

**ANÁLISIS, MANTENIMIENTO, DISEÑO, DESARROLLO E
IMPLEMENTACIÓN DE LOS SERVICIOS DE ENCUESTAS,
INSCRIPCIONES, GESTOR DE PROYECTOS Y NOTICIAS PARA EL
PORTAL WEB**

**CLAUDIA PATRICIA PATIÑO ROJAS
CLAUDIA LILIANA REY SANTOS**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2016**

**ANÁLISIS, MANTENIMIENTO, DISEÑO, DESARROLLO E
IMPLEMENTACIÓN DE LOS SERVICIOS DE ENCUESTAS,
INSCRIPCIONES, GESTOR DE PROYECTOS Y NOTICIAS PARA EL
PORTAL WEB**

**CLAUDIA PATRICIA PATIÑO ROJAS
CLAUDIA LILIANA REY SANTOS**

Trabajo de grado para optar al título de Ingeniero de Sistemas

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

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2016**

AGRADECIMIENTOS

Agradezco en primera instancia a Dios por la bendición de culminar esta etapa, por cada cosa que he aprendido en el transcurso de estos años, por las personas que conocí en este proceso y por la fortaleza que me ha dado en las dificultades.

A mi abuela María del pilar y a mis tíos, Jorge Maldonado y Luis Antonio Rojas, por su apoyo y compañía, porque sin su ayuda esto no hubiera sido posible, gracias a su continua motivación hoy alcanzo este logro tan anhelado. Con ellos también a mi madre, mi padre, hermanos y demás familiares que sin duda alguna también me brindaron frases de aliento y buenos consejos.

A mi pareja, Alexis Cuevas por su compañía, apoyo y motivación en este tiempo, porque en los últimos momentos, que fueron de los más difíciles, siempre me ayudó a ver una razón para seguir adelante con más energía y dedicación.

A mi compañera de proyecto Claudia Liliana Rey por su colaboración, por culminar conmigo este trabajo de grado, y porque más que una compañera de trabajo es una amiga incondicional, este logro también es resultado del apoyo mutuo.

Al profesor Luis Ignacio por permitirme ser parte del grupo CALUMET y por cada una de sus enseñanzas en el ámbito académico y personal.

A Duvan Jamid Vargas por su paciencia, colaboración y amistad; y con él también a todo el grupo CALUMET, a todos los compañeros que hacen que el espacio de trabajo, no sea solo eso, sino también un lugar para compartir gratos momentos.

A todos mis amigos por su compañía en las largas jornadas de estudio, los momentos de risas, y cada instante que compartimos, que sin duda hicieron que el camino para llegar a este punto fuera más agradable.

Claudia Patricia Patiño Rojas

AGRADECIMIENTOS

Primeramente agradezco a Dios por darme la oportunidad de vivir esta maravillosa experiencia y hoy estar aquí a punto de culminar una de mis metas, por darme la fortaleza cada vez que asumo un reto nuevo y no dejarme desfallecer.

A mi madre quien con su comprensión y amor se ha convertido también en mi amiga, quien se esfuerza sin importar su cansancio y dolencias para que yo solo me dedique a sacar esta meta adelante y todo se me haga más fácil.

A mis tías por ser mi apoyo incondicional en todo lo que emprendo, por brindarme su compañía y atenciones en todas esas noches que pase estudiando.

A mi esposo por su comprensión y apoyo para que mis sueños se realicen, por esas palabras de aliento que me brinda cada vez que desfallezco o me preocupo.

Al profesor Luis Ignacio González por permitirme ingresar al grupo y pertenecer a esa hermosa familia que es CALUMET, por sus palabras de motivación para que nunca dejemos de pedalear por nuestros sueños por sentarse a compartir con los integrantes del grupo y enseñarnos que el trabajo es más fácil en comunidad.

A los integrantes del grupo Calumet quienes con su conocimiento y forma de ser me motivaban cada vez que las cosas se veían difíciles, al Ingeniero Duvan Castillo por compartir sus conocimientos, por el tiempo dedicado y paciencia que con sus ideas hizo que los días no se tornaran monótonos.

A mis amigas Patricia Rojas y María Eugenia Suarez quienes fueron un apoyo a lo largo de la carrera con quienes pasé muchos ratos maravillosos.

A todos mis amigos que me apoyaron de una u otra forma para culminar esta etapa y sin su presencia no hubiese tenido el mismo sentido.

Claudia Liliana Rey Santos

CONTENIDO

INTRODUCCIÓN	16
1. PLANTEAMIENTO DEL PROYECTO	17
1.1 DEFINICIÓN DE LA SITUACIÓN ACTUAL.....	17
1.2 JUSTIFICACIÓN	17
1.3 OBJETIVOS.....	19
1.3.1 OBJETIVO GENERAL	19
1.3.2 OBJETIVOS ESPECÍFICOS.....	19
1.4 IMPACTO Y VIABILIDAD	22
1.4.1 Impacto	22
1.4.2 Viabilidad	22
2. MARCO TEÓRICO	24
2.1 ARQUITECTURA CLIENTE/SERVIDOR.....	24
2.1.1 Características de la arquitectura Cliente/Servidor	25
2.1.2 Clasificación de las arquitecturas Cliente/Servidor	25
2.1.3 Arquitectura Cliente/Servidor aplicada	27
2.1.4 Ventajas del esquema Cliente/Servidor	27
2.1.5 Desventajas del esquema Cliente/Servidor	28
2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS	28
2.2.1 Código del Lado del Cliente (Client Side Scripts)	28
2.2.2 Código del Lado del Servidor (Server Side Scripts).....	29
2.2.3 Tecnología aplicada	30
2.3 BASES DE DATOS	31
2.3.1 Modelos de Bases de Datos	32
2.3.2 Manejadores o Gestores de Bases de Datos	33
2.3.3 MySQL	34
2.3.4 Ventajas de MySQL	34
2.4 NETBEANS	35
2.5 SISTEMA DE CONTROL DE VERSIONES.....	35
2.5.1 Subversión	36
2.6 PROGRAMACIÓN UTILIZADA.....	37
2.6.1 Clases	37
2.6.2 Objetos	37
2.6.3 Atributos	38
2.6.4 Métodos.....	38
2.6.5 Herencia	38
2.6.6 Beneficios de la Programación Orientada a Objetos	38
2.6.7 Java y JDK (Java Development Kit)	39
2.7 SERVIDORES WEB.....	39

2.7.1 Servidor Jakarta Tomcat.....	40
3. MARCO METODOLÓGICO	41
3.1 PROTOTIPO EVOLUTIVO	41
3.2 LENGUAJE DE MODELADO UNIFICADO	43
3.2.1 Diagramas de UML.....	43
3.2.2 Diagramas de casos de uso.....	44
3.2.3 Diagramas de secuencias	45
3.3 ESTÁNDARES DE PROGRAMACIÓN.....	47
3.3.1 Modelo de datos.....	47
3.3.2 Nombres de las tablas	47
3.3.3 Clases	48
3.3.4 Páginas JSP	49
3.3.5 Organización de Directorios	49
4. DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO.....	50
4.1 PROTOTIPO ESPERADO.....	50
4.1.1 DIAGRAMAS DE CASOS DE USO	58
4.1.2 Documentación de Casos de Uso del Sistema	63
4.1.3 Diseño y Análisis	70
4.1.4 Modelo de Procesos del Sistema	77
4.1.5 Implementación, Implantación y Pruebas Generales	101
4.2 MANTENIMIENTO Y ADMINISTRACIÓN	102
4.2.1 Actividades de Mantenimiento.....	102
4.2.2 Actividades de Soporte a Usuarios	105
4.2.3 Actividades de Administración	105
5 PRUEBAS DEL SISTEMA.....	107
5.1 PRUEBAS DE VERIFICACIÓN.....	107
5.1.1 Pruebas por componente	107
6. CONCLUSIONES.....	112
7. RECOMENDACIONES.....	114
BIBLIOGRAFÍA	115

Lista de tablas

Tabla 1. Casos de uso: Proyectos de extensión	64
Tabla 2. Casos de uso: Ejes de gestión	64
Tabla 3. Casos de uso: Noticias	65
Tabla 4. Casos de uso: Encuestas	66
Tabla 5. Casos de uso: Inscripciones.....	68
Tabla 6. Casos de uso: Subsistema de Mantenimiento y Administración.....	69
Tabla 7. Descripción de las Entidades	77
Tabla 8. Pruebas Realizadas: Inscripciones	109
Tabla 9. Pruebas Realizadas: Encuestas.....	110
Tabla 10. Pruebas Realizadas: Noticias.....	110
Tabla 11. Pruebas Realizadas: Proyectos de extensión.....	111
Tabla 12. Pruebas Realizadas: Ejes de gestión.....	111

Lista de figuras

Figura 1. Modelo Cliente/Servidor	24
Figura 2. Modelo de acceso a JSP.....	31
Figura 3. Prototipo Evolutivo	41
Figura 4. Diagramas de Casos de Uso	44
Figura 5. Diagrama de Secuencias	47
Figura 6. Diagrama de Casos de Uso: Proyectos de Extensión - Director y Codirector	58
Figura 7. Diagrama de Casos de Uso: Proyectos de Extensión - Auxiliar	59
Figura 8. Diagrama de Casos de Uso: Proyectos de Extensión - Profesor y Responsable	59
Figura 9. Diagrama de Casos de Uso: Ejes de gestión	60
Figura 10. Diagrama de Casos de Uso: Encuestas	60
Figura 11. Diagrama de Casos de Uso: Inscripciones.....	61
Figura 12. Diagrama de Casos de Uso: Noticias	61
Figura 13. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.....	62
Figura 14. Diagrama E/R: Encuestas	70
Figura 15. Diagrama E/R: Inscripciones.....	71
Figura 16. Diagrama E/R: Proyectos de Extensión.....	72
Figura 17. Diagrama E/R: Noticias Grupo	73
Figura 18. Diagrama E/R: Noticias Comunidad UIS	74
Figura 19. Diagrama de secuencia: Gestionar Noticias-Crear Noticia	77
Figura 20. Diagrama de secuencia: Gestionar Noticias-Editar Noticia.....	78
Figura 21. Diagrama de secuencia: Ver Titulares-Calificar Noticia	79
Figura 22. Diagrama de secuencia: Gestionar Proyecto-Crear Proyecto	80
Figura 23. Diagrama de secuencia: Gestionar Proyecto-Editar Proyecto	81
Figura 24. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Finalizar Actividad	83
Figura 25. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Reactivar Actividad.....	84
Figura 26. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Participar Actividad.....	85
Figura 27. Diagrama de secuencia: Gestionar Proyecto-Gestionar Etapas-Crear Etapa	86
Figura 28. Diagrama de secuencia: Gestionar Proyecto-Gestionar Etapas-Crear Actividad.....	87
Figura 29. Diagrama de secuencia: Gestionar Encuestas-Crear Encuesta.....	89
Figura 30. Diagrama de secuencia: Gestionar Encuestas-Editar Encuesta.....	90
Figura 31. Diagrama de secuencia: Gestionar Encuestas-Ver Resultados Encuesta..	91
Figura 32. Diagrama de secuencia: Gestionar Encuestas-Duplicar encuesta.....	92
Figura 33. Diagrama de secuencia: Gestionar Encuestas-Responder Encuesta	93

Figura 34. Diagrama de secuencia: Gestionar Inscripción-Crear Inscripción	94
Figura 35. Diagrama de secuencia: Gestionar Inscripción-Editar Inscripción.....	95
Figura 36. Diagrama de secuencia: Gestionar Inscripción-Duplicar Inscripción	96
Figura 37. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Estado	97
Figura 38. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Enviar Correo.....	98
Figura 39. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Información Interesado	99
Figura 40. Diagrama de secuencia: Ver Inscripciones - ParticiparInscripcion	100

Resumen

TÍTULO: Análisis, mantenimiento, diseño, desarrollo e implementación de los servicios de encuestas, inscripciones, gestor de proyectos y noticias para el portal web*.

AUTORES:** Claudia Patricia Patiño Rojas
Claudia Liliana Rey Santos

PALABRAS CLAVE: Portal Web, Inscripción, Gestor de proyectos, Encuestas, Noticias

Descripción

En la actualidad Facultades y Escuelas de la Universidad Industrial de Santander cuentan con el Portal Web Comunidad Académica que permite el encuentro e interacción de los diferentes miembros de estas comunidades. Este Portal soporta diferentes actividades académicas y provee servicios que requieren una continua mejora y labores de mantenimiento.

Se cuenta con un servicio de noticias donde publican sólo usuarios con permiso de administrador, se hace necesario que todos los usuarios puedan publicar y que desde los grupos puedan publicar noticias a su portal grupo, a su escuela y a otras escuelas; el servicio contará con una reingeniería y nuevo diseño.

El servicio Inscripciones va dirigido sólo a comunidad escuela por lo que se hace necesario que este servicio soporte a la Comunidad UIS, lo que permitiría crear una inscripción y dirigirla a varias escuelas.

En los módulos que ofrecen los portales se encuentra un Gestor de Proyectos que permite crear proyectos con actividades agrupadas en etapas, actualmente el cronograma de actividades está planeado por semanas, se requiere que las actividades se puedan registrar por fechas. Además agregar la opción de reactivar las actividades una vez finalizadas ya que el usuario actualmente no cuenta con esta opción.

Se brinda a la comunidad el servicio de crear encuestas donde participan usuarios de las diferentes escuelas, se requiere cambiar el diseño y usabilidad del servicio y que cuente con una actualización en la vista de resultados con el fin de mejorar la experiencia de usuario.

* Trabajo de grado Modalidad Trabajo de Investigación

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

ABSTRACT

TITLE: Analysis, maintenance, design, development and implementation of the services survey, registration, project manager and news for the web portal*.

AUTHORS:** Claudia Patricia Rojas Patiño
Claudia Liliana Rey Santos

KEYWORDS: Web Portal, Registration, Project Management, Surveys, News.

Description

Currently Faculties and Schools of the Universidad Industrial de Santander have the Academic Community Web Portal that allows the meeting and interaction of the different members of these communities. This Portal supports different academic activities and provides services that require continuous improvement and maintenance.

It has a news service where users post only with administrator permission, it is necessary that all users can publish and that from their groups they can publish news on their group portal, their school and other schools; the service will feature a new design and reengineering.

The registration service is aimed only to the school community so it's necessary that this service supports the UIS Community, which would allow to create a registration and direct it to different schools.

In the modules offered by portals there is a project manager for creating projects with activities grouped in stages, currently the schedule of the activities is planned for weeks, it is required that these activities can be registered by date. Also adding the option of reactivate the activities after they are completed because the user does not currently have this option.

It's provided to the community a service where users create surveys and different schools participate, it's required to change the design and usability of the service and update the result's view with the purpose of improving the user experience.

* Degree Work: Research Paper.

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

Director: M.Sc. Luis Ignacio González Ramírez

INTRODUCCIÓN

Los Portales Web Comunidad Académica son los principales canales de comunicación e integración entre los miembros de la comunidad de las escuelas y miembros en general de la comunidad UIS. Actualmente los servicios se han extendido a más escuelas de las distintas facultades gracias a la aceptación de los usuarios por lo que se adopta el objetivo de mantener y mejorar los servicios que se ofrecen, además de crear nuevos servicios que satisfagan las necesidades crecientes de los usuarios de la comunidad académica.

El grupo CALUMET, grupo de desarrollo de software de la escuela de Ingeniería de Sistemas, se encarga de desarrollar los nuevos servicios y darle mantenimiento a los existentes de manera que su actualización responda al continuo cambio. Con el fin de llevar a su cumplimiento el objetivo principal del grupo y proporcionar portales web con contenido dinámico, se cuenta con herramientas software de libre distribución como Netbeans, Github, Sqlyog, entre otras.

Este documento presenta un soporte teórico, metodológico y técnico del desarrollo del módulo de inscripciones comunidad UIS con el fin de extender el servicio de inscripción de usuarios a las distintas escuelas; el módulo de encuestas para la mejora de la presentación de los resultados y la usabilidad de todo el servicio; la presentación de las noticias y reingeniería a su desarrollo, y la usabilidad del módulo de proyectos de extensión además de agregar la funcionalidad de asociar proyectos a un eje de gestión.

1. PLANTEAMIENTO DEL PROYECTO

1.1 DEFINICIÓN DE LA SITUACIÓN ACTUAL

En la actualidad la Facultad de Ingenierías Físico Mecánicas, la Facultad de Ingenierías Físico Químicas y escuelas de estas y otras facultades de la Universidad Industrial de Santander cuentan con el Portal Web Comunidad Académica que permite el encuentro e interacción de los diferentes miembros de estas comunidades. Este Portal soporta las diferentes actividades académicas que se realizan dentro de las escuelas y las Facultades, así como define a los usuarios los permisos y servicios que se les proporcionan.

Los servicios del Portal Web Comunidad Académica de las escuelas deben mejorar constantemente y adaptarse a los cambios que se presenten en su entorno, a su vez deben 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, actualización y reingeniería se hacen indispensables.

1.2 JUSTIFICACIÓN

El Portal Web Comunidad Académica, en línea con su objetivo de aportar lugares para la comunidad, permite al usuario solicitar la publicación de noticias dirigidas a su escuela, o a cualquier otra dentro del sistema. Se hace necesario que todo usuario pueda publicar y pueda elegir en qué escuelas se realiza dicha publicación dando lugar a la interacción entre los estudiantes de las diferentes escuelas para que se informen; además que permita dar una puntuación a las diferentes noticias para lo que es preciso agregar un ranking. Además, se hace necesario dar mayor relevancia a las noticias de los decanatos en sus escuelas, así como abrir este servicio a los grupos.

Actualmente se cuenta con el servicio Inscripciones comunidad escuela que permite a los profesores y administrativos crear inscripciones a las que pueden aplicar los usuarios interesados de cada escuela. Se hace necesario que este servicio soporte a la Comunidad UIS, lo que permitiría que todo miembro con perfil de administrador, de la comunidad, pueda crear una inscripción y dirigirla a diversas escuelas. Además, es necesario agregar estadísticas y gráficos de los resultados de dichas inscripciones.

En las herramientas que ofrecen los portales se encuentra un Gestor de Proyectos que permite la creación de proyectos que se dividen en etapas, cada etapa con sus respectivas actividades, actualmente el cronograma de actividades se ejecuta o está planeado por semanas, se requiere la opción de modificar las fechas de dichas actividades de modo que se puedan registrar por días y no por semanas completas.

La gestión de proyectos está controlada por un encargado que lo ejecuta y un director y/o auxiliar que se ocupa de la revisión. Los responsables se pueden comunicar a través de comentarios que se pueden utilizar mientras la actividad se encuentra activa, actualmente una actividad finalizada no se puede reactivar, se debe agregar esta opción para controlar posibles eventualidades. Hay una vista de los proyectos y una de las actividades, donde se encuentran clasificados por estados (finalizados, inactivos, etc.) según el cronograma; se debe mejorar el diseño de éstas vistas agregando usabilidad al servicio.

Se desea agregar alertas relacionadas con los cambios y control de las actividades, envío de correos programados y un campo descripción para las actividades. Los proyectos tienen objetivos, se busca asociar los proyectos a un nivel superior de áreas de gestión; cada proyecto puede estar asociado a una o varias de éstas.

El servicio de Encuestas brinda a la comunidad la opción de crear encuestas donde participan usuarios de las diferentes escuelas, actualmente los resultados de una encuesta pueden ser vistos por los usuarios de acuerdo a los permisos

que asigne el administrador de la encuesta, entre las posibilidades está ver sólo los resultados, ver los votantes o ver la información completa, se requiere cambiar la vista de estos resultados con el fin de mejorar la experiencia de usuario; además es necesario que los gráficos que se muestran actualmente tengan una mejor presentación e información más precisa.

1.3 OBJETIVOS

1.3.1 OBJETIVO GENERAL

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

1.3.2 OBJETIVOS ESPECÍFICOS

Efectuar labores de administración de los portales web de las escuelas de la Universidad Industrial de Santander tales como

- Generar diariamente *backups* (copias de respaldo) de la base de datos.
- Actualizar semanalmente 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 realicen los usuarios con el fin de mejorar y/u ofrecer nuevos servicios en la próxima versión de los portales WEB.

- Realizar una revisión constante de los archivos que se encuentran en la base de datos del sitio para evitar que los portales se saturen de información innecesaria y/o desactualizada.

Llevar a cabo labores de mantenimiento a los portales de las Escuelas y la Facultad de Ingenierías Físico-Mecánicas, dentro de éstas podemos encontrar:

- Hacer seguimiento del funcionamiento de los portales para corregir posibles errores en el código fuente.
- Revisar y depurar la estructura de directorios y archivos de los portales Web.

Análisis, diseño, desarrollo e implementación de nuevos servicios o reingenierías de los siguientes módulos:

- Cambiar el diseño, vista y presentación de gráficos de los resultados en el servicio Encuestas. Se debe presentar una vista de los resultados de las encuestas en pestañas de acuerdo a los permisos de los usuarios, mostrando las estadísticas con sus respectivos gráficos, mejorando la precisión de los resultados y así mismo la interpretación de los datos.
- Reingeniería del servicio noticias comunidad escuela y noticias comunidad UIS. Cambiar el diseño de publicaciones y edición de noticias en el servicio Publicar Cartelera. Se debe dar mayor relevancia a las noticias de los decanatos en sus escuelas. Crear el ranking para destacar las noticias más relevantes y una vista en pestañas de las noticias de acuerdo a su debida clasificación. Crear éste mismo servicio para los portales grupo de las diferentes escuelas.
- Modificar el servicio Gestionar mis proyectos con el fin de mejorar la usabilidad del servicio. Cambiar el diseño de la vista de los proyectos,

agregar la opción de reactivar una actividad u etapa finalizada, enviar correos programados, incluir el campo descripción del proyecto y pasar a programar actividades por días. Además se busca asociar los proyectos a una debida área de gestión.

- Es necesario crear el servicio inscripciones comunidad UIS, que permite crear inscripciones donde participen estudiantes de todas las escuelas, se deben agregar estadísticas y gráficos al igual que el servicio Encuestas tanto para éste servicio como para inscripciones comunidad escuela con el fin de mejorar la experiencia de usuario.

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

- Capacitar usuarios y estudiantes del primer nivel en el uso de servicios dentro de los portales de las escuelas, promoviendo así su 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 o la facultad y que por algún motivo necesitan registrarse en el sitio.
- Modificar los estados de los usuarios de acuerdo a la relación con la escuela o la facultad (activo, inactivo, suspendido).

Capacitar a los estudiantes que relevaran las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro de los portales de las escuelas, en cuanto a:

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

1.4 IMPACTO Y VIABILIDAD

1.4.1 Impacto

Los portales web de las Escuelas y la Facultad de Ingenierías Físico mecánicas han sido herramientas útiles para el manejo de la información, por lo tanto es necesario realizar labores de administración, mantenimiento y soporte para ofrecer al usuario un sitio confiable.

Se pretende que los procesos que se realizan en las Escuelas y en la Facultad cada día sean más ágiles, dinámicos, seguros y eficientes, permitiendo una mejor organización de la información, razón por la cual se crean nuevos servicios que se ajusten a las necesidades que puedan surgir.

1.4.2 Viabilidad

La administración del sitio, soporte a usuarios, mantenimiento y desarrollo de nuevos servicios es completamente viable pues se usará software de libre distribución, recurso humano preparado para tal fin, servidores que marchan de

manera legal en la escuela y la facultad, equipos disponibles y todo el soporte tecnológico necesario para el desarrollo del mismo.

Además se cuenta con la supervisión por parte del director del proyecto y la colaboración del equipo de trabajo CALUMET, agentes de gran apoyo en la realización de este proyecto.

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, el servidor envía uno o varios mensajes con la respuesta (ver figura 1). 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

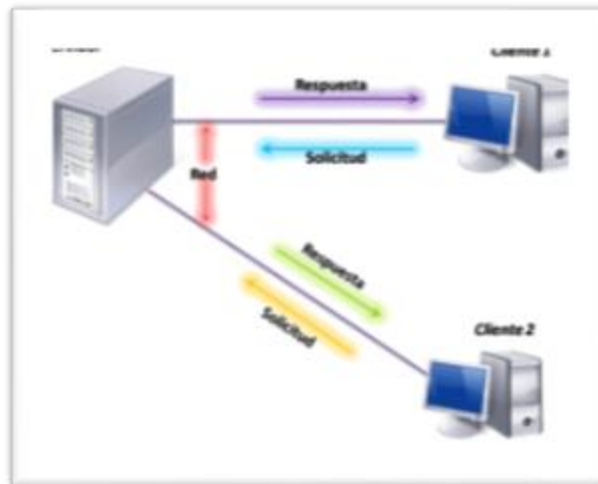


Figura 1. Modelo Cliente/Servidor

Fuente: Internet. <https://eddieramos.wordpress.com/category/topicos-avanzados/page/3/>

2.1.1 Características de la arquitectura Cliente/Servidor

Las características básicas de una arquitectura Cliente / Servidor son:

- Es quien inicia solicitudes o peticiones, tienen por tanto un papel activo en la comunicación.
- El proceso del cliente da la interface entre usuarios y el resto del sistema, 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 para la clasificación:

- Presentación: Presentación de resultados al usuario de forma comprensible.

- Lógica de aplicación: Esta capa es la responsable del procesamiento de la información que tiene lugar en la aplicación.
- Base de datos: Está compuesta por los archivos que contienen los datos persistentes de la aplicación.

La siguiente es la clasificación de la arquitectura 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 requiere poco procesamiento de datos en la organización.
- Cuando se tiene una base de datos centralizada en un solo servidor.
- Cuando la base de datos es relativamente estática.
- Cuando se requiere un mantenimiento mínimo.

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.

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 máquinas existentes puedan ser utilizadas con interfaces amigables al usuario, de esta forma integrar los computadores con sistemas medianos y grandes, sin necesidad de que todos tengan que utilizar el mismo sistema operacional.
- Facilita a los diferentes departamentos de una organización soluciones locales, permitiendo la integración de la información principal totalmente.

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

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. Da 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 fue JSP, por lo tanto los nuevos servicios son desarrollados con esta misma tecnología, ya que permite producir aplicaciones independientes de la plataforma y portables a otros sistemas operativos y servidores web.

Las 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

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

Figura 2. Modelo de acceso a JSP

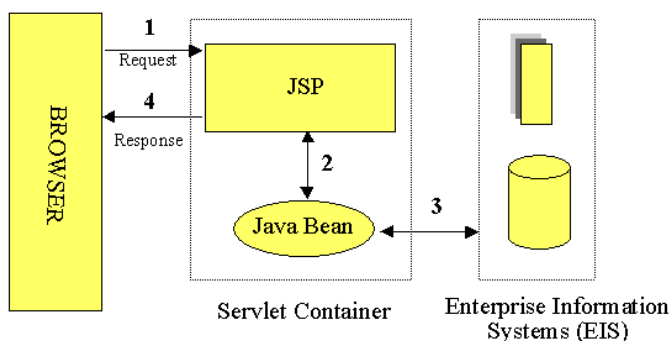


Figura 2. Modelo de acceso a JSP

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

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.

Existen modelos que describen la estructura de una base de datos (entidades, atributos y relaciones), la mayoría de los modelos de datos poseen un conjunto de operaciones básicas como consultar y actualizar y eliminar.

2.3.1 Modelos de Bases de Datos

Las bases de datos se clasifican 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 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.

2.3.3 MySQL

Es un sistema de base de datos operacional considerado uno de los más importantes, utilizado 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.

2.3.4 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 8.1 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.
- 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, permite 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, en el año 2009 fue adquirida por la compañía Oracle.

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

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

- Un compilador: `javac`
- Un intérprete: `java`.
- Un generador de documentación: *javadoc*
- Un visor de applet para generar sus vistas previas, ya que un applet carece de método `main` y no se puede ejecutar con el programa `java: 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 los portales web comunidad académica de las diferentes escuelas y facultades de la Universidad Industrial de Santander se propone como metodología de desarrollo el prototipo evolutivo.

Figura 3. Prototipo Evolutivo.

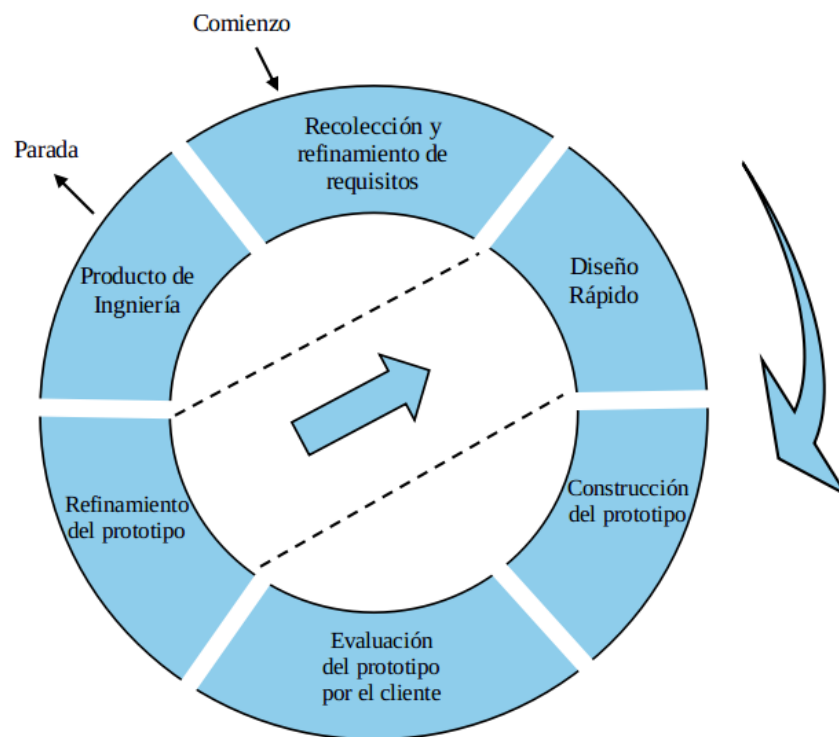


Figura 3. Prototipo Evolutivo

La elección de la metodología de prototipo evolutivo se debe a las siguientes razones:

- Los usuarios de los portales web de la comunidad académica detectan constantemente nuevas necesidades, por lo tanto estos portales requieren el desarrollo de nuevos servicios y así mismo la mejora de los ya existentes, el sistema no es considerado un producto final, es necesaria su continua actualización.
- El desarrollo de las vistas del sistema y su usabilidad pueden estar sujetas a cambios por requerimiento de los usuarios de los portales web de la comunidad académica, incluso durante el desarrollo del proyecto.
- Durante la primera etapa de los nuevos servicios es necesario tener una clara interpretación de los requerimientos de los usuarios y escuelas, aunque en repetidas ocasiones los usuarios no tienen una idea clara de lo que necesitan. Esto hace necesaria la construcción de prototipos brindando así la posibilidad de efectuar refinamientos de los requerimientos en forma sucesiva a fin de acercarse al producto deseado.
- Otra ventaja es tener la posibilidad de realizar cambios en etapas tempranas y crear varios prototipos evaluables durante el desarrollo, de modo que se obtenga una metodología integral para el proceso de evaluación del programa.
- Esta metodología favorece la realimentación constante y autocrítica del sistema, lo que conlleva a que se produzca muchas pruebas antes de dar un nuevo prototipo, así como mejoras rápidas a problemas, en etapas tempranas de desarrollo, evitando así fallos durante su uso.

Procedimiento a seguir para la metodología planteada:

- Se hace un análisis de los requerimientos para la construcción de los prototipos.
- Se desglosan los objetivos globales con el fin de tener una idea más detallada del software a realizar, mediante reuniones entre los desarrolladores y los usuarios, en las cuales se identifican los requerimientos de los usuarios y se concluyen los aspectos que requieren una mayor definición.

- Posteriormente se presenta al usuario el diseño de un prototipo enfocado en los aspectos visuales del software, métodos de entrada y formatos de salida, para proceder a la construcción del mismo.
- 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 gráfico estándar para visualizar, especificar, construir y documentar un sistema para describir un modelo del sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

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

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

3.2.2 Diagramas de casos de uso

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

Figura 4. Diagramas de Casos de Uso

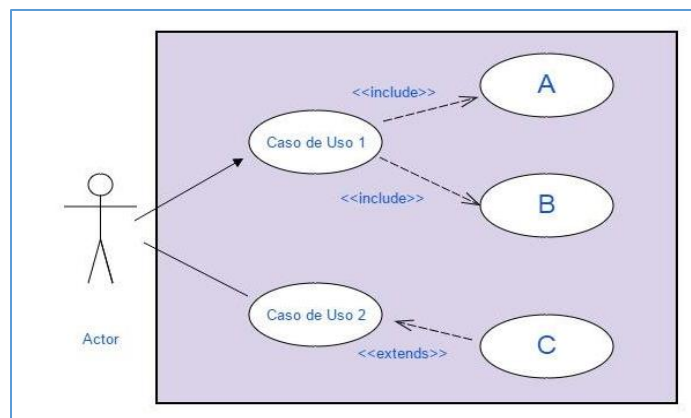


Figura 4. Diagramas de Casos de Uso

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 tiene permisos extras a los que tiene un usuario comúnmente dentro del sitio; alguno de estos son los auxiliares de administración del portal, los profesores, las secretarías de las escuelas, entre otras. Dentro de esta categoría se incluye también el súper administrador.
- **Súper Administrador:** Es el usuario que puede administrar, controlar y modificar los portales web de las escuelas, sus parámetros y sus usuarios.
- **Usuario Portal Web Comunidad Académica:** Es el tipo de usuario común de los portales web y a quien van dirigidas las páginas de servicio. Este usuario solo tiene control sobre sus servicios permitidos.

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

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

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 5. Diagrama de Secuencias

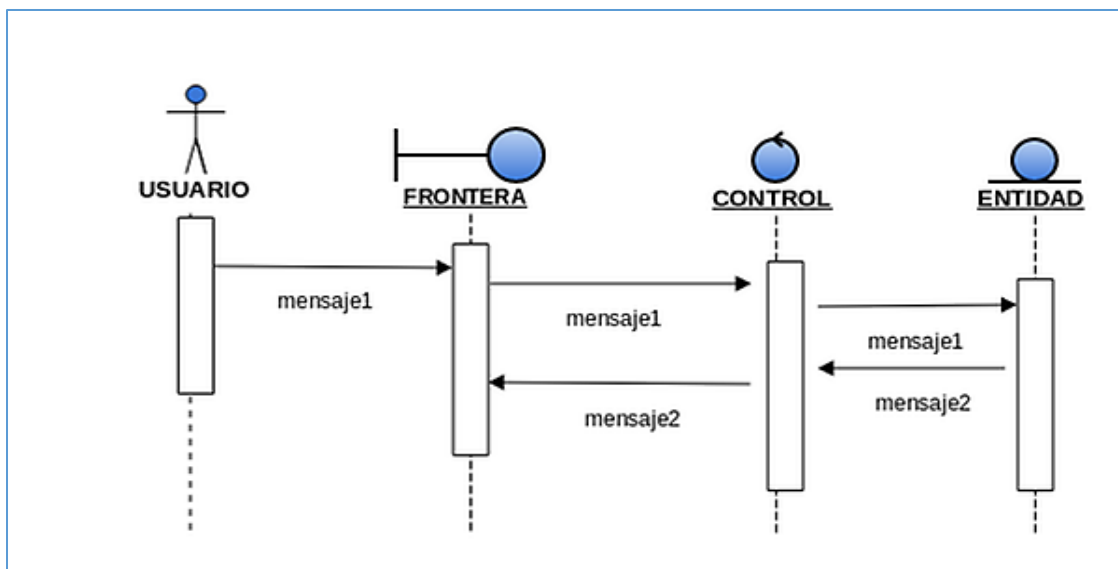


Figura 5. Diagrama de Secuencias

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 los prefijos TP, TR y TB que indican si la tabla es principal, relacional o básica, respectivamente, y exceptuando también 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.

Como se mencionó, se han definido tres categorías para las diferentes tablas que conforman la base de datos. Dada la categoría de la tabla, se agrega 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 tabla que almacena las categorías de clasificación de los usuarios del portal web es llamada “TB_Categorias”, por ejemplo.
- **Tabla de Relación:** Surge de la relación muchos a muchos de una o dos tablas cualquiera. Los nombres de las tablas de relación deben ser siempre descriptivos para cada relación. El prefijo a anteponer a los nombres de estas tablas es “TR_”, por ejemplo la tabla “Votantes” de una encuesta es conocida como “TR_ Votantes”.
- **Tabla Principal:** Aquella cuyo número 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 EISWeb. 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: Inscripciones.java, Noticias.java.

3.3.4 Páginas JSP

Los nombres de las páginas JSP que componen los portales web comunidad académica son escritos de manera que la primera letra es una mayúscula seguido de letras minúsculas, en caso de que el nombre del JSP sea compuesto por dos o más palabras, entonces la primera de cada palabra debe ir en mayúscula , por ejemplo, CrearActividad.jsp, VerInscritos.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: ProyectosDeExtensión, InscripcionesUis.

- El sitio cuenta con un directorio llamado “images”; en éste se encuentran almacenados todos los archivos .jpg, .gif, .png.
- Para el desarrollo de este proyecto se crearon los directorios “ProyectosDeExtension”, “InscripcionesUis” y “Noticias” los cuales almacenan los archivos correspondientes al funcionamiento de dichos servicios.

4. DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO

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

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

4.1 PROTOTIPO ESPERADO

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

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

Cambiar el diseño, vista y presentación de gráficos de los resultados en el servicio Encuestas. Se debe presentar una vista de los resultados de las encuestas en pestañas de acuerdo a los permisos de los usuarios, mostrando las estadísticas con sus respectivos gráficos, mejorando la precisión de los resultados y así mismo la interpretación de los datos.

Objetivo inicial:

Cambiar el diseño para una mejor presentación de los resultados y usabilidad del servicio.

Requisitos finales del objetivo:

- Usuarios, con permisos de administrador, pueden gestionar encuestas con una vista nueva de mayor usabilidad.
- Permite crear encuestas, en las cuales se señala un título y descripción, el rango de fechas de participación en la encuesta por parte de los usuarios, y se agregan las preguntas estándar y personalizadas que deben responder los usuarios encuestados, además se puede dar permisos para los usuarios, de manera que puedan ver resultados y/o listados de votantes o no.
- Los usuarios que crean una encuesta pueden elegir si activar esta sólo en la escuela a la que pertenece o en otras escuelas de la comunidad uis, además incluir si es pública o privada y en este último caso debe indicar a qué categorías de usuario va dirigida, por ejemplo egresados, estudiantes de pregrado, profesores, entre otras, puede escoger una o varias categorías de usuario.
- Permite agregar campos personalizados, ofreciendo diferentes tipos de campos tales como texto, párrafo, selección múltiple, título y salto de sección.
- Permite modificar el orden de los campos, así como editarlos y eliminarlos.
- Permite editar, duplicar y eliminar encuestas.
- Se tiene una vista de resultados presentada por pestañas que están activas a los usuarios según los permisos del administrador de la encuesta. Dicho administrador tiene acceso a todas estas pestañas.
- Se muestran estadísticas de las respuestas; en los casos de preguntas de opción múltiple única respuesta y opción múltiple múltiple respuesta se presentan gráficos para su más fácil interpretación.

Reingeniería del servicio noticias comunidad escuela y noticias comunidad UIS. Cambiar el diseño de publicaciones y edición de noticias en el servicio Publicar Cartelera. Se debe dar mayor relevancia a las noticias de los decanatos en sus escuelas. Crear el ranking para destacar las noticias más

relevantes y una vista en pestañas de las noticias de acuerdo a su debida clasificación. Crear éste mismo servicio para los portales grupo de las diferentes escuelas.

Objetivo inicial:

Cambio en la presentación de las noticias y reingeniería del servicio.

Requisitos finales del objetivo:

- Cualquier usuario de los portales web tanto de los grupos como de la comunidad UIS puede publicar una noticia.
- Al crear una noticia el usuario puede elegir si la publica o no en el grupo, en la escuela o en otras escuelas.
- Cuando el usuario ingrese a la sección de noticias le debe aparecer de primero las noticias del decanato de la escuela a la cual pertenezca.
- El usuario puede calificar una noticia mediante estrellas siendo cinco estrellas la máxima calificación y una estrella la menor calificación.

Modificar el servicio Gestionar mis proyectos con el fin de mejorar la usabilidad del servicio. Cambiar el diseño de la vista de los proyectos, agregar la opción de reactivar una actividad finalizada, enviar correos programados, incluir el campo descripción del proyecto y pasar a programar actividades por días. Además se busca asociar los proyectos a una debida área de gestión.

Objetivo inicial:

Mejorar la usabilidad del módulo proyectos de extensión y además agregar la funcionalidad de asociar proyectos a un eje de gestión.

Requisitos finales del objetivo para el módulo proyectos de extensión:

- El usuario puede crear ejes de gestión donde podrá asociar proyectos.

- El usuario puede crear proyectos asignando fechas y no semanas para su inicio y fin.
- El usuario puede reactivar una actividad finalizada.
- AL crear un proyecto, una etapa y una actividad el usuario debe agregar una descripción de ellas.
- Los usuarios participantes del proyecto recibirán un correo de notificación del proyecto.

Es necesario crear el servicio inscripciones comunidad UIS, que permite crear inscripciones donde participen estudiantes de todas las escuelas, se deben agregar estadísticas y gráficos al igual que el servicio Encuestas tanto para éste servicio como para inscripciones comunidad escuela con el fin de mejorar la experiencia de usuario.

Objetivo inicial:

Desarrollo del módulo de inscripciones comunidad UIS con el fin de extender el servicio de inscripción de usuarios a las distintas escuelas.

Requisitos finales del módulo administrador de inscripciones:

- Permite crear inscripciones, en las cuales se señala un título y descripción, el rango de fechas de validez de la inscripción, y se agregan los campos estándar y personalizados que deben diligenciar los interesados, se puede señalar si la inscripción es pública (dirigida a usuarios sin acceso o session en alguno de los portales) o privada (dirigida únicamente a usuarios de los portales); además puede seleccionar una o varias escuelas de destino.
- Permite agregar campos personalizados, ofreciendo diferentes tipos de campos tales como texto, párrafo, selección múltiple, título y salto de sección.
- Permite modificar el orden de los campos, así como editarlos y eliminarlos.
- Permite editar, duplicar y eliminar inscripciones.

- Identifica y agrupa a los usuarios dependiendo de su estado, el cual puede ser “Interesado”, “Pre-inscrito” o “Inscrito”.
- Permite el envío de correos.
- Permite ver la información diligenciada por los usuarios, así como actualizarla en caso de ser necesario.
- Permite cambiar el estado de los interesados.

Los siguientes son los requisitos del formulario de inscripción:

- Este servicio puede ser accedido por usuarios externos si se especifica la inscripción como pública, y sólo por usuarios del portal si se especifica la inscripción como privada.
- Se carga la información diligenciada por el usuario si ya se ha inscrito previamente, dándole la posibilidad de editar su inscripción.
- Si la persona sólo diligencia sus datos personales ingresa a la inscripción como “interesado” y si diligencia todo el formulario ingresa a la inscripción como “pre-inscrito”.

Administración.

Objetivos iniciales:

- Efectuar labores de administración a los portales web de la Escuela de Estudios Industriales y Empresariales y la Escuela de Ingeniería de Petróleos; realizando las siguientes actividades:
- 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.

- Atender consultas y sugerencias que realicen los usuarios con el fin de mejorar y/u ofrecer nuevos servicios en la próxima versión de los portales WEB.
- Realizar una revisión constante de los archivos que se encuentran en la base de datos del sitio para evitar que los portales se saturen de información innecesaria y/o desactualizada.

Requisitos finales del objetivo para usuarios con perfil de administrador:

- Corregir cada uno de los fallos que a diario se presentan en los portales web.
- Mantenimiento a la bases de datos para eliminar tablas que ya no se usan o crear nuevas que se necesitan para que los nuevos servicios funcionen.
- Actualizar el diagrama Entidad/Relación de la base de datos Diamante y subirlo al portal EISIWeb, en el espacio del grupo Calumet; para realizar futuras actualizaciones sobre estos últimos.
- Revisar la estructura de directorios del portal, borrar los archivos que ya no se usan y agregar nuevos servicios implantados en el portal.
- Realizar limpiezas en cuanto a conversaciones y datos que ya no sean necesarios dentro de las Bases de datos.
- Mantener los JavaBeans actualizados y subirlos al portal EISIWeb, en el espacio del grupo Calumet; para realizar futuras modificaciones sobre estos últimos.

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

Mantenimiento.

Objetivos iniciales:

Llevar a cabo labores de mantenimiento a los portales EEIEWeb y EIPWeb en los que podemos encontrar:

- Implementar los nuevos servicios o mejoras realizadas por otros desarrolladores del grupo CALUMET, dentro de los portales EEIEWeb y EIPWeb, 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.

Requisitos finales del objetivo para usuarios con perfil de administrador:

- Realizar copias periódicas de la base de datos Diamante y de los directorios de los sitios EEIEWeb y EIPWeb.
- Eliminar de la base de datos diamante información que no se usa, ya sean archivos o conversaciones.
- Hacer la actualización periódica de la base de datos Diamante.

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

Soporte a Usuarios.

Objetivo inicial:

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

- Capacitar usuarios y estudiantes del primer nivel en el uso de servicios dentro de los portales web 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.
- Modificar los estados de los usuarios de acuerdo a la relación con la escuela (activo, inactivo, suspendido).

Requisitos finales del objetivo para usuarios con perfil de administrador:

- Conocer el total funcionamiento de los portales EEIEWeb y EIPWeb, 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 en cuanto a:

- Implantación del sitio local para la creación de nuevos servicios y realización de pruebas.
- Llevar a cabo la familiarización con el entorno de los portales web.

Requisitos finales del objetivo para usuarios con perfil de administrador:

- Dar orientación a los nuevos integrantes del grupo sobre posibles errores en la instalación del sitio local.

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

4.1.1 Diagramas de Casos de Uso

4.1.1.1 Servicio de Proyectos de Extensión

Figura 6. Diagrama de Casos de Uso: Proyectos de Extensión - Director y Codirector

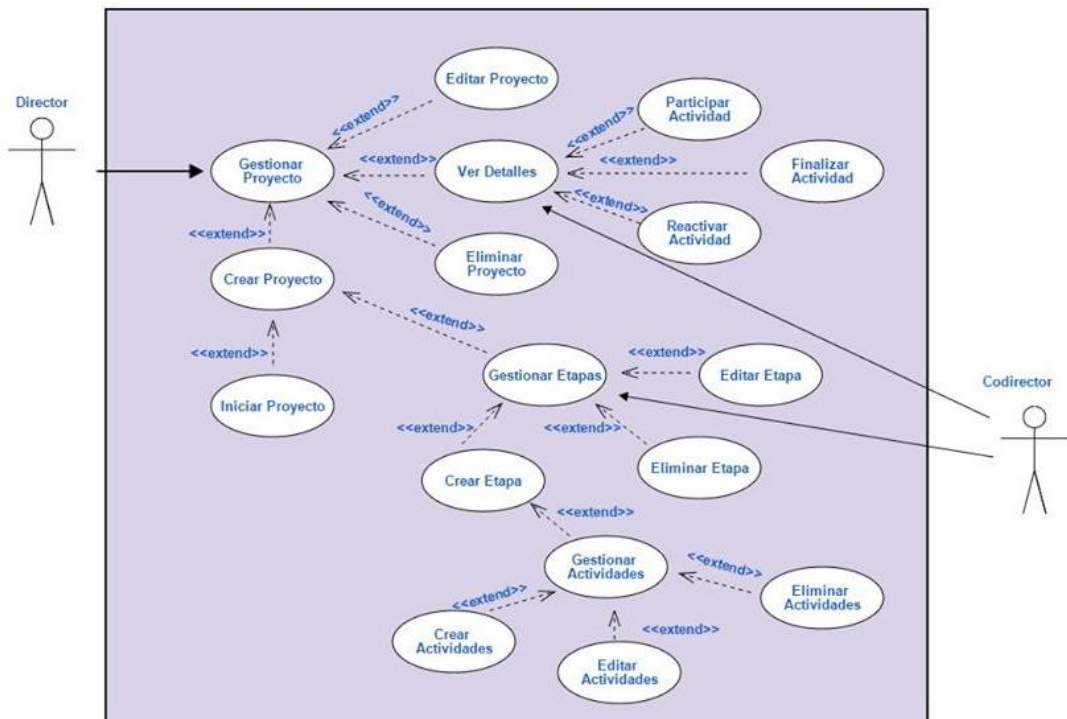


Figura 6. Diagrama de Casos de Uso: Proyectos de Extensión - Director y Codirector

Figura 7. Diagrama de Casos de Uso: Proyectos de Extensión - Auxiliar

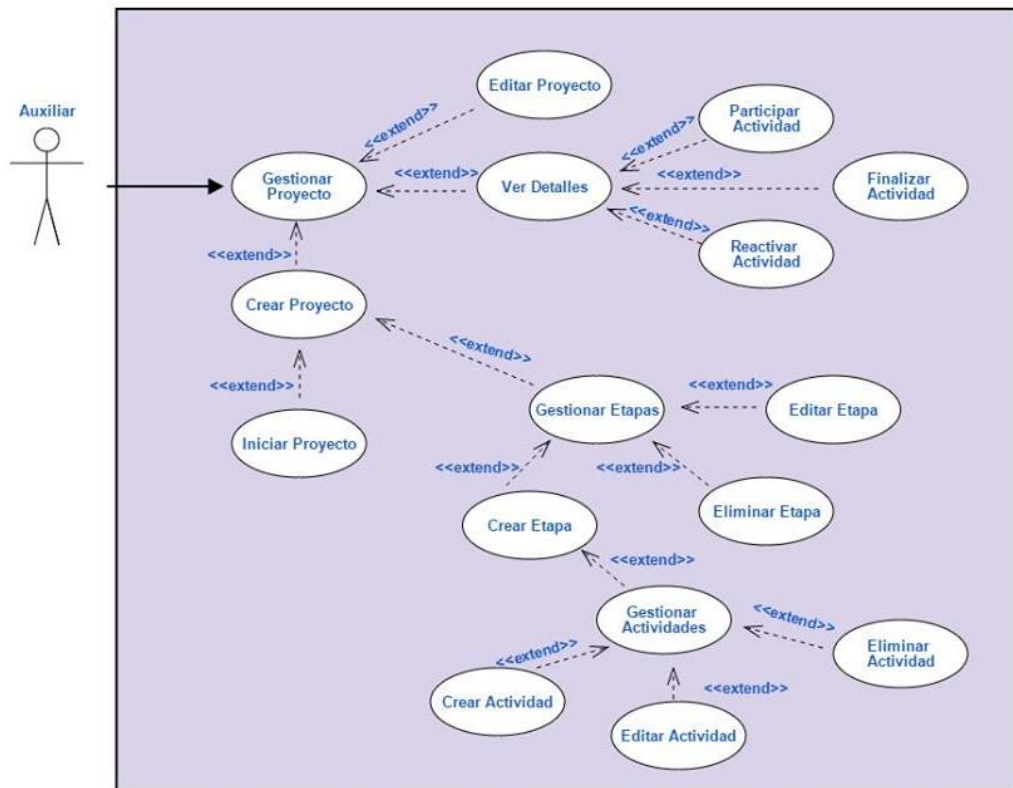


Figura 7. Diagrama de Casos de Uso: Proyectos de Extensión - Auxiliar

Figura 8. Diagrama de Casos de Uso: Proyectos de Extensión - Profesor y Responsable

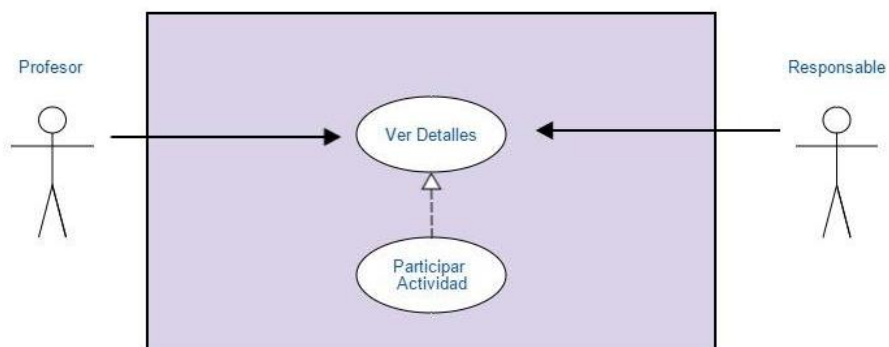


Figura 8. Diagrama de Casos de Uso: Proyectos de Extensión - Profesor y Responsable

Figura 9. Diagrama de Casos de Uso: Ejes de gestión

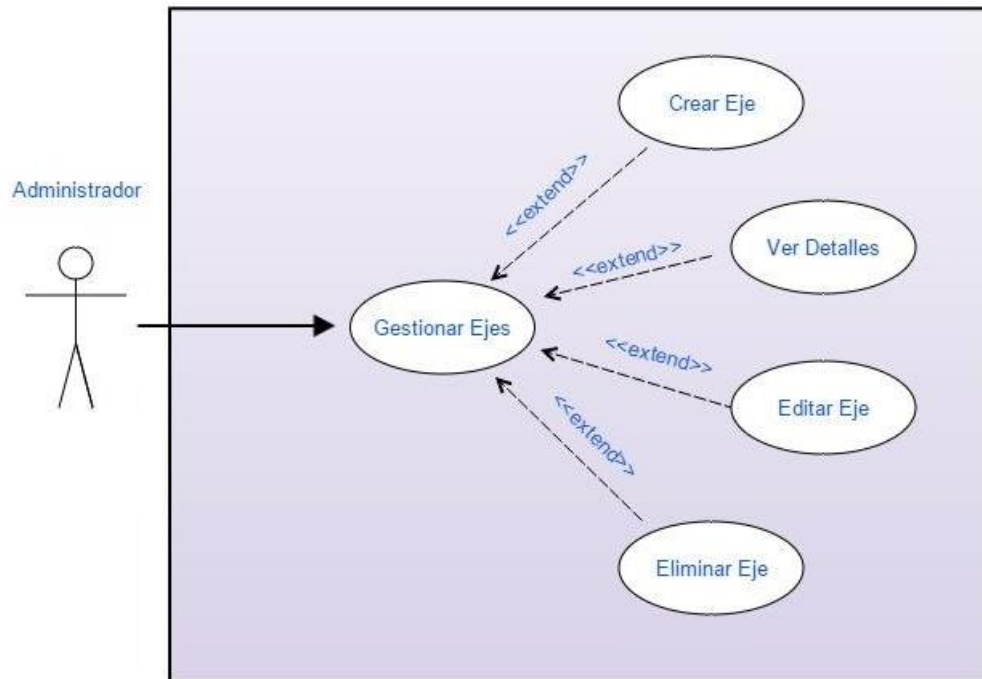


Figura 9. Diagrama de Casos de Uso: Ejes de gestión

4.1.1.2 Servicio de Encuestas

Figura 10. Diagrama de Casos de Uso: Encuestas

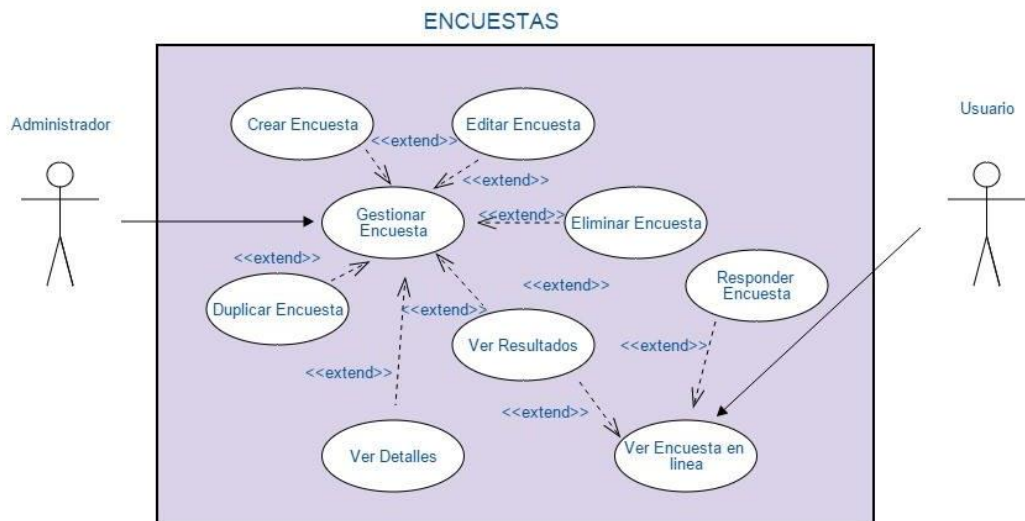


Figura 10. Diagrama de Casos de Uso: Encuestas

4.1.1.3 Servicio de Inscripciones

Figura 11. Diagrama de Casos de Uso: Inscripciones

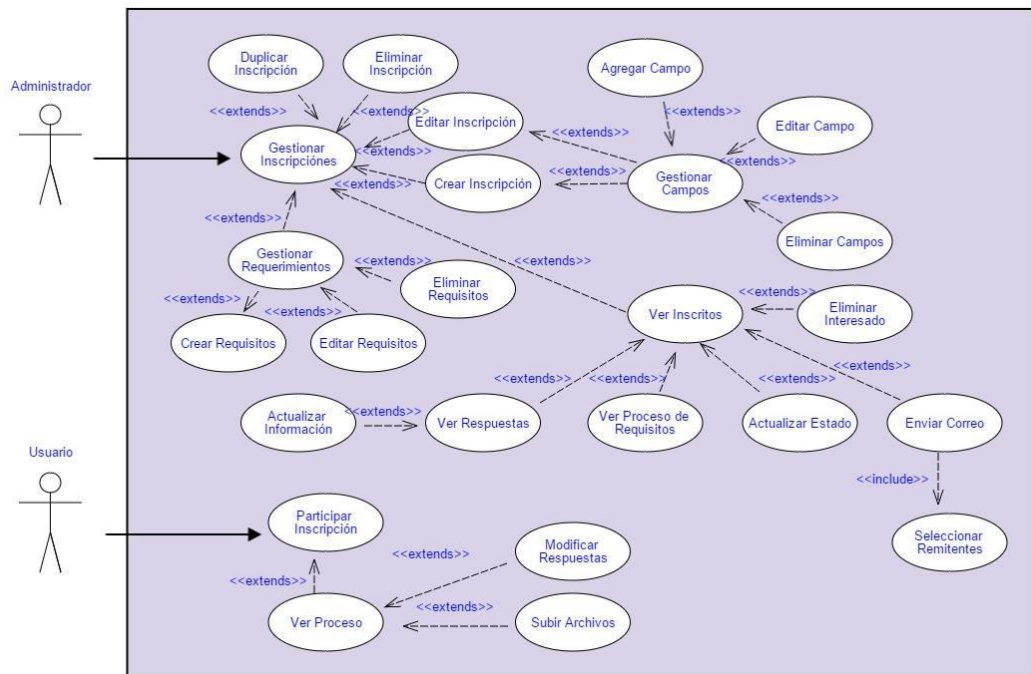


Figura 11. Diagrama de Casos de Uso: Inscripciones

4.1.1.4 Servicio de Noticias

Figura 12. Diagrama de Casos de Uso: Noticias

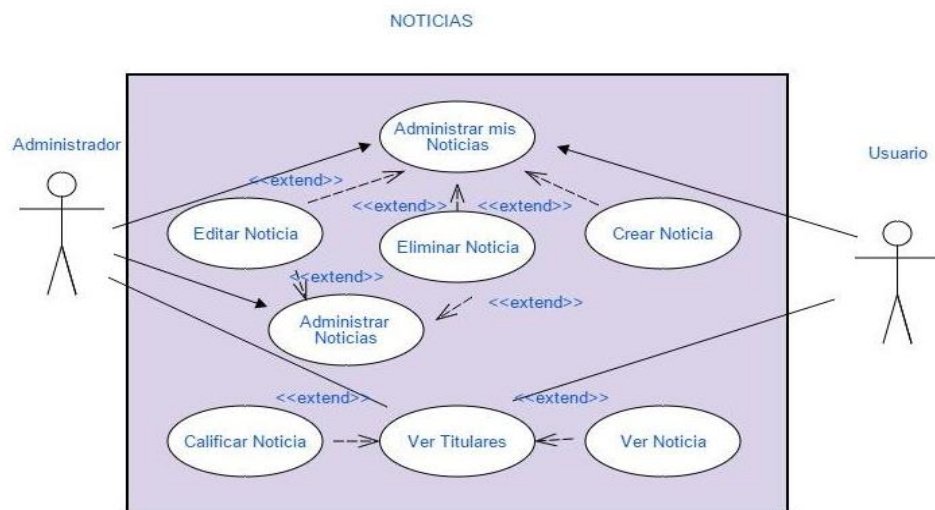


Figura 12. Diagrama de Casos de Uso: Noticias

4.1.1.5 Mantenimiento y Administración de los portales de la Escuela de Petróleos y la Escuela de Estudios Industriales y empresariales.

Figura 13. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.

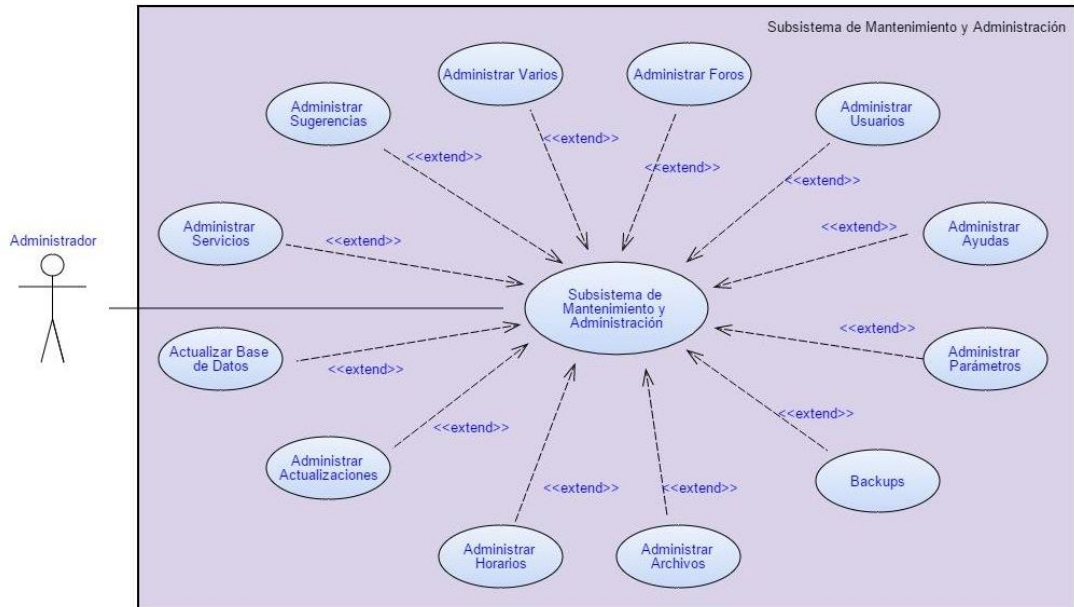


Figura 13. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración

4.1.2 Documentación de Casos de Uso del Sistema

4.1.2.1 Servicio de Proyectos de Extensión

Tabla 1. Casos de uso: Proyectos de extensión

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Proyecto	Acceder al servicio que lista los proyectos en los que el usuario tiene un rol asignado.
Crear Proyecto	Crear un nuevo proyecto de extensión asignando diferentes participantes y estableciendo fechas.
Editar Proyecto	Modifica y guarda actualizaciones de un proyecto anteriormente creado.
Eliminar Proyecto	Borra un proyecto que haya sido creado anteriormente y no puede ser consultado nuevamente.
Ver Detalles	Muestra los detalles del proyecto en general y también de las actividades de cada etapa.
Participar Actividad	Permite comunicarse mediante comentarios con los demás miembros del proyecto en cada actividad.
Finalizar Actividad	Cambiar el estado de una actividad de activo ha finalizado, lo cual indica que la actividad ya fue realizada.
Reactivar Actividad	Cambiar el estado de la actividad de finalizado a activo.
Iniciar Proyecto	Inicia un proyecto creado anteriormente puesto que por defecto se crea inactivo.
Gestionar Etapas	Acceder a una lista de las etapas del proyecto, si las hay, de modo contrario está la opción de crear una etapa.
Crear Etapa	Crear una nueva etapa al proyecto con su debida información.

Tabla 1. Casos de uso: Proyectos de extensión (Continuación)

Editar Etapa	Modificar y actualizar los detalles de una etapa.
Eliminar Etapa	Eliminar una etapa del proyecto.
Gestionar Actividades	Acceder a una lista de las Actividades del proyecto, si hay, de modo contrario está la opción de crear una.
Crear Actividad	Crear una nueva actividad a la etapa de un proyecto estableciendo fechas.
Editar Actividad	Modificar y actualizar los detalles de una actividad.
Eliminar Actividad	Eliminar una actividad de cierta etapa del proyecto.

*Tabla 1. Casos de uso: Proyectos de extensión***Tabla 2. Casos de uso: Ejes de gestión**

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Ejes	Acceder a una lista de los ejes de gestión creados hasta el momento.
Crear Eje	Agregar un nuevo eje de gestión al listado, con los detalles requeridos, incluso unos proyectos asociados.
Eliminar Eje	Borrar un eje creado anteriormente.
Editar Eje	Modificar y actualizar la información de un eje de gestión, incluso borrar, agregar o cambiar los proyectos asociados.
Ver Detalles	Acceder a la información detallada del eje creado anteriormente.

Tabla 2. Casos de uso: Ejes de gestión

4.1.2.2 Servicio de Noticias

Tabla 3. Casos de uso: Noticias

TÍTULO	FUNCIONES PRIMARIAS
Administrar Mis Noticias	Acceder al listado de noticias que fueron creadas por el mismo usuario anteriormente. Aquí un usuario no encuentra noticias creadas por otros.
Editar Noticia	Modificar y actualizar tanto la información que contiene la noticia como las fechas de publicación y en qué escuelas serán publicadas.
Eliminar Noticia	Borrar una noticia creada anteriormente.
Crear Noticia	Crear una noticia con los detalles de publicación y sus debidas fechas asignadas.
Administrar Noticias	Acceder al listado de todas las noticias creadas anteriormente por los usuarios. Aquí se encuentran todas las noticias.
Ver Titulares	Acceder a la vista de los titulares de las noticias que se encuentran activas.
Ver Noticia	Acceder a la vista de toda la noticia.
Calificar Noticia	Dar una calificación a una noticia.

Tabla 3. Casos de uso: Noticias

4.1.2.4 Servicio de Encuestas

Tabla 4. Casos de uso: Encuestas

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Encuesta	Acceder al listado de las encuestas que han sido creadas hasta el momento por el usuario.

Tabla 4. Casos de uso: Encuestas (continuación).

Crear Encuesta	Crear una encuesta con la debida información, estableciendo fechas y con preguntas personalizadas por el usuario.
Editar Encuesta	Modificar y actualizar la información de una encuesta creada anteriormente.
Eliminar Encuesta	Borrar una encuesta creada anteriormente.
Duplicar Encuesta	Crear una encuesta igual que otra creada anteriormente, pero con estado Inactivo.
Ver Detalles	Mostrar información relevante de la encuesta, por ejemplo donde ha sido publicada.
Ver Resultados	Acceder a la vista de los resultados de la encuesta, el administrador puede ver los resultados detallados y los votantes de la encuesta, el usuario que participa puede ver lo que el administrador le permita.
Ver Encuesta En Línea	Acceder al listado de las encuestas que pueden ser respondidas por el usuario en sesión o usuario invitado.
Responder Encuesta	Participar en una encuesta en línea.

Tabla 4. Casos de uso: Encuestas

4.1.2.5 Servicio de Inscripciones

Tabla 5. Casos de uso: Inscripciones

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Inscripciones	Acceder al servicio que lista las inscripciones que se hayan creado.
Crear Inscripción	Crear una nueva inscripción con la información requerida, incluyendo establecimiento de fechas.
Editar Inscripción	Modificar y actualizar la información de una encuesta creada anteriormente.

Tabla 5. Casos de uso: Inscripciones (continuación).

Eliminar Inscripción	Borrar una encuesta que haya sido creada anteriormente.
Duplicar Inscripción	Crear una inscripción igual a otra pero con estado inactivo.
Gestionar Campo	Acceder a las opciones que tiene el usuario respecto a los campos que haya creado o que desea crear.
Agregar Campo	Agregar un nuevo campo en el formulario de inscripción.
Editar Campo	Editar la información de un campo que haya sido agregado al formulario.
Eliminar Campo	Eliminar un campo que haya sido agregado al formulario.
Ver Inscritos	Ver las personas inscritas en la inscripción, aparecen agrupadas por su estado de inscripción "Interesado", "Pre-Inscrito", "Requisitos cumplidos" e "Inscrito".
Actualizar Estado	Actualizar el estado de un interesado en la inscripción.
Eliminar Interesado	Eliminar un interesado del proceso de inscripción.
Ver Respuestas	Ver la información diligenciada por el interesado en el proceso de inscripción.
Actualizar Información	Acceder a las opciones que tiene el usuario que creó la inscripción para actualizar la información de los interesados.
Ver Proceso De Requisitos	Acceder a las opciones que tiene el usuario que creó la inscripción para hacer seguimiento a los requisitos de la misma.
Gestionar Requerimientos	Acceder a las opciones que tiene el usuario respecto al manejo de los requisitos de la encuesta.
Crear Requisitos	Crear requisitos que deben ser cumplidos por los usuarios para aspirar a ser inscritos.

Tabla 5. Casos de uso: Inscripciones (continuación).

Editar Requisitos	Modificar y actualizar los requisitos que deben ser cumplidos por los usuarios para continuar con el proceso de inscripción.
Eliminar Requisitos	Borrar requisitos que se hayan creado para la inscripción.
Enviar Correos	Acceder al servicio de enviar correos para que el usuario que creó la inscripción se comuniquen con los interesados.
Participar Inscripción	Acceder al servicio que lista las inscripciones que se encuentren vigentes, las cuales pueden ser privadas (dirigidas únicamente a usuarios del portal) o públicas (usuarios externos).
Ver Proceso	Ver el cumplimiento de los requerimientos de la inscripción.
Modificar Respuestas	Editar y actualizar la información que se otorgó en el formulario de inscripción.
Subir Archivos	Subir los archivos que se requieren para la inscripción.

Tabla 5. Casos de uso: Inscripciones

4.1.2.6 Mantenimiento y Administración de los portales EEIWeb y de

EIPWEB

Tabla 6. Casos de uso: Subsistema de Mantenimiento y Administración.

TÍTULO	FUNCIONES PRIMARIAS
Administrar Servicios	Crear, modificar y eliminar servicios del portal.
Administrar Sugerencias	Revisar todas las sugerencias para atender o eliminar.
Administrar Varios	Insertar, cambiar y eliminar banner.

Tabla 6. Casos de uso: Subsistema de Mantenimiento y Administración (continuación).

Administrar Usuarios	Crear usuarios manualmente. Ver información de usuarios. Cambiar clave. Cambiar parámetros. Autorizar servicios a categorías y perfiles. Crear, modificar, eliminar categorías y perfiles Enviar correos.
Backups	Generar copias de la base de datos División y Diamante
Actualizar Base de Datos	Actualizar tablas de datos de División y Diamante. Ver contadores de registros.

Tabla 6. Casos de uso: Subsistema de Mantenimiento y Administración

4.1.3 Diseño y Análisis

Diagrama Entidad/Relación de los servicios desarrollados

Servicio Encuestas

Figura 14. Diagrama E/R: Encuestas

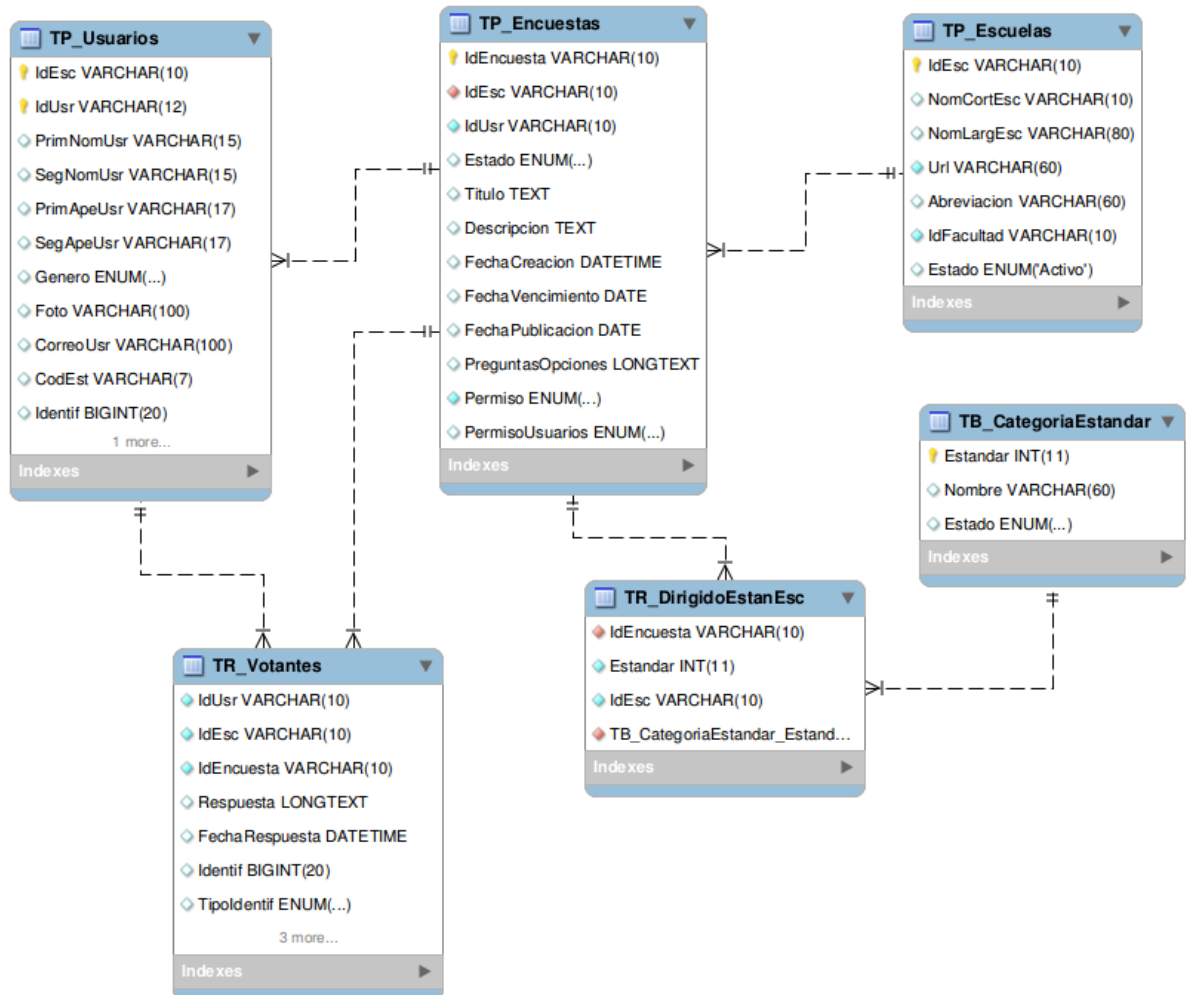


Figura 14. Diagrama E/R: Encuestas

Servicio Inscripciones

Figura 15. Diagrama E/R: Inscripciones

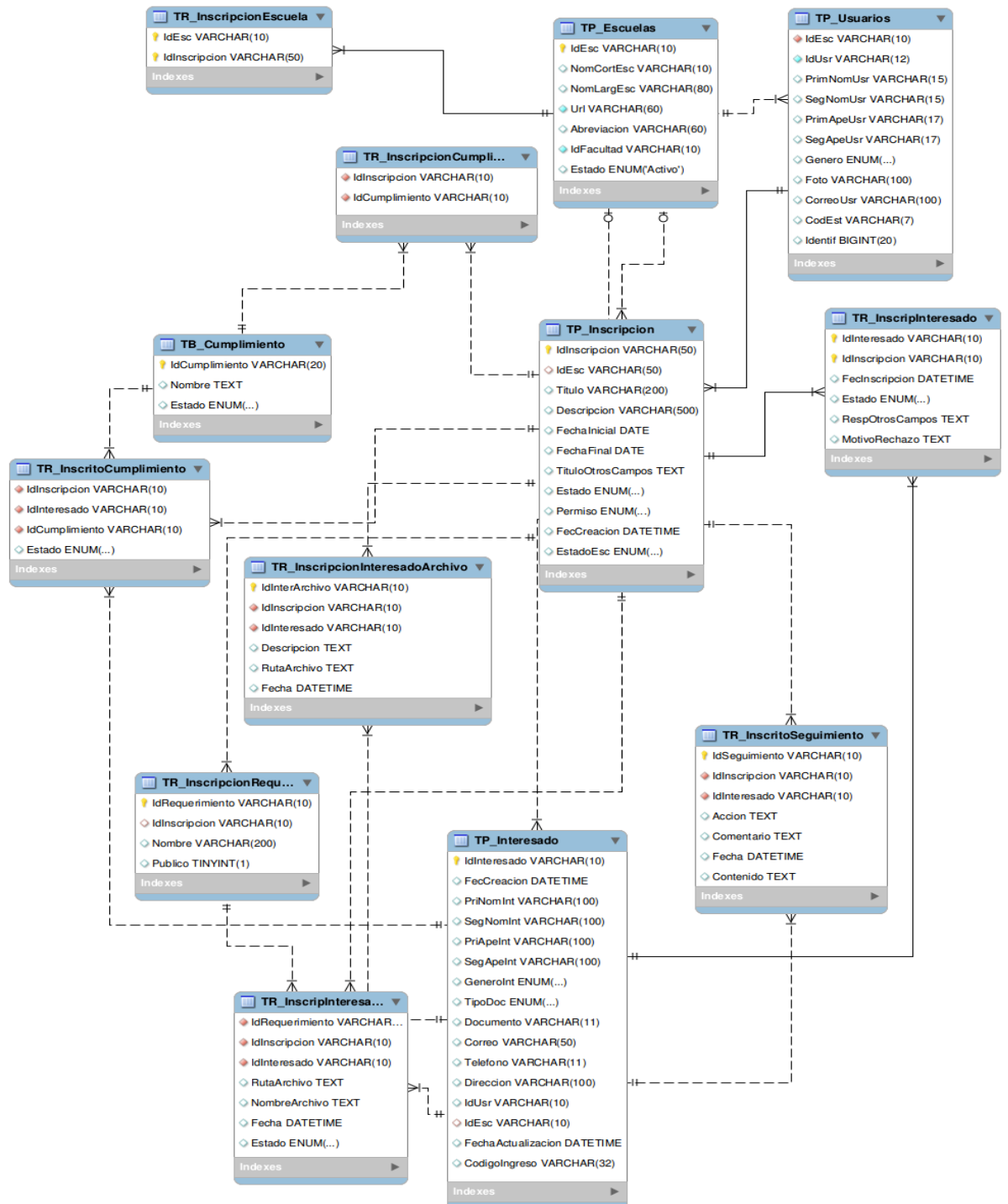


Figura 15. Diagrama E/R: Inscripciones

Servicio Noticias Grupo

Figura 17. Diagrama E/R: Noticias Grupo

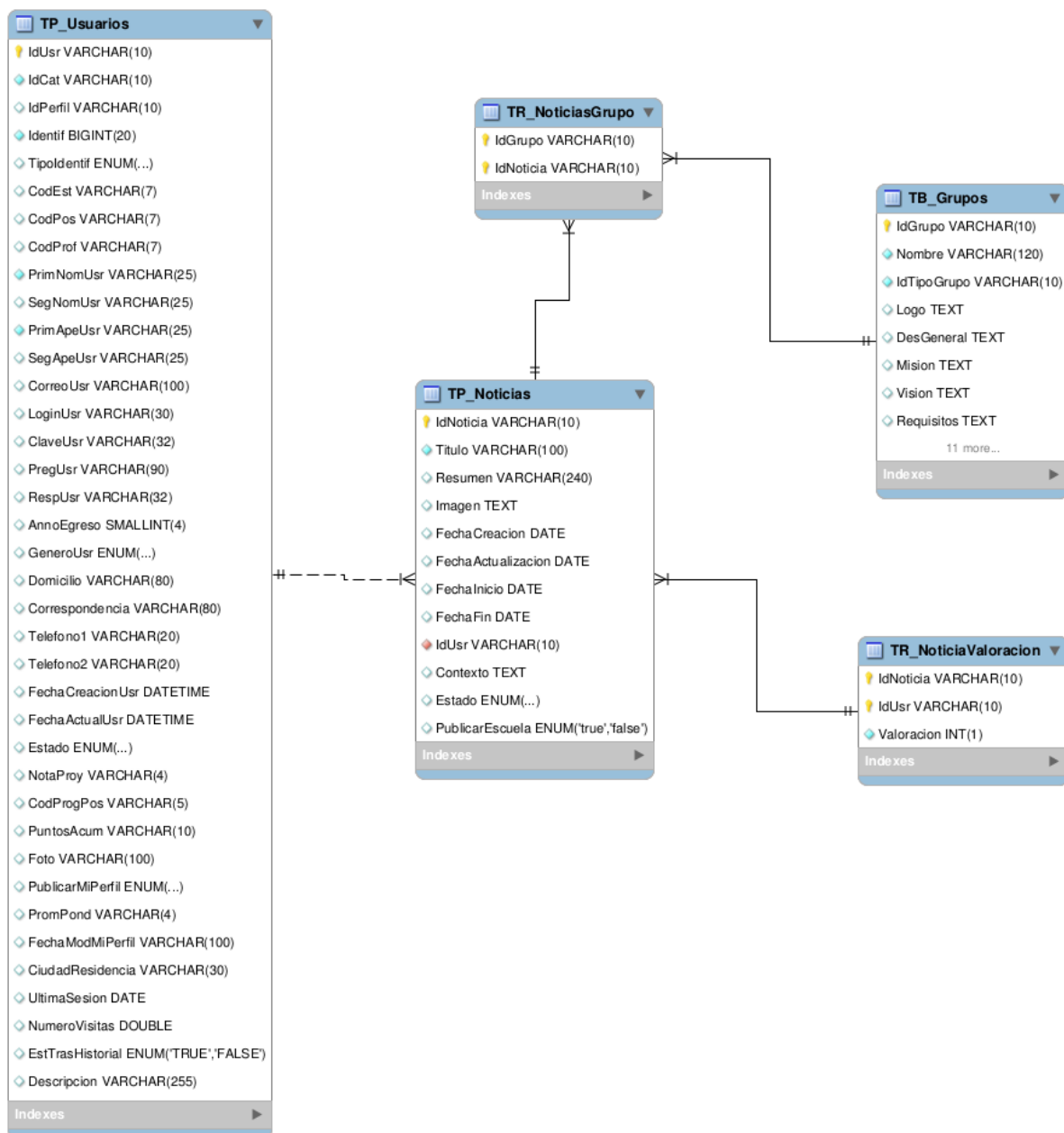


Figura 17. Diagrama E/R: Noticias Grupo

Servicio Noticias Comunidad UIS

Figura 18. Diagrama E/R: Noticias Comunidad UIS

Noticias Poseidon

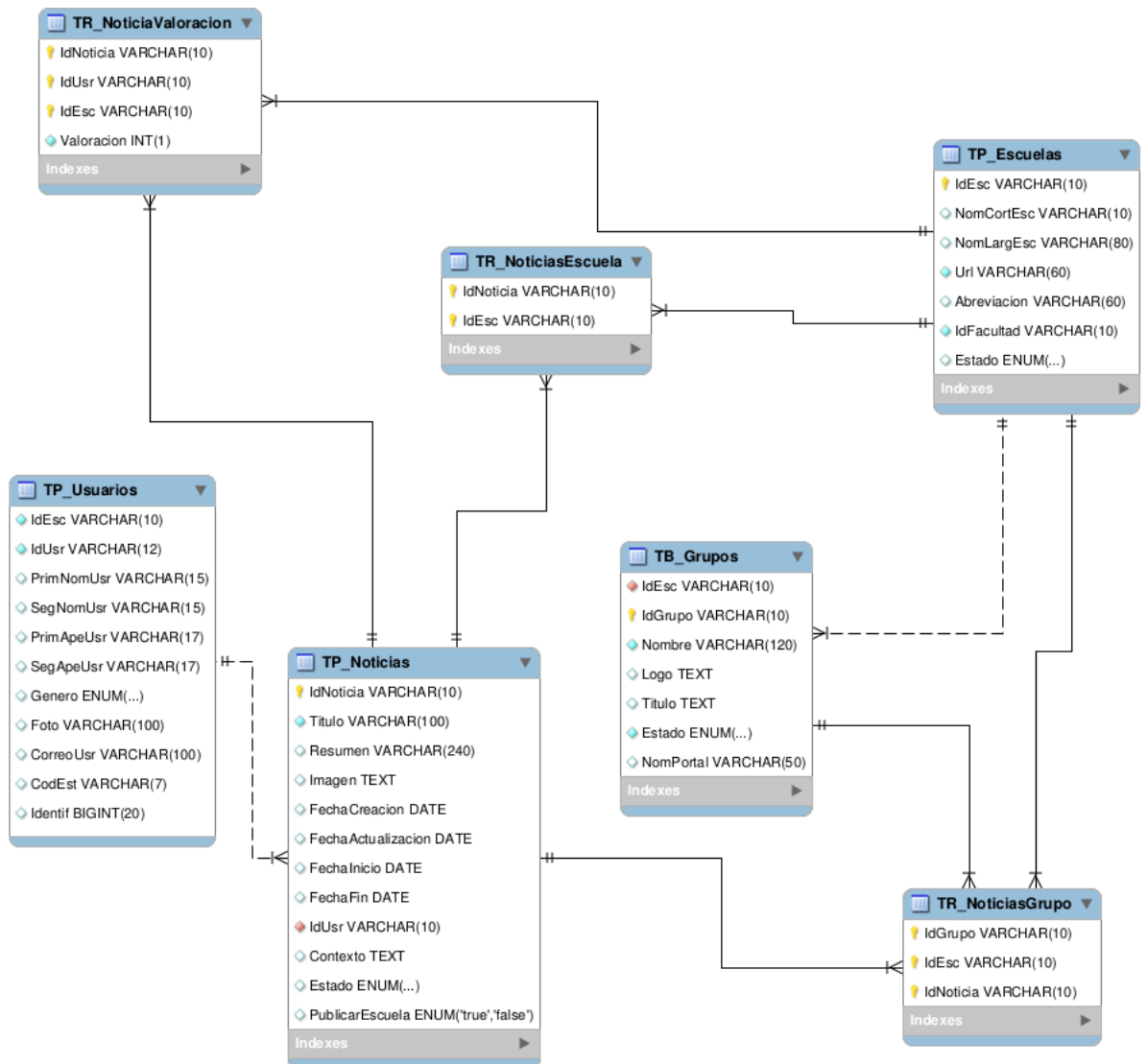


Figura 18. Diagrama E/R: Noticias Comunidad UIS

Descripción de las Entidades

Definición de las entidades utilizadas y creadas en la base de datos Poseidón ubicada en el servidor del portal web de la Vicerrectoría Académica y de la base de datos Diamante.

Tabla 7. Descripción de las Entidades

ENTIDAD	DESCRIPCION
TP_Usuarios	Contiene la información de todos los usuarios registrados en el portal.
TB_Grupos	Contiene la información de los grupos de investigación.
TP_Escuelas	Contiene la información de las escuelas de la comunidad UIS.
TP_Noticias	Contiene la información básica de las noticias.
TR_NoticiasGrupo	Relaciona las noticias con los grupos.
TR_NoticiasEscuelas	Relaciona las noticias con las escuelas.
TR_NoticiaValoración	Contiene la valoración que un usuario le da a una noticia.
TP_ProyectoExt	Contiene la información básica de los proyectos.
TP_Etapas	Contiene la información básica de las etapas.
TP_ActividadesEtapa	Contiene la información básica de las actividades.
TP_EntidadInteresada	Contiene la información básica de las entidades interesadas en un proyecto.
TR_ProyectoEntidad	Relaciona los proyectos de extensión con las entidades
TR_ActividadRequisito	Contiene el Id de la actividades que son requeridas para realizar una actividad posterior.
TR_ActividadComentario	Contiene los comentarios de los usuarios en las actividades.
TR_ProyectoGrupo	Relaciona los proyectos con los grupos
TR_ProyectoFinanciación	Relaciona los proyectos con la entidad que lo financia.

Tabla 7. Descripción de las Entidades (Continuación)

TB_FuenteFinanciación	Contiene la información básica de las fuentes de financiación.
TR_ProyectoObjetivo	Relaciona los proyectos con los objetivos generales y específicos asociados
TR_ProjExtParticipantes	Relaciona los proyectos con los participantes asociados.
TP_Eje	Contiene la información básica de los ejes.
TR_EjeProyecto	Relaciona los ejes de los proyectos.
TP_Encuestas	Contiene la información básica de una encuesta.
TR_DirigidoEstanEsc	Relaciona las encuestas con las categorías estándar y las escuelas.
TR_Votantes	Relaciona los participantes con las encuestas.
TB_CategoriaEstandar	Contiene todas las categorías que puede tener un usuario.
TP_Inscripcion	Contiene la información básica de las inscripciones
TB_Cumplimiento	Contiene la información básica del cumplimiento de los requerimientos.
TR_InscripcionEscuela	Relaciona una inscripción con las escuelas.
TR_InscripcionCumplimiento	Relaciona una inscripción con el cumplimiento.
TP_Interesado	Contiene información básica de las personas interesadas en participar en una inscripción.
TR_InscripInteresado	Relaciona una inscripción con un interesado.
TR_InscritoCumplimiento	Relaciona un inscrito con un cumplimiento.
TR_InscritoSeguimiento	Contiene el seguimiento de un interesado en una inscripción.
TR_InscripcionInteresadoArchiv o	Contiene los archivos de un interesado que participa en una inscripción.

Tabla 7. Descripción de las Entidades (Continuación)

TR_InscripcionInteresadoRequerimiento	Contiene los requerimientos de un interesado que participa en una inscripción.
TR_InscripcionRequerimiento	Contiene los requerimientos de una inscripción.

Tabla 7. Descripción de las Entidades

4.1.4 Modelo de Procesos del Sistema

Para una mejor interpretación de los modelos de los procesos del sistema se realizaron los diagramas de secuencia necesarios para cada caso de uso en los que se explica con detalle los pasos para el funcionamiento de cada uno de los servicios.

Servicio de Noticias

Figura 19. Diagrama de secuencia: Gestionar Noticias-Crear Noticia

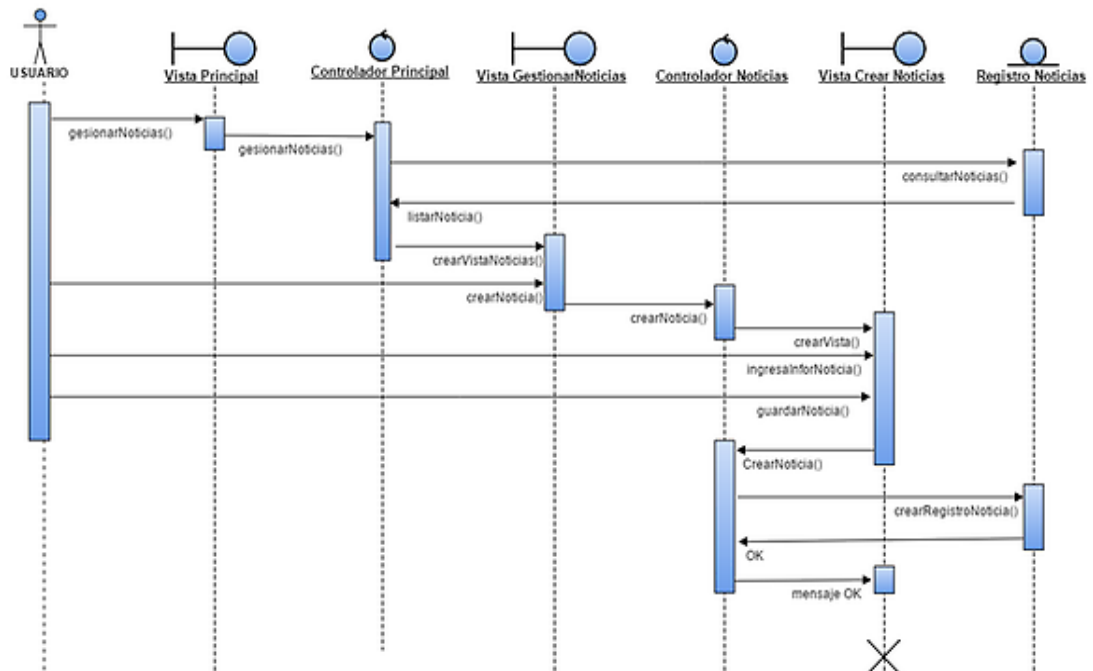


Figura 19. Diagrama de secuencia: Gestionar Noticias-Crear Noticia

1. El Usuario solicita gestionar noticias en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta las noticias en base de datos y despliega la vista VistaGestionarNoticias.
3. El Usuario solicita crear una noticia en la vista VistaGestionarNoticias.
4. El controlador ControladorNoticia recibe la petición de crearNoticia, crea la noticia en base de datos y retorna un mensaje de Ok al usuario en la vista vistaCrearNoticia.
5. Se destruye la vista vistaCrearNoticia.

Figura 20. Diagrama de secuencia: Gestionar Noticias-Editar Noticia

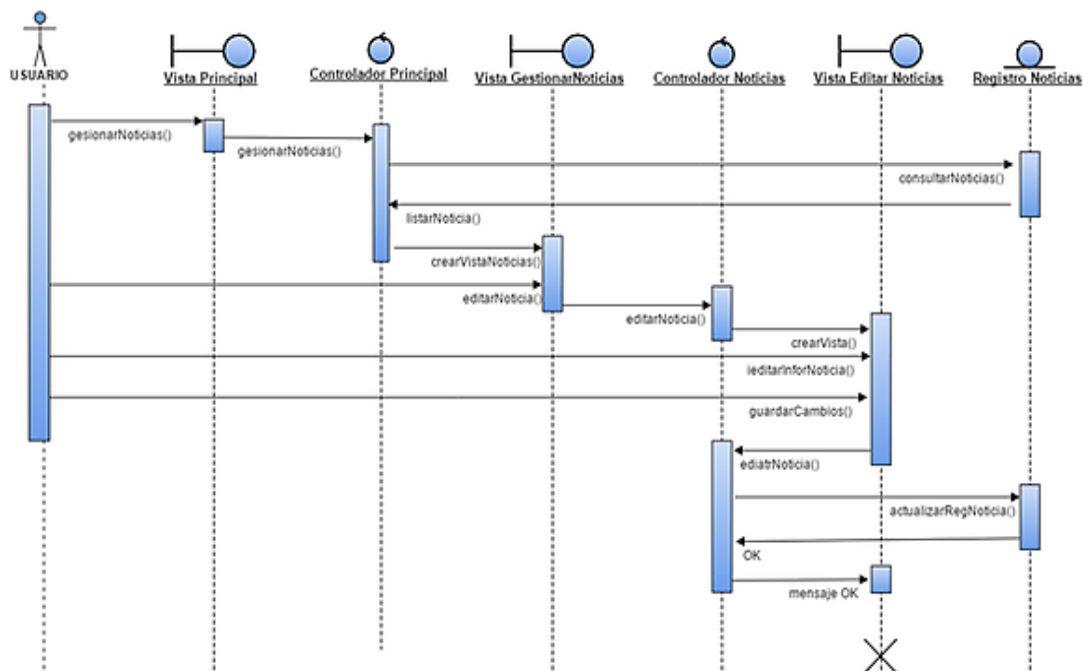


Figura 20. Diagrama de secuencia: Gestionar Noticias-Editar Noticia

1. El Usuario solicita gestionar noticias en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta las noticias en base de datos y despliega la vista VistaGestionarNoticias.
3. El Usuario solicita editar una noticia en la vista VistaGestionarNoticias.

4. El controlador ControladorNoticia recibe la petición de editarNoticia, actualiza la noticia en base de datos y retorna un mensaje de Ok al usuario en la vista vistaEditarNoticia.
5. Se destruye la vista vistaEditarNoticia.

Figura 21. Diagrama de secuencia: Ver Titulares-Calificar Noticia

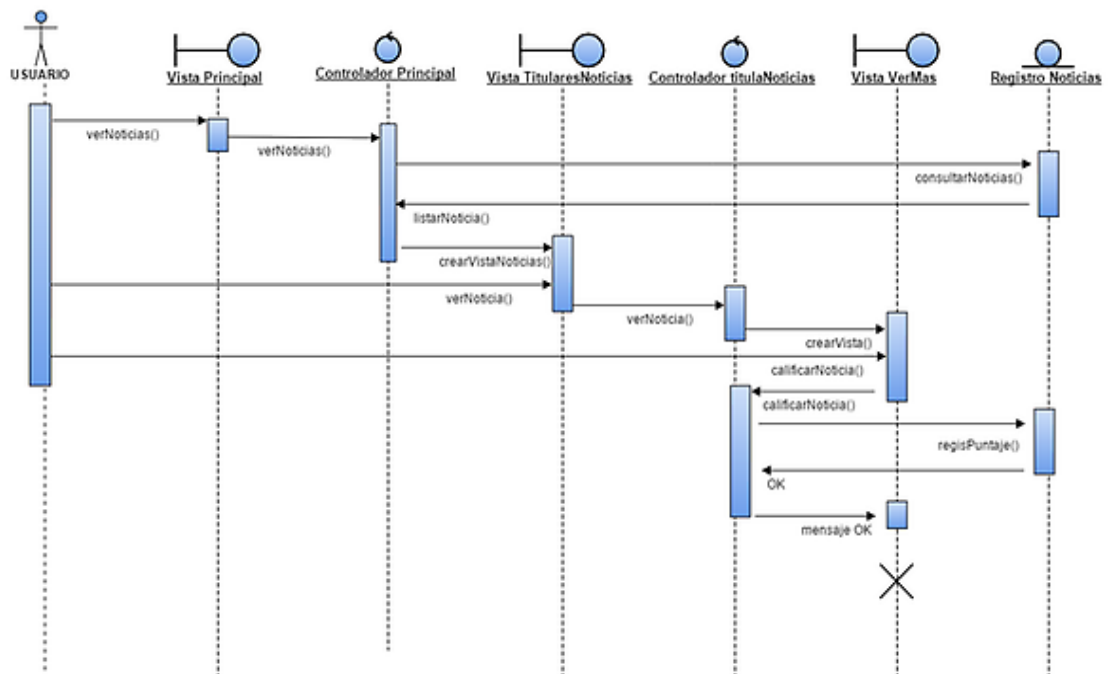


Figura 21. Diagrama de secuencia: Ver Titulares-Calificar Noticia

1. El Usuario solicita ver noticias en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta las noticias en base de datos y despliega la vista VistaTitularesNoticias.
3. El Usuario solicita calificar noticia en la vista VistaTitularesNoticias.
4. El controlador ControladorTitularNoticia recibe la petición de calificarNoticia, actualiza la noticia en base de datos y retorna un mensaje de Ok al usuario en la vista vistaVerMas.
5. Se destruye la vista vistaVerMas.

Servicio de Proyectos de Extensión

Figura 22. Diagrama de secuencia: Gestionar Proyecto-Crear Proyecto

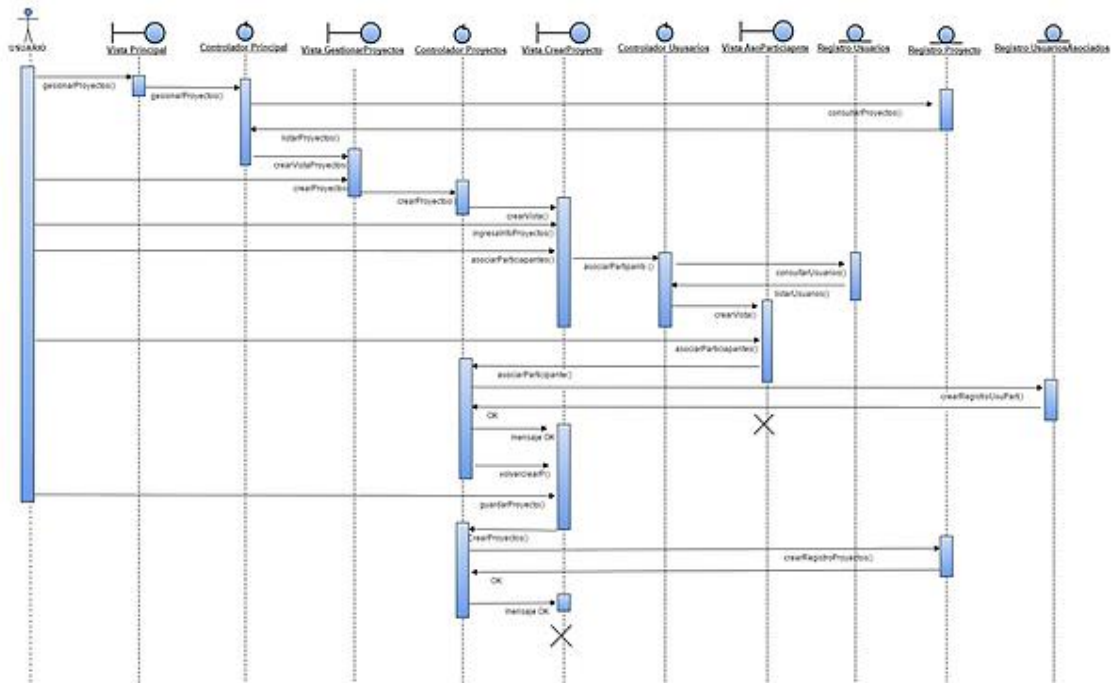


Figura 22. Diagrama de secuencia: Gestionar Proyecto-Crear Proyecto

1. El Usuario solicita gestionar proyecto en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta los proyectos en base de datos y despliega la vista VistaGestionarProyectos.
3. El Usuario solicita crear un proyecto en la vista VistaGestionarProyectos.
4. El controlador ControladorProyecto recibe la petición de crearProyecto, crea la vista para crear un proyecto.
5. El usuario solicita asociar participantes en la vista vistaCrearProyecto
6. El controlador controlUsuarios recibe la petición, consulta los usuarios en base de datos y despliega la vista asociarParticipantes.
7. El usuario asocia a cada participante asignandole un rol en la vista asociarParticipantes

8. El controlador controladorProyectos recibe la petición y crea el registro en base de datos y retorna un mensaje de Ok en la vista vistaCrearProyecto.
9. El usuario guarda los datos del proyecto y el controlador controladorProyectos recibe la petición, crea el registro en la base de datos y retorna un mensaje de ok en la vista vistaCrearProyecto.
10. Se destruye la vista vistaCrearProyecto.

Figura 23. Diagrama de secuencia: Gestionar Proyecto-Editar Proyecto

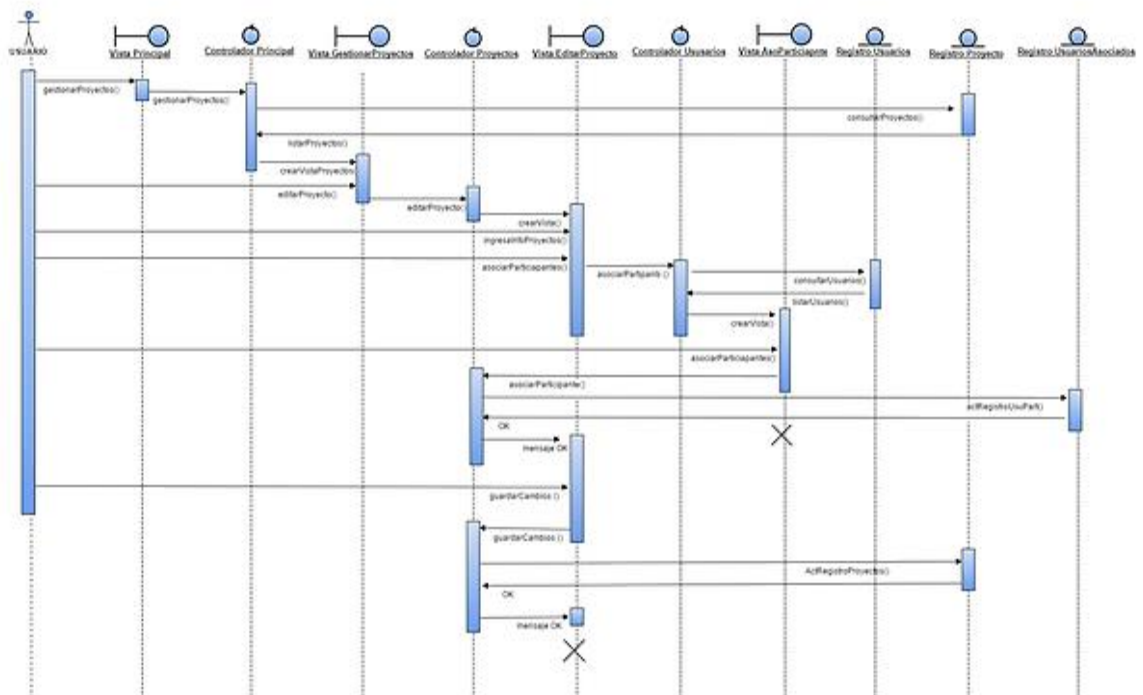


Figura 23. Diagrama de secuencia: Gestionar Proyecto-Editar Proyecto

1. El Usuario solicita gestionar proyecto en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta los proyectos en base de datos y despliega la vista VistaGestionarProyectos.
3. El Usuario solicita editar un proyecto en la vista VistaGestionarProyectos.

4. El controlador ControladorProyecto recibe la petición de editarProyecto, crea la vista para editar un proyecto.
5. El usuario solicita actualizar participantes en la vista vistaCrearProyecto
6. El controlador controlUsuarios recibe la petición, consulta los usuarios en base de datos y despliega la vista asociarParticipantes.
7. El usuario asocia a cada participante asignándole un rol en la vista asociarParticipantes
8. El controlador controladorProyectos recibe la petición y actualiza el registro en base de datos y retorna un mensaje de Ok en la vista vistaEditarProyecto.
9. El usuario guarda los cambios del proyecto y el controlador controladorProyectos recibe la petición, actualiza el registro en la base de datos y retorna un mensaje de ok en la vista vistaEditarProyecto.
10. Se destruye la vista EditarProyecto

Figura 24. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Finalizar Actividad

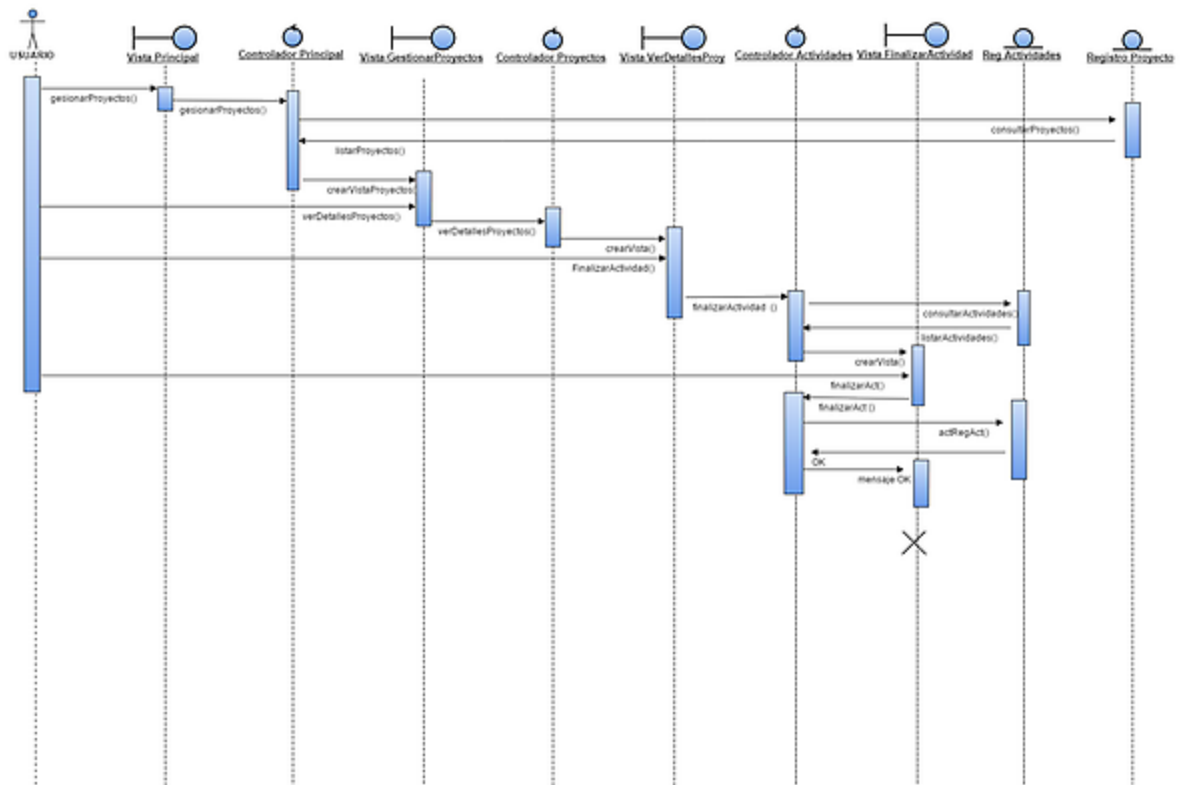


Figura 24. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Finalizar Actividad

1. El Usuario solicita gestionar proyecto en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta los proyectos en base de datos y despliega la vista VistaGestionarProyectos.
3. El Usuario solicita ver detalles de un proyecto en la vista VistaGestionarProyectos.
4. El controlador ControladorProyecto recibe la petición de verDetalles, crea la vista para ver detalles de un proyecto.
5. El usuario solicita finalizar actividad en la vista verDetallesProyecto
6. El controlador controladorActividades recibe la petición, consulta las actividades en base de datos y despliega la vista verDetallesProy
7. El usuario finaliza una actividad en la vista vistafinalizarActividad

8. El controlador controladorActividades recibe la petición y actualiza el registro en base de datos y retorna un mensaje de Ok en la vista vistaFinalizarActividad.
9. Se destruye la vista vistaFinalizarActividad.

Figura 25. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Reactivar Actividad

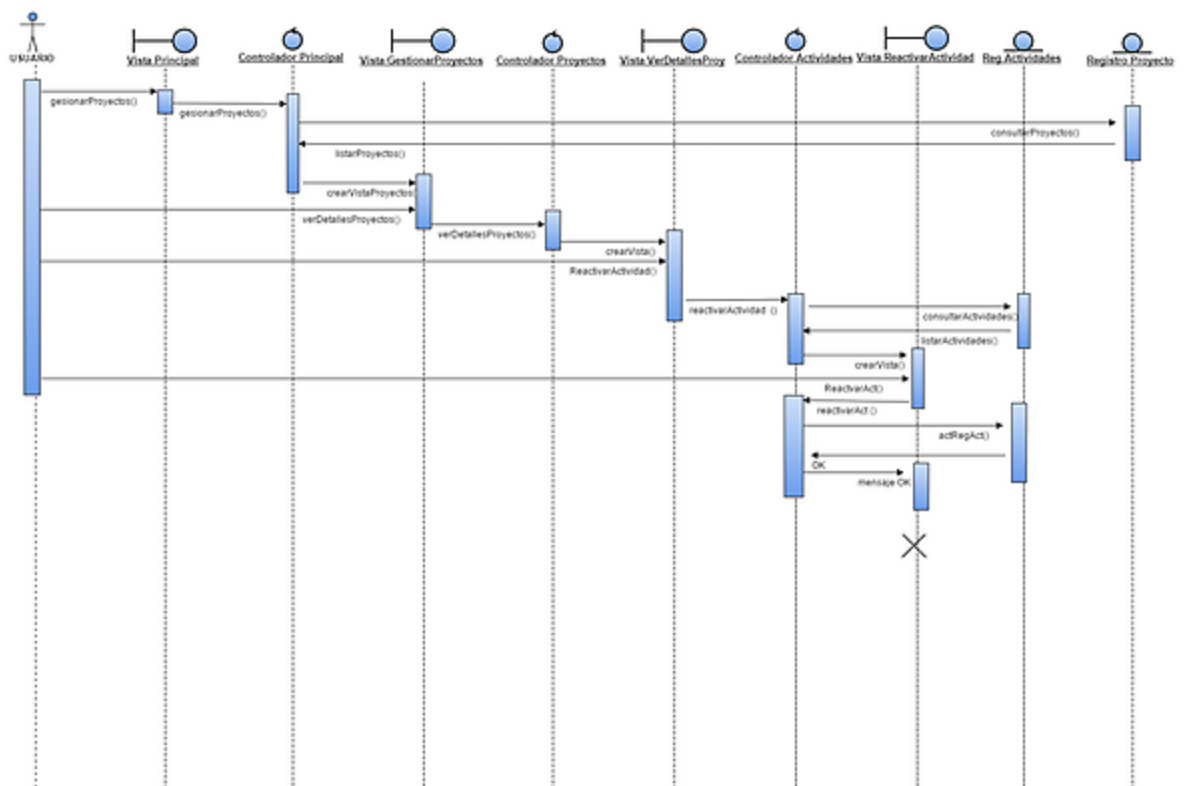


Figura 25. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Reactivar Actividad

1. El Usuario solicita gestionar proyecto en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta los proyectos en base de datos y despliega la vista VistaGestionarProyectos.
3. El Usuario solicita ver detalles de un proyecto en la vista VistaGestionarProyectos.
4. El controlador ControladorProyecto recibe la petición de verDetalles, crea la vista para ver detalles de un proyecto.

5. El usuario solicita reactivar actividad en la vista verDetallesProyecto
6. El controlador controladorActividades recibe la petición, consulta las actividades en base de datos y despliega la vista verDetallesProy
7. El usuario reactiva una actividad en la vista vistaReactivarActividad
8. El controlador controladorActividades recibe la petición y actualiza el registro en base de datos y retorna un mensaje de Ok en la vista vistaReactivarActividad.
9. Se destruye la vista vistaReactivarActividad.

Figura 26. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Participar Actividad

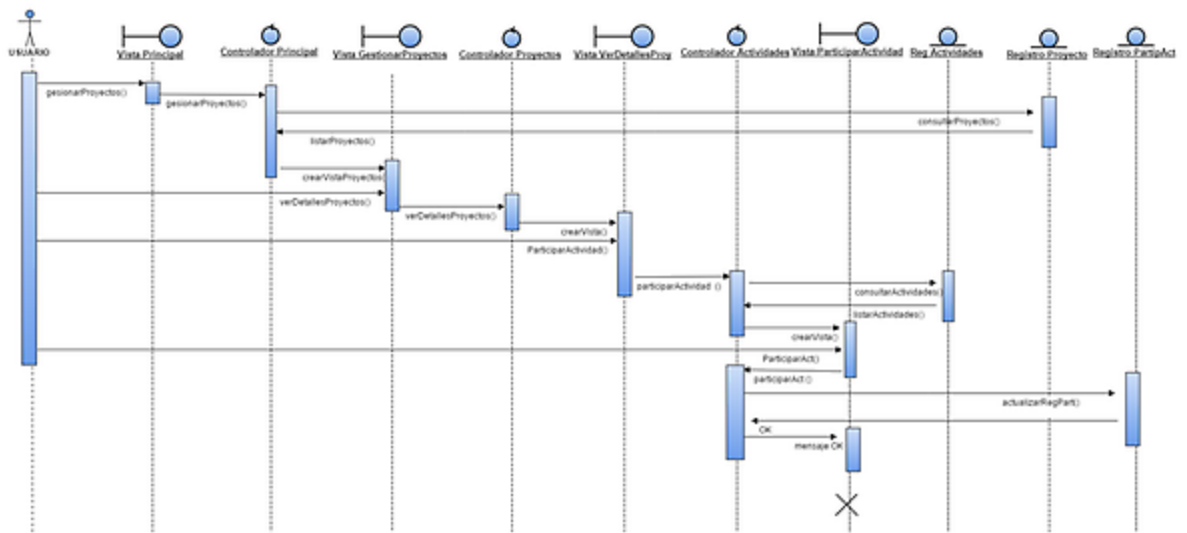


Figura 26. Diagrama de secuencia: Gestionar Proyecto-Ver Detalles-Participar Actividad

1. El Usuario solicita gestionar proyecto en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta los proyectos en base de datos y despliega la vista VistaGestionarProyectos.
3. El Usuario solicita ver detalles de un proyecto en la vista VistaGestionarProyectos.

4. El controlador ControladorProyecto recibe la petición de verDetalles, crea la vista para ver detalles de un proyecto.
5. El usuario solicita participar actividad en la vista vistaVerDetallesProy
6. El controlador controladorActividades recibe la petición, consulta las actividades en base de datos y despliega la vista vistaParticiparActividad
7. El usuario participa en una actividad en la vista vistaParticiparActividad
8. El controlador controladorActividades recibe la petición y actualiza el registro en base de datos y retorna un mensaje de Ok en la vista vistaParticiparActividad.
9. Se destruye la vista vistaParticiparActividad.

Figura 27. Diagrama de secuencia: Gestionar Proyecto-Gestionar Etapas-Crear Etapa

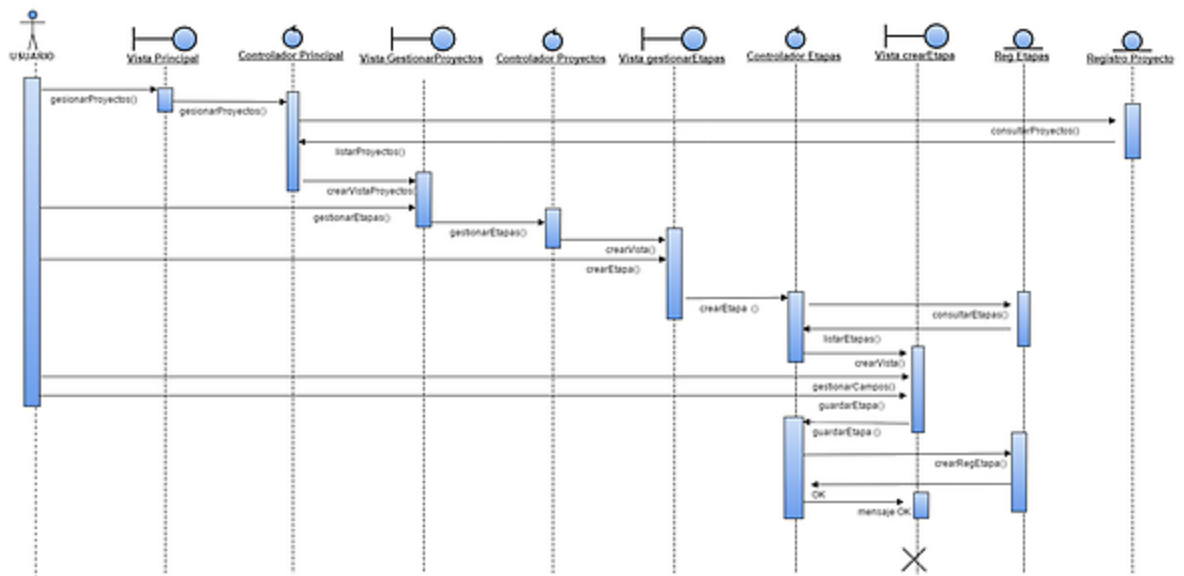


Figura 27. Diagrama de secuencia: Gestionar Proyecto-Gestionar Etapas-Crear Etapa

1. El Usuario solicita gestionar proyecto en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta los proyectos en base de datos y despliega la vista VistaGestionarProyectos.

3. El Usuario solicita Gestionar etapas de un proyecto en la vista VistaGestionarProyectos.
4. El controlador ControladorProyecto recibe la petición de gestionarEtapas, crea la vista para gestionar etapas de un proyecto.
5. El usuario solicita crear etapa en la vista vistaGestionarEtapas,
6. El controlador controladorEtapas recibe la petición, consulta las etapas en base de datos y despliega la vista vistaCrearEtapa.
7. El usuario llenas los campos en la vista vistaCrearEtapa.
8. El controlador controladorEtapas recibe la peticion y actualiza el registro en base de datos y retorna un mensaje de Ok en la vista vistaCrearEtapa.
9. Se destruye la vista vistaCrearEtapa.

Figura 28. Diagrama de secuencia: Gestionar Proyecto-Gestionar Etapas-Crear Actividad

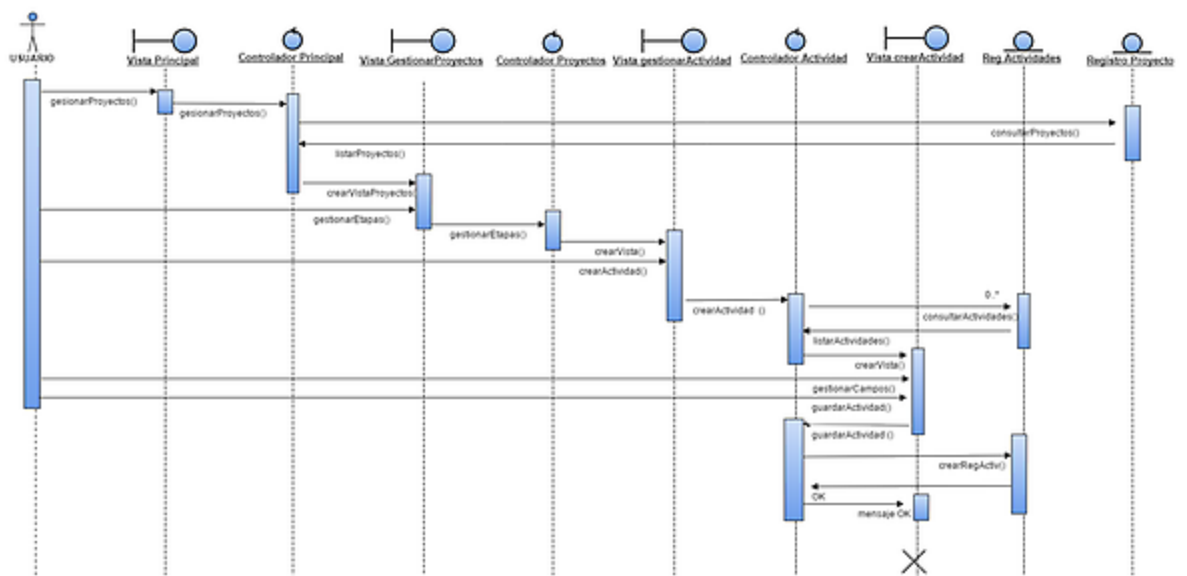


Figura 28. Diagrama de secuencia: Gestionar Proyecto-Gestionar Etapas-Crear Actividad

1. El Usuario solicita gestionar proyecto en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta los proyectos en base de datos y despliega la vista VistaGestionarProyectos.
3. El Usuario solicita Gestionar etapas de un proyecto en la vista VistaGestionarProyectos.
4. El controlador ControladorProyecto recibe la petición de gestionarActividades, crea la vista para gestionar actividades de un proyecto.
5. El usuario solicita crear actividad en la vista vistaGestionarEtapas.
6. El controlador controladorActividades recibe la petición, consulta las actividades en base de datos y despliega la vista vistaCrearActividad.
7. El usuario llenas los campos en la vista vistaCrearActividad.
8. El controlador controladorActividad recibe la peticion y actualiza el registro en base de datos y retorna un mensaje de Ok en la vista vistaCrearActividad.
9. Se destruye la vista vistaCrearActividad.

Figura 29. Diagrama de secuencia: Gestionar Encuestas-Crear Encuesta

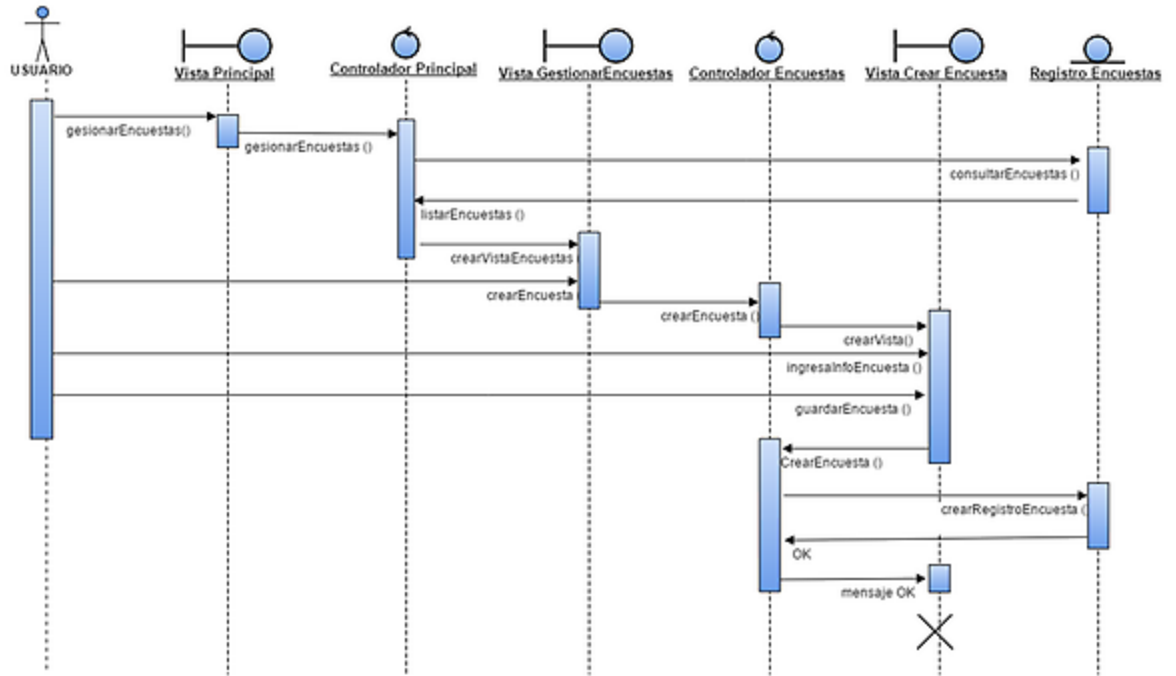


Figura 29. Diagrama de secuencia: Gestionar Encuestas-Crear Encuesta

1. El Usuario solicita gestionar Encuestas en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta las encuestas en base de datos y despliega la vista VistaGestionarEncuestas.
3. El Usuario solicita crear una encuesta en la vista VistaGestionarEncuestas.
4. El controlador ControladorEncuesta recibe la petición de crearEncuesta, crea la encuesta en base de datos y retorna un mensaje de Ok al usuario en la vista VistaCrearEncuestas.
5. Se destruye la vista VistaCrearEncuesta.

Figura 30. Diagrama de secuencia: Gestionar Encuestas-Editar Encuesta

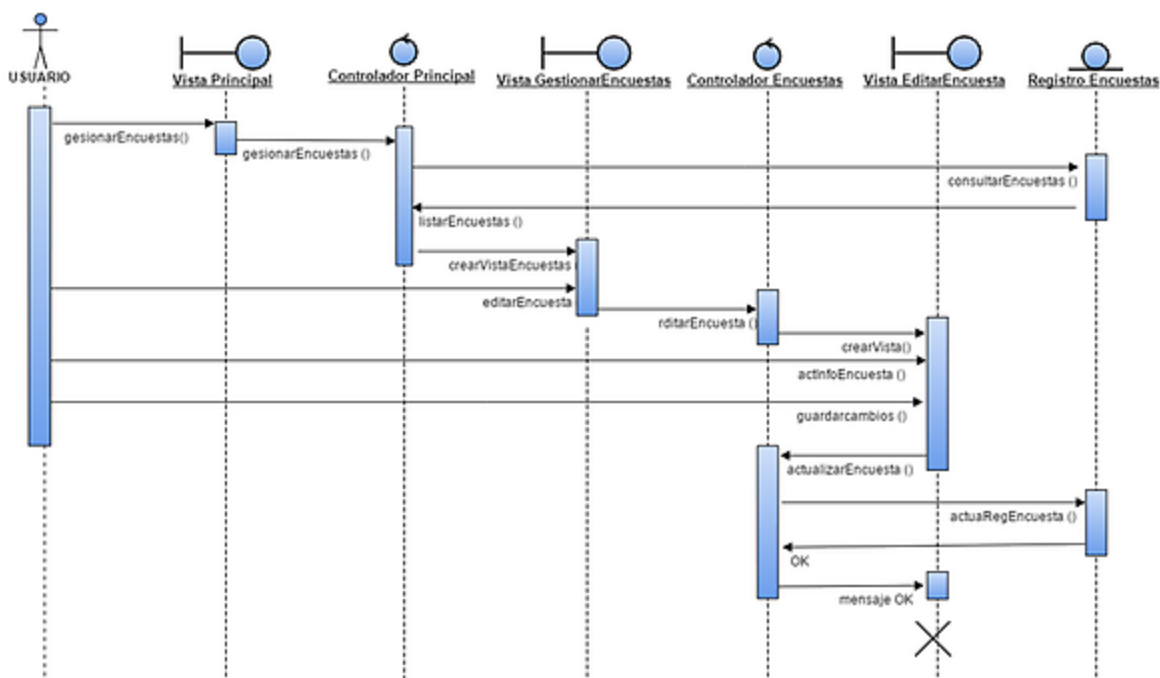


Figura 30. Diagrama de secuencia: Gestionar Encuestas-Editar Encuesta

1. El Usuario solicita gestionar Encuestas en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta las encuestas en base de datos y despliega la vista VistaGestionarEncuestas.
3. El Usuario solicita editar una encuesta en la vista VistaGestionarEncuestas.
4. El controlador ControladorEncuesta recibe la petición de editarEncuesta, actualiza la encuesta en base de datos y retorna un mensaje de Ok al usuario en la vista VistaEditarEncuestas.
5. Se destruye la vista VistaEditarEncuesta.

Figura 31. Diagrama de secuencia: Gestionar Encuestas-Ver Resultados Encuesta

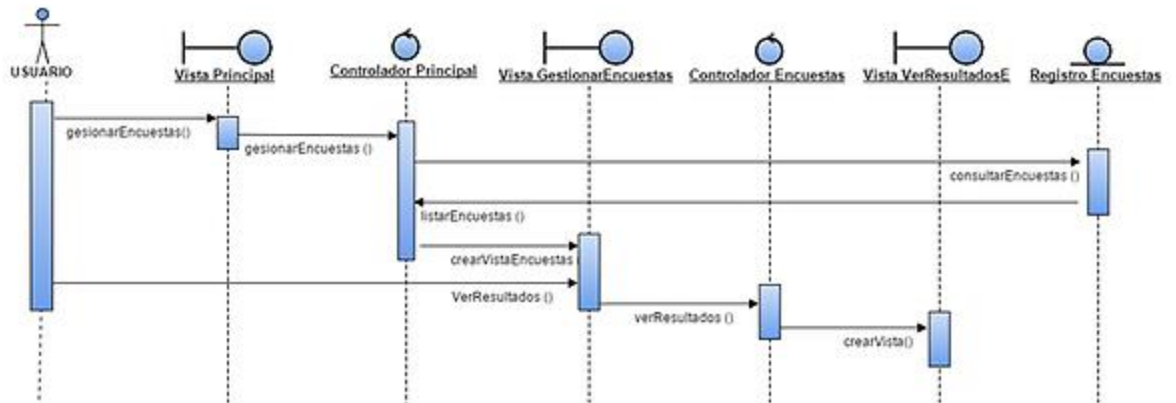


Figura 31. Diagrama de secuencia: Gestionar Encuestas-Ver Resultados Encuesta

1. El Usuario solicita gestionar Encuestas en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta las encuestas en base de datos y despliega la vista VistaGestionarEncuestas.
3. El Usuario solicita ver Resultados en una encuesta en la vista VistaGestionarEncuestas.
4. El controlador ControladorEncuesta recibe la petición de verResultados, y crea la vista vistaVerResultadosEncuesta.

Figura 32. Diagrama de secuencia: Gestionar Encuestas-Duplicar encuesta

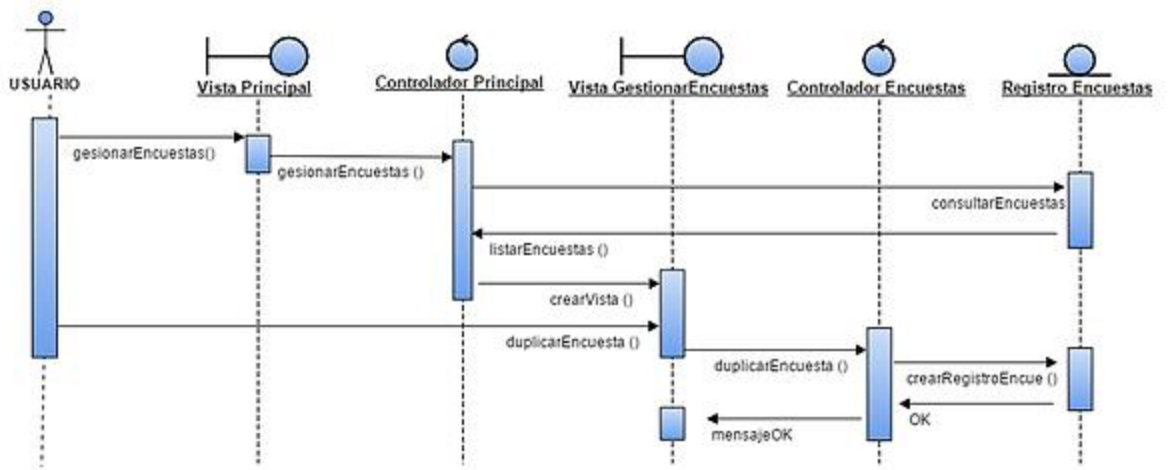


Figura 32. Diagrama de secuencia: Gestionar Encuestas-Duplicar encuesta

1. El Usuario solicita gestionar Encuestas en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta las encuestas en base de datos y despliega la vista VistaGestionarEncuestas.
3. El Usuario solicita duplicarEncuesta en una encuesta en la vista VistaGestionarEncuestas.
4. El controlador ControladorEncuesta recibe la petición de duplicarEncuesta, actualiza el registro de encuestas y devuelve un mensaje de ok en la vista VistaGestionarEncuestas

Figura 33. Diagrama de secuencia: Gestionar Encuestas-Responder Encuesta

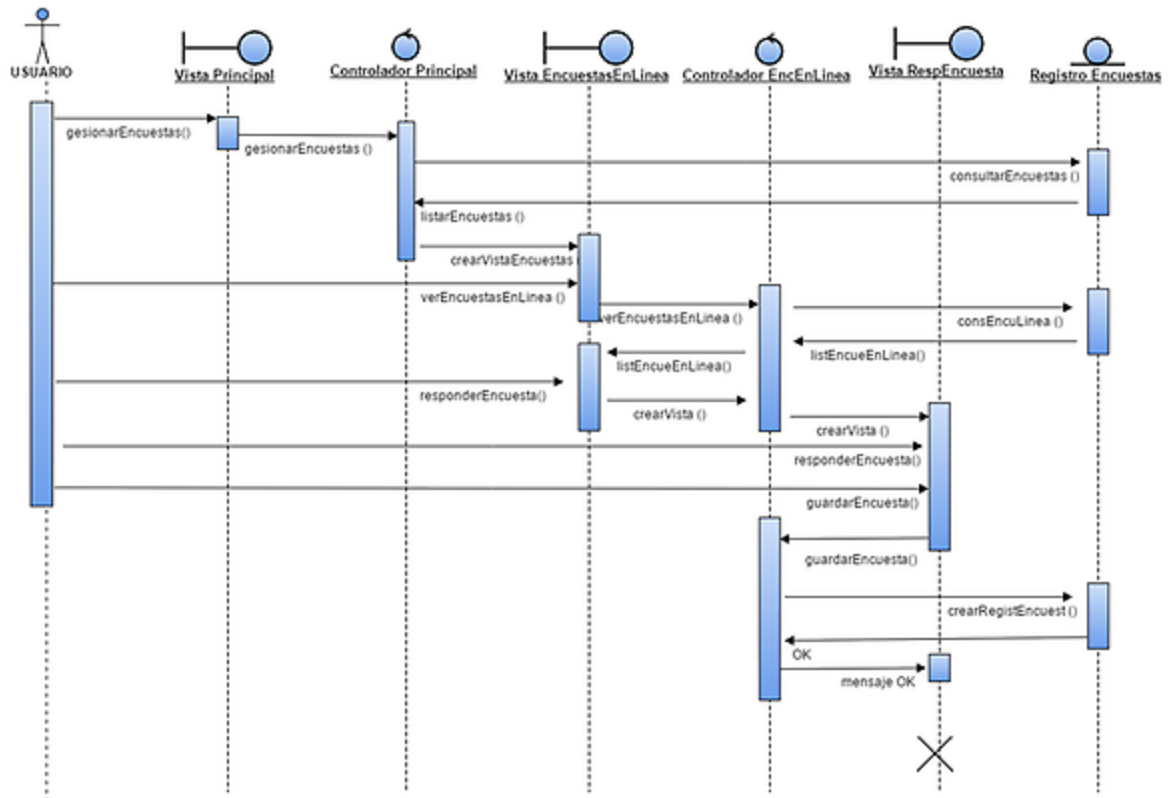


Figura 33. Diagrama de secuencia: Gestionar Encuestas-Responder Encuesta

1. El Usuario solicita gestionar Encuestas en la vista VistaPrincipal.
2. El controlador ControladorPrincipal recibe la petición, consulta las encuestas en base de datos y despliega la vista VistaGestionarEncuestas.
3. El Usuario solicita verEncuestaEnLinea en la vista VistaGestionarEncuestas y selecciona en cual encuesta que esté en linea participa.
4. El controlador ControladorEncuesta recibe la petición de responderEncuesta, y crea la vistaResponderEncuesta.
5. El usuario diligencia los campos y guarda la encuesta.

6. El controlador ControladorEncuesta recibe la petición de guardarEncuesta, actualiza el registro de encuestas y devuelve un mensaje de ok en la vista VistaResponderEncuestas

Servicio de Inscripciones

Figura 34. Diagrama de secuencia: Gestionar Inscripción-Crear Inscripción

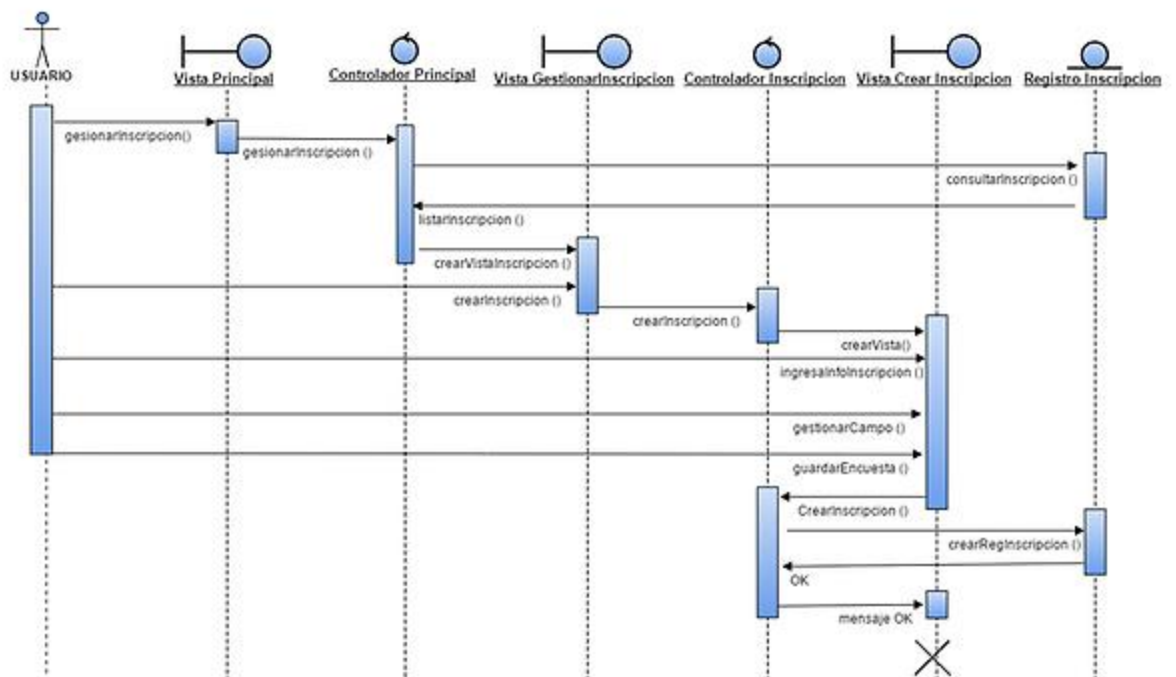


Figura 34. Diagrama de secuencia: Gestionar Inscripción-Crear Inscripción

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestionarInscripciones.
3. El Usuario solicita crear inscripción en la vista VistaGestiónInscripciones.
4. El controlador ControlInscripciones recibe la petición y despliega la vista VistaCrearInscripción.

5. El Usuario ingresa la información de la inscripción, agrega los campos que necesite y la guarda.
6. El controlador ControlInscripciones recibe la petición de guardar la inscripción, crea el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaCrearInscripción.
7. Se destruye la vista VistaCrearInscripción.

Figura 35. Diagrama de secuencia: Gestionar Inscripción-Editar Inscripción

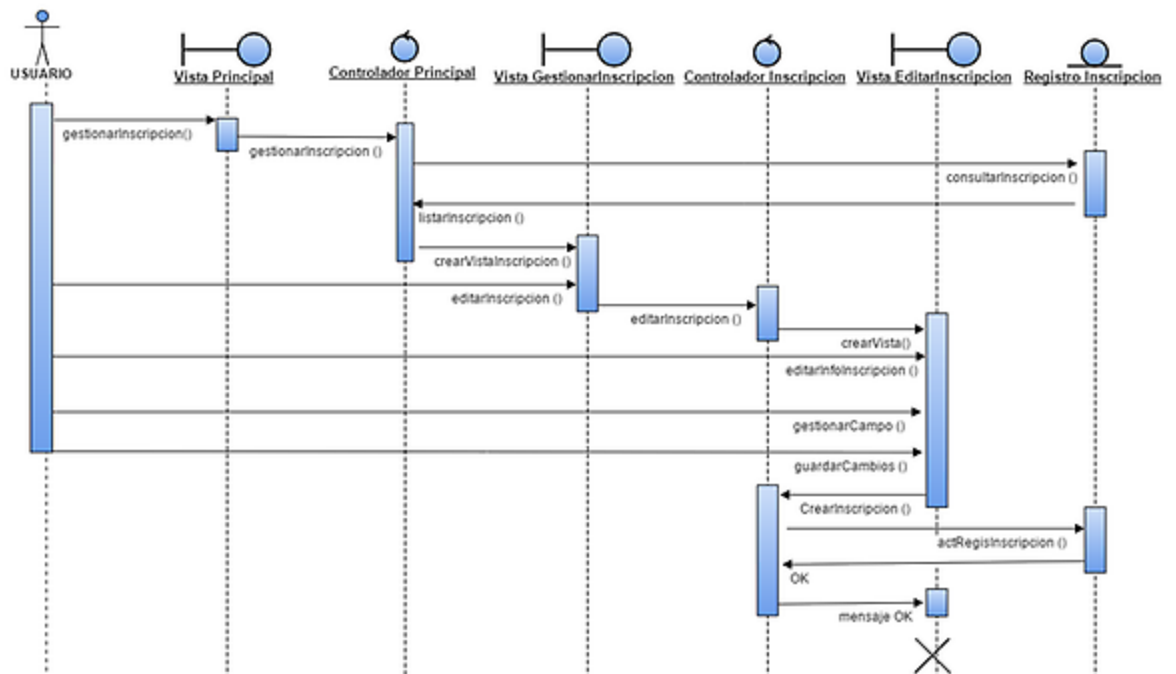


Figura 35. Diagrama de secuencia: Gestionar Inscripción-Editar Inscripción

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.

3. El Usuario solicita editar una inscripción en la vista VistaGestiónCronogramasAcadémicos.
4. El controlador ControlInscripciones recibe la petición y despliega la vista VistaEditarInscripcion.
5. El Usuario actualiza la información de la inscripción, agrega o elimina campos y la guarda.
6. El controlador ControlInscripciones recibe la petición de editar la inscripción, actualiza el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaEditarInscripción.
7. Se destruye la vista VistaEditarInscripción.

Figura 36. Diagrama de secuencia: Gestionar Inscripción-Duplicar Inscripción

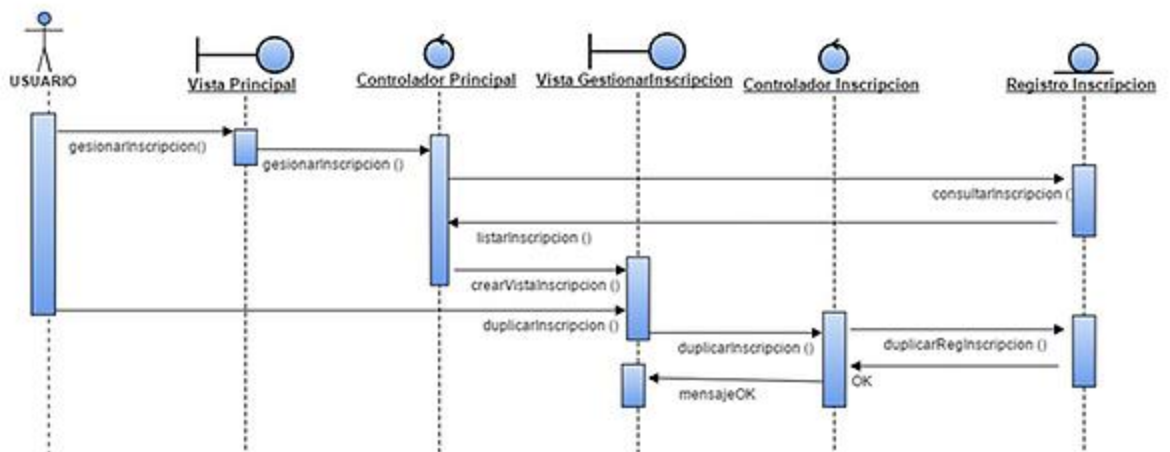


Figura 36. Diagrama de secuencia: Gestionar Inscripción-Duplicar Inscripción

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.

3. El Usuario solicita duplicar una inscripción en la vista VistaGestiónInscripciones.
4. El controlador Controllncripciones recibe la petición de duplicar la inscripción, duplica el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaGestiónInscripciones.

Figura 37. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Estado

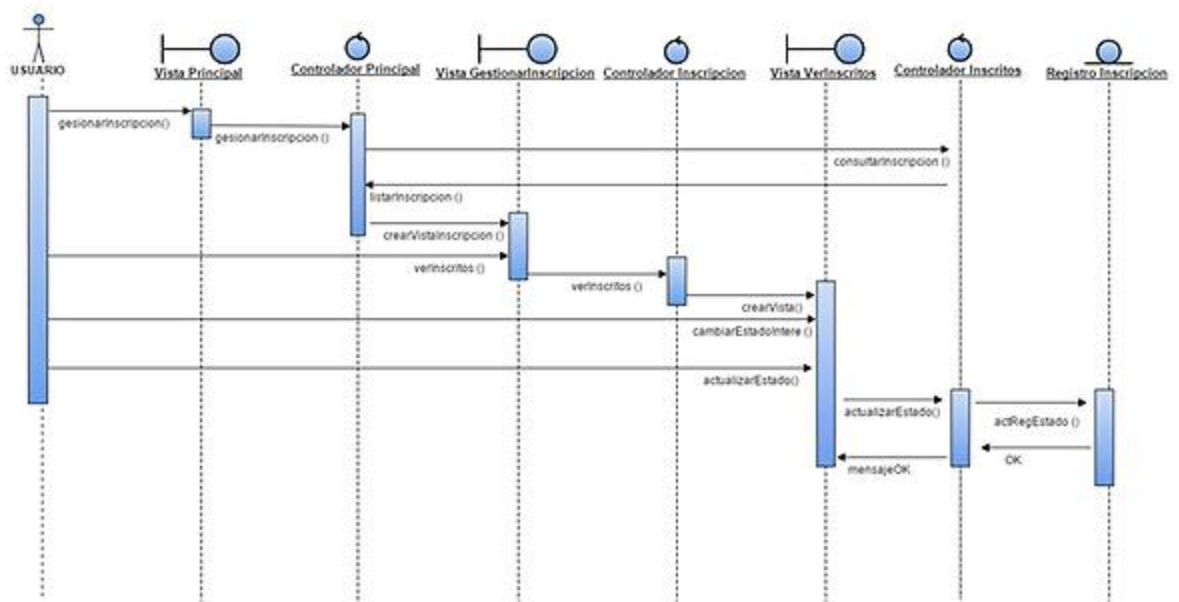


Figura 37. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Estado

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita ver los inscritos de una inscripción en la vista VistaGestiónInscripciones.
4. El controlador Controllncripciones recibe la petición de ver los inscritos de la inscripción y despliega la vista VistaVerInscritos.

5. El Usuario cambia el estado de los interesados que va a inscribir y envía los datos.
6. El controlador `ControlInscripciones` recibe la petición, actualiza los registros de los interesados en la base de datos y retorna un mensaje de Ok al usuario en la vista `VistaVerInscritos`.

Figura 38. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Enviar Correo

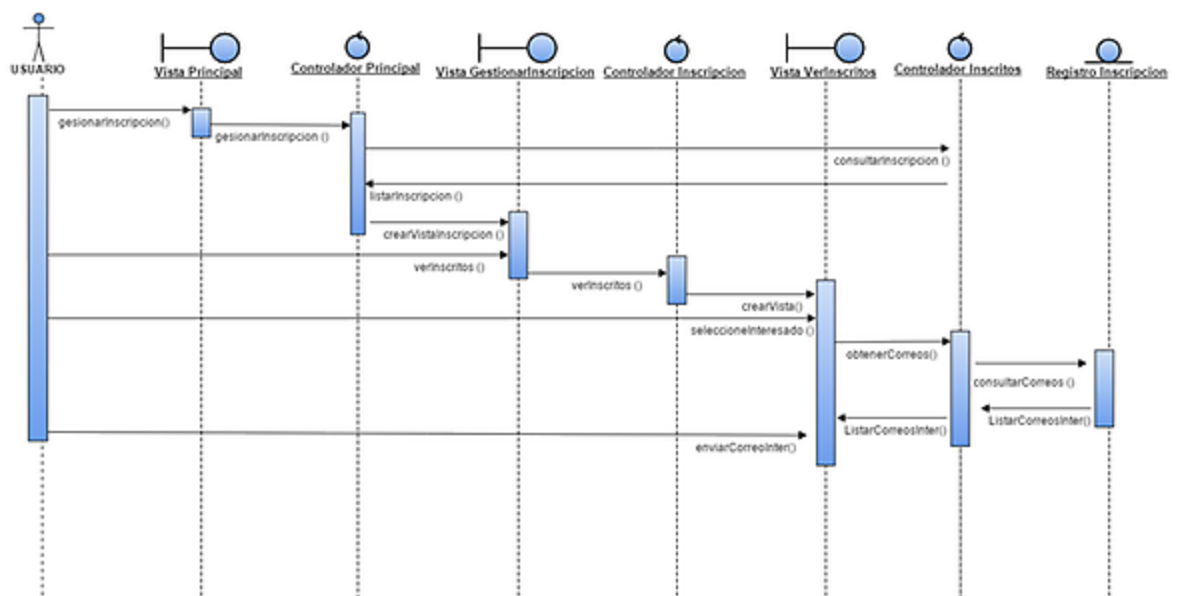


Figura 38. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Enviar Correo

1. El Usuario solicita gestionar inscripciones en la vista `VistaPrincipal`.
2. El controlador `ControlPrincipal` recibe la petición, consulta las inscripciones en base de datos y despliega la vista `VistaGestiónInscripciones`.
3. El Usuario solicita ver los inscritos de una inscripción en la vista `VistaGestiónInscripciones`.
4. El controlador `ControlInscripciones` recibe la petición de ver los inscritos de la inscripción y despliega la vista `VistaVerInscritos`.

5. El Usuario selecciona los interesados a los cuales desea enviarles un correo.
6. El controlador ControlInscripciones recibe la petición, consulta los correos de los interesados en la base de datos y retorna la consulta a la vista VistaVerInscritos.
7. El Usuario envía el correo.

Figura 39. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Información Interesado

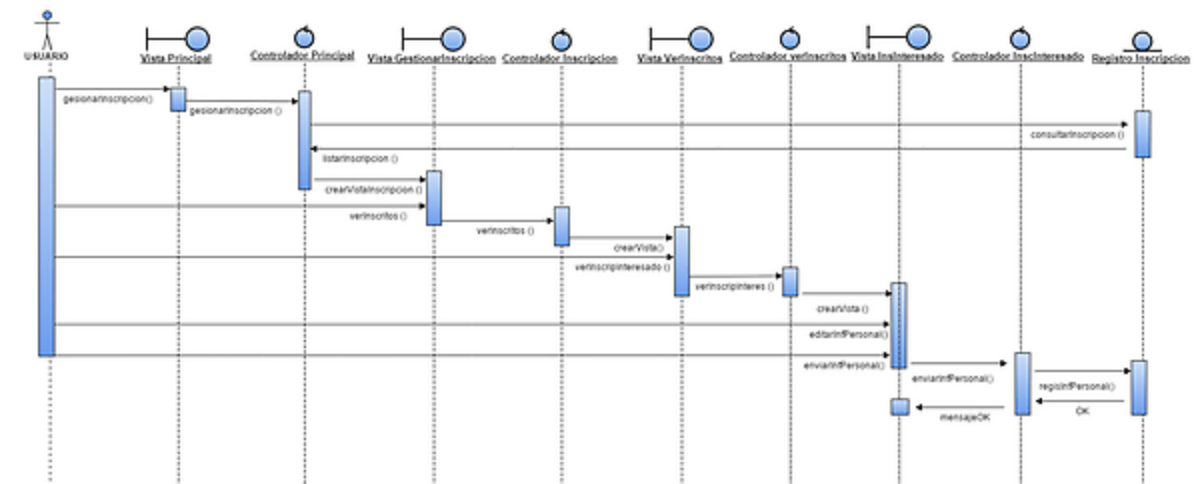


Figura 39. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Información Interesado

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita ver los inscritos de una inscripción en la vista VistaGestiónInscripciones.
4. El controlador ControlInscripciones recibe la petición y despliega la vista VistaVerInscritos.

5. El Usuario solicita ver la inscripción de un interesado en la vista VistaVerInscritos.
6. El controlador ControlVerInscritos recibe la petición y despliega la vista VistaInscripInteresado.
7. El Usuario edita la información personal o los campos de la inscripción del interesado y envía los datos.
8. El controlador ControlInscripInteresado recibe la petición, actualiza el registro del interesado en la base de datos y retorna un mensaje de Ok al usuario en la vista VistaInscripInteresado.

Figura 40. Diagrama de secuencia: Ver Inscripciones - ParticiparInscripcion

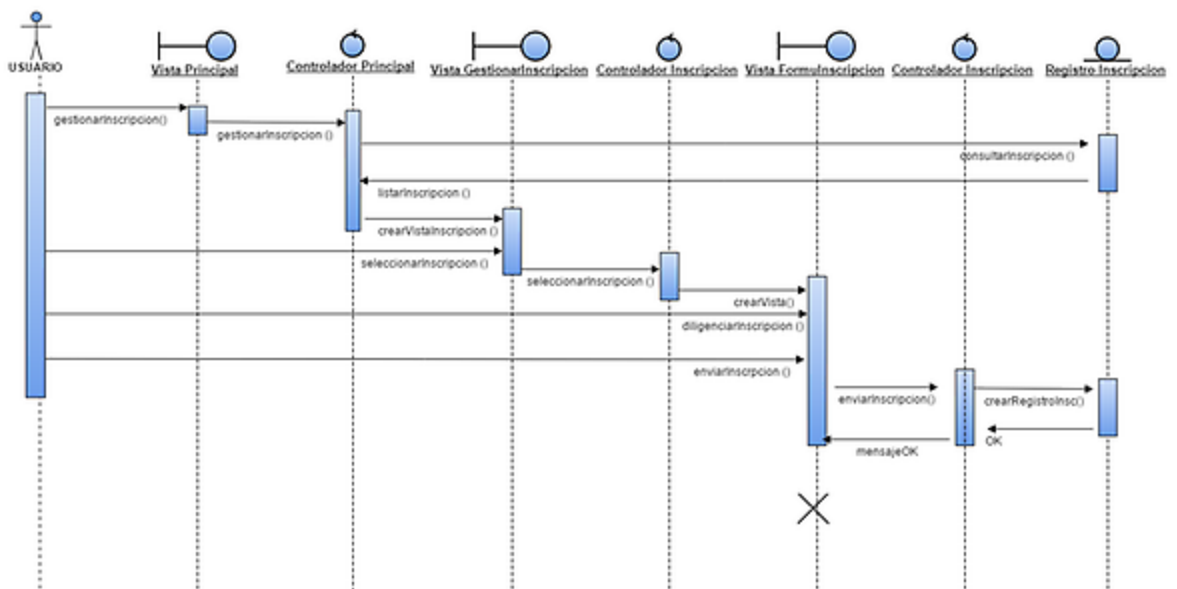


Figura 40. Diagrama de secuencia: Ver Inscripciones - ParticiparInscripcion

1. El Usuario solicita ver las inscripciones abiertas en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consultas las inscripciones en base de datos y despliega la vista VistaVerInscripciones.
3. El usuario selecciona la inscripción a la cual desea aplicar en la vista VistaVerInscripciones.

4. El controlador ControlInscripciones recibe la petición y despliega la vista VistaFormularioInscripción.
5. El usuario diligencia los datos de la inscripción y la envía.
6. El controlador ControlFormularioInscripción recibe la petición, crea el registro de la inscripción del interesado en la base de datos y retorna un mensaje de Ok al usuario en la vista VistaFormularioInscripción.

4.1.5 Implementación, Implantación y Pruebas Generales

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

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

Haciendo uso de las anteriores herramientas, y con la asesoría y seguimiento del director de proyecto se diseñaron los nuevos servicios y se estructuraron los

datos para el primer prototipo. También se tomaron en cuenta sugerencias recibidas por miembros del Grupo Calumet, para realizar un posterior refinamiento a las interfaces.

Para el desarrollo se trabajó con la base de datos "Diamante" ubicada en todos los servidores de los portales de las escuelas, y con la base de datos centralizada "Poseidón" ubicada en el servidor del portal web de la Vicerrectoría Académica. En las cuales se crearon las tablas necesarias para el funcionamiento de los servicios.

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

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

4.2 MANTENIMIENTO Y ADMINISTRACIÓN

4.2.1 Actividades de Mantenimiento

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

- Permitir más caracteres en el asunto de los correos

Funcionalidad:

Al enviar un correo el asunto debe tener el espacio suficiente para ser explicado.

Desarrollo del problema:

Inicialmente al enviar un correo solo se contaba con un espacio de 50 caracteres para el asunto, mediante su uso se detectó la falta de espacio al tratar de describir el asunto. Se dio más espacio a este requerimiento mediante un div con un diseño que permite más caracteres.

Problemática:

Permitir que un correo pueda tener más caracteres en el asunto.

Descripción:

Se cambió el div en eisi/Aula/Clase/EnviarCorreosEst.jsp y en eisi/Correos/envioCorreos.jsp

- Permitir que los editores tuvieran más opciones como subir archivos tipo flash, imagen y enlaces entre otras opciones.

Funcionalidad:

Al editar o publicar una noticia se debe poder subir archivos tipo flash, imagen o enlaces, que le editor le dé más funcionalidades al usuario.

Desarrollo del problema:

Inicialmente al publicar no se contaba con la opción de subir archivos, por lo cual se cambió el editor por uno que contara con estas opciones.

Problemática:

Permitir subir un archivo o una imagen.

Descripción:

Se cambió el editor por un ck-editor para que los editores tuvieran mas funcionalidad en eisi/Aula/Clase/EnviarCorreosEst.jsp y en eisi/Correos/envioCorreos.jsp

- Cambiar el mensaje de archivos adjuntos dando aviso del plazo con el que cuenta un usuario para descargar los archivos antes de ser eliminados.

Funcionalidad:

Al compartir un archivo se le debe avisar al usuario que cuenta con 30 días para descargarlo.

Desarrollo del problema:

Inicialmente al adjuntar un archivo, el archivo permanecía por vario tiempo pero por evitar la congestión en el servidor se validó que los archivos solo permanezcan 30 días y después de este plazo se eliminen, no sin antes haberle dado un aviso a su destinatario que esto sucederá al cabo de los treinta días.

Problemática:

El servidor no debe almacenar archivos por más de treinta días y el destinatario debe saber de esta nueva eventualidad.

Descripción:

Se cambió el mensaje a los destinatario en el jsp eisi/Aula/Clase/EnviarCorreosEst.jsp y en eisi/Correos/envioCorreos.jsp

- Creación de ayudas para el servicio de trabajos de grado: ¿Cómo inscribir el tema del proyecto de grado?, ¿Cómo inscribir una solicitud?, ¿Cómo avalar una solicitud? y ¿Cómo avalar un tema de proyecto de grado?

Funcionalidad:

El módulo de proyectos de grado permite a los estudiantes ingresar los datos para el formulario de inscripción de tema definido por cada escuela, permite ingresar los datos para las respectivas solicitudes además de avalar solicitudes y temas de proyecto de grado.

Desarrollo del problema:

Se implementó las ayudas para inscribir tema, inscribir solicitud, avalar solicitud y avalar un tema de proyecto de grado para facilitar el uso de las tecnologías a los estudiantes de forma clara y precisa.

Problemática:

Se requiere la creación de ayudas para los usuarios que les permita ingresar el tema o solicitudes además de avalar solicitudes y temas de una manera sencilla.

Descripción:

Se realizaron las ayudas para inscribir el tema del proyecto de grado, inscribir una solicitud, avalar una solicitud y avalar un tema de proyecto de grado.

4.2.2 Actividades de Soporte a Usuarios

Las escuelas cuentan en sus portales con un servicio de consultas y sugerencias, a través del cual se responden preguntas y se resuelven problemas de los usuarios. Diariamente se reciben consultas de estudiantes y profesores, que necesitan asesoría en el funcionamiento de servicios o soporte en el manejo de su usuario y contraseña. También se resuelven problemas de manera presencial, donde el usuario acude directamente a las oficinas del Grupo Calumet y es atendido directamente por alguno de sus miembros, que le prestan la asesoría necesaria. Algunas de las labores realizadas comúnmente son:

- Restablecimiento de contraseña a usuarios.
- Dar respuesta y solución a las inquietudes y problemas que tengan los usuarios de los portales respecto al uso de los servicios.
- 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 y el uso de sus principales servicios.

4.2.3 Actividades de Administración

Dentro de las actividades realizadas por los miembros del Grupo Calumet, se encuentra la tarea de administrar los servidores de los portales de las escuelas, cada semestre se asignan nuevos administradores, los cuales cuentan con un

usuario dentro de uno de los portales a los cuales presta soporte el grupo. Como administrador las tareas que se realizan comúnmente son:

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

5 PRUEBAS DEL SISTEMA

Para garantizar el correcto desarrollo de los nuevos servicios creados y a los que se le hizo reingeniería, se realizaron las siguientes pruebas:

5.1 PRUEBAS DE VERIFICACIÓN

Esta prueba es una de las más utilizadas en desarrollo de software mediante esta se aplican diferentes técnicas para detectar errores en el sistema antes de ser utilizado.

Se efectúa ejecutando paso a paso el proceso del servicio de manera que se explora cada funcionalidad que tiene el módulo desarrollado, realizando verificaciones de validación, los campos que son obligatorios no pueden quedar vacíos, por ejemplo.

Las siguientes tablas describen las pruebas de cada caso de uso de los servicios que fueron desarrollados.

5.1.1 Pruebas por componente

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

5.1.1.1 Servicio de Inscripciones

Tabla 8. Pruebas Realizadas: Inscripciones.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Inscripciones	Al acceder al servicio de Inscripciones, lista correctamente las inscripciones creadas por el usuario.	✓
Crear Inscripción	Al crear una inscripción, valida que se diligencien todos los campos obligatorios y almacena correctamente.	✓
Editar Inscripción	Al editar la inscripción, almacena las actualizaciones correctamente.	✓

Tabla 8. Pruebas Realizadas: Inscripciones. (Continuación)

Eliminar Inscripción	Elimina la inscripción y los registro de interesados e inscritos asociados a ella.	✓
Duplicar Inscripción	Se crea una inscripción igual a la anterior con estado inactivo y diferente id.	✓
Gestionar Campo	Se muestran y funcionan correctamente las opciones de editar, eliminar, subir y bajar campo.	✓
Agregar Campo	Se agrega correctamente campos estándar y personalizados como tipo texto, párrafo, selección múltiple, entre otros.	✓
Editar Campo	Actualiza y guarda correctamente los cambios en los campos modificados.	✓
Eliminar Campo	Elimina el registro del campo eliminado.	✓
Ver Inscritos	Muestra correctamente las pestañas de los interesados, pre-inscritos e inscritos.	✓
Actualizar Estado	Actualiza correctamente los cambios de estado de los interesados.	✓
Eliminar Interesado	Elimina el registro del interesado y demás registros asociados a él.	✓
Ver Respuestas	Muestra correctamente las respuestas de los interesados.	✓
Actualizar Información	Actualiza y guarda correctamente la información y respuestas del interesado.	✓
Ver Proceso De Requisitos	Muestra correctamente el proceso de los requisitos que lleva un interesado.	✓
Gestionar Requerimientos	Muestra correctamente los requisitos que ya han sido agregados a una inscripción.	✓
Crear Requisitos	Agrega correctamente los requisitos creados por el usuario.	✓
Editar Requisitos	Actualiza y guarda correctamente los requisitos modificados por el usuario.	✓

Tabla 8. Pruebas Realizadas: Inscripciones. (Continuación)

Eliminar Requisitos	Elimina correctamente el requisito no deseado por el usuario.	✓
Enviar Correos	Envía correctamente los correos a los interesados.	✓
Participar Inscripción	Muestra correctamente el listado de inscripciones que puede responder determinado usuario y entra correctamente al formulario de inscripción.	✓
Ver Proceso	Muestra el proceso de requisitos cumplidos y faltantes de determinado usuario.	✓
Modificar Respuestas	Actualiza y guarda correctamente las respuestas modificadas por el usuario.	✓
Subir Archivos	Sube y guarda correctamente los archivos del usuario.	✓

*Tabla 8. Pruebas Realizadas: Inscripciones***5.1.1.2 Servicio de Encuestas****Tabla 9. Pruebas Realizadas: Encuestas**

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Encuesta	Muestra correctamente las encuestas en línea, fuera de línea y vencidas.	✓
Crear Encuesta	Crea y guarda correctamente una encuesta.	✓
Editar Encuesta	Actualiza y guarda correctamente una encuesta editada.	✓
Eliminar Encuesta	Elimina correctamente una encuesta no deseada.	✓
Duplicar Encuesta	Se crea una encuesta igual a la anterior con estado inactivo y diferente id.	✓
Ver Detalles	Muestra correctamente los detalles de una encuesta seleccionada.	✓
Ver Resultados	Muestra correctamente los resultados de una encuesta seleccionada que contenga respuestas.	✓
Ver Encuesta En Línea	Muestra correctamente un listado de las encuestas en las que puede participar determinado usuario.	✓

Tabla 9. Pruebas Realizadas: Encuestas (continuación).

Responder Encuesta	Guarda correctamente las respuestas del usuario.	✓
--------------------	--	---

Tabla 9. Pruebas Realizadas: Encuestas

5.1.1.3 Servicio de Noticias

Tabla 10. Pruebas Realizadas: Noticias

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Administrar Mis Noticias	Muestra correctamente mis noticias en línea y fuera de línea.	✓
Editar Noticia	Actualiza y guarda correctamente una noticia editada.	✓
Eliminar Noticia	Elimina correctamente una noticia no deseada.	✓
Crear Noticia	Crea y guarda correctamente una noticia.	✓
Administrar Noticias	Muestra correctamente las noticias de todos los usuarios, en línea y fuera de línea.	✓
Ver Titulares	Muestra correctamente los titulares de las noticias que se encuentran activas.	✓
Ver Noticia	Muestra correctamente la noticia completa que se ha seleccionado.	✓
Calificar Noticia	Guarda correctamente la calificación que un usuario le da a una noticia.	✓

Tabla 10. Pruebas Realizadas: Noticias

5.1.1.4 Servicio de Proyectos de extensión

Tabla 11. Pruebas Realizadas: Proyectos de extensión

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Proyecto	Muestra correctamente una lista de los proyectos activos, inactivos y detenidos.	✓
Crear Proyecto	Crea y guarda correctamente un proyecto.	✓
Editar Proyecto	Actualiza y guarda correctamente un proyecto editado.	✓
Eliminar Proyecto	Elimina correctamente un proyecto no deseado.	✓

Tabla 11. Pruebas Realizadas: Proyectos de extensión (continuación).

Ver Detalles	Muestra correctamente los detalles de un proyecto seleccionado.	✓
Participar Actividad	Guarda correctamente los comentarios y archivos con los que un usuario participa.	✓
Finalizar Actividad	Cambia y actualiza correctamente el cambio de estado de la actividad seleccionada.	✓
Reactivar Actividad	Cambia y actualiza correctamente el cambio de estado de la actividad seleccionada.	✓
Iniciar Proyecto	Cambia y actualiza correctamente el cambio de estado del proyecto seleccionado.	✓
Gestionar Etapas	Muestra correctamente una lista de las etapas asociadas a un proyecto.	✓
Crear Etapa	Crea y guarda correctamente una etapa.	✓
Editar Etapa	Actualiza y guarda correctamente una etapa editada.	✓
Eliminar Etapa	Elimina correctamente una etapa no deseada, verificando que antes se hayan eliminado las actividades que tenga asociadas.	✓
Gestionar Actividades	Muestra correctamente una lista de las actividades asociadas a una etapa.	✓
Crear Actividad	Crea y guarda correctamente una actividad.	✓
Editar Actividad	Actualiza y guarda correctamente una actividad editada.	✓
Eliminar Actividad	Elimina correctamente una actividad no deseada.	✓

*Tabla 11. Pruebas Realizadas: Proyectos de extensión***Tabla 12. Pruebas Realizadas: Ejes de gestión**

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Ejes	Muestra correctamente una lista de los ejes activos e inactivos.	✓
Crear Eje	Crea y guarda correctamente un eje.	✓
Eliminar Eje	Elimina correctamente un eje no deseado.	✓
Editar Eje	Actualiza y guarda correctamente un eje editado.	✓
Ver Detalles	Muestra correctamente los detalles de un eje seleccionado.	✓

Tabla 12. Pruebas Realizadas: Ejes de gestión

6. CONCLUSIONES

- La reingeniería del servicio encuestas permite a los usuarios y las escuelas realizar formularios personalizados de manera sencilla y con un diseño más claro y usable al anterior diseño. Sin usar servicios externos o aplicaciones puede crear, editar y activar estas encuestas para obtener respuestas de usuarios de la misma escuela y de otras de la comunidad académica.
- Las mejoras al módulo gestor de proyectos permiten una navegabilidad del servicio más cómoda y sencilla, además de un diseño más amigable con el usuario. Se agrega funcionalidades que hacen que el servicio se haga más flexible, como por ejemplo se permite que las actividades que se hayan finalizado se puedan reactivar nuevamente.
- El desarrollo de un nuevo cronograma para los proyectos de extensión permite que los usuarios tengan una vista de las actividades de los proyectos, agrupadas por etapas, y además presentadas por colores dependiendo del estado de las mismas, otorgando una mejor experiencia de usuario.
- El servicio de Inscripciones UIS ofrece a las escuelas y los usuarios la posibilidad de crear formularios personalizados de manera sencilla, sin la necesidad de usar servicios externos, y desde el cual puede obtener interesados de la misma escuela o de otras escuelas de la comunidad académica. A través de estas se podrán dar a conocer los programas académicos y eventos de las escuelas, y los administrativos contarán con una herramienta sencilla para contactar a todos aquellos que estén interesados en participar, además una opción para hacer seguimiento de los requerimientos de la inscripción.

- El módulo de noticias brinda una vista amigable a los usuarios y la oportunidad de que este de una calificación a la noticia. Se crea este servicio para los grupos de manera que ahora los grupos tienen la opción de publicar sus noticias para el grupo, para la escuela y para otras escuelas de la comunidad académica.

- La orientación a los nuevos integrantes del grupo calumet, su familiarización con las herramientas utilizadas en el mantenimiento de los portales y la dinámica de trabajo de los miembros de éste, permiten la continuidad de la labor realizada por el grupo, en pro de una mejora continua de los servicios.

- La pronta atención a las sugerencias de los usuarios permite dar también una oportuna respuesta de manera que se logre dar rápido y eficaz soporte al software manteniendo los servicios actualizados y con continuas mejoras en su funcionalidad.

7. RECOMENDACIONES

- Se sugiere agregar al servicio de encuestas la funcionalidad de permitir que las preguntas tengan preguntas hijas, es decir, que haya preguntas que sean dependientes de otras, de manera que las dependientes aparecen en el formulario de encuesta o no, dependiendo de la respuesta de la pregunta padre.
- Permitir otros tipos de preguntas en el formulario de encuesta, extras a las preguntas estándar y personalizadas que existen actualmente, como por ejemplo, preguntas con imágenes y preguntas tipo tabla; agregar también otras nuevas opciones para el formulario de inscripciones.
- Permitir a los usuarios ver estadísticas de las noticias que publican, es decir contar con datos de visitas e impacto de una noticia con el fin de ofrecer una vista donde puedan percibir la acogida que ha tenido dicha noticia por la comunidad académica.
- Permitir que los usuarios comenten en una noticia.

BIBLIOGRAFÍA

1. 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.
2. 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.
3. 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.
4. 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.
5. STUMPF, Robert, TEAGUE, Lavette. Object-Oriented Systems Analysis and Design with UML. Prentice Hall. 2004. Este libro introduce los conceptos y métodos del análisis y diseño de sistemas orientados a objetos.
6. WEITZENFELD, Alfredo. Ingeniería de Software Orientada a Objetos con UML, JAVA e Internet. Thomson International, 2003. En este libro se encuentra información sobre desarrollo de software orientado a objetos.

7. <http://g701giadar.wikispaces.com/Arquitctura+Cliente+Servidor>. Información acerca de la Arquitectura Cliente Servidor.
8. <http://manuales.dgsca.unam.mx/jsp>. Manuales básicos de JSP.
9. <http://www.aprender-informatica.com/>. (n.d). Tutoriales del lenguaje de programación JavaScript.
10. <http://www.htmlquick.com/es/tutorials.html>, tutoriales de cómo programar en html de gran ayuda en detalles básicos y sencillos.
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.