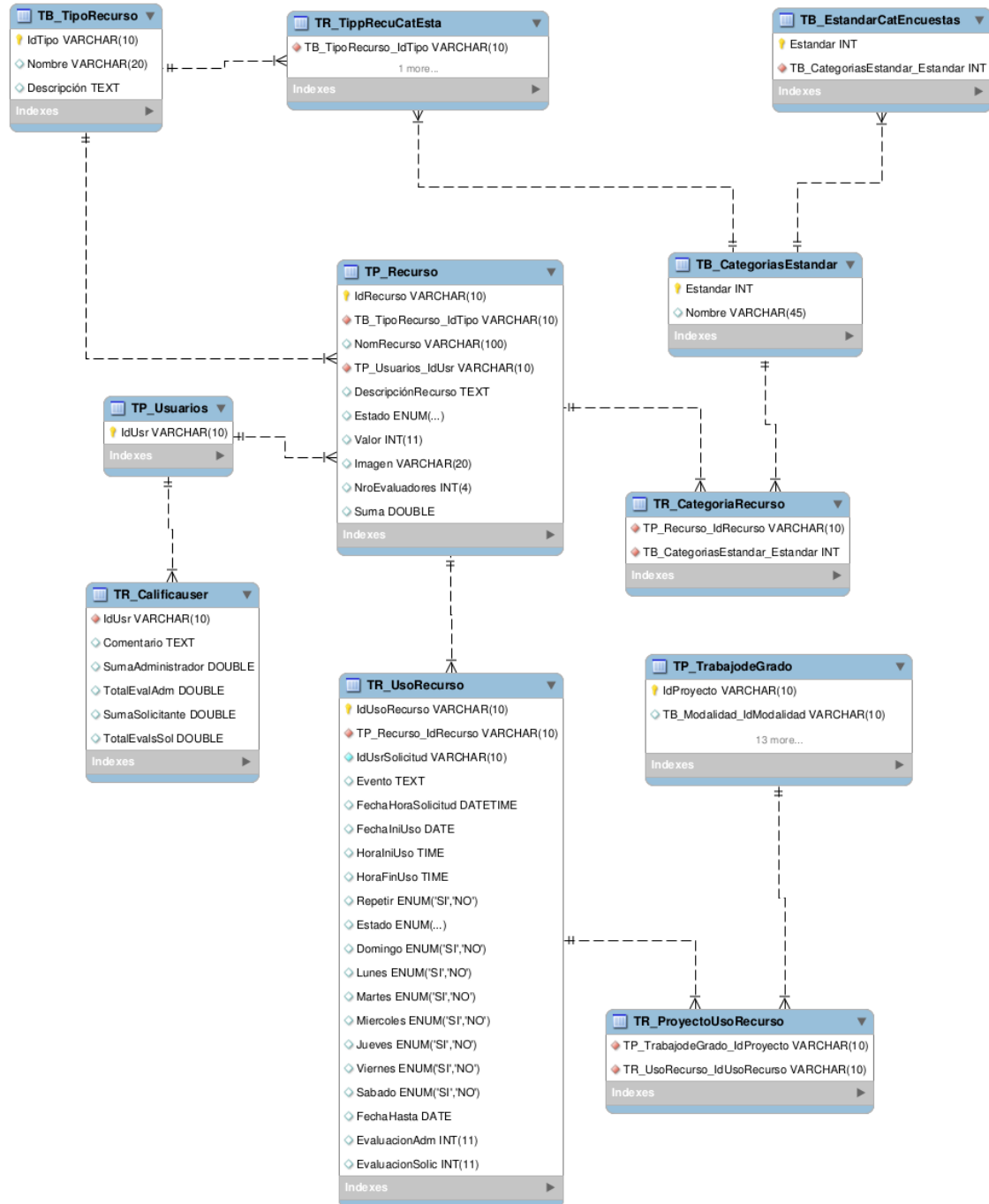
Figura 14. Diagrama E/R: Foros del aula virtual.



Fuente: Autor.

6.1.3.4. Recursos

Figura 15. Diagrama E/R: Recursos.



Fuente: Autor.

6.1.4. Descripción de las Entidades

Definición de las entidades utilizadas en la base de datos Diamante ubicada en el servidor de los portales de las escuelas.

Tabla 10. Descripción de las Entidades

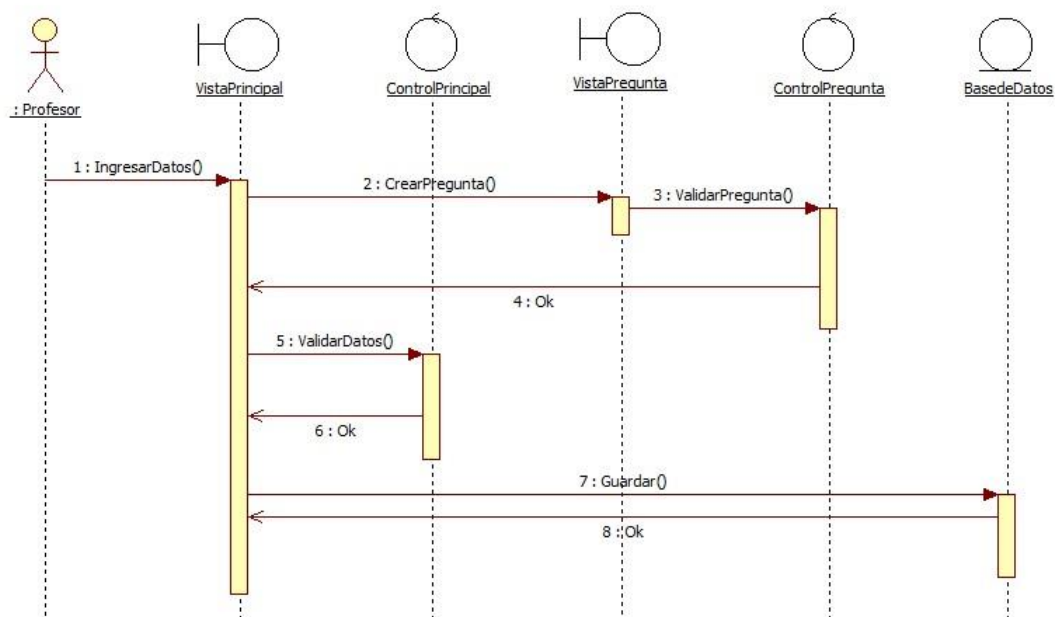
ENTIDAD	DESCRIPCIÓN
TP_Encuestas	Contiene todas las encuestas que se realizan la base de datos diamante.
TR_GuionQuizEncuesta	Contiene la relación entre las encuestas y el guión (aula), y además se guarda la información sobre su activación.
TR_RespuestasQuizEncuesta	Contiene las respuestas de los estudiantes sobre los quices y las encuestas del aula.
TR_GuionEncuesta	Tabla que contiene la información de las encuesta que están siendo desarrolladas o que ya lo fueron dentro del aula.
TR_QuizPlan	Contiene la relación entre los quices y un plan dentro del aula.
TR_GuionActividades	Contiene la información de las diferentes actividades que en esta están activas.
TR_PlanActClase	Contiene la relación entre las actividades de clase con su respectivo plan.
TP_TemasPropuestos	Contiene los temas de una materia dentro del aula, es necesaria para la creación de los quices pues estos cuentan con un tema.
TP_Conversaciones	Contiene la información de todas las conversaciones dentro del sitio.
TR_ConversacionesGuion	Contiene la relación entre las conversaciones y el guión.

TR_Mensajes	Contiene la relación entre las conversaciones y los respectivos mensajes, en este se guardan cada uno de los mensajes para una conversación.
TB_CalificarMensajes	Contiene los votos que se hacen a cada uno de los mensajes dentro de las diferentes conversaciones.
TR_CalifMensaProf	Contiene las calificaciones que el profesor otorga a los diferentes mensajes que hacen parte de los foros calificables.
TP_Recurso	Contiene la información de cada uno de los recursos que están disponibles en el sitio.
TB_TipoRecurso	Contiene los tipos de recursos disponibles para integrar al tipo de recurso.
TR_UsoRecurso	Contiene la información de la disponibilidad de un recurso.
TR_Calificauser	Contiene la información de las calificaciones de los usuarios acerca de un recurso.

6.1.5 Modelo de Procesos del Sistema: Para una mejor interpretación de los modelos de los procesos del sistema, se realizaron los diagramas de secuencias necesarios para cada caso de uso, en los que se explica con detalle los pasos para el funcionamiento de cada uno de los servicios.

6.1.5.1 Servicio de Quices y Encuestas en el Aula Virtual.

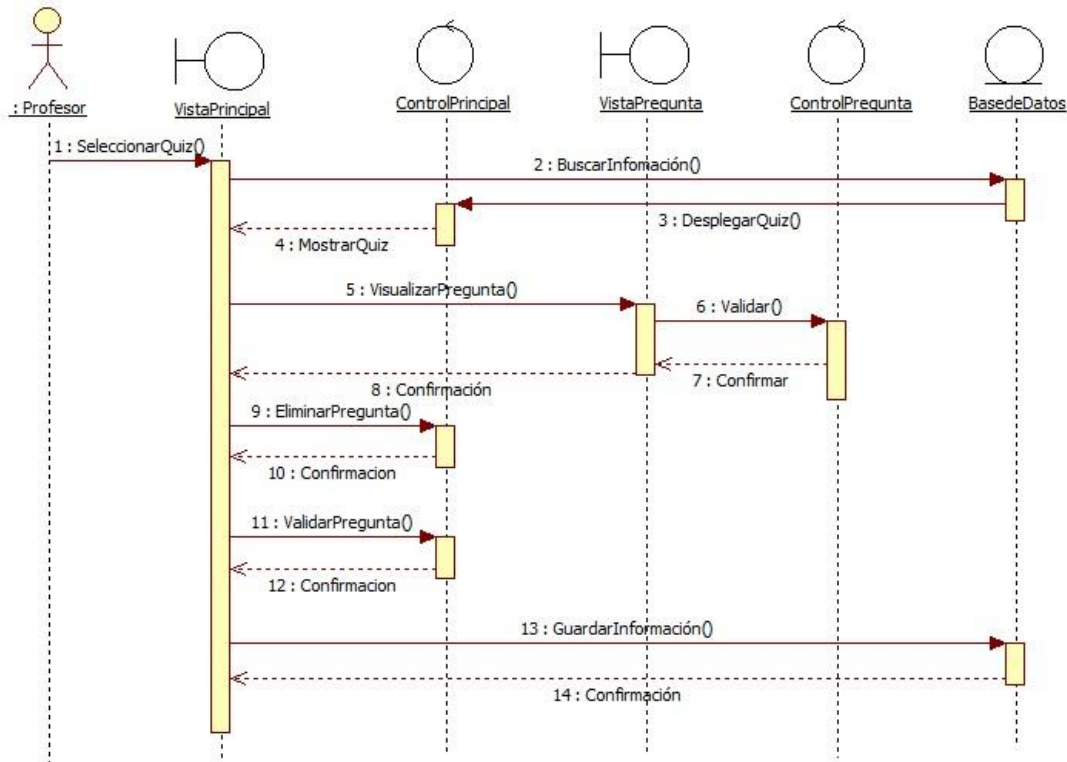
Figura 16. Diagrama de secuencia: Gestionar Quices - Crear Quiz



Fuente: Autor.

1. El profesor ingresa los datos del quiz en la vista principal.
2. El profesor se dispone a crear una pregunta, esto realiza una solicitud a la interfaz de Vista pregunta la cual se despliega.
3. Este crea la pregunta con los datos requeridos, se envía la información al controlador de las preguntas, el cual valida la información.
4. El controlador de pregunta responde a la vista principal agregando la pregunta en la tabla correspondiente, este procedimiento se realiza en repetidas ocasiones si es necesario.
5. Cuando el usuario así lo dispone guarda el quiz, su información es validada por el controlador principal.
6. El controlador principal envía la respuesta a la vista principal y se remite la información a la base de datos para que sea almacenada.
7. La base de datos confirma el proceso y se devuelve un mensaje de confirmación a la interfaz principal, de esta forma termina el proceso.

Figura 17. Diagrama de secuencia: Gestionar Quiz - Editar Quiz

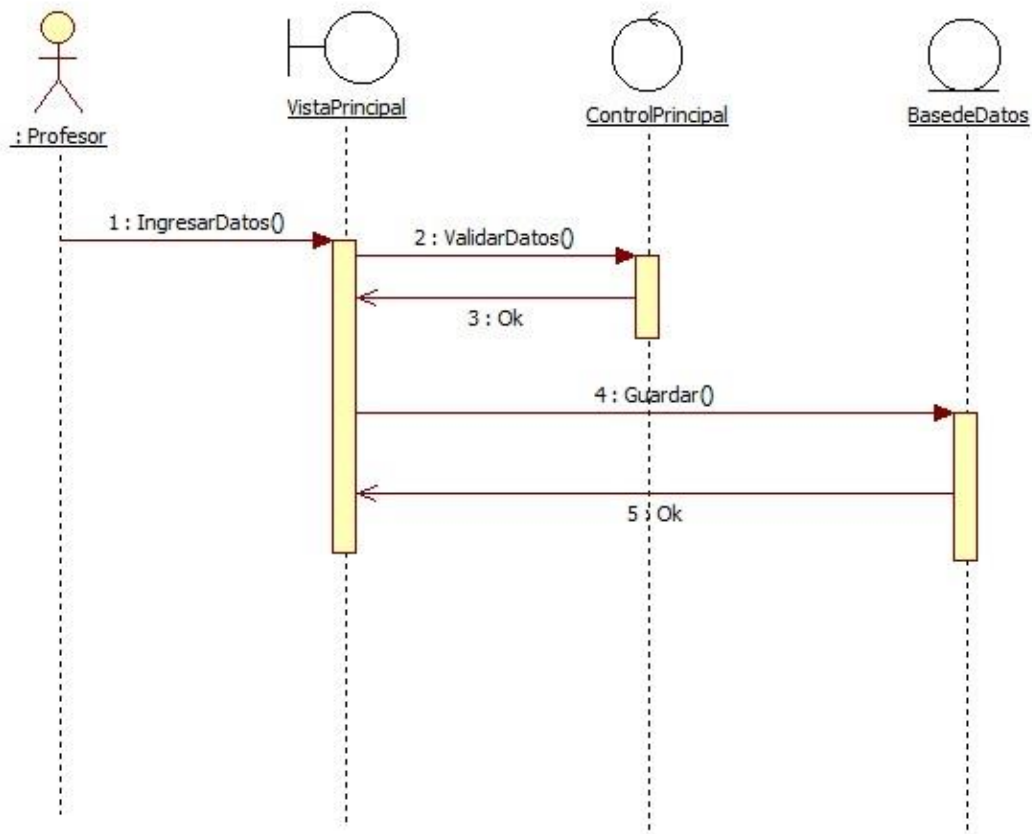


Fuente: Autor.

1. El profesor selecciona el Quiz que desea editar.
2. Esté al ingresar a la interfaz Vista Principal realiza una solicitud a la base de datos para cargar la información del quiz correspondiente.
3. La base de datos despliega la información del quiz, este es enviado al controlador principal.
4. El controlador principal se encarga de mostrar el quiz para que de esta forma pueda ser modificado.
5. Desde la Vista Principal se pueden realizar diferentes procedimientos sobre el quiz, el primero de ellos es editar una pregunta por lo cual se realiza la solicitud a la Vista Pregunta con el fin de cargarla.
6. El usuario realiza los cambios y esta entra a ser validada por el controlador de de pregunta.
7. El Controlador Pregunta envía la información a la interfaz principal.
8. El usuario Selecciona una de las preguntas y la elimina, esto realiza una confirmación en el Controlador Principal.

9. Dependiendo la respuesta del mismo esta se envía a la Vista Principal.
10. Al terminar el usuario debe guardar la información, esta se valida en el controlador principal.
11. La base de datos responde al procedimiento y se confirma en la Vista principal.

Figura 21. Diagrama de secuencia: Gestionar Quiz - Activar Quiz

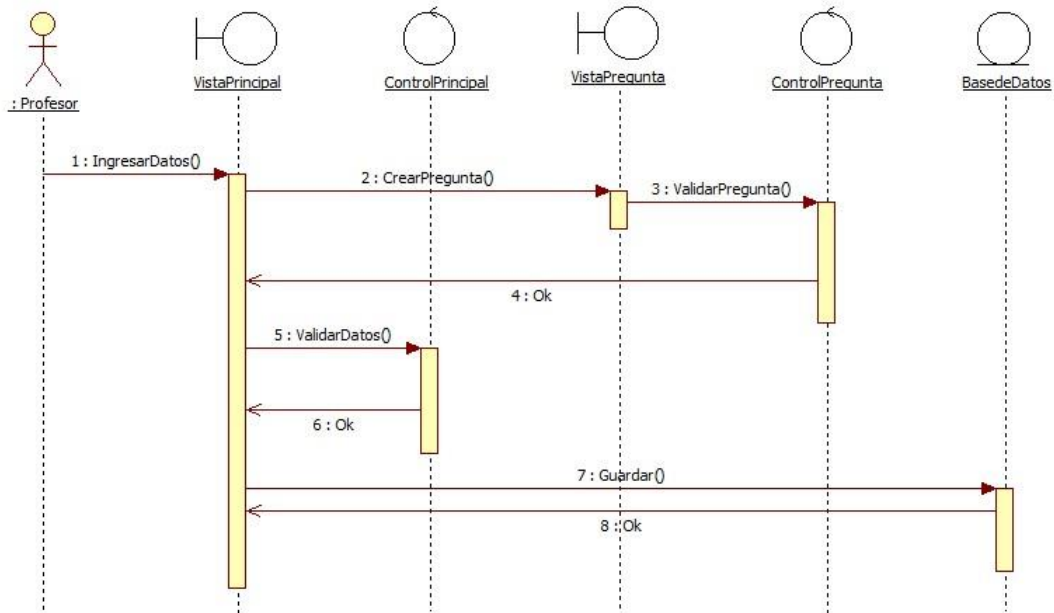


Fuente: Autor.

1. El Profesor selecciona el quiz, el cual despliega la Vista Principal de activar un quiz.
2. Se envía la información correspondiente a la operación, al controlador principal.
3. Este controlador valida la información, y responde a la Vista Principal.
4. El usuario guarda la información, este procedimiento envía los datos a la base de datos.
5. Se realiza la confirmación por medio de la base de datos y se envía un

mensaje a la Vista Principal.

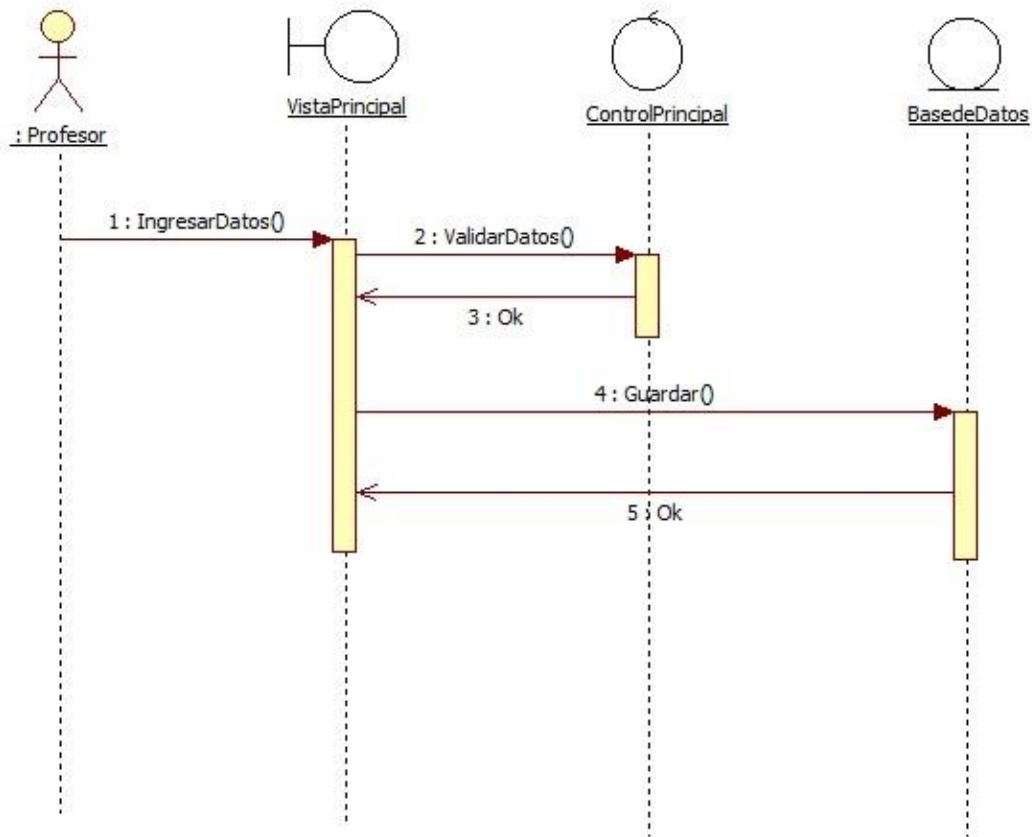
Figura 19. Diagrama de secuencia: Gestionar Encuestas - Crear Encuesta



Fuente: Autor.

1. El profesor ingresa los datos de la encuesta en la vista principal.
2. El profesor se dispone a crear una pregunta, esto realiza una solicitud a la interfaz de Vista pregunta la cual se despliega.
3. Esté crea la pregunta con los datos requeridos, se envía la información al controlador de las preguntas, el cual valida la información.
4. El controlador de pregunta responde a la vista principal agregando la pregunta en la tabla correspondiente, este procedimiento se realiza en repetidas ocasiones si es necesario.
5. Cuando el usuario así lo dispone guarda la encuesta, su información es validada por el controlador principal.
6. El controlador principal envía la respuesta a la vista principal y se remite la información a la base de datos para que sea almacenada.
7. La base de datos confirma el procedimiento y se devuelve un mensaje de confirmación a la interfaz principal, de esta manera termina el procedimiento.

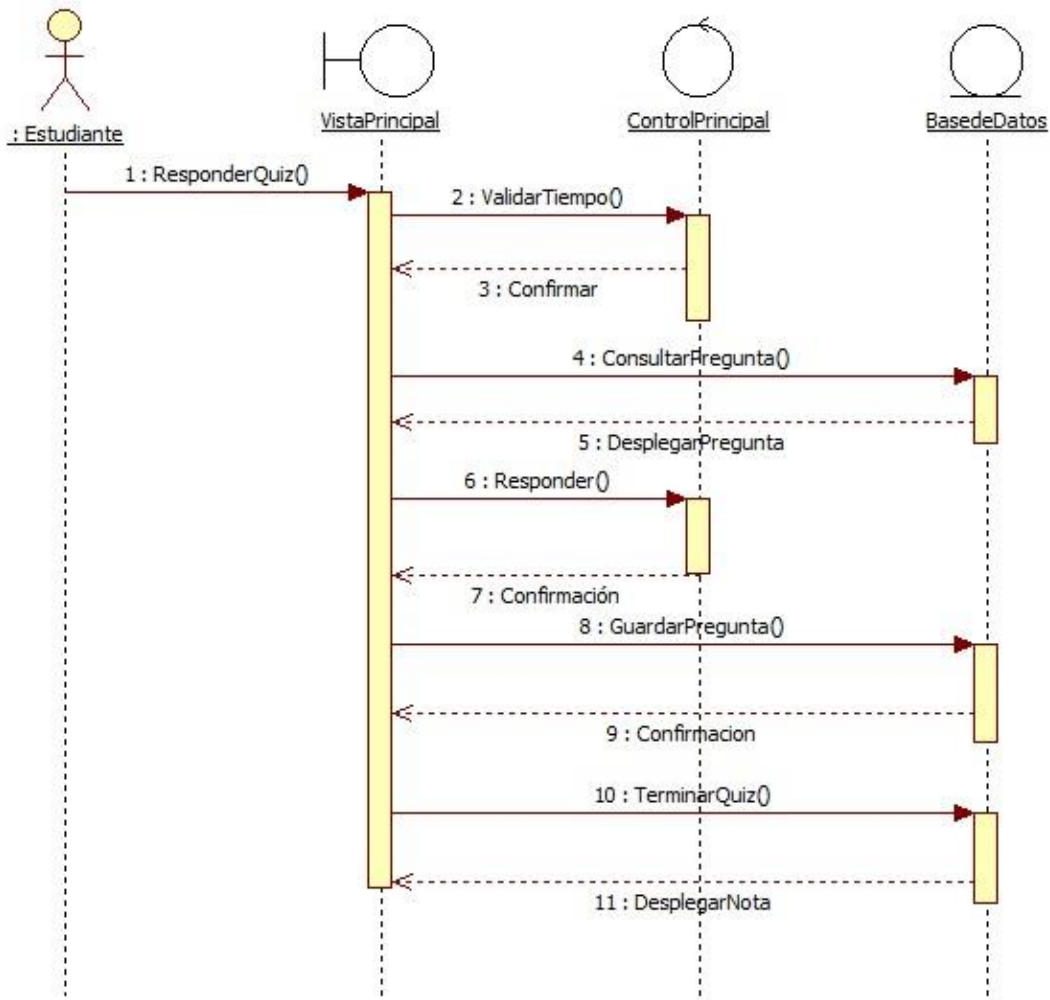
Figura 20. Diagrama de secuencia: Gestionar Encuesta - Activar Encuesta



Fuente: Autor.

1. El Profesor selecciona el quiz, el cual despliega la Vista Principal de activar la encuesta.
2. Se envía la información correspondiente a la operación, al controlador principal.
3. Este controlador valida la información, y responde a la Vista Principal.
4. El usuario guarda la información, este procedimiento envía los datos a la base de datos.
5. Se realiza la confirmación por medio de la base de datos y se envía un mensaje a la Vista Principal.

Figura 21. Diagrama de secuencia: Gestionar Quiz Estudiante - Responder Quiz



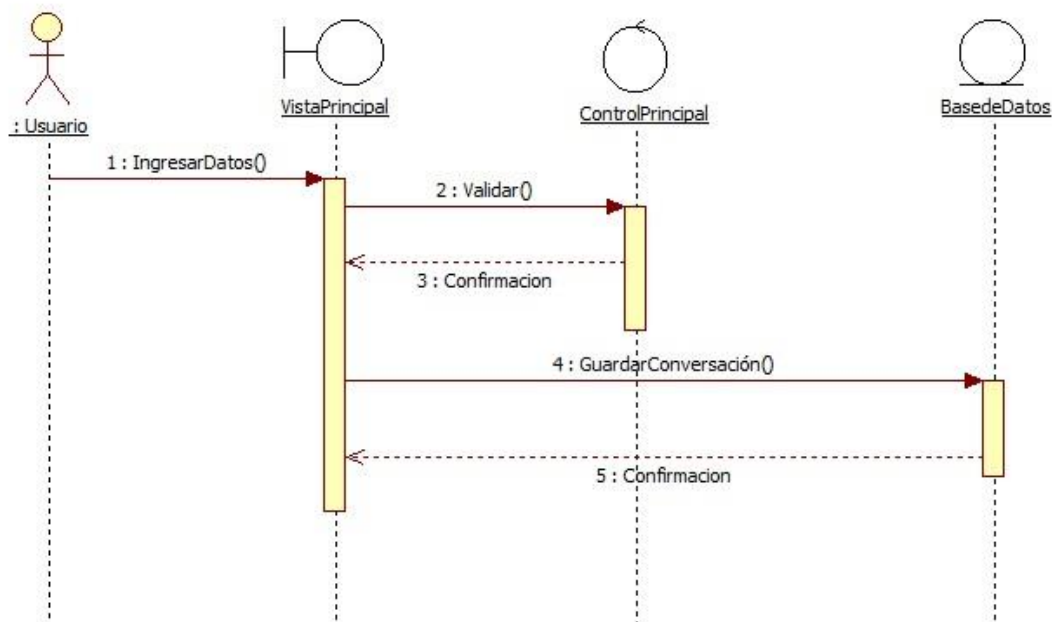
Fuente: Autor.

1. El estudiante selecciona el quiz a responder.
2. Se realiza una validación por parte del controlador principal definiendo si está vigente aún la actividad.
3. El controlador realiza la confirmación a la vista principal, la cual llama a la base de datos para traer la primera pregunta.
4. Esta consulta a la base de datos la devuelve la información y esta es mostrada en la vista principal.
5. El usuario responde la pregunta, esta se valida en el controlador principal.
6. El controlador principal confirma la operación, y esto se envía a la interfaz principal.

7. Desde esta última se envía la información a la base de datos.
8. La base de datos responde y se envía una confirmación a la Vista Principal, este procedimiento se realiza varias veces dependiendo el número de preguntas.
9. Por último, al responder la pregunta final se confirma por parte de la base de datos y devuelve la nota del quiz a la vista principal.

6.1.5.2. Servicio de Foros Evaluables y No Evaluables del Aula Virtual.

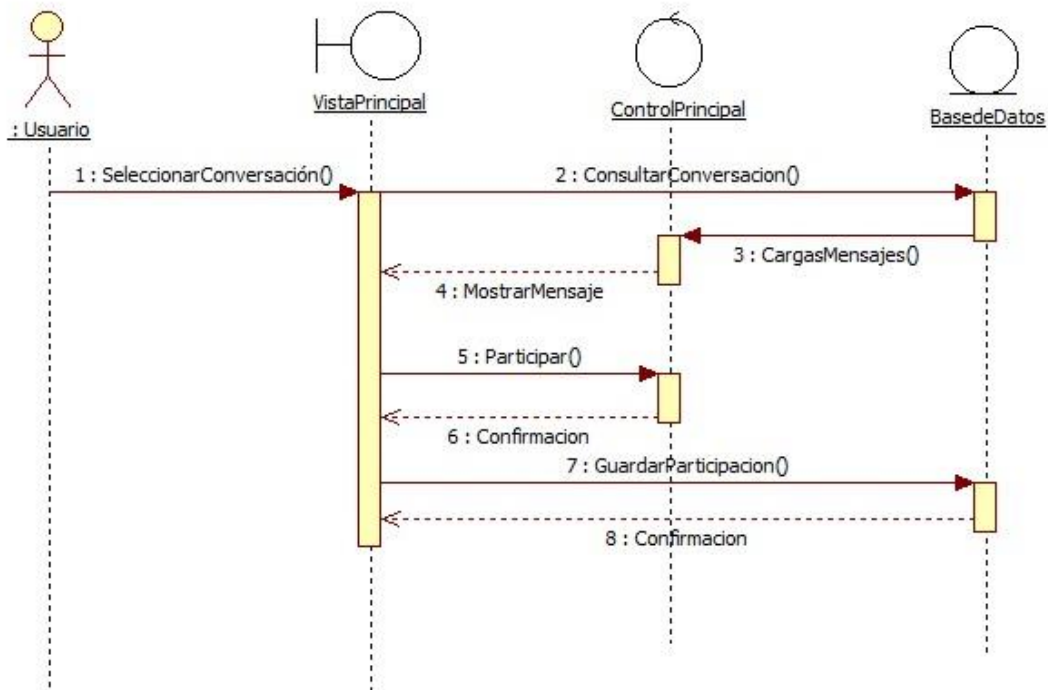
Figura 22. Diagrama de secuencia: Gestionar Foros - Crear Conversación.



Fuente: Autor.

1. El usuario ingresa los datos requeridos para crear la nueva conversación.
2. Estos datos son validados por el controlador principal.
3. La confirmación llega a hasta la interfaz principal y realiza la solicitud a la base dato para ser guardada.
4. La base de datos confirma la operación y envía un mensaje a la Vista Principal.

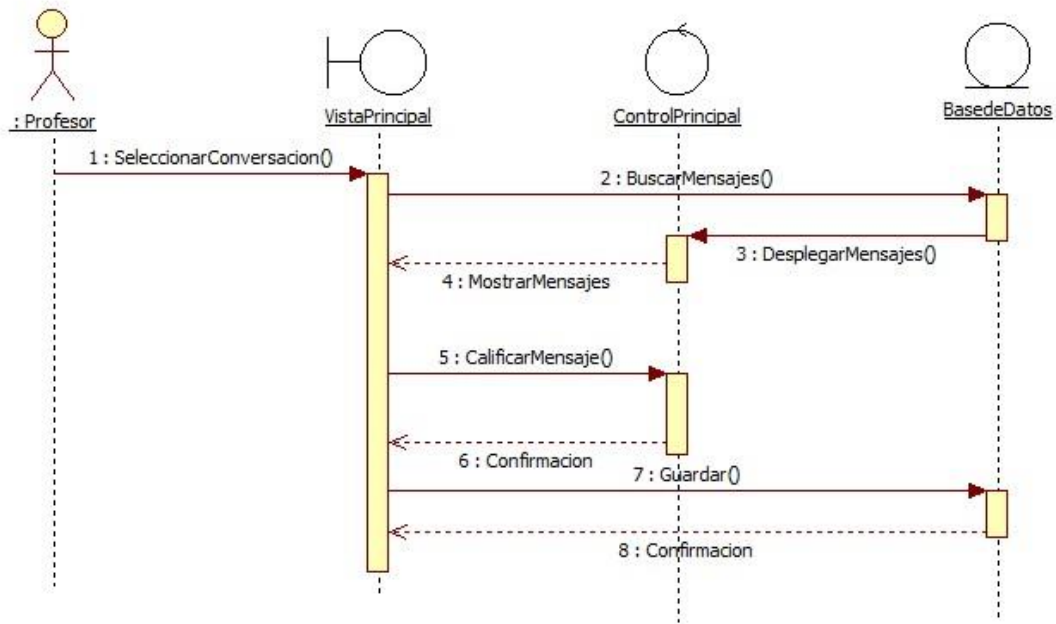
Figura 23. Diagrama de secuencia: Gestionar Foros -Participar Conversación.



Fuente: Autor.

1. El usuario selecciona la conversación en la cual desea participar.
2. Esta realiza una consulta a la base datos por medio de la cual busca los mensajes de la misma.
3. La base de datos devuelve la información esta es cargada, pero antes validad por el controlador principal.
4. Se despliegan los mensajes en la vista Principal.
5. El usuario ingresa los datos requeridos para crear la nueva conversación.
6. Estos datos son validados por el controlador principal.
7. La confirmación llega a hasta la interfaz principal y realiza la solicitud a la base dato para ser guardada.
8. La base de datos confirma la operación y envía un mensaje a la Vista Principal.

Figura 24. Diagrama de secuencia: Gestionar Foros - Calificar Mensaje.

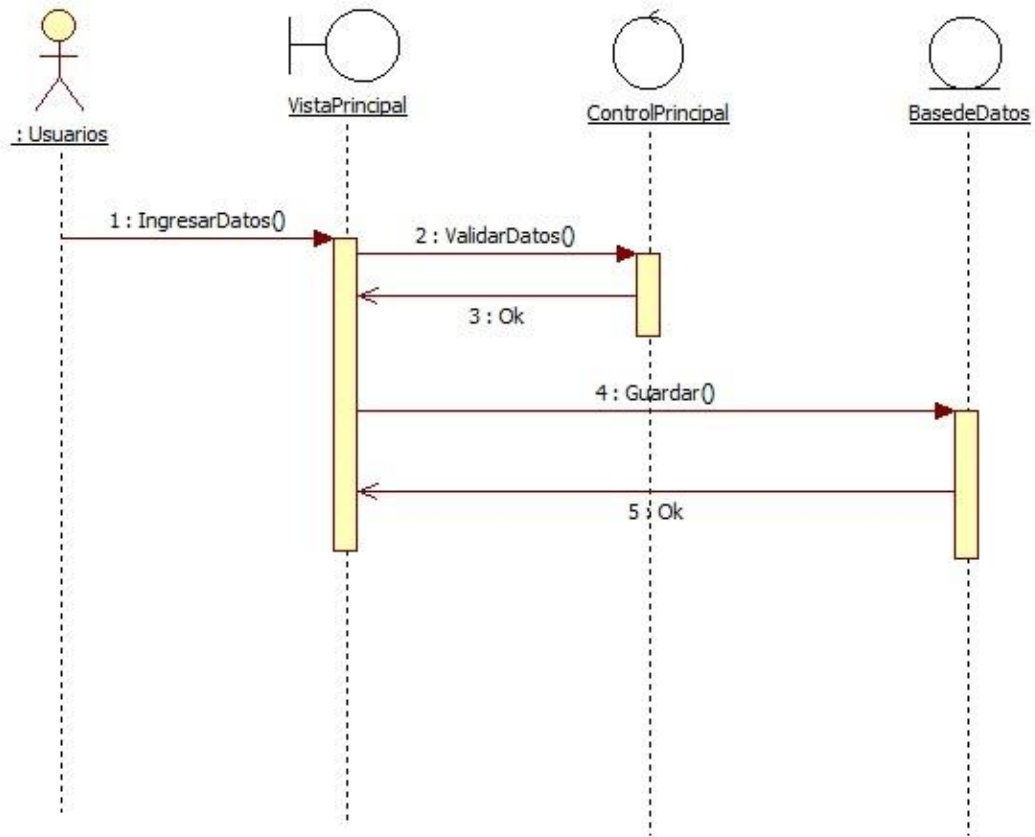


Fuente: Autor.

1. El usuario selecciona la conversación que desea calificar algún mensaje.
2. Esta realiza una consulta a la base datos por medio de la cual busca los mensajes de la misma.
3. La base de datos devuelve la información esta es cargada, pero antes validada por el controlador principal.
4. Se despliegan los mensajes en la vista Principal.
5. El usuario puntea el mensaje, esta información se valida en el controlador principal.
6. Se confirma a la Vista Principal la información, se envía esto a la base de datos, la cual responde la operación.
7. Se muestra el mensaje calificado y su respectiva puntuación en la vista principal.

6.1.5.3. Servicio de Recursos.

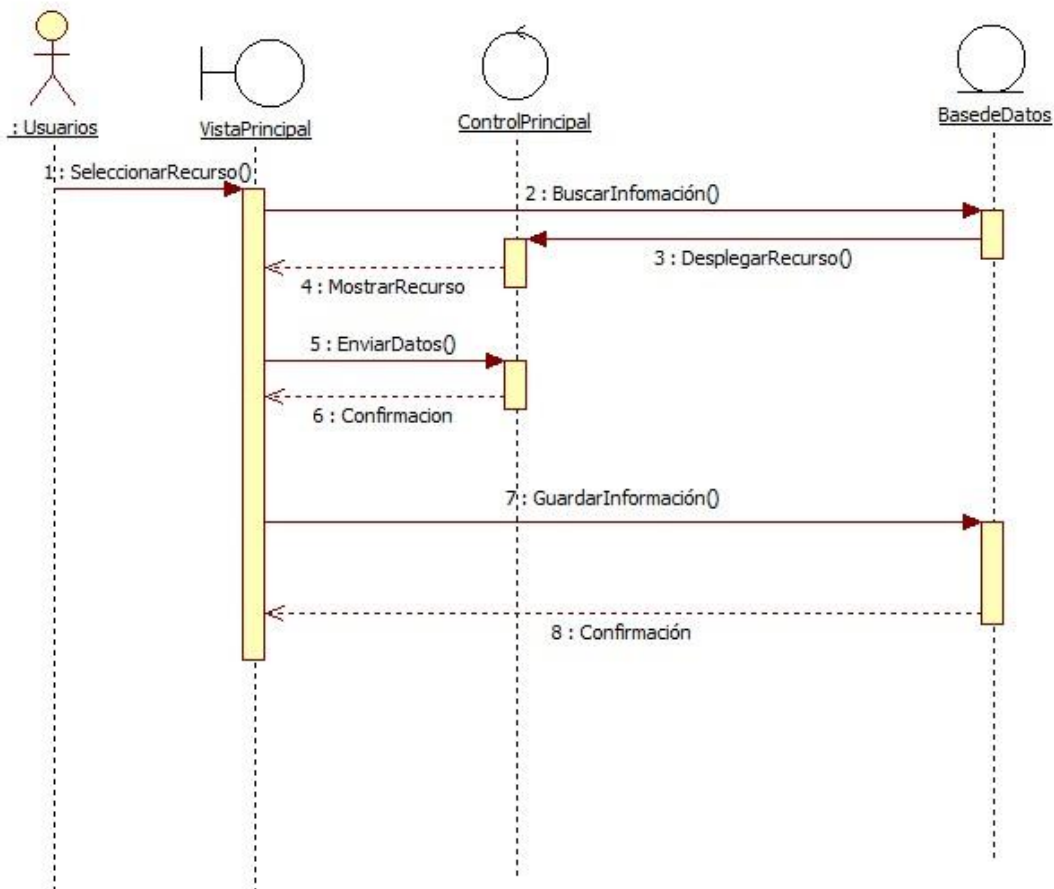
Figura 25. Diagrama de secuencia: Gestionar Recursos - Crear Recurso.



Fuente: Autor.

1. El usuario ingresa los datos requeridos para crear el nuevo recurso.
2. Estos datos son validados por el controlador principal.
3. La confirmación llega a hasta la interfaz principal y realiza la solicitud a la base dato para ser guardada.
4. La base de datos confirma la operación y envía un mensaje a la Vista Principal.

Figura 26. Diagrama de secuencia: Gestionar Recursos - Editar Recurso.



Fuente: Autor.

1. El usuario selecciona el recurso a editar.
2. Se busca la información en la base de datos, la cual se envía al controlador principal.
3. El controlador Principal válida y muestra los datos en la Vista Principal.
4. Se envían los datos, el controlador principal confirma los datos de la edición del recurso.
5. La información se guarda en la base de datos.
6. Se confirma la información y se muestra un mensaje en la interfaz principal.

6.1.6 Implementación, Implantación y Pruebas Generales: Para la implementación de los servicios se utilizaron las siguientes herramientas:

- Lenguaje de marcado para la elaboración de páginas web, HTML.
- Lenguaje de programación orientado a la web, JSP.
- Lenguaje de programación interpretado orientado a objetos, JavaScript.
- Hojas de estilo en cascada, CSS.
- Lenguaje Java.
- NetBeans, IDE para desarrollar las clases de Java y los archivos JSP.
- Servidor Jakarta Tomcat.
- Manejador de Base de datos, MySQL 5.0.

Haciendo uso de las anteriores herramientas, y con la asesoría y seguimiento del director de proyecto se diseñaron los nuevos servicios y se estructuraron los datos para el primer prototipo. También se tomaron en cuenta sugerencias recibidas por miembros del Grupo Calumet, para realizar un posterior refinamiento a las interfaces.

Para el desarrollo se trabajó con la base de datos "Diamante" ubicada en todos los servidores de los portales de las escuelas. En esta se crearon las tablas necesarias para el funcionamiento de los servicios.

Se llevaron a cabo pruebas para cada subsistema propuesto, verificando que los resultados correspondieran con lo esperado, de esta forma, se evidenció el correcto funcionamiento en la captura de datos, selección de ítems y almacenamiento de información.

La implantación de los servicios se realizó en primer lugar en el portal web de pruebas del Grupo Calumet, donde se realizan las pruebas a todos los componentes. Finalmente se implementan en todos los portales web a los cuales el Grupo Calumet presta soporte.

6.2 MANTENIMIENTO Y ADMINISTRACIÓN

6.2.1 Actividades de Mantenimiento: Dentro de las funciones que se realizan en la administración y mantenimiento de los servicios de los portales web se encuentra la tarea de corregir errores, las cuales se denominan incidencias y que se corrigen durante la primera fase como integrantes del Grupo Calumet. Las siguientes son las incidencias que se realizaron paralelamente al desarrollo de este proyecto.

- **En trabajos de grado, agregar dos filtros adicionales para realizar las búsquedas tanto en el público como en el privado(Buscador).**

Funcionalidad:

El sistema de trabajos de grado permite a cualquier usuario de la comunidad buscar información acerca de un proyecto de grado.

Problemática:

El buscador no contiene todos los filtros, necesarios para una búsqueda más eficiente y específica, por lo tanto, se ve la necesidad de agregarle un nuevo filtro tanto en el buscador público como en el privado.

Desarrollo del Problema:

Solucionando la problemática, se implementó dos nuevos filtros, buscar por tipo de proyecto, postgrado o pregrado, esto fue desarrollado tanto en el buscador privado, para usuarios logueados como para usuarios ajenos a la comunidad.

- **En trabajos de grado, crear la opción para el comité de poder aplazar, rechazar o aprobar y agregar comentarios en el menú de asignar evaluador.**

Funcionalidad:

El sistema de trabajos de grado permite dentro de la sesión de un comité realizar modificaciones a los estados de los proyectos de grado.

Problemática:

Durante las sesiones de comité, los profesores no tienen la posibilidad de aplazar, rechazar o aprobar el proyecto, estando en ciertos estados, por lo cual se encuentra la necesidad de agregar estas opciones dentro del módulo, además de esto permitiendo agregar un comentario para mayor claridad al respecto.

Desarrollo del Problema:

Solucionando la problemática, se implementó las nuevas opciones para el comité y se agregó el campo para el comentario, para contener la información en la respectiva decisión.

- **Mostrar los objetivos específicos en el buscador de proyectos.**

Funcionalidad:

El sistema de trabajos de grado tiene la opción de búsqueda de los mismos, estos en el momento de realizar la búsqueda muestra cierta información del proyecto.

Problemática:

Los objetivos específicos del proyecto no se pueden visualizar de ninguna manera dentro de los buscadores (públicos o privados).

Desarrollo del Problema:

Solucionando la problemática, se implementó el nuevo campo en la interfaz de los proyectos de grado, esto se llevó a cabo en el buscador privado, dando así la posibilidad de conocer los objetivos específicos de cada uno de los proyectos.

6.2.3 Actividades de Administración: Dentro de las actividades realizadas por los miembros del Grupo Calumet, se encuentra la tarea de administrar los servidores de los portales de las escuelas, cada semestre se asignan nuevos administradores, los cuales cuentan con un usuario dentro de uno de los portales a los cuales presta soporte el grupo. Como administrar las tareas que se realizan comúnmente son:

- Dar aval a las solicitudes de publicación de los usuarios en la cartelera para que puedan ser accedidas por la comunidad.
- Atender las sugerencias hechas por los usuarios del sistema a través del servicio de consultas y sugerencias.
- 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.

7. PRUEBAS DEL SISTEMA

Para garantizar el correcto desarrollo de los servicios realizados, se realizaron las siguientes pruebas:

7.1 PRUEBAS DE VERIFICACIÓN

Esta prueba es una de las más prácticas y más utilizadas mediante la cual se aplican diferentes técnicas para detectar errores en el sistema antes de ser utilizado.

Se efectúa elaborando actividades para cada servicio desarrollado, realizando pruebas como: verificar que los campos que contienen datos obligatorios no queden vacíos, o examinar que los campos contengan el tipo de dato que le corresponde.

7.1.1. Pruebas por componente: Esta prueba se realizó para cada caso de uso de cada servicio desarrollado y descrito anteriormente. De igual manera en las mejoras a los servicios ya existentes. Los resultados obtenidos con la realización de las pruebas, evidencian el cumplimiento de los objetivos propuestos, en la medida en que los servicios y mejoras funcionan correctamente.

Tabla 11. Pruebas Realizadas: Gestionar Quiz.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Gestión Quiz	Acceder al servicio de quices dentro del aula virtual, listando los quices existentes en su respectivo estado.	✓
Crear Quiz	Cuando se crea el quiz se validan los campos pertinentes y se hace la	✓

	inserción en la base de datos.	
Editar Quiz	Después de hacer las validaciones, edita el quiz y lo guarda en la base de datos.	✓
Eliminar Quiz	Elimina correctamente el quiz y elimina las notas dependiendo el caso.	✓
Activar Quiz	Activa correctamente el quiz validando las fechas y los grupos disponibles.	✓
Reiniciar Quiz	Reactiva el quiz, modificando el tiempo en línea del mismo de forma adecuada.	✓
Individualizar Nota	Las notas se individualizan correctamente, llenando las tablas en la base de datos y enviando los correos pertinentes.	✓

Fuente: Autor.

Tabla 12. Pruebas Realizadas: Gestión quices estudiante.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Consultar Quices	Accede al menú que lista los quices dependiendo de su estado dentro del aula correctamente.	✓
Responder Quiz	El usuario puede entrar y responder el quiz, se guarda en la base de datos y retorna de forma adecuada los datos pertinentes.	✓
Consultar Nota	El estudiante consulta las notas de los quices y estas son retornadas de la manera adecuada.	✓

Fuente: Autor.

Tabla 13. Pruebas Realizadas: Gestionar Encuestas.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Gestionar Encuestas	Accede al servicio que lista las encuestas dependiendo su estado dentro del aula además de múltiples opciones, el listado y los botones de la interfaz funcionan de la manera adecuada.	✓
Crear Encuesta	Crea las encuestas en el aula, validando los datos y haciendo la respectiva inserción en la base de datos.	✓
Editar Encuesta	Edita la encuesta, haciendo las validaciones pertinentes y agregando la información a la base de datos.	✓
Eliminar Encuesta	Elimina la encuesta correctamente, efectuando una confirmación y después realizando la eliminación pertinente en la base de datos.	✓
Activar Encuesta	Activa la encuesta, validando el rango de fechas y guarda exitosamente en la base de datos.	✓
Reiniciar Encuesta	Reinicia la encuesta adecuadamente, borrando los datos anteriores y validando las nuevas fechas.	✓
Ver Resultados	El profesor visualiza correctamente las tablas y los diferentes diagramas con la información de la encuesta.	✓

Fuente: Autor.

Tabla 14. Pruebas Realizadas: Gestión encuestas estudiantes.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Consultar Encuesta	Accede al servicio que lista las	✓

	encuestas dependiendo su estado dentro del aula, este proceso se realiza adecuadamente, además de esto los botones dentro del menú funcionan correctamente.	
Responder Encuesta	Se realiza una encuesta con varias preguntas y diferentes tipos de diseños con el fin de probar su buen funcionamiento.	✓
Ver Resultados	Permite al estudiante ver los diferentes resultados de las encuestas, se prueba que dependiendo la decisión del profesor de mostrar o no los resultados, sea acorde a las opciones que se muestran para el usuario.	✓

Fuente: Autor.

Tabla 15. Pruebas Realizadas: Gestión Foros Evaluables.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Gestionar Foros	Accede al servicio que lista las foros dependiendo de su estado, los lista y nos muestra las opciones de la manera adecuada.	✓
Cambiar Fechas	Accede a cambiar las fechas de la actividad (Foro), realizando las validaciones respectivas y su envío a la base de datos.	✓
Crear Conversación	Crea una conversación en el aula, validando la información necesaria.	✓
Activar Conversación	Activa las conversaciones dentro del aula, valorando la pertinencia de las fechas.	✓
Calificar Conversación	Califica los mensajes adecuadamente y realiza la respectiva inserción en la base de datos.	✓

Participar Conversación	Se crea una participación en alguna de las conversaciones, validando los campos necesarios.	✓
Evaluar Mensaje	Se evaluaron algunos de los mensajes y se realizan de la manera adecuada, mostrando la información pertinente.	✓
Ver Archivo	Ingresamos a ver los archivos que los diferentes grupos de estudiantes deben subir y se muestran de la manera adecuada.	✓

Fuente: Autor.

Tabla 16. Pruebas Realizadas: Gestión Foros No Evaluables.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Consultar Foros	Accede al servicio que lista las foros dependiendo su estado, los lista y nos muestra las opciones de la manera adecuada.	✓
Participar Conversación	Se crea una participación en alguna de las conversaciones, validando los campos necesarios.	✓
Evaluar Mensaje	Se evaluaron algunos de los mensajes y se realizan de la manera adecuada, mostrando la información pertinente.	✓
Crear Conversación	Crea una conversación en el aula, validando la información necesaria.	✓

Fuente: Autor.

Tabla 17. Pruebas Realizadas: Gestión Foros estudiantes Evaluables.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Consultar Foros	Accede al servicio que lista los foros dependiendo su estado, los lista y nos muestra las opciones de la manera adecuada.	✓
Participar Conversación	Se crea una participación en alguna de las conversaciones, validando los campos necesarios.	✓
Evaluar Mensaje	Se evaluaron algunos de los mensajes y se realizan de la manera adecuada, mostrando la información pertinente.	✓
Subir Archivo	Permite subir adecuadamente un archivo con las especificaciones obligatorias.	✓

Fuente: Autor.

Tabla 18. Pruebas Realizadas: Gestión Foros estudiantes No Evaluables.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Consultar Foros	Accede al servicio que lista los foros dependiendo su estado, los lista y nos muestra las opciones de la manera adecuada.	✓
Participar Conversación	Se crea una participación en alguna de las conversaciones, validando los campos necesarios.	✓
Evaluar Mensaje	Se evaluaron algunos de los mensajes y se realizan de la manera adecuada, mostrando la información pertinente.	✓
Crear Conversación	Crea una conversación en el aula, validando la información	✓

	necesaria.	
--	------------	--

Fuente: Autor.

Tabla 19. Pruebas Realizadas: Recursos.

TÍTULO	PRUEBA REALIZADA	RESULTADO
Gestionar Recursos	Accede al menú donde se encuentran todos los recursos con sus respectivas opciones, listando de la manera adecuada los recursos en sus diferentes estados.	✓
Crear Recurso	Crear un nuevo recurso, validando los datos necesarios y guardando en la base de datos.	✓
Asignar Recurso	Se le asigna un recurso a alguno de los usuarios interesados de manera correcta, además se envía el email pertinente.	✓
Editar Recurso	Edita de manera adecuada el recurso y valida la información necesaria para su validación.	✓
Gestionar Solicitudes	Genera una solicitud de manera adecuada validando los campos necesarios y realizando el envío del correo y su respectiva inserción en la base de datos.	✓
Crear Solicitud	Crear una solicitud para un recurso validando las fechas y su disponibilidad en las mismas de la forma esperada.	✓
Calificar Usuario	Calificar un usuario dependiendo su comportamiento con alguno de los recursos, validando la información y guardando en la base de datos.	✓
Ver Calificaciones	Visualiza las diferentes calificaciones de los recursos del usuario de la manera adecuada.	✓

7.2 PRUEBAS DE INTEGRACIÓN

Una vez creados todos los servicios se verificó que cada uno funcionara correctamente y no interfirieran con el funcionamiento del portal. Se verificó que las consultas a la base de datos se hicieran de manera correcta, que en el sistema se visualizarán correctamente los servicios y que cada tipo de usuario tuviera acceso a ellos dependiendo del perfil y categoría al que está asociado.

7.3 PRUEBAS DE VALIDACIÓN

Con el fin de probar el funcionamiento de cada servicio implementado se realizaron pruebas de validación elaboradas por integrantes del grupo Calumet, administrativos de las escuelas y el director de proyecto. Para algunos de los servicios observamos irregularidades las cuales fueron corregidas durante la marcha hasta obtener los resultados requeridos.

Las consultas a la base de datos se hicieran de manera correcta, que en el sistema se visualizarán correctamente los servicios y que cada tipo de usuario tuviera acceso a ellos dependiendo del perfil y categoría al que está asociado.

8. CONCLUSIONES

- Se implementó el servicio de quices dentro del aula virtual el cual permite a los profesores evaluar de una manera rápida y precisa a sus estudiantes, integrándolo con el manejo de notas y de planes con los que contaba el aula.
- Se creó el servicio de encuestas dentro del aula virtual el cual permite a los profesores realizar consultas sobre cualquier temática dentro del aula, mostrando los resultados en forma detallada gracias al uso de diagramas y tablas.
- Con la reingeniería al servicio de recursos se generan nuevos espacios para crear comunidad, dando a los usuarios una plataforma eficiente para el préstamo de recursos.
- Con la nueva implementación de los foros en el aula virtual se da la posibilidad de generar nuevos y mejorados canales de comunicación y aprendizaje mutuo entre los diferentes usuarios del aula.
- Re-implementando el servicio de foros evaluables dentro del aula virtual se logró un canal de comunicación profesor- estudiante más eficiente y seguro, además de esto se brinda una forma diferente de aprendizaje de las nuevas temáticas, resaltando la importancia de las participaciones dentro de los procesos académicos.

9. RECOMENDACIONES

- Para lograr un verdadero aprovechamiento de los nuevos servicios del aula virtual se debe hacer una pronta divulgación de los beneficios de esta plataforma.
- Se recomienda el mejoramiento de algunas de las interfaces del aula virtual, que cumplen un muy buen servicio pero que poco a poco se han venido quedando atrás y que de esta manera generan en algunos casos desconfianza sobre los procesos que se efectúan dentro de esta.
- Se debe dar a conocer el servicio de recursos, el cual no cuenta aún con el reconocimiento necesario dentro de la comunidad para que sea realmente aprovechado.

BIBLIOGRAFÍA

Cómo extender y profundizar en herramientas para crear aplicaciones web front-end avanzadas. [en línea] [Bucaramanga, Colombia]: Elise Library V2. noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#liberias>

Componentes de interfaz de usuario dinámica e interactiva para crear aplicaciones. [en línea] [Bucaramanga, Colombia]: Elise Library V2. noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#liberias>

Estilos y diseños básicos y complementarios para crear componentes de la Librería Elise. [en línea] [Bucaramanga, Colombia]: Elise Library V2. noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/design.jsp>

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.

GISSET. Información acerca de la Arquitectura Cliente Servidor. [en línea]. <http://g701giadar.wikispaces.com/page/diff/Arquitctura+Cliente+Servidor/22996+8658> [citado el 05 Octubre de 2015]

Guía de documentación de selectores, eventos, manipulación y procedimientos de jQuery. [en línea] [Bucaramanga, Colombia]: Elise Library V2. noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet: <http://www.oscarotero.com/jquery/>

Herramientas adicionales en el proceso de testear y codificar servicios y aplicaciones. [en línea] [Bucaramanga, Colombia]: Elise Library V2. noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#otros>

Herramientas JavaScript y JQuery para la creación de procedimientos para aplicaciones [en línea] [Bucaramanga, Colombia]: Elise Library V2. noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/codes.jsp>

Herramientas para facilitar y agilizar la creación de diseños para objetos HTML con CSS. [en línea] [Bucaramanga, Colombia]: Elise Library V2. noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#librerias>

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.

ORACLE CORPORATION. MySQL Documentation: MySQL Reference Manuals. [en línea]. <http://dev.mysql.com/doc/> [citado el 05 Octubre de 2015]

ORACLE. Api de Java [en línea]. <http://docs.oracle.com/javase/7/docs/api/> citado el 05 Octubre de 2015]

PIATTINI, Mario, CALVO-MANZANO, José A., CERVERA, Joaquín, Plantillas estándar para crear servicios, aplicaciones o cualquier proyecto para Calumet. [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet:

PONCE DE LEÓN, Diego. Tutoriales HTML. [en línea]. <http://www.htmlquick.com/es/tutorials.html> [citado el 05 Octubre de 2015]

RICH, Clark, et al. Html element index [online]. Available from Internet: <http://html5doctor.com/element-index/> [citado el 05 Octubre de 2015]

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.

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.

STUMPF, Robert, TEAGUE, Lavette. Object-Oriented Systems Analysis and Design with UML. Prentice Hall. 2004. Este libro introduce los conceptos y métodos del análisis y diseño de sistemas orientados

a objetos.

UNIVERSIDAD INDUSTRIAL DE SANTANDER. Grupo Desarrollador Calumet. Algunas herramientas de recursividad y utilidad para cuando se codifica JavaScript. [en línea] [Bucaramanga, Colombia]: Elise Library V2. noviembre 2014. [citado el 05 octubre de 2015]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#js>

WEITZENFELD, Alfredo. Ingeniería de Software Orientada a Objetos con UML, JAVA e Internet. Thomson International, 2003. En este libro se encuentra información sobre desarrollo de software orientado a objetos.

ANEXOS

Anexo A. ARQUITECTURA CLIENTE/SERVIDOR

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. 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: Presentación de resultados al usuario de forma comprensible.
- 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 persistentes de la aplicación.

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

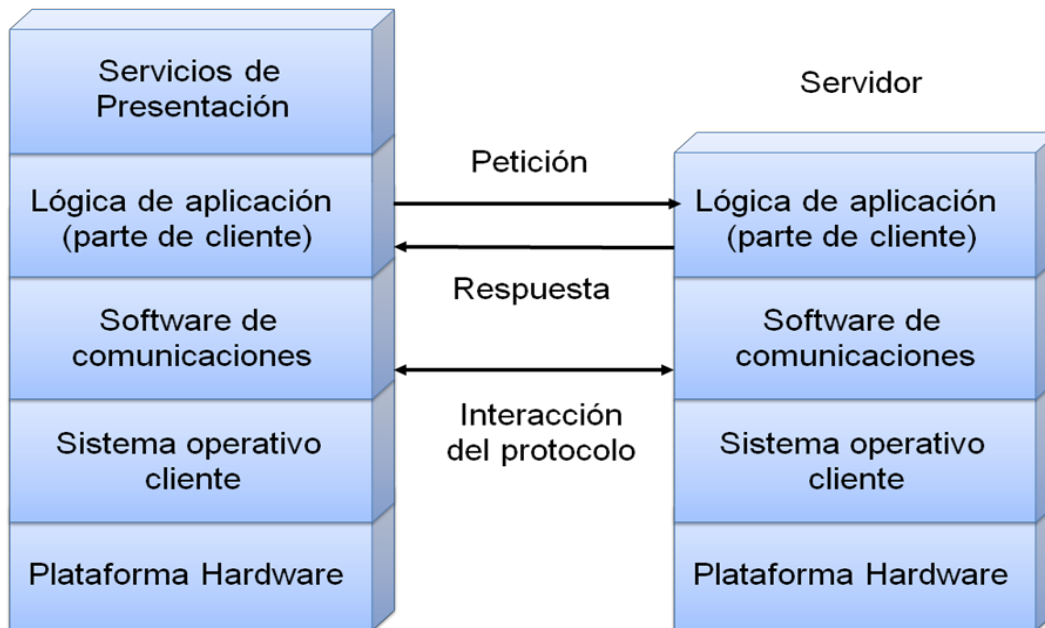
1.1. ARQUITECTURA CLIENTE/SERVIDOR DE DOS CAPAS

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 requiere 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 27. Arquitectura Cliente/Servidor de Dos Capas

Estación Trabajo Cliente



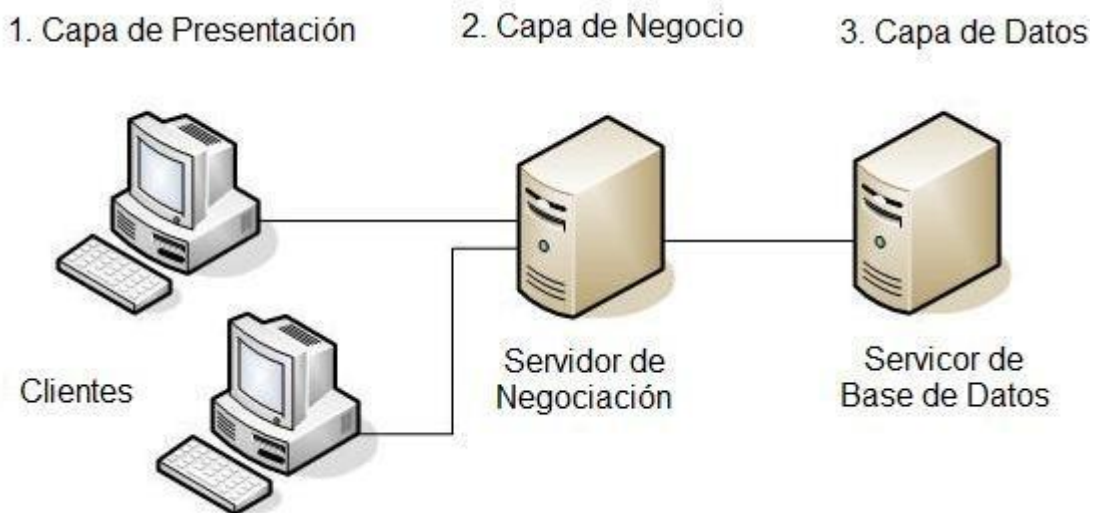
1.2. ARQUITECTURA CLIENTE/SERVIDOR DE TRES CAPAS

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 28. Arquitectura Cliente/Servidor de Tres Capas



Fuente: Arquitectura Cliente/Servidor de tres capas. [en línea]. http://www.ecured.cu/index.php/Archivo:Tres_capas.JPG [citado el 10 octubre de 2015]

Anexo B.

TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS

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. 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.

Anexo C. BASES DE DATOS

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 consultas y actualizaciones. Las bases de datos se pueden clasificar de acuerdo a su modelo de administración de datos. Algunos modelos utilizados con frecuencia son:

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.

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.

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. 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.

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.

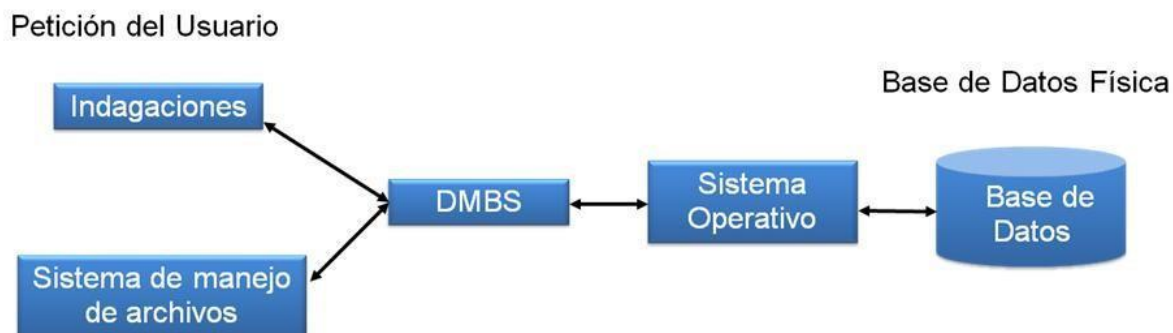
4. Manejadores o Gestores de Bases de Datos

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 29. Función del DBMS.



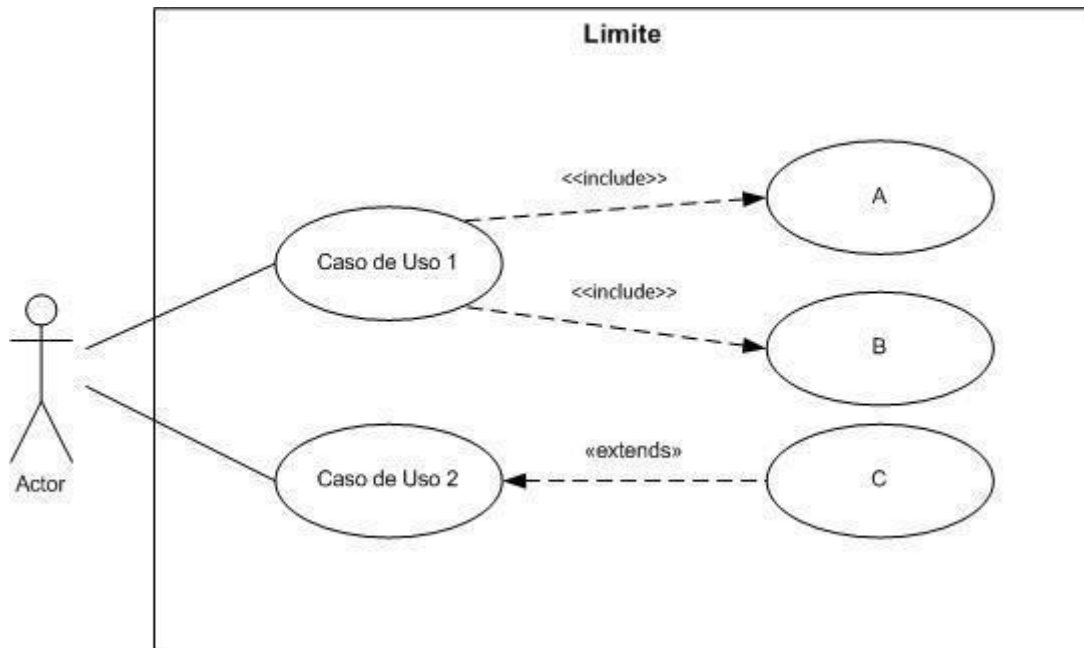
Fuente: Función del DBMS [en línea].
<http://www.ecured.cu/index.php/Archivo:DBMS.JPG> [citado el 10 Octubre de 2015]

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.

Anexo D. LENGUAJE DE MODELADO UNIFICADO (UML)

1. Diagramas de casos de uso

Figura 30. Diagramas de Casos de Uso



Fuente: Diagrama de casos de uso [en línea].
<http://www.ecured.cu/index.php/Archivo:CasosDeUso.JPG> [citado el 10 Octubre de 2015]

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 las siguientes 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.

- Usuario: Es el tipo de usuario común de los portales de las escuelas a quien van dirigidos los servicios. Este usuario solo tiene control sobre sus privilegios.
- Profesor: Son usuarios que cuentan con ciertos privilegios, por lo cual tienen acceso a determinados servicios, y a realizar ciertas acciones.
- Sistema: Es el sistema encargado de realizar determinadas funciones, que le solicita el usuario.
- Público: Personas que ingresan a los portales, sin necesidad de registrarse solo tienen permisos públicos.
- Encargado del grupo: Usuario que pertenece a un grupo de investigación, desarrollo u otro determinado grupo, pero es el encargado de administrar el portal del grupo.
- Integrante del grupo: Usuario que pertenece a un grupo de investigación, desarrollo u otro determinado grupo.

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 "Caso de uso 1" depende de los casos de uso "A" y "B".

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.

2. Diagramas de secuencias

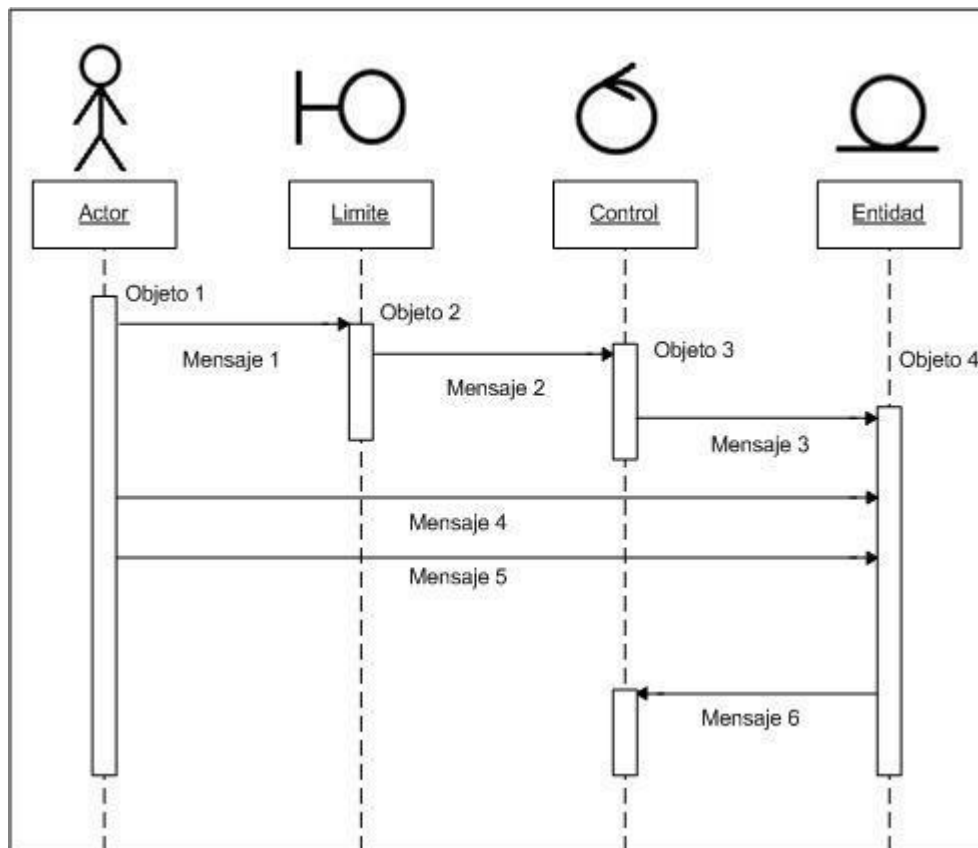
Objetos: Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de cada objeto será una línea discontinua conocida como la *línea de vida* de un objeto. Con la línea de vida se encuentra un pequeño rectángulo conocido 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 31. Diagrama de Secuencias



Fuente: Diagrama de secuencia [en línea].
<http://www.ecured.cu/index.php/Archivo:Secuencias.JPG> [citado el 10
Octubre de 2015]