

ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE UNA NUEVA ACTIVIDAD  
EVALUATIVA DE QUICES EN LA SECCIÓN DE AULA VIRTUAL Y SERVICIOS  
RELACIONADOS CON LA GESTIÓN DE TRABAJOS DE GRADO EN LA  
PLATAFORMA WEB COMUNIDAD ACADÉMICA COMA

DIEGO ARMANDO VILLAMIZAR CORREA

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

2021

ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE UNA NUEVA ACTIVIDAD  
EVALUATIVA DE QUICES EN LA SECCIÓN DE AULA VIRTUAL Y SERVICIOS  
RELACIONADOS CON LA GESTIÓN DE TRABAJOS DE GRADO EN LA  
PLATAFORMA WEB COMUNIDAD ACADÉMICA COMA

DIEGO ARMANDO VILLAMIZAR CORREA

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 FISICOMECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA

2021

## **AGRADECIMIENTOS**

A mis padres Jorge Helí Villamizar y Yaneth Correa Rueda por todo el apoyo y esfuerzo que hicieron durante mi proceso de formación.

Al profesor Luis Ignacio por su orientación constante durante la realización de este proyecto de grado.

## CONTENIDO

	pág.
INTRODUCCIÓN .....	11
1 PLANTEAMIENTO DEL PROYECTO.....	12
1.1 DEFINICIÓN DE LA SITUACIÓN ACTUAL .....	12
1.2 JUSTIFICACIÓN.....	13
1.3 OBJETIVOS.....	14
1.4 IMPACTO Y VIABILIDAD .....	15
2. MARCO TEÓRICO .....	16
2.1 ARQUITECTURA CLIENTE/SERVIDOR.....	16
2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS.....	20
2.3 BASES DE DATOS.....	24
2.4 NETBEANS .....	27
2.5 SISTEMA DE CONTROL DE VERSIONES.....	27
2.6 PROGRAMACIÓN UTILIZADA.....	29
2.7 SERVIDORES WEB .....	31
3. MARCO METODOLÓGICO.....	33
3.1 MODELO DE PROTOTIPO EVOLUTIVO.....	33
3.2 LENGUAJE DE MODELADO UNIFICADO.....	34
3.3 ESTÁNDARES DE PROGRAMACIÓN.....	38
4. DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO.....	41
4.1 PROTOTIPO ESPERADO.....	41
4.2 DIAGRAMAS DE CASOS DE USO .....	44
4.3 DOCUMENTACIÓN DE CASOS DE USO DEL SISTEMA.....	50
4.4 DISEÑO Y ANÁLISIS.....	54

4.5 DESCRIPCIÓN DE LAS ENTIDADES.....	60
4.6 MODELO DE PROCESOS DEL SISTEMA .....	62
4.7 IMPLEMENTACIÓN, IMPLANTACIÓN Y PRUEBAS GENERALES.....	63
4.8 INTERFACES DE USUARIO DESARROLLADAS.....	64
4.9 MANTENIMIENTO Y ADMINISTRACIÓN .....	74
5. PRUEBAS DEL SISTEMA .....	77
5.1 PRUEBAS DE VERIFICACIÓN .....	77
6. CONCLUSIONES .....	84
BIBLIOGRAFÍA.....	86
ANEXOS.....	88

## LISTA DE TABLAS

Tabla 1. Casos de uso: Servicio de quices .....	50
Tabla 2. Casos de uso: Servicio formato de plan E3T .....	51
Tabla 3. Casos de uso: Servicio de indicadores .....	52
Tabla 4. Casos de uso: Servicio de gestión de archivo final .....	53
Tabla 5. Casos de uso: Asignar hasta 2 evaluadores a un trabajo de grado .....	53
Tabla 6. Casos de uso: Asignar hasta 6 miembros al comité de trabajos de grado .....	54
Tabla 7. Descripción de las entidades .....	60
Tabla 8. Pruebas realizadas: Servicio de quices .....	77
Tabla 9. Pruebas realizadas: Servicio formato de plan E3T .....	79
Tabla 10. Pruebas realizadas: Servicio de indicadores .....	80
Tabla 11. Pruebas realizadas: Servicio de gestión de archivo final .....	81
Tabla 12. Pruebas realizadas: Asignar hasta 2 evaluadores aun trabajo de grado .....	82
Tabla 13. Pruebas realizadas: Asignar hasta 6 miembros al comité de trabajos de grado .....	83

## LISTA DE FIGURAS

	pág.
Figura 1. Modelo Cliente/Servidor.....	16
Figura 2. Modelo de acceso a JSP .....	23
Figura 3. Ciclo de vida modelo de prototipos evolutivos .....	33
Figura 4. Ejemplo diagrama de casos de uso .....	36
Figura 5. Ejemplo diagrama de secuencias .....	38
Figura 6. Diagrama de casos de uso: Servicio de quices .....	45
Figura 7. Diagrama de casos de uso: Servicio formato plan E3T .....	45
Figura 8. Diagrama de casos de uso: Servicio de indicadores .....	47
Figura 9. Diagrama de casos de uso: Servicio de gestión de archivo final .....	48
Figura 10. Diagrama de casos de uso: Asignar hasta 2 evaluadores a un trabajo de grado.....	49
Figura 11. Diagrama de casos de uso: Asignar hasta 6 miembros del comité de trabajos de grado.....	50
Figura 12. Diagrama E/R: Servicio de quices .....	55
Figura 13. Diagrama E/R: Servicio formato de plan E3T .....	56
Figura 14. Diagrama E/R: Servicio de indicadores .....	57
Figura 15. Diagrama E/R: Servicio de gestión de archivo final .....	58
Figura 16. Diagrama E/R: Asignar hasta 2 evaluadores a un trabajo de grado .....	59
Figura 17. Diagrama E/R: Asignar hasta 6 miembros al comité de trabajos de grado .....	60
Figura 18 . Interfaz de usuario (profesor – Vista general): Servicio de quices .....	64
Figura 19. Interfaz de usuario (profesor – Agregar quiz): Servicio de quices .....	65
Figura 20. Interfaz de usuario (profesor – Modificar preguntas): Servicio de quices .....	66
Figura 21. Interfaz de usuario (estudiante): Servicio de quices .....	67
Figura 22. Interfaz de usuario (admin): Servicio de formato plan E3T .....	68
Figura 23. Interfaz de usuario (evaluador): Servicio de formato de plan E3T .....	69

Figura 24. Interfaz de usuario (estudiante): Servicio de formato de plan E3T .....	69
Figura 25. Interfaz de usuario (cliente): Servicio de indicadores.....	70
Figura 26. Interfaz de usuario (admin): Servicio de indicadores .....	71
Figura 27. Interfaz de usuario: Servicio de gestión del archivo final .....	72
Figura 28. Interfaz de usuario: Asignar 2 evaluadores a un trabajo de grado.....	73
Figura 29. Interfaz de usuario: Asignar hasta 6 miembros al comité de trabajos de grado .....	74
Figura 30. Diagrama de secuencia: Servicio de quices .....	88
Figura 31. Diagrama de secuencia: Servicio formato de plan.....	90
Figura 32. Diagrama de secuencia: Servicio de indicadores .....	93
Figura 33. Diagrama de secuencia: Gestión de archivo final.....	96

## RESUMEN

**TÍTULO:** ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE UNA NUEVA ACTIVIDAD EVALUATIVA DE QUICES EN LA SECCIÓN DE AULA VIRTUAL Y SERVICIOS RELACIONADOS CON LA GESTIÓN DE TRABAJOS DE GRADO EN LA PLATAFORMA WEB COMUNIDAD ACADÉMICA COMA \*

**AUTOR:** DIEGO ARMANDO VILLAMIZAR CORREA\*\*

**PALABRAS CLAVE:** PORTAL WEB, QUICES, COMUNIDAD, COMA, INDICADORES.

## DESCRIPCIÓN

Las escuelas actualmente cuentan con un sistema de información orientado a la Web que se encarga de la administración y control de las diferentes actividades tanto académicas, como administrativas que se realizan dentro de las escuelas y las facultades, así como del control de usuarios y servicios que se les proporcionan.

La plataforma COMA busca proveer una gran variedad de servicios a las escuelas de la Universidad Industrial de Santander de tal forma que estas puedan apoyarse en esta y mejorar su proceso de enseñanza por parte de los docentes, aprendizaje por parte de los estudiantes y por último sus procesos internos de gestión para hacerlos más eficientes.

Actualmente el portal web presta sus servicios a más de treinta escuelas y debido a las necesidades particulares de cada escuela se requiere nuevos servicios para mejorar sus procesos de comunicación de información (como un servicio de indicadores), proveer más herramientas para la evaluación de sus estudiantes de manera rápida como lo es un servicio de quices, proveer un servicio completo e íntegro de trabajos de grado buscando que optimice el proceso actual (vía correo electrónico) además de mejorar la comunicación entre el estudiante con sus directores y calificadores de trabajo de grado.

---

\* Trabajo de grado

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

## ABSTRACT

**TITLE:** ANALYSIS, DEVELOPMENT AND IMPLEMENTATION OF A NEW QUICES EVALUATIVE ACTIVITIES IN THE VIRTUAL CLASSROOM SECTION AND SERVICES RELATED TO THE MANAGEMENT OF DEGREE WORK ON THE COMA ACADEMIC COMMUNITY WEB PLATFORM\*

**AUTHOR:** DIEGO ARMANDO VILLAMIZAR CORREA\*\*

**KEY WORDS:** WEB PORTAL, QUIZZES, COMMUNITY, COMA, INDICATORS.

## DESCRIPTION

The schools currently have a Web-oriented information system that is responsible for the administration and control of the different academic and administrative activities that are carried out within the schools and the Faculties, as well as the control of users and services that are provided to them.

The COMA platform seeks to provide a wide variety of services to the schools of the Universidad Industrial de Santander in such a way that they can rely on it and improve their teaching process by teachers, learning by students and finally their internal management processes to make them more efficient.

Currently, the web portal provides its services to more than thirty schools and due to the needs of each school, new services are required to improve their information communication processes (such as an indicator service), provide more tools for the evaluation of their students. in a fast way as it is a service of quices, to provide a complete and integral service of degree projects seeking to optimize the current process (via email) in addition to improving communication between the student with their directors and qualifiers of degree work.

---

\* Degree work

\*\* Faculty of Physico-Mechanical Engineering. Department of System Engineering and Computing Science. Supervisor: Msc. 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 a las necesidades del momento. 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, JavaScript, jQuery, entre otras.

El proyecto insignia para responder a las necesidades de las escuelas por parte del grupo CALUMET es el proyecto comunidad académica (COMA) el cual es el conjunto de código, bases de datos y demás que se replican en los servidores de cada escuela para que así estas puedan gestionarlo con su propia información y hacer uso de los servicios que provee.

En este documento se presenta un soporte teórico, metodológico y técnico del desarrollo en las secciones de: trabajo de grado proveyendo servicios de gestión y visualización del formato de plan junto con la subida del archivo final en pregrado y posgrado, grupos implementando un servicio de creación y visualización de gráficos (indicadores) y en la sección de aula virtual añadiendo un servicio para hacer una nueva actividad evaluativa (quices).

# 1 PLANTEAMIENTO DEL PROYECTO

## 1.1 DEFINICIÓN DE LA SITUACIÓN ACTUAL

En la actualidad las escuelas y las facultades de la UIS, cuentan con un sistema de información orientado a la web que se encarga de la administración y control de las diferentes actividades tanto académicas como administrativas que se realizan dentro de la escuela y la facultad, así como del control de usuarios y servicios que se les proporcionan.

Los servicios de los portales web 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 y actualización se hacen indispensables.

La sección de trabajos de grado de la plataforma COMA necesita de nuevos servicios que mejoren la interacción entre los usuarios y facilite su uso, estos son: una interfaz de gestión para poder subir un formato del plan y también un servicio para que el estudiante pueda subir su trabajo de grado final para mejorar la comunicación entre el calificador y el estudiante. También en la sección de aula virtual requiere un nuevo servicio que permita realizar una actividad evaluativa de quiz y así los profesores tengan más herramientas para evaluar a sus estudiantes. Por último, en la sección de grupos actualmente no existe menú para poder crear gráficos y poder así representar datos en estos.

## 1.2 JUSTIFICACIÓN

Actualmente el grupo CALUMET ha cubierto diversas necesidades de la comunidad académica a través de sus portales web, sin embargo, esta herramienta continúa en crecimiento conforme los profesores y administrativos de la UIS noten la necesidad de digitalizar sus servicios y/o procesos de gestión al ver las ventajas que esto provee. Teniendo en cuenta esto los requerimientos demuestran su relevancia dependiendo del usuario que plantea su necesidad al grupo CALUMET:

A petición de varios profesores que hacen uso del aula virtual:

- Que los profesores puedan realizar quices en el aula virtual de aprendizaje haciendo uso del banco de preguntas. Esto para poder disponer de un mecanismo mucho más rápido (que el de evaluaciones) para evaluar conceptos aprendidos durante las clases.

A petición mayormente de la escuela de petróleos se requiere:

- Un nuevo servicio donde se pueda anexar archivos con el formato del plan a los evaluadores ya que actualmente este proceso se hace vía correo electrónico, por lo que se busca que la plataforma COMA supla este servicio para así hacer este proceso más eficiente.
- Nuevo servicio donde se muestre el plan a los calificadores en trabajos de grado. Esto para que el calificador tenga a la mano la información necesaria para definir su respuesta.

A petición mayormente del observatorio de mujeres y equidad de género del departamento de Santander OMEGS

- Nuevo servicio en la sección de grupos que permita la representación de datos haciendo uso de gráficos de varios tipos (barras, tortas, etc). Esto para mejorar la presentación visual de estos datos y logrando así una comprensión más rápida de estos.

## **1.3 OBJETIVOS**

**1.3.1 Objetivo general.** Analizar, diseñar, desarrollar e implementar nuevos servicios al portal web de las escuelas en la sección de trabajos de grado y aula virtual con el fin de proveer y facilitar el trámite de los trabajos de grado y la actividad académica de quices relacionada con el aula virtual.

### **1.3.1 Objetivos específicos**

1. Implementar la actividad evaluativa de quices en el aula virtual de tal manera que no se vea afectada por la semana actual, sino que se pueda desarrollar en cualquier momento, además debe tener la opción de usar las preguntas del banco de preguntas para crear su contenido.
2. Agregar al portal de las escuelas en la sección de trámites de grado un nuevo servicio donde los evaluadores de un proyecto de grado puedan acceder al formato del plan creado por el E3T.
3. Implementar un servicio nuevo para que a los calificadores de un proyecto de grado se les muestre un link hacia el proyecto de grado que están calificando en la sección donde ellos emiten nota.
4. En el módulo de trabajos de grado de pregrado y posgrado, permitir asignar dos evaluadores a un proyecto de grado y no solo uno.
5. En el módulo de trabajos de grado, permitir aumentar el número de miembros del comité a 6 miembros, tanto en pregrado como en posgrado.

6. En el módulo de trabajos de grado, permitir al estudiante con trabajo de grado poder subir el archivo final y que después de un tiempo de aprobado este archivo se elimine automáticamente del servidor.
7. En la sección de grupos (sección usada por grupos de investigación, centros de estudio, etc.) de la plataforma COMA, crear un nuevo servicio conocido como indicadores que permita la creación, modificación y eliminación de gráficos de barras, líneas, tortas y columnas para que así estos grupos puedan representar datos de una mejor forma.

## **1.4 IMPACTO Y VIABILIDAD**

**1.4.1 Impacto.** Gracias a la utilidad y eficacia de estos portales para suplir necesidades de las diferentes escuelas; además de ser una herramienta útil para el manejo de la información es necesario realizar labores de administración y mantenimiento para ofrecer al usuario un sitio más confiable.

Se pretende que los procesos que se realizan en las escuelas cada día sean más ágiles, dinámicos, seguros y eficientes, permitiendo mejorar y optimizar el proceso de enseñanza y aprendizaje por parte de los estudiantes y profesores.

**1.4.1 Viabilidad.** Implementar y mantener la plataforma web COMA es viable ya que el software que se usa es de libre distribución, los cambios planteados en este trabajo de grado no acarrearán un impacto desmedido en el rendimiento actual de esta plataforma por lo que una actualización o mejora de recursos del servidor no es necesaria; es igualmente viable a nivel intelectual gracias al programa de capacitaciones y continua mejora del proceso de desarrollo de software en el grupo CALUMET.

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.

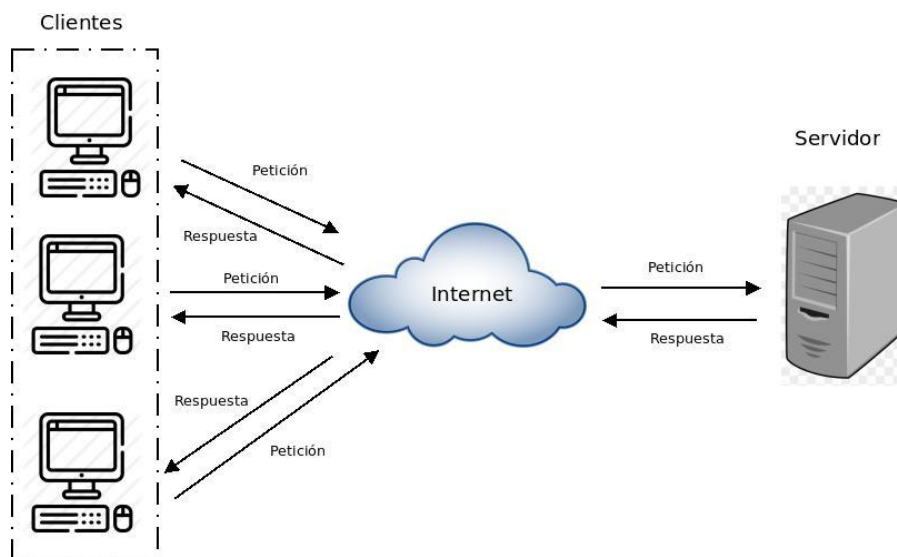
## 2. MARCO TEÓRICO

### 2.1 ARQUITECTURA CLIENTE/SERVIDOR

La arquitectura cliente/servidor se caracteriza por el hecho de que cada uno de los clientes produce un mensaje solicitando un determinado servicio a un servidor (hace una petición) estos envían uno o varios mensajes como respuesta (responden peticiones o provee un servicio). La mayoría del trabajo pesado (procesos de base de datos, procesar la lógica de la aplicación entre otros) está a cargo de los servidores, el cliente comúnmente se encarga de las funciones de administración de la interfaz de usuario, interacción con el usuario, recibir resultados del servidor, generar requerimientos de base de datos, entre otros.

Esta idea se puede aplicar tanto a programas que se están ejecutando en una sola máquina, pero es más ventajosa en un sistema operativo multiusuarios distribuidos a través de una red de computadores.

Figura 1. Modelo Cliente/Servidor



Fuente: Schiaffarino. Modelo cliente servidor. [en Línea]. [Recuperado el 10 noviembre 2020]. Disponible en: <https://blog.infranetworking.com/modelo-cliente-servidor>

**2.1.1 Características de la arquitectura Cliente/Servidor.** Las características básicas de una arquitectura Cliente / Servidor son:

- El cliente es quien inicia solicitudes o peticiones y el servidor se encarga de proveer servicios a 1 o varios clientes, por esto ambos tienen 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:** Mensajes de respuesta 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 persistentemente en la aplicación.

Este modelo se clasifica en dos:

**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 de 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.

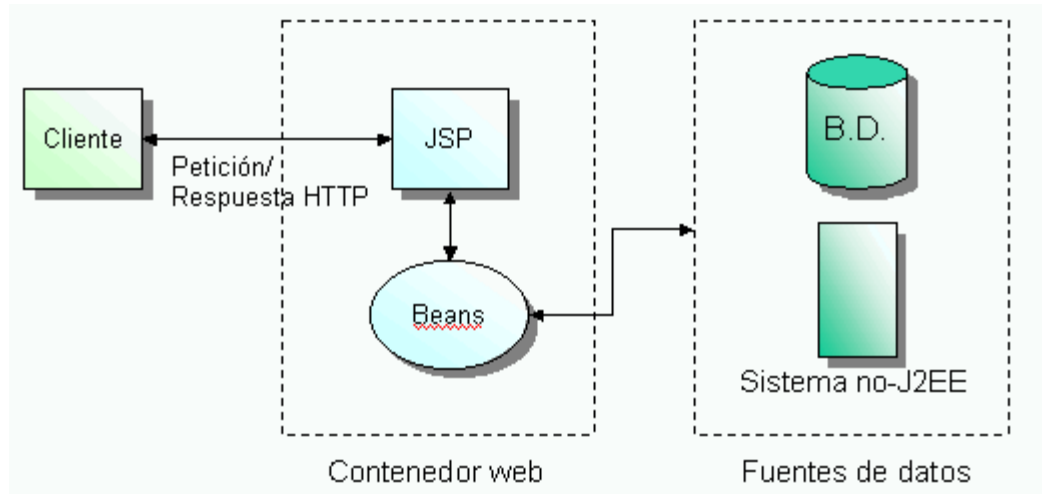
Las paginas JSP y servlets se ejecutan en la Máquina Virtual de Java, lo cual permite que se puedan usar en cualquier tipo de computador, siempre y cuando este instalada la Máquina Virtual de Java. Cada JSP se ejecuta en su propio contexto (llamado también hilo o hebra); pero no se comienza a ejecutar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



Fuente: Jtech. Arquitectura jsp. [en Línea]. [Recuperado el 10 noviembre 2020].  
 Disponible en:

[http://www.jtech.ua.es/j2ee/2002-2003/modulos/jsp/apuntes/apuntes4\\_1.htm](http://www.jtech.ua.es/j2ee/2002-2003/modulos/jsp/apuntes/apuntes4_1.htm)

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 con 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.
- Manejar los datos de acuerdo a las peticiones de los usuarios.
- Establecer y mantener las trayectorias de acceso a la base de datos de tal forma que los datos puedan ser capturados rápidamente.
- 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 (MySQL, 2021).

#### **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 varios sistemas de control de versiones pero CVS (Concurrent Versions System) 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 que componen 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 Apache 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 las 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 MODELO DE 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. Ciclo de vida modelo de prototipos evolutivos



Fuente: is3ados. Los Ciclos de Vida del Software (II). [en Línea]. [Recuperado el 10 noviembre 2020]. Disponible en: <http://is3ados.blogspot.com/2008/11/los-ciclos-de-vida-del-software-ii.html>

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

- Los usuarios no siempre tienen una idea clara al inicio de los requerimientos que necesitan y/o de cuáles son realmente plausibles.
- Se parte de una base de requerimientos que, si bien es general, sirven para dar una primera aproximación al problema y al sistema que se quiere implementar.
- Gracias al desarrollo incremental rápido y a la comunicación constante se puede transformar los requerimientos del software con relativo bajo costo gracias a que el sistema no crece en demasía entre cada entrega
- Las entregas continuas mejoran el entendimiento del problema objetivo, tanto para los desarrolladores como para el cliente (mejorando de esta manera todo el proceso de desarrollo).

### **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.

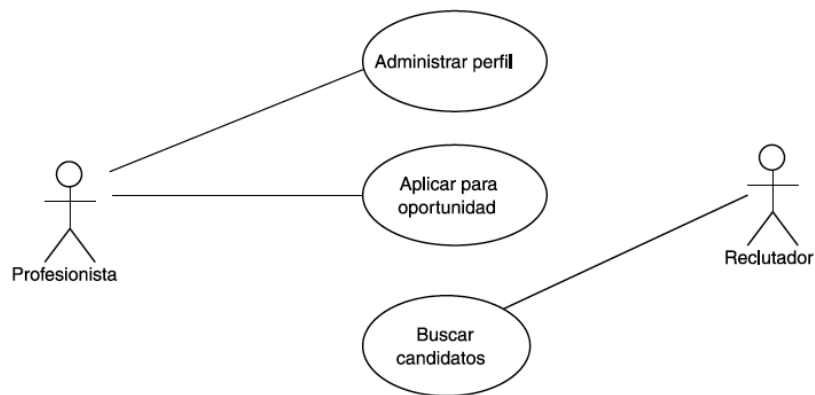
UML no es un método de desarrollo porque no indica los pasos a seguir para llegar al código, es decir, no especifica como pasar del análisis al diseño y de este al código. Al no ser un método de desarrollo resulta ser independiente del ciclo de desarrollo que se siga, puede encajar en un ciclo en cascada,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



Fuente: SG Buzz. *Entre lo Estático y lo Dinámico: El papel del analista y del programador.* [en Línea]. [Recuperado el 10 noviembre 2020]. Disponible en: [https://sg.com.mx/sites/default/files/inline-images/sg27-fundamentos1\\_0.png](https://sg.com.mx/sites/default/files/inline-images/sg27-fundamentos1_0.png)

**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 tienen 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 unos pocos servicios permitidos.

**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.

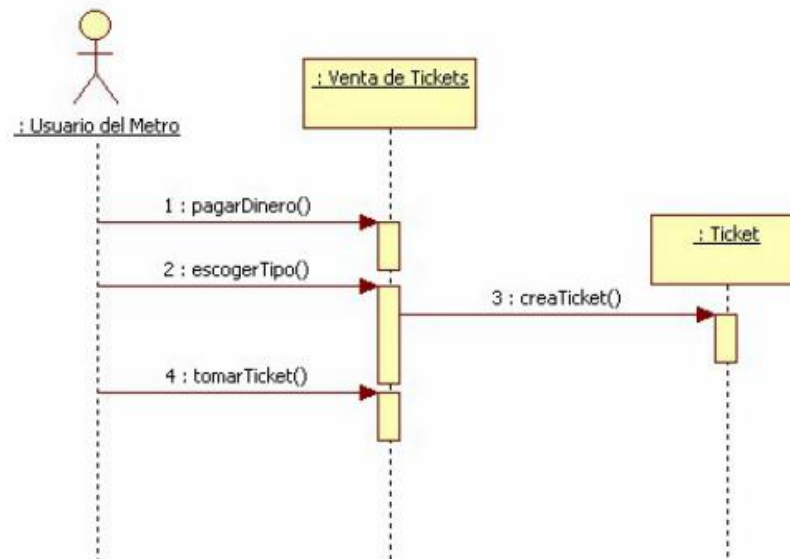
**Objetos:** Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de cada objeto será una línea discontinua conocida como la línea de vida de un objeto. Con la línea de vida se encuentra un pequeño rectángulo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



Fuente: Cevallos, K. UML: Diagrama de Secuencia. [en Línea]. [Recuperado el 10 noviembre 2020]. Disponible en:

<https://ingsoftwarekarlacevallos.wordpress.com/2015/07/07/uml-diagrama-de-secuencia/>

### 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 para anteponer a los nombres de estas tablas es “TB\_”, es decir la tabla que almacena los servicios que se despliegan en la página inicial del portal web es llamada “TB\_ServiciosInicio”, 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 para anteponer a los nombres de estas tablas es “TR\_”, por ejemplo, la tabla “ConversGrupo” de los foros es conocida como “TR\_ConversGrupo”.
- **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 EISIWeb. El prefijo para anteponer a los nombres de estas tablas es “TP\_”, es decir la tabla “Conversaciones”, es conocida como “TP\_Conversaciones”.
- **3.3.3 Clases.** 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.

- **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, EditarServiciosInicio.jsp, Foros.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.

## 4. DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO

Para el desarrollo de este proyecto se siguió el modelo 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 iniciar el proyecto aún no se contaba con una concepción clara de lo que sería el producto final, sin embargo, en el transcurso del 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 el desarrollo a la solución de las necesidades de los usuarios.

**4.1.1 Requerimientos de objetivo** 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 listaron los requerimientos detallados de este, los cuales se cumplieron para el prototipo final.

**Nota:** Entiéndase los requerimientos no listados a continuación como los requerimientos iniciales que, al no necesitar cambios, se convirtieron en finales.

**4.1.1.1 Implementar la actividad evaluativa de quices en el aula virtual de tal manera que no se vea afectada por la semana actual, sino que se pueda desarrollar en cualquier momento, además debe tener la opción de usar las preguntas del banco de preguntas para crear su contenido.**

**Objetivo inicial:**

- Crear un nuevo servicio de quices que use el banco de preguntas y se pueda iniciar en cualquier momento

**Requisitos finales del objetivo:**

- Permitir crear, modificar, listar y eliminar la nueva actividad de quices
- Permitir usar el banco de preguntas que actualmente se usa para talleres y exámenes para la construcción del quiz
- Permitir activar el quiz en cualquier momento sin importar la semana actual del aula virtual
- Permitir modificar el % de incidencia en la nota del estudiante esta actividad evaluativa
- Permitir consultar si un quiz está activo desde cualquier parte del aula (para el usuario Estudiante)
- Permitir reiniciar un quiz (borrar respuestas y habilitar el botón de iniciar actividad)

**4.1.1.2 Agregar al portal de las escuelas en la sección de trámites de grado un nuevo servicio donde los evaluadores de un proyecto de grado puedan acceder al formato del plan creado por el E3T**

**Objetivo inicial:**

- Permitir a los evaluadores visualizar el formato del plan E3T en la sección de trámites de grado

**Requisitos finales del objetivo:**

- Permitir al usuario súper administrador subir, descargar y eliminar un formato de plan E3T para pregrado y para posgrado
- Permitir a los estudiantes de pregrado y posgrado visualizar en la sección donde ellos suben su plan el formato del plan E3T
- Permitir a los evaluadores tanto de pregrado y posgrado descargar el formato del plan E3T en la sección de trámites de grado

**4.1.1.3 Implementar un servicio nuevo para que a los calificadores de un proyecto de grado se les muestre un link hacia el proyecto de grado que están calificando en la sección donde ellos emiten nota**

**Objetivo inicial:**

- Poder visualizar el plan de trabajo de grado del estudiante al cual están calificando

**Requisitos finales del objetivo:**

- Permitir al calificador visualizar el proyecto final del estudiante en caso de que se le halla requerido subirlo a la plataforma COMA (en un futuro será obligatorio realizar este procedimiento)
- Permitir al calificador visualizar el plan del estudiante al cual están emitiendo nota (para estudiantes de pregrado y posgrado)

**4.1.1.4 En la sección de grupos (sección usada por grupos de investigación, centros de estudio, etc.) de la plataforma COMA, crear un nuevo servicio**

**conocido como indicadores que permita la creación, modificación y eliminación de gráficos de barras, líneas, tortas y columnas para que así estos grupos puedan representar datos de una mejor forma.**

**Objetivo inicial:**

- Poder crear, leer , actualizar y eliminar gráficos llamados indicadores y que sean visibles dependiendo de si el usuario está logueado o no

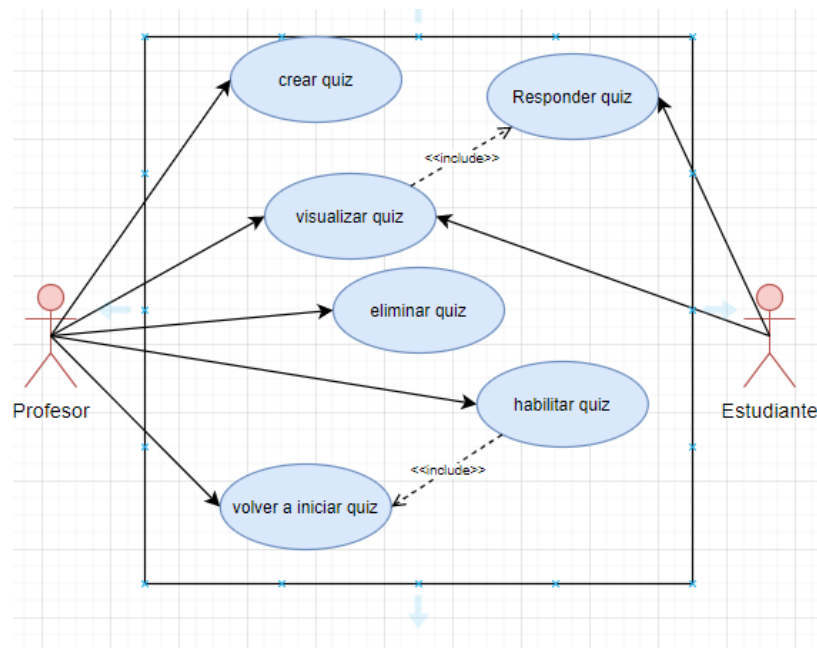
**Requisitos finales del objetivo:**

- Organizar los indicadores por cuatro filtros: área temática, categoría, grupo de indicadores y por último el nombre del indicador (representará los indicadores de una ciudad)
- Permitir exportar los gráficos y los datos de los gráficos en más de un formato (png, csv (para datos), xls (para datos), jpeg, pdf y svg)
- Permitir adjuntar a cada indicador un archivo en formato pdf
- Permitir crear diferentes tipos de gráficos (líneas, tortas, barras, columnas)
- Permitir crear, leer , actualizar y eliminar los gráficos mediante una interfaz de administrador
- Ajustar la visibilidad de los indicadores dependiendo del rol del usuario (usuario, estudiante y profesor principalmente)

**4.2 DIAGRAMAS DE CASOS DE USO**

**4.2.1 Servicio actividad evaluativa de quices**

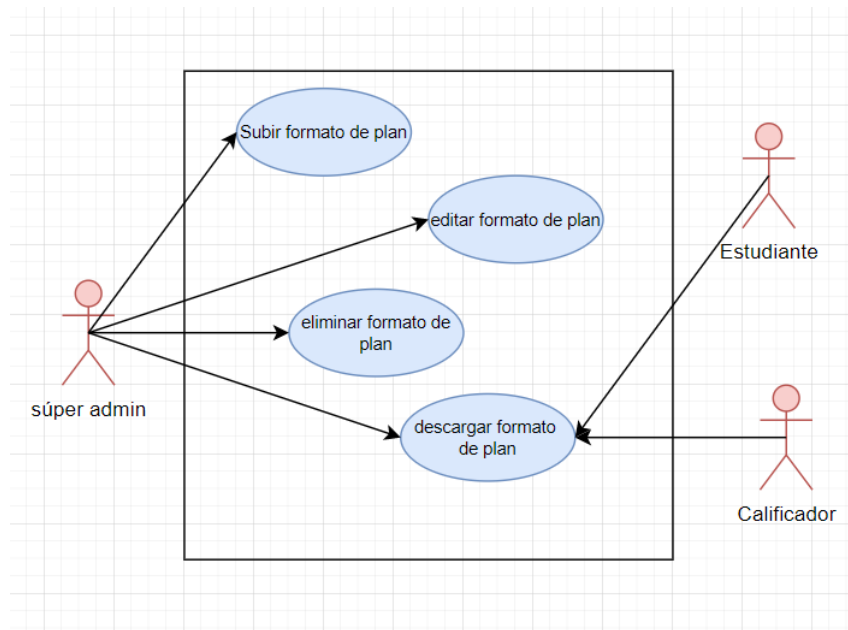
**Figura 6. Diagrama de casos de uso: Servicio de quices**



Para ambos casos la autenticación por usuario y contraseña es obligatorio para acceder a los servicios. El usuario profesor puede gestionar los quices y habilitar y reiniciarlos en cualquier momento, por otro lado el estudiante únicamente puede visualizar y responder los quices

#### **4.2.2 Servicio formato de plan E3T**

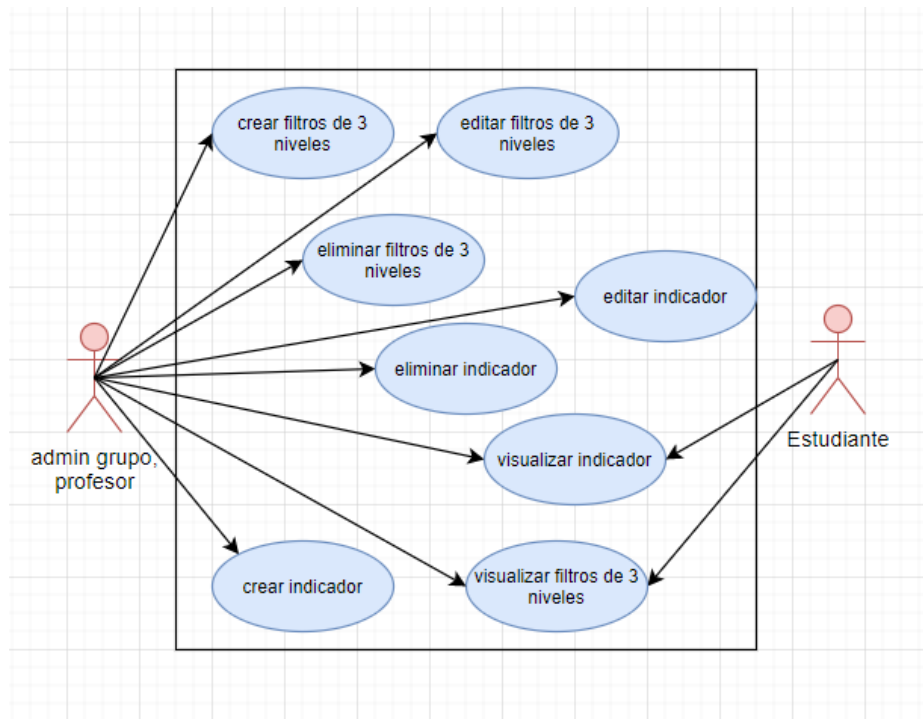
**Figura 7. Diagrama de casos de uso: Servicio formato plan E3T**



Para ambos casos la autenticación por usuario y contraseña es obligatorio para acceder a los servicios. El usuario súper administrador es quien gestiona los formatos del plan (en pdf), el estudiante lo visualiza para guiarse para elaborar su plan de trabajo de grado y el calificador lo visualiza para basarse en este para dar su calificación.

#### 4.2.3 Servicio de indicadores

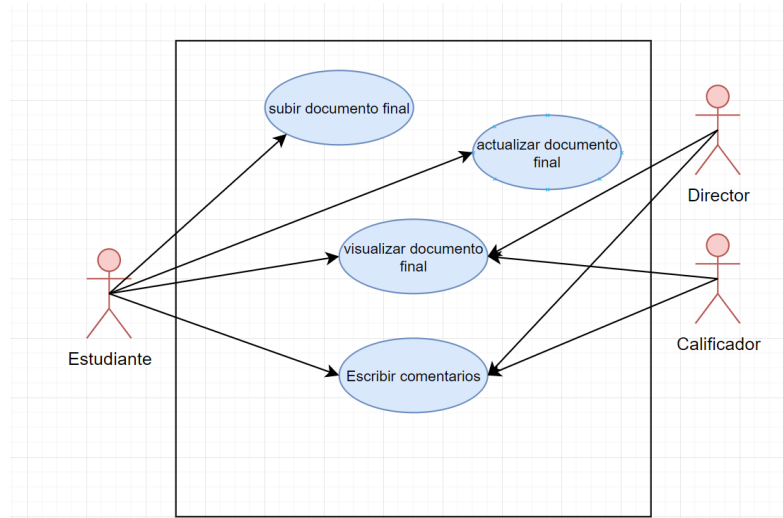
Figura 8. Diagrama de casos de uso: Servicio de indicadores



Filtro de 3 niveles se refiere al filtro de área temática, categoría, grupo de indicadores, cada uno de estos filtros se puede editar. El usuario administrador del grupo habilita el menú de gestión de indicadores según el rol (profesores, etc) y estos pueden crear, leer, actualizar y eliminar indicadores que ellos hallan creado.

#### 4.2.4 Servicio de gestión de archivo final

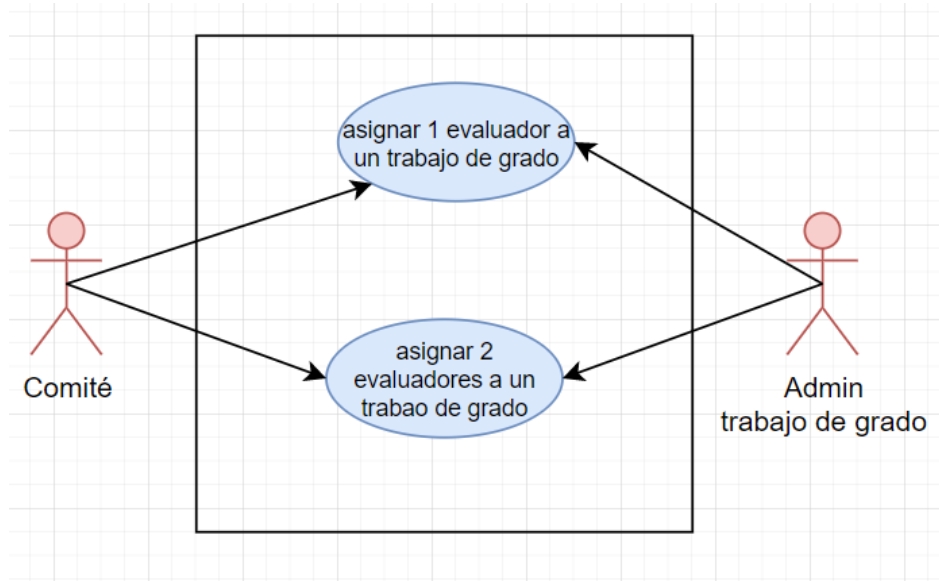
Figura 9. Diagrama de casos de uso: Servicio de gestión de archivo final



Un estudiante puede subir varias versiones de su documento final con el fin de llenar a la versión requerida ya sea por el calificador o por el director y el medio de comunicación habilitado para estos es un servicio de chat que está en ese mismo servicio.

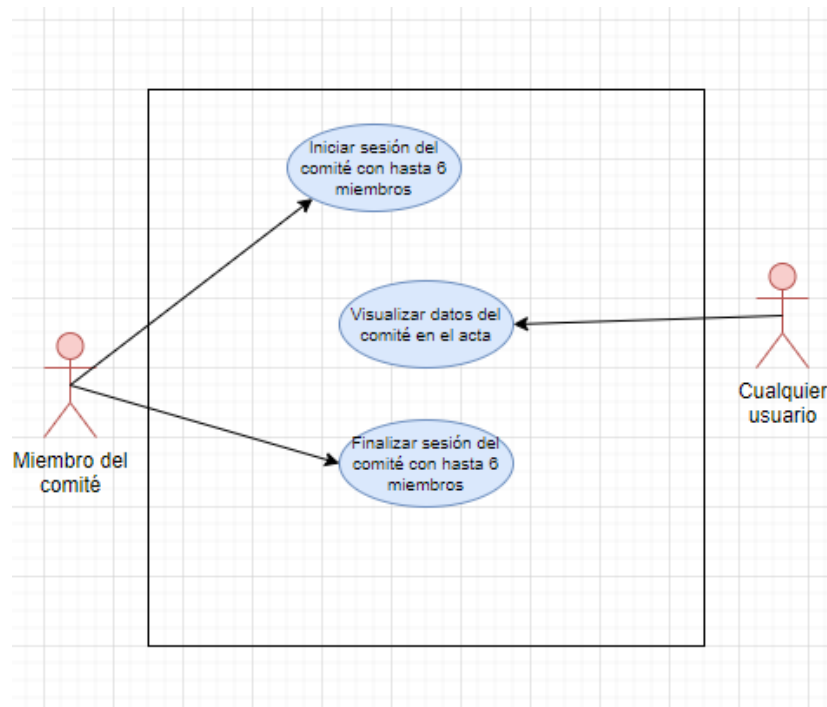
#### 4.2.5 Asignar 2 evaluadores a un trabajo de grado

Figura 10. Diagrama de casos de uso: Asignar hasta 2 evaluadores a un trabajo de grado



El comité y el administrador de trabajos de grado ahora pueden asignar hasta 2 evaluadores a un trabajo de grado.

**Figura 11. Diagrama de casos de uso: Asignar hasta 6 miembros del comité de trabajos de grado**



El comité ahora puede conformarse por hasta 6 personas y además la información de todos ellos se ven reflejadas en las actas.

### 4.3 DOCUMENTACIÓN DE CASOS DE USO DEL SISTEMA

#### 4.3.1 Servicio actividad evaluativa de quices

**Tabla 1. Casos de uso: Servicio de quices**

TÍTULO	FUNCIONES PRIMARIAS
CREAR QUIZ	Crear un quiz especificando: título, descripción, las preguntas del banco de preguntas y la cantidad de preguntas a mostrar

EDITAR QUIZ	Editar los mismos campos que cuando se crea
LISTAR QUIZ	Listar los quices (cantidad de preguntas,cantidad preguntas a mostrar, descripción, etc )
BORRAR QUIZ	Eliminar un quiz que aún no tenga respuestas asociadas
HABILITAR QUIZ	Cambiar el estado del quiz para que los estudiantes puedan hacerlo
VOLVER A INICIAR QUIZ	Borrar las respuestas asociadas a un quiz
VISUALIZAR QUIZ (ESTUDIANTE)	Listar información básica del quiz
RESPONDER QUIZ	Botón para desplegar la interfaz donde los estudiantes pueden responder las preguntas

#### 4.3.2 Servicio formato de plan E3T

Tabla 2. Casos de uso: Servicio formato de plan E3T

TÍTULO	FUNCIONES PRIMARIAS
CREAR FORMATO DE PLAN	Subir el formato del plan en formato .pdf tanto para pregrado como para posgrado con un peso máximo de 20 MB
EDITAR FORMATO DE PLAN	Permitir Eliminar y subir un nuevo plan con una sola acción
LEER FORMATO DE PLAN	Permite descargar el formato de plan (en caso de que exista)
BORRAR FORMATO DE PLAN	Borrar el archivo del formato de plan en el servidor y su referencia en BD

### 4.3.3 Servicio de indicadores

Tabla 3. Casos de uso: Servicio de indicadores

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
CREAR FILTRO DE 3 NIVELES	Poder crear 3 principales filtros de categorías (área temática, categoría, grupo de indicadores)
EDITAR FILTRO DE 3 NIVELES	Permitir cambiar el nombre de cada filtro
LEER FILTRO DE 3 NIVELES	Permite listar esas categorías para poder filtrar los indicadores
ELIMINAR FILTRO DE 3 NIVELES	Permite eliminar cada filtro y los filtros que contenga (ejemplo: una categoría puede tener muchos grupos de indicadores asociados)
CREAR INDICADOR	Permite elegir el tipo de indicador (barras, columnas, líneas y tortas), asignarle cada uno de los filtros de 3 niveles para categorizarlo y añadir los datos
EDITAR INDICADOR	Permite editar los datos y el tipo de gráfico del indicador
VISUALIZAR INDICADOR	Permite ver el indicador además de poder ver el archivo adjunto, descripción y la posibilidad de exportar los datos en los formatos antes mencionados

ELIMINAR INDICADOR	Permite al creador del indicador borrarlo junto con su descripción y archivo adjunto
--------------------	--

#### 4.3.4 Servicio de gestión de archivo final

**Tabla 4. Casos de uso: Servicio de gestión de archivo final**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
SUBIR DOCUMENTO FINAL	El estudiante puede subir en formato .pdf el documento de su trabajo de grado
ACTUALIZAR DOCUMENTO FINAL	El estudiante puede subir nuevas versiones de su trabajo de grado
VISUALIZAR DOCUMENTO FINAL	Se permite descargar el archivo para su visualización
ESCRIBIR COMENTARIOS	Se permite el envío de mensajes por medio de un chat entre el/los estudiante/s y el/los calificador/es

**Tabla 5. Casos de uso: Asignar hasta 2 evaluadores a un trabajo de grado**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
ASIGNAR HASTA 2 EVALUADORES A UN TRABAJO DE GRADO	El comité o el admin podrán asignar hasta 2 evaluadores a un trabajo de grado

**Tabla 6. Casos de uso: Asignar hasta 6 miembros al comité de trabajos de grado**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
INICIAR SESIÓN DEL COMITÉ CON HASTA 6 MIEMBROS	El comité ahora puede conformarse hasta un máximo de 6 personas
FINALIZAR SESIÓN DEL COMITÉ CON HASTA 6 MIEMBROS	El comité ahora puede conformarse hasta un máximo de 6 personas
VISUALIZAR DATOS DEL COMITÉ EN EL ACTA	El acta generada tiene en cuenta a todos sus miembros

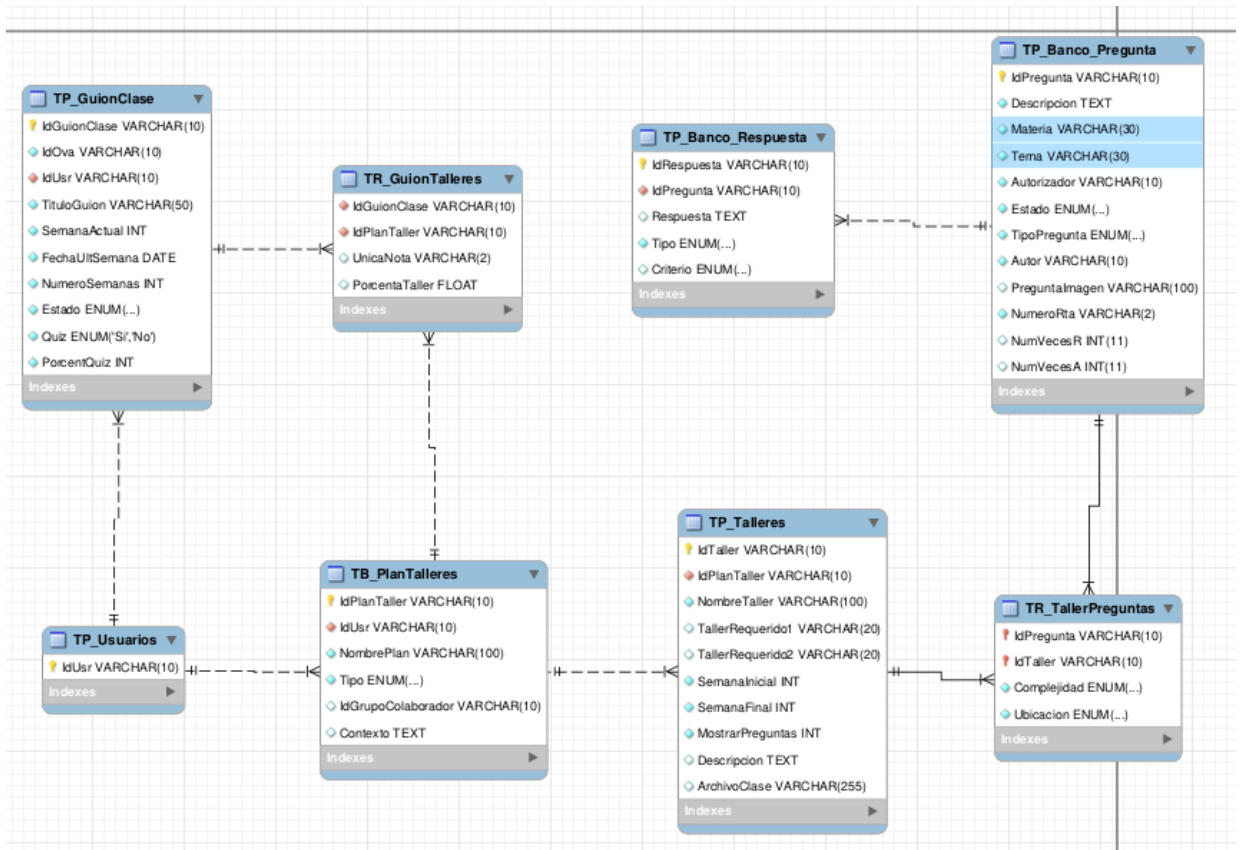
#### **4.4 DISEÑO Y ANÁLISIS**

##### **4.4.1 Diagrama Entidad/Relación de los servicios desarrollados**

Diagramas simplificados: Los diagramas a continuación contiene el esquema esencial para entender el funcionamiento básico de cada servicio, el esquema original tiene muchas más tablas y relaciones.

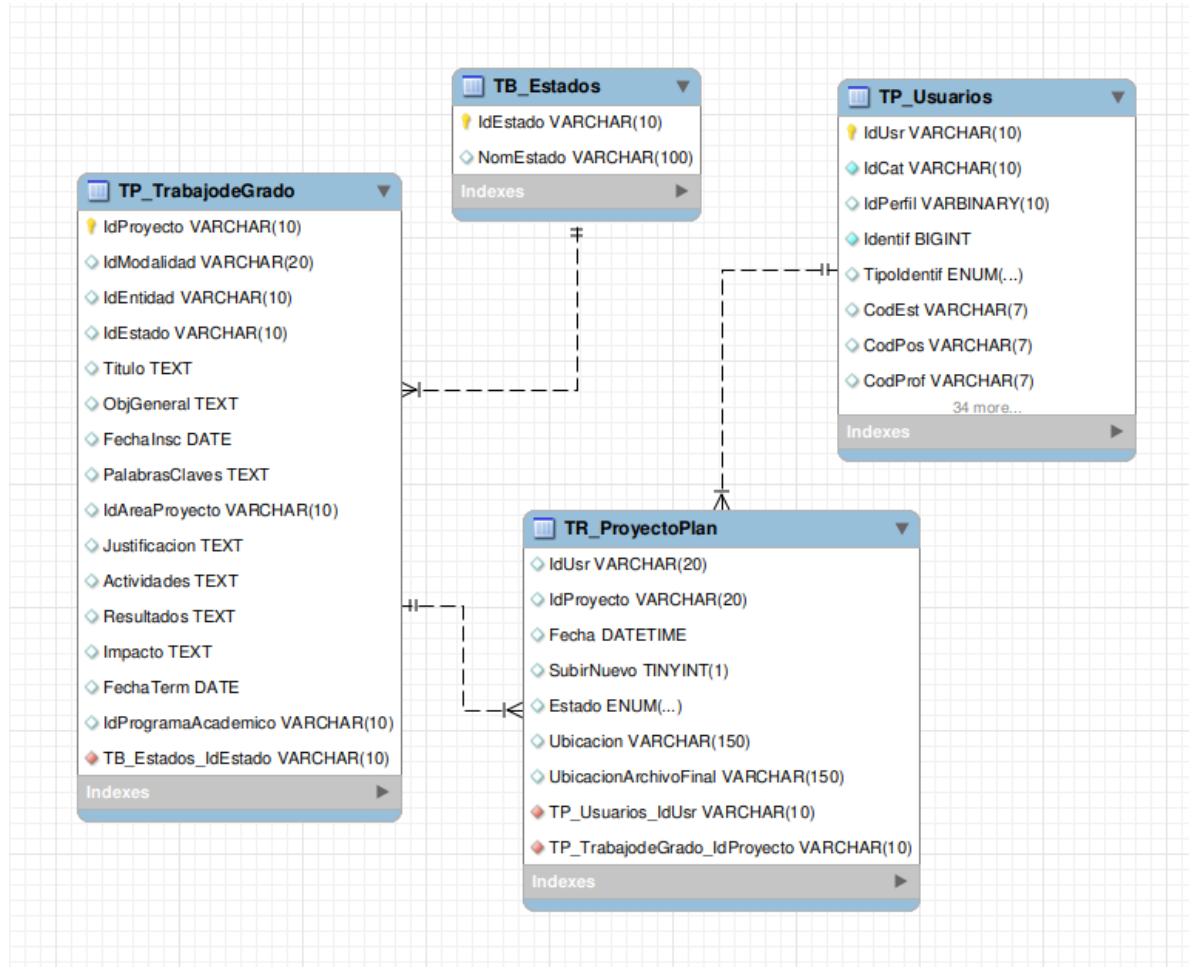
### 4.4.1.1 Servicio de quices

Figura 12. Diagrama E/R: Servicio de quices



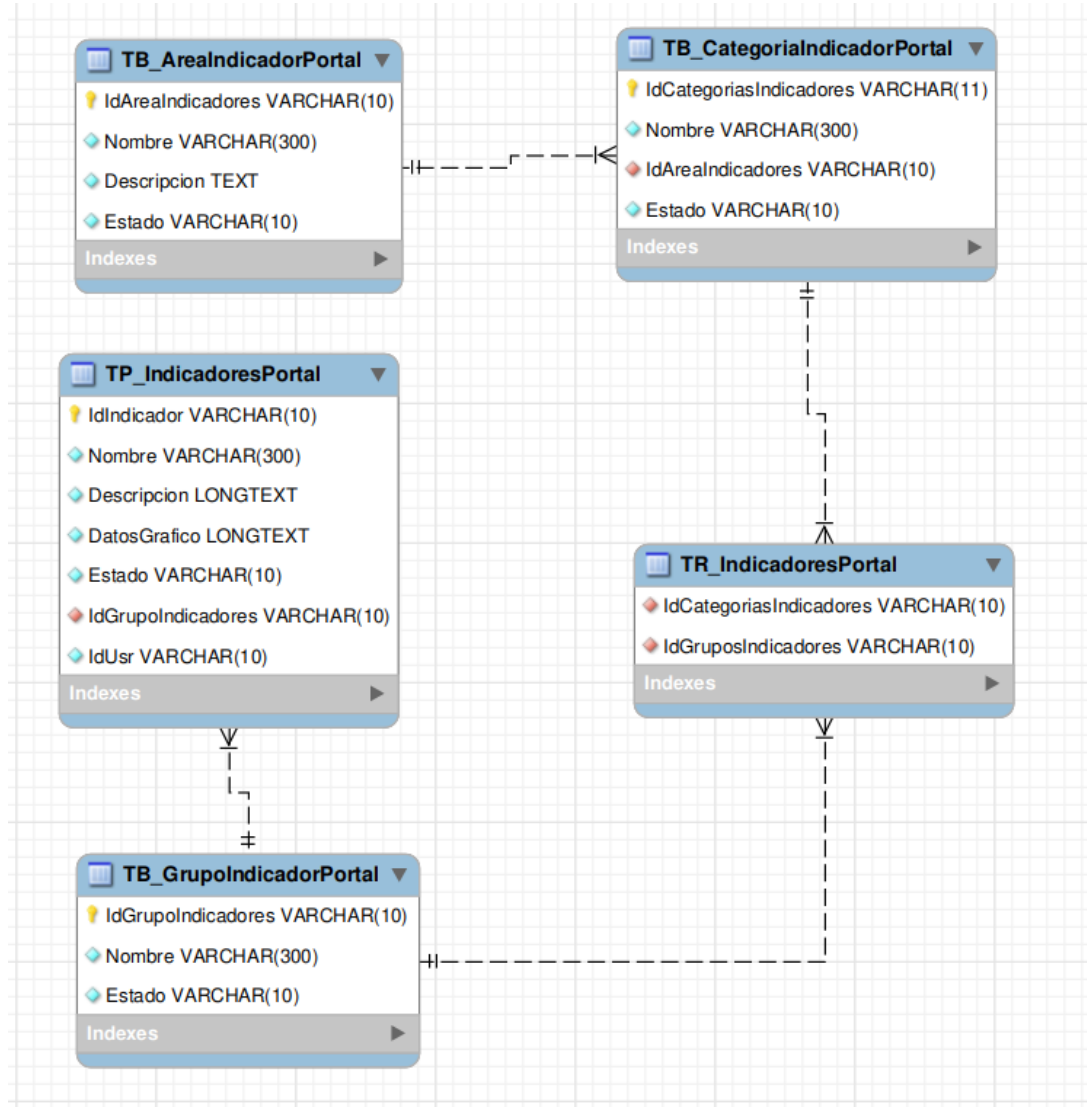
#### 4.4.1.2 Servicio formato de plan E3T

Figura 13. Diagrama E/R: Servicio formato de plan E3T



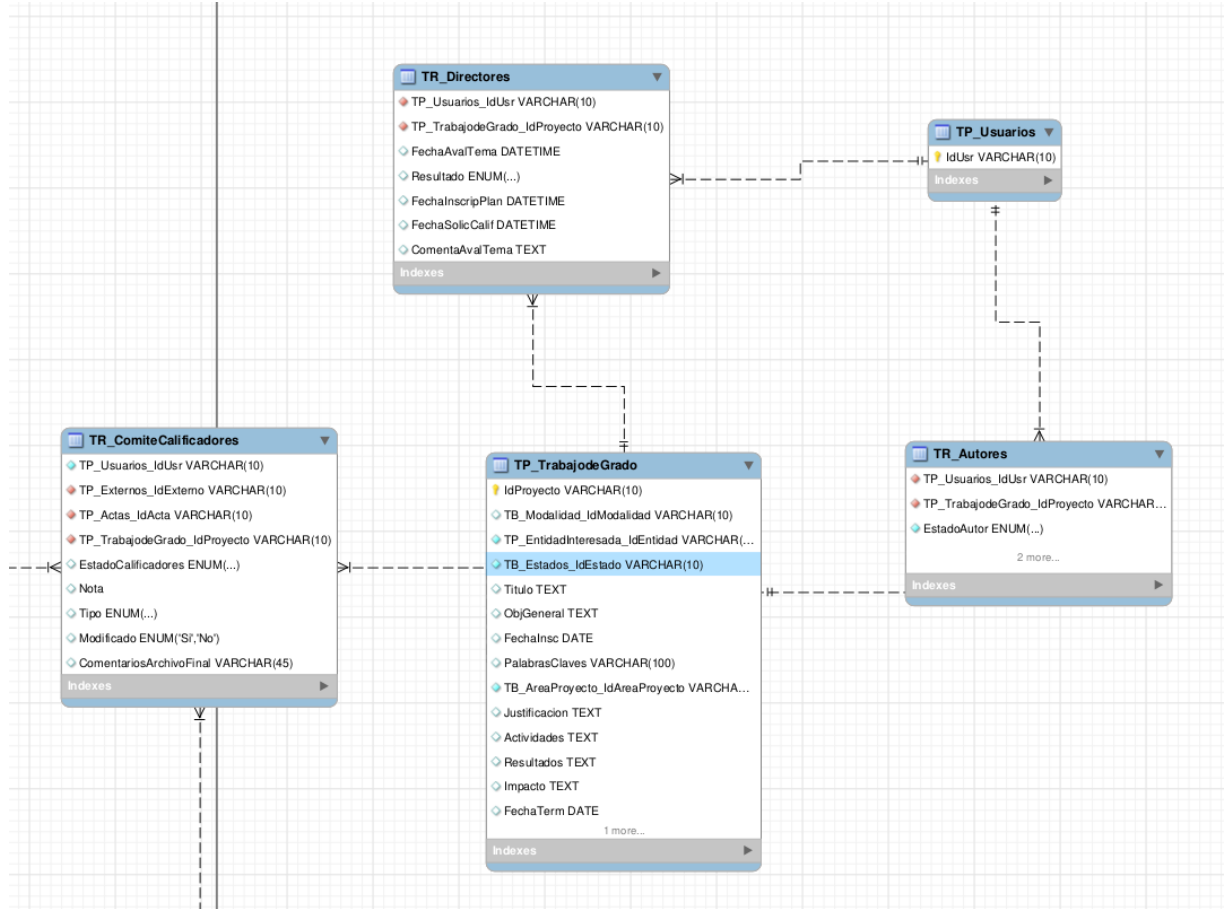
### 4.4.1.3 Servicio de indicadores

Figura 14. Diagrama E/R: Servicio de indicadores



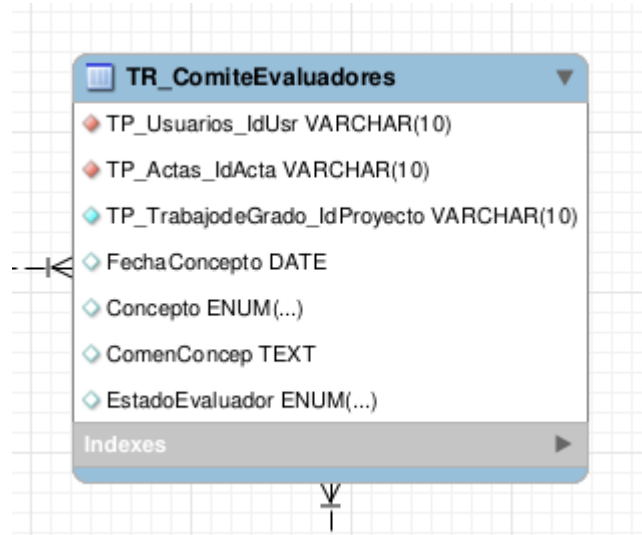
#### 4.4.1.4 Servicio de gestión de archivo final

Figura 15. Diagrama E/R: Servicio de gestión de archivo final



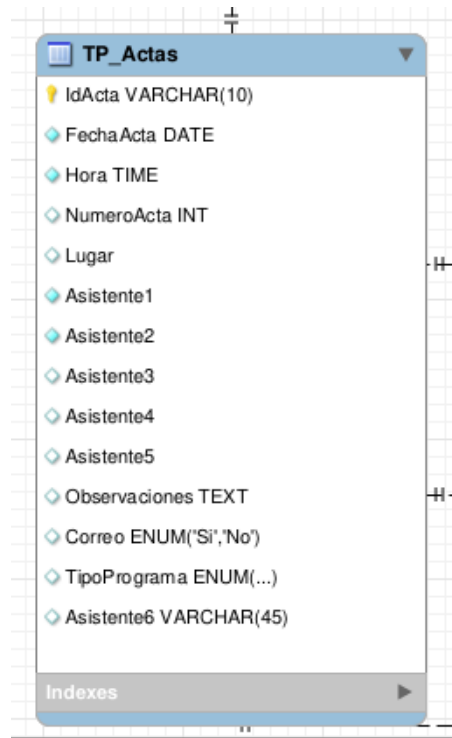
#### 4.4.1.5 Asignar hasta 2 evaluadores a un trabajo de grado

Figura 16. Diagrama E/R: Asignar hasta 2 evaluadores a un trabajo de grado



#### 4.4.1.6 Asignar hasta 6 miembros al comité de trabajos de grado

Figura 17. Diagrama E/R: Asignar hasta 6 miembros al comité de trabajos de grado



#### 4.5 DESCRIPCIÓN DE LAS ENTIDADES

Descripción de las entidades creadas y utilizadas en las bases de datos Diamante ubicada en el servidor de cada portal.

Tabla 7. Descripción de las entidades

ENTIDAD	DESCRIPCIÓN
TP_GuionClase	Contiene el estado del aula (detenida, en línea) además de información básica sobre esta.
TR_GuionTalleres	Tabla de relación que enlaza las aulas con los planes de talleres

TP_Usuarios	Contiene información de identificación y contacto de todos los usuarios creados en el portal
TB_PlanTalleres	Contiene información general de un plan de talleres como lo es su nombre, tipo (Quiz, Taller, Examen, etc.)..
TP_Talleres	Contiene información general de un taller en sí (nombre, número de preguntas, etc).
TR_TallerPreguntas	Relaciona un taller con las preguntas del banco de preguntas.
TP_Banco_Pregunta	Tabla que contiene cada una de las preguntas que se crean en el portal y son usadas para exámenes, talleres, etc.
TP_Banco_Respuesta	Tabla de soluciones a las preguntas de TP_Banco_Pregunta.
TP_TrabajodeGrado	Contiene información principal de un trabajo de grado de un usuario de el portal.
TB_Estados	Contiene el estado de avance de un trabajo de grado, ejemplo: Plan aprobado, Plan en espera de calificador, etc.
TR_ProyectoPlan	Relaciona un usuario del portal con su trabajo de grado, además guarda información importante como lo es la ubicación de su plan de proyecto de grado y archivo final.

TB_AreaIndicadorPortal	Contiene el área temática de un indicador
TB_CategorialIndicadorPortal	Contiene la categoría de un indicador
TB_GrupoIndicadorPortal	Contiene el grupo al cual pertenece de un indicador
TR_IndicadoresPortal	Relaciona una categoría con un grupo de indicadores
TB_IndicaodresPortal	Contiene los datos principales del indicador (título, datos, etc)
TR_Directores	Relaciona los directores de un proyecto de grado con un trabajo de grado en específico
TR_ComiteCalificadores	Relaciona los calificadores con un trabajo de grado en específico
TR_Autores	Contiene el/los autor/es específico/s de un trabajo de grado además de contener la ruta hacia el documento final subido por el estudiante.
TR_ComiteEvaluadores	Relaciona los evaluadores con un trabajo de grado en específico
TP_Actas	Contiene información relacionada con las decisiones tomadas en las sesiones que realiza el comité de trabajos de grado

#### 4.6 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 (Ver Anexo A).

#### **4.7 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. 61
- Manejador de Base de datos, MySQL 5.0
- Librería de javascript para representar datos HIGHCHART.

Las herramientas listadas y el buen manejo de estas fueron esenciales para poder dar desarrollo a las actividades que son expuestas, además de esto fue de gran ayuda la guía y enfoque provistos por el director de proyecto y la información provista por calumet estándar.

Para el desarrollo se trabajó con la base de datos llamada “Diamante” ubicada en todos los servidores de los portales de las escuelas. 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 el resultado correspondiera con lo esperado, de esta forma, se evidenció el correcto funcionamiento en la captura de datos, selección de ítems y almacenamiento de información.

Estos servicios se implementaron de forma local primeramente, luego en el servidor de pruebas que tiene el grupo CALUMET, después en el servidor de petróleos y por último estos nuevos servicios fueron propagados a los demás servidores de las demás escuelas a las que el proyecto COMA presta sus servicios.

## 4.8 INTERFACES DE USUARIO DESARROLLADAS

Por cuestión de privacidad de los datos (para los servicios que aplique) se presentan imágenes del entorno local que se usó para el desarrollo donde los datos han sido transformados de tal forma que no halla exposición de información sensible o confidencial.

### 4.8.1 Servicio de quices

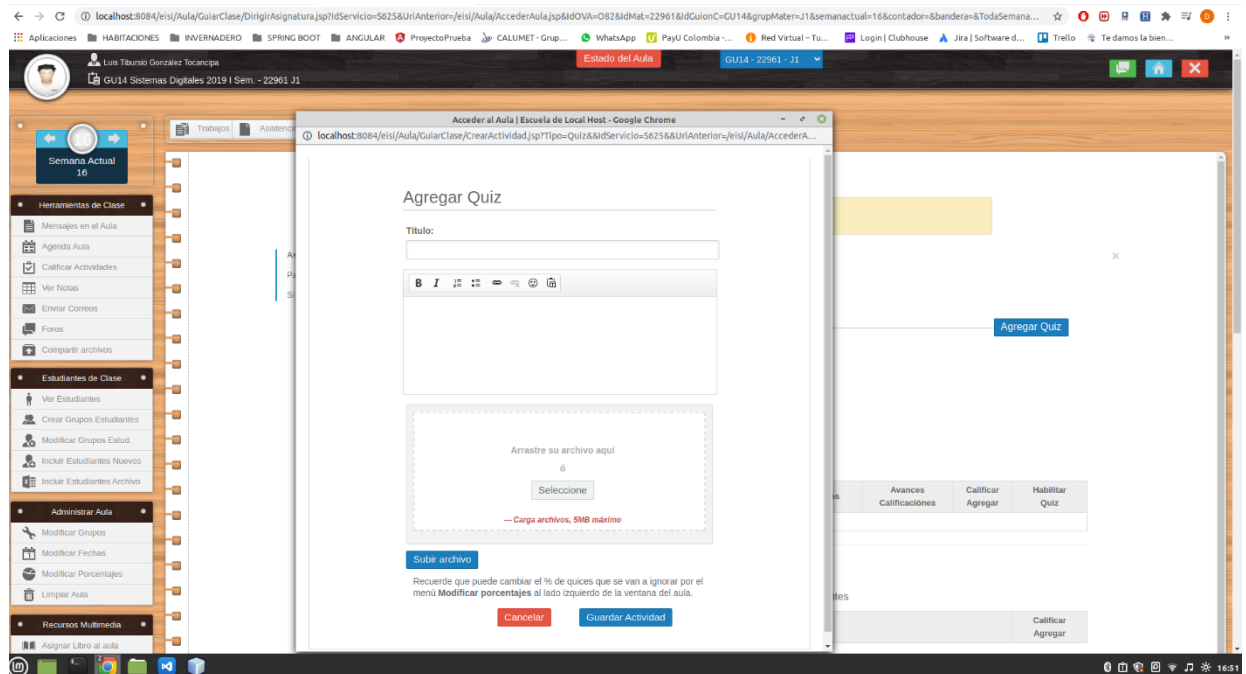
Figura 18 . Interfaz de usuario (profesor – Vista general): Servicio de quices

The screenshot displays a web application interface for a teacher's general view of the quiz service. The interface includes a sidebar with navigation options like 'Estudiantes de Clase', 'Administrar Aula', and 'Recursos Multimedia'. The main content area shows a 'Plan de Quices' for 'PT30 - Quiz - 22961' and three sections of quiz data: 'Quices en línea', 'Quices con cero preguntas a responder por los estudiantes', and 'Quices por habilitar'. The 'Quices por habilitar' section contains a table with columns for 'Actividad', 'Editar Datos', 'Porcentaje', 'Titulo Actividad', 'Descripción', 'Avances Subgrupos', 'Avances Calificaciones', 'Calificar Agregar', and 'Habilitar Quiz'. A dropdown menu is open over the 'Habilitar Quiz' button, showing options 'Agregar Pregunta' and 'Modificar Preguntas'.

Actividad	Editar Datos	Porcentaje	Titulo Actividad	Descripción	Avances Subgrupos	Avances Calificaciones	Calificar Agregar	Habilitar Quiz
T105		5	Primer Quiz	El Quiz consta de 1 preguntas	A la espera del inicio de la actividad.	Por favor, habilite el quiz mediante el botón Habilitar.		

Interfaz de visualización del panel administrativo para los quices. Se puede apreciar el estado de los quices actuales para el profesor en sesión en específico además de los menús de gestión para cada quiz (modificar preguntas, habilitar)

**Figura 19. Interfaz de usuario (profesor – Agregar quiz): Servicio de quices**



Interfaz de visualización del panel administrativo para los quices. Se puede apreciar el control para añadir un nuevo quiz.

**Figura 20. Interfaz de usuario (profesor – Modificar preguntas): Servicio de quices**

**Modifica Taller**

Título:

Plan de Talleres o Exámenes:

Descripción:

Archivo:

Semana Inicial:  Semana Final:

**Datos sobre las preguntas del taller**

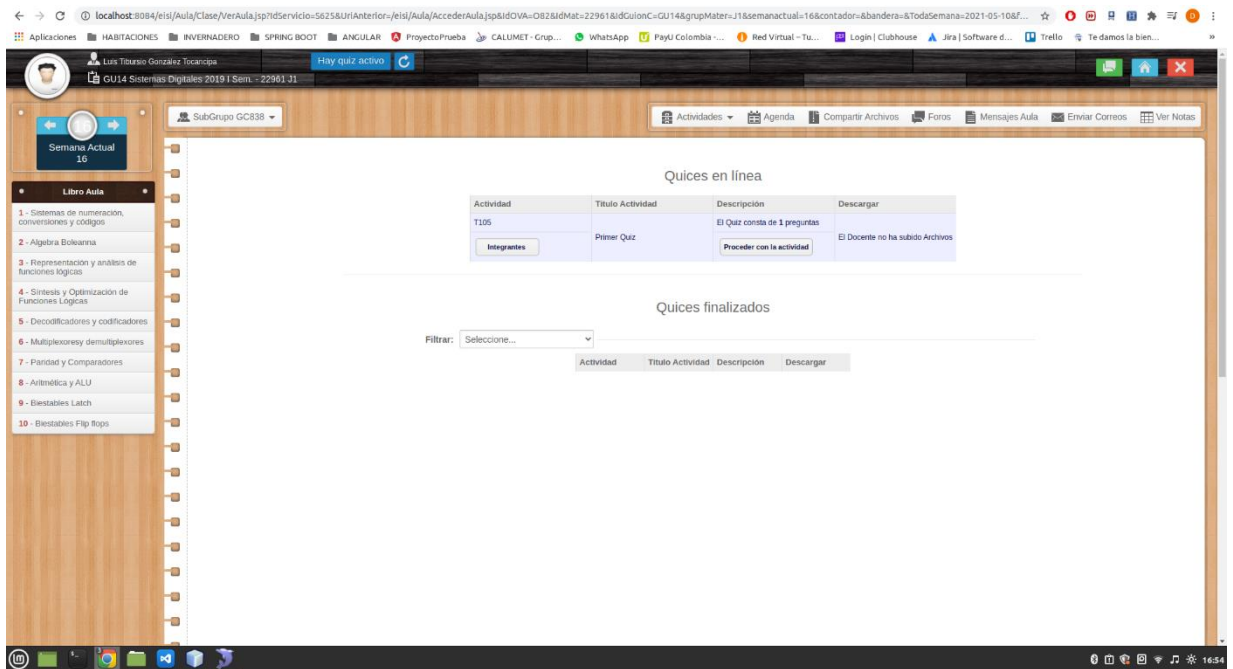
	Principio de Tema	Mitad de Tema	Final de Tema
Complejidad Baja	0	0	0
Complejidad Media	0	3	0
Complejidad Alta	0	0	0

Las preguntas para responder por los estudiantes son:  
3 preguntas se han agregado al taller

Preguntas	Complejidad	Ubicación en la temática	Estadísticas	Eliminar
BP577 - Si las entradas de habilitación están inactivas el decodificador: <a href="#">ver pregunta completa +</a>	<input type="text" value="Medio"/>	<input type="text" value="Mitad"/>	Nº veces respondidas: 407 Nota promedio: NaN	<input type="checkbox"/>
BP829 - La comunicación entre todos los miembros del Aula se realiza: <a href="#">ver pregunta completa +</a>	<input type="text" value="Medio"/>	<input type="text" value="Mitad"/>	Nº veces respondidas: 57 Nota promedio: NaN	<input type="checkbox"/>

Interfaz de visualización del panel de modificar preguntas para un quiz en específico. Se pueden apreciar los controles para gestionar las preguntas de ese quiz además de que la semana inicial y final están inhabilitadas ya que un quiz no se verá afectado por esta (podrá ser iniciado en cualquier momento de vida del aula).

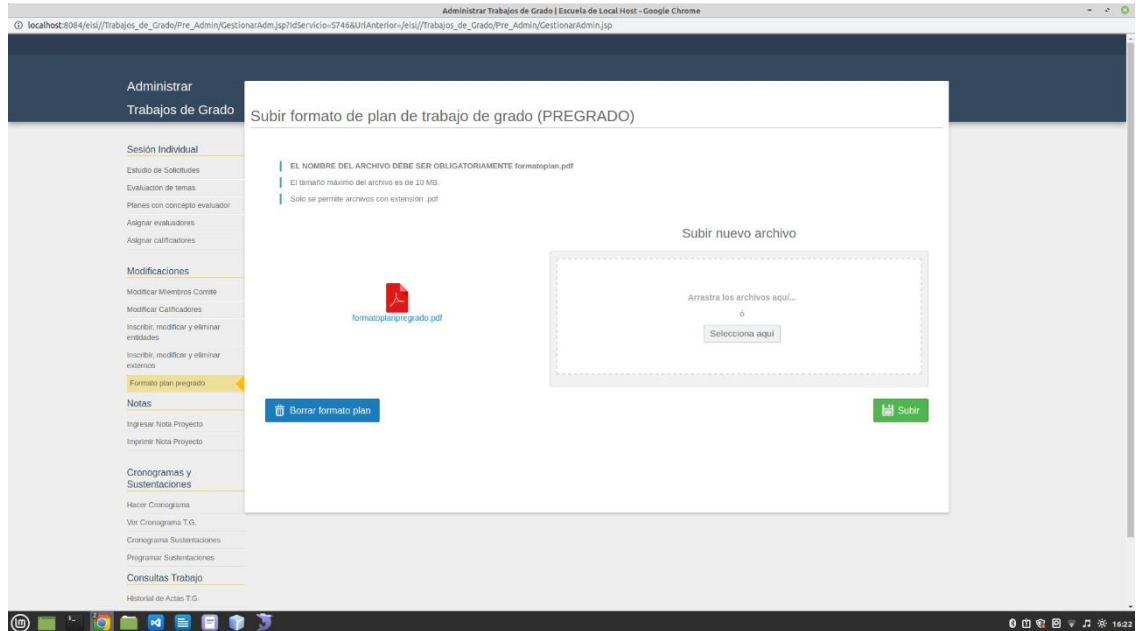
Figura 21. Interfaz de usuario (estudiante): Servicio de quices



Interfaz de visualización de quices para un estudiante. Se puede apreciar los quices habilitados en el momento y los quices que ya desarrolló el estudiante.

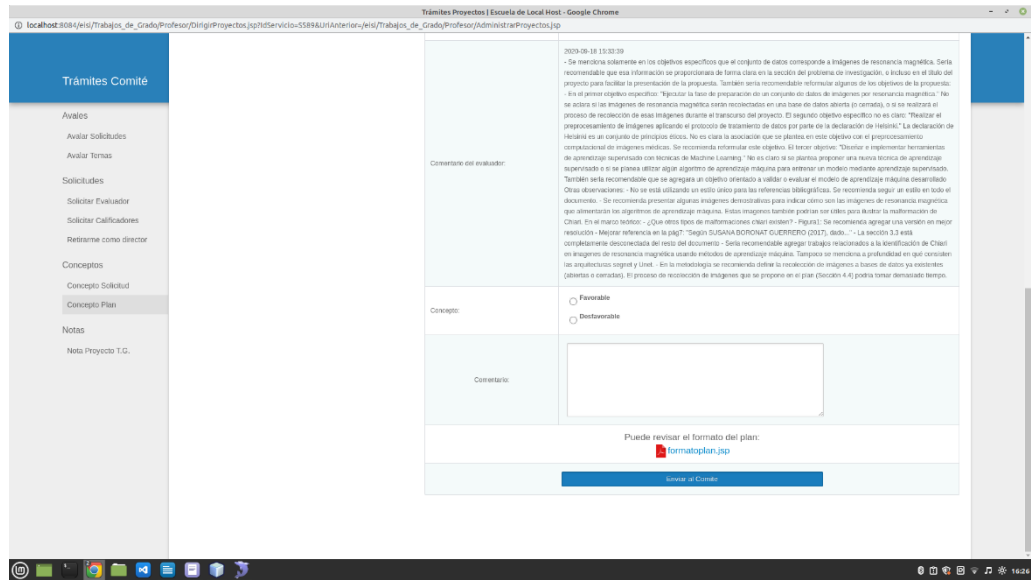
## 4.8.2 Servicio formato de plan E3T

Figura 22. Interfaz de usuario (admin): Servicio de formato plan E3T



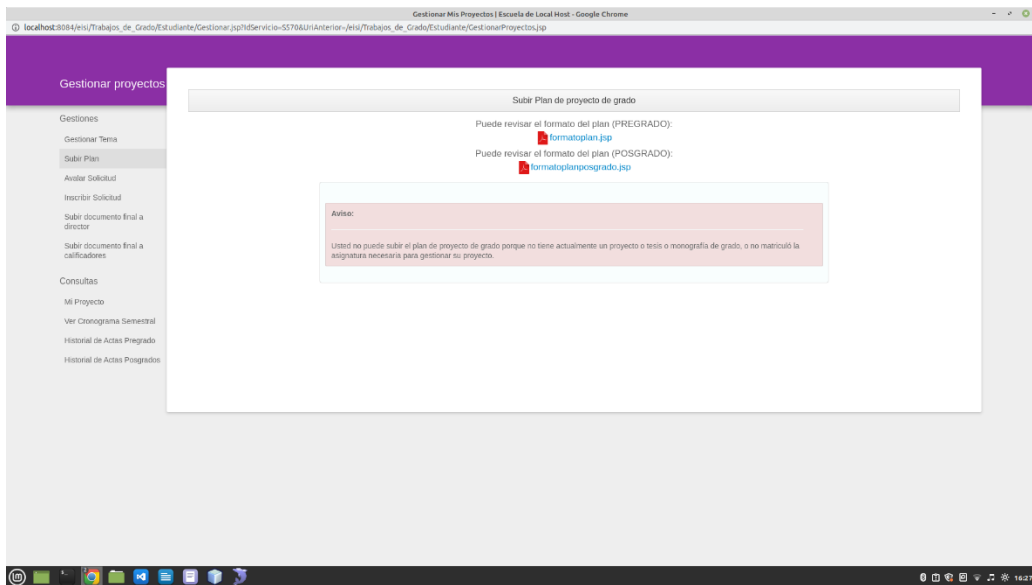
Interfaz de visualización del panel administrativo para pregrado (el panel de posgrado varía en detalles mínimos) para la gestión de el formato de plan. Se puede apreciar los botones para borrar el formato del plan y para subir uno nuevo.

**Figura 23. Interfaz de usuario (evaluador): Servicio de formato de plan E3T**



Interfaz de visualización del panel de concepto de plan para un evaluador. Se puede apreciar el control para poder descargar el formato del plan, este control aparece tanto para proyectos de pregrado como de posgrado.

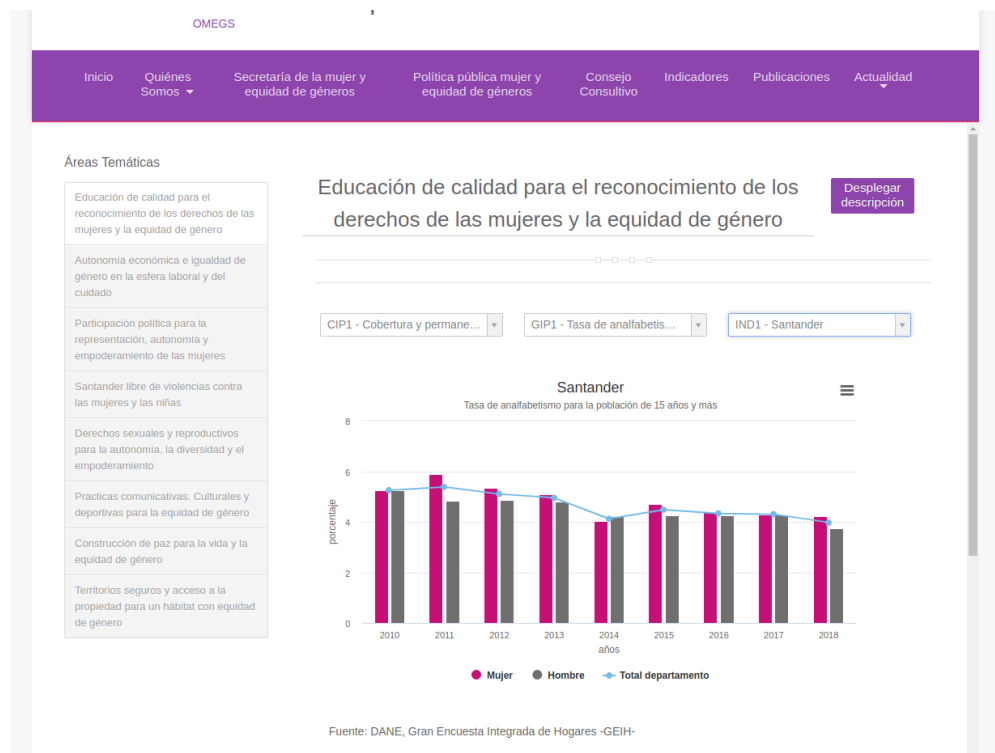
**Figura 24. Interfaz de usuario (estudiante): Servicio de formato de plan E3T**



Interfaz de visualización del panel de subir plan. Se puede apreciar los controles para poder descargar el formato del plan tanto para pregrado como para posgrado.

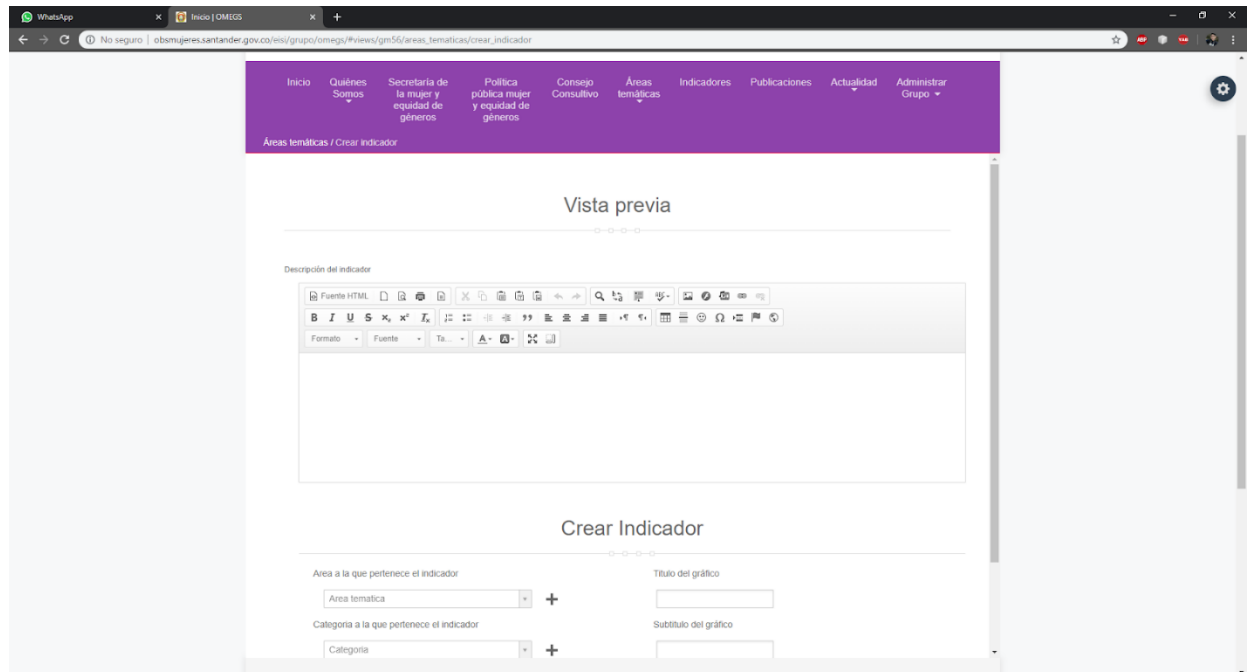
### 4.8.3 Servicio de indicadores

Figura 25. Interfaz de usuario (cliente): Servicio de indicadores



Interfaz de visualización de los indicadores que actualmente está expuesta a todo el público, se puede notar el filtro de 3 niveles junto con los datos asociados para cada género.

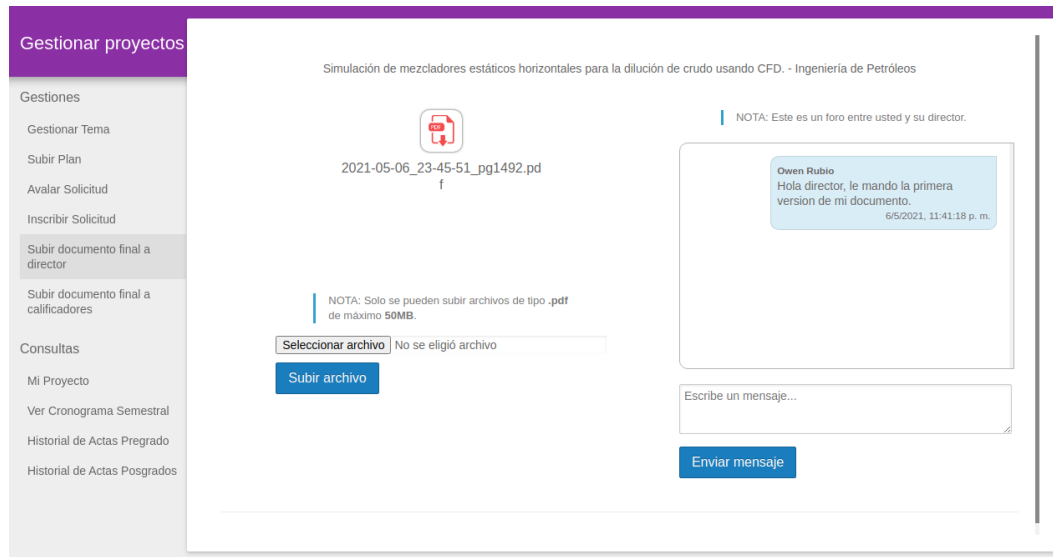
**Figura 26. Interfaz de usuario (admin): Servicio de indicadores**



Interfaz de administrador de los indicadores. Se puede apreciar los controles para llenar la información de un indicador en específico.

## 4.8.4 Servicio de gestión del archivo final

Figura 27. Interfaz de usuario: Servicio de gestión del archivo final



Interfaz de visualización de subida del archivo final de un trabajo de grado para un estudiante. Se puede apreciar el chat en el se pueda establecer una comunicación entre ambas partes además de la posibilidad de subir diferentes versiones del documento (esto debido a sugerencias del director y/o calificador/es).

## 4.8.5 Asignar 2 evaluadores a un trabajo de grado

**Figura 28. Interfaz de usuario: Asignar 2 evaluadores a un trabajo de grado**

Sesion Comite T.G. | Escuela de Local Host - Google Chrome  
 localhost:8081/eisi/Trabajos\_de\_Grado/Pre\_Comite/AsignarEvaluador3.jsp?IdProyecto23=PG1492&IdServicio=5538&...

Asignar Evaluador

En la siguiente tabla se muestran los profesores vinculados a la Escuela y el número de proyectos a evaluar y calificar por cada uno de ellos

Profesores Planta | Profesores Catedra | Profesores Proyectos

PROYECTOS A EVALUAR Y CALIFICAR POR CADA PROFESOR				
NOMBRE DEL PROFESOR	PROYECTOS ASIGNADOS PARA EVALUAR Y CALIFICAR			
	Total evaluados	Evaluados este año	En evaluación	En calificación
<input checked="" type="checkbox"/> Astrid Tibursio Rodriguez Tocancipa	16	0	1 PG1340	3
<input checked="" type="checkbox"/> Edelberto Hernandez Trejos	1	0	1 PG1340	0
<input type="checkbox"/> Emiliano Tibursio Ariza Tocancipa	37	0	0	15
<input type="checkbox"/> Erik Tibursio Montes Tocancipa	70	0	1 PG1478	16
<input type="checkbox"/> Fernando Tibursio Calvete Tocancipa	85	0	1 PG1199	10
<input type="checkbox"/> Julio Tibursio Perez Tocancipa	43	0	0	6
<input type="checkbox"/> Maika Tibursio Gambus Tocancipa	15	0	1 PG1355	4
<input type="checkbox"/> Manuel Tibursio Cabarcas Tocancipa	54	0	0	9
<input type="checkbox"/> Nicolas Tibursio Santos Tocancipa	28	0	0	3
<input type="checkbox"/> Olga Tibursio Ortiz Tocancipa	23	0	0	4
<input type="checkbox"/> Oscar Tibursio Vanegas Tocancipa	37	0	0	7
<input type="checkbox"/> Samuel Tibursio Muñoz Tocancipa	47	0	0	4

Interfaz de asignación de evaluadores para un trabajo de grado. Se puede apreciar a nivel de interfaz el poder seleccionar hasta 2 evaluadores para un proyecto en específico.

**Figura 29. Interfaz de usuario: Asignar hasta 6 miembros al comité de trabajos de grado**

Trabajos y Monografías / Sesión Comité T.G.

Sesión de Comité de Trabajos de Grado de Pregrado

**Información!**  
Existe una sesión de comité en curso, si desea puede continuar en la misma!

Quórum

- Olga Tibursio Ortiz Tocancipa
- Fernando Tibursio Calvete Tocancipa
- Manuel Tibursio Cabarcas Tocancipa
- Comité Tibursio Comité Tocancipa
- Comité Tibursio Comité Tocancipa
- MiembroPre1 Tibursio NoAsignadoPre1 Tocancipa

Hora: 22:57:02

Lugar:

[Continuar sesión](#)

Interfaz donde se retoma la sesión del comité de trabajos de grado. Se puede apreciar que ese comité en específico tiene 6 miembros.

## 4.9 MANTENIMIENTO Y ADMINISTRACIÓN

**4.7.1 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, que se corrigen durante la primera fase como integrantes del Grupo Calumet.

**4.7.2 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.

**4.7.3 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.
- 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 quices

Tabla 8. Pruebas realizadas: Servicio de quices

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Crear quiz	Crear un quiz usando los menús dentro del aula y corroborar la creación de este en BD.	✓
Visualizar quiz	Entrar con un usuario con rol de profesor y estudiante y corroborar la	✓

	presentación de los quices creados.	
Eliminar quiz	Usar el botón de la interfaz de profesor y corroborar en BD la eliminación de este registro.	✓
Volver a iniciar quiz	Usar el botón en la interfaz de profesor y corroborar el cambio en BD de la fecha de inicio del quiz.	✓
Habilitar quiz	Usar el botón en la interfaz de profesor y corroborar tanto en BD como con un usuario estudiante el inicio del quiz.	✓
Responder quiz	Ingresar con usuario estudiante y corroborar la visualización del quiz habilitado, envío de preguntas y finalización de este.	✓

### 5.1.1.2 Servicio formato de plan E3T

Tabla 9. Pruebas realizadas: Servicio formato de plan E3T

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Subir formato de plan	Ingresar como usuario súper administrador, subir un archivo con extensión pdf de máximo 20 MB y corroborar por ssh y en BD la existencia del archivo.	✓
Eliminar formato de plan	Como usuario súper administrador usar el botón y corroborar por medio de conexión ssh y en BD la eliminación del archivo del formato de plan.	✓
Descargar formato de plan	Como estudiante, súper administrador y profesor dar click en el botón y corroborar la descarga del archivo.	✓
Editar formato de plan	Como súper administrador usar el botón y corroborar por ssh y BD la eliminación del archivo anterior y la actualización por el archivo nuevo.	✓

### 5.1.1.3 Servicio de indicadores

**Tabla 10. Pruebas realizadas: Servicio de indicadores**

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Crear filtro de 3 niveles	Crear cada uno de los 3 filtros (área temática, categoría, y grupo de indicadores) y corroborar en BD los registros.	✓
Visualizar filtro de 3 niveles	Visualizar en la interfaz la correcta visualización de los 3 selects.	✓
Editar filtro de 3 niveles	Editar cada uno de los filtros y corroborar en BD los registros.	✓
Eliminar filtro de 3 niveles	Eliminar cada uno de los filtros y corroborar en BD la eliminación de los registros.	✓
Crear indicador	Crear un indicador y corroborar en BD los registros en todas las tablas implicadas.	✓
Visualizar indicador	Entrar a la interfaz de indicadores y corroborar la correcta visualización de estos.	✓
Editar indicador	Editar un indicador y corroborar en BD los	✓

	registros en todas las tablas implicadas.	
Eliminar indicador	Eliminar un indicador y corroborar en BD la eliminación de los registros en todas las tablas implicadas.	✓

#### 5.1.1.4 Servicio de gestión de archivo final

Tabla 11. Pruebas realizadas: Servicio de gestión de archivo final

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Subir documento final	Subir un documento en formato .pdf de no más de 50MB.	✓
Visualizar documento final	Descargar el archivo subido y visualizar su formato para validar que sea correcto.	✓

Actualizar documento final	Subir una nueva versión del documento en formato .pdf de no más de 50MB.	✓
Escribir comentarios	Escribir comentarios con rol de estudiante vs rol de calificador y director y visualizar el correcto despliegue de estos mensajes.	✓

#### 5.1.1.5 Asignar hasta 2 evaluadores a un trabajo de grado

Tabla 12. Pruebas realizadas: Asignar hasta 2 evaluadores aun trabajo de grado

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Asignar 1 evaluador a un trabajo de grado	Validar la asignación en interfaz y base de datos de un solo evaluador.	✓
Asignar 2 evaluador a un trabajo de grado	Validar la asignación en interfaz y base de datos de los dos evaluadores	✓

### 5.1.1.6 Asignar hasta 6 miembros al comité de trabajos de grado

Tabla 13. Pruebas realizadas: Asignar hasta 6 miembros al comité de trabajos de grado

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Iniciar sesión del comité con hasta 6 miembros	Validar el correcto inicio de sesión de un comité de 6 miembros en la interfaz y en base de datos.	✓
Visualizar datos del comité en el acta	Visualizar los datos de los 6 miembros en el acta pública generada.	✓
Finalizar sesión del comité con hasta 6 miembros	Validar que al finalizar el día se cierre la sesión correctamente y se genere el acta.	✓

## 6. CONCLUSIONES

- El servicio de quices añade un servicio esencial al aula virtual y brinda a los profesores que usan esta plataforma una herramienta para poder evaluar el grado de aprendizaje de sus estudiantes de una forma más rápida para así permitirles tomar acción de una manera más pronta.
- El servicio de formato de plan E3T brinda una herramienta bastante importante a las escuelas para poder informar a sus estudiantes y calificadores sobre las directrices a tomar al momento de crear/calificar un plan de trabajo de grado.
- El servicio de visualización del plan actual en la sección de “calificar planes” facilita el acceso a estos mejorando así la eficiencia de este proceso.
- La mejora de poder asignar 2 evaluadores a un trabajo de grado cumplió con las expectativas funcionales que requería la escuela de petróleos.
- La mejora de poder asignar 6 miembros al comité de trabajo de grado cumplió con las expectativas funcionales que requería la escuela de petróleos.
- El servicio de subida de archivo final por parte de los estudiantes (que por ahora es opcional para los estudiantes) cumplió con los requisitos funcionales planteados.
- El servicio de indicadores les permite a los grupos de las escuelas representar grandes cantidades de datos en gráficos que permiten una transmisión y un entendimiento mucho más eficaz de los datos presentados.

- La plataforma COMA gracias al trabajo continuo del grupo CALUMET sigue en expansión y mejora buscando adaptarse y proveer los servicios que las escuelas requieren con el paso del tiempo.

## BIBLIOGRAFÍA

*Apache Tomcat* [en Línea]. [Fecha de consulta: 17 noviembre 2020]. Disponible en: <http://tomcat.apache.org>

Docs.oracle.com [en Línea]. *JavaServer Pages Documents - The Java EE 5 Tutorial*, 2020. [Fecha de consulta: 23 noviembre 2020]. Disponible en: <https://docs.oracle.com/javaee/5/tutorial/doc/bnajo.html>

Developer.mozilla.org [en Línea]. [Fecha de consulta: 10 noviembre 2020]. Disponible en: <https://developer.mozilla.org/es>

Es.wikipedia.org [en Línea]. [Fecha de consulta: 10 noviembre 2020]. Disponible en: <https://es.wikipedia.org>

Infranetworking [en Línea]. *Modelo cliente servidor: ¿Qué es? Características, Ventajas y Desventajas*, 2021. [Fecha de consulta: 3 noviembre 2020]. Disponible en: <https://blog.infranetworking.com/modelo-cliente-servidor>

Medium [en Línea]. [Fecha de consulta: 10 noviembre 2020]. Disponible en: <https://medium.com/javascript-scene/master-the-javascript-interview-what-is-functional-programming-7f218c68b3a0>

MySQL [en Línea]. *MySQL: MySQL Documentation*, 2020. [Fecha de consulta: 17 noviembre 2020]. Disponible en: <https://dev.mysql.com/doc/>

SSH.COM [en Línea]. *SSH.COM - Protecting your critical data*, 2020. [Fecha de consulta: 23 noviembre 2021]. Disponible en: <https://www.ssh.com>

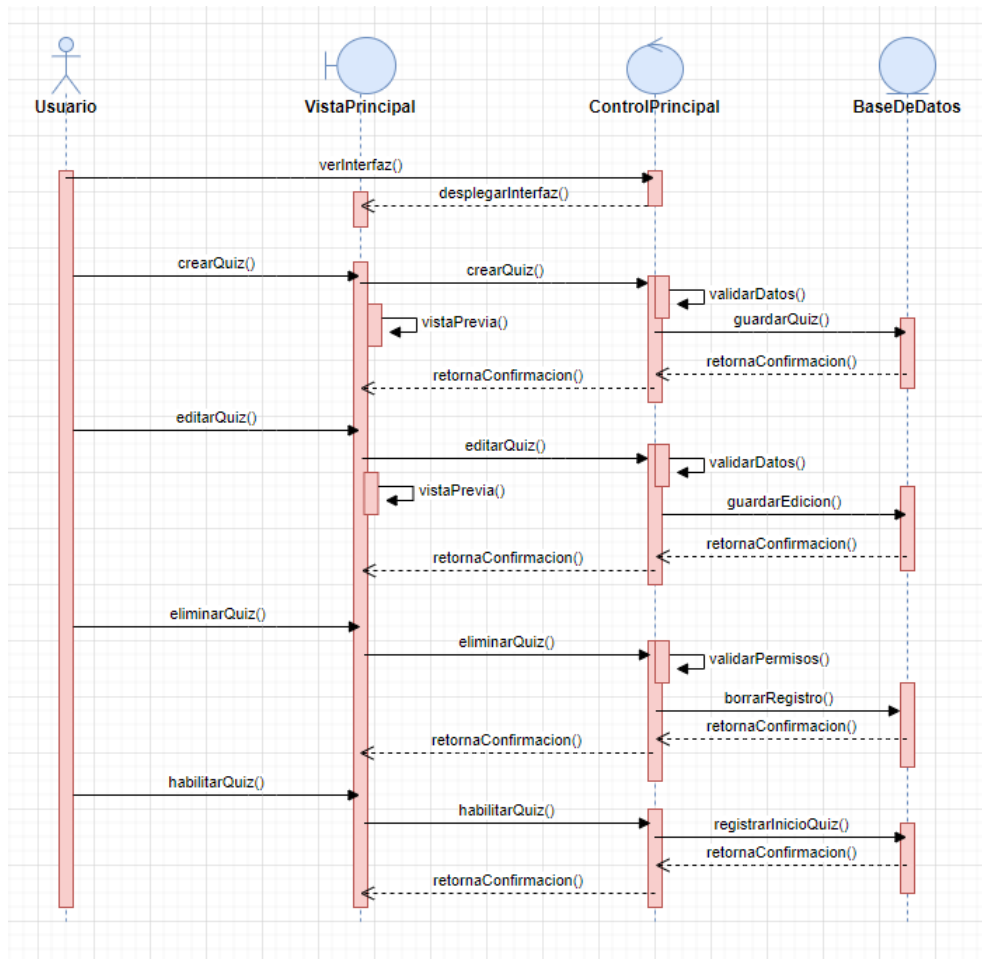
Stack Overflow en español [en Línea]. [Fecha de consulta: 11 noviembre 2020].  
Disponible en: <https://es.stackoverflow.com>

## ANEXOS

### Anexo A. Modelo de Procesos del Sistema

#### Servicio de quices

Figura 30. Diagrama de secuencia: Servicio de quices

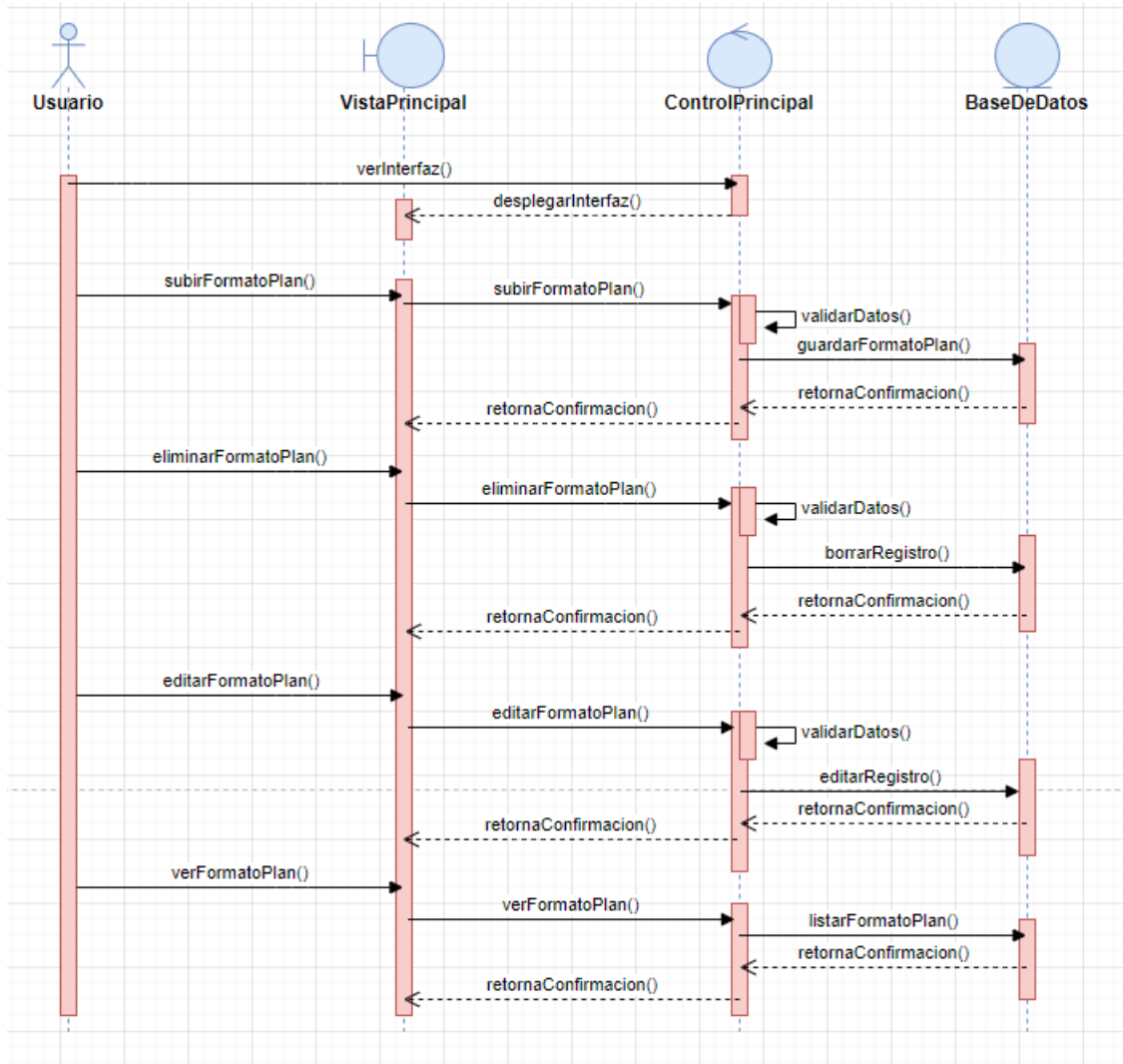


1. Cuando el usuario ingresa al aula virtual en la sección de foros se despliega la interfaz con los botones para gestionar los quices.
2. En esta interfaz el usuario usa el botón para crear quiz y llena el formulario.

3. El controlador principal valida los datos del formulario y solicita guardar el quiz en base de datos.
4. La base de datos guarda el registro y retorna el estado de la operación, el controlador recibe este estado y lo retorna a la vista principal.
5. En esta interfaz el usuario usa el botón de editar quiz y modifica el formulario.
6. El controlador principal valida los datos del formulario y solicita editar el quiz en base de datos.
7. La base de datos actualizar el registro y retorna el estado de la operación, el controlador recibe este estado y lo retorna a la vista principal.
8. En esta interfaz el usuario usa el botón de eliminar quiz y acepta el mensaje de confirmación.
9. El controlador principal valida los permisos del usuario y solicita eliminar el quiz en base de datos.
10. La base de datos elimina el registro y retorna el estado de la operación, el controlador recibe este estado y lo retorna a la vista principal.
11. En esta interfaz el usuario usa el botón de habilitar quiz (solo para quices que no se han hecho).
12. El controlador principal solicita habilitar el quiz en base de datos.
13. La base de datos actualiza el estado del registro del quiz para habilitarlo y retorna el estado de la operación, el controlador recibe este estado y lo retorna a la vista principal.

## Servicio formato de plan E3T

Figura 31. Diagrama de secuencia: Servicio formato de plan



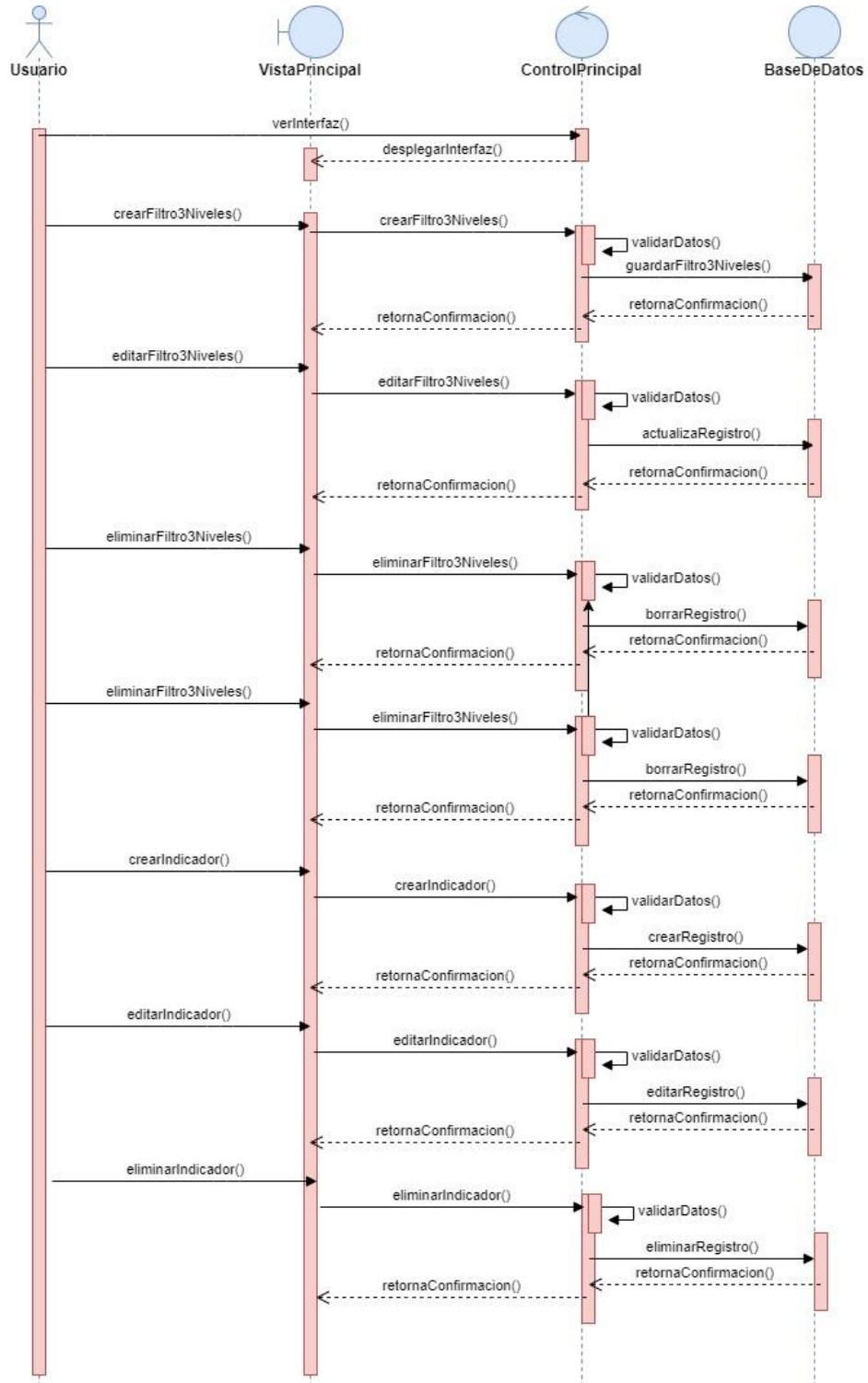
1. Cuando el usuario súper administrador ingresa a la sección principal de cormorán en la sección de gestión formato de plan se despliega la interfaz con los botones para gestionar los formatos de pregrado y posgrado.

2. En esta interfaz el usuario súper administrador usa el botón subir formato de plan (ya sea en pregrado o posgrado) y selecciona un plan en formato pdf.
3. El controlador principal valida el formato, peso máximo y nombre del archivo y lo guarda en una carpeta de archivos, después solicita guardar el registro en base de datos.
4. La base de datos guarda el registro y retorna el estado de la operación al controlador y este lo retorna a la vista principal.
5. En esta interfaz el usuario súper administrador usa el botón eliminar formato de plan (ya sea en pregrado o posgrado).
6. El controlador principal valida el tipo de usuario y eliminar el archivo de la carpeta de archivos, después solicita eliminar el registro en base de datos.
7. La base de datos borra el registro y retorna el estado de la operación al controlador y este lo retorna a la vista principal.
8. En esta interfaz el usuario súper administrador usa el botón subir formato de plan (ya sea en pregrado o posgrado) en el caso de que ya halla un formato subido y selecciona un nuevo plan en formato pdf.
9. El controlador principal valida el formato, peso máximo y nombre del archivo y lo guarda en una carpeta de archivos, después solicita guardar el registro en base de datos.
10. La base de datos guarda el registro y retorna el estado de la operación al controlador.
11. Si la operación fue exitosa se elimina el archivo anterior y se retorna el estado de la transacción a la vista principal, sino entonces solo se hace lo último.
12. En la interfaz del estudiante “Subir formato plan” o en la de “calificar plan” para el usuario calificador.
13. Al momento de ingresar a estas interfaces en caso de existir un formato de plan se despliega un botón para descargar el formato.
14. El usuario usa el botón de descargar formato de plan.
15. El controlador principal solicita el archivo en base de datos.

16. La base de datos retorna el estado de la operación al controlador y este lo retorna a la vista.

## Servicio de indicadores

Figura 32. Diagrama de secuencia: Servicio de indicadores

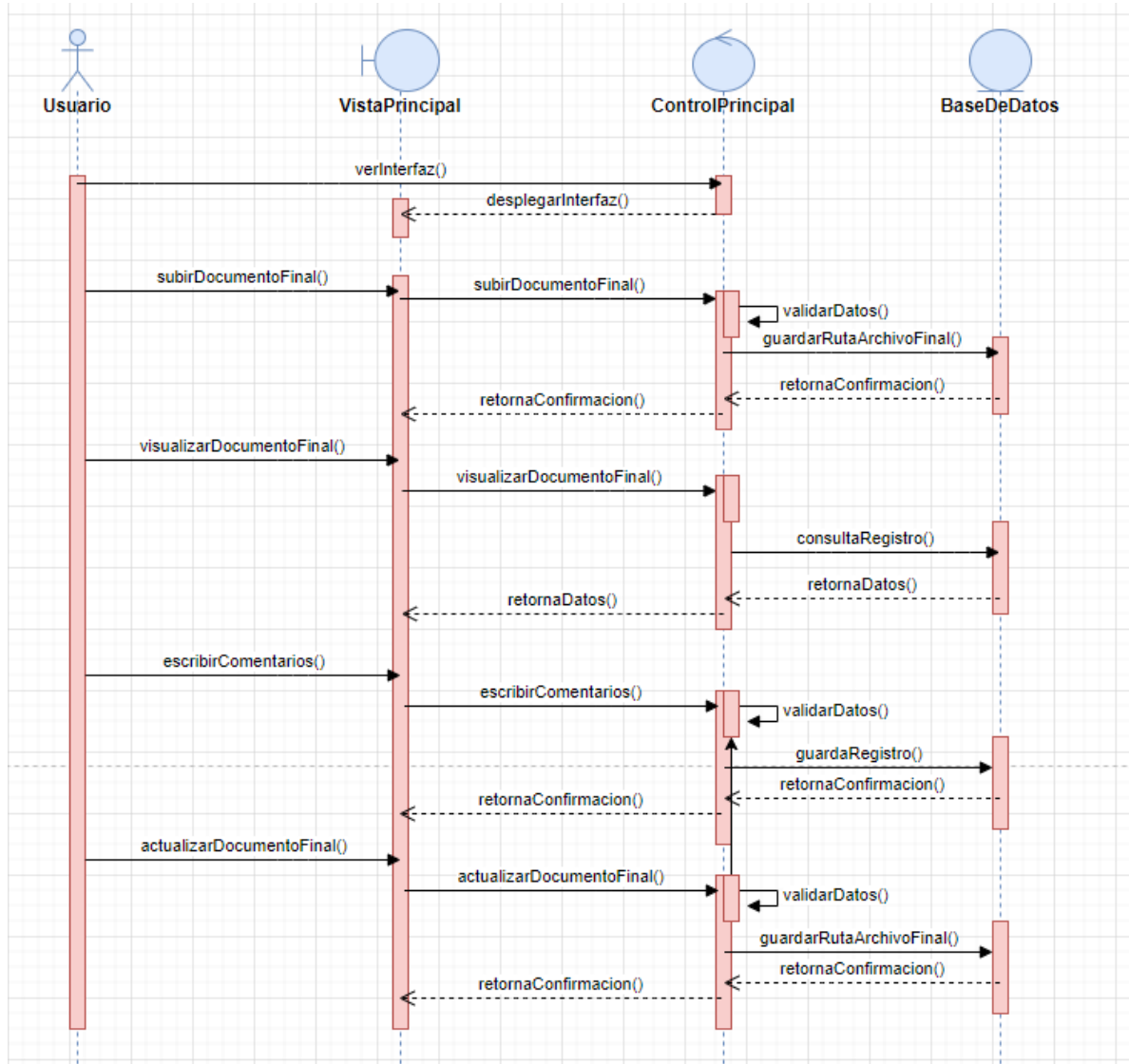


1. Cuando el usuario ingresa a la sección principal de grupos en la sección de gestión de indicadores se despliega la interfaz con los botones para gestionar los indicadores.
2. En esta interfaz el usuario registra en los formularios los 3 niveles de categorización de un indicador: área temática, categoría y grupo de indicadores
3. El controlador principal valida los datos enviados y solicita a la base de datos guardar los registros.
4. La base de datos guarda los registros y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.
5. En esta interfaz el usuario edita en los formularios uno o varios niveles de categorización de un indicador: área temática, categoría y grupo de indicadores.
6. El controlador principal valida los datos enviados y solicita a la base de datos actualizar los registros.
7. La base de datos actualiza los registros y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.
8. En esta interfaz el usuario elimina en los formularios uno o varios niveles de categorización de un indicador: área temática, categoría y grupo de indicadores.
9. El controlador principal valida que ese nivel de categorización no tenga ningún indicador asociado y solicita a la base de datos eliminar los registros.
10. La base de datos borra los registros y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.
11. En esta interfaz el usuario usa el botón de crear indicador y registra en los formularios los datos para crear un indicador.
12. El controlador principal valida los datos enviados y solicita a la base de datos guardar el registro del indicador.
13. La base de datos crea el registro y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.
14. En esta interfaz el usuario usa el botón de editar indicador y registra en los formularios los datos para editar un indicador.

15. El controlador principal valida los datos enviados y solicita a la base de datos editar el registro del indicador.
16. La base de datos actualiza los registros y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.
17. En esta interfaz el usuario usa el botón de eliminar indicador y busca en un select el indicador a eliminar.
18. El controlador principal valida los permisos del usuario y solicita a la base de datos borrar el registro del indicador.
19. La base de datos borra los registros y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.

### **Servicio de gestión de archivo final**

Figura 33. Diagrama de secuencia: Gestión de archivo final



1. Cuando el usuario ingresa a la sección principal de grupos en la sección de gestión de archivo final se despliega la interfaz.
2. En esta interfaz el usuario sube su documento en formato .pdf.
3. El controlador principal valida los datos enviados y solicita a la base de datos guardar los registros.
4. La base de datos guarda los registros y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.

5. En esta interfaz el usuario descargar el archivo final para visualizarlo.
6. En esta interfaz el usuario puede escribir comentarios en un chat dispuesto ya sea con el director o los calificadores.
7. El controlador principal valida los datos enviados y solicita a la base de datos guardar los registros.
8. La base de datos guarda los registros y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.
9. En esta interfaz el usuario puede subir nuevas versiones del documento en formato .pdf.
10. El controlador principal valida los datos enviados y solicita a la base de datos guardar los registros.
11. La base de datos guarda los registros y retorna el estado de la operación al controlador principal y este a su vez retorna este estado a la vista principal.