

**ADMINISTRACIÓN, ANÁLISIS DISEÑO, DESARROLLO E IMPLEMENTACIÓN
DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE
INGENIERÍA SISTEMAS, INGENIERÍA MECÁNICA E INGENIERÍA QUÍMICA.**

**CARLOS ANDRÉS NEIRA VARGAS
HABIB SAKER HERNÁNDEZ**

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

**ADMINISTRACIÓN, ANÁLISIS DISEÑO, DESARROLLO E IMPLEMENTACIÓN
DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE
INGENIERÍA SISTEMAS, INGENIERÍA MECÁNICA E INGENIERÍA QUÍMICA.**

**CARLOS ANDRÉS NEIRA VARGAS
HABIB SAKER HERNÁNDEZ**

Trabajo de grado para optar el título de Ingeniero de Sistemas

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

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

Agradecimientos

Quisiera expresar mi profundo agradecimiento a todas las personas que hicieron este triunfo posible, a mi familia por su apoyo, comprensión y confianza. A mis amigos y compañeros de Calumet por tantos momentos agradables. Este logro también es de ustedes.

Habib Saker Hernández

A mi familia en especial a mis padres Germán y Martha, a mis hermanos Mauro y Yuly, a Kate, Mao y Alejita con sus consejos, cariño y apoyo incondicional he podido obtener este logro en mi vida.

A mis compañeros de Calumet que me acompañaron en esta etapa convirtiéndose en una experiencia agradable y divertida. A Habib y Sarita que me brindaron su amistad en el momento que más la necesitaba.

A mis amigos que me acompañaron en las buenas y en las malas Rubiano, Toño, el papi, machero killer, William, Eusebio, were, varguitas, Julián, jeep, veneco, Jimmy, renzo, manuel, mauro, marcos y todos los vaguitos de halo.

A mi segunda familia Nx, Mike, Xamilox, Josefer, Hizuko, Laurita, Pao y Don Antonio patrocinador oficial del trono.

Muchas gracias por brindarme su amistad y apoyo a lo largo de mi carrera

Carlos Andrés Neira Vargas

CONTENIDO

INTRODUCCIÓN	21
1 PRESENTACIÓN DEL PROYECTO	23
1.1 ORIENTACIÓN SOBRE EL CONTENIDO DEL INFORME	23
1.2 ANTECEDENTES.....	24
1.3 DEFINICIÓN DEL PROBLEMA	157
1.4 OBJETIVOS.....	160
1.4.1 Objetivo General	160
1.4.2 Objetivos Específicos	1608
1.5 JUSTIFICACIÓN	32
1.6 ALCANCES Y LIMITACIONES.....	34
2 MARCO TEÓRICO	167
2.1 ARQUITECTURA CLIENTE/SERVIDOR	167
2.1.1 Características de la arquitectura Cliente/Servidor.....	168
2.1.2 Clasificación de las arquitecturas Cliente/Servidor.	169
2.1.3 Arquitectura Cliente/Servidor aplicada.....	171
2.1.4 Ventajas del esquema Cliente/Servidor	40
2.1.5 Desventajas del esquema Cliente/Servidor	40
2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS.....	41
2.2.1 Código del Lado del Cliente (Client Side Scripts).	41
2.2.2 Código del Lado del Servidor (Server Side Scripts).	42
2.2.3 Tecnología aplicada.	43
2.3 BASES DE DATOS.....	45
2.3.1 Modelos de Bases De Datos.....	45
2.3.2 Acceso a Base de Datos.	46
2.3.3 Conectores más utilizados.	46
2.3.4 Manejadores o Gestores de Bases de Datos.	48
2.3.5 MySQL.....	49
2.3.6 Ventajas de MySQL.	50
2.4 NETBEANS.....	50
2.5 SISTEMA DE CONTROL DE VERSIONES	51
2.5.1 Subversión.	51
2.6 PROGRAMACIÓN UTILIZADA	52
2.6.1 Clases.....	52
2.6.2 Objetos.	53

2.6.3	Atributos.	53
2.6.4	Métodos.	53
2.6.5	Herencia.	53
2.6.6	Beneficios de la Programación Orientada a Objetos.	53
2.6.7	Java y JDK (Java Development Kit).....	54
2.7	SERVIDORES WEB	54
2.7.1	Servidor Jakarta Tomcat	55
3	MARCO METODOLÓGICO	188
3.1	PROTOTIPO EVOLUTIVO	188
3.2	LENGUAJE DE MODELADO UNIFICADO	190
3.2.1	Diagramas de UML.	191
3.2.2	Diagramas de casos de uso.....	191
3.2.3	Diagramas de secuencias.	61
3.3	ESTÁNDARES DE PROGRAMACIÓN	63
3.3.1	Modelo de datos.....	63
3.3.2	Nombres de las tablas.....	63
3.3.3	Clases.....	64
3.3.4	Páginas JSP.....	64
3.3.5	Organización de Directorios.....	64
4	DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO	66
4.1	PROTOTIPO ESPERADO	66
4.1.1	Diagramas de Casos de Uso.	79
4.1.2	Documentación de Casos de Uso del Sistema.....	88
4.1.3	Diseño y Análisis.	230
4.1.4	Modelo de Procesos del Sistema.....	241
4.1.5	Implementación, Implantación y Pruebas Generales.	254
4.2	MANTENIMIENTO Y ADMINISTRACIÓN	255
4.2.1	Actividades de Mantenimiento.	255
4.2.2	Actividades de Soporte a Usuarios.	256
4.2.3	Actividades de Administración.	257
5	PRUEBAS DEL SISTEMA.....	260
5.1	PRUEBAS DE VERIFICACIÓN	260
5.1.1	Pruebas por componente.	260
5.2	PRUEBAS DE INTEGRACIÓN.....	268
5.3	PRUEBAS DE VALIDACIÓN.....	268

6	CONCLUSIONES.....	260
7	RECOMENDACIONES	270
	BIBLIOGRAFÍA	270
	ANEXOS	270

LISTA DE TABLAS

Tabla 1. Casos de uso: Asociar Quices al Aula de Clase.	220
Tabla 2. Casos de uso: Asignar y modificar nota en el Aula.	221
Tabla 3. Casos de uso: Crear y Gestionar Proyecto.	90
Tabla 4. Casos de Uso: Gestionar Etapas y Actividades.	91
Tabla 5. Caso de Uso: Buscar Proyecto.	92
Tabla 6. Casos de uso: Vista General Proyecto.	93
Tabla 7. Casos de uso: Gestionar Semana.	227
Tabla 8. Casos de uso: Registrar Producto.	95
Tabla 9. Casos de uso: Mantenimiento y Administración de los portales.	228
Tabla 10. Descripción de las Entidades.	238
Tabla 11. Pruebas Realizadas: Asociar Quices al Aula de Clase.	261
Tabla 12. Pruebas realizadas: Asignar y modificar nota en el Aula.	262
Tabla 13. Pruebas realizadas: Crear y Gestionar Proyecto.	263
Tabla 14. Pruebas realizadas: Gestionar Etapas y Actividades.	264
Tabla 15. Pruebas realizadas: Buscar Proyecto.	265
Tabla 16. Pruebas realizadas: Vista General Proyecto.	265
Tabla 17. Pruebas realizadas: Gestionar Semana.	267
Tabla 18. Pruebas realizadas: Registrar Producto.	268

LISTA DE FIGURAS

Figura 1. Modelo Cliente/Servidor	168
Figura 2. Arquitectura Cliente/Servidor de Dos Capas	170
Figura 3. Arquitectura Cliente/Servidor de Tres Capas	171
Figura 4. Modelo de acceso a JSP	44
Figura 5. Función del DBMS.	49
Figura 6. Prototipo Evolutivo.	56
Figura 7. Diagramas de Casos de Uso	60
Figura 8. Diagrama de Secuencias.....	62
Figura 9. Diagrama de Casos de Uso: Asociar Quices al Aula.	79
Figura 10. Diagrama de Casos de Uso: Asignar y modificar Notas en el Aula.	212
Figura 11. Diagrama de Casos de Uso: Crear y Gestionar Proyectos.	81
Figura 12. Diagrama de Casos de Uso: Gestionar Etapas y actividades.....	82
Figura 13. Diagrama de Casos de Uso: Buscar Proyecto.	83
Figura 14. Diagrama de Casos de Uso: Vista General del Proyecto.....	84
Figura 15. Diagrama de Casos de Uso: Gestionar Semana.	218
Figura 16. Diagrama de Casos de Uso: Registrar Producto.	218
Figura 17. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.	218
Figura 18. Diagrama E/R: Asociar Quices al Aula.....	230
Figura 19. Diagrama E/R: Asignar y modificar Notas en el Aula	231
Figura 20. Diagrama E/R: Crear y Gestionar Proyectos.	100
Figura 21. Diagrama E/R: Gestionar Etapas y actividades..	101
Figura 22. Diagrama E/R: Buscar Proyecto.....	102
Figura 23. Diagrama E/R: Vista General del Proyecto.	103
Figura 24. Diagrama E/R: Gestionar Semana.	104
Figura 25. Diagrama E/R: Registrar Producto.	237
Figura 26. Diagrama de secuencia: Asociar Quices al Aula.....	241
Figura 27. Diagrama de secuencia: Asignar y modificar Notas en el Aula.....	242
Figura 28. Diagrama de secuencia: Crear y Gestionar Proyectos	244
Figura 29. Diagrama de secuencia: Gestionar Etapas y actividades	246
Figura 30. Diagrama de secuencia: Buscar Proyecto.	248
Figura 31. Diagrama de secuencia: Vista General del Proyecto.....	249
Figura 32. Diagrama de secuencia: Gestionar Semana.....	252
Figura 33. Diagrama de secuencia: Registrar Producto.....	253

LISTA DE ANEXOS

ANEXO A MANUAL DE USUARIO.....	270
--------------------------------	-----

GLOSARIO

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

RESUMEN

TÍTULO: ADMINISTRACIÓN, ANÁLISIS DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE INGENIERÍA SISTEMAS, INGENIERÍA MECÁNICA E INGENIERÍA QUÍMICA.

AUTORESⁱ: Carlos Andrés Neira Vargas
Habib Saker Hernández

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

DESCRIPCIÓN

El portal web de la EISI se ha consolidado como la plataforma de comunicación y difusión más relevante para el ejercicio de las actividades de los miembros de la escuela, pero esto es solo posible gracias a cada una de las contribuciones de la comunidad, elemento imprescindible para el mejoramiento y el fortalecimiento de los servicios que ofrece el portal. En este sentido, el grupo de desarrollo de software Calumet ha asumido la tarea de implementar mejoras a los servicios existentes y crear nuevos servicios para el beneficio de la comunidad.

La necesidad de hacer seguimiento y gestión a los proyectos de los grupos de la comunidad universitaria ha propiciado la creación del servicio Gestor de proyectos, el cual ofrece módulos para la búsqueda, creación, y edición de proyectos, y para la gestión de las etapas y actividades. Además, se creó un módulo para registrar los productos intelectuales de los proyectos y de los profesores de la comunidad UIS.

En el área de actualización de estados y categorías de los usuarios de la escuela, fue necesaria la creación de una tabla nueva para relacionar por medio del documento de identidad los diferentes programas académicos que tenga el estudiante.

En el Aula Virtual se implementó el servicio de individualizar las notas de los estudiantes, para llevar mejor control del desempeño individual, además se asociaron los quices al Aula de clase para registrarlos en la individualización de las notas.

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

ABSTRACT

TITLE: MANAGEMENT, ANALYSIS, DESIGN, DEVELOPMENT AND IMPLEMENTATION OF THE ACADEMIC MANAGEMENT SYSTEM FOR THE SYSTEMS ENGINEERING, MECHANICAL ENGINEERING, AND CHEMICAL ENGINEERING PORTALS.

AUTHORSⁱ: Carlos Andrés Neira Vargas
Habib Saker Hernández

KEYWORDS: Website, Web Portal, Module, Service, EIMT (Metallurgical and Materials Science Engineering School), EEIE (Industrial and Business Studies School), EIMTWeb, EIEEWeb.

DESCRIPTION

The web portal EISI has consolidated itself as the most relevant platform for communicating and distributing information, at the service of the members of the school. But, this is only possible thanks to the contributions of the community, a necessary element for the improvement and strengthening of the services offered by the portal. In this sense, the software development group Calumet has undertaken the task of implementing improvements to existing services and creating new services for the benefit of the community.

The need to track and manage the projects of the university community groups has led to the creation of the Project Management Service, which offers modules for searching, creating, and editing the project, and for managing the stages and activities. It was also created a module to register the intellectual products of the project.

In the area of renovation of states and categories of users of the portal, it was necessary to create a new table to relate through identification, academic programs to an enrolled student.

A virtual classroom service was deployed to individualize student grades, to better assess individual performance, and quizzes were associated with each course to register them during the individualization of grades.

ⁱ Physical Mechanical Engineering Faculty, School of Systems Engineering and Computing science. Director: M.Sc. Luis Ignacio González Ramírez

INTRODUCCIÓN

Los portales EISiWeB, EIMWeb y EIQWeB pertenecientes respectivamente a la Escuela de Ingeniería de Sistemas e Informática, la Escuela de Ingeniería Mecánica y la Escuela de Ingeniería Química, han logrado ser el principal canal de comunicación e integración entre los miembros de las escuelas; y es debido a la gran aceptación y utilidad que demuestran donde surge el objetivo fundamental de consolidar y mejorar los servicios que ofrecen y además la creación de módulos para suplir las necesidades de los miembros de las escuelas.

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

A pesar de la gran cantidad de servicios con los que cuenta el portal, surgió la necesidad de complementar el entorno Asignar Notas y Quices dentro del servicio de Aula Virtual; orientado específicamente a individualizar las notas de los estudiantes para llevar un mejor control del desempeño de los estudiantes por parte de los profesores, así también se asocio los quices al Aula De Clase para poder implementarlos al modulo de Notas, brindando un mejor desempeño, control y visualización en las Notas del estudiante.

Para individualizar el modulo de notas se crearon 2 tablas una para Talleres y Exámenes y la otra Para Actividades, Exposiciones y Quices. Inicialmente el profesor o el sistema (en el caso de talleres, exámenes y quices) calculan o definen la nota. Después existirá el botón asignar nota, que automáticamente

copia las notas y las individualiza en la nueva tabla. Este botón se llamará "Asignar Nota" cuando se oprima por primera vez. Después se convertirá en modificar nota, cuando las notas ya se encuentren en la nueva tabla.

Se creó una tabla en diamante "TR_UsuariosPrograma" que tiene IdUsr, código de programa, código de estudiante en el programa, y relaciona el código de los programas en TR_UsuariosPrograma. El asunto es que puede haber estudiantes en más de un programa y hay que relacionarlos, estos campos seguirán estando en TP_Usuarios mientras se hacen las reingenierías necesarias, además se aplicó reingeniería a Actualización Semestral, Actualización Periódica, Actualización de horarios, y Actualización profesores. Así en el módulo de Horarios de los profesores se puede visualizar correctamente todas las materias que dictan los profesores de cada escuela.

Se creó un servicio nuevo que soporta el manejo de Proyectos con responsables, encargados, fechas de asignación y entrega. Incluye la creación de un servicio para ingresar la Producción intelectual de la comunidad en especial los Profesores.

Los módulos de este servicio sirven para crear y editar los con tenidos del Proyecto, etapas y actividades, les permite ver el desempeño de los responsables de las actividades en la semana, facilitando el cumplimiento de los periodos de tiempo propuesto para cada actividad. Mejora la planeación de un Proyecto ofreciendo una vista en el tiempo, donde se muestra las actividades, su duración y su estado, con una breve descripción de las mismas.

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

1 PRESENTACIÓN DEL PROYECTO

1.1 ORIENTACIÓN SOBRE EL CONTENIDO DEL INFORME

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

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

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

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

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

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

CAPITULO 6. Conclusiones del trabajo realizado.

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

1.2 ANTECEDENTES

A medida que la comunidad de la Universidad Industrial de Santander crece y particularmente la comunidad de la Escuela de Ingeniería de Sistemas e Informática, se hace más valioso el manejo de información y comunicación entre los miembros de la comunidad. Ante esta necesidad en el año 2004 nació la idea de crear un portal web que facilitara la comunicación y la realización de diferentes actividades dentro de la escuela, labor que se llevó a cabo por los integrantes del grupo de desarrollo software Calumet y que se está mejorando con el paso de generaciones a través del desarrollo de sus respectivos proyectos de grado, para obtener lo que hoy se conoce como el portal web de la Escuela de Ingeniería de Sistemas e Informática EISIWeb, también ha sido implantado en la Escuela de Ingeniería Mecánica EIMWeB y la Escuela de Ingeniería Química EIQWeb entre otras escuelas.

En vista de la acogida del portal EISIWeb, surge la iniciativa de extender los servicios a usuarios nuevos y la creación de otros servicios para satisfacer las necesidades de la comunidad. Hoy solo la Escuela de Ingeniería de Sistemas e Informática cuenta con un portal web dinámico que presta servicio a todos los miembros de su comunidad y habitualmente enriquecido con nuevos aportes hechos por parte de los integrantes del grupo de desarrollo software Calumet.

1.3 DEFINICIÓN DEL PROBLEMA

El portal EISIWeb es un sitio en crecimiento continuo, sujeto a cambios e inconvenientes diarios, razón por la cual es necesario delegar personas para ejecutar labores de administración, mantenimiento y soporte a usuarios del portal.

Con el paso de cada generación de estudiantes por el grupo de desarrollo Calumet, se requieren estudiantes nuevos que hereden labores y responsabilidades; con este nuevo grupo se hace un empalme con las labores que se llevan a cabo para la continuidad en la construcción y mantenimiento del portal.

El soporte a la labor docente, desde el portal web de las escuelas, se hace desde el concepto de Aula Virtual de Aprendizaje (A.V.A). Este se entiende como el espacio creado para complementar el proceso de aprendizaje de manera no presencial como una forma de garantizar el cumplimiento de las horas de trabajo independiente (T.I.) por parte del estudiante.

Actualmente los estudiantes cuentan con la interface dentro del Aula Virtual en la cual pueden, principalmente, contestar las actividades propuestas por el docente dentro de un guión de clases para cada una de sus diferentes asignaturas. El docente puede crear o modificar un guión de clases asociándole planes de actividades (foros, trabajos, laboratorios y exposiciones) y/o planes de talleres y exámenes, dichos planes previamente estructurados.

El Aula Virtual soporta al docente el proceso de calificación de las actividades de la materia. Esta calificación se hace sobre actividades realizadas en subgrupos de clase. No obstante, al momento de publicar la nota y hacer los cálculos de la nota definitiva, esta se debe individualizar para poder tener en cuenta los casos particulares y las posibles reclamaciones por parte de los estudiantes.

Debe crearse una nueva versión que permita la asignación e individualización de la nota, separando la calificación de la actividad de la nota finalmente asignada a cada estudiante. A su vez, la nota calculada por el sistema de talleres y exámenes podrá ser modificada por el docente de acuerdo a sus apreciaciones.

Gracias a los servicios que ofrece el aula virtual los docentes tenían herramientas suficientes para crear las actividades de las asignaturas y posteriormente evaluar a sus estudiantes, pero los quices no estaban asociados al Aula de Clase como tal, se podían realizar quices pero no se asociaban a los estudiante por lo tanto no se podía individualizar las notas, es necesario dar la opción de agregar quices en la construcción del Aula de Clase y así tenerlos en cuenta al momento de individualizar las notas y almacenarlas en las nuevas tablas. Estos inconvenientes motivaron una reconstrucción con el objetivo de entregarle a los docentes un módulo donde se asignara e individualizara la nota más fácilmente.

La Base de Datos Diamante en su tabla de usuarios tenía la capacidad de guardar información hasta de 2 programas, uno de pregrado y otro de posgrado, para cada usuario. No obstante, se podía dar el caso del usuario que se relacionara con más de 2 programas de pregrado y posgrado, en estos casos no habría como hacer el soporte a estos usuarios en el portal de la Escuela.

Se debe realizar los ajustes necesarios en la bases de datos para que se tenga en cuenta la posibilidad de tener estudiantes en más de un programa académico al mismo tiempo. Para lo anterior es necesario crear una tabla para guardar el código de estudiante, el código del programa, su promedio ponderado actual, los puntos y el estado con respecto al programa. Además se debe revisar la actualización periódica y semestral para que el estado de los usuarios que tengan como categoría profesor proyectos no se cambie. También es necesario revisar la actualización de horarios, para que los horarios de los profesores de la escuela muestren correctamente las materias, así no se estén dictando a estudiantes

pertenecientes a la escuela, a su vez cargar los horarios de los profesores que no son de la escuela, pero dictan materias a estudiantes pertenecientes a ella.

Motivados por brindarle a los profesores, grupos de investigación y la comunidad en general un mejor servicio y un sistema robusto capaz de suplir todas sus necesidades se creó un servicio para la gestión, control y ejecución de proyectos de diferente índole. Estos proyectos tienen etapas, y estas a su vez actividades con fechas, responsables y controladores asignados. Estos proyectos generan productos de diferente tipo, que deben ser publicitados en algunos casos a todos los interesados, y en otros solo a los usuarios del Portal, se creó un módulo para el registro de los productos, donde se tiene en cuenta tesis de grado, libros, artículos y participaciones en eventos como congresos, seminarios etc.

Soportar dentro del portal el manejo de proyectos con responsables, encargados, fechas de asignación y entrega. Además permite relacionar la producción intelectual con el grupo, los directores y los proyectos. Estos proyectos se relacionan con tesis de maestría y trabajos de grado. En general la comunidad tiene la posibilidad de gestionar sus proyectos y puede asociarlos con otros servicios ya establecidos como los son grupos y profesores, es la primera fase de una herramienta que construye comunidad en la escuela.

1.4 OBJETIVOS

1.4.1 Objetivo General

Realizar las funciones de administración, mantenimiento, Análisis diseño y desarrollo de nuevos servicios para el portal Web existente en las escuelas de Ingeniería de Sistemas, Ingeniería Mecánica e Ingeniería Química para hacer óptimos, eficientes y sostenibles los trámites dentro de las escuelas y el acceso a la información de la misma.

1.4.2 Objetivos Específicos

1.4.2.1 Efectuar labores de administración del portal web de la Escuela de Ingeniería de Sistemas, de la Escuela de Ingeniería Mecánica y de la Escuela de Ingeniería de Química teniendo en cuenta principalmente:

- Generar Backups (copias de respaldo) diariamente de la Base de Datos.
- Salvar la información de los Sitios Web una vez por semana por medio de copias de respaldo y mantener un histórico de los mismos en caso de alguna falla.
- Hacer seguimiento del uso que hacen los usuarios a los servicios de Foros, Eventos Propuestos, Cartelera, Archivos y Mi perfil dentro de los portales EISIWEB, EIMWEB y EIQWEB para detectar usos indebidos o incorrectos por parte de éstos.
- Actualizar periódicamente las Bases de Datos con el objeto de mantener al día la información referente a matrículas, horarios, estados y categorías de los usuarios.

- Atender las consultas y sugerencias que los usuarios hagan para proponerlas como mejoramiento y ofrecimiento de nuevos servicios en la próxima versión del portal web.
- Realizar una revisión constante de los archivos que se suben al sitio, eliminando los que no son necesarios para evitar saturación del portal.
- Realizar periódicamente una copia de seguridad a la copia maestra de EISIWEB, EIMWEB y EIQWEB almacenada en el servidor de repositorios.

1.4.2.2 Llevar a cabo labores de mantenimiento en los portales EISIWEB, EIMWEB y EIQWEB en los que podemos encontrar:

- Implementar los nuevos servicios o mejoras realizadas por otros desarrolladores del grupo CALUMET dentro de los portales EISIWEB, EIMWEB y EIQWEB, ajustando el nuevo código, modificando la Base de Datos y realizando las pruebas necesarias.
- Hacer el seguimiento del funcionamiento de los portales para corregir posibles defectos generados por errores en el código fuente que se puedan presentar.
- Revisar y depurar la estructura de directorios y archivos del Portal Web.

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

- Se debe crear una tabla en diamante que tenga IdUsr, código de programa, código de estudiante en el programa, y que se relacione con el código de los programas en TR_UsuariosPrograma. El asunto es que puede haber estudiantes en más de un programa. Estos campos (código de estudiante, código de programa, ...) se mantendrían en TP-Usuarios mientras se hacen reingenierías a los servicios. Revisar la actualización periódica y semestral para que el estado de los usuarios que tengan como categoría profesor proyectos no se cambie.
- Soportar dentro del portal de los grupos el manejo de proyectos con responsables, encargados, fechas de asignación y entrega. Incluye la creación de un servicio para ingresar las tesis de maestría por la secretaria de maestría. Permitirá usarse en el horario de profes. Primera fase.
- En el módulo de notas del aula. Las notas deben individualizarse en el Aula, para esto es necesario crear 3 tablas: para talleres y exámenes, para las otras actividades y para los quices. Inicialmente el profesor o el sistema (en el caso de talleres, exámenes y quices) calcula o define la nota. Después existirá el botón asignar nota, que automáticamente copia las notas y las individualiza en una nueva tabla. Este botón se llamará "Asignar Nota" cuando se oprima por primera vez. Después se convertirá en modificar nota, cuando ellas ya se encuentren en la nueva tabla.

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

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

1.4.2.5 Capacitar a los estudiantes que recibirán las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro de los portales EISIWEB, EIMWEB y EIQWEB en cuanto a:

- Implantación del sitio local para la creación de nuevos servicios y realización de pruebas.
- Realizar inducción en cuanto al manejo y utilización de los JSP, Beans y Base de Datos.
- Llevar a cabo la familiarización con el entorno del portal web.

- Administrar el Repositorio de CALUMET a fin de sacar Backups.

1.5 JUSTIFICACIÓN

En la actualidad la escuela de Ingeniería de Sistemas e Informática, escuela de Ingeniería Mecánica y la escuela de Ingeniería Química cuentan con un sistema de información orientado a la Web que se encarga de la administración y control de las diferentes actividades que se realizan dentro de las Escuelas, así como el control de usuarios y servicios que se les proporciona.

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

Por otra parte, el soporte a la labor docente desde el Portal se realiza a partir del concepto de Aula Virtual. Este se entiende como el espacio creado para complementar el proceso de aprendizaje de manera no presencial como una forma que garantiza el cumplimiento de las horas de trabajo independiente (T.I.) por parte del estudiante. Nuestra Aula Virtual comprende: contenidos, actividades, visualización de calificaciones (notas) a profesores y estudiantes en el Aula Virtual. Es necesario que tanto profesores como estudiantes se sientan cómodos utilizando el Aula Virtual por lo que se hace indispensable mejorar la interfaz y operatividad con la que interactúan. Los profesores al calificar las diferentes actividades de una asignatura deben poder asignar la nota a cada estudiante, emitiendo un concepto de apreciación de la nota individualizada,

además deben poder modificarlas en que caso tal que el profesor lo vea necesario.

El portal web debe soportar los usuarios que están en más de dos programas académicos de pregrado y posgrado, se debe crear una nueva tabla en diamante que relacione los códigos de los programas de tr_usuariosprograma, además en las actualizaciones semestral y periódica verificar los estados y los horarios de los profesores de la escuela.

El portal brinda un espacio a los profesores, grupos de investigación y comunidad en general donde pueden gestionar sus proyectos, con responsables, encargados, fechas de asignación y entrega. Este espacio lo comprenden etapas, actividades, visualización del tiempo del proyecto, las actividades tienen un responsable el cual puede ser supervisado por el director o el auxiliar del proyecto, ellos son los encargados de iniciar y finalizar las actividades según el rendimiento que observen, se crearon varios tipos de visualización de las actividades con el fin de entender fácilmente el recorrido en el tiempo. Con el gestor de Proyectos es posible asociar Los profesores, estudiantes y Grupos de investigación, con los resultados o el producto, integrando estos servicios y brindando un mejor soporte a la docencia y a la comunidad.

1.6 ALCANCES Y LIMITACIONES

La administración y mantenimiento en los portales EISIWEB, EIMWEB y EIQWEB, permite y garantiza su correcto funcionamiento, de esta forma responder a la demanda diaria de los usuarios, quienes contarán con un soporte para realizar diferentes actividades y resolver algunos inconvenientes que se presentan a menudo en el manejo y utilización del portal. Por otra parte, el desarrollo de nuevos servicios enfocados al Gestor de Proyectos fortalece las funcionalidades de cada portal, con lo que se constituyen, para los usuarios, en una herramienta que facilita el acceso a la información y el manejo de esta. El Gestor de Proyectos se promueve como un espacio interactivo y dinámico que permite la planeación y desarrollo de Proyectos, darle seguimiento con la asignación de fechas iniciales y finales, además integrando en el portal los Profesores, Grupos de Investigación y estudiantes con el fin de construir comunidad.

2 MARCO TEÓRICO

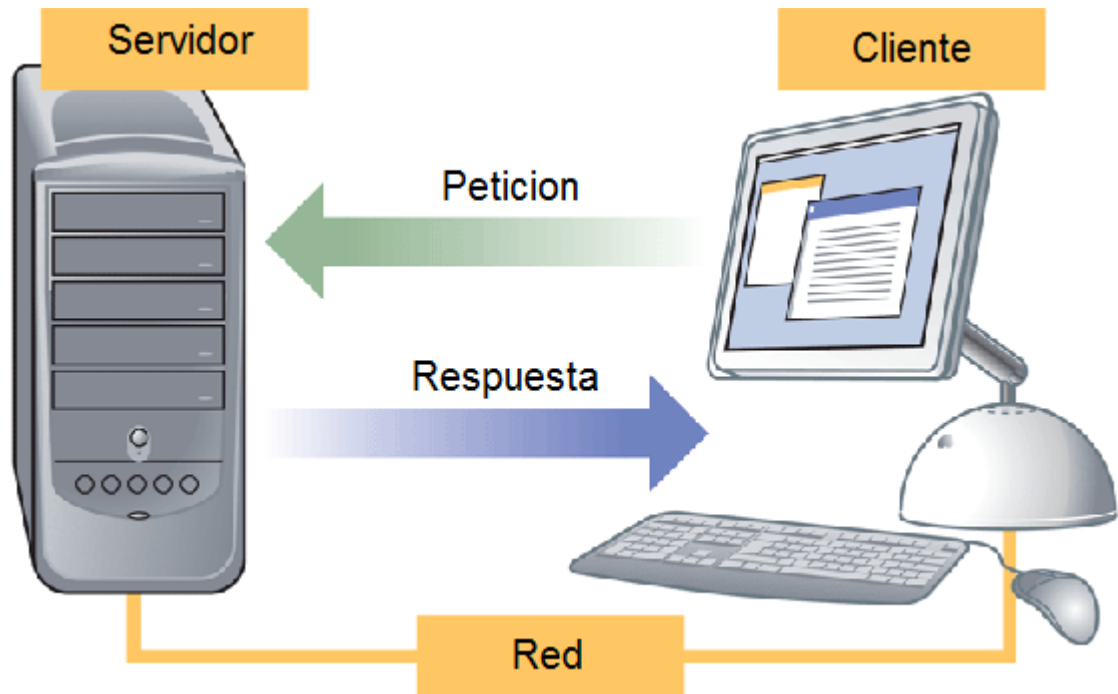
2.1 ARQUITECTURA CLIENTE/SERVIDOR

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

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

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

Figura 1. Modelo Cliente/Servidor



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

2.1.1 Características de la arquitectura Cliente/Servidor.

Las características básicas de una arquitectura Cliente / Servidor son:

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

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

2.1.2 Clasificación de las arquitecturas Cliente/Servidor.

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

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

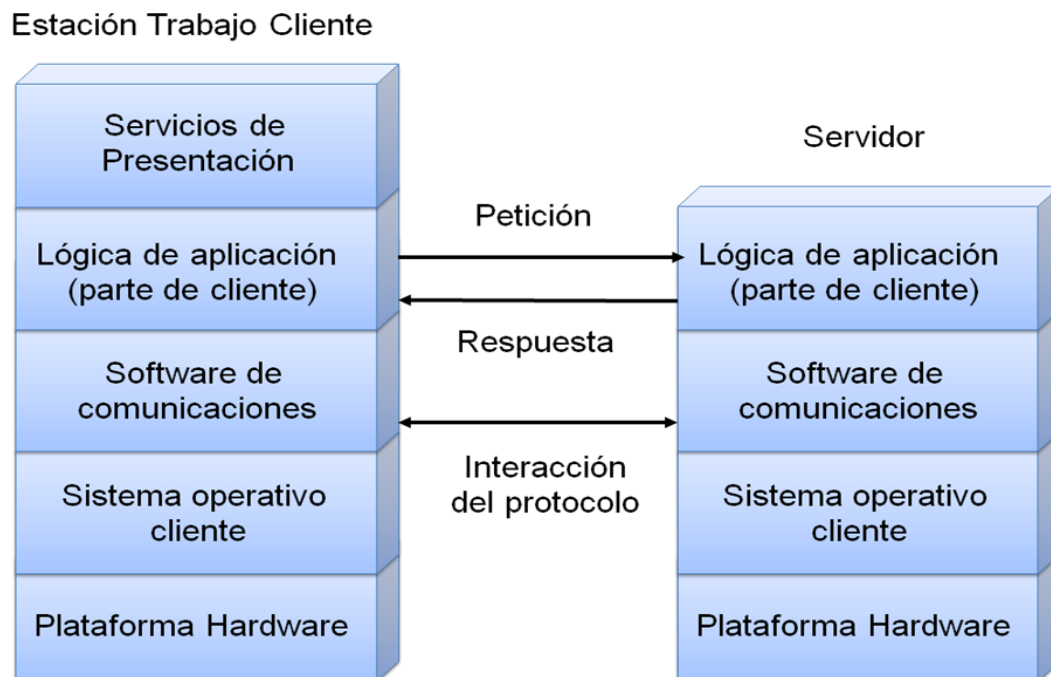
A continuación se muestra la clasificación de los sistemas Cliente / Servidor:

2.1.2.1 Arquitectura Cliente/Servidor de dos capas.

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

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

Figura 2. Arquitectura Cliente/Servidor de Dos Capas



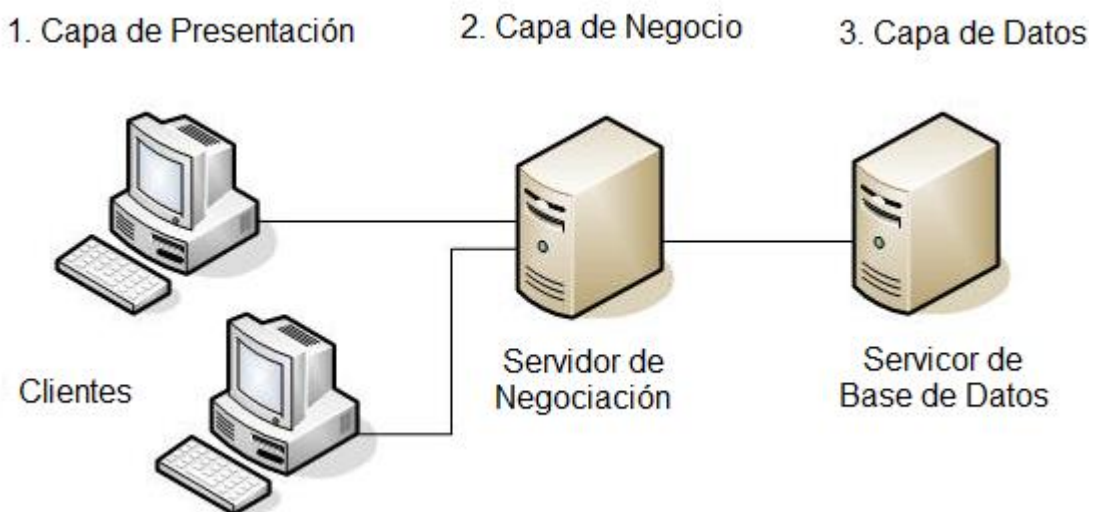
Fuente: Autores.

2.1.2.2 Arquitectura Cliente/Servidor de tres capas.

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

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

Figura 3. Arquitectura Cliente/Servidor de Tres Capas



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

2.1.3 Arquitectura Cliente/Servidor aplicada.

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

- Capa de Cliente: Interfaz con el usuario, se usa un navegador web.

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

2.1.4 Ventajas del esquema Cliente/Servidor

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

2.1.5 Desventajas del esquema Cliente/Servidor

- El mantenimiento de los sistemas es complejo pues implica la interacción de diferentes partes hardware y software de diferentes proveedores, lo cual dificulta el diagnostico de fallas.

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

2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS

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

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

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

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

- *JavaScript*: Lenguaje de programación interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Permite, crear ventanas, mostrar texto en movimiento y verificar las entradas a un formulario.

- *Controles Activos*: tecnología Microsoft que permite la creación de aplicaciones Windows, como pueden ser Visual Basic Script o Visual C. Es la respuesta de Microsoft a los Applets de Java.
- *Java Applets*: Programas escritos en lenguaje de programación Java, se incrustan en HTML y se ejecutan en el navegador gracias a la Máquina Virtual de Java (JVM) que lleva éste incorporado.

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

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

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

- JSP (Java Server Pages): tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. Permiten la utilización de código Java mediante scripts.

2.2.3 Tecnología aplicada.

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

Las paginas JSP y servlets se ejecutan en la Máquina Virtual de Java, lo cual permite que se puedan usar en cualquier tipo de computador, siempre y cuando este instalada la Máquina Virtual de Java. Cada JSP se ejecuta en su propio contexto (llamado también hilo o hebra); pero no se comienza a ejecutar cada vez que recibe una petición, sino que persiste de una petición a la siguiente, de forma que no se pierde tiempo en invocarlo. Su persistencia permite hacer cosas de forma más eficiente como la conexión a bases de datos y manejo de sesiones.

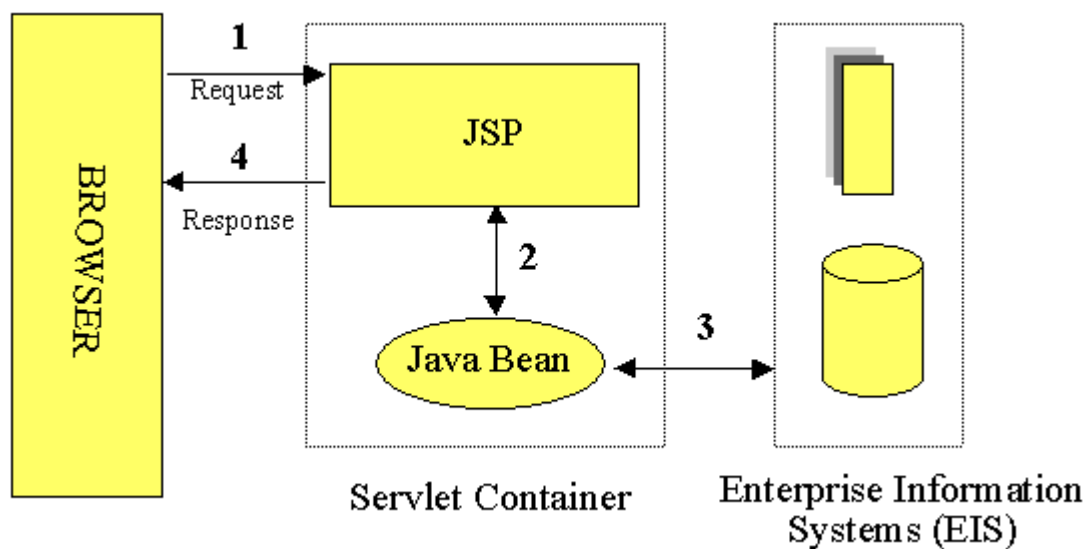
Una página JSP se compila a una aplicación Java la primera vez que se invoca, y de esta aplicación Java se crea una clase que empieza a ejecutarse en el servidor como un servlet. Un JSP es una página web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa que recibe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

2.2.3.1 Modelo de acceso a JSP.

1. Un usuario en su navegador web cliente hace una petición que es enviada a un archivo JSP. Este archivo accede a componentes del servidor que generan contenido dinámico y lo presentan en el navegador.

2. Después de recibir la petición del cliente, el archivo JSP pide información de un JavaBean si es necesario.
3. El JavaBean a su vez puede pedir información de otro JavaBean o de una base de datos.
4. Una vez el JavaBean genera el contenido, el archivo JSP puede consultar y presentar el contenido del JavaBean al navegador.

Figura 4. Modelo de acceso a JSP



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

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

2.3 BASES DE DATOS

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso, con una redundancia controlada y una estructura que refleja las interrelaciones y restricciones existentes en el mundo real. En la base de datos se almacena información considerada necesaria para una determinada organización o negocio.

2.3.1 Modelos de Bases De Datos.

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

2.3.1.1 Base de Datos Jerárquica.

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

2.3.1.2 Base de Datos de Red.

En este modelo se permite que un mismo nodo tenga varios padres. Ofrece una solución eficiente al problema de redundancia de datos; sin embargo, la dificultad

para administrar los datos en una base de datos de red ha conllevado a que sea un modelo usado más por programadores que por usuarios finales.

2.3.1.3 Base de Datos Relacional.

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

2.3.2 Acceso a Base de Datos.

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

2.3.3 Conectores más utilizados.

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

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

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

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

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

Las operaciones primordiales realizadas durante la ejecución del JDBC son:

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

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

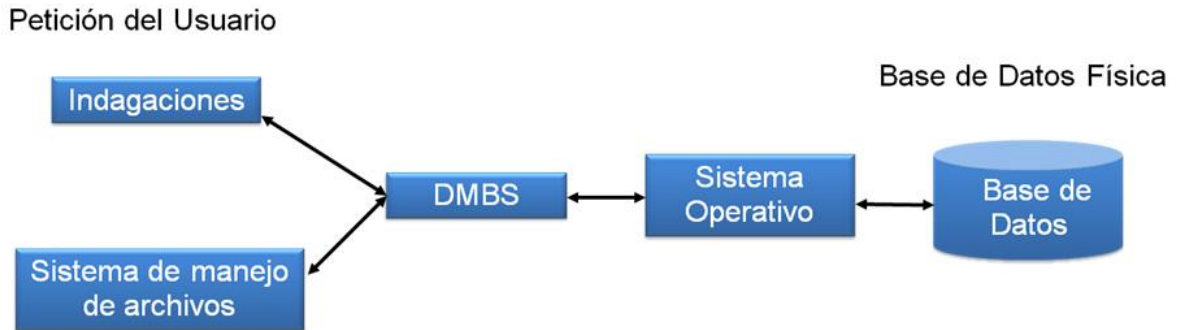
2.3.4 Manejadores o Gestores de Bases de Datos.

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

Las funciones principales de un DBMS son:

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

Figura 5. Función del DBMS.



Fuente: Autores.

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

2.3.5 MySQL.

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

2.3.6 Ventajas de MySQL.

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

2.4 NETBEANS

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

- El Netbeans es un entorno de desarrollo integrado de código abierto escrito completamente en Java usando la plataforma Netbeans, soporta desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles).
- La versión actual es NetBeans IDE 7.1.2. Desde NetBeans IDE 6.5 se extienden las características existentes del Java EE (incluyendo Soporte a Persistencia, EEJB 3 y JAX-WS). Adicionalmente, el Netbeans Enterprise Pack soporta el desarrollo de Aplicaciones empresariales java EE 5, incluyendo herramientas de desarrollo visuales de SOA, herramientas de

esquemas XML, orientación a web servicios (for BPEL), y modelado UML. El NetBeans C/C++ Pack soporta proyectos de C/C++.

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

2.5 SISTEMA DE CONTROL DE VERSIONES

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

2.5.1 Subversión.

Sistema de control de versiones 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.

2.6 PROGRAMACIÓN UTILIZADA

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

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

2.6.1 Clases.

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

2.6.2 Objetos.

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

2.6.3 Atributos.

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

2.6.4 Métodos.

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

2.6.5 Herencia.

Se fundamenta en usar una clase ya creada para tomar sus características en clases más especializadas o derivadas de ésta para reutilizar el código que sea común con la clase base, y solamente definir nuevos métodos o redefinir algunos de los existentes para ajustarse al comportamiento particular de esta subclase.

2.6.6 Beneficios de la Programación Orientada a Objetos.

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

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

2.6.7 Java y JDK (Java Development Kit)

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

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

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

2.7 SERVIDORES WEB

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

2.7.1 Servidor Jakarta Tomcat

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

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

3 MARCO METODOLÓGICO

3.1 PROTOTIPO EVOLUTIVO

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

Figura 6. Prototipo Evolutivo.



Fuente: Autores.

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

- Los portales EISIWEB, EIMWEB y EIQWEB presentan constantemente necesidades por parte de los usuarios, por tanto los portales requieren nuevos servicios o mejorar existentes en el menor tiempo posible, debido a

que el sistema no es un producto final y por lo tanto está en constante reconstrucción.

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

Procedimiento a seguir para la metodología planteada:

- Se toma nota de los requisitos para la construcción de los prototipos.

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

3.2 LENGUAJE DE MODELADO UNIFICADO

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

UML no es un método de desarrollo porque no indica los pasos a seguir para llegar al código, es decir, no especifica como pasar del análisis al diseño y de este al código. Al no ser un método de desarrollo resulta ser independiente del ciclo de desarrollo que se siga, puede encajar en un ciclo en cascada, evolutivo, espiral o en métodos ágiles de desarrollo.

3.2.1 Diagramas de UML.

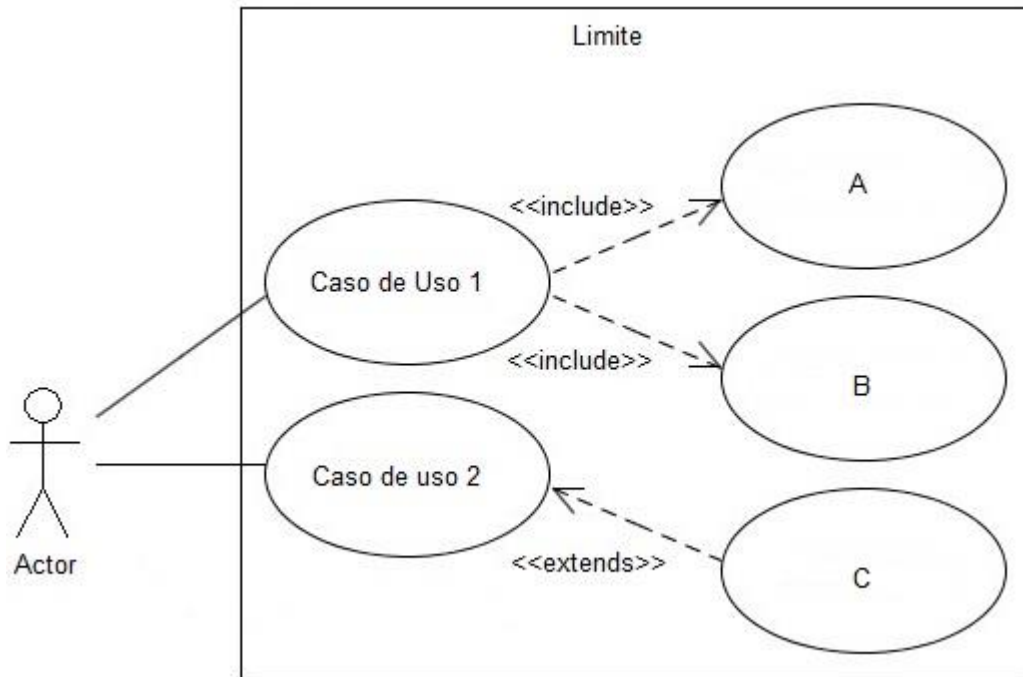
Los diagramas de UML utilizados en el desarrollo de este proyecto fueron: diagramas de casos de uso y diagramas de secuencias. Las principales razones por las cuales se prefirió UML como el lenguaje de modelado son:

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

3.2.2 Diagramas de casos de uso.

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

Figura 7. Diagramas de Casos de Uso



Fuente: Autores.

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

En este proyecto se destacan tres clases de actores:

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

- **Súper Administrador:** Es el tipo de usuario que puede administrar, controlar y modificar el sitio EISIWEB, EIMWEB y EIQWEB, sus parámetros y sus usuarios.
- **Usuario EISI, EIM, EIQ:** Es el tipo de usuario común del sitio EISIWEB, EIMWEB y EIQWEB, y a quien van dirigidos los servicios. Este usuario solo tiene el control sobre sus privilegios.

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

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

3.2.3 Diagramas de secuencias.

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

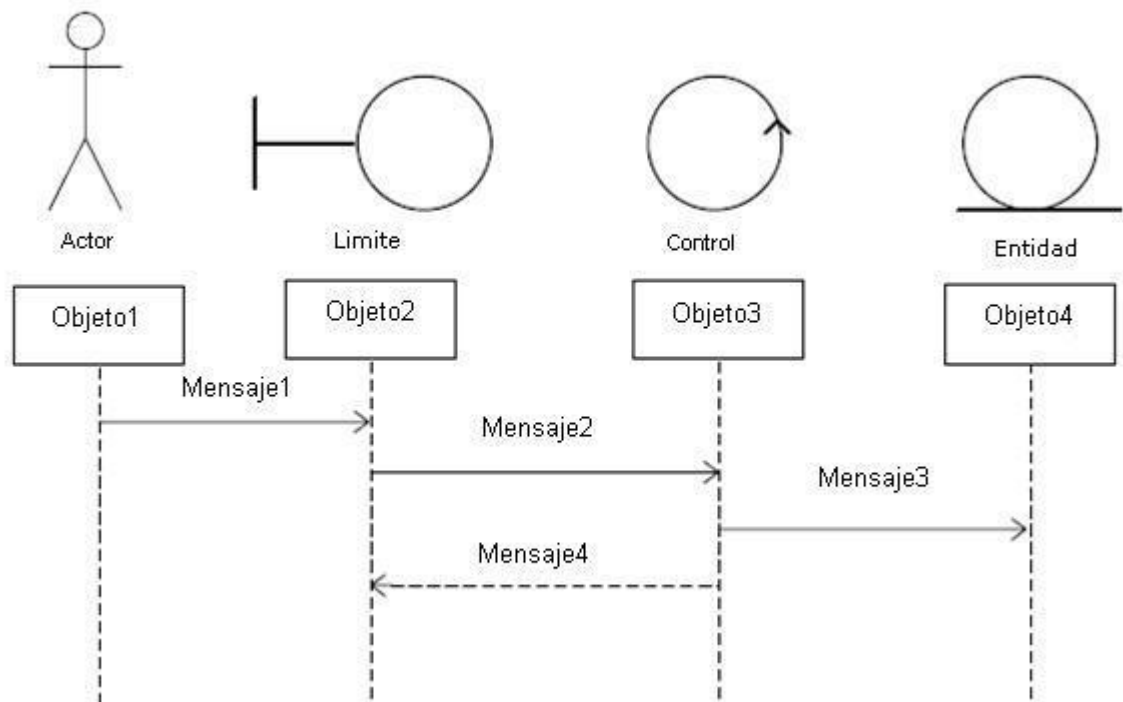
Objetos: Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de cada objeto será una línea discontinua conocida como la *línea de vida* de un objeto. Con la línea de vida se encuentra un pequeño rectángulo conocido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

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

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

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

Figura 8. Diagrama de Secuencias



Fuente: Autores.

3.3 ESTÁNDARES DE PROGRAMACIÓN

3.3.1 Modelo de datos.

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

3.3.2 Nombres de las tablas.

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

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

- *Tabla básica:* Aquella cuyos registros son necesarios para el correcto funcionamiento de la base de datos. Estas tablas no experimentan muchos cambios en los datos. El prefijo a anteponer a los nombres de estas tablas es “TB_”, es decir la que almacena las Materias de Grado es llamada “TB_MateriasGrado”.
- *Tabla de Relación:* Surge de la relación muchos a muchos de una o dos tablas cualquiera. Los nombres de las tablas de relación deberán ser descriptivos para cada relación. El prefijo a anteponer a los nombres de

estas tablas es “TR_”, es decir la tabla “Directores” es conocida como “TR_Directores”.

- *Tabla Principal:* Aquella cuya población de registros tiende a crecer en gran cantidad y que además no es posible clasificar como tabla básica o de relación. Un ejemplo de tabla principal es la tabla que almacena los usuarios del portal EISIWeb. El prefijo a anteponer a los nombres de estas tablas es “TP_”, es decir la tabla “Usuarios”, es conocida como “TP_Usuarios”.

3.3.3 Clases.

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

3.3.4 Páginas JSP.

Los nombres de las paginas JSP que componen el portal EEIEWeb, y EIMTWeb 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:

VistaActividad.jsp, DirigirAsignatura.jsp

3.3.5 Organización de Directorios.

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

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

4 DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO

Como se mencionó inicialmente para la reingeniería de los servicios de Aula Virtual, Actualización Periódica, Actualización Semestral, y el desarrollo del Gestor de Proyectos; se siguió la metodología de prototipo evolutivo.

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

4.1 PROTOTIPO ESPERADO

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

Análisis de Requisitos.

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

- 1. La BD Diamante en su tabla de usuarios esta en capacidad de guarda la información hasta de 2 programas, uno de pregrado, y otros de posgrado, para cada usuario. No obstante, puede darse el caso del usuario que se**

relaciona con más de 2 programas de pregrado, o/y posgrado, y en este caso no habría como hacer este soporte.

Objetivo Inicial:

- Se debe crear una tabla en diamante que tenga IdUsr, código de programa, código de estudiante en el programa, y que se relacione con el código de los programas en tbprogramas. El asunto es que puede haber estudiantes en más de un programa. Estos campos (código de estudiante, código de programa, ...) se mantendrían en tp-usuarios mientras se hacen reingenierías a los servicios. Revisar la actualización periódica y semestral para que el estado de los usuarios que tengan como categoría profesor proyectos no se cambie.

Requisitos finales del objetivo:

- Sólo tienen acceso a este servicio los administradores del Portal.
- Cuando se llena la Tabla TP_Usuarios, alternamente se llena la nueva tabla TR_UsuariosPrograma.
- En TP_Usuarios queda registrado el ultimo Programa Académico actualizado para el usuario, en TB_EstudientePrograma quedan registrados todos los programas académicos del usuario de tal forma no se creen nuevos IdUsr.
- Se debe hacer la búsqueda por cedula en la vista de División, remplazando la búsqueda por código de estudiante.
- TP_Usuarios debe conservar los campos (código de estudiante, código de programa, ...), mientras se realizan las reingenierías correspondientes.
- En la actualización de profesores se deben obtener todos los profesores de la escuela y además los profesores de otras escuelas que dictan materias a estudiantes de la escuela.

- En la actualización de horarios se deben agregar los horarios de las materias del plan de estudios, los horarios de las materias que dictan los profesores a estudiantes de la escuela, y los horarios de las materias que dictan los profesores de la escuela a estudiantes de otras escuelas.
- En la actualización de horarios se debe agregar a la tabla TB_MateriasN todas las materias nuevas que se actualizaron y que no estaban antes en esta tabla de la base de datos Diamante.
- Al final debe llenar la nueva tabla, y actualizar toda la información correspondiente con la construcción de los horarios de la comunidad de la Escuela.

2. Tanto los profesores, como los grupos de investigación, y en general, toda la comunidad requiere de la gestión, control y ejecución de proyectos de diferente índole. Estos proyectos deben contener fases, y estas a su vez tareas con fechas, responsables y controladores asignados. Estos proyectos generan productos de diferente tipo, que deben ser publicitados en algunos casos a todos los interesados, y en otros solo a los usuarios del Portal.

Objetivo inicial:

- Soportar dentro del portal de los grupos el manejo de proyectos con responsables, encargados, fechas de asignación y entrega. Incluye la creación de un servicio para ingresar las tesis de maestría por la secretaria de maestría. Permitirá usarse en el horario de profes. Primera fase.

Requisitos finales del objetivo:

- Es accesible por docentes, estudiantes y personas asociadas al proyecto, las personas ajenas al proyecto pueden acceder a una vista limitada del mismo.
- Debe tener un director de Proyecto, dar la opción de agregar un auxiliar, los responsables de las actividades, profesores asociados.
- Debe dar la opción de crear un proyecto, agregando sus componentes como lo son el título, objetivos, director de proyecto, fuentes de financiación, grupos asociados, entidades relacionadas.
- Debe permitir editar el proyecto, detener el proyecto, activar el proyecto y eliminar el proyecto, solo el director o el auxiliar de proyecto pueden editar, detener o eliminar.
- Cada proyecto tendrá Etapas y cada etapa tendrá actividades, estarán relacionadas una a la otra, por lo tanto para poder eliminar un proyecto, se tendrá que eliminar primero las actividades.
- Cada etapa debe tener una breve descripción de sí misma.
- Cada actividad debe tener un encargado o responsable, además se le asignará una semana inicial y una semana final indicando el principio y fin de la actividad, se le debe poder asignar actividades requeridas a una actividad, de tal forma que una actividad puede depender de otras actividades.
- La actividad puede ser editada o eliminada, por el director o auxiliar de Proyecto.
- Debe tener un buscador, donde muestre los proyectos asociados a un profesor o muestre los proyectos asociados a un grupo.
- Debe tener una interfaz donde se pueda acceder al proyecto y a la información básica, esta interfaz debe mostrar las actividades por semana y debe permitir publicar comentarios, mostrando la fecha en que se hicieron.

- Al hacer un comentario debe mandar un correo a los participantes de esa actividad, informándoles que hay un comentario nuevo.
- Al activar o finalizar actividades, se debe enviar un correo informando el cambio de estado de la actividad.
- Se deben tomar las fechas reales en que se activa o se finaliza una actividad.
- Cuando el proyecto esta detenido no se puede acceder a las actividades
- Debe tener una parte grafica, algo que muestre el avance del proyecto en el tiempo.
- Se debe crear un modulo donde se puedan registrar los resultados del proyecto, de tal forma que este modulo pueda usarse en otros servicios para registrar producción intelectual de los profesores y de los grupos de la escuela.
- Este modulo debe cumplir con algunas de las especificaciones que exige Colciencias en el registro de producción intelectual en el país (CvLAC).

3. El Aula Virtual soporta al docente el proceso de calificación de las actividades de la materia. Esta calificación se hace sobre actividades realizadas en subgrupos de clase. No obstante, al momento de publicar la nota y hacer los cálculos de definitiva, esta se debe individualizar para poder tener en cuenta los casos particulares y las posibles reclamaciones por parte de los estudiantes.

Objetivo inicial:

- En el módulo de notas del aula. Las notas deben individualizarse en el Aula, para esto es necesario crear 3 tablas: para talleres y exámenes, para las otras actividades y para los quices. Inicialmente el profesor o el sistema (en el caso de talleres, exámenes y quices) calcula o define la nota. Después existirá el botón asignar nota, que automáticamente copia las notas y las individualiza en una nueva tabla. Este botón se llamará "Asignar Nota" cuando se oprima por

primera vez. Después se convertirá en modificar nota, cuando ellas ya se encuentren en la nueva tabla.

Requisitos finales del objetivo:

- Solo puede acceder al servicio los dueños de un guion o el profesor de la materia.
- En el momento en que se califica una actividad del Aula, el botón califica actividad cambia por asignar nota.
- Al seleccionar asignar nota, se muestra el subgrupo de clase, el código y la foto del estudiante, se da la opción de hacer un comentario y se muestra la nota computada individualizada de cada estudiante.
- Si se hace un cambio en la nota se guarda los cambios y se le envía un correo al estudiante informándole el cambio.
- Al momento de asignar nota, el botón asignar nota cambia a modificar nota.
- Al modificar nota se muestra el subgrupo de clase, el código y la foto del estudiante, se da la opción de hacer un comentario y se muestra la nota computada individualizada de cada estudiante, el profesor puede modificar cualquier nota y comentario, se enviara un correo notificando el cambio.
- El botón asignar Nota y modificar Nota, debe colocarse en talleres y exámenes, laboratorios, foros y exposiciones.
- El aula de clase debe permitir asociar quices y asignarle un porcentaje de la nota final.
- Se debe permitir modificar el porcentaje de la nota a quices.
- Se debe modificar el modulo de quices colocando el botón asignar Nota y modificar Nota.
- En los módulos de ver notas se deben traer las notas definitivas de las tablas nuevas, evitando calcularlas en los módulos.
- Se debe mostrar las Notas definitivas a cada estudiante Por dirigir asignatura e Ingresar al aula.

Administración.

Objetivos iniciales:

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

Requisitos finales del objetivo:

Para usuarios con perfil de administrador:

- Corregir cada uno de los fallos que a diario se presentan en los portales web.
- Mantenimiento a la bases de datos para eliminar tablas que ya no se usan o crear nuevas que se necesitan para que los nuevos servicios funcionen.
- Actualizar el diagrama Entidad/Relación de la base de datos Diamante y subirlo al portal EISIWEB, EIMWEB y EIQWEB, 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, EIMWEB y EIQWEB, 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 EISIWEB, EIMWEB y EIQWEB, en los que podemos encontrar:
 - Implementar los nuevos servicios o mejoras realizadas por otros desarrolladores del grupo CALUMET, dentro de los portales EISIWEB, EIMWEB y EIQWEB, ajustando el nuevo código, modificando la Base de Datos y realizando las pruebas necesarias.

- Hacer el seguimiento del funcionamiento de los portales para corregir posibles defectos generados por errores en el código fuente que se puedan presentar y dañar el funcionamiento del sistema.
- Revisar y depurar la estructura de directorios y archivos del portal web.
- Realizar el mantenimiento de un archivo clasificado de las solicitudes de cambios y correcciones que hagan los usuarios.

Requisitos finales del objetivo:

Para usuarios con perfil de administrador:

- Revisar regularmente el historial de cambios para supervisar el uso correcto que los usuarios le dan a los servicios.
- Realizar copias periódicas de la base de datos Diamante y de los directorios de los sitios EISIWEB, EIMWEB y EIQWEB.
- 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 llevo a cabo durante la práctica.

Soporte a Usuarios.

Objetivo inicial:

- Desempeñar labores de soporte a los usuarios de los portales EISIWEB, EIMWEB y EIQWEB, brindando así solución a sus diferentes necesidades, conflictos o situaciones que se puedan presentar. Entre estas labores se destacan:
 - Capacitar usuarios y estudiantes del primer nivel en el uso de servicios dentro del portal EISIWEB, EIMWEB y EIQWEB, 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 EISIWEB, EIMWEB y EIQWEB 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 EISIWEB, EIMWEB y EIQWEB, a fin de dar solución a las posibles dificultades que se presenten.
- Investigar las soluciones a nuevas dificultades que se presenten.

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

Capacitación a nuevos integrantes del grupo Calumet.

Objetivo inicial:

- Capacitar a los estudiantes que relevaran las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro de los portales ESIWEB, EIMWEB y EIQWEB 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.

Incidencias.

- Asignación de preguntas del banco a una actividad.

Funcionalidad

Como una de las actividades evaluables dentro del aula se encuentran los talleres y exámenes, que requieren de la conformación de un banco de preguntas. Para esto existe el servicio de Crear y modificar Actividades. Por este menú se crean y agregan preguntas a las actividades de talleres y exámenes.

Problemática

Agregar preguntas del banco a un taller, por el menú actividades "modificar taller", no muestra bien todas las opciones, algunas salen vacías. Donde se agregan preguntas a taller y exámenes. Por Actividades no salen bien las opciones, y salen brs de html.

- Correos entre autores de preguntas y sus autorizadores al banco de preguntas.

Funcionalidad

Es posible la participación de estudiantes y diferentes usuarios en la creación de preguntas con destino al banco. Cuando un estudiante propone una pregunta un correo es enviado al profesor autorizador. De igual forma, cuando un autorizador revisa una pregunta y la avala, o la devuelve o la elimina un correo avisa de esto al autor.

Problemática

Los correos enviados por el banco usan siempre el mismo asunto. Debe de agregarse el Id de pregunta en el asunto para diferenciarlos. Estos son los correos que se dan entre el profesor autorizador y los estudiantes proponentes.

- Toma de datos para las estadística de respuestas de los estudiantes a las preguntas

Funcionalidad

Por cada vez que sea respondida una pregunta se lleva el número de veces respondida, y la suma de las notas obtenidas. De esta forma, el docente podrá determinar con el tiempo la complejidad de una pregunta del banco

Problemática

Calificar las preguntas abiertas no se está sumando la nota asignada. El número de veces asignadas si se está llevando, es decir, se suma en el momento en que se asigna la pregunta.

- Agregar el título del archivo subido por un usuario en el correo que se envía a los interesados

Funcionalidad

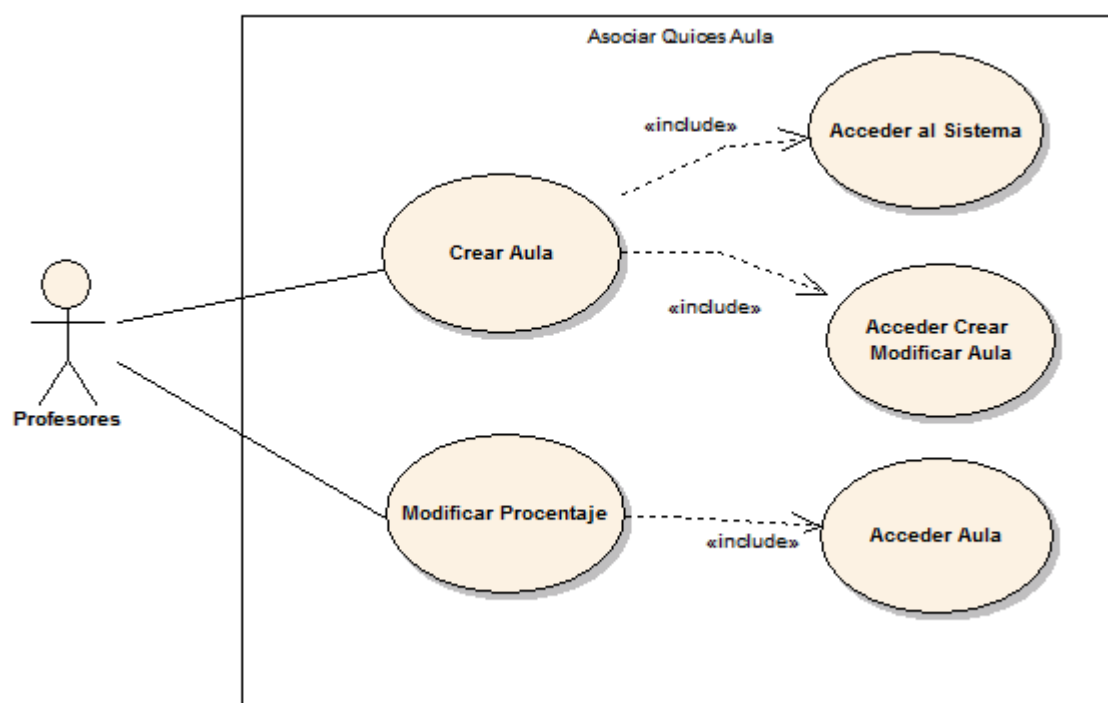
Al subir un archivo, a las diferentes opciones de la comunidad se envía un correo a los interesados indicando quien lo subió y a dónde.

Problemática

Agregar en el correo, a los interesados, el título del archivo que se suba.

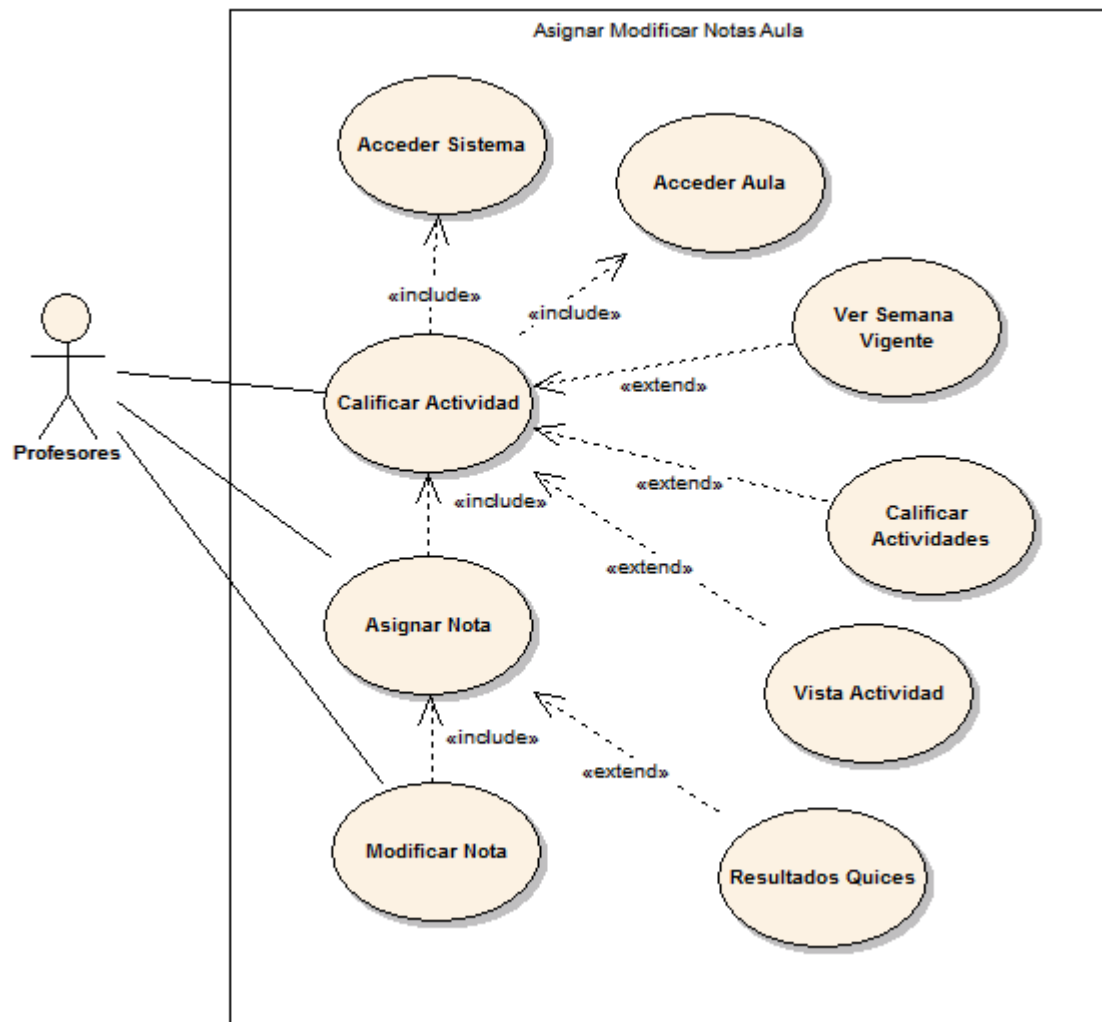
4.1.1 Diagramas de Casos de Uso.

Figura 9. Diagrama de Casos de Uso: Asociar Quices al Aula.



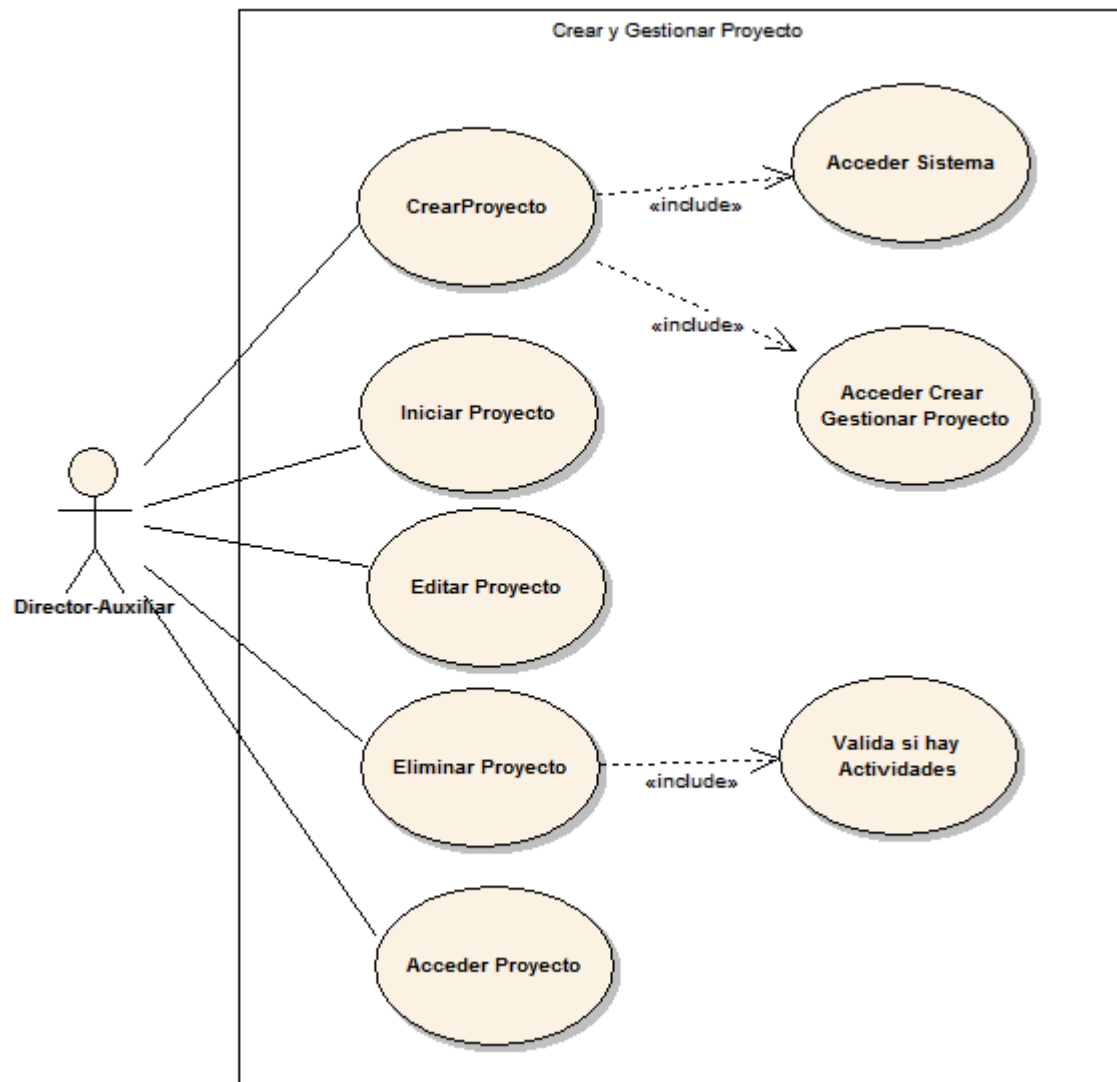
Fuente: Autores.

Figura 10. Diagrama de Casos de Uso: Asignar y Modificar Nota en el Aula.



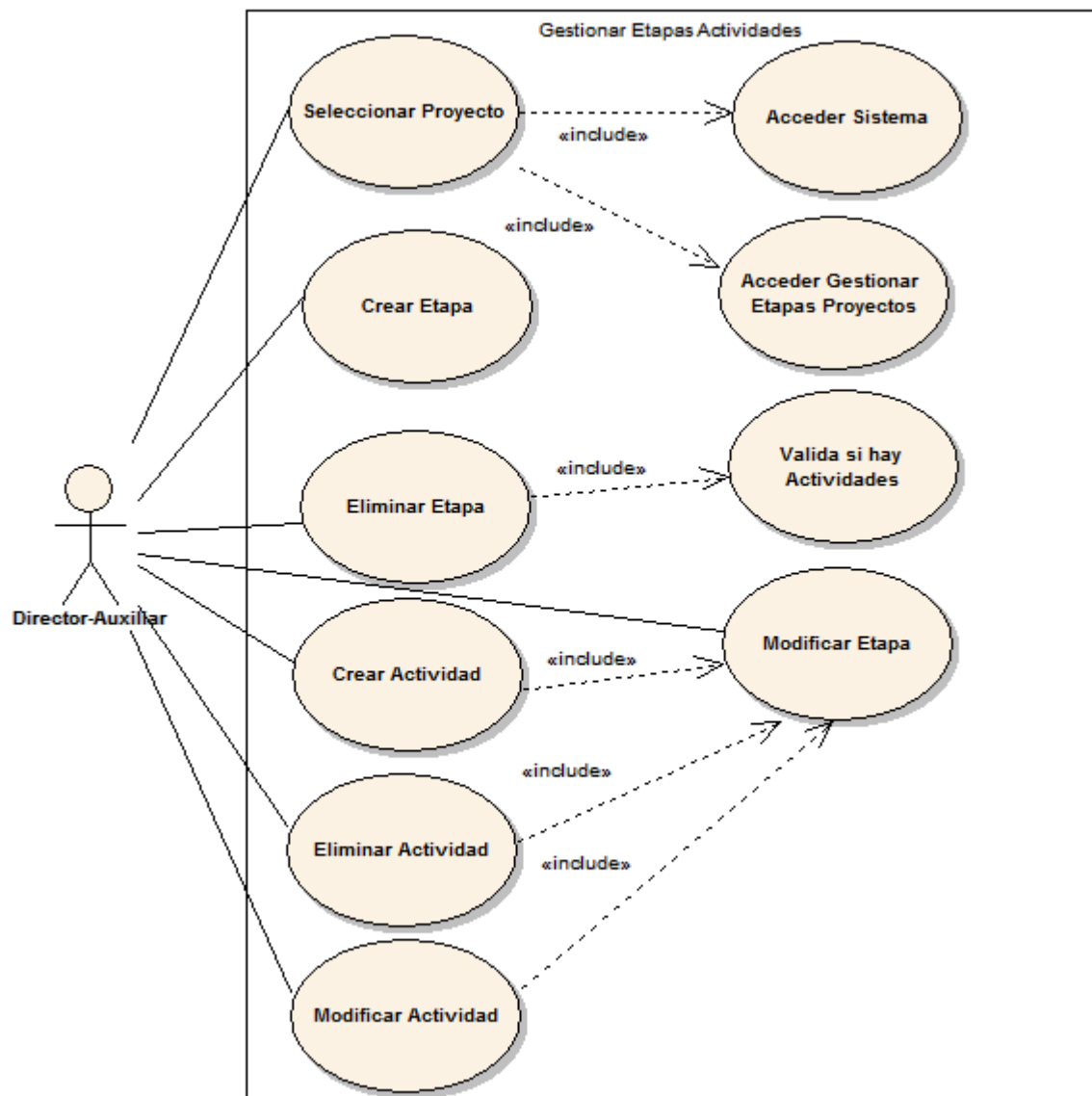
Fuente: Autores.

Figura 11. Diagrama de Casos de Uso: Crear y Gestionar Proyecto.



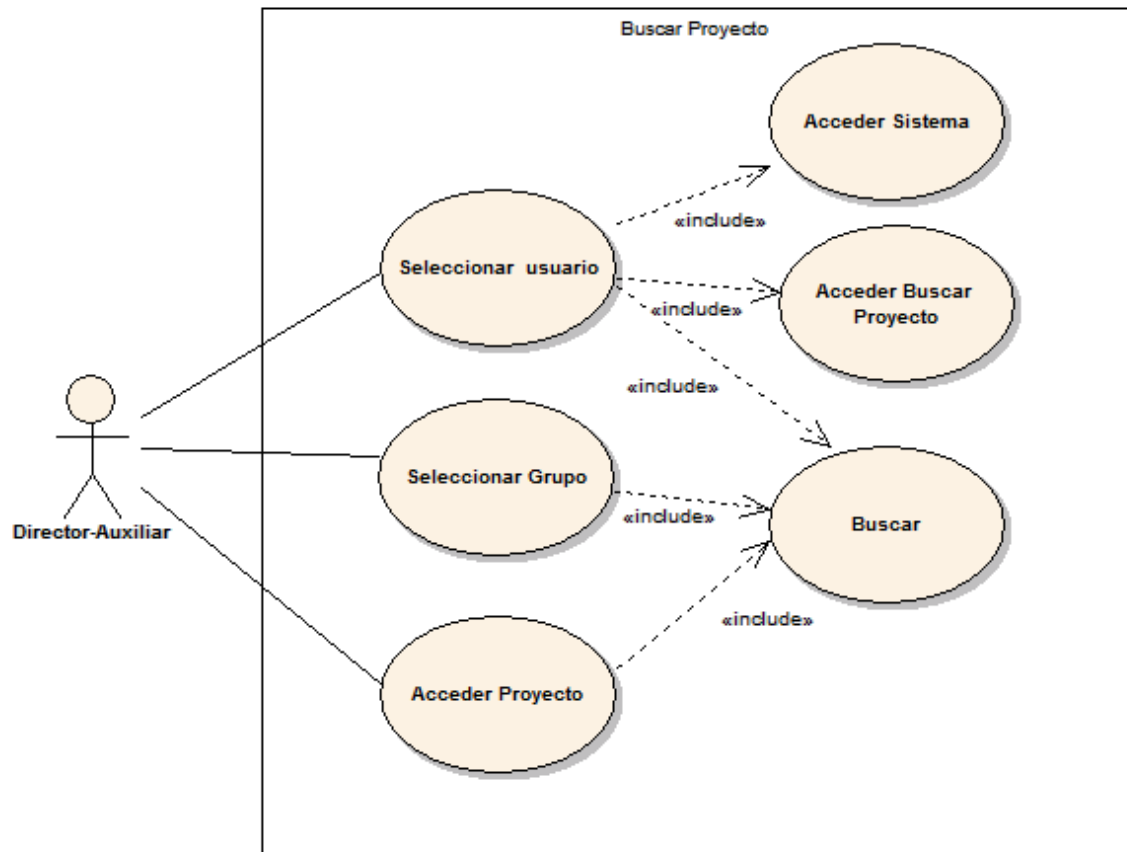
Fuente: Autores.

Figura 12. Diagrama de Casos de Uso: Gestionar Etapas y Actividades.



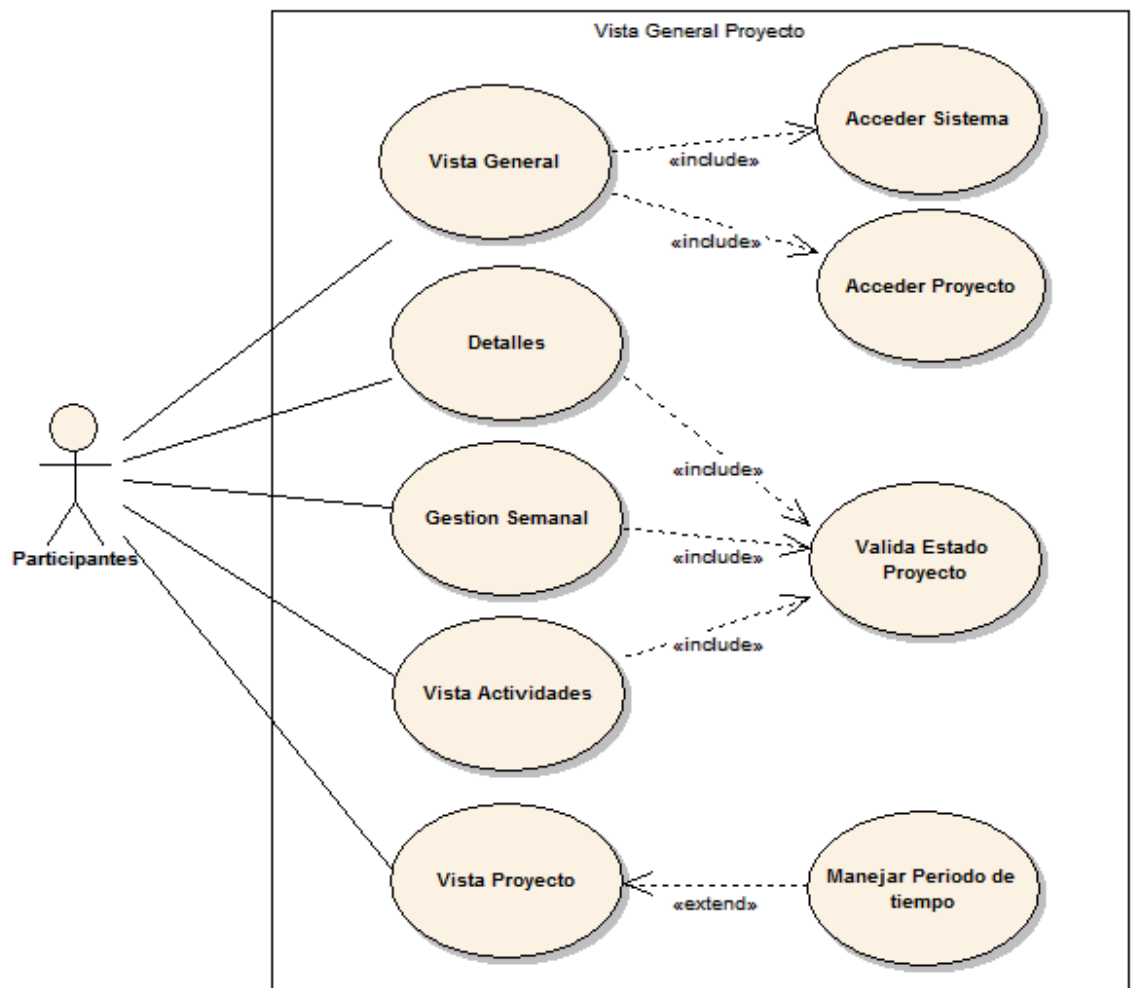
Fuente: Autores.

Figura 13. Diagrama de Casos de Uso: Buscar Proyecto.



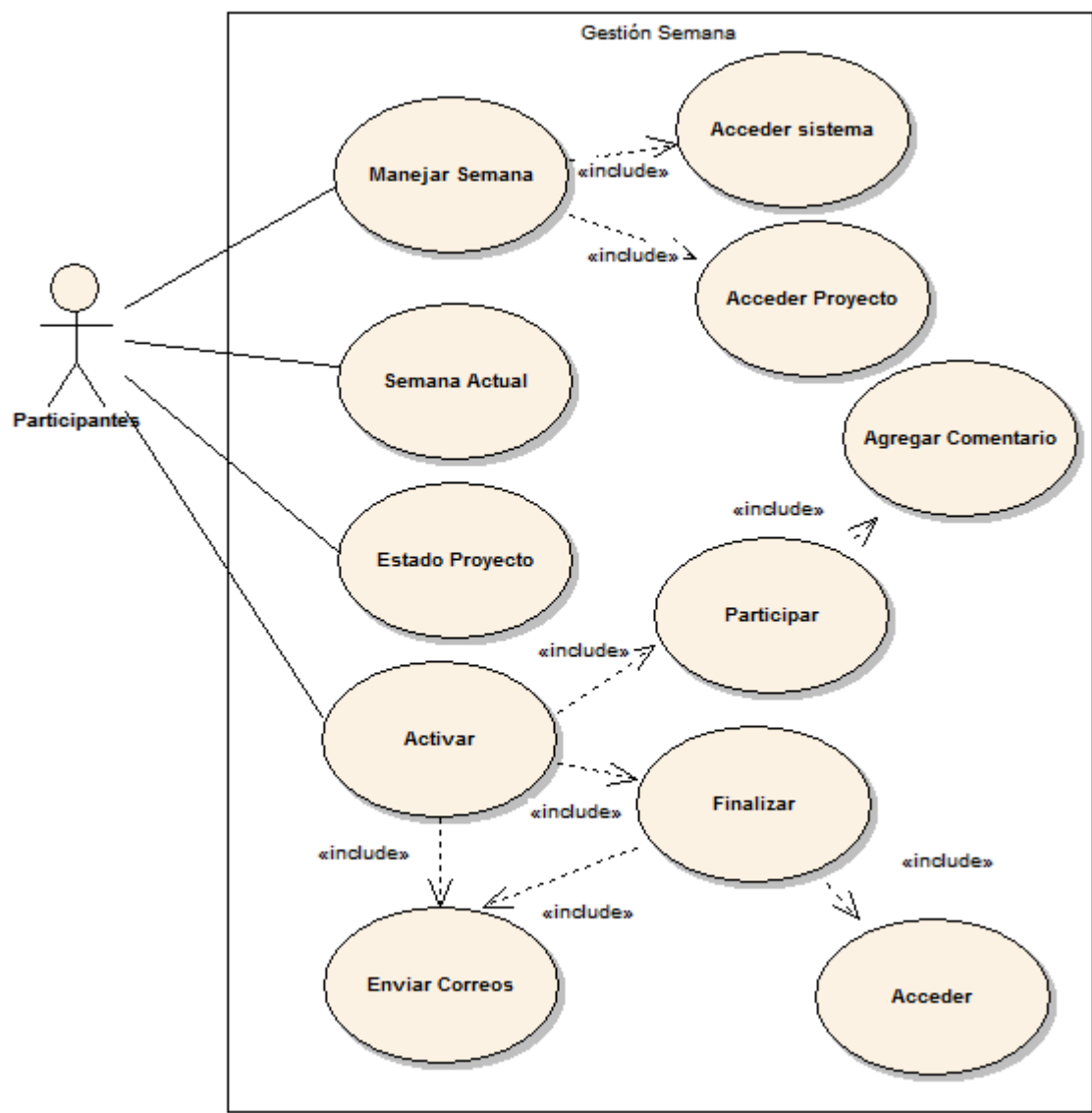
Fuente: Autores.

Figura 14. Diagrama de Casos de Uso: Vista General del Proyecto.



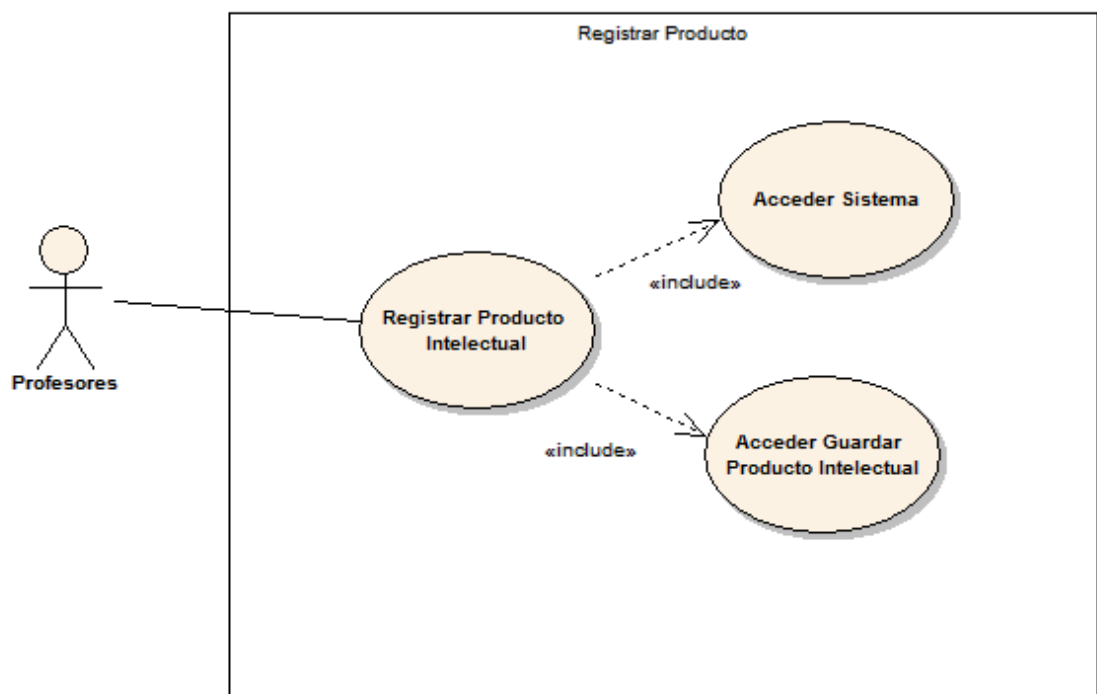
Fuente: Autores.

Figura 15. Diagrama de Casos de Uso: Gestionar Semana.



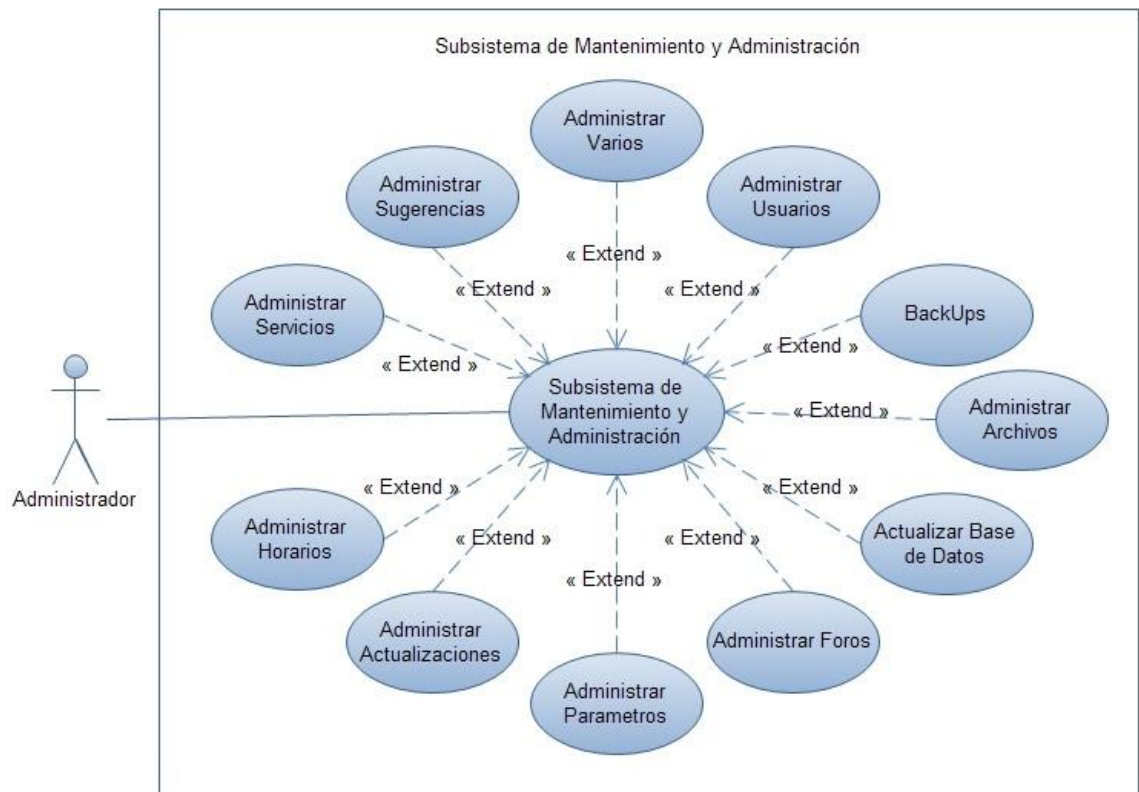
Fuente: Autores.

Figura 16. Diagrama de Casos de Uso: Registrar Producto.



Fuente: Autores

Figura 17. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.



Fuente: Autores

4.1.2 Documentación de Casos de Uso del Sistema.

4.1.2.1 Servicio para asociar Quices al Aula.

Tabla 1. Casos de uso: Asociar Quices al Aula de Clase.

TITULO	FUNCIONES PRIMARIAS
Acceder Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder Crear Modificar Aula	Acceder al servicio de aula virtual, seleccionar Crear Modificar Libro Aula
Crear Aula	Procede a crear una Nueva Aula de Clase asociándole planes y Quices.
Acceder Aula	Acceder al servicio de aula virtual, seleccionar un guion de clases e ingresar a la asignatura.
Modificar Porcentaje	Permite modificar los porcentajes de la nota final a los Planes y Quices del Aula.

Fuente: Autores.

4.1.2.2 Servicio para asignar y modificar notas en el Aula.

Tabla 2. Casos de uso: Asignar y Modificar Nota en el Aula.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder Aula	Acceder al servicio de aula virtual, seleccionar un guion de clases e ingresar a la asignatura.
Ver Semana Vigente	Listar todo lo que está ocurriendo en la semana vigente. La cartelera de la asignatura y las actividades.
Vista Actividad	Lista las actividades del Plan Seleccionado.
Calificar Actividades	Lista las actividades Calificadas y Pendientes de la asignatura.
Resultados Quices	Lista los quices finalizados en la asignatura.
Calificar Actividad	Muestra una lista de los subgrupos de clase , les construye y muestra la nota a cada subgrupo
Asignar Nota	Muestra una lista de los estudiantes de clase, individualiza y muestra la nota a cada estudiante.
Modificar Nota	Permite editar las notas individualizadas a cada estudiante.

Fuente: Autores.

4.1.2.3 Servicio que permite Crear y Gestionar un Proyecto.

Tabla 3. Casos de uso: Crear y Gestionar Proyecto.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder Crear Gestionar Proyecto	Acceder al servicio de Gestor de Proyectos y seleccionar Crear Gestionar Proyecto.
Crear Proyecto	Permite crear un proyecto nuevo para que el profesor lo gestione como el desee creando Etapas.
Editar Proyecto	Permite editar los proyectos ya existentes.
Eliminar Proyecto	Borrar registro de bases de datos del Proyecto, Etapas y actividades que el director haya creado.
Inicia Proyecto	Inicia el Proyecto, asigna fechas a todas las actividades y permite el acceso a su contenido.
Acceder Proyecto	Ingresa al contenido del Proyecto como Participante o usuario normal.
Valida si hay Actividades	Verifica que el proyecto no tenga ninguna actividad.

Fuente: Autores.

4.1.2.4 Servicio que permite Gestionar Etapas y Actividades.

Tabla 4. Casos de Uso: Gestionar Etapas y Actividades.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder a Gestionar Etapas Proyecto	Acceder al servicio de Gestor de Proyectos y seleccionar gestionar Etapas y Actividades.
Seleccionar Proyecto	Lista todos los proyectos que tiene el Director o auxiliar de Proyecto y permite escoger uno.
Crear Etapa	Crea una Etapa nueva al proyecto para que el usuario le asocie las actividades que quiera.
Modificar Etapa	Permite editar las Etapas ya existentes.
Eliminar Etapa	Borra registro de la base de datos de la Etapa y las actividades asociadas.
Crear Actividad	Crea una actividad nueva y es asociada a la etapa correspondiente y otras actividades.
Modificar Actividad	Permite editar las actividades ya existentes.

Eliminar Actividad	Borra registro de la base de datos de las actividades.
Valida si hay actividades	Verifica que la etapa no tenga ninguna actividad

Fuente: Autores.

4.1.2.5 Servicio que permite Buscar Proyectos.

Tabla 5. Caso de Uso: Buscar Proyecto.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder Buscar Proyecto	Acceder al servicio de Gestor de Proyectos y seleccionar gestionar Buscar Proyecto.
Seleccionar Usuario	Lista los usuarios que tienen Proyecto y permite escoger uno.
Seleccionar Grupo	Lista los Grupos que tienen Proyecto y permite escoger uno.
Buscar	Muestra los Proyectos o el Proyecto del usuario o Grupo seleccionado.

Acceder Proyecto	Ingresa al contenido del Proyecto como Participante o usuario normal.
------------------	---

Fuente: Autores.

4.1.2.6 Servicio para mostrar la Vista General del Proyecto.

Tabla 6. Casos de uso: Vista General Proyecto.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresa al sitio web utilizando un nombre de usuario y una contraseña.
Acceder al Proyecto	Acceder al servicio de Gestor de Proyectos, seleccionar un Proyecto y acceder.
Vista General	Muestra información del Proyecto, Estado, Semana Actual, fin de semana y actividades activas.
Detalles	Muestra detalles del proyecto tales como el título, objetivos, Director, auxiliar, tiempo estimado.
Vista Proyecto	Muestra un gráfico de las actividades del Proyecto, la línea de tiempo y la cantidad de semanas.
Vista Actividades	Lista las Etapas del proyecto con sus respectivas actividades.
Vista Semanal	Lista las actividades que están vigentes en la semana actual, permite escoger una, además ir a semanas en el futuro o en el pasado.

Manejar Periodo Tiempo	Permite adelantar o retrasar un periodo de 24 semanas en la línea del tiempo del proyecto.
Valida Estado Proyecto	Valida si el estado del Proyecto es Detenido.

Fuente: Autores.

4.1.2.7 Servicio que Gestiona la Semana.

Tabla 7. Casos de uso: Gestionar Semana

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder al Proyecto	Acceder al servicio de Gestor de Proyectos, seleccionar un Proyecto y acceder.
Manejar Semana	Muestra las actividades que hay para la semana, permite cambiar a una semana futura o pasada.
Semana Actual	Muestra las Actividades de la semana Actual.
Estado Proyecto	Permite activar o Detener el proyecto, junto con sus actividades y etapas, asignando fechas automáticamente o seleccionadas por el usuario.
Activar	Activa la Las actividades, permite finalizar y participar en ellas.
Finalizar	Finaliza las actividades, permite acceder a las actividades.
Participar	Muestra todos los comentarios realizados en la actividad, y permite hacer uno nuevo.
Acceder	Muestra todos los comentarios realizados en

	la actividad.
Agregar comentario	Agrega un comentario a la actividad junto con la fecha y el nombre que lo hizo.
Enviar correos	Envía un correo a los participantes de la actividad

4.1.2.8 Servicio que registra los Productos Intelectuales.

Tabla 8. Casos de uso: Registrar Producto.

TITULO	FUNCIONES PRIMARIAS
Acceder al Sistema	Ingresar al sitio web utilizando un nombre de usuario y una contraseña.
Acceder Registrar Producto	Acceder al servicio de Gestor de Proyectos, seleccionar Guardar Producto Int
Registrar Producto	Permite Guardar en base de datos los productos intelectuales, para publicarlos en un futuro.

Fuente: Autores.

4.1.2.9 Mantenimiento y Administración de los portales EISIWEB, EIMWEB y EIQWEB

Tabla 8. Casos de uso: Mantenimiento y Administración de los portales EISIWEB, EIMWEB y EIQWEB.

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

Administrar conversaciones en foros	Ver información y eliminar conversaciones de los usuarios en foros.
Administrar parámetros	Crear, modificar y eliminar los diferentes parámetros del portal.
Administrar actualizaciones	Ver historial y estadísticas de cambios del portal web.
Administrar horarios	Cambiar, restaurar, ver incoherencias y cambios de horarios. Crear grupo de materia.
Editar información estática	Insertar, modificar y eliminar información para los usuarios.

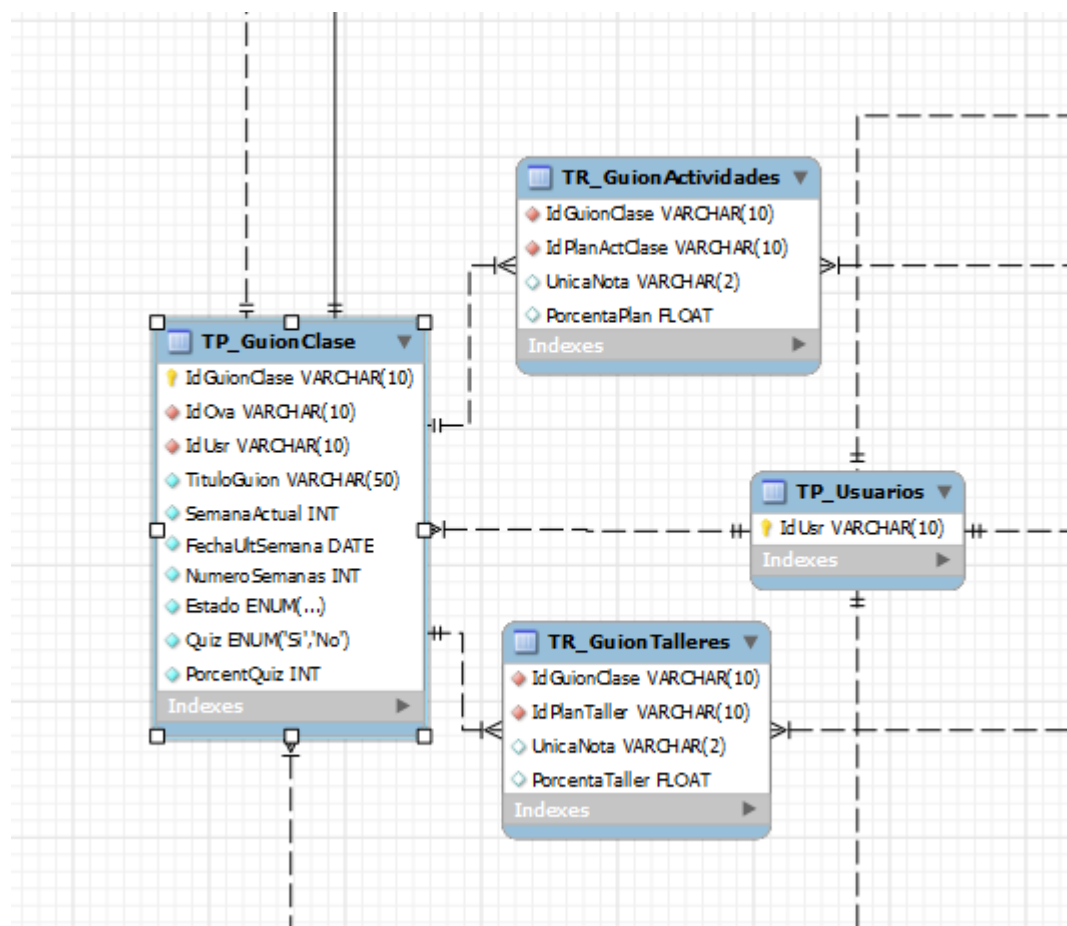
Fuente: Autores.

4.1.3 Diseño y Análisis.

4.1.3.1 Diagrama Entidad/Relación de los servicios desarrollados.

- Servicio para asociar Quices al Aula:

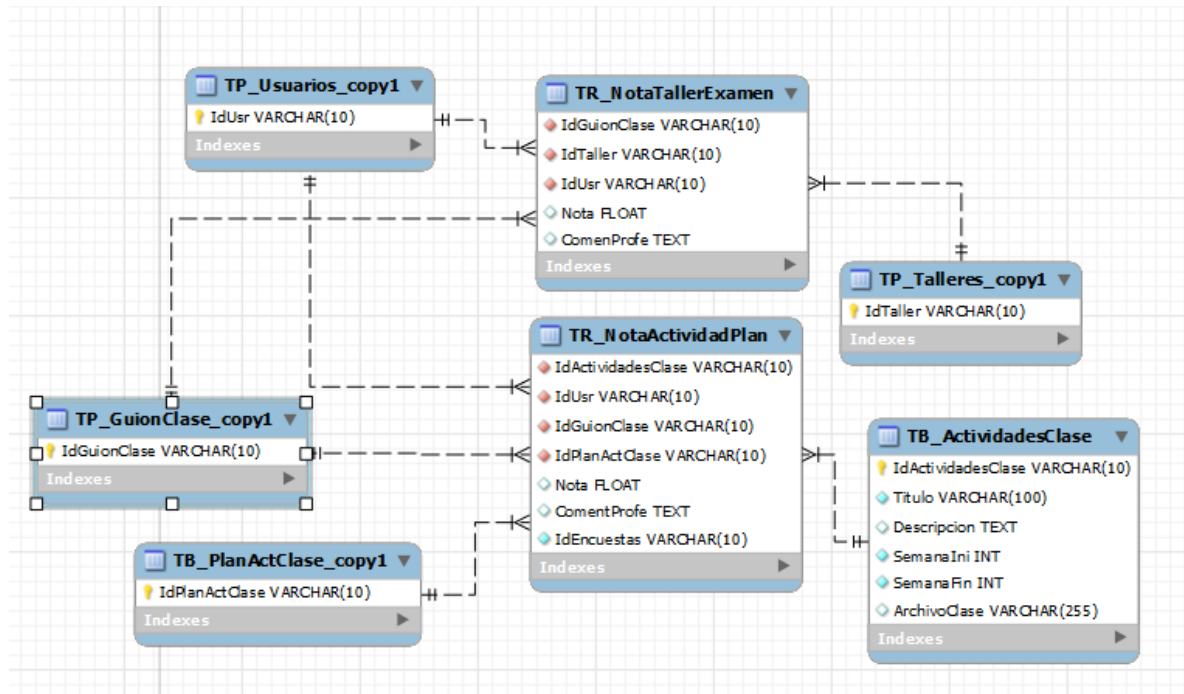
Figura 15. Diagrama E/R: Asociar Quices al Aula.



Fuente: Autores.

- Servicio para Asignar y Modificar Nota en el Aula:

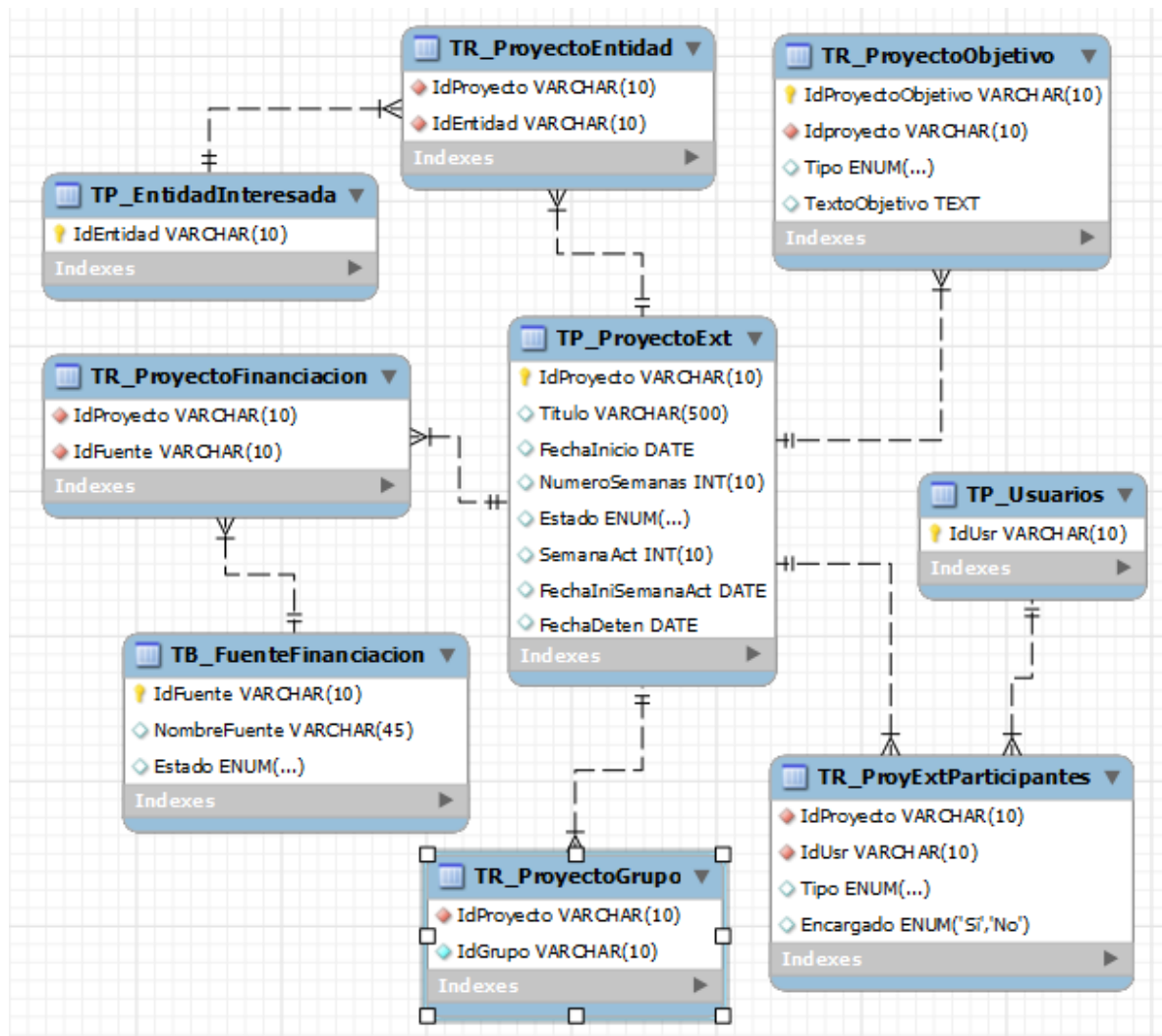
Figura 16. Diagrama E/R: Asignar y Modificar Notas en el Aula



Fuente: Autores.

- Servicio que permite Crear y Gestionar un Proyecto.

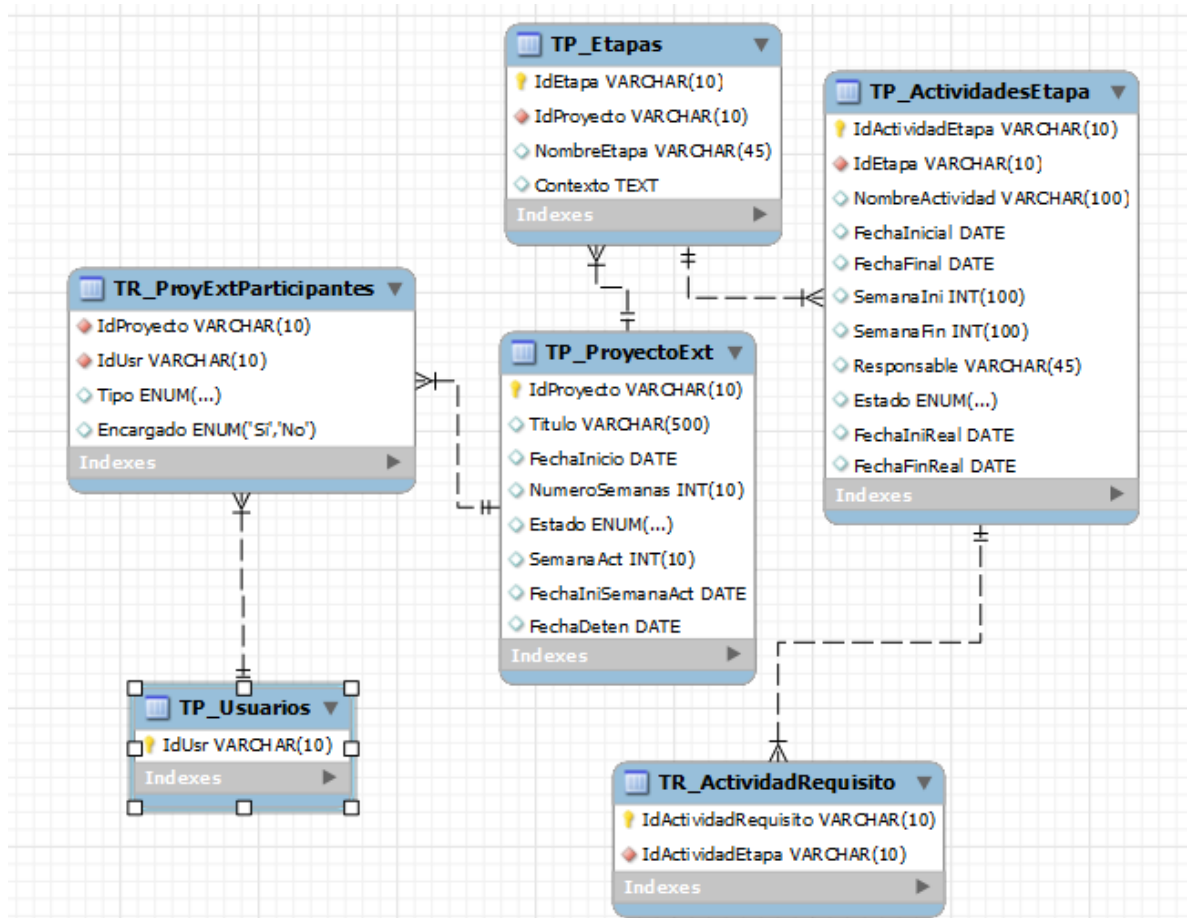
Figura 17. Diagrama E/R: Crear y Gestionar un Proyecto.



Fuente: Autores.

- Servicio que permite Gestionar Etapas Actividades:

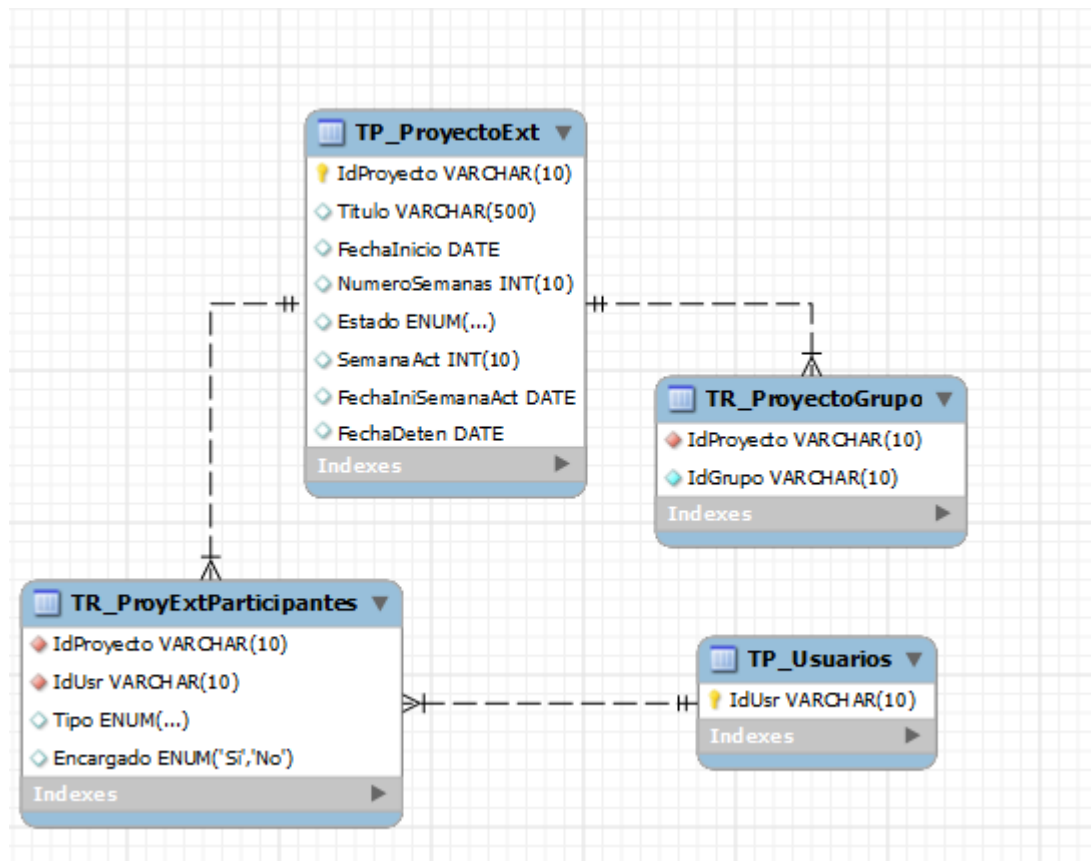
Figura 18. Diagrama E/R: Gestionar Etapas y Actividades.



Fuente: Autores.

- Servicio que permite buscar un proyecto:

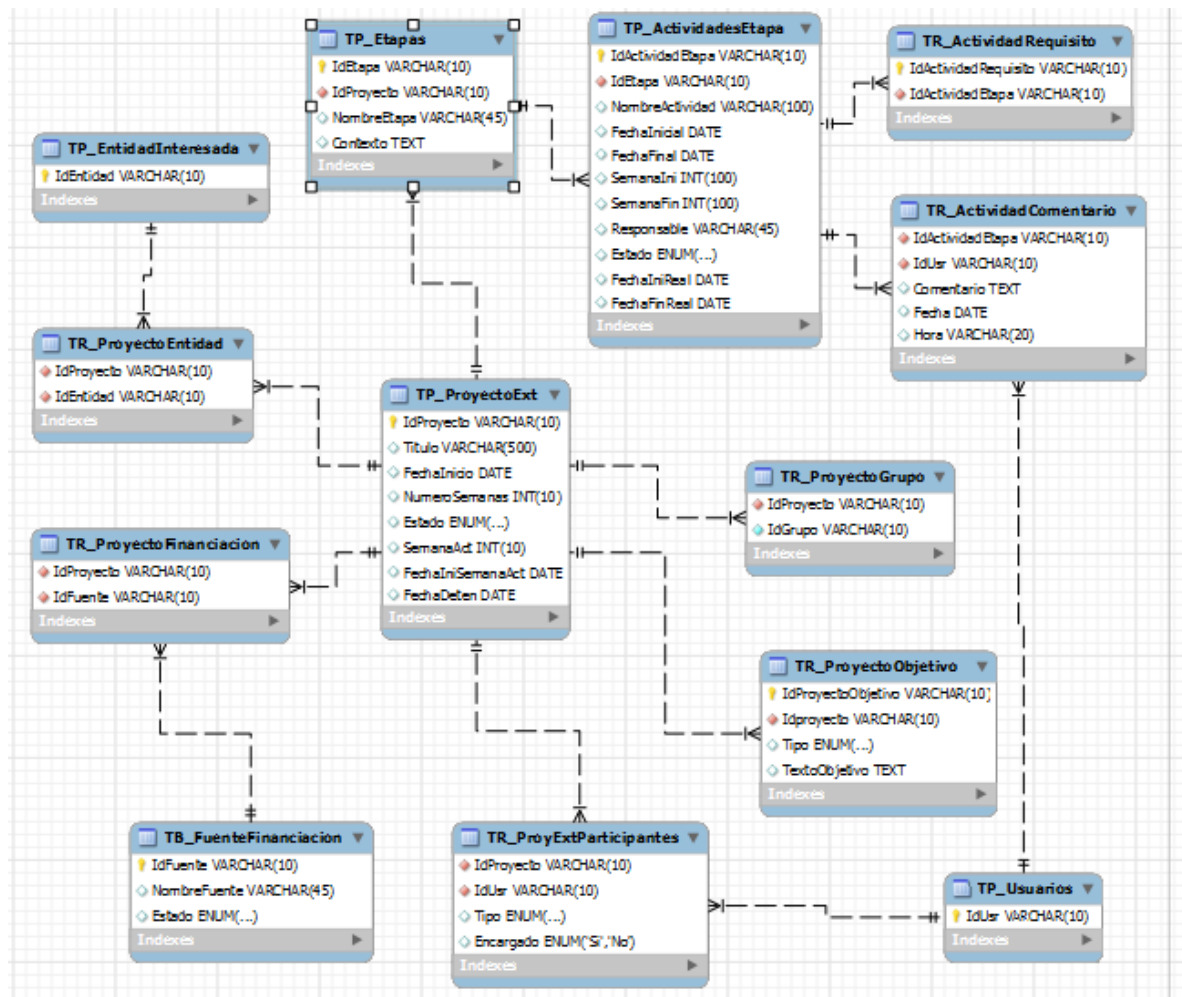
Figura 19. Diagrama E/R: Buscar Proyectos.



Fuente: Autores.

- Servicio para mostrar la Vista General del Proyecto:

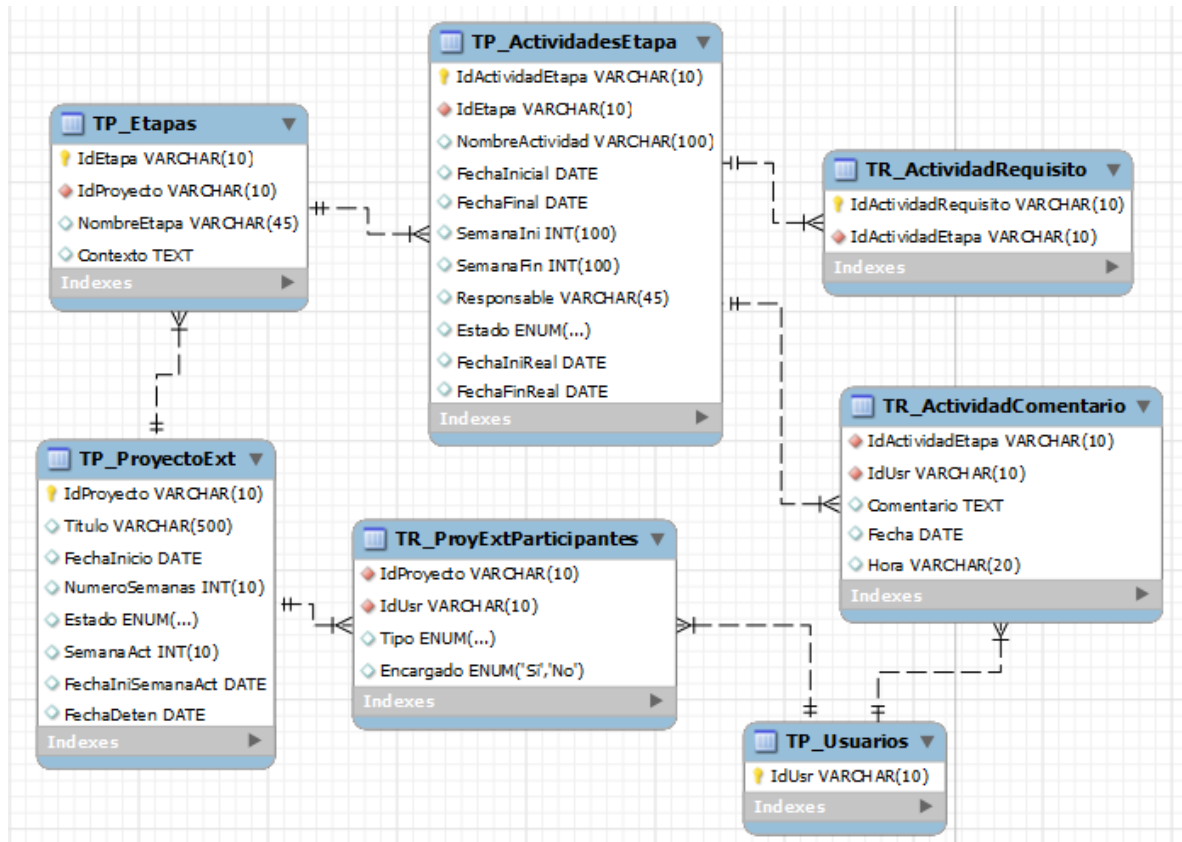
Figura 20. Diagrama E/R: Vista General del Proyecto.



Fuente: Autores.

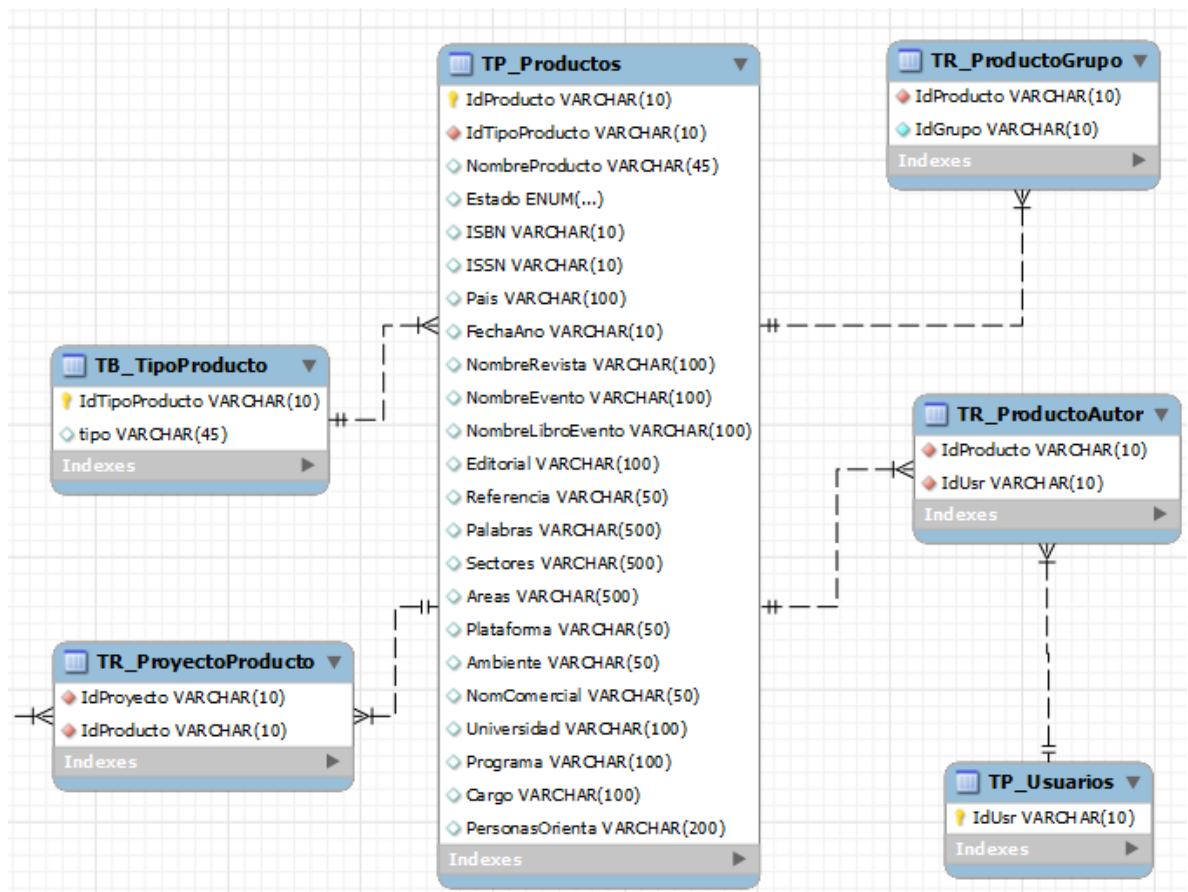
- Servicio que permite Gestionar el trabajo de la Semana:

Figura 21. Diagrama E/R: Gestionar Semana.



- Servicio que permite Registrar un Producto:

Figura 25. Diagrama E/R: Registrar Producto.



Fuente: Autores.

4.1.3.2 Descripción de las Entidades.

Definición de las entidades utilizadas y creadas en la base de datos Diamante de los sitios EISiWeB, EIMWeb y EIQWeB.

Tabla 9. Descripción de las Entidades

ENTIDAD	DESCRIPCIÓN
TB_Grupos	Contiene los grupos de investigación, extensión, culturales y académicos de las escuelas.
TB_FuenteFinanciacion	Contiene las Entidades que financian los Proyectos.
TB_TipoProducto	Contiene las clases de Productos Intelectuales que registran los profesores.
TP_GuionClase	Esta tabla contiene los guiones de clase con su respectivo nombre, dueño, el objeto virtual de aprendizaje asociado, la fecha de la última semana activa y el estado, si el guion tiene quices y su respectivo porcentaje.
TP_ActividadesEtapa	Esta tabla contiene las actividades del proyecto, sus fechas iniciales y finales, el responsable de la actividad y a que etapa pertenece.
TP_Etapas	Se guardan las Etapas de los proyectos con su descripción.
TP_Productos	Esta tabla contienen todos los Productos con su respectiva información.

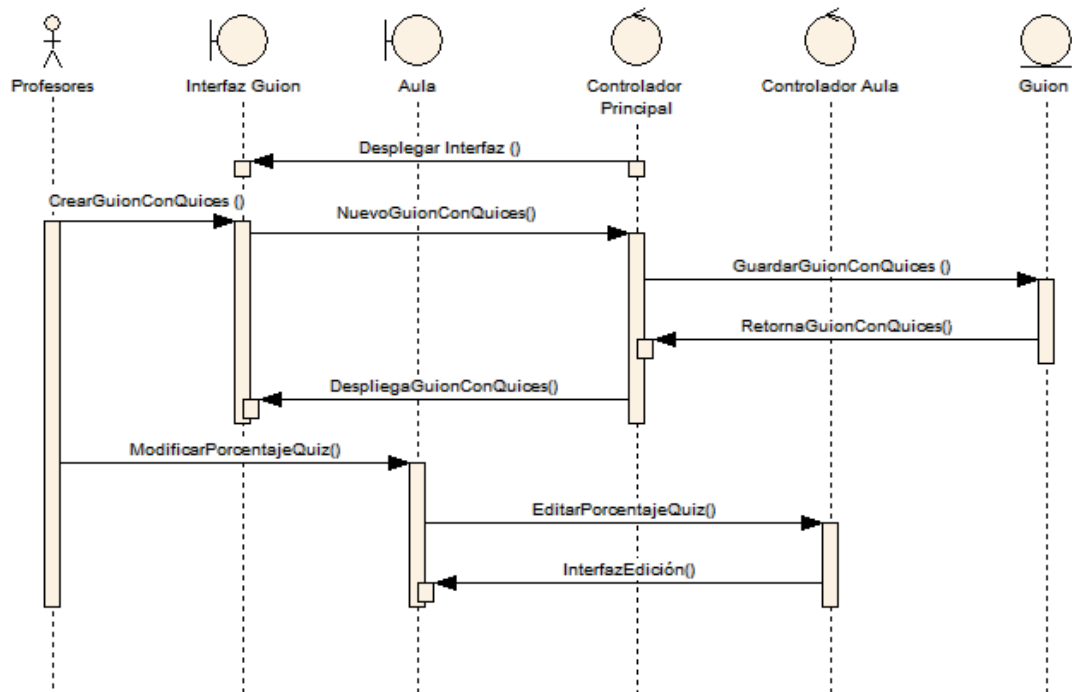
TP_ProyectoExt	Esta tabla contiene los proyectos, el título, fecha de inicio, la semana Actual y la fecha de inicio de la semana Actual.
TR_ActividadComentario	Guarda los comentarios que hacen los participantes de las actividades, la fecha y hora de la publicación.
TR_ActividadRequisito	Guarda las actividades que son requisito para otra actividad.
TR_ProductoAutor	Guarda la relación entre los productos intelectuales y los usuarios.
TR_ProductoGrupo	Guarda la relación entre los productos intelectuales y los grupos de investigación.
TR_ProyectoEntidad	Guarda la relación entre los proyectos y las entidades interesadas en el proyecto.
TR_ProyectoFinanciacion	Guarda la relación entre el proyecto y las fuentes de financiación.
TR_ProyectoGrupo	Guarda la relación entre los proyectos y los grupos de investigación.
TR_ProyectoObjetivo	Guarda la relación entre los proyectos y los objetivos, el tipo de objetivo y el objetivo.
TR_ProyectoProducto	Guarda la relación entre los proyectos y los Productos intelectuales.

TR_ProyExtParticipantes	Guarda la relación entre el Proyecto y los participantes del proyecto, el tipo de participante y si es encargado o no.
TR_NotaTallerExamen	Contiene las Notas individualizadas de los talleres y exámenes de cada estudiante que pertenece al Guion de clase, junto con un comentario del profesor.
TR_NotaActividadPlan	Contiene las Notas individualizadas de las actividades, foros, exposiciones y quices de cada estudiante que pertenece al Guion de clase, junto con un comentario del profesor.
TR_GuionActividades	Acá se guardan todas las actividades del Guion de clase
TR_GuionTalleres	Esta tabla guarda los porcentajes de talleres y exámenes.
TR_CalifNotaxActividad	La tabla contiene la nota de las actividades de un plan.
TR_NotaTallerExamen	Guarda la nota que el sistema construye para los talleres y exámenes y la que el profesor asigna.
TR_CalifActNotaxPlan	Guarda la nota que el profesor le pone a los planes que tienen única nota.

Fuente: Autores.

4.1.4 Modelo de Procesos del Sistema

Figura 221. Diagrama de secuencia: Asociar Quices al Aula.

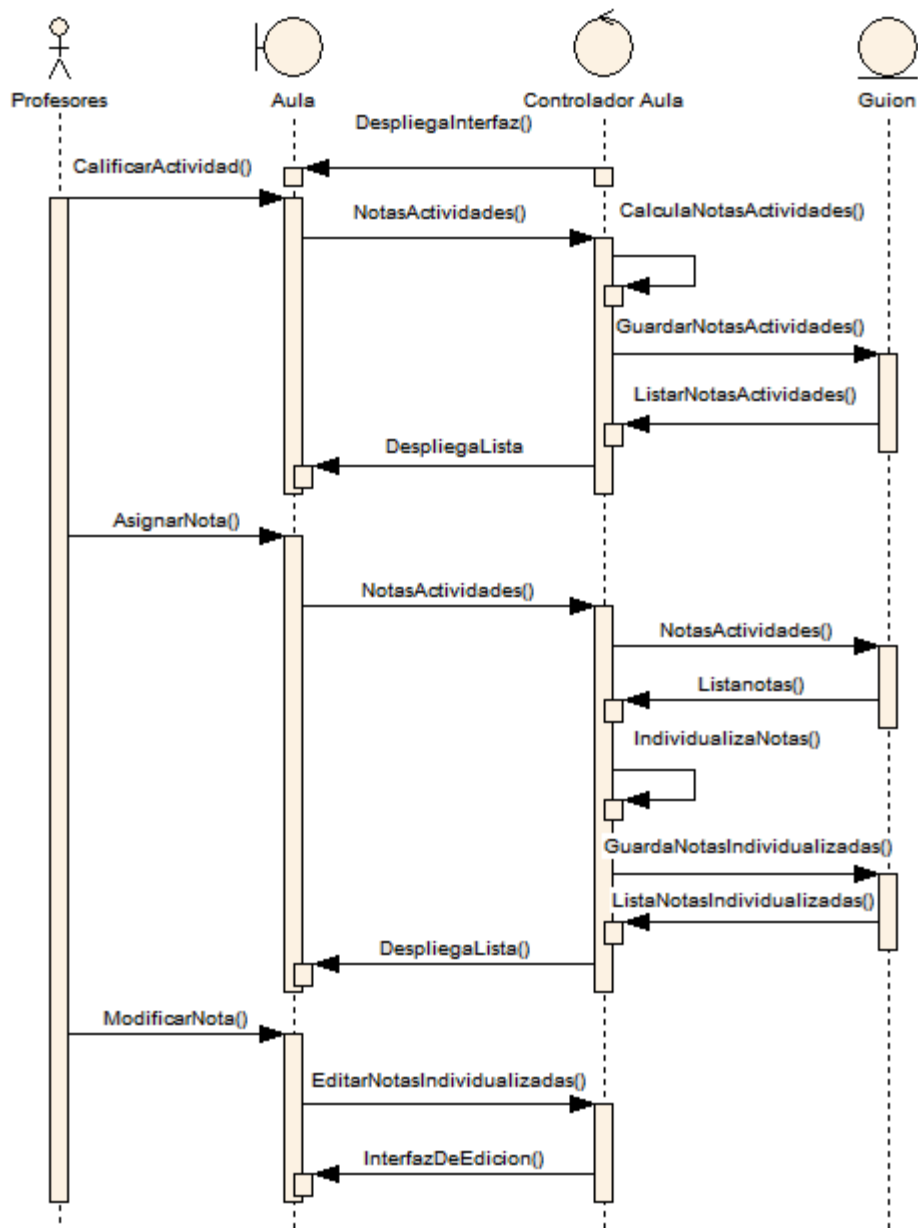


Fuente: autores.

1. Una vez los profesores entran al aula el controlador principal despliega la interfaz del Guion de clase.
2. El Profesor crea un nuevo Guion de Clase asociándole Quices y su porcentaje, el controlador principal recibe el nuevo Guion y pide guardarlo en la Base de datos.
3. Cuando está guardado el controlador principal retorna el guion y se lo despliega al profesor.
4. Ahora que el Guion está guardado con sus actividades el profesor puede acceder para modificar los porcentajes, el controlador del aula despliega la interfaz.

5. En la edición el profesor solo podrá modificar los porcentajes de las actividades incluyendo los quices.

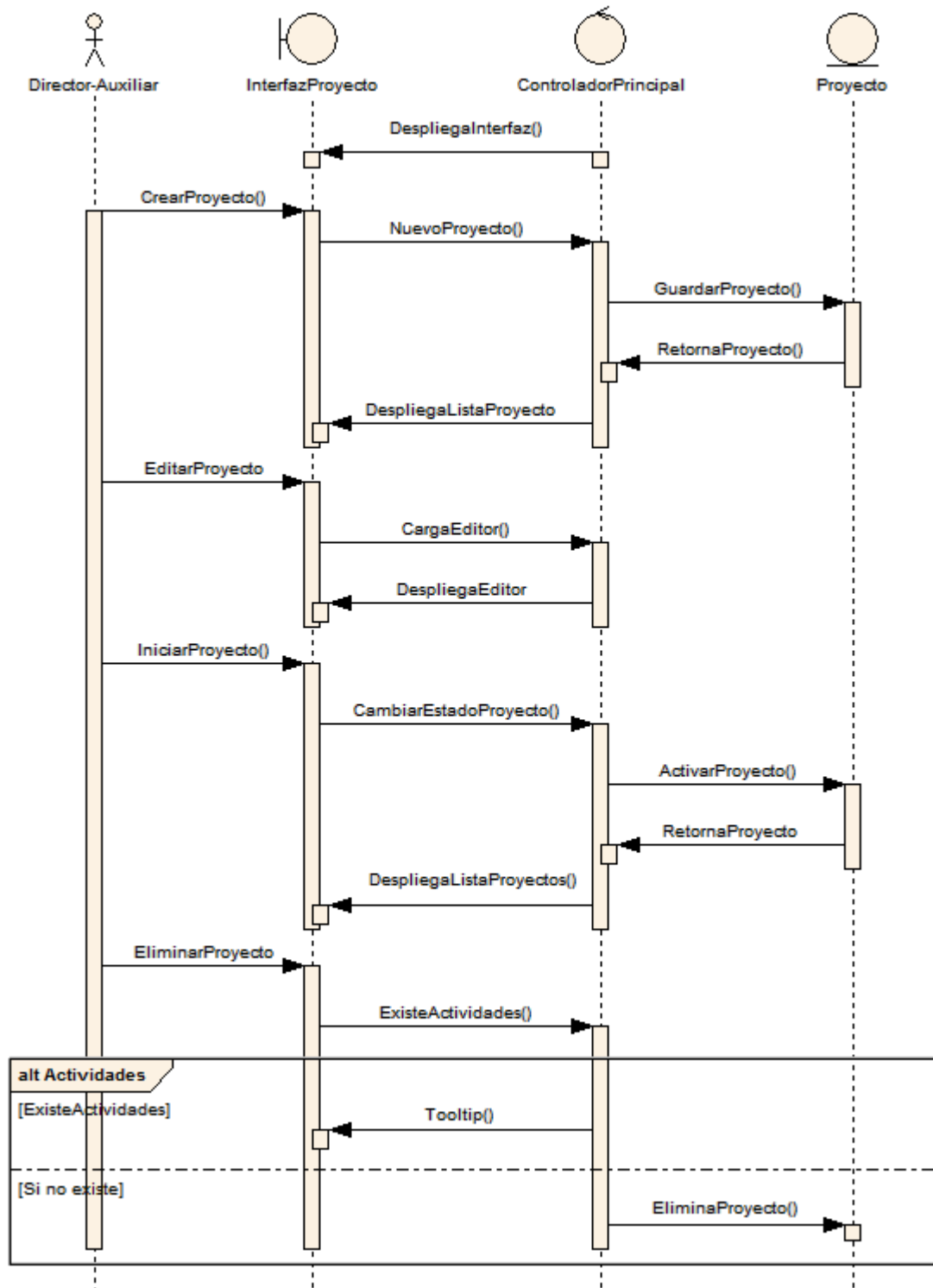
Figura 23. Diagrama de secuencia: Asignar y Modificar Nota en el Aula.



Fuente: autores.

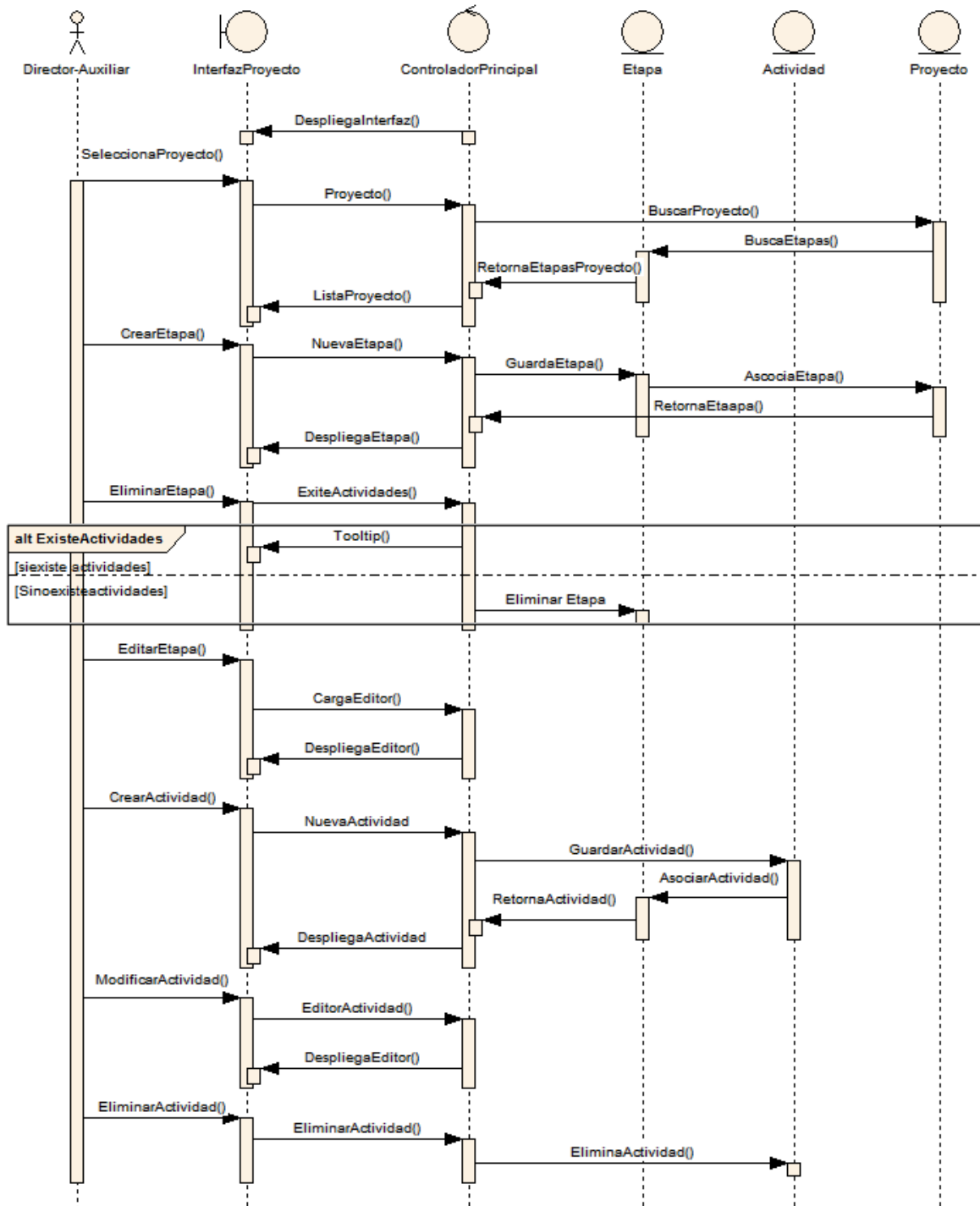
1. Una vez los profesores entra al aula el controlador del aula despliega la interfaz de las actividades para la semana vigente.
2. El profesor puede acceder a las actividades, y se le despliega una lista con las actividades vigentes, el profesor ahora puede calificar la actividad.
3. Cuando se Califica la actividad el controlador del aula recibe las notas del profesor y calcula las notas y las guarda en las bases de datos.
4. Cuando está guardada la nota de la actividad el controlador principal se la despliega al profesor.
5. Ahora que las notas están guardadas el profesor puede Asignar las notas de los Estudiantes, El controlador del aula trae las notas de las bases de datos luego las Individualizadas y guarda nuevamente las notas individualizadas en la bases de datos.
6. Una vez se guardan las notas el controlador le despliega la lista de notas individualizadas al profesor.
7. El profesor puede acceder a modificar las notas individualizadas, el controlador del aula despliega la interfaz.

Figura 24. Diagrama de secuencia: Crear y Gestionar un Proyecto



1. Una vez el director ingresan al gestor de proyectos, el controlador principal despliega la interfaz de Crear Gestionar Proyectos.
2. Cuando se despliega la interfaz el profesor puede escoger Crear Nuevo Proyecto, el controlador los recibe y pide guardarlo en la base de datos.
3. Una vez el proyecto es almacenado en la base de datos, el director puede acceder a editarlo, el controlador despliega la interfaz de edición.
4. Estando el proyecto en la base de datos, el profesor puede Iniciar el proyecto, el controlador recibe el proyecto y le cambia el estado a activo, para que puedan participar de él.
5. El profesor puede decidir eliminar el proyecto, una vez que este almacenado en la base de datos.
6. El controlador verifica si el proyecto tiene actividades, si no tiene actividades el controlador elimina el proyecto de la base de datos.
7. El director Puede Acceder al Proyecto, el controlador despliega la interfaz de Vista General del Proyecto.

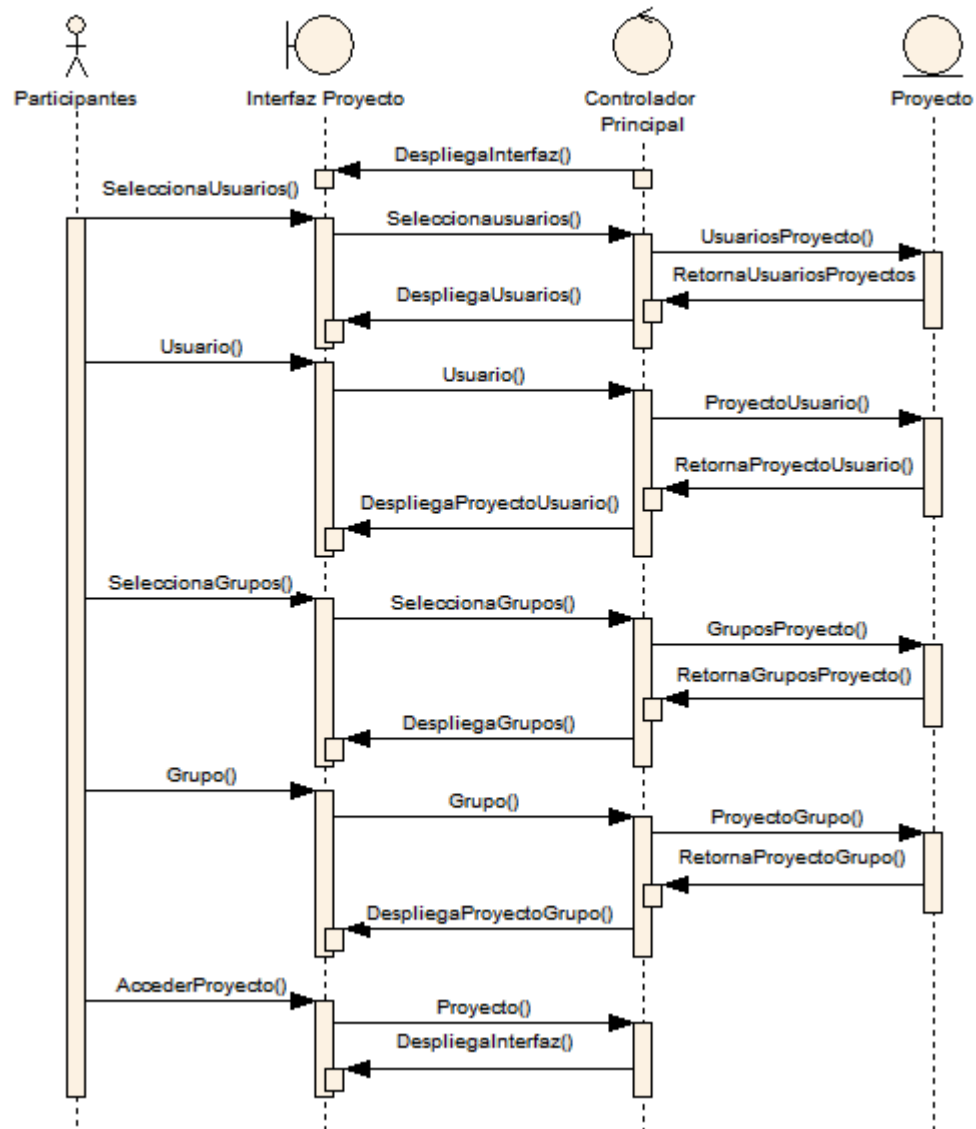
Figura 25. Diagrama de secuencia: Gestionar Etapas y Actividades.



Fuente: Autores.

1. Una vez el director ingresan al gestor de proyectos, el controlador principal despliega la interfaz de Crear Gestionar Etapas.
2. El Director selecciona un proyecto para asociarle etapas, el controlador recibe el proyecto carga las etapas relacionadas y se las despliega al director o auxiliar.
3. El director puede crear una etapa nueva, el controlador recibe la etapa, la guarda en base de datos y la asocia al proyecto, luego carga el proyecto con las etapas asociadas.
4. Una vez la etapa está en base de datos el directo puede eliminarla, el controlador revisa si no hay actividades asociadas, si no hay elimina la actividad.
5. El director puede acceder a modificar la etapa, el controlador recibe la etapa y despliega la interfaz de edición.
6. Estando en la interfaz de modificar etapa, el director puede Crear una actividad nueva, el controlador guarda la actividad en la base de datos y la asocia a la etapa, luego despliega la interfaz de crear una actividad.
7. Estando la actividad en la base de datos, el profesor puede Editarla actividad, el controlador recibe la actividad y despliega la interfaz de edicion.
8. El profesor puede eliminar la actividad, el controlador recibe la actividad y la elimina de base de datos.

Figura 26. Diagrama de secuencia: Buscar Proyecto.

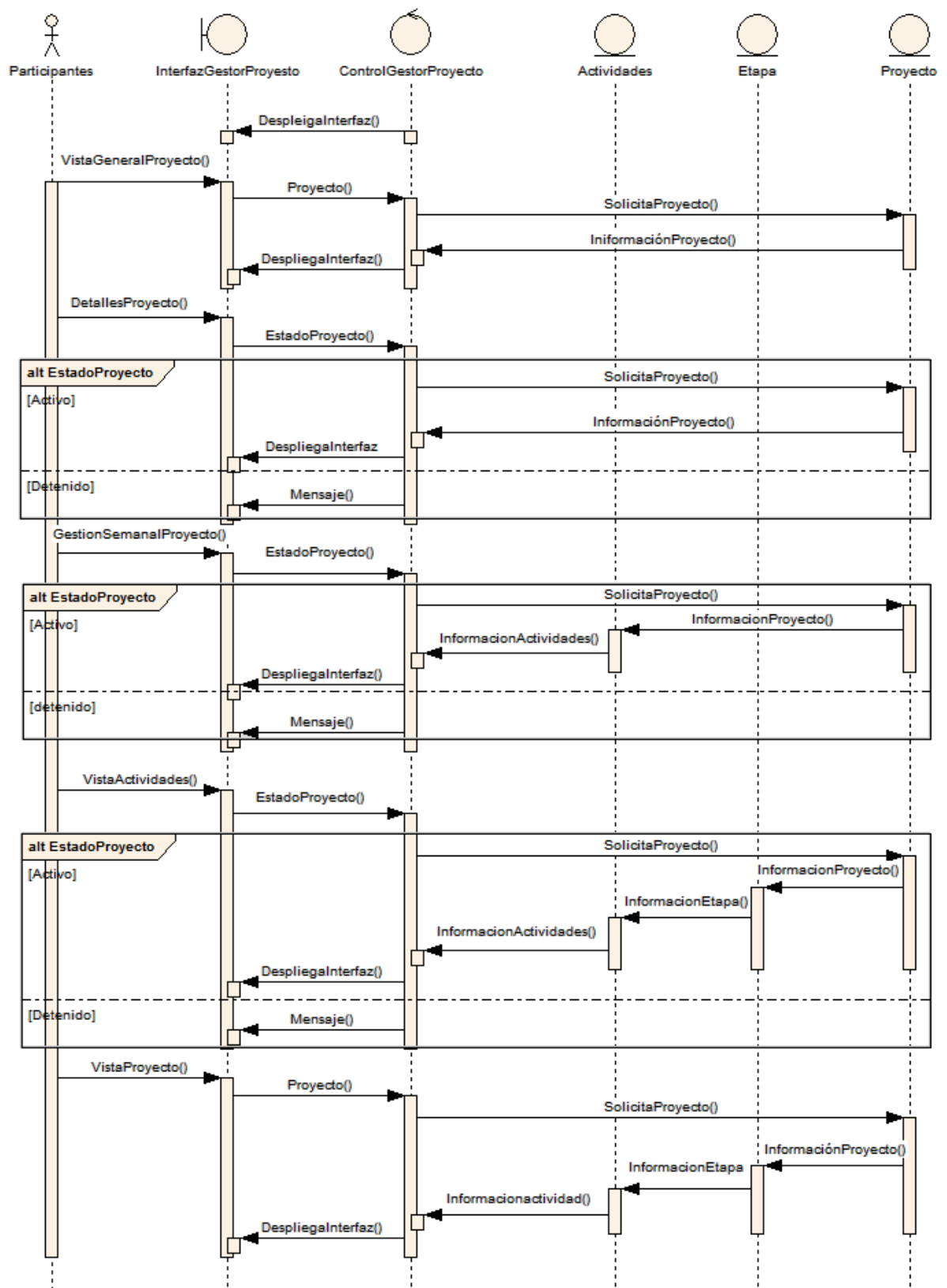


Fuente: Autores.

1. El controlador principal despliega la interfaz de Buscar Proyecto.
2. Los participantes selecciona la opción usuarios, el controlador recibe la solicitud y carga los usuarios que tienen Proyectos.

3. Una vez cargan los usuarios, los participantes escogen el usuario, el controlador lo recibe y busca en base de datos los proyectos asociados al usuario, luego despliega los proyectos a los participantes.
4. En caso que no seleccione usuarios, los participantes pueden seleccionar grupos, el controlador recibe la solicitud y carga los grupos que tienen proyectos.
5. Los participantes escogen el grupo, el controlador lo recibe y busca en base de datos los proyectos asociados al grupo, luego despliega los proyectos a los participantes.
6. Al desplegar la lista de proyectos los participantes pueden acceder a los proyectos, el controlador principal recibe la solicitud y despliega la interfaz de la vista general del proyecto.

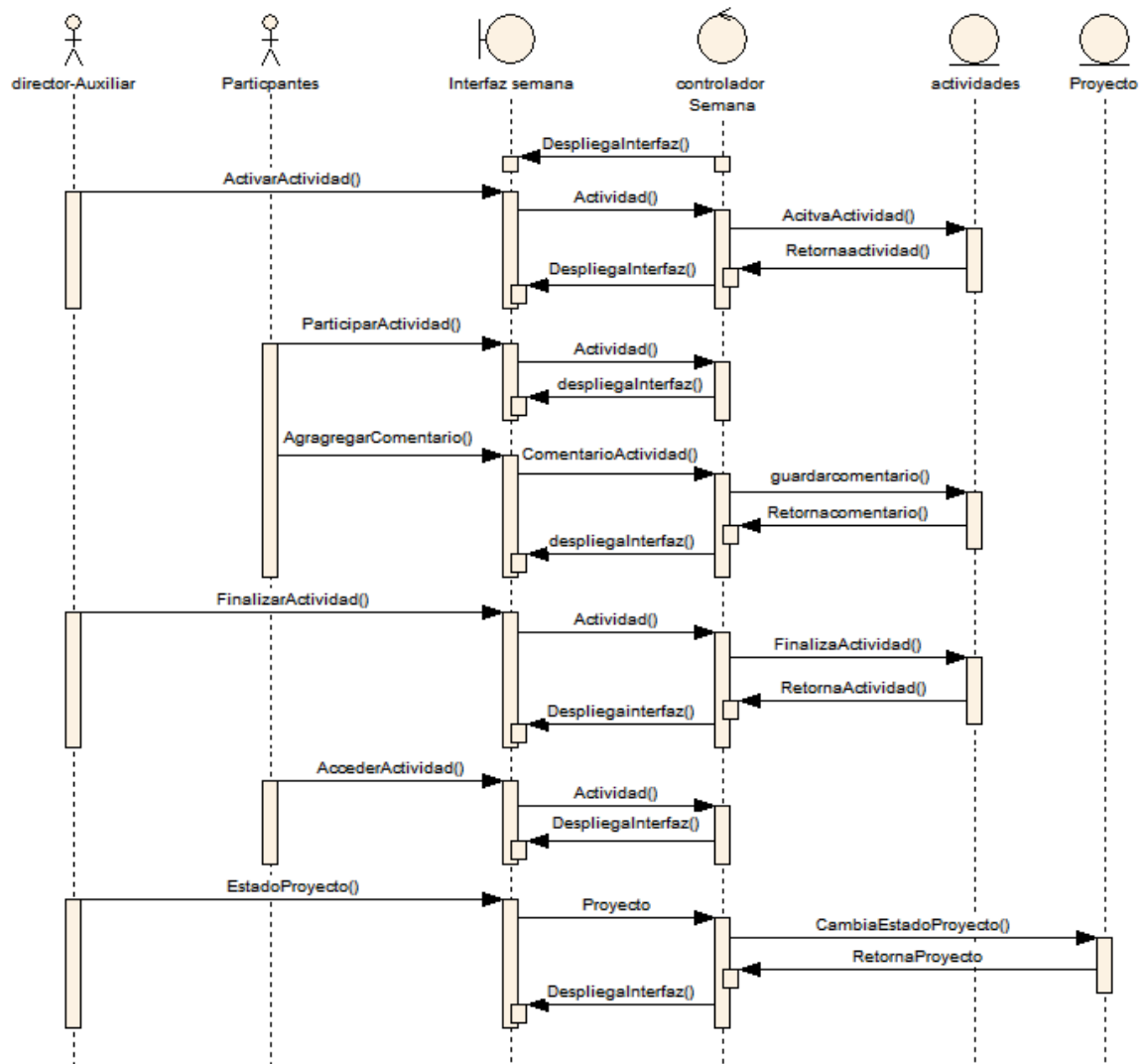
Figura 27. Diagrama de secuencia: Vista General del Proyecto.



Fuente: Autores.

1. Una vez que los participantes entran la gestor de Proyectos y acceden al proyecto, el controlador solicita la información del proyecto y le despliega la información de proyecto en la interfaz.
2. Los participantes pueden acceder a detalles del proyecto, el controlador verifica que el estado del proyecto sea activo y solicita la información, el controlador despliega en la interfaz información del proyecto.
3. Los participantes pueden acceder a gestión semanal, el controlador verifica que el estado del proyecto sea activo y solicita información del proyecto, el proyecto solicita las actividades de la semana y retorna la información al controlador, el controlador despliega la interfaz de Gestión Semanal.
4. Los participantes pueden acceder a la vista de las actividades, el controlador verifica que el estado del proyecto sea activo y solicita información del proyecto, el proyecto solicita las etapas y cada etapa solicita sus actividades, el controlador recibe la información y despliega una lista en la interfaz.
5. Los participantes pueden acceder a la vista del proyecto, el controlador solicita el tiempo estimado del proyecto, proyecto solicita las etapas y las etapa solicita sus actividades, el controlador recibe la información y despliega una esquema de actividades contra tiempo en la interfaz.

Figura 28. Diagrama de secuencia: interface que permite gestionar la semana

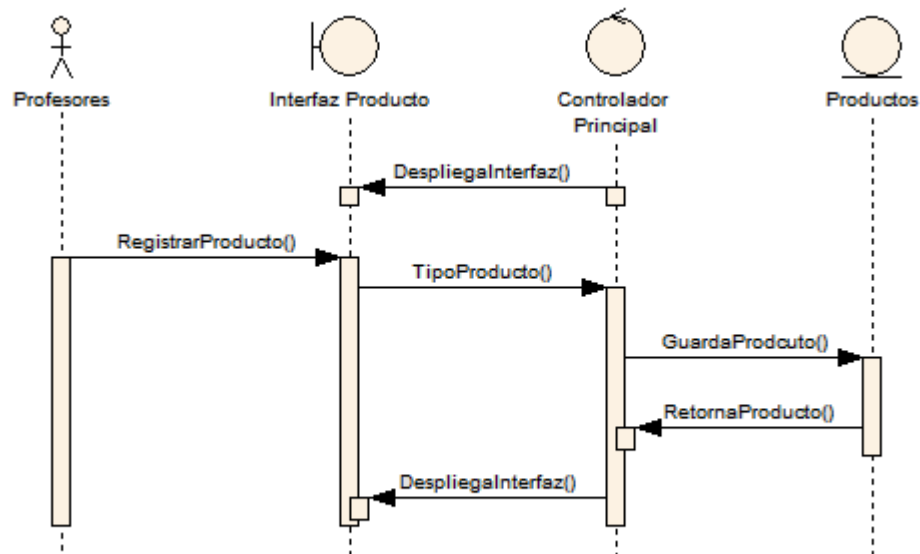


Fuente: Autores.

1. El controlador de la semana despliega la interfaz mostrando una lista de actividades de la semana actual.
2. Una vez se despliega la lista de actividades el director o auxiliar de proyecto pueden activar una actividad, el controlador recibe la actividad y cambia el estado en base de datos a Activo, el controlador despliega la interfaz mostrando el cambio.

3. Una vez la actividad está activa, los participantes pueden participar en la actividad, el controlador recibe la actividad y despliega la interfaz.
4. Cuando los participantes participan en la actividad pueden agregar un comentario, el controlador recibe el comentario y lo asocia a la actividad, luego despliega la interfaz mostrando la lista de comentarios.
5. El director o auxiliar de proyecto pueden Finalizar la actividad, el controlador recibe la actividad y cambia el estado en bases de datos a Finalizado, luego despliega la interfaz mostrando los cambios.
6. Cuando la actividad está finalizada los participantes pueden acceder a la actividad, el controlador recibe la actividad y despliega la interfaz.
7. Cuando el controlador de la semana despliega la interfaz, el director o auxiliar de proyecto puede cambiar el estado del proyecto, el controlador recibe el proyecto y cambia el estado en bases de datos, Proyecto retorna el estado y el controlador despliega la interfaz.

Figura 29. Diagrama de secuencia: Registrar Producto



Fuente: Autores.

1. El controlador Principal despliega la interfaz cuando los profesores acceden a Guardar Producto intelectual.
2. Una vez se despliega la interfaz, los profesores seleccionan el tipo de producto y crean uno nuevo.
3. El controlador recibe el tipo de producto y guarda en bases de datos el producto nuevo.
4. El controlador principal despliega la interfaz.

4.1.5 Implementación, Implantación y Pruebas Generales.

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

- Lenguaje de programación orientado a la web, JSP.
- Hojas de estilo en cascada, CSS.
- Lenguaje Java.
- NetBeans, IDE para desarrollar las clases de Java y los archivos JSP y CSS.
- Servidor Jakarta Tomcat.
- Bloc de notas para desarrollar con HTML.
- Manejador de Base de datos, MySQL 5.0.

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

Para el diseño realizado, se trabajó con la base de datos “Diamante” que ya existía y estaba implantada en el Servidor Cormorán. Según el diseño realizado

para el primer prototipo, se modificaron algunas tablas y se crearon otras; también se trabajó siguiendo la estructura de directorios mencionada anteriormente. Actualmente el tamaño del portal es de 490 MB.

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

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

4.2 MANTENIMIENTO Y ADMINISTRACIÓN

4.2.1 Actividades de Mantenimiento.

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

errores encontrados durante el uso. La función del administrador es mejorar el sistema y corregir las diferentes fallas que se puedan presentar.

- Colaborar a integrantes del grupo Calumet en la parametrización del sitio EISIWeb para que la implantación en otras escuelas se realizara de una manera más fácil.
- Colaborar constantemente en la adaptación de los servicios de EISIWeb al portal web de la Escuela de Estudios Industriales y Empresariales y de las demás Escuelas a las que presta sus servicios el grupo CALUMET.
- Dar solución a los problemas de código fuente que se puedan presentar, impidiendo el funcionamiento de los servicios debido a diferentes factores (parametrización, migración del sitio a versiones nuevas de Tomcat y Java), estos problemas deben ser solucionados inmediatamente luego de ser descubiertos.
- Implantar el sitio en otras escuelas. Modificar JavaBeans para filtrar la información descargada desde la División de Servicios de Información de acuerdo a la escuela.
- Realizar en el portal EISIWeb los enlaces a portales de otras escuelas que actualmente soporta el grupo Calumet.
- Revisar archivos de los portales web y borrar los que ya no eran utilizados.

4.2.2 Actividades de Soporte a Usuarios.

Mantener una comunicación abierta con los usuarios y directivos es de gran importancia para analizar y evaluar las percepciones con respecto al sistema. Esta forma de participación con los usuarios durante el soporte aumenta el grado de

confianza y credibilidad. En esta actividad, los usuarios del sistema informan sobre los problemas en la utilización del mismo y los encargados del soporte responden con: cambios en los procedimientos de operación, información adicional y mejoramiento en el servicio. Algunas de las labores realizadas comúnmente son:

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

4.2.3 Actividades de Administración.

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

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

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

Incidencias

- Correos entre autores de preguntas y sus autorizadores al Banco de Preguntas.

Se revisó el módulo de asignar preguntas a una actividad, en ManejoPreguntasProfesores.jsp se invoca la función enviarcorreo() del bean Enviarcorreos.java, al invocar la función se pasan dos parámetros, el correo del usuario y el asunto del correo, para diferenciar los asuntos de cada pregunta se le agrego el identificador de la pregunta y el nombre del profesor al que se le envía, con esto cada correo que se envía será diferenciado por el identificador de la pregunta.

- Asignación de preguntas del banco a una actividad

Se revisó el módulo de asignar preguntas del banco a un taller, buscando el motivo por que no muestra bien las respuestas donde algunas salen vacías, en VerPreguntas.jsp se encontró que el identificador de una de las cajas de texto donde se muestran las respuestas estaba repetido, lo que generaba un error al momento de construir las respuestas, se procedió a modificar el identificador asignándole uno diferente, así ya no se generan las cajas vacías y se muestran las respuestas correctamente.

- Toma de datos para las estadísticas de respuestas de los estudiantes a las preguntas.

Se verificó en la tabla tr_califtalleres_examenes si la pregunta tenía una nota asignada, en caso negativo se procedió a aumentar en uno el campo NumVecesR y adicionar la nota de la pregunta multiplicada por diez al campo NumVecesA, ambos de la tabla tp_banco_pregunta.

- Agregar el título del archivo subido por un usuario en el correo que se envía a los interesados

Se utilizaron los métodos getNomArchivo() y getDescripcion() del Bean Archivos para obtener el título y la descripción del archivo que sube el usuario. Luego se envía un mensaje de correo electrónico a los interesados con la notificación de la subida del archivo, el título y la descripción.

5 PRUEBAS DEL SISTEMA

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

5.1 PRUEBAS DE VERIFICACIÓN

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

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

5.1.1 Pruebas por componente.

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

Tabla 10. Pruebas Realizadas: Asociar Quices al Aula de Clase.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Acceder al Sistema	Ingresar al portal usando un nombre de usuario y contraseña.	✓
Acceder al Aula	Ingresar al servicio “Acceder al Aula” con perfil de Estudiante, Docente y Administrador y verificar que se listen para cada uno sólo los guiones de clases que pueden visualizar en relación a su perfil.	✓
Acceder Crear Modificar Aula	Ingresar al servicio “Crear Modificar Aula” con el perfil de docente y verificar que se listen las aulas de clase y el botón crear nueva aula.	✓
Crear Aula	Verificar que al crear Aula se liste la opción de Quices, y se le pueda asignar cualquier valor en el porcentaje siempre y cuando el docente asocie quices al Aula nueva.	✓
Modificar Porcentaje	Al ingresar al módulo de “Modificar porcentajes” verificar que si el aula tiene asociados quices, el porcentaje se pueda modificar en este módulo.	✓

Fuente: Autores.

Tabla 11. Pruebas realizadas: Asignar y Modificar Nota en el Aula.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Vista Actividad	Cargar todas las actividades de un plan	✓
Calificar Actividades	Carga todas las actividades pendientes del guion de clase.	✓
Resultados Quices	Carga todos los quices que han finalizado y contestado los estudiantes.	✓
Calificar Actividad	Despliega una interfaz listando los subgrupos de clase y asignándole la nota de la actividad.	✓
Asignar Nota	Despliega una interfaz listando los estudiantes de clase individualizando la nota, permite enviar un correo y agregar un comentario ala nota de cada estudiante.	✓
Modificar Nota	Despliega una interfaz listando los estudiantes de clase individualizando la nota, permite enviar un correo y agregar un comentario ala nota de cada estudiante, los profesores pueden modificar la nota y comentario de la nota a cada estudiante.	✓

Fuente: Autores.

Tabla 12. Pruebas realizadas Crear y Gestionar Proyecto.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Acceder Crear Gestionar Proyecto	Acceder al servicio “Crear Gestionar Proyecto” y listar los proyectos asociados al usuario y mostrar el botón crear nuevo proyecto.	✓
Crear Proyecto	Oprimir el botón crear nuevo proyecto, verificar que no envíe campos inválidos.	✓
Editar Proyecto	Modificar un Proyecto, agregar o eliminar objetivos, profesores y grupos.	✓
Eliminar Proyecto	Eliminar un Proyecto y verificar en base de datos que todo quedara borrado.	✓
Inicia Proyecto	Iniciar un proyecto y verificar que el estado del proyecto sea activo.	✓

Fuente: Autores.

Tabla 13. Pruebas realizadas: Gestionar Etapas Actividades.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Seleccionar Proyecto	Entrar como participante y mirar que no permita editar el proyecto, verificar que se listen todos los proyectos asociados al usuario, ya sea como director o auxiliar de proyecto.	✓
Crear Etapa	Oprimir el botón crear etapa, verificar que no se envíen campos inválidos, al guardar la etapa muestre el botón crear actividad.	✓
Editar Etapa	Editar una etapa agregándole o eliminando actividades	✓
Eliminar Etapa	Eliminar una etapa y revisar en bases de datos que los datos estén borrados.	✓
Crear Actividad	Oprimir el botón crear actividad, verificar que no se envíen campos inválidos.	✓
Editar Actividad	Editar una actividad, agregando o eliminando requisitos, modificando la semana inicial y final.	✓
Eliminar Actividad	Eliminar una actividad y verificar que en bases de datos que todos los datos estén borrados.	✓

Fuente: Autores.

Tabla 14. Pruebas realizadas: Buscar Proyecto.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Seleccionar Usuario	Oprimir el botón usuario y verificar que se listen solo usuarios asociados a un proyecto	✓
Seleccionar Grupo	Oprimir el botón Grupo y verificar que se listen solo los grupos que tienen un proyecto asociado.	✓
Buscar	Verificar que se listen los proyectos asociado a ese usuario o grupo, corroborar la información que se muestra.	✓

Fuente: Autores.

Tabla 15. Pruebas realizadas: Vista General Proyecto.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Vista General	Verificar la información que se muestra con la que esta almacenada en bases de datos, corroborar la fechas con la fecha en el sistema.	✓
Detalles	Verificar la información que se muestra con la que esta almacenada en bases de datos.	✓

Vista Proyecto	Verificar la información que se muestra con la que esta almacenada en bases de datos, oprimir el botón adelante verificar que muestre las actividades para las siguientes semanas si las tiene.	✓
Vista Actividades	Acceder con un usuario que no esté asociado al proyecto y comprobar que no le salga vista actividades en el menú, Acceder como participante y verificar la información que se muestra con la que esta almacenada en bases de datos.	✓
Vista Semanal	Acceder con un usuario que no esté asociado al proyecto y comprobar que no le salga vista actividades en el menú, Acceder como participante y verificar la información que se muestra con la que esta almacenada en bases de datos.	✓

Fuente: Autores.

Tabla 16. Pruebas realizadas Gestionar Semana.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Manejar Semana	Verificar que se listen solo las actividades de la semana, que se pueda ir a una semana en el futuro o en el pasado.	✓
Estado Proyecto	Oprimir el botón activar o detener, verificar que el estado del proyecto cambia a detenido o activo, ingresar una fecha y una semana manualmente en la activación.	✓
Activar	Oprimir el botón activar, y verificar que se cambió el estado de la actividad a activo y que se muestran los botones participar y finalizar, verificar que se envíe correos.	✓
Participar	Verificar que se listen solo los comentarios de esa actividad, agregar un comentario nuevo verificar la fecha y hora que se muestra, corroborar la información que se muestra, verificar que se envíe correos.	✓
Finalizar	Oprimir el botón finalizar, verificar que se cambió el estado a finalizado, que muestre el botón acceder, verificar que se envíe correos.	✓

Fuente: Autores.

Tabla 17. Pruebas realizadas: Registrar Producto.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Registrar Producto	Ingresar como profesor y crear un producto nuevo, verificar que no se envíen campos inválidos.	✓

Fuente: Autores.

5.2 PRUEBAS DE INTEGRACIÓN

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

5.3 PRUEBAS DE VALIDACIÓN

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

6 CONCLUSIONES

- La realización de la nueva interfaz para asignar nota permitió mejorar la experiencia de los profesores al momento de calificar cada estudiante, le dio la posibilidad de tomar decisiones individuales, asignando mayor o menor nota a un estudiante de un subgrupo según sus méritos de clase.
- El gestor de proyectos nace como un servicio nuevo que contribuye en la gestión y resolución de proyectos de extensión, se constituye en una herramienta fundamental en la que el director de un proyecto puede controlar el tiempo de las actividades, asignar responsables, y llevar un control de una forma más dinámica.
- La mejora en la actualización de la base de datos Diamante permitirá consultar de manera efectiva los diferentes programas que este cursando un estudiante. Esto es especialmente evidente en las escuelas que tienen más de un programa de pregrado o posgrado.
- Las mejoras en las actualizaciones de horarios de materias y profesores permitirán completar la información de usuario al que tiene acceso el estudiante. Anteriormente solo podía acceder a los horarios de las materias de la escuela.

7 RECOMENDACIONES

- Crear un módulo que muestre los productos registrados por el profesor además de eso una interfaz que permita modificar la información registrada en ellos.
- Crear un servicio en el gestor de proyectos que permita analizar y comparar las fechas reales con las fechas esperadas, así los participantes pueden analizar posibles fallos en el transcurso del proyecto.
- Utilizar procedimientos almacenados para las actualizaciones de las bases de datos Diamantes y División, debido a sus ventajas en tiempo de ejecución y seguridad.
- Implementar operaciones Rollback para las bases de datos, para que en caso de algún problema en una transacción, la base de datos se pueda revertir a un estado consistente.
- Ofrecer los servicios del portal a otras escuelas de la universidad e integrarlo a las redes sociales.
- Desarrollar una versión para equipos móviles del portal, optimizada para estos dispositivos.

BIBLIOGRAFÍA

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

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

ANEXOS

El manual de usuario el cual es un manual guía para que el usuario pueda acceder a los diferentes servicios creados, donde se explica su manejo, paso a paso. Se puede encontrar en el menú de Ayudas en todos los portales del grupo, por ejemplo, en el nuevo portal del decano al cual se puede acceder desde el siguiente enlace:

<http://cormoran.uis.edu.co/eisi/Ayudas/index.jsp?IdServicio=S661>

Así como en el canal del grupo en youtube:

<http://www.youtube.com/feed/UClIYsDiHVug2FfZHICEEzEg/u>

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