

Diseño de una plataforma transmedia que permita la gestión de trabajos de capstone design

Leonardo Marín Ortiz
Denis Hernán Franco Arias

Trabajo de grado para optar al título de Ingeniero Electrónico.

Director
Homero Ortega Boada
Doctor en ciencias de la ingeniería, radiocomunicaciones

Co-Director
Carlos Mario Martínez Castro
Ingeniero electrónico

Universidad Industrial de Santander
Facultad de ingeniería físico-mecánicas
Escuela de ingeniería eléctrica, electrónica y de telecomunicaciones
Ingeniería Electrónica
Bucaramanga

2024

Agradecimientos.

Leonardo Marín Ortiz.

A mis padres, por ser la fuerza constante detrás de cada uno de mis logros, enseñándome el valor de la perseverancia y apoyándome incluso en los momentos de duda. Este proyecto es tanto mío como suyo, porque sin su amor y dedicación nada de esto sería posible. A mi hermana, por ser mi compañera de vida, dándome palabras de aliento y recordándome en cada paso de este camino que nunca estoy solo. Y a mi pareja, cuyo amor, paciencia y comprensión han sido un pilar en cada etapa de este proyecto; gracias por escucharme, aconsejarme y ofrecerme una perspectiva única que me ha impulsado a crecer y avanzar con seguridad. No puedo expresar lo agradecido que estoy por tu apoyo y por brindarme la tranquilidad necesaria para concentrarme y dar lo mejor de mí.

Denis Hernán Franco Arias.

Agradezco primero a la vida, a Dios, y a mis padres, quienes fueron mi motor durante este proceso, quienes siempre confiaron en que se podría lograr las metas, En mis hermanos, quienes siempre me brindaron las mejores palabras de apoyo y nunca dejaron que me hiciera falta algo. En mi pareja quien fue mi bastón durante los momentos más difícil, y quien siempre estuvo en disposición de ayudarme y escucharme cuando lo necesita, En mis amigos de carrera, cómplices del aprendizaje adquirido durante esta etapa universitaria, En compañeros de Danzas UIS, quienes fueron una gran oportunidad de crecer en ámbitos extracurriculares, y seguramente más que amigos, se terminan volviendo parte de una familia. Y por último a la UIS, por brindarme tantos espacios de aprendizaje y formación de calidad.

Tabla de contenido

Introducción.....	9
1. Objetivos.....	10
1.1. Objetivo general.....	10
1.2. Objetivos específicos.....	10
2. Marco teórico.....	10
2.1 AWS.....	11
2.2 Capstone Design.....	11
2.3 EC2 (Elastic Compute Cloud).....	11
2.4 Procedimiento para la presentación de trabajos de grado en la E3T.....	12
2.5 Web service.....	12
3. Desarrollo de solución.....	13
3.1 Modelo de negocio.....	13
3.1.1 Interesados.....	13
3.1.2 Identificación de Necesidades.....	13
3.1.3 Visión a Futuro.....	14
3.1.4 Lógica de Negocio.....	15
3.2 Implementación.....	26
3.2.1 Arquitectura del sistema.....	26
3.2.2 Modelo de Capas del Sistema.....	27
3.2.3 Elección de figma para el diseño frontend.....	28
3.2.4 Diseño del backend.....	28
3.2.5 Elección de una base de datos relacional (MySQL).....	33
3.2.6 Web Service para Subida de Archivos a Google Drive.....	34
3.2.7 Implementación AWS.....	35

PLATAFORMA TRANSMEDIA	4
3.3 Experiencia de usuario.....	36
3.4 Validación y verificación.....	39
3.4.1 Metodología de pruebas.....	39
3.4.2 Plan de pruebas.....	39
3.4.3 Resultados de las pruebas.....	41
4. Conclusiones.....	43
5. Recomendaciones.....	44
Referencias Bibliográficas.....	46

Lista de tablas.

Tabla1. 1 Plan de pruebas para realizar en la página.	39
--	----

Lista de figuras

Figura 1. Interesados.	13
Figura 2. Diagrama de la visión a futuro.	14
Figura 3. Lógica del negocio.	15
Figura 4 Diagrama de clases.	17
Figura 5 Diagrama de casos No registrado/Estudiante.	18
Figura 6 Diagrama de uso Administrador/Docente.	19
Figura 7 Diagrama de fase 1.	20
Figura 8 Diagrama de fase 2.	21
Figura 9 Diagrama de fase 3.	22
Figura 10 Etapas del proyecto de grado.	22
Figura 11 Evaluación Fase 1.	23
Figura 12 Evaluación Fase 2.	24
Figura 13 Arquitectura del prototipo.	26
Figura 14 Desarrollo backend.	28
Figura 15 Visor phpMyAdmin.	34
Figura 16 Vista inicio de sesión.	37
Figura 17 Perfil estudiante.	37
Figura 18 Pestaña trabajo de grado.	38
Figura 19 Pestaña estadística. Por estudiante.	38

Lista de apéndices.

(Los apéndices pueden ser consultados en la base de datos de la Biblioteca UIS)

Apéndice A. Marco teórico.

Apéndice B. Marco legal.

Apéndice C. Procedimiento para la presentación de trabajos de grado en la E3T.

Apéndice D. Prototipo inicial desarrollado en figma.

Apéndice E. Por qué se escogió la base de datos MYSQL.

Apéndice F. Web service (Homero Ortega, 2024).

Apéndice G. AWS EC2.

Apéndice H. Interfaz de usuario completa.

Apéndice I. Plan de pruebas y ejecución de pruebas del sistema.

Apéndice J. Diagrama entidad-relación.

Apéndice K. Propuesta de mejora para la creación automática de páginas web de proyectos grado de la E3T.

Resumen

Título: Diseño de una plataforma transmedia que permita la gestión de trabajos de capstone design*

Autores: Leonardo Marín Ortiz, Denis Hernán Franco Arias**

Palabras claves: Capstone design, ABET, Plataforma transmedia.

Descripción: Este proyecto desarrolla una innovadora plataforma transmedia diseñada para transformar el proceso de Capstone Design en la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones de la Universidad Industrial de Santander. La carencia de herramientas tecnológicas efectivas, que obstaculizan la gestión de proyectos de grado, ha sido el motor detrás de esta propuesta. La plataforma no solo facilita la selección de temas y la planificación de los proyectos, sino que también optimiza la entrega final de los trabajos. Usando tecnologías como HTML, CSS, JavaScript y PHP, junto con el respaldo de AWS EC2, la plataforma ofrece una experiencia intuitiva y dinámica para los estudiantes y un control eficiente para los docentes. Entre los resultados se destacan una mejor organización y seguimiento de los avances, favoreciendo tanto el rendimiento académico de los estudiantes como la labor docente. Esta solución promete un impacto significativo, impulsando la eficiencia y modernizando el proceso académico, alineándose con las demandas tecnológicas actuales.

*Degree work

**Faculty of Physical-mechanical engineering. School of electrical, electronic and telecommunications engineering. Director: Homero Ortega Boada Academic title: Philosophy Doctor (PhD) of Engineering Science. Co-director: Carlos Mario Martínez Castro. Engineering Electronic.

Abstract

Title: Design of a Transmedia Platform for Managing Capstone Design Projects

Authors: Leonardo Marín Ortiz, Denis Hernán Franco Arias

Keywords: Capstone design, ABET, Transmedia platform

Description: This project develops an innovative transmedia platform designed to transform the Capstone Design process at the School of Electrical, Electronics, and Telecommunications Engineering at the Universidad Industrial de Santander. The lack of effective technological tools, which obstructs the management of final-year projects, has been the driving force behind this proposal. The platform not only facilitates the selection of topics and project planning but also optimizes the final submission of the work. Using technologies like HTML, CSS, JavaScript, and PHP, along with AWS EC2, the platform provides an intuitive and dynamic experience for students and efficient control for teachers. Among the results, improved organization and progress tracking stand out, benefiting both student academic performance and teaching. This solution promises a significant impact by boosting efficiency and modernizing the academic process, aligning it with current technological demands.

*Degree work

**Faculty of Physical-mechanical engineering. School of electrical, electronic and telecommunications engineering. Director: Homero Ortega Boada Academic title: Philosophy Doctor (PhD) of Engineering Science. Co-director: Carlos Mario Martínez Castro. Engineering Electronic.

Introducción.

La acreditación internacional ABET, un estándar global de calidad educativa, es uno de los mayores logros que un programa académico puede obtener. Universidades de prestigio mundial como Harvard, Stanford, MIT y UC Berkeley cuentan con esta acreditación, que valida la excelencia de sus programas de ingeniería. En Colombia, la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T) de la Universidad Industrial de Santander ha alcanzado este reconocimiento y se esfuerza por mantenerlo. Un componente clave de este proceso de re-acreditación es el cumplimiento del criterio 5 de ABET, también conocido como Major Engineering Design Experience (Capstone Design), el cual exige que los estudiantes demuestren sus habilidades en un proyecto de diseño integral al final de sus estudios.

Cuando los estudiantes de la E3T llegan a la culminación de sus materias, deben implementar el Capstone Design para culminar sus estudios de forma satisfactoria. Sin embargo, la E3T enfrenta dificultades para gestionar y conservar evidencias de los procesos del Major Engineering Design Experience, como la selección del tema, las entregas de plan, y la sustentación final, aspectos cruciales para la re-acreditación. Además, el docente encargado del proceso maneja más de 150 estudiantes, lo que genera una alta demanda de orientación acerca de los pasos a seguir para superar cada fase del proceso.

Ante estos desafíos, proponemos el diseño y desarrollo de una plataforma transmedia que facilite la gestión de los proyectos del Capstone Design de manera eficiente e interactiva, permitiendo una adecuada gestión de la información. Esta plataforma no sólo visibiliza el trabajo personal de cada estudiante o equipo, sino que también fomenta el aprendizaje colaborativo al ofrecer herramientas para la comparación y evaluación de enfoques y soluciones adoptados por otros estudiantes, desde la fase de registro del tema hasta la sustentación final del proyecto.

Con esta solución tecnológica, se busca no solo optimizar el proceso académico de los estudiantes de la E3T, sino también ofrecer una herramienta moderna que conserve la evidencia necesaria para los procesos de acreditación ABET, asegurando así el éxito académico y profesional de toda la comunidad universitaria.

1. Objetivos.

1.1. Objetivo general.

Desarrollar una plataforma transmedia que permita gestionar los trabajos de Capstone Design para estudiantes de E3T durante todo el proceso de desarrollo, desde las primeras fases de registro de tema, hasta la entrega final y sustentación.

1.2. Objetivos específicos.

- Establecer las necesidades del sistema mediante un estudio a nivel de casos de uso e historias de usuario con los interesados.
- Diseñar un prototipo de solución teniendo en cuenta las necesidades establecidas anteriormente.
- Implementar el prototipo diseñado considerando las limitaciones del sistema establecidas.
- Validar mediante pruebas unitarias el prototipo desarrollado.

2. Marco teórico.

En la educación superior, la recepción y evaluación de proyectos de grado son clave en la formación profesional de los estudiantes. Para mejorar estos procesos y enriquecer la experiencia estudiantil, nuestro proyecto se centrará en el diseño y desarrollo de una plataforma web que cumpla con los rigurosos estándares de calidad establecidos por la Agencia ABET (Accreditation Board for Engineering and Technology). Esta plataforma utilizará tecnologías modernas, incluyendo HTML, CSS, JavaScript, Python, PHP y MySQL, para implementar una interfaz de usuario interactiva y receptiva. HTML y CSS serán empleados para la estructura y diseño del frontend, mientras que JavaScript añadirá interactividad y dinamismo. PHP se encargará de la recepción y procesamiento de datos, junto a gestionar la lógica del servidor. MySQL proporcionará una gestión eficiente y escalable de la base de datos de la plataforma, garantizando la integridad y disponibilidad de la información relacionada con los proyectos de grado. Para tener claridad sobre algunos de estos conceptos puntuales, visitar **Apéndice A**.

Para entender el desarrollo de este informe necesitamos tener conocimientos de algunas palabras claves que son importantes.

2.1 AWS.

Es una plataforma de servicios en la nube ofrecida por Amazon, que permite a las empresas y desarrolladores acceder a una amplia gama de recursos computacionales de manera flexible y escalable, sin necesidad de gestionar hardware físico. Aunque el concepto de computación en la nube puede ser familiar para muchos ingenieros, algunos aspectos técnicos de AWS podrían no ser obvios para todos.

Conceptos clave que necesitan aclaración:

- **Escalabilidad en la nube:** AWS permite ajustar dinámicamente los recursos utilizados, algo útil para aplicaciones web con fluctuaciones en el número de usuarios.
- **Servicios distribuidos globalmente:** AWS opera en diversas regiones, permitiendo desplegar aplicaciones con baja latencia y alta disponibilidad, un aspecto que puede no ser común en infraestructuras locales. (Golden B. 2013).

2.2 Capstone Design.

Es una experiencia integradora que culmina el proceso formativo en los programas de ingeniería. Este proyecto final permite a los estudiantes aplicar tanto los conocimientos técnicos como los no técnicos adquiridos a lo largo de su carrera para abordar problemas complejos, teniendo en cuenta restricciones realistas como las económicas, ambientales, sociales y éticas. Además, el proyecto prepara a los estudiantes para el entorno laboral real, fomentando el desarrollo de habilidades esenciales en trabajo en equipo, diseño, comunicación y toma de decisiones, que son cruciales para su éxito profesional.

2.3 EC2 (Elastic Compute Cloud).

EC2 es uno de los servicios más importantes de AWS que permite alquilar servidores virtuales (instancias) para ejecutar aplicaciones. Si bien los ingenieros suelen estar familiarizados con servidores, el concepto de servidores virtuales bajo demanda y su flexibilidad en la nube puede no ser inmediatamente claro.

Conceptos clave que necesitan aclaración:

- **Instancias bajo demanda:** A diferencia de los servidores físicos, EC2 permite alquilar servidores en función de la demanda en tiempo real, lo cual es crucial para proyectos web con necesidades fluctuantes.
- **Control y personalización:** EC2 ofrece control total sobre las instancias, lo que incluye acceso root y la configuración del servidor según las necesidades del proyecto.
- **Seguridad en la nube:** AWS ofrece medidas de seguridad avanzadas, como grupos de seguridad y cifrado de datos, que son clave para la protección de los sistemas.

En este proyecto, EC2 proporciona la infraestructura para alojar la aplicación web y escalarla según sea necesario, manteniendo el rendimiento a medida que el número de usuarios crece. (Golden B. 2013).

2.4 Procedimiento para la presentación de trabajos de grado en la E3T.

Se identificaron las necesidades y los problemas relacionados con la gestión de los trabajos de grado en la E3T. A través de reuniones con docentes y un estudio detallado del proceso, se propuso una solución que abarca desde el registro de temas hasta la entrega final del proyecto. Para más detalles sobre este análisis, revisar el **Apéndice B**, correspondiente.

2.5 Web service.

Un servicio web es una tecnología que permite a las aplicaciones intercambiar datos a través de la web, usando protocolos como HTTP y formatos como JSON o XML. Entre los conceptos clave están:

- **REST (Transferencia de Estado Representacional):** Un estilo arquitectónico que utiliza HTTP para realizar operaciones CRUD sobre recursos en formatos JSON o XML.
- **API (Interfaz de Programación de Aplicaciones):** Facilita la comunicación entre aplicaciones mediante servicios web.
- **Interoperabilidad y escalabilidad:** Los servicios web permiten integrar aplicaciones en distintos lenguajes y plataformas. (Masse M, 2020).

Es fundamental consultar el **Apéndice C**, que contiene el marco legal aplicable al proyecto. Este documento incluye las normativas y estándares que debemos considerar para asegurar que el desarrollo de la plataforma web cumpla con las regulaciones pertinentes.

3. Desarrollo de solución.

3.1 Modelo de negocio.

3.1.1 Interesados.

Figura 1.

Interesados.



Nota: interesados plataforma transmedia.

3.1.2 Identificación de Necesidades.

La plataforma se creó para abordar problemas específicos que enfrentan tanto estudiantes como docentes en la gestión de proyectos de grado:

- **Estudiantes:** Dificultades para organizar y gestionar sus proyectos en un solo lugar. Necesidad de recibir retroalimentación y evaluaciones de forma eficiente y accesible. Seguimiento deficiente de las diferentes fases del proyecto.

- **Docentes:** Carga administrativa pesada al gestionar múltiples proyectos, archivos y estudiantes. Falta de una herramienta que centralice toda la información y facilite la revisión y evaluación de los proyectos de grado.
- **Universidad:** Necesidad de un sistema que centralice los proyectos de grado, asegurando un control y acceso organizados a la información. Dificultades para auditar y verificar los progresos de los estudiantes de manera ágil.

3.1.3 Visión a Futuro.

Figura 2.

Diagrama de la visión a futuro.



Nota: Visión a futuro.

La plataforma tiene un gran potencial para evolucionar, adaptándose tanto a las nuevas demandas tecnológicas como a los cambios en el entorno académico.

- **Escalabilidad:** Posibilidad de integrar más funcionalidades como módulos de trabajo colaborativo o inteligencia artificial para mejorar la retroalimentación.
- **Mercado Global:** Expansión de la plataforma como un servicio para otras universidades, aprovechando su flexibilidad como **SaaS** (Software as a Service).
- **Reforma Académica:** La plataforma está diseñada para seguir siendo funcional ante reformas académicas, permitiendo la implementación de nuevas metodologías y enfoques educativos. Su capacidad de adaptación facilita la gestión del avance y el seguimiento de los estudiantes a lo largo de sus proyectos, asegurando así su relevancia y eficacia incluso con la introducción de cambios en el currículo.

3.1.4 Lógica de Negocio.

Figura 3.

Lógica del negocio.



Nota: Diseño de la lógica del sistema

3.1.4.1 Usuarios.

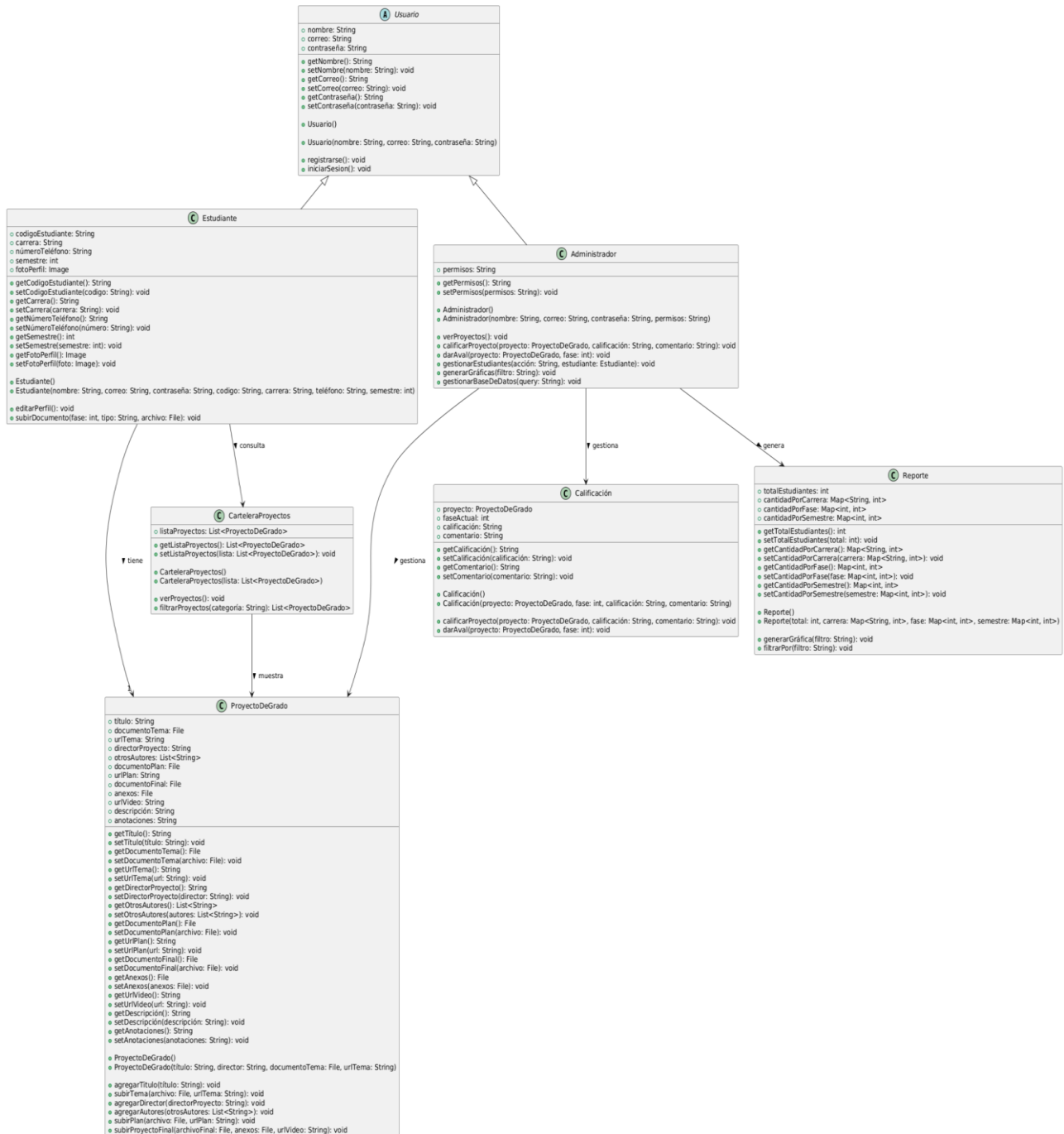
El modelo de negocio está fundamentado en una lógica estructurada basada en UML, que facilita la representación clara de las funcionalidades del sistema y sus requisitos. A través de los

diagramas de clases y casos de uso, es posible visualizar de manera estandarizada las interacciones entre los diferentes actores del sistema y sus respectivas funciones, permitiendo una mejor comprensión de la complejidad del sistema.

A continuación, se describen brevemente las clases presentes en el diagrama:

- **Usuario (Clase Abstracta):** Representa la base para los distintos tipos de usuarios del sistema. Incluye atributos como nombre, correo, y contraseña, así como métodos fundamentales para el registro e inicio de sesión.
- **Estudiante (Hereda de Usuario):** Representa a los estudiantes que utilizan la plataforma. Además de los atributos heredados, incluye codigoEstudiante, carrera, númeroTeléfono, semestre, y fotoPerfil. Métodos como editarPerfil() y subirDocumento() permiten a los estudiantes gestionar su información y proyectos.
- **Administrador (Hereda de Usuario):** Representa a los docentes o administradores del sistema. Posee un atributo adicional permisos que define su nivel de acceso. Sus métodos permiten gestionar proyectos, calificar y dar aval a los estudiantes, así como generar reportes y gráficas.
- **ProyectoDeGrado:** Representa los trabajos de grado que los estudiantes presentan. Contiene atributos como título, documentoTema, urlTema, directorProyecto, y otrosAutores, así como métodos que permiten agregar títulos, subir documentos y gestionar la información del proyecto.
- **Calificación:** Representa la evaluación de un proyecto de grado en una fase específica. Incluye atributos como proyecto, faseActual, calificación, y comentario, con métodos para calificar y dar aval a un proyecto.
- **CarteleraProyectos:** Muestra la lista de proyectos disponibles para que los estudiantes los consulten. Contiene un atributo listaProyectos, y métodos para ver y filtrar proyectos según categorías específicas.
- **Reporte:** Genera estadísticas y reportes sobre los estudiantes y proyectos. Sus atributos incluyen totalEstudiantes, cantidadPorCarrera, cantidadPorFase, y cantidadPorSemestre, con métodos para generar gráficas y filtrar información.

Figura 4
Diagrama de clases.



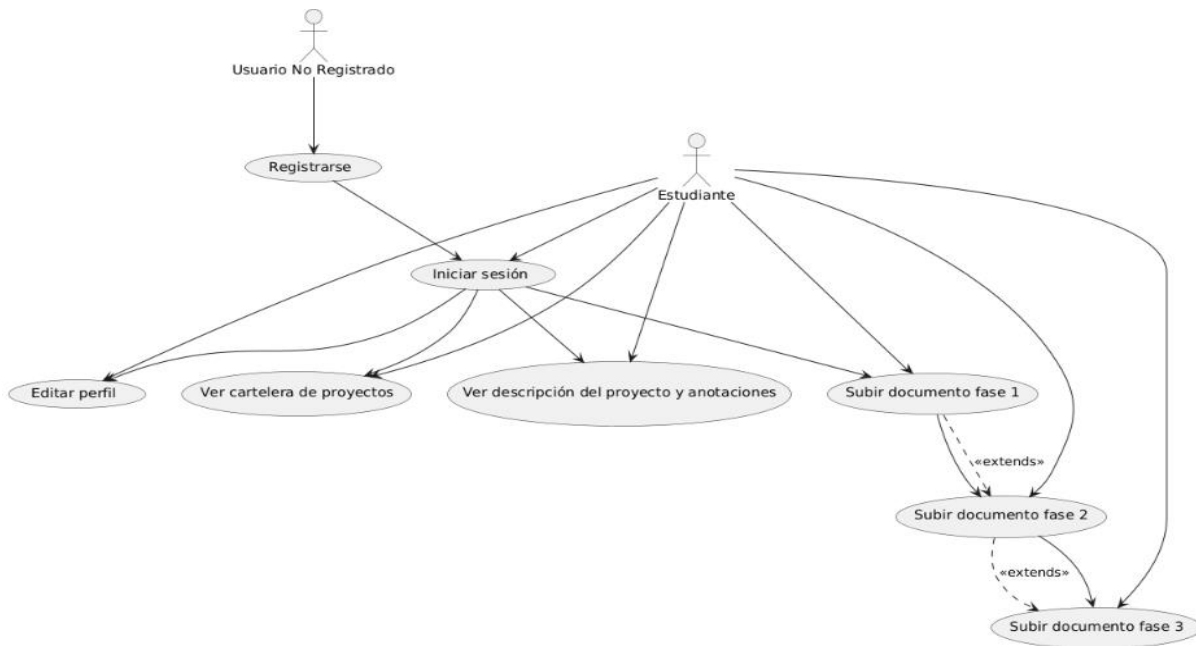
Nota: Diagrama donde se expone de manera técnica el funcionamiento de las clases, y atributos de la página Fuente: autores

- **Casos de uso**

Los diagramas de casos de uso son esenciales en el análisis y diseño de sistemas, ya que permiten visualizar los actores que interactúan con la aplicación y las acciones que pueden realizar. Son fáciles de entender, lo que facilita la comunicación entre clientes y usuarios, y representan claramente los requisitos funcionales. Esto asegura que el desarrollo del software se enfoque en las necesidades reales, favoreciendo una aplicación alineada con los objetivos del proyecto.

Figura 5

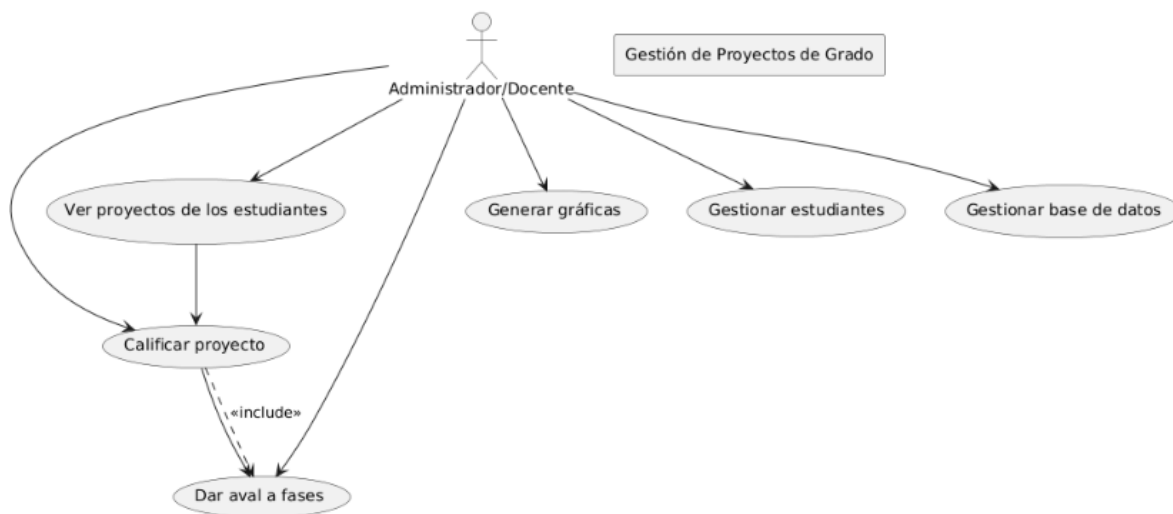
Diagrama de casos No registrado/Estudiante.



Nota: Diagrama donde se expone de manera técnica el funcionamiento de los estudiantes y los usuarios no registrados. Fuente: autores.

Figura 6

Diagrama de uso Administrador/Docente.



Nota: Diagrama donde se expone de manera técnica el funcionamiento del administrador. Fuente: autores.

4.1.4.2 Proyecto de grado (Fases).

El proyecto se gestiona en tres fases, facilitando el seguimiento y retroalimentación continua entre estudiantes y docente, desde la definición del proyecto hasta la entrega final.

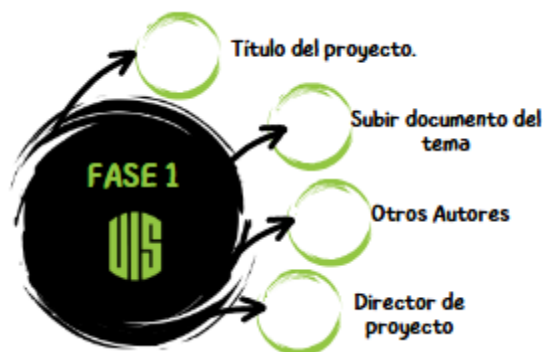
Fase 1: Definición del Proyecto

En esta fase, los estudiantes establecen los fundamentos del proyecto de grado, definiendo el tema y alcance. Los componentes clave incluyen:

- **Título del Proyecto:** Campo para ingresar el nombre del proyecto.
- **Subir Documento del Tema:** Documento PDF que detalla el objetivo, justificación, restricciones, normas aplicables, múltiples soluciones, impactos, y habilidades requeridas, modalidad del proyecto, entidad interesada.
- **Director del Proyecto:** Nombre del profesor supervisor.
- **Otros Autores:** Opción de agregar compañeros al proyecto.

Figura 7

Diagrama de fase 1.



Nota: Descripción de la Fase 1.

Esta fase es crítica porque establece la base sobre la cual se construirá todo el trabajo posterior. Una vez que el profesor / administrador haya aprobado esta fase, los estudiantes pueden proceder a la siguiente.

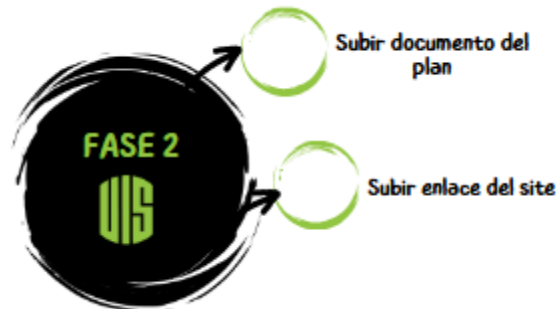
Fase 2: Desarrollo del Plan del Proyecto y página web:

En esta fase, los estudiantes elaboran un plan detallado para su proyecto de grado, que incluye:

Documento del Plan de Proyecto debe incluir:

- **Introducción:** Importancia del proyecto.
- **Definición del Problema:** Presentación del problema a abordar.
- **Estudios Previos:** Análisis de literatura y antecedentes relevantes.
- **Ideación:** Generación y selección de soluciones.
- **Cronograma:** Fechas de actividades.
- **Presupuesto:** Estimación de recursos financieros.
- **Bibliografía:** Fuentes consultadas.

Enlace a la Página Web: Un enlace a la página web en Google Sites que presenta el proyecto y su información clave.

*Figura 8**Diagrama de fase 2.*

Nota: Descripción de la fase 2.

El documento del plan y el enlace del sitio son obligatorios, y el contenido debe cumplir con todos los requisitos especificados para avanzar a la fase siguiente. El profesor revisará el contenido del plan de proyecto y dará retroalimentación. Solo después de que se hayan cumplido todos los criterios, el profesor desbloqueará la fase 3

Fase 3: Entrega Final del Proyecto:

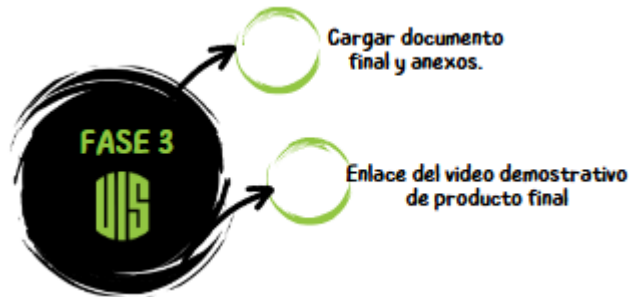
En esta fase, los estudiantes deben presentar el resultado final del proyecto, que incluye:

- **Documento del Libro Final del Proyecto:** Un PDF detallado que contenga la descripción del proyecto, resultados, anexos, análisis, conclusiones y recomendaciones, cumpliendo con las normativas de la biblioteca.
- **Enlace del Video Demostrativo:** Un video que muestre el funcionamiento del proyecto, claro y profesional.

El profesor revisará los documentos y, si se cumplen todos los requisitos, proporcionará un visto bueno y será esencial para la aprobación del trabajo de grado y la obtención del título en ingeniería.

Figura 9

Diagrama de fase 3.

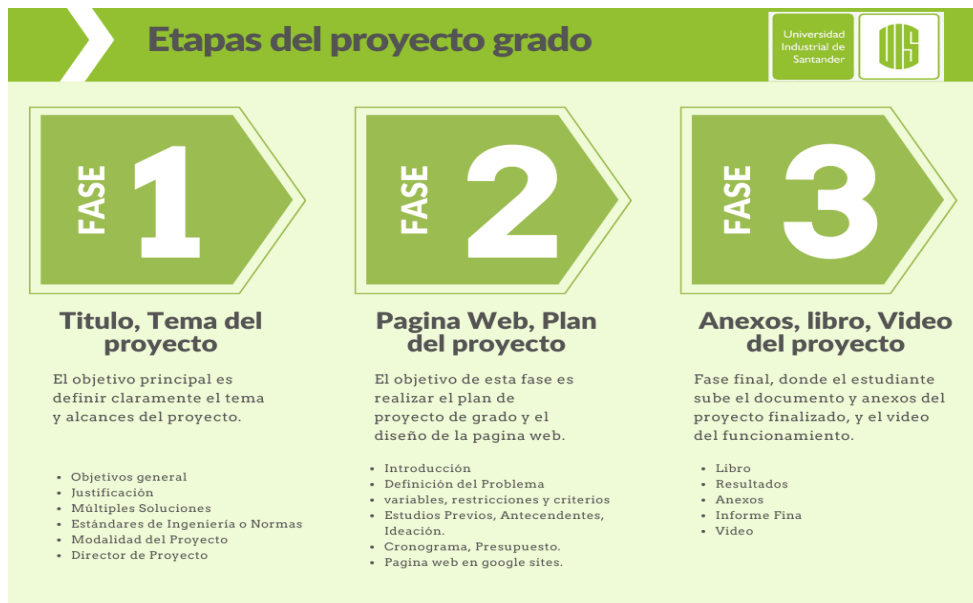


Nota: Descripción de la fase 3.

La finalización de la Fase 3 marca el cierre del desarrollo y gestión del trabajo de grado. La organización del proyecto en tres fases ha sido esencial para asegurar un progreso ordenado, desde la definición del tema hasta la entrega final. Cada fase permite a los estudiantes gestionar eficientemente sus recursos y tiempos, cumpliendo con los requisitos académicos de la Escuela de Ingeniería Eléctrica y Electrónica.

Figura 10

Etapas del proyecto de grado.



Nota: Descripción de todas las fases del proyecto de grado.

4.1.4.3 Evaluación.

En esta sección, el docente o administrador podrá evaluar los archivos del estudiante según criterios establecidos. Esta evaluación está dividida en tres fases, comenzando con la fase "Título y Tema del Proyecto", que revisa el título, objetivos y otros aspectos clave. El docente enviará observaciones detalladas, permitiendo que el estudiante realice correcciones. Al completar esta evaluación, se habilitará la fase 2, donde el estudiante podrá subir más documentación para su evaluación, cuyo resultado será presentado ante el comité de trabajo de grado.

Figura 11

Evaluación Fase 1.

Contiene una entidad interesada en dicho proyecto?.

Contiene

Contiene palabras clave asociadas al proyecto.

Contiene

Sabe distinguir correctamente el área del Proyecto a trabajar.

Excelente

Contiene Director del Proyecto

Contiene

Escriba Observación calificativa sobre el proyecto.

Formulario Fase 1

Objetivo General.

Excelente

Justificación.

Excelente

Identificación de restricciones.

Excelente

Estándares de ingeniería o normas.

Contiene

¿El trabajo posee múltiples soluciones?

Contiene

El trabajo tiene impactos que solucionen una necesidad o problema?.

Contiene

Aplica asignaturas y habilidades adquiridas en el programa que aportan al proyecto.

Contiene

La modalidad del proyecto está bien asignada al tema a tratar?.

Excelente

Guardar **Cancelar**

Nota: Preguntas y requisitos para aprobar la fase 1.

En la fase 2, denominada "Plan y Página Web", el docente tendrá la responsabilidad de realizar una evaluación basada en parámetros y requisitos mínimos establecidos, que son fundamentales para la aprobación del documento. El cumplimiento de estos criterios permitirá que el proyecto avance hacia su presentación ante el comité de trabajo de grado.

Figura 12

Evaluación Fase 2.

Formulario Fase 2

La entrada de la introducción contiene información que sorprende al lector por la calidad de la información? ¿eleva la credibilidad en el escritor por su originalidad? logra atrapar la atención del lector al punto de sentirse motivado para seguir leyendo?

Excelente

El cuerpo de la introducción aborda la problemática global de la que nace el problema concreto a resolver en el proyecto? ¿brinda un contexto para preparar a un lector, que, aunque es ingeniero, no es experto en el tema puntual, para comprender la propuesta y valorarla? ¿para un inversor interesado en el tema esto podría ser suficiente para querer conocer más, o suspendería la lectura?

Excelente

La introducción es tan completa como para lograr el objetivo de que el lector pueda comprender todo el trabajo con solo leer la introducción?

Excelente

El cierre de la introducción logra el objetivo de sorprender al lector al conocer que el trabajo tiene impactos que van más allá de lo obvio? ¿si el lector fuese un inversor, el cierre lograría el efecto de hacerlo pensar en invertir en el proyecto?

Excelente

El capítulo de Definición del problema existe y comienza por presentar el problema específico que el proyecto debe resolver? ¿es específica para los objetivos? ¿es concreta, sin generar ambigüedad?

Excelente

¿El capítulo de Definición del problema contiene listas de las variables, restricciones y criterios, los explica?

Excelente

El capítulo de Definición del problema en la lista de variables, restricciones y criterios hay un esfuerzo por considerar: ¿estándares o normas o políticas? Y los aspectos no técnicos como lo: ¿ambiental, social, de salud, seguridad, economía)?

Excelente

¿Siguiendo los objetivos planteados, el problema será resuelto?

Excelente

Los estudios previos incluyen resultados de la exploración empática que realmente aportan a una solución del problema en beneficio particular de los interesados? ¿queda claro quiénes son los actores o interesados en la solución? ¿hay coherencia entre las manifestaciones de esos actores y los compromisos que este proyecto asume?

Excelente

Los estudios previos presentan un análisis de las causas del problema mayor e identifican la causa o causas que ataca este proyecto?

Excelente

Los estudios previos analizan antecedentes que sirven como punto de partida para el proyecto? ¿se analizan experiencias previas locales o internacionales?

Excelente

Los estudios previos incluyen la ideación? ¿en la ideación divergente se han considerado las ideas más audaces? ¿la ideación convergente demuestra que el proyecto se enfoca en la mejor solución, en idea que más conviene bien sea por el interés de los actores o de otras causas como económicas, de protección al medio ambiente, Etc?

Excelente

Escriba Observación calificativa sobre el proyectos

Guardar
Cancelar

Nota: Preguntas y requisitos para aprobar la fase 2.

De igual manera, el docente realizará las observaciones pertinentes tras una revisión rigurosa de los documentos, las cuales serán enviadas al perfil del estudiante, donde éste podrá visualizarlas y realizar las mejoras correspondientes.

Una vez el docente haya evaluado de manera efectiva y verificado que se cumplen los requisitos solicitados, podrá aprobar la documentación y habilitar la tercera fase del proceso. Esta

fase estará destinada a la carga del libro o informe final, el video final del proyecto, y un apartado para anexos, en caso de que el proyecto lo requiera.

4.1.4.4 Reporte o retroalimentación.

El docente puede generar reportes y ofrecer retroalimentación continua en cada fase del proyecto, ayudando a los estudiantes a mejorar y ajustar su trabajo de manera oportuna.

- **Retroalimentación Personalizada:** El docente añade comentarios específicos sobre el progreso del estudiante, resaltando logros y sugiriendo mejoras antes de pasar a la siguiente fase.
- **Estadísticas del Progreso:** El sistema ofrece estadísticas detalladas sobre el rendimiento de los estudiantes por fase, semestre y programa académico, facilitando el seguimiento del avance del proyecto.

Esta retroalimentación continua asegura que los estudiantes puedan mejorar la calidad de su trabajo y permite al docente tener una visión clara de los avances antes de la evaluación final.

4.1.4.5 Resultados.

El resultado final es la culminación del proyecto de grado por parte del estudiante. A través de la plataforma desarrollada, se gestiona y almacena todo el proceso, desde la definición del proyecto hasta la entrega final. La plataforma facilita el seguimiento del progreso en cada fase y garantiza que todos los documentos y entregables se encuentren organizados y accesibles para su revisión.

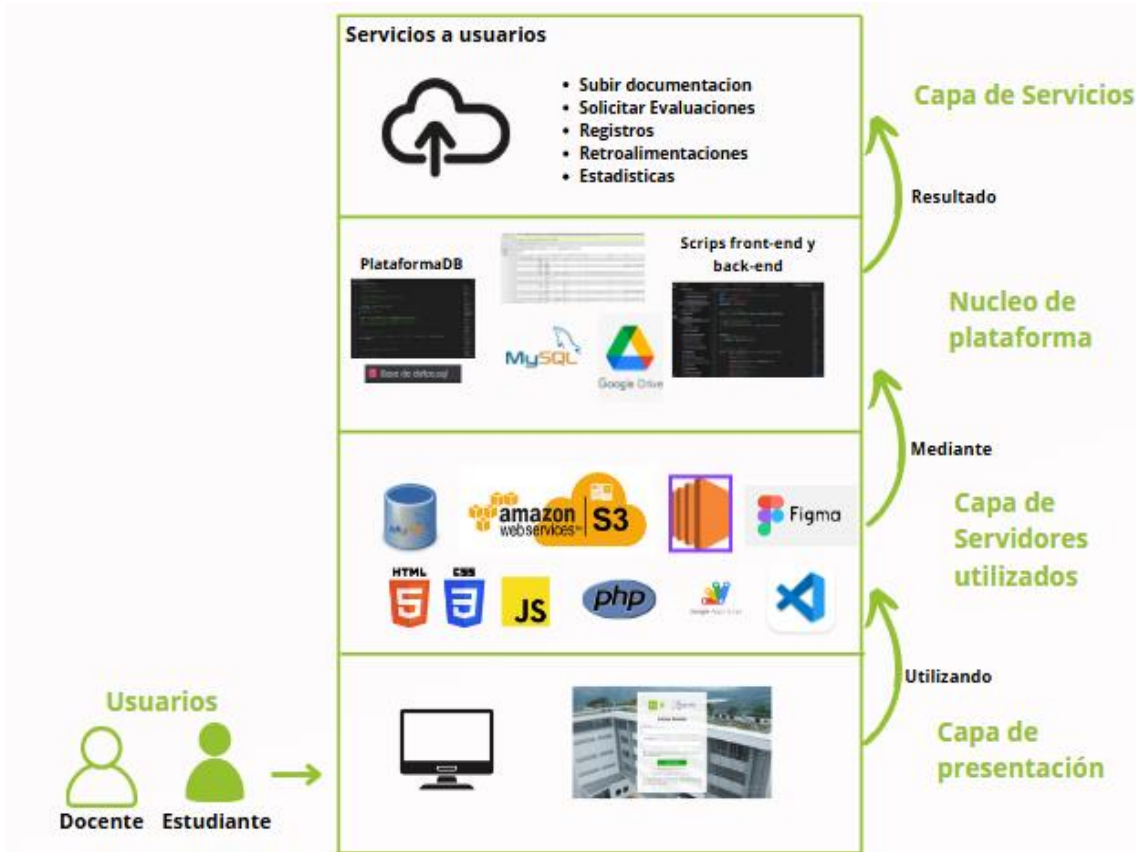
El sistema no solo permite que los estudiantes completen sus proyectos de manera eficiente, sino que también centraliza y resguarda toda la información relacionada, asegurando que los trabajos de grado de la E3T queden debidamente almacenados y disponibles para futuras consultas o auditorías.

3.2 Implementación.

3.2.1 Arquitectura del sistema.

La arquitectura del sistema está diseñada para gestionar eficientemente los proyectos de grado. La Capa de Presentación ofrece una interfaz interactiva para estudiantes y docentes. La Capa de Aplicación gestiona la lógica del sistema y las conexiones necesarias. La Capa de Almacenamiento se centra en una base de datos MySQL, donde se almacena toda la información de los estudiantes y proyectos, asegurando un acceso organizado a los datos. Además, la Capa de Servicios en la Nube proporciona funcionalidades como la subida de documentos, el alojamiento de la plataforma y el envío de correos de recuperación de contraseña.

Figura 13
Arquitectura del prototipo.



Nota: Arquitectura de la plataforma. Fuente: Autores.

3.2.2 Modelo de Capas del Sistema.

Capa de Presentación: Proporciona la interfaz interactiva donde estudiantes y docentes pueden interactuar con la plataforma, permite acciones como registrar temas, subir documentos y consultar proyectos.

Capa de servicios utilizados: Todos los servidores utilizados para poder la ejecución de nuestro diseño, desde la parte de diseño, ejecución, programación y despliegue, es importante reconocer que el 100% de estos servidores se usaron en planes gratuitos para garantizar la ejecución al diseño y poder trabajar en pro a las retroalimentaciones generadas.

Capa de núcleo de plataforma: Se habla de la ejecución de la plataforma, donde se trabaja mediante códigos en los diferentes interfaces Frontend y Backend, así mismo como la creación de las bases de datos y asignaciones de espacios de almacenamiento para los diferentes documentos y/o enlaces generados. así mismo, el trabajo de los diferentes Script que se encargan de los enlaces entre pantallas y el funcionamiento óptimo de los servicios creados.

Capa de Servicio: Se habla de los servicios ofrecidos por la plataforma para los diferentes usuarios, esto siguiendo la premisa de que la plataforma debe ser intuitiva y eficaz al momento de realizar los procesos.

- Subida de documentos de todas las fases a Google Drive.
- Registro de estudiantes nuevos.
- Entrega de documentos por cada fase.
- Envío de correos de recuperación de contraseña.
- Evaluación de las diferentes fases de trabajo.
- Retroalimentación por parte del evaluador.
- Gestión del avance visual de su trabajo de grado.
- Gestión de toda la base de datos por parte del docente.
- Gestión de las estadísticas relacionadas a la comunidad estudiantil matriculada.
- Consulta la cartera de proyectos para la elección del trabajo de grado a realizar.

3.2.3 Elección de figma para el diseño frontend.

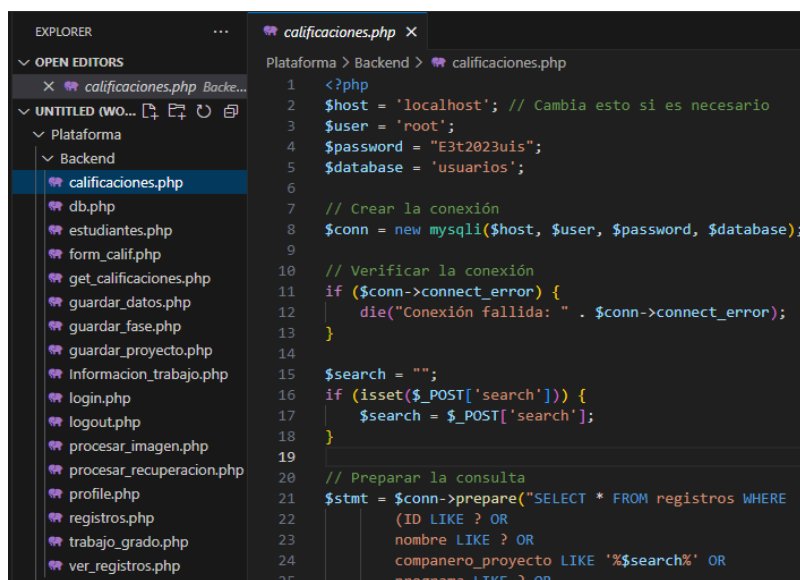
En la fase inicial, revisamos el proceso actual de gestión de trabajos de grado en la E3T para identificar posibles mejoras. Evaluamos diversas herramientas de diseño frontend y seleccionamos Figma por ser gratuito, permitir colaboración en tiempo real, ser accesible desde cualquier plataforma y ofrecer avanzadas herramientas de prototipado. Estas características lo hacen ideal para nuestro proyecto. Para más detalles, incluyendo pantallas y prototipos iniciales, consulte el **Apéndice D**.

3.2.4 Diseño del backend.

El desarrollo del backend resulta fundamental en cualquier proyecto, ya que es la parte encargada de garantizar un rendimiento óptimo del sistema. Este componente se encarga de interactuar con la base de datos, gestionar las sesiones de usuarios y atender las solicitudes provenientes del frontend, entre otras funciones clave. Aunque estas operaciones no son visibles para el usuario final, son esenciales para proporcionar los datos necesarios que permiten organizar las vistas de manera eficiente, lo que contribuye directamente a una experiencia de usuario satisfactoria y fluida.

Figura 14

Desarrollo backend.



```
1 <?php
2 $host = 'localhost'; // Cambia esto si es necesario
3 $user = 'root';
4 $password = "E3t2023uis";
5 $database = 'usuarios';
6
7 // Crear la conexión
8 $conn = new mysqli($host, $user, $password, $database);
9
10 // Verificar la conexión
11 if ($conn->connect_error) {
12     die("Conexión fallida: " . $conn->connect_error);
13 }
14
15 $search = "";
16 if (isset($_POST['search'])) {
17     $search = $_POST['search'];
18 }
19
20 // Preparar la consulta
21 $stmt = $conn->prepare("SELECT * FROM registros WHERE
22     (ID LIKE ? OR
23     nombre LIKE ? OR
24     companero_proyecto LIKE '%$search%' OR
25     programa LIKE ? OR
```

Nota: Desarrollo del backend en PHP.

login.php, se encarga de validar las credenciales de un usuario cuando intenta iniciar sesión en la plataforma web. Verifica si el nombre de usuario (email) y la contraseña introducidos están registrados en la base de datos y, si coinciden, redirige al usuario según su rol: administradores a una página especial y usuarios regulares a su perfil. Si las credenciales no son correctas, muestra un mensaje de error y devuelve al usuario a la página de inicio de sesión.

registros.php, Este código PHP gestiona el proceso de registro de un nuevo usuario en una base de datos MySQL. Primero, establece una conexión con la base de datos y recibe la información del usuario desde un formulario HTML (nombre, código de estudiante, carrera, celular, correo, contraseña, semestre y matrícula). Luego, verifica si el usuario ya está registrado, revisando si el código de estudiante o el correo ya existen en la base de datos. Si ya está registrado, redirige a una página de error; si no, inserta los datos del nuevo usuario en la base de datos y lo redirige a la página de inicio.

Profile.php, gestiona el perfil del usuario en la plataforma web, comenzando por verificar si el usuario ha iniciado sesión y redirigiendo a la página de inicio si no es así. Recupera los datos del usuario de la base de datos mediante una consulta SQL segura y presenta esta información en una interfaz HTML bien estructurada, que incluye un menú lateral y una tabla con datos relevantes. Además, permite al usuario cambiar su foto de perfil utilizando la biblioteca Cropper.js para seleccionar y recortar la imagen, que luego se muestra en su perfil, asegurando así que solo los usuarios autenticados tengan acceso a sus datos personales.

logout.php, se encarga de cerrar la sesión del usuario en la plataforma web. Al iniciar, inicia la sesión actual y luego utiliza la función `session_destroy()`, para eliminar todos los datos de la sesión. Finalmente, redirige al usuario a la página de inicio (`index.html`), asegurando que, tras el cierre de sesión, el usuario no tenga acceso a las funcionalidades protegidas de la aplicación.

trabajo_grado.php, este código PHP y JavaScript implementa un sistema de gestión de trabajos de grado, permitiendo a los usuarios registrar y gestionar su proyecto a través de tres fases. Los usuarios pueden rellenar formularios con información del proyecto, subir documentos clave como planes y libros finales a Google Drive, y recibir retroalimentación del docente. La interfaz presenta un menú de navegación por fases, donde cada fase se habilita según el progreso. Los datos

se envían y almacenan en una base de datos mediante peticiones asíncronas, y el sistema notifica al usuario sobre el éxito o fallo en el guardado de los datos y subida de archivos.

Ver_registros.php, se conecta a una base de datos MySQL para mostrar registros de usuarios y proyectos, excluyendo al usuario administrador. La aplicación cuenta con una barra de búsqueda que filtra los resultados según diferentes criterios (nombre, programa, proyecto, etc.). La interfaz HTML presenta los resultados en una tabla, con columnas para datos como nombres de los estudiantes, título del proyecto y enlaces a recursos relacionados (plan, libro, anexos, etc.). Los resultados se muestran de manera interactiva con un diseño que incluye una barra lateral para navegación y una barra de búsqueda.

informacion_trabajo.php, permite a un usuario autenticado ver y editar los detalles de su proyecto (descripción, justificación, objetivos y modalidad) almacenados en una base de datos MySQL. Utiliza sesiones para identificar al usuario y mostrar un formulario relleno con los datos del proyecto, permitiendo su modificación. El envío del formulario se maneja mediante AJAX, evitando la recarga de la página, y muestra una notificación emergente de éxito.

procesar_recuperacion.php, permite a los usuarios restablecer su contraseña mediante un formulario que solicita su correo electrónico. Cuando se envía el correo, se genera una nueva contraseña aleatoria, se encripta utilizando `password_hash`, y se actualiza en la base de datos. Luego, se envía un correo electrónico al usuario con la nueva contraseña. El código incluye una función para generar contraseñas aleatorias y otra para enviar el correo con las cabeceras adecuadas. Si todo se ejecuta correctamente, se notifica al usuario que la nueva contraseña ha sido enviada a su correo, y en caso de error, se muestran mensajes de error específicos.

procesar_imagen.php, permite a un usuario autenticado cambiar su imagen de perfil y visualizar su información personal desde una base de datos. Al iniciar sesión, el usuario puede seleccionar una nueva imagen a través de un formulario que la guarda en la base de datos como un archivo binario. Los datos del usuario, como su ID, nombre, programa y líder de proyecto, se recuperan de la base de datos y se muestran en una tabla. Además, el código incluye funcionalidad JavaScript para previsualizar la imagen seleccionada antes de enviarla y un sistema básico de manejo de sesiones.

guardar_proyecto.php, permite a un usuario autenticado actualizar la información de un proyecto en la base de datos, incluyendo descripción, justificación, objetivos y modalidad. Después de iniciar la sesión y establecer una conexión con la base de datos, se recupera el nombre de usuario de la sesión y se verifican los datos del formulario. Luego, se obtiene el ID del usuario para construir dinámicamente una consulta de actualización que modifica solo los campos que el usuario ha enviado. Si la actualización es exitosa, se redirige al usuario a una página de confirmación; de lo contrario, se muestra un mensaje de error.

guardar_fase.php, está diseñado para actualizar registros en una base de datos a través de una solicitud AJAX. Primero, establece una conexión con la base de datos y verifica si la conexión fue exitosa. Luego, obtiene varios datos enviados desde un formulario, incluyendo el ID del registro a actualizar, la fase del proyecto y hasta doce ítems, así como observaciones. Si el ID es válido, se prepara y ejecuta una consulta SQL de actualización para modificar el registro correspondiente en la tabla "registros". Si la ejecución es exitosa, se devuelve un mensaje de confirmación; si ocurre un error, se informa del problema. Finalmente, se cierra la conexión a la base de datos.

guardar_datos.php, este script PHP actualiza los datos de un usuario en una base de datos a través de solicitudes POST, asegurando que la sesión esté activa. Incluye tres formularios para actualizar el título y tema del proyecto, enlaces al plan y página, y enlaces a un libro, video y anexos. Se limpian los datos ingresados, se preparan consultas SQL de actualización y se manejan errores con mensajes de respuesta en formato JSON. La función limpiarDatos se utiliza para sanitizar la entrada de datos. Al final, se cierra la conexión a la base de datos.

get_calificaciones.php, Este script PHP recupera el estado de un ítem de la base de datos utilizando el ID del ítem enviado a través de una solicitud POST. Primero, realiza una consulta para obtener el estado del ítem correspondiente y luego devuelve esta información como una respuesta en formato JSON.

form_calif.php, Este script inicia una conexión a la base de datos y verifica su éxito. Luego, recoge datos enviados a través de un formulario, incluyendo el ID del registro y la fase, y se asegura de que todos los datos necesarios estén presentes. Según el valor de la fase, se actualizan diferentes campos en la tabla de registros, utilizando consultas preparadas para evitar inyecciones SQL. Si la actualización es exitosa, se muestra un mensaje de éxito y se redirige al usuario a calificaciones.php.

Además, incluye manejo de errores para asegurar que se informen adecuadamente los problemas que puedan surgir durante el proceso.

estudiantes.php, el código permite a los usuarios seleccionar filtros (semestres, matrículas, programas y fases) mediante casillas de verificación para generar estadísticas sobre estudiantes desde una base de datos llamada usuarios. Utiliza PHP para conectarse a la base de datos y ejecutar consultas SQL que recuperan valores únicos para cada filtro. Al enviar el formulario, se cuentan los estudiantes que cumplen con los filtros seleccionados y se representan visualmente mediante un gráfico de pastel utilizando Chart.js. La interfaz está diseñada con CSS para ser atractiva y fácil de usar, incluyendo una barra lateral de navegación y un diseño responsivo

db.php, El código en PHP establece una conexión a una base de datos MySQL utilizando la clase mysqli, definiendo las credenciales necesarias como el host, usuario, contraseña y nombre de la base de datos. Si la conexión falla, se termina la ejecución del script y se muestra un mensaje de error. Este código es el primer paso para interactuar con la base de datos, permitiendo posteriormente realizar consultas y otras operaciones.

calificaciones.php, el código es una página web para gestionar evaluaciones de proyectos de estudiantes, que incluye una barra lateral de navegación, un formulario de búsqueda y una tabla que muestra información sobre los estudiantes y sus proyectos. Cada entrada de la tabla tiene botones que abren un formulario emergente para ingresar calificaciones en dos fases del proyecto. El formulario permite evaluar varios criterios mediante menús desplegables y enviar los datos utilizando AJAX. También incluye funciones JavaScript para mostrar y ocultar el formulario emergente, así como manejar la lógica de envío de datos al servidor.

El diseño del backend en PHP ha permitido crear un sistema eficiente para la gestión de calificaciones de proyectos de grado en la E3T. Cada componente cumple una función clave, desde la interfaz de usuario hasta la lógica de negocio, lo que garantiza un procesamiento y presentación adecuados de los datos. Esto asegura una experiencia de usuario fluida y una administración efectiva de la información.

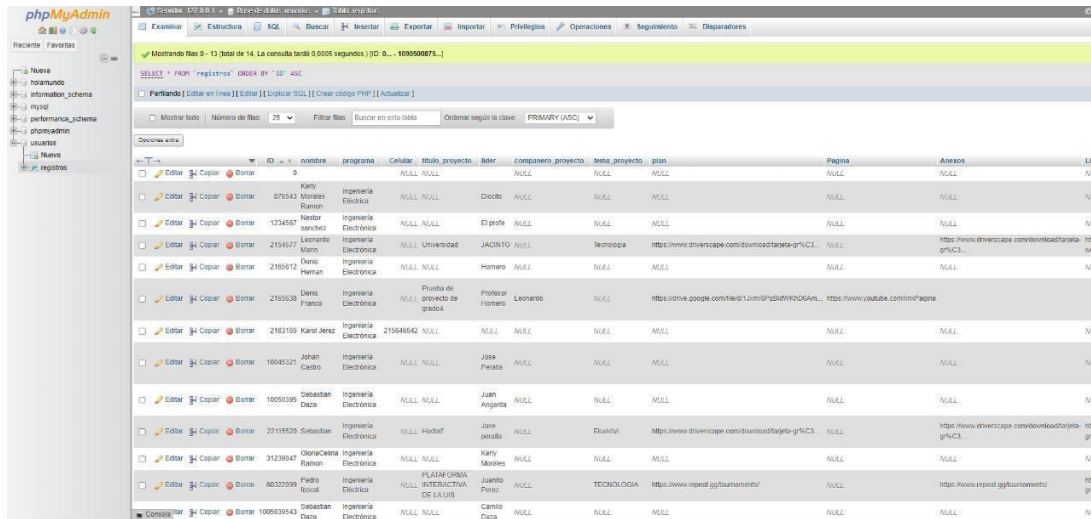
3.2.5 Elección de una base de datos relacional (MySQL).

En este proyecto, se decidió utilizar una base de datos relacional (MySQL) debido a varios factores clave:

- **Estructura de los datos:** Los datos de este sistema tienen una estructura clara y bien definida, donde las entidades están relacionadas entre sí. Por ejemplo, en la base de datos de "usuarios", cada registro tiene información sobre nombres, programas, descripciones, observaciones, etc. Este tipo de datos es ideal para manejarse en un esquema relacional.
- **Relaciones claras:** Las tablas en la base de datos pueden estar relacionadas entre sí. Por ejemplo, una tabla de registros puede estar vinculada a otras tablas como usuarios, donde los datos de un usuario pueden relacionarse con múltiples proyectos o fases.
- **Garantía de Consistencia:** La base de datos necesita garantizar que las transacciones sean confiables y consistentes, algo que MySQL hace muy bien gracias a las propiedades ACID. Esto es especialmente útil para sistemas que requieren integridad y precisión, como los sistemas académicos.
- **Lenguaje SQL:** El uso de SQL permite realizar consultas complejas y optimizar la manipulación de datos mediante índices y otras características avanzadas, lo cual es beneficioso cuando el sistema crece en tamaño y complejidad.
- **Soporte y Comunidad:** MySQL es ampliamente utilizado, tiene gran documentación y una comunidad activa que lo respalda, lo que facilita el desarrollo y mantenimiento del sistema.

Otro factor importante es el soporte y la facilidad de administración que ofrece phpMyAdmin, un visor gráfico que permite interactuar con la base de datos de manera visual e intuitiva. phpMyAdmin nos permite gestionar, consultar y modificar tablas, realizar backups, y monitorizar el rendimiento de la base de datos sin necesidad de comandos complicados. Esta herramienta es fundamental para nosotros, ya que agiliza las tareas administrativas y nos brinda una forma fácil de trabajar con la base de datos. MySQL, en combinación con phpMyAdmin, ofrece un entorno robusto, seguro y fácil de gestionar, ideal para nuestro proyecto de base de datos. Para más detalles, consultar el **Apéndice E**.

Figura 15
Visor phpMyAdmin.



Nota: Base de datos desde el visor phpMyAdmin.

3.2.6 Web Service para Subida de Archivos a Google Drive.

El sistema cuenta con un Web Service dedicado para la gestión de archivos PDF y otros documentos relevantes, como los proyectos de grado. Este servicio permite subir archivos a Google Drive y obtener una URL pública, que se almacena automáticamente en la base de datos.

- **Descripción del Web Service**

El Web Service se encarga de subir los archivos a una carpeta específica en Google Drive, gestionando los permisos de acceso y devolviendo la URL pública para que los usuarios puedan acceder al documento desde la plataforma. Esta URL se almacena automáticamente en la base de datos de la plataforma en el campo correspondiente.

Funcionalidades Clave:

1. Subida de Archivos: El servicio recibe una solicitud POST que contiene el archivo en formato Base64, su nombre y tipo MIME, así como el ID de la carpeta de destino en Google Drive.

2. **Generación de URL:** Tras la subida exitosa del archivo, el Web Service devuelve una URL pública para acceder al documento. Esta URL se va directo a la base de datos de la plataforma para ser vinculada con el proyecto correspondiente.

Flujo de Trabajo del Web Service:

- **Subir un archivo:** El usuario selecciona un archivo PDF a través de la interfaz web.
- **Generación de la URL:** El Web Service devuelve la URL pública del archivo, que se almacena automáticamente en la base de datos, Para más información revisar el **Anexo F**.

3.2.7 Implementación AWS.

La implementación de la página web se realizó en un entorno de servidor utilizando Amazon Web Services (AWS), específicamente a través de una instancia EC2 (Elastic Compute Cloud). Inicialmente, la aplicación fue desarrollada y probada en un entorno local con XAMPP, lo que facilitó el desarrollo y la depuración inicial del código.

1. Creación de la instancia EC2: Se creó una instancia EC2 de AWS, seleccionando un sistema operativo Ubuntu o Amazon Linux y configurando las especificaciones de la máquina según las necesidades del proyecto (CPU, RAM y almacenamiento). Durante la configuración, se aseguraron de que las políticas de seguridad permitieran el acceso a los puertos necesarios, especialmente el puerto 80 para HTTP, puerto 22 para SSH y puerto 3306 para MySQL, para este caso se utiliza una IP pública proporcionada por EC2 de AWS en 3.128.45.96 para hacer el lanzamiento de la página. Para más información en **Apéndice G**.

2. Instalación del servidor web: Una vez configurada la instancia, se instaló un servidor web (Apache o Nginx) para alojar la aplicación. Se realizaron configuraciones adicionales para optimizar el rendimiento del servidor, incluyendo la activación de módulos necesarios y la creación de archivos de configuración específicos para la aplicación.

3. Transferencia de archivos: Los archivos de la aplicación, que fueron desarrollados y probados en el entorno local, se transfirieron a la instancia EC2 utilizando el protocolo SFTP (Secure File Transfer Protocol) en FileZilla. Se organizaron en el directorio raíz del servidor web para que fueran accesibles a través de la dirección IP de la instancia.

4. Configuración de la base de datos: La configuración de la base de datos se realizó mediante una instalación del motor de base de datos, para posteriormente hacer la integración creando un usuario y contraseña para poder trabajar con la misma base de datos que veníamos trabajando en pruebas locales, adicionalmente utilizando DBeaver, una herramienta de gestión de bases de datos que facilitó administración de la base de datos MySQL. Se visualizaron las tablas necesarias, se definieron las relaciones y se importaron los datos previamente almacenados en la base de datos local. DBeaver permitió una visualización clara de la estructura de la base de datos y facilitó la ejecución de consultas y la gestión de datos de manera eficiente.

5. Ajustes de seguridad: Se implementaron medidas de seguridad adicionales, como la configuración de grupos de seguridad en AWS, el uso de claves SSH para acceder a la instancia y la instalación de un firewall para proteger el servidor contra accesos no autorizados.

6. Pruebas finales: Finalmente, se realizaron pruebas para garantizar que la plataforma funcionará correctamente en el entorno de página web, Esto incluyó pruebas de funcionalidad, rendimiento y seguridad, asegurando que todos los componentes de la aplicación interactúan sin problemas. Tanto desde el perfil del estudiante como desde el Administrador. Estas pruebas se hicieron simulando una cantidad de estudiantes matriculados en la asignatura, para verificar el envío correcto de la información y visualización tanto en el perfil administrador o docente, como en el perfil del estudiante. Adicional, el correcto alojamiento de los archivos en Google Drive proporcionado para la disponibilidad de manejo por parte del administrador.

3.3 Experiencia de usuario.

A continuación, se muestra la página para iniciar sesión y registrarse por parte de los usuarios.

Figura 16

Vista inicio de sesión.

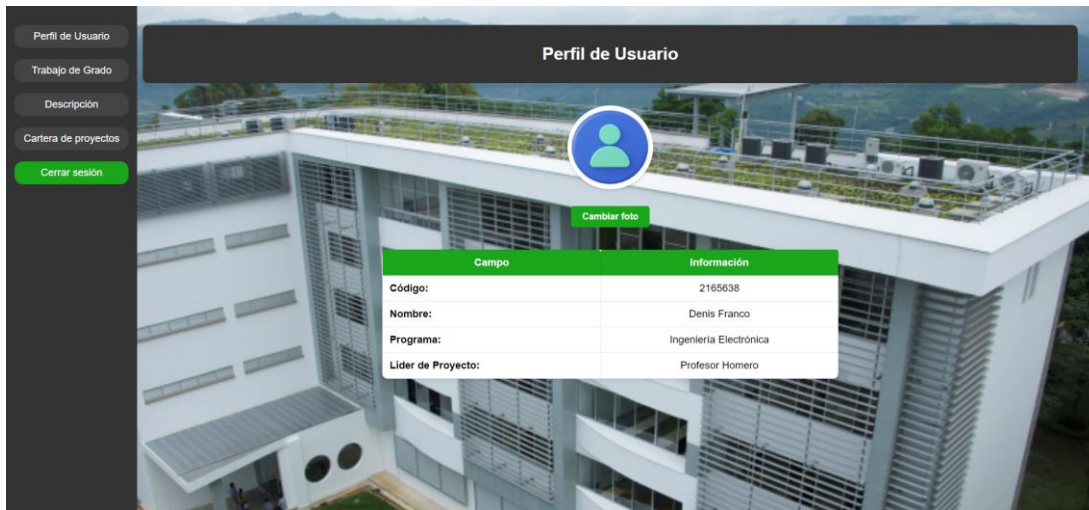


Nota: Diseño del frontend del inicio de sesión.

Esta página permite a los usuarios ingresar sus credenciales para acceder a la aplicación. Es simple y fácil de usar, garantizando un acceso rápido.

Figura 17

Perfil estudiante.



Nota: diseño frontend del perfil estudiante.

Se puede observar la información del estudiante, así como las pestañas en el panel lateral que son fundamentales para la navegación y funcionalidad del proyecto.

Figura 18

Pestaña trabajo de grado.



Nota: Frontend de las fases del proyecto, perfil estudiante.

Donde los estudiantes pueden cargar la información y los archivos de su proyecto de grado.

Figura 19

Pestaña estadística. Por estudiante.



Nota: Perfil docente donde puede generar tipos de gráficas estadísticas.

Esta pestaña permite al docente y al administrador visualizar la cantidad de estudiantes mediante gráficas, facilitando el seguimiento de su progreso en cada fase, semestre y programa académico.

A continuación, se presentó el diseño de la interfaz para estudiantes y docentes, se han seleccionado algunas pantallas específicas para optimizar el uso del espacio en el informe, revisar **Apéndice H**, donde están todas las pantallas.

3.4 Validación y verificación.

3.4.1 Metodología de pruebas.

En la metodología de las pruebas se realiza una simulación del rol estudiante y rol de administrador o docente. Se revisa su correcto envío de información y correcta visualización de documentación suministrada por el estudiante en el perfil del administrador o docente.

3.4.2 Plan de pruebas.

Tabla 1. 1

Plan de pruebas para realizar en la página.

Tipo de Prueba	Objetivo	Descripción
Funcionalidad Registro de Estudiantes	de Verificar que los estudiantes puedan registrarse correctamente.	Ingresar datos válidos (nombre, correo, contraseña, carrera, semestre, nivel, celular) y confirmar que se complete el registro, asegurando que se almacenen en la base de datos.
Funcionalidad Inicio de Sesión	Comprobar que los usuarios puedan iniciar sesión con credenciales correctas.	Probar el inicio de sesión con credenciales válidas e inválidas, verificando el acceso a la plataforma y los

				mensajes de error correspondientes.
--	--	--	--	-------------------------------------

Funcionalidad	Carga de Documentación	de	Evaluar que los estudiantes puedan subir archivos correspondientes a cada fase.	Intentar cargar documentos válidos y no válidos (tipos y tamaños de archivo incorrectos) y verificar que se gestionan los errores adecuadamente.
----------------------	------------------------	----	---	--

Funcionalidad	Calificación de Fases	de	Asegurar que el docente pueda calificar las fases adecuadamente.	Ingresar a la sección de calificaciones y comprobar que se puedan calificar las dos primeras fases, y verificar que se reflejen correctamente en el perfil del estudiante las observaciones generadas.
----------------------	-----------------------	----	--	--

Funcionalidad	Habilitación de Fases	de	Verificar que el acceso a las fases se active tras la calificación de las fases anteriores.	Calificar la primera y segunda fase y comprobar que el acceso a la siguiente fase en el perfil del estudiante se habilite correctamente.
----------------------	-----------------------	----	---	--

Usabilidad	Evaluación de Interfaz	de	Probar la usabilidad de la plataforma para asegurar que sea intuitiva y fácil de navegar.	Realizar una evaluación de la interfaz, comprobando la claridad de los menús, botones y mensajes de error para asegurar una experiencia fluida para los usuarios.
-------------------	------------------------	----	---	---

Integración	Almacenamiento en Base de Datos	Comprobar que los datos de usuarios y calificaciones se almacenen correctamente en la base de datos.	Verificar que, tras el registro, los datos de los estudiantes y las calificaciones se reflejen correctamente en la base de datos utilizando herramientas como DBeaver.
Visualización	Generar estadísticas	Visualizar de manera clara las métricas de los estudiantes realizando trabajos de grado.	Verificar desde el perfil del administrador que se pueda visualizar y filtrar de manera para obtener el gráfico de las estadísticas de estudiantes realizando trabajo de grado.

Nota: Planes de pruebas.

3.4.3 Resultados de las pruebas.

Para la ejecución de las pruebas, se replicó el proceso típico que llevaría a cabo un estudiante al utilizar la plataforma para la gestión de su trabajo de grado. Estas pruebas siguieron el plan detallado previamente.

El proceso comenzó con el registro del estudiante, cuyo objetivo era validar que la información suministrada se almacena correctamente en la base de datos. Una vez verificada la validez del registro, se procedió a realizar el inicio de sesión utilizando las credenciales proporcionadas (correo y contraseña). Después de esta actividad, se exploraron las diferentes pestañas y secciones de la plataforma con el propósito de evaluar su accesibilidad y funcionalidad, asegurando que cada apartado funcionara según lo esperado.

Posteriormente, se procedió a la selección del proyecto a trabajar, lo que permitió seguir el flujo de la plataforma hacia la fase 1. En esta fase, se revisaron los requisitos solicitados para adjuntar la información requerida: el título del proyecto, un documento PDF con el desarrollo del tema, el nombre del compañero de trabajo, y el director del proyecto (ver Figura 14). Se comprobó que el

sistema permitiera la carga de los archivos y la correcta vinculación de estos datos con el perfil del estudiante.

Una vez completada la fase 1, se accedió al perfil del docente, donde se corroboró que la información enviada por el estudiante estaba efectivamente almacenada en la base de datos. Esto permitió al docente proceder con la evaluación, generar observaciones y, si fuera necesario, solicitar correcciones a los documentos subidos. Al guardar la calificación de la fase 1, se confirmó que el sistema mostraba correctamente las observaciones al estudiante y habilitaba la fase 2.

El mismo procedimiento se aplicó a las fases y calificaciones restantes, verificando que el proceso de gestión de los proyectos de grado cumpliera con su objetivo de manera eficiente. Adicionalmente, se validó que tanto la base de datos de información (configurada en DBEaver) como la de archivos (almacenados en Google Drive) recibieron toda la información suministrada desde ambas interfaces de usuario.

Finalmente, se realizó una prueba para verificar las estadísticas de la plataforma, comprobando que el estudiante de prueba aparecía en los datos estadísticos y contribuía al análisis cuantitativo, confirmando así el correcto funcionamiento de esta funcionalidad.

Para ver la información más detallada, con ilustraciones gráficas de las pruebas, revisar **Apéndice I**. Plan de pruebas y ejecución de pruebas del sistema.

4. Conclusiones

El desarrollo de la plataforma transmedia para la gestión de trabajos de grado de E3T ha permitido abordar la problemática inicial planteada, proporcionando una solución integral para el seguimiento y la organización de proyectos Capstone Design. A lo largo de este proyecto, se lograron cumplir los objetivos propuestos de manera efectiva.

En primer lugar, se establecieron claramente las necesidades del sistema mediante un análisis exhaustivo de casos de uso e historias de usuario, lo que permitió crear una plataforma que se adapta a los diferentes roles (estudiantes y docente) y flujos de trabajo requeridos para la correcta gestión de los trabajos de grado. Este enfoque garantizó que la solución fuera relevante y funcional para todos los interesados.

El diseño del prototipo tomó en cuenta las limitaciones y requerimientos del sistema, resultando en una interfaz intuitiva y una estructura lógica que facilita el registro y el seguimiento de las fases de cada proyecto. Además, la implementación del prototipo permitió la integración eficiente de bases de datos y el almacenamiento en Google Drive, asegurando una gestión ordenada y accesible de la información y documentación relacionada.

Las pruebas realizadas, tanto unitarias como integradas, confirmaron la solidez del sistema. El proceso de validación mediante pruebas de usuario aseguró que los estudiantes puedan navegar sin dificultad por las distintas fases del proyecto, desde el registro hasta la entrega final. Asimismo, el docente puede realizar un seguimiento claro y preciso de los avances, evaluar los resultados y proporcionar retroalimentación efectiva a lo largo de todo el proceso.

La plataforma no solo facilita la gestión y el seguimiento de los trabajos de grado a distintos ritmos, sino que también permite una evaluación cuantitativa a través de las estadísticas proporcionadas, lo que mejora la toma de decisiones y el monitoreo del progreso académico. La implementación de una solución tecnológica basada en conceptos de transmedia, tecnologías tradicionales y nuevas tecnologías como Web Service no solo mejora la eficiencia del proceso, sino que también ofrece un sistema flexible y accesible que puede ser adaptado a futuras necesidades de la E3T.

Como último el proyecto ha cumplido su objetivo general de crear una herramienta funcional y adaptable que cubre las carencias tecnológicas previamente identificadas, mejorando así el proceso académico de los estudiantes de E3T y facilitando el trabajo de los docentes involucrados en la gestión de los proyectos de grado.

5. Recomendaciones.

Para la culminación y mejoras de este proyecto, hacemos las siguientes recomendaciones para futuras mejoras de seguimiento al diseño.

1. Es aconsejable separar la información en diferentes tablas dentro de la base de datos en lugar de almacenarla toda en una sola tabla. Actualmente, datos sensibles, como información personal, detalles del proyecto, calificaciones y credenciales de usuario, están concentrados en una única tabla, lo que incrementa el riesgo de exposición en caso de vulnerabilidades de seguridad, revisar el diagrama entidad relación **Apéndice J**. Al dividir la información en tablas específicas según su tipo o nivel de sensibilidad, se puede mejorar tanto la seguridad como el rendimiento del sistema, facilitando un control de acceso más preciso y reduciendo la superficie de ataque. Esta segmentación también permite una mejor escalabilidad, optimizando cada tabla de acuerdo a sus necesidades particulares.
2. Una opción para mejorar la gestión de datos por parte de los docentes es integrar la base de datos con Google Sheets. Esta herramienta ofrece una interfaz sencilla y familiar, lo que permite a los docentes acceder y manipular la información de manera eficiente sin necesidad de tener conocimientos avanzados en bases de datos. Además, Google Sheets permite la colaboración en tiempo real, facilitando la revisión y actualización conjunta de datos, así como funciones avanzadas de filtrado y análisis, útiles para el seguimiento del progreso de los estudiantes.
3. Una recomendación importante para enriquecer la experiencia del usuario es la inclusión del chatbot e3t tg, que puede proporcionar asistencia automatizada a estudiantes y docentes al resolver dudas sobre el uso de la plataforma, la carga de documentos y la

revisión de calificaciones. La implementación de este chatbot disminuiría la carga administrativa para los docentes, permitiéndoles enfocarse en la evaluación académica. Además, al ofrecer soporte 24/7, se brindaría a los estudiantes un recurso accesible para resolver inquietudes en tiempo real, mejorando así la eficiencia y satisfacción en el uso de la plataforma.

4. Se sugiere mejorar la interfaz visual de la plataforma para optimizar la experiencia del usuario. Una interfaz más moderna y atractiva, con una distribución clara y navegación intuitiva, facilita el acceso a las funciones principales para estudiantes y docentes. Un diseño responsivo adaptado a distintos dispositivos y tamaños de pantalla permitiría un uso cómodo desde móviles o tablet. Además, colores contrastantes y tipografía legible mejorarían la usabilidad y accesibilidad, mientras que iconos y gráficos interactivos proporcionarían una experiencia más dinámica y visualmente atractiva, aumentando la eficiencia y satisfacción de los usuarios.
5. Se recomienda crear un perfil específico para cada director de proyectos en la plataforma. Esto permitirá a los directores gestionar de manera más efectiva sus grupos y proyectos, accediendo a herramientas y funciones adaptadas a sus necesidades. Con un perfil dedicado, los directores podrán supervisar el progreso de cada proyecto, dar seguimiento a calificaciones y comunicarse directamente con los estudiantes, mejorando así la seguridad y organización del sistema.
6. Se propone automatizar la creación de páginas web para los proyectos Capstone Design, utilizando la información de Google Sheets y Google Apps Script. Esta mejora reduciría el trabajo manual de los estudiantes, garantiza la coherencia en los formatos de las páginas y aumentaría la eficiencia en la gestión de proyectos. Consultar **Apéndice K**.
7. Se recomienda desarrollar una versión móvil de la plataforma para mejorar la accesibilidad y usabilidad, permitiendo a estudiantes y docentes acceder fácilmente desde dispositivos móviles. Esto proporciona una experiencia más fluida, optimizada para pantallas pequeñas, facilitando tareas como registro, carga de documentos y revisión de calificaciones en cualquier momento. Además, permitiría la recepción de notificaciones en tiempo real, contribuyendo a una gestión más eficiente y flexible, acorde con las tendencias tecnológicas actuales.

Referencias Bibliográficas

- Accreditation Board of Engineering and Technology. (2015, marzo 24). *About Abet*. ABET.
<https://www.abet.org/about-abet/>
- Alves, C., Bach, J. (2021). TypeScript: TypeScript para principiantes. (n.p.): Independently Published.
- Amazon, E. C. (2023). *Computación en la nube con AWS*. Amazon.com.
<https://aws.amazon.com/es/ec2/>
- Arias, Á. (2014). Bases de Datos con MySQL. Estados Unidos: CreateSpace Independent Publishing Platform.
- Arias, M. Á. (2017). Aprende Programación Web con PHP y MySQL: 2a Edición. (n.p.): CreateSpace Independent Publishing Platform.
- Arjonilla, R. (2016). *¿Qué es BackEnd? ¿Qué diferencia al backend del frontend?*
<https://rafarjonilla.com/que-es/backend/>
- Barzanallana, R. (2017). *Introducción a CSS*. Universidad de Murcia. Www.um.es.
<https://www.um.es/docencia/barzana/DAWEB/2017-18/daweb-tema-10-introduccion-css.html>
- Barzanallana., R. (2017). *Desarrollo de Aplicaciones WEB. Introducción a HTML y CSS*. Universidad de Murcia. Wwww.um.es. <https://www.um.es/docencia/barzana/DAWEB/2017-18/daweb-tema-1-introduccion-html-css.html>
- Buttu, M. (2020). *El gran libro de Python*. Marcombo.
- Carbonell, M. M. (2015). *phpMyadmin Prof. Moisés Mañas Carbonell Dpto. Escultura. UPV moimacar@esc.upv.es INTRODUCCION A LA CREACIÓN DE BASES DE DATOS*. Cloudfront.net.
- CEI: Escuela de Diseño y Marketing. (2021, septiembre 27). *¿Qué es figma?* CEI. <https://cei.es/que-es-figma/>

- Cumare, J. (2014). *FRAMEWORK DE DESARROLLO DE SOFTWARE*. Tópicos Generales de Ingeniería de Software. <https://ingsoftwarei2014.wordpress.com/category/framework-de-desarrollo-de-software/>
- Dimes, T., & Parraud, A. (2015). *JavaScript Una Guía de Aprendizaje para el Lenguaje de Programación JavaScript*. Babelcube.
- Erl, T. (2005). *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice Hall
- Flores, F. (2022). *Qué es Visual Studio Code y qué ventajas ofrece*. Openwebinars.net. <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>
- Golden, B. (2013). *Amazon Web Services for Dummies*. John Wiley & Sons.
- Herrero, A. (2020, diciembre 31). *Frontend. ¿Qué es el Front end?* <https://neoattack.com/neowiki/front-end/>
- Jenkins, H. (2016). *La narrativa transmedia en el contexto de la convergencia mediática*. Revistavirtualis.mx. <https://www.revistavirtualis.mx/index.php/virtualis/article/view/381/452>
- Norman, N. (2016, junio 1). *What is User Experience (UX) Design*. The Interaction Design Foundation; Interaction Design Foundation. <https://www.interaction-design.org/literature/topics/ux-design>
- Smith, A. J. (2021). *Mysql: La Guía Completa de la Base de Datos Más Utilizada en el Desarrollo Del Lado Del Servidor. Contiene Query SQL de Muestra y Muchos Consejos de Arquitectura para Principiantes*. (n.p.): Independently Published.
- The Interaction Design Foundation. (2016, junio 2). *User Interface (UI) Design*. <https://www.interaction-design.org/literature/topics/ui-design>
- Tipología de proyectos*. (2023). Minciencias. https://minciencias.gov.co/viceministerios/conocimiento/direccion_transferencia/beneficios-tributarios/tipologia-proyectos