



**Sistema de seguimiento a graduados multifacultad para la Universidad Santo
Tomás Seccional Tunja**

PROPONENTE(S)

Richard Eduardo Molano Cubillos
1049603755
2344333

DIRECTOR

Diego Alejandro Vela Beltrán

Tunja

Fecha de presentación (11, 5, 2026)

CONTENIDO

1. FICHA TÉCNICA DEL PROYECTO	4
2. PLANTEAMIENTO DEL PROBLEMA	5
3. JUSTIFICACIÓN	6
4. OBJETIVOS	7
5. MODELO DE DESARROLLO	8
6. ENTREGABLES	12
7. DESARROLLO DEL PROYECTO	14
7.1. FASE 1. ANÁLISIS DEL SISTEMA ACTUAL	14
7.1.1. DEFINICIÓN DE REQUISITOS	14
7.1.2. DIAGRAMAS DE CASOS DE USO	17
7.1.3. FASE 1.2 SELECCIÓN DE TECNOLOGÍA	29
7.1.4. FASE 2. DISEÑO DEL SISTEMA	31
7.2. FASE 2.2 DESARROLLO DE LA APLICACIÓN	32
7.2.1. FASE 2.3. CICLO DE VIDA – NESTJS	35
7.2.2. FASE 3 ESTRUCTURA DEL PROYECTO	35
7.2.3. FASE 3.3 DOCUMENTACIÓN	43
7.3. FASE 4 PUESTA EN PRODUCCIÓN	52
7.3.1. FASE 4.1 DOKER	52
7.3.2. FASE 4.2. PRUEBAS	55
8. CONCLUSIONES Y ANÁLISIS DE RESULTADOS	60
9. REFERENCIAS	61
10. ANEXOS	63
ANEXO 1. FICHA TÉCNICA	63
ANEXO 2. MANUAL DE USUARIO	64
1. Acceso a la página	64
2.1 Pantalla principal	64
2.2 Inicio de sesión	66
3. Panel administrativo	70
3.1 Cambio de tema (modo claro / oscuro)	70
3.2 Información del usuario y filtrado por facultad	71
4. Gestión del carrusel	72
4.1 Editar URL de la imagen	74
4.2 Editar hipervínculo	74
5. Gestión de eventos	76
5.1 Administrar eventos	76
5.2 Editar un evento	77
5.3 Crear un evento	79
6. Gestión de noticias	81

Al acceder al menú lateral, expande la sección de Noticias.	81
6.1 Administrar noticias	81
6.2 Editar una noticia	82
6.3 Crear una noticia	84
7. Gestión de graduados	87
7.1 Crear un graduado	87
7.2 Administrar graduados	89
Elija la opción «Administrar» para ver los graduados creados.	89
7.3 Importar / Exportar graduados (CSV)	90
7.4 Editar un graduado	91
7.4.1 Empleos y posgrados	93
8. Gestión de docentes	96
8.1 Administrar docentes	96
8.2 Eliminar un docente o decano	98
8.3 Papelera y restauración	99
9. Gestión de facultades	103
9.1 Administrar facultades	103
9.2 Editar una facultad	105
9.3 Crear una facultad	107
9.4 Eliminar una facultad	108
10. Reportes y estadísticas	109
Anexo 3. Evidencias Reunión Actores clave	112
Anexo 4. GitHub	112

1. FICHA TÉCNICA DEL PROYECTO

Título	Sistema de seguimiento a graduados multifacultad para la Universidad Santo Tomás Seccional Tunja
Autor (es)	Richard Eduardo Molano Cubillos
Director del proyecto	Diego Alejandro Vela Beltrán
Palabras claves	Graduados Escalabilidad Facultad
Descripción	
Refactorización de código con el fin de Escalar el programa para la gestión de graduados de la Facultad de Ingeniería de Sistemas orientado a un entorno multi-facultad en la Universidad Santo Tomás Seccional Tunja, mediante la estructuración de modelo de datos y adopción de arquitectura modular, para mejorar la trazabilidad y la mantenibilidad del software manteniendo los frameworks usados originalmente de Angular, NestJs y generando una dockerizacion para facilitar su despliegue en máquinas junto con una refactorización de las interfaces. Se propuso una arquitectura de base de datos equivalente, junto con la implementación de las interfaces y la lógica que permitieran el escalado deseado.	
Título	Sistema Para Seguimiento A Graduados Universidad Santo Tomás Seccional Tunja
Autor (es)	Richard Eduardo Molano Cubillos
Palabras claves	Seguimiento de graduados, escalabilidad de software, arquitectura de software, refactorización, sistemas de información.

2. PLANTEAMIENTO DEL PROBLEMA

El seguimiento de graduados es una condición de calidad obligatoria para las instituciones de educación superior en Colombia. *"El seguimiento a graduados es una condición de calidad obligatoria para las Instituciones de Educación Superior en Colombia. El Artículo 2.5.3.2.3.1.5 del Decreto 1075 de 2015, modificado por el Decreto 1330 de 2019, establece que las instituciones deben demostrar la existencia e implementación de mecanismos que promuevan el seguimiento de la actividad profesional de sus graduados (MEN, 2019)."* La ausencia de herramientas institucionales para este fin representa, por tanto, un riesgo directo para los procesos de acreditación de los programas académicos.

La Universidad Santo Tomás de Aquino Seccional Tunja cuenta con un sistema de seguimiento a graduados en la Facultad de Ingeniería de Sistemas, el cual no se encuentra desplegado ni activo, puesto que carece de una estructura para su despliegue y de una arquitectura para ser usado por otras facultades y, en general, por la universidad para el seguimiento a graduados. Está diseñado con un alcance limitado a la Facultad de Sistemas. Esta situación dificulta la gestión integral de la información sobre graduados a nivel institucional, lo que limita la generación de indicadores académicos y el análisis del impacto de los programas en el mercado laboral.

Ante esta problemática, se propone escalar el sistema existente mediante el rediseño de sus módulos. Al mismo tiempo, se propone un cambio en la interfaz de usuario y en las lógicas para que la universidad disfrute de una mejor experiencia de uso del aplicativo, lo que reduce las necesidades de mantenimiento a corto plazo.

3. JUSTIFICACIÓN

El sistema de seguimiento a graduados se da como un aerramienta util para el conocimiento y medicion efectiva de las disitntas facultades de la universidad brindando posibles indicios de nesesidad de vercado laboral con el interes de los estudiantes por posgrados junto con el segimiento donde se encuentra el usar esa infromacion para procimos graduados de distintas facultades fortaleciendo la toma de deciciones estrategicas relacionadas con la mejora continua de programas academicos De manera complementaria en el modelo de acreditacion “ CNA evalua en su factor 4 el segimiento y impoacto de los graduados como un criterio de alta calidad de los programas academicos de instituciones de alta calidad (CNA, 2020)” Dando esto como nesesidad intitucional debido a que se podria convertirse en un riesgo en la acreditacion de los programas de la universidad.

El desarrollo de un sistema escalable de seguimiento de graduados permitirá consolidar la información institucional sobre la trayectoria profesional de los graduados de los distintos programas académicos. Esto facilitará la generación de indicadores para los procesos de acreditación, el análisis del impacto laboral de los graduados y la toma de decisiones estratégicas por parte de la universidad.

La facultad de sistemas, la cual actualmente cuenta con un prototipo suscitado para la misma facultad, puesto que el programa se construyó con el fin de acoger la facultad de ingeniería de sistemas, pero no se encuentra activo, actualmente se puede ver necesario para las demás facultades. Siendo este proyecto totalmente pertinente debido a su correlación con la ingeniería de sistemas, dado que se realizarán análisis de requerimientos, diseño de arquitectura, implementación de tecnologías como NestJS, Angular y PostgreSQL, contenerización con Docker y automatización de despliegues mediante GitHub Actions . Y abordarlo implica rediseñar una arquitectura existente para soportar múltiples facultades, realizar el despliegue y garantizar la disponibilidad en producción.

4. OBJETIVOS

4.1. Objetivo General

Rediseñar una solución para el sistema de seguimiento de graduados de la Facultad de Ingeniería de Sistemas, orientándolo a una solución institucional multifacultad en la Universidad Santo Tomás Seccional Tunja, por medio de la refactorización de código, rediseño arquitectónico y despliegue con fines de centralizar la gestión de graduados y su información fortaleciendo los procesos de aseguramiento de calidad académica institucional

4.2. Objetivos específicos

Tabla 1. Objetivos específicos

Nro.	Objetivo específico
1	Identificar los requerimientos funcionales y no funcionales del sistema mediante entrevistas con actores clave y el análisis del sistema existente, para documentar el alcance de la multifacultad en una especificación de requerimientos formal.
2	Diseñar una arquitectura escalable multifacultad mediante la elaboración de diagramas UML de casos de uso, clases, secuencia y despliegue, así como el modelo relacional, con definición de controles de acceso por rol, para establecer las bases técnicas que soporten la expansión del sistema a todos los programas académicos de la universidad
3	Implementar las funcionalidades definidas en la especificación de requerimientos mediante la refactorización y la extensión del código existente, para obtener un sistema estable, escalable y alineado con las necesidades de gestión de graduados de las facultades.
4	Automatizar los procesos de integración y despliegue continuo mediante Docker y GitHub Actions, desplegar el sistema en los servidores institucionales y poner en operación la solución en su primera fase de expansión multifacultad.

Fuente: Autor

5. MODELO DE DESARROLLO

Para el desarrollo de este proyecto se adoptó una metodología ágil KANBAN; esta combinación se ajusta bien al contexto de un proyecto individual: permite mantener una estructura de trabajo clara y entregas periódicas, sin necesidad de un equipo scrum completo usando un tablero Kanban con columnas de ToDo(Backlog) , in progress y done , donde por medio de un tablero realizado en GitHub Projects , permitiendo visualizar el estado de cada tarea atendiendo primero los módulos críticos de escalabilidad multifacultad y despliegue y dividiendo la metodología en 3 fases:

Fase 1. Análisis del sistema actual:

Revisión del código existente brindado , análisis para identificar las posibles necesidades del sistema y conocimiento del prototipo, junto con una reunión con un actor clave para conocer las necesidades por parte de la universidad y, con ello, construir los requerimientos del sistema para la implementación multifacultad.

Fase 2. Diseño e implementación:

Diseño de arquitectura modular , refactorización del backend y del frontend , junto con la implementación de nuevos módulos y lógicas necesarias para la reestructuración del software, utilizando herramientas como el tablero kanban para la continuidad en el desarrollo de tareas, mediante Visual Studio Code, NestJS, Angular, PostgreSQL, TypeORM y otras que sean necesarias para el desarrollo o el cumplimiento de los requerimientos.

Fase 3. Integración y despliegue:

Dockerización del sistema, configuración de integración continua (CI/CD) con GitHub Actions y despliegue en servidores institucionales.

[illegible]

[illegible]

[illegible]

6. ENTREGABLES

Con el propósito de trazar y verificar los resultados, considerando la gestión del estudiante, se abarcó la entrega de cada resultado del proyecto en la tabla 2; además, se presenta la relación entre cada objetivo específico, las actividades asociadas, los entregables esperados y los medios de verificación correspondientes. Dado el mecanismo de desarrollo, junto con que se encuentra desplegado en el servidor institucional de la Universidad Santo Tomás Seccional Tunja, los medios de verificación corresponden a la evidencia objetiva del sistema en producción, incluyendo el repositorio en GitHub con historial completo de commits, capturas del sistema funcionando, lo cual permite evidenciar de manera concreta el cumplimiento de cada fase de desarrollo.

Tabla 2. Matriz de trazabilidad de objetivos, entregables y medios de verificación

Objetivo específico	Fase	Actividades principales	Entregable	Medio de verificación
OE1 — Identificar requerimientos	Requerimientos	La Entrevista , para la elaboración de historias de usuario, demanda la necesidad de generar un Backlog fue dada por el ingeniero Diego Alejandro Vela Beltrá Coordinador de Graduados y explicación estado por parte de Julián Camilo Escobar Araque Egresado y Diseñador de una solución a nivel facultad de un sistema de egresados	Entrevistas y reuniones con personal	Documento Resumen de reuniones

Objetivo específico	Fase	Actividades principales	Entregable	Medio de verificación
OE2 — Diseñar arquitectura y modelos	Análisis y Diseño	Elaboración de diagramas ER definición de arquitectura, construcción estructura Docker	Documento de Diagramas UML	Tablero Kanban en Github Project
OE3 — Construir módulos funcionales	Implementación	Desarrollo de módulos y control de versiones en GitHub	Sistema ejecutable con 100 % del backlog implementado	Demostración en vivo + repositorio GitHub con historial de commits
OE4 — Despliegue	Despliegue	Despliegue del programa junto con verificación de conexión con github Actions	Sistema en ejecución en servidor de producción	Demostración en vivo + Pipeline Github Actions

7. DESARROLLO DEL PROYECTO

7.1. Fase 1. Análisis del sistema actual

En esta fase se decidió tomar en cuenta el desarrollo realizado por el egresado y diseñador de la primera propuesta, Julián Camilo Escobar Araque, debido a que su propuesta de aplicativo cumplía correctamente con las bases necesarias para realizar una refactorización, proporcionando un punto de partida correcto para el desarrollo y el enfoque en la reestructuración de elementos necesarios para la puesta en producción.

Junto a esto se tomo la ayuda del Ingeniero Diego Alejandro Vela Beltrá quien gracias a su rol estrategico de cordinador de graduados permitio la correcta formulacion de los requerimientos nuevos del sistema.

7.1.1. Definición de requisitos

La entrevista , para la elaboración de historias de usuario, da la necesidad de generar un Backlog fue dada por el ingeniero Diego Alejandro Vela Beltrá, siendo él quien permitió la comunicación de las necesidades de la universidad y el análisis de las necesidades existentes mediante una presentación del software brindada por Julián Camilo Escobar Araque, quien propuso en un comienzo el sistema, a lo cual se encontraron algunas necesidades principales del sistema, las cuales se dividieron en los 9 módulos para mejorar el sistema, los cuales son :

1. Autenticación y Seguridad
2. Gestión de graduados
3. empleados y posgrados
4. noticias
5. eventos
6. carrusel
7. facultades
8. docentes / ingenieros
9. estadísticas y gráficos

Los cuales fueron modificados para la mejora del sistema, dándonos como resultado que se presenten los requerimientos funcionales y no funcionales identificados:

Tabla 2. Requerimientos

ID	Requerimiento
RF01	permitir autenticación con JWT que incluya rol y facultad
RF02	Token JWT debe ser requerido en todas las rutas privadas
RF03	Frontend debe verificar y renovar automáticamente el token
RF04	permitir crear, consultar, editar y eliminar cuentas de usuario
RF05	Usuario debe tener nombre único, contraseña, rol y facultad opcional
RF06	Usuario autenticado puede cambiar su propia contraseña
RF07	listar graduados con paginación y filtro automático por facultad del usuario
RF08	permitir actualización parcial de datos del egresado incluyendo empleos y postgrados
RF09	carga masiva de graduados mediante CSV con validación
RF10	exportar todos los graduados a CSV
RF11	Modificar y reparar errores eventos y noticias
RF12	Visualización de graduados y estadísticas debe estar segmentada por facultad del usuario
RF13	Iniciar por primera vez el sistema debe crear automáticamente los datos base necesarios
RF14	Sistema debe ofrecer panel administrativo privado para gestión de contenido
RF15	Todas las colecciones deben soportar paginación
RF16	Eliminaciones lógicas deben mantener integridad de datos relacionados y ser reversibles
RF17	Filtrar todos los datos dependiendo del rol y de la facultad a la que se pertenezca

Fuente: Autor

Tabla 3. Requerimientos no funcionales

ID	Requerimiento
RNF01	Backend debe estar organizado en módulos independientes para facilitar mantenimiento y escalabilidad
RNF02	debe ejecutarse en cualquier entorno mediante Docker Compose
RNF03	Dependencias deben estar fijadas mediante lockfile para garantizar builds reproducibles
RNF04	La API debe estar documentada para explorar y probar endpoints sin código externo
RNF05	Servicio de imágenes debe imponer restricciones de tipo y categoría
RNF06	Interfaz debe ser grata para el usuario y ofrecer modo claro y oscuro
RNF07	Todos los recursos privados deben estar protegidos y segmentados correctamente por rol
RNF08	La interfaz debe ser limpia y comprensible y mantener similitud a marca de la universidad

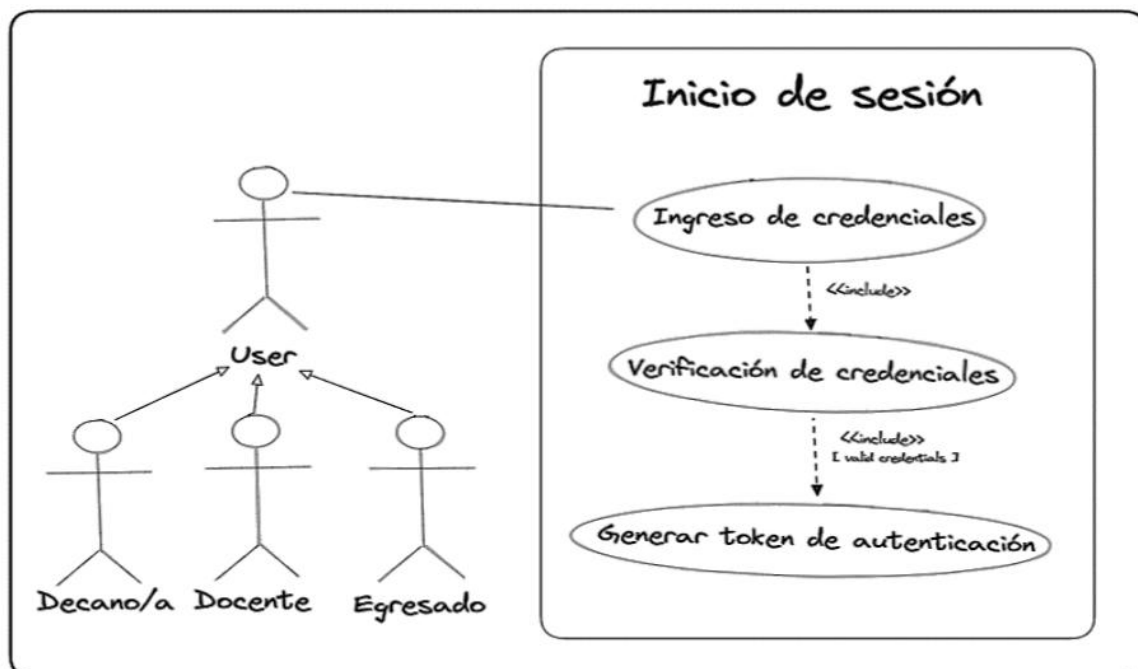
Fuente: Autor

7.1.2. Diagramas de casos de uso

Para la elaboración de los diagramas de caso de uso se adoptó una guía en base a los cambios que se le realizarán a los diagramas propuestos por J. C. Escobar en (Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja 2025).

Tomando en cuenta los requerimientos se van a modificar los diagramas de casos de uso desarrollados por J. C. Escobar en (Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja 2025)

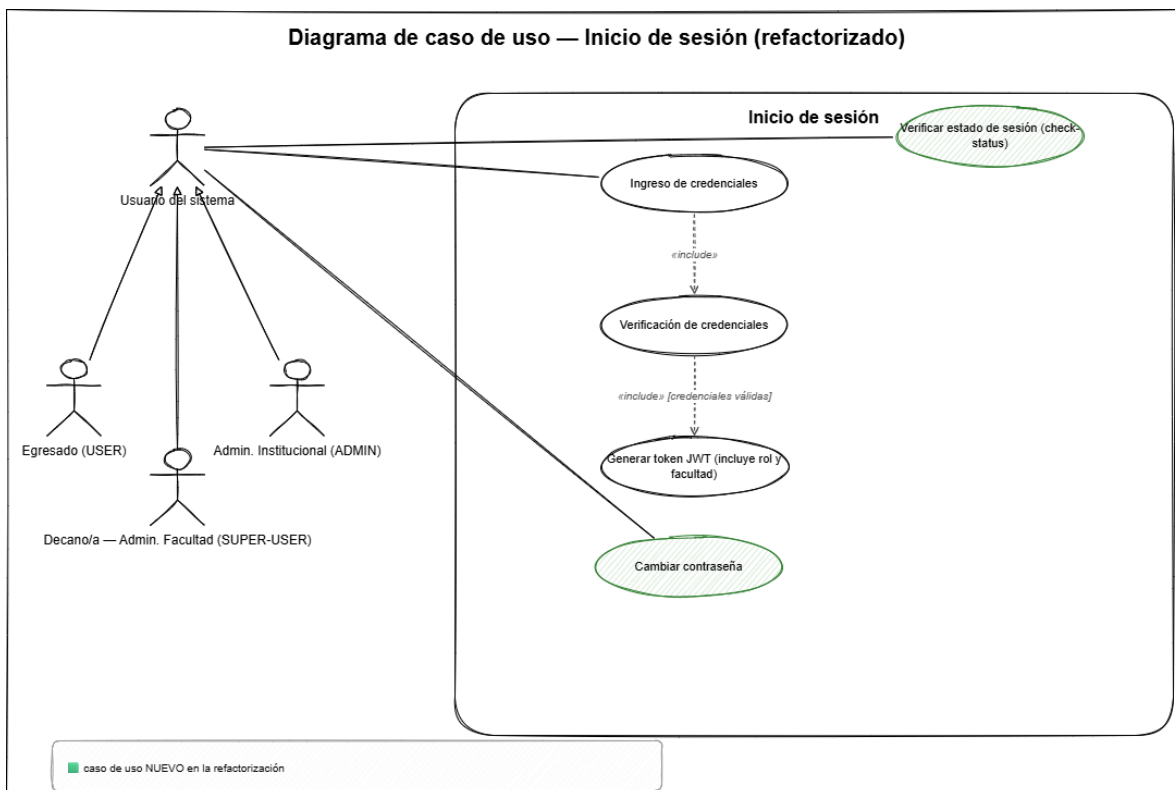
Figura 1 Diagrama de caso de uso Inicio de sesión



Nota. Tomado de Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja (p. XX), por J. C. Escobar, 2025.

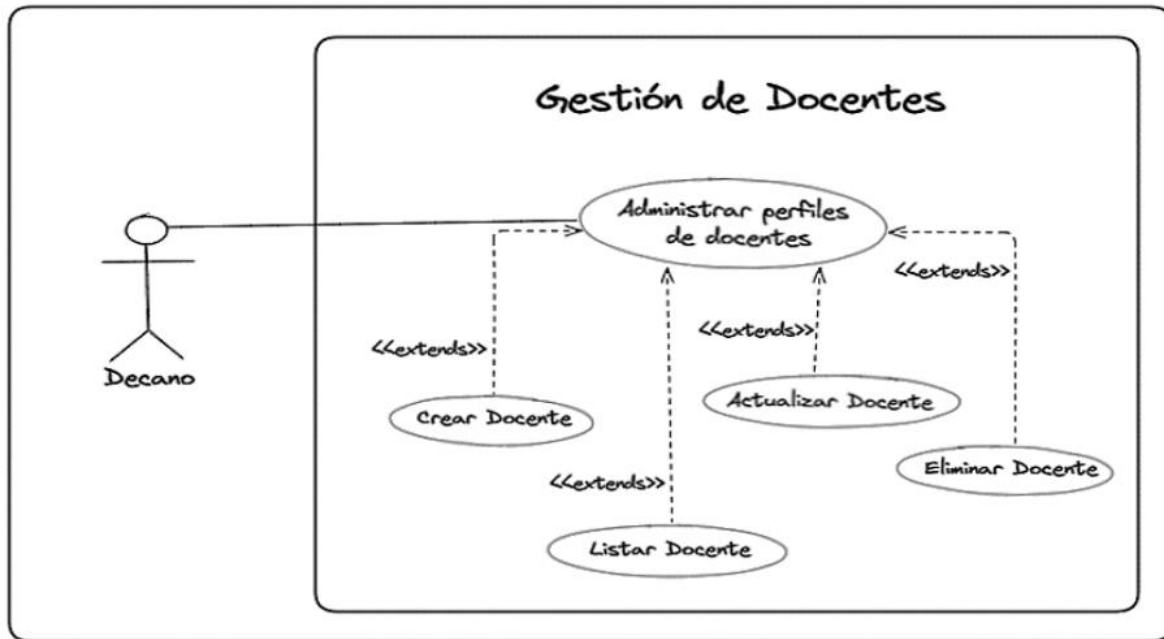
El principal cambio de que se encontró para realizar en este cos fue la verificación de sesión junto con que el usuario pueda cambiar contraseña junto con un ajuste a los roles y usuarios además de que se genera el token de autenticación y incluir rol y facultad con ellos se alcanzan de mejor manera y se podrán cumplir con los requerimientos RF01 , RF03 y RF06

Figura Diagrama de casos de uso Inicio de sesión refactorizado



Fuente: Autor

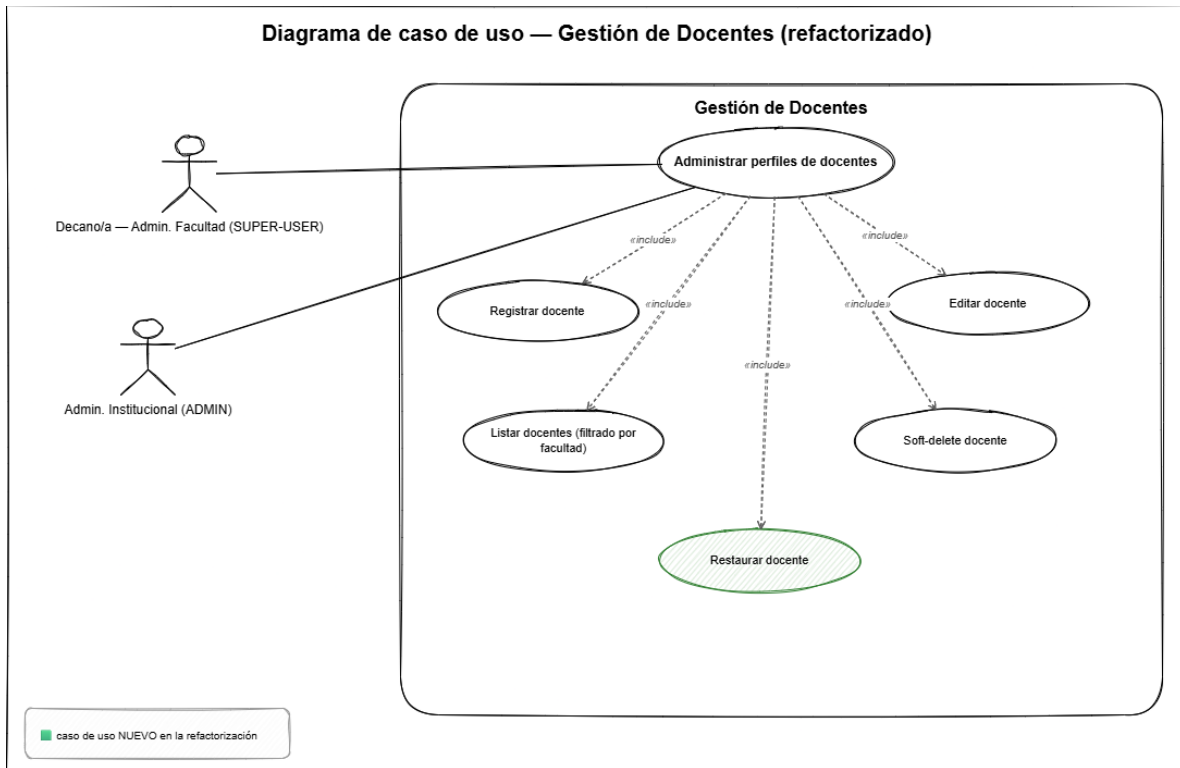
Figura 2 Diagrama de caso de uso Gestión de docentes



Nota. Tomado de *Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja* (p. XX), por J. C. Escobar, 2025.

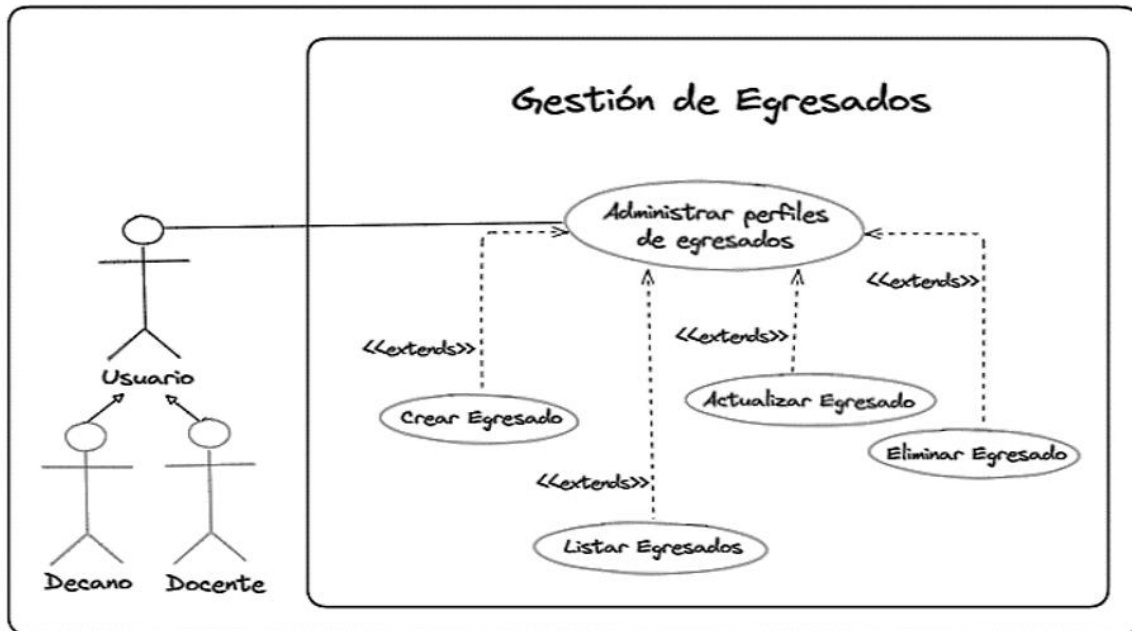
Gestión de Docentes se alteró para dar un alcance mejor a controlar la entrada de admin y super user junto a que este se verá filtrado por facultad y se agregaron cambios de soft delete para los posibles errores que se puedan encontrar entorno a la restauración de datos siendo que las eliminaciones lógicas sean reversibles con su integridad dando cumplimiento al RF16

Figura Diagrama de caso de uso — Gestión de Docentes (refactorizado)



Fuente: Autor

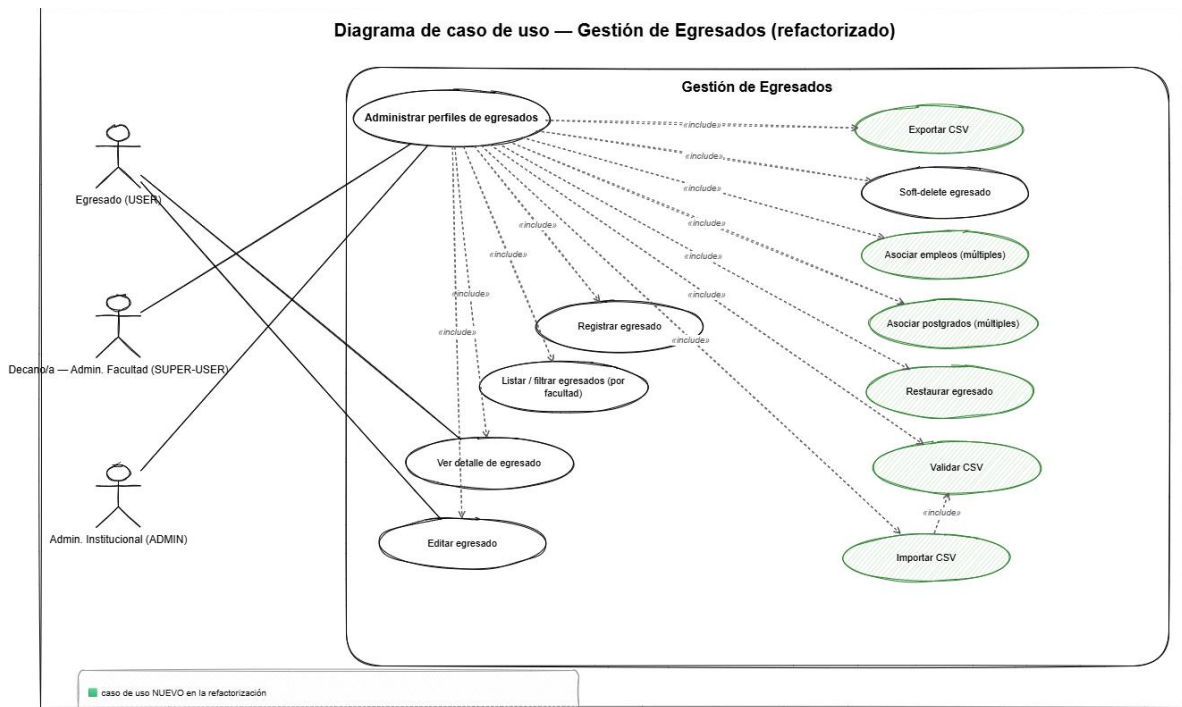
Figura 3 *Diagrama de caso de uso Gestión de egresados*



Nota. Tomado de Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja (p. XX), por J. C. Escobar, 2025.

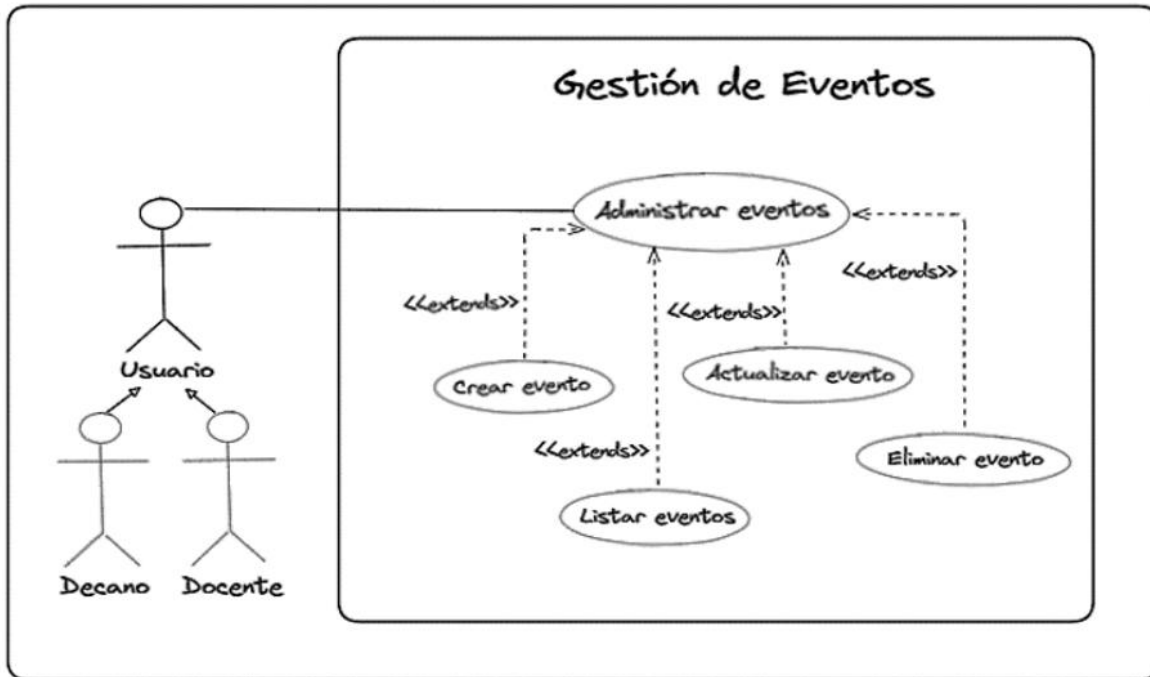
Este diagrama fue uno de los más afectados debido generado un filtro para graduados por facultad junto con que los posgrados y trabajos deberian ser listas buscando asi una relacion de uno a muchos ya que existe la posibilidad de que un estudiante tenga mas de un trabajo o posgrado junto a ello se agrego un sistema de soft delete junto con validacion de los csv con ello cumpliendo requerimientos RF04, RF08, RF09 ,RF10 ,RF12 y RF16.

Figura Diagrama de caso de uso — Gestión de Egresados (refactorizado)



Fuente: Autor

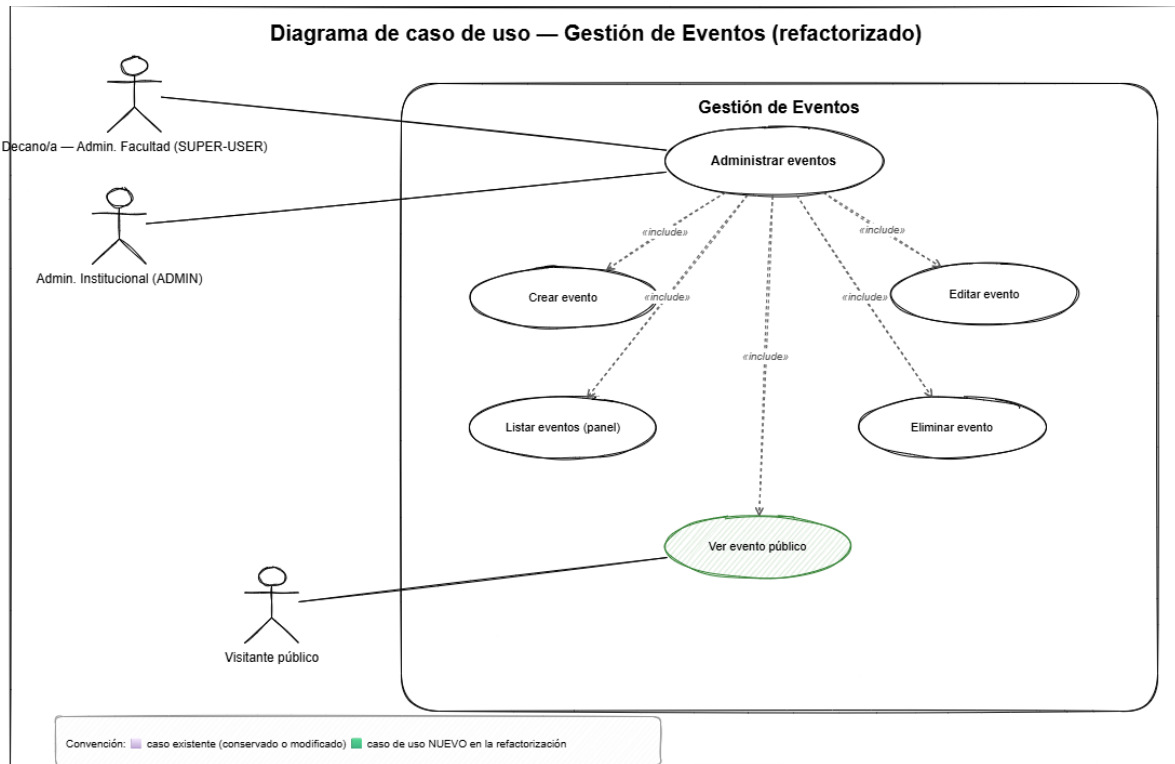
Figura 4 Diagrama de caso de uso Gestión de eventos



Nota. Tomado de Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja (p. XX), por J. C. Escobar, 2025.

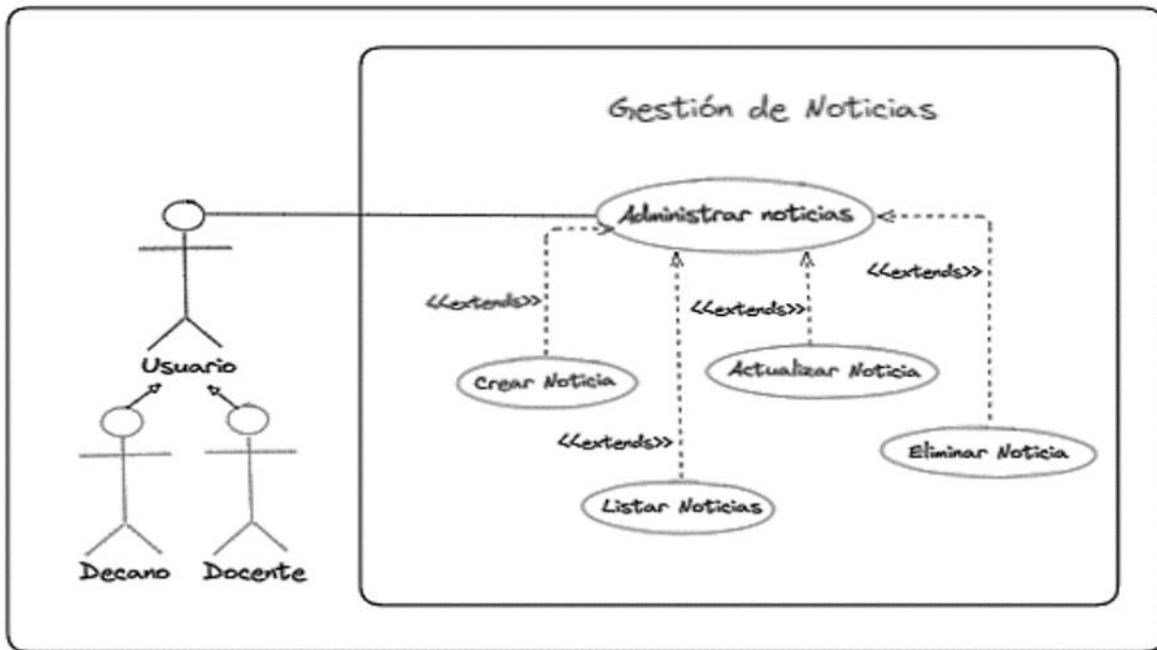
En eventos se modifica al manipular la vista de eventos publicos y al nivel de acceso de roles junto a ello se realiza un modificacion para la solucion al RF11 acceso limitado solo admin o superior

Figura Diagrama de caso de uso — Gestión de Eventos (refactorizado)



Fuente: Autor

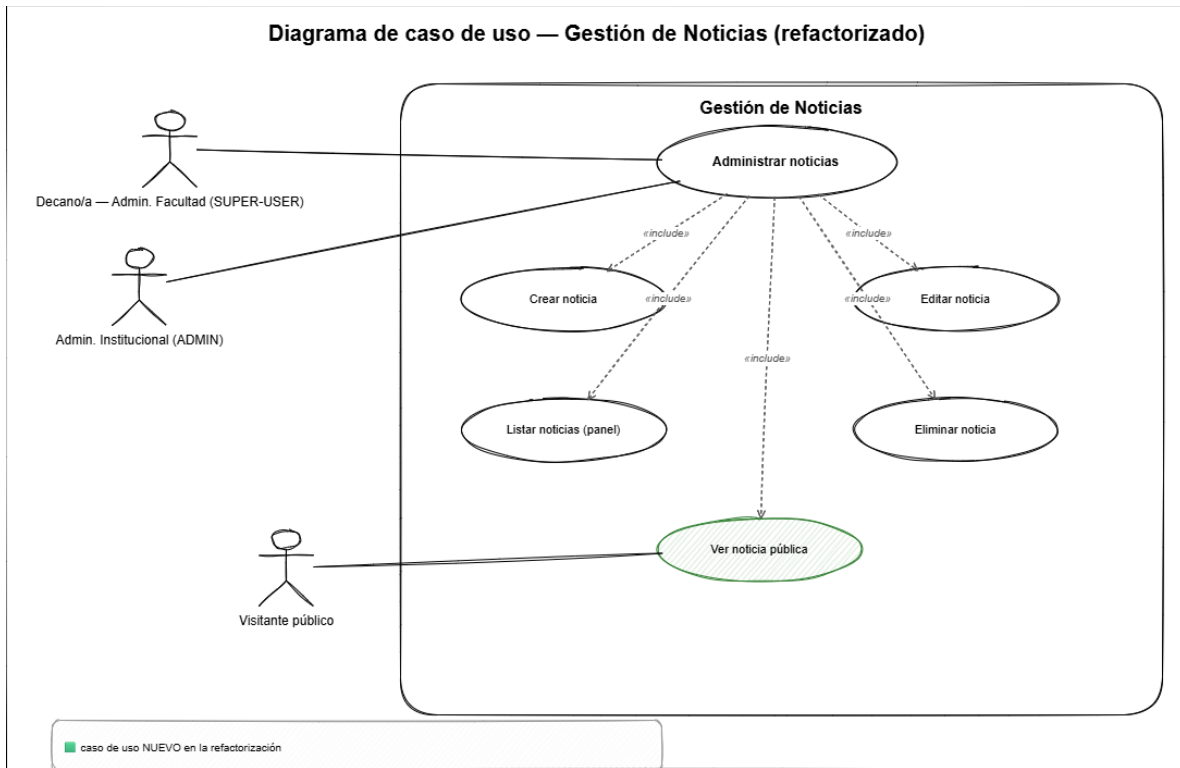
Figura 5 Diagrama de caso de uso Gestión de noticias



Nota. Tomado de Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja (p. XX), por J. C. Escobar, 2025.

En eventos se modifica al manipular la vista de noticias publicas y al nivel de acceso de roles junto a ello se realiza un modificacion para la solucion al RF11 y brindando acceso limitado solo admin o superior

Figura Diagrama de caso de uso — Gestión de Noticias (refactorizado)



Fuente: Autor

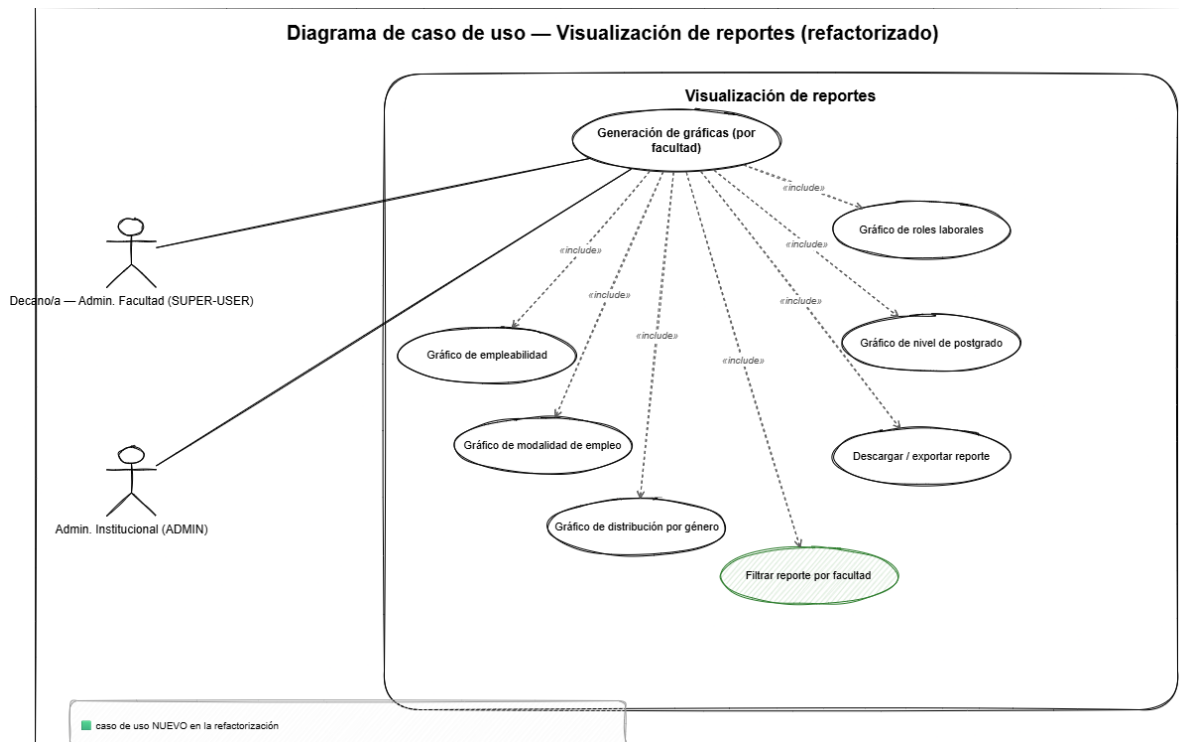
Figura 6 Diagrama de caso de uso Visualización de reportes



Nota. Tomado de Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja (p. XX), por J. C. Escobar, 2025.

En la visualización de reportes la generación gráfica filtradas por facultad junto al acceso correspondiente segun su rol dando solucion a una parte muy importante en la expansion multifacultad y cumpliendo el RF17.

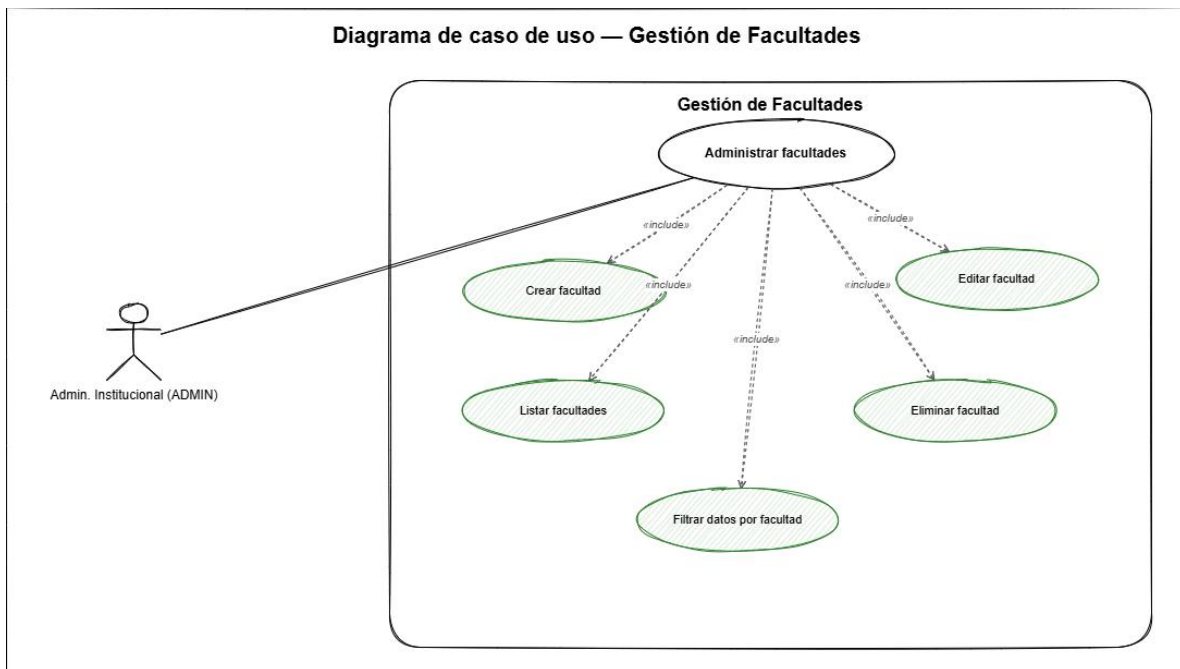
Figura Diagrama de caso de uso — Visualización de reportes (refactorizado)



Fuente: Autor

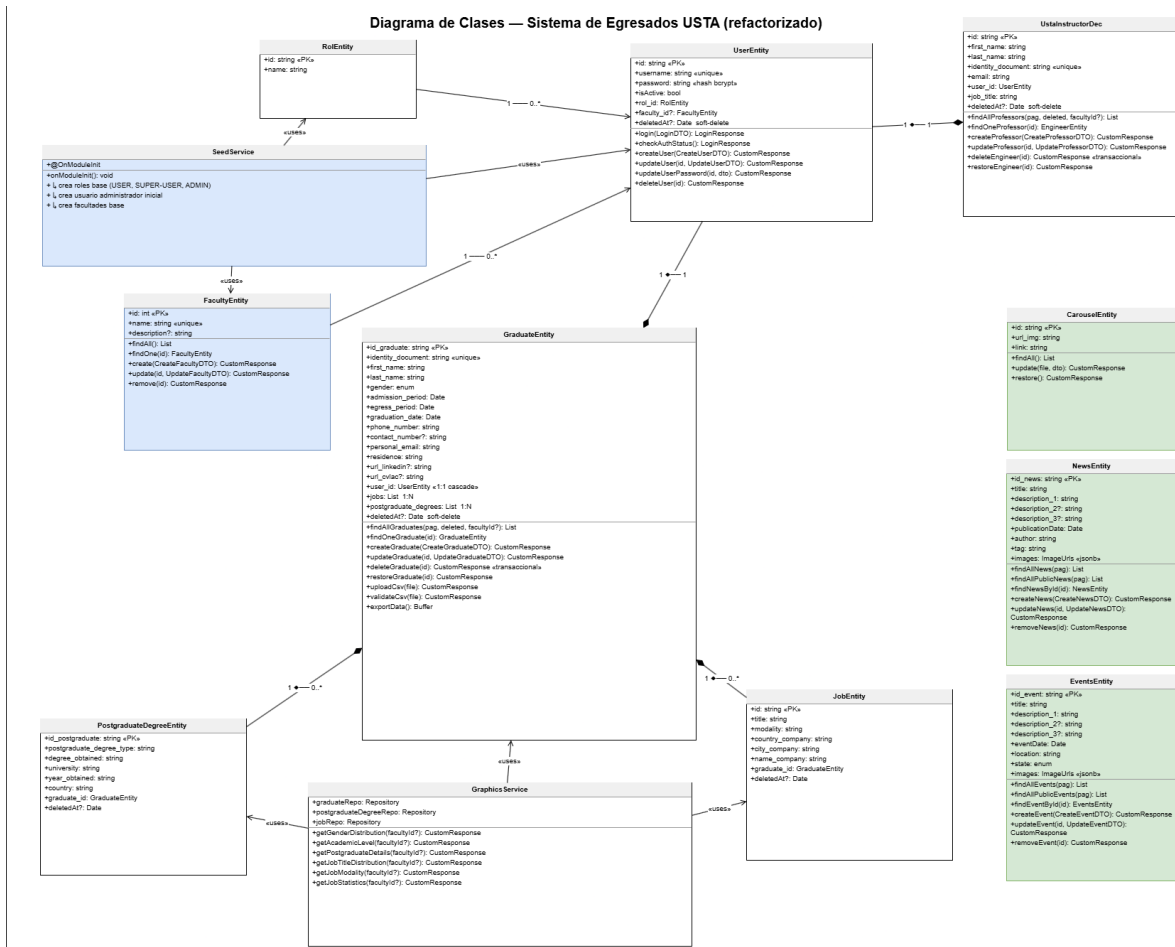
Como nueva parte se agregara la gestion de facultades permitiendo ampliar a medida que la universidad lo necesite otras facultades y poder in agregando valor al sistema permitiendo que la generacion de nuevas facultades dependa de la facultad y este siendo un paso muy importante para el cumplimiento del RF17 debido a que es uno de los pilares fundamentales de esta expansion.

Figura Diagrama de caso de uso Gestión de Facultades



Fuente : Autor

Figura Diagrama De Clases



Fuente : Autor

7.1.3. Fase 1.2 Selección de tecnología

Debido a la base de código existente brindada por Julian Camilo Escobar Araque se decidió continuar con las tecnologías usadas por el Angular NestJS PostgreSQL dada esta decisión permitió preservar la continuidad del proyecto, escalabilidad e implementar los nuevos módulos. Como complemento de la arquitectura existente se agregó el uso de Docker y Docker compose para la orquestación de los servicios en contenedores permitiendo que el sistema

completo pueda ejecutarse en cualquier entorno con un único comando adicionalmente se genero un repositorio en github opara control de versiones y gestión de historial de cambios

Tabla 2. Herramientas tecnológicas utilizadas

Tecnología / Herramienta	Versión	Uso en el proyecto
TypeScript	5.x	Superset de JavaScript utilizado para tipado estático usado en el backend, frontend y servicio de imágenes
PostgreSQL	16.x	Base de datos relacional principal
Postman	11.x	Herramienta para el diseño y ejecución de pruebas de integración sobre los endpoints de la API
Git	2.44.x	Sistema de control de versiones del código fuente
Swagger	7.4	Documentación automática de la API con Swagger
GitHub	—	Plataforma de alojamiento del repositorio y gestión de commits e historial del proyecto
pgAdmin 4	latest	Administración gráfica de la base de datos
NestJS	10.x	Framework principal del backend, módulos, controladores y servicios
Angular	18.x	Framework principal del frontend, componentes y enrutamiento
Docker	—	Contenerización de todos los servicios
Docker Compose	—	Orquestación multi-contenedor para desarrollo y producción
pnpm	10.17.0	Gestor de paquetes con lockfile para builds reproducibles
TypeORM	0.3.x	ORM para mapeo objeto-relacional con PostgreSQL
Visual Studio Code	1.89.x	Editor utilizado para el desarrollo en TypeScript

7.1.4. Fase 2. Diseño del sistema

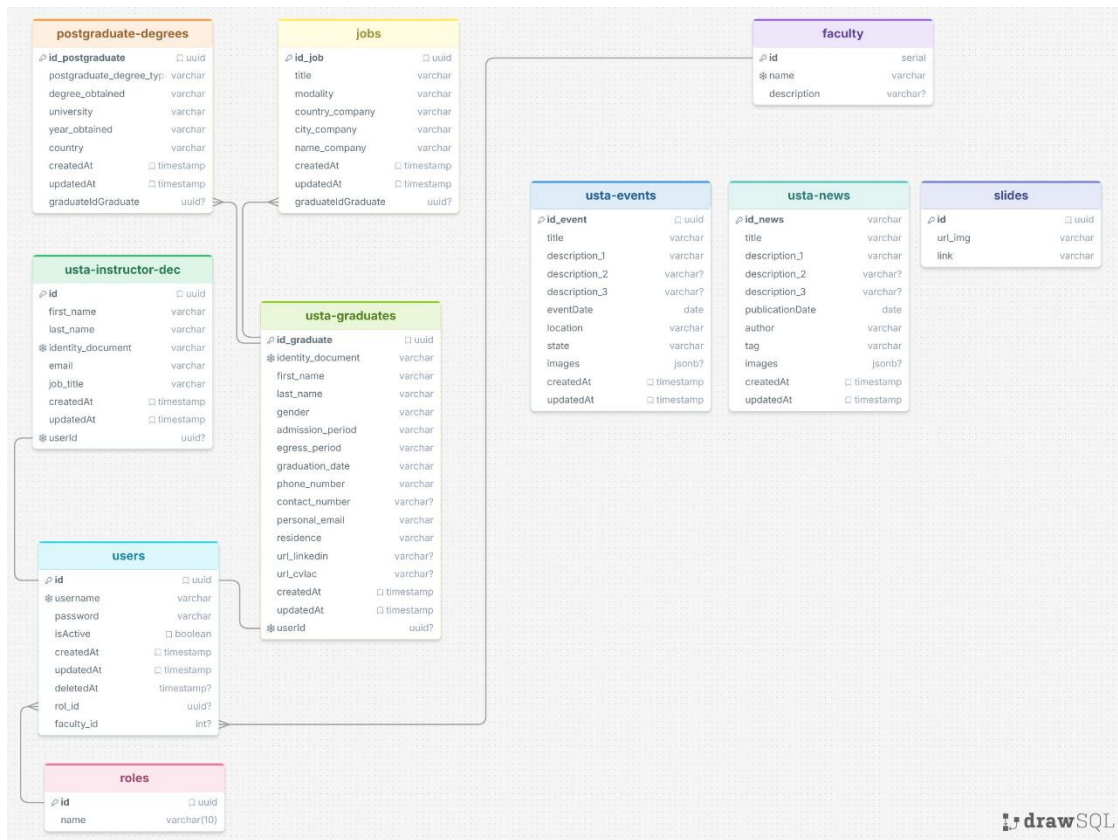
En esta sección se deben incluir los diagramas que representen el diseño del software, por ejemplo: casos de uso, entidad-relación, clases y demás que, junto con el director, consideren pertinentes. Además, se deben explicar

Tal como se reestructuró se decidió cambiar la base de datos cambiando elementos de las relaciones las cuales fueron pertinentes cumplir con los requerimientos propuestos en los cuales se modificaron como principales:

- relaciones entre “usta-graduates” a las tablas de “jobs” y la tabla de “posgrate-degrees” la cual era de 1 – 1 y fueron movidas a relaciones 1 – n para poder generar varias instancias de trabajos o de posgrados según el egresado lo necesite
- Se renombró la tabla de “engineers” a “usta-instructor-dec” haciendo referencia a que es la tabla de usuarios que tienen acceso de admin ofreciendo allí un cambio para un número de identificación
- Se agregó en la tabla user un campo de isActive para conocer si se encuentra activo o no y agregando una relación de n - 1 dándonos allí una manera de conectar tanto a los “usta-instructor-dec” como a los “usta-graduates” manipulándolos como extensiones de usuario el cual está alineado con una facultad la cual se agregó la nueva tabla de “faculty”
- De esta manera logrando una refactorización de la arquitectura ofreciendo cambios junto con agregar a las tablas importantes datos de auditoría como lo fueron “createdAt” , “updateAt” y “DeletedAt”

tal como se muestra en el siguiente diagrama ER:

Figura 6. Diagrama Entidad-Relación.



Fuente: Autor

7.2. Fase 2.2 Desarrollo de la aplicación

Describir la estructura lógica del proyecto e incluir las evidencias pertinentes (propuestas en la sección de entregables). No incluir código, sino la explicación de los componentes principales, la arquitectura, el estilo de diseño visual, etc.

En esta fase se comenzó con un análisis de uso de la refactorización que iba a ser necesaria dependiendo de los elementos que ya tenía el software. Ante ellos se decidió revisar y comenzar con un desarrollo modificando inicialmente la base de datos para llegar al punto donde se deseaba, dado que facilitaría el desarrollo próximo, dado que ayuda al desarrollo del backend continuando con el API REST y facilitando la integración y de esa manera continuar con el front permitiendo un avance a modificar el interfaz asegurando la sincronía de los datos y el frontend.

El desarrollo de la aplicación de sistema de gestión de egresado para la usta se utilizo el stack que se llevaba usando para mantener la base de código y con ello poder mejoralo y darle una mayor escalabilidad se agrego un sistema de roles de ADMIN · SUPER-USER · USER los cuales a ecepcion edel super user todo tienen filtrado dependiendo de la facultad ala cual pertencescan se comenzó con un desarrollo por módulos para dar cumplimiento a los requerimientos:

Autenticación y Seguridad:

Para este cumplimiento se modificaron tanto el JWT como la revisión de cada endpoint de la información a enviar , el endpoint `'POST /auth/login'` , tener el rol , su facultad y que su contraseña estuviese hasheada con bcrypt, y `'GET /auth/check-status'`, y en rutas privadas se revisan el rol y su facultad para así dar mayor seguridad. Junto a ello, el administrador debe poder crear cuentas asignadas a los otros roles y delegar accesos de forma controlada al sistema.

- Contraseña generada con patrón `'$Ab{documento}'`
- Validación de correo único y datos requeridos

Asimismo, se actualizaron los endpoints de autenticación y manejo de los usuarios existentes para adaptarse al nuevo uso que se les daría, siendo esos:

- `'GET /auth/users'`
- `'PATCH /auth/update-user/:id.'`
- `'DELETE /auth/delete-user/:id.'`

Junto a ellos, se añadió el control de acceso por rol, en el que se agregaron a cada endpoint decoradores de autenticación `@Auth(...ValidRoles)` para que los accesos a la información se filtraran correctamente.

Luego se profundizó en el desarrollo del módulo 2 de Gestión de Graduados:

Se modifiko el resgistro de graduados modificando las listas y métodos de búsqueda de los graduado a que dependan de la facultad se agrego un eliminar a un egresado junto con agregar borrado en cascada para evitar problemas y se agregaron botones modificados para ver importar y exportar csv de el nuevo modelo de graduadoagregando la tabla de relación con facultades en la cual conecta a los graduados con una facultad y con ellos maneja mejor el estado de las listas y podemos manejar filtros por facultad.

En el módulo de noticias se agregó al super usuario y al administrador la parte de publicar noticias se escaló al servicio de imágenes a png y máximo de 10 megas para evitar cargas largas y esperas

y las listas se agregaron con acciones para editar y eliminar para poder manejar y administrar el panel de noticias más fácil junto con un editar y un borrar para poder completar el crud en noticias.

Con este mismo enfoque se desarrolló el módulo de eventos buscando expandir su uso sin el cual se encontraba limitado por la data de claridad en los endpoints anteriores donde estaban como un plus se decidió tomar para informar sobre campañas y información venidera con mayor margen de acción para el administrador

En el módulo del carrusel se modificaron algunos elementos y se generaron imágenes reemplazables para mejorar su uso y adaptarlo a los cambios realizados en el servidor de imágenes.

El modulo de facultades este se creo de sde cero se agregaron los enpoinis correconientes para eliminar crear buscar poid editar por id y listar con ellos se realizo la interface para estos endpoints nuevos con el fil de que pudieran ser manejado y se agrego una relación de este hasta graduados y luego a docentes para poder clasificar a todos los usuarios de la plataforma así mismo se modificaron los enpoinis de engineers para poder ser manejado por facultad

En el módulo gráfico y estadístico se modificaron los cálculos tanto como los enpoinis para que estos se vieran filtrados dependiendo del acceso que tenga aquel que lo vea y se repararon errores encontrados en distribución por género por trabajo y por posgrado los cuales presentaban errores ya que no estaban completados estos módulos

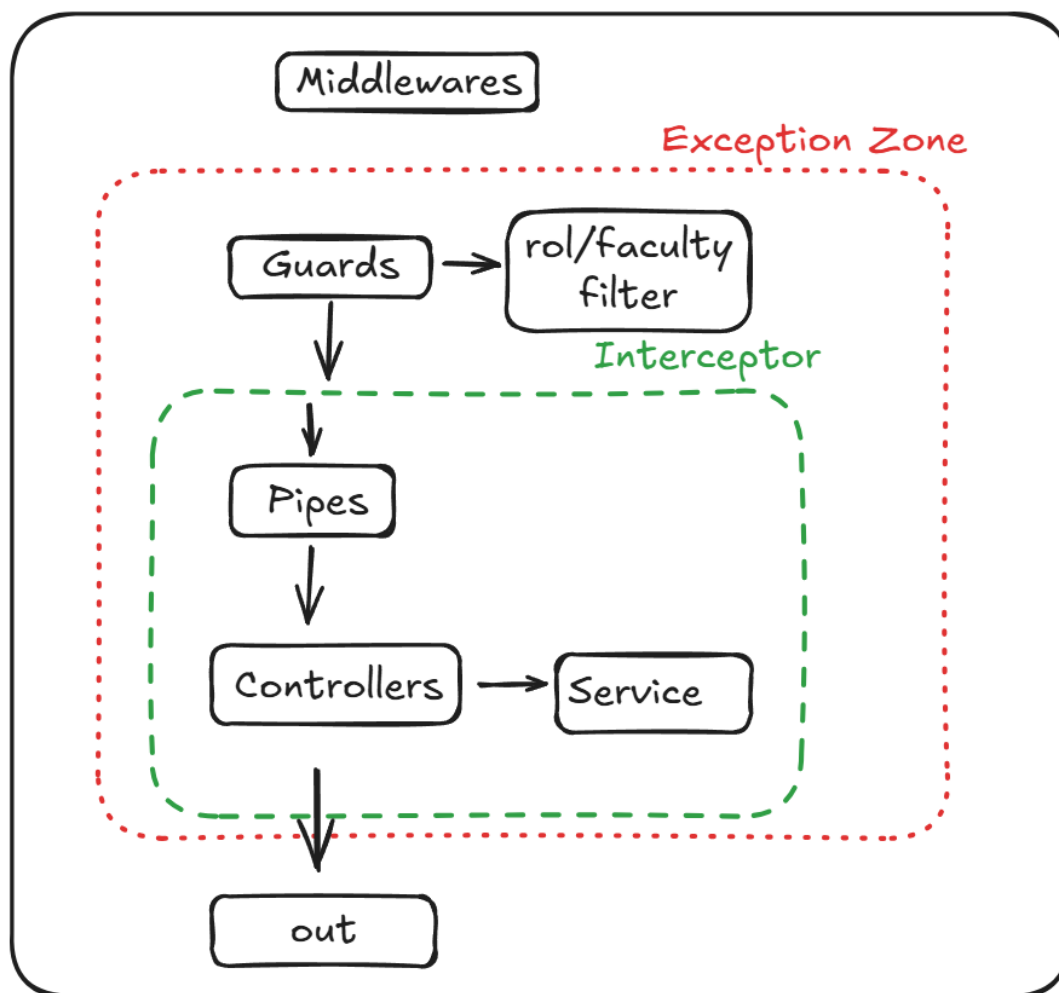
Junto a ellos y la presentación a ingeniero Diego Vela en el fin de semestre 2025-2 en los comienzos del semestre 2026-1 se propuso una mejora gráfica al sistema para facilitar la comprensión de los elementos en la cual se refactorizó toda la interface buscando una afinidad con los emblemas y el lenguaje visual de la universidad dándonos como resultado elementos visuales en 2 esquemas de color claro y oscuro.

En la última fase se desarrollaron varios cambios importantes en la plataforma agregando un CICD para darle un mejor control al software y con ello pudiendo manejar en futuro el despliegue continuo de la plataforma por medio de github Actions

7.2.1. Fase 2.3. Ciclo de Vida – NestJs

En la figura 8 se da el modelo actualizado de ciclo de vida de NestJs, en el cual se agregó en guards la verificación de rol y de facultad para el procesamiento correcto a través de los módulos y servicios, proporcionándonos una visión más amplia del backend, siendo así un método seguro de validación de información a enviar y de evitar fugas de datos, proporcionando mayor seguridad .

Figura 8. Ciclo de vida - NestJS



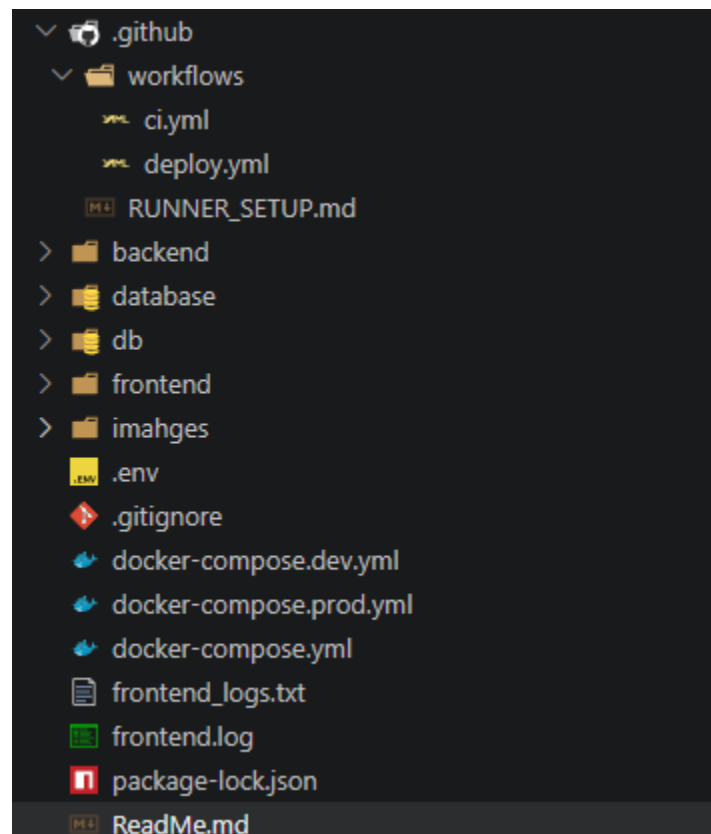
Fuente: Autor

7.2.2. Fase 3 Estructura del proyecto

La estructura se rediseñó para presentar un modelo capas de desplegarse por medio de Docker Compose y GitHub Actinos en el servidor de la universidad dada la necesidad de una base de datos Postgres junto con el despliegue de los servidores usados se decidió optar por el siguiente ordenamiento de carpetas brindando un enfoque ordenado y modular buscando almacenar la base

dedatos a esta altura junto con los archivos Docker compose que se encargara de separar los entornos de producción y desarrollo junto con la parpeta de .github con los elementos necesarios para el pipeline de despliegue de github actions.

Figura 9. Ordenamiento carpetas raíz



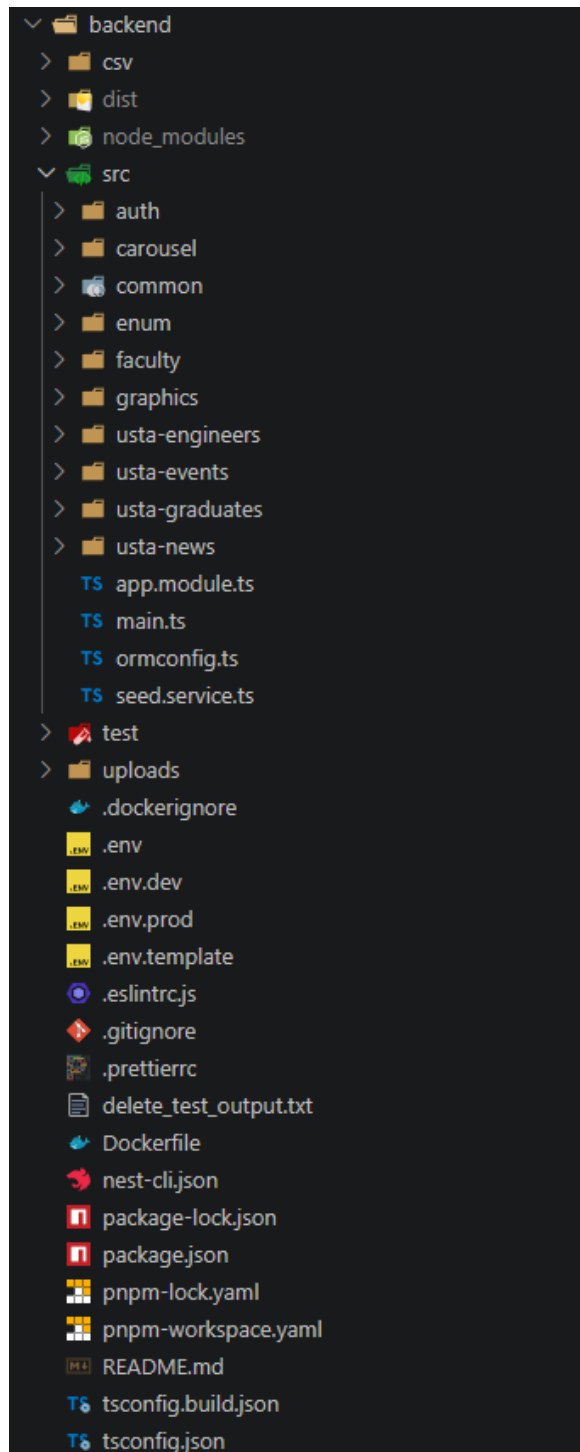
Fuente: Autor

7.2.2.1 Fase 3. 1 Estructura backend

La estructura del backend se mantuvo debido a que es la recomendada por nestJs siendo esta una estructura modular donde cada modulo del negocio es una carpeta brindando alta cohesión, acoplamiento y cada modulo teniendo un 'espec.ts' para las pruebas unitarias . Dado esto se agregaron los nuevos módulos para que typeorm creara la estructura en la base dedatos y entre ello los cambios antes mencionados en la véase 7.3 (Fase 3. Diseño del sistema) en los cuales se vieron modificados varios archivos preexistentes junto a ello se agregaron configuraciones tanto al orm como configuraciones de ingesta de datos base para el inicio correcto del programa rellenando facultades , junto con la creación de el user super admin junto con roles y información base necesaria

para el correcto inicio del sistema junto a ello las ritas privadas se vieron modificadas debido al cambio de lógica en el filtro de información dependiendo de la facultad.

Figura 10. Ordenamiento carpetas Backend



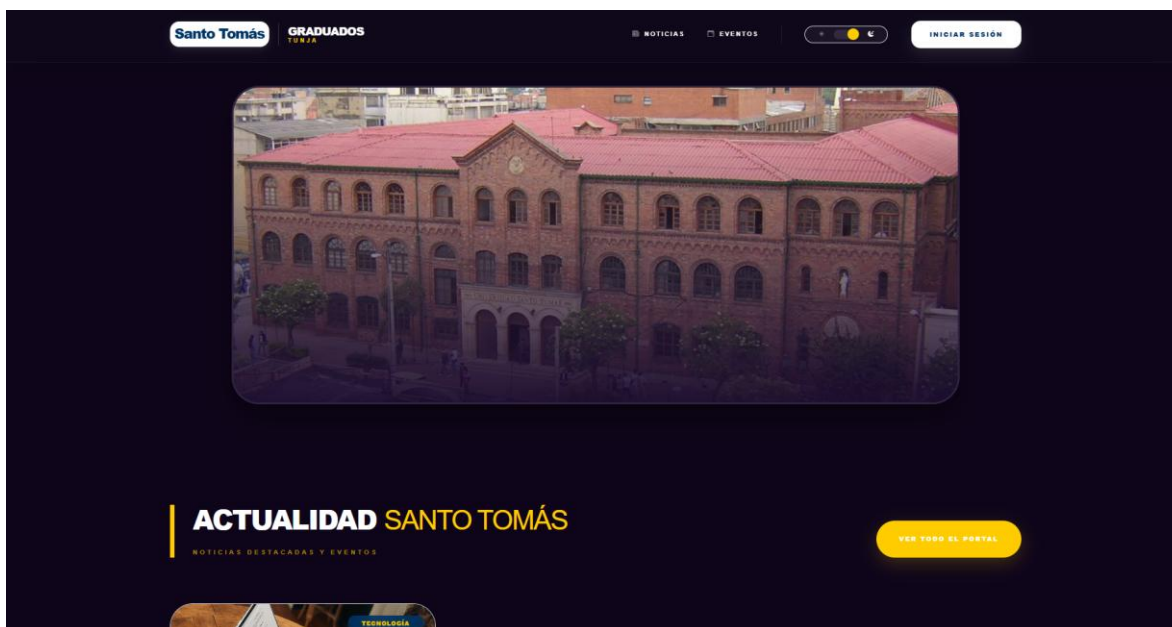
Fuente: Autor

7.2.2.2 Face 3. 2 Estructura frontend

Se mantuvo la Separacion por nivel de Acceso el cual es estándar de angular la cual es una división por responsabilidad y audiencia mas que por lo técnico lo cual es muy útil brindando un enfoque perfecto al programa que se desarrollo a esta estructura se agregaron siguiendo el orden de desarrollo se modificaron las lógicas necesarias junto con las interfaces para dar mayor coherencia visual con la universidad junto con la integracion y cambios realizados en el backend.

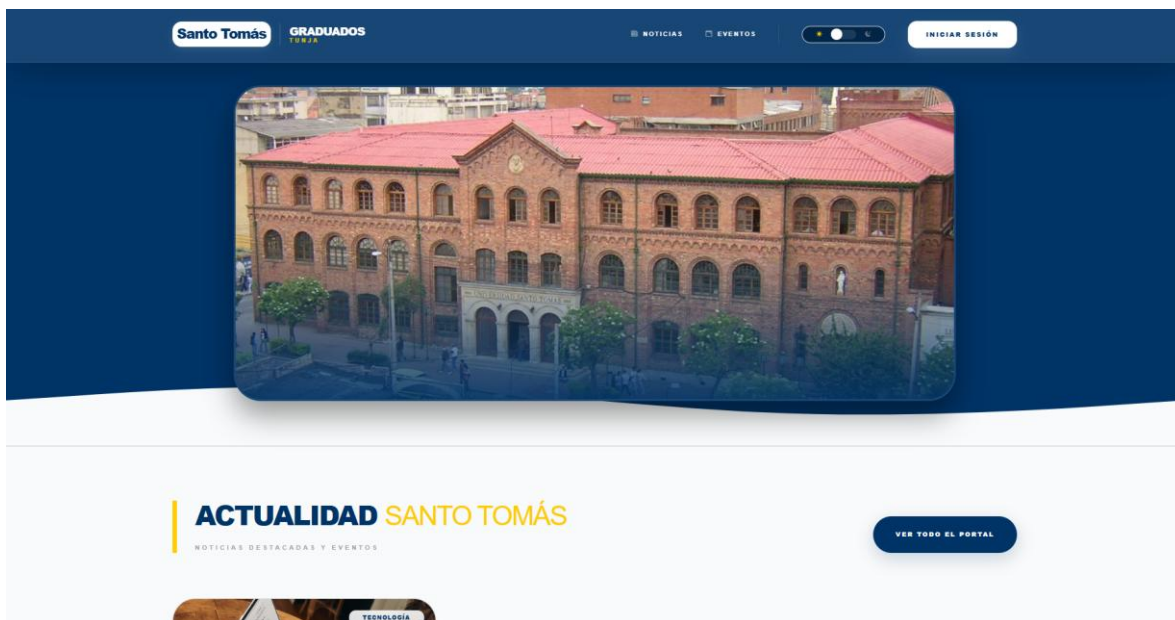
Junto a ello, según la revisión, el ingeniero Vela propuso una reestructuración de la interfaz, solicitando un modo oscuro y otro claro, así como un diseño más acorde con los colores de la universidad.

Figura. Pantalla de Home Oscuro



Fuente: Autor

Figura 10. Pantalla de Home Claro



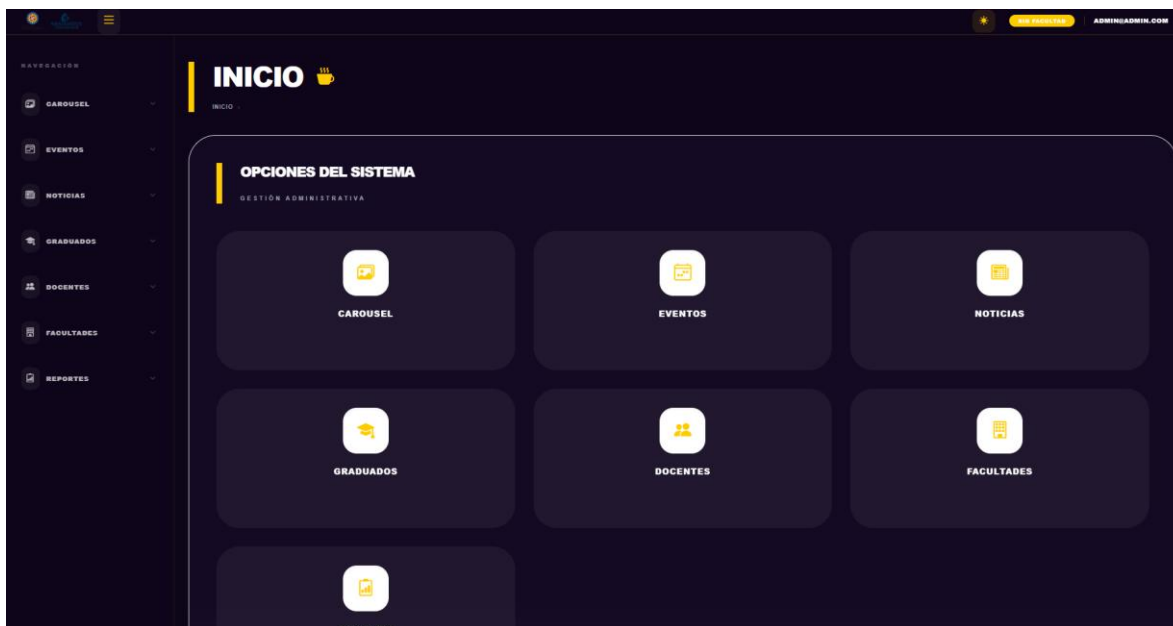
Fuente: Autor

Figura 10. Pantalla de Login



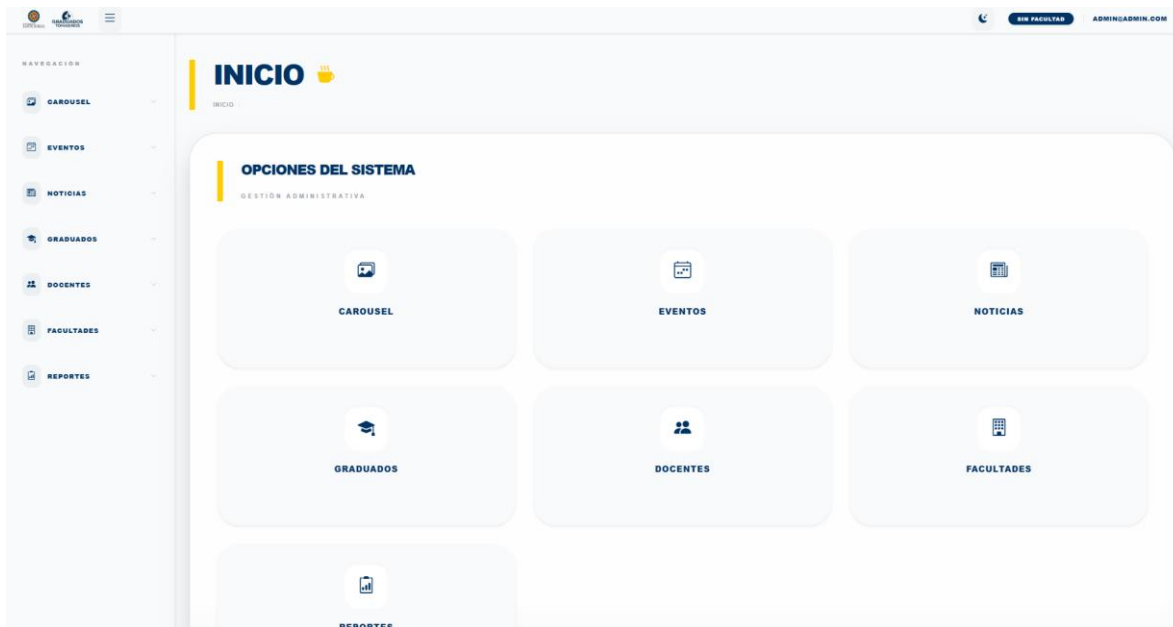
Fuente: Autor

Figura 10. Pantalla de Inicio Oscuro



Fuente: Autor

Figura 10. Pantalla de Inicio Claro



Fuente: Autor

Figura 10. Muestra edición oscura



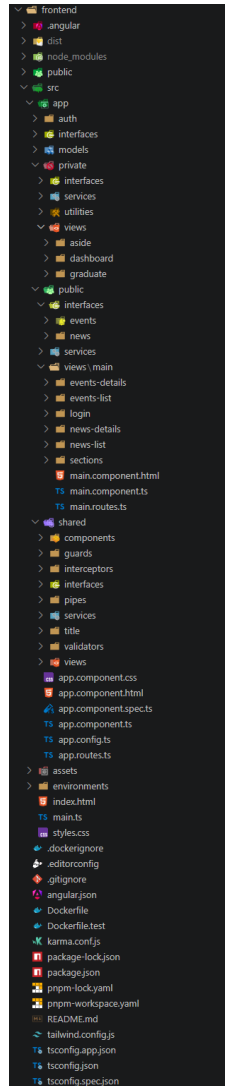
Fuente: Autor

Figura 10. Muestra edición oscura



Fuente: Autor

Figura 11. Ordenamiento carpetas front-end



Fuente: Autor

7.2.3. Fase 3.3 Documentación

La documentación central se encontraba en Swagger, la cual fue mejorada para proporcionar un mejor entendimiento de cada endpoint, la cual se encuentra en la ruta del servidor:3100/docs, lo cual es la forma recomendada por Swagger. En ella se agrega un total de 8 módulos con sus correspondientes endpoints y, junto a ello, los esquemas de Dto de los envíos de información, los cuales fueron utilizados. Brindándonos documentación más completa sobre el funcionamiento de cada endpoint.

Figura 15.

1

1

Autentica al usuario y retorna un token JWT con su rol y facultad.

Parameters

Cancel

No parameters

Request body **required**

```
application/json
```

```
{
  "username": "admin@admin.com",
  "password": "$Ab123456"
}
```

Execute

Clear

Responses

Curl

```
curl -X 'POST' \
  'http://localhost:3100/api/auth/login' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "username": "admin@admin.com",
    "password": "$Ab123456"
  }'
```

Request URL

<http://localhost:3100/api/auth/login>

Server response

Code	Details
------	---------

201

Undocumented

Response body

[illegible]

Response headers

```
access-control-allow-origin: *
connection: keep-alive
content-length: 281
content-type: application/json; charset=utf-8
date: Thu, 14 May 2026 03:40:32 GMT
etag: W/"119-8obxh1ZATO8Ufn2QmsE+9CHI0B1"
keep-alive: timeout=5
x-powered-by: Express
```

Responses

God

Description

Links

200

Token JWT generado correctamente.

No links

Media type

application/json

Controls Accept header.

Example Value | Schema

```
{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXLTJ5In0:"
}
```

401

Credenciales incorrectas

No links

Fuente: autor

02. Egresados

GET /api/graduates/template Descargar plantilla CSV

Retorna un archivo CSV vacío con las cabeceras esperadas para importación. Requiere SUPER-USER o ADMIN.

Parameters

No parameters

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:3100/api/graduates/template' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZiZmZjZjQWZkt1ZmZ5IjoIbmhMTQWQW3NDVlYiwiYm99IjoIQU9NSU41LCJ0aWZpdmU0OnRydWUsInVzZXJ0VWllIjoIYmhtak5A'
```

Request URL

http://localhost:3100/api/graduates/template

Server response

Code	Details
200	Response body Download file Response headers <pre>access-control-allow-origin: * connection: keep-alive content-disposition: attachment; filename="plantilla_egresados.csv" content-length: 383 content-type: text/csv; charset=utf-8 date: Thu, 14 May 2026 03:45:16 GMT etag: W/"17f-5TC9ueFP5lg3TYkq50tu9KwV0U" keep-alive: timeout=5 x-powered-by: Express</pre>

Responses

Code	Description	Links
200	Archivo CSV de plantilla.	No links

03. Noticias

POST /api/news Crear noticia

GET /api/news Listar todas las noticias (panel)

Requiere SUPER-USER o ADMIN.

Parameters

Name	Description
limit number (query)	Cantidad de registros por página (mínimo 1) 10
page number (query)	Número de página (mínimo 1) 1
param string (query)	Texto de búsqueda o filtro Juan

ExecuteClear

Cancel

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:3100/api/news?limit=10&page=1&param=Juan' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZiI6IjZyIjo1ZmFIZjQwMzktMjE5OS00ZTg0LWI3YjI1MmhhWTQwNGJlNDVlITlwIiwiaWF0IjoiQURNSU41LlC3Y3Rpdml1OnRydmlsInVzZXJ0VW11ITJo1VWItam54
```

Request URL

http://localhost:3100/api/news?limit=10&page=1¶m=Juan

Server response

Code	Details
200	Response body <pre>{ "data": [], "total": 0, "page": 1, "limit": 10, "totalPages": 0 }</pre> Download Response headers <pre>access-control-allow-origin: * connection: keep-alive content-length: 66 content-type: application/json; charset=utf-8 date: Thu, 14 May 2026 03:44:55 GMT etag: W/"38-yvovrPqeqhyefvYzKc5rAGqB" keep-alive: timeout=5 x-powered-by: Express</pre>

Responses

Code	Description	Links
200	Lista paginada de noticias.	No links

Figura 16.

05. Carrusel

GET

/api/carousel

Obtener imágenes del carrusel

Reloma todos los slots del carrusel. No requiere autenticación.

Parameters

No parameters

Cancel

Execute

Clear

Responses

Curl

curl -X 'GET' \n'http://localhost:3100/api/carousel' \n-H 'accept: */*\n'

Request URL

http://localhost:3100/api/carousel

Server response

Code

Details

200

Response body

{\n "data": [\n {\n "id": "4d0c8538-852b-41d4-8b16-96f3aada2834",\n "url_img": "https://www.santototunja.edu.co/images/01-USTATunja/10-USTA-Tunja-DepAdministrativos/AdmisionesVMercadeo/2022/Oferta_Posgrados_Santoto_Tunja_2022.png",\n "link": "https://www.santototunja.edu.co/programas-academicos/programas/posgrados-presenciales"\n },\n {\n "id": "15cd2bf8-5a57-4d43-bacb-dac62d226d40",\n "url_img": "https://graduados.usta.edu.co/images/Bolsadeempleo-100.jpg",\n "link": "https://graduados.usta.edu.co/index.php/bolsa-de-empleo/que-es-la-bolsa-de-empleo"\n },\n {\n "id": "5060f071-65af-42fb-bcdd-808ae075e628",\n "url_img": "https://www.santototunja.edu.co/images/01-USTATunja/01-USTA-Tunja-Imagenes/Departamentos_Unidades/2019/apoyo-a-egresados.png",\n "link": "https://www.santototunja.edu.co/inicio-direccion-de-graduados"\n },\n {\n "id": "4c3adecc-2774-4d95-b70d-59b0cf8c0a44",\n "url_img": "https://www.santototunja.edu.co/images/01-USTATunja/10-USTA-Tunja-DepAdministrativos/AdmisionesVMercadeo/2022/Oferta_Posgrados_Santoto_Tunja_2022.png",\n "link": "https://www.santototunja.edu.co/programas-academicos/programas/posgrados-presenciales"\n },\n {\n "id": "f1a2ddb1-eb58-4a35-be64-4bd0b35c0cdb",\n "url_img": "https://www.santototunja.edu.co/images/01-USTATunja/01-USTA-Tunja-Imagenes/Departamentos_Unidades/2019/apoyo-a-egresados.png",\n "link": "https://www.santototunja.edu.co/inicio-direccion-de-graduados"\n }\n]\n}

Download

Response headers

access-control-allow-origin: *\ncontent-length: 1332\ncontent-type: application/json; charset=utf-8\ndate: Thu, 14 May 2026 03:43:28 GMT\netag: W/"334-R2Xc0M6Mh28x11GhRuZ1PHg"\nx-powered-by: Express

Responses

Code

Description

Links

200

Lista de imágenes del carrusel.

No links

Fuente: autor

Figura 17.

Fuente: autor

Figura 18.

Figura 17.

GET

/api/graphics/job-status-counts Distribución por estado de empleo

Conteo de egresados por estado: empleado, desempleado, independiente. Requiere SUPER-USER o ADMIN.

Parameters

Cancel

Name	Description
facultyId	Filtrar por facultad
string (query)	1

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
'http://localhost:3100/api/graphics/job-status-counts?facultyId=1' \
-H 'accept: */*' \
-H 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZmVzc2VybmFtZSwiaWF0IjoiYWRtaWUiLCJhdw%3D'`
```

Request URL

```
http://localhost:3100/api/graphics/job-status-counts?facultyId=1
```

Server response

Code	Details
200	Response body<pre>{ "result": { "sin_trabajo": 0, "con_trabajo": 1 } }</pre> Download Response headers<pre>access-control-allow-origin: * connection: keep-alive content-length: 44 content-type: application/json; charset=utf-8 date: Thu, 14 May 2020 03:48:51 GMT etag: W/"2c-RAGM8A4QW@q3dHrVOWYN0B9HK" keep-alive: timeout=5 x-powered-by: Express</pre>

Responses

Code	Description	Links
200	Conteo por estado de empleo.	No links

7.3. Fase 4 Puesta en producción

Para la dockerización se optó por Docker Compose el cual utiliza el patron oficial recomendado por Docker compose en el cual se define una topología de servicios (db , backend , frontend , images , pgadmin) junto a los puertos definidos y dependencias y volúmenes persistentes y un bootstrapping de Db para inicializar la base de datos correctamente para luego las acciones de TypeOrm junto con las acciones que sobre escriben entornos dándonos una configuración de producción y una de desarrollo variando entre ellas el arranque permitiendo a la versión de desarrollo (hot-reload) útil para el desarrollo utilizando 'docker compose -f docker-compose.yml -f docker-compose.dev.yml up'

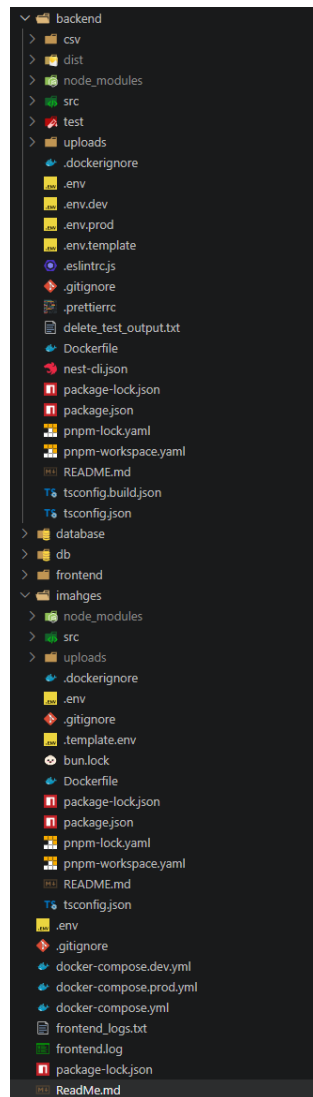
dándonos ventajas de cumplimientos con DRY junto con reproducibilidad de entornos junto con un Onboarding local de un solo comando muy útil para el entorno de trabajo debido a los varios servicios necesitados.

Junto con un servicio de imágenes separado, dado que este podría ser reemplazado a largo plazo por servicios como CloudFront sin tocar el backend, esto se debe a que se encuentra separado del backend, ofreciendo un enfoque de microservicio independiente.

Como último, un servicio de pgAdmin, el cual se despliega después de la db, específico para desarrollo, el cual permite revisar y manipular la db o sus datos si se requiere.

Asimismo, se generaron Dockerfiles multiestágio en los que el runtime final solo copia el dist, node_modules y package.json desde la etapa de build, descartando lo demás y produciendo una imagen final más pequeña.

Figura 12. Ordenamiento carpetas para Docker



Fuente: Autor

Figura 13. Ordenamiento Contenedores Docker

☐	▼	●	egresados	-	-	-	☐	⋮
☐		●	frontend	b2310d4cc88b	egresados-	4300:4300	☐	⋮
☐		●	images	c160618c1bf2	egresados-	8089:8089	☐	⋮
☐		●	backend	71d4a6b30ef7	egresados-	3100:3100	☐	⋮
☐		●	pgadmin	319cd0908054	dpage/pgadmin	8080:80	☐	⋮
☐		●	postgres	374b38ecb2d7	postgres:1	5432:5432	☐	⋮

Fuente: Autor

7.3.2. Fase 4.2. Pruebas

Describir las pruebas aplicadas y el resultado obtenido.

Se realizaron pruebas unitarias con el fin de dar coberturas por módulo dando así que en el primero **ErrorService:**

Se revisó en caso de entrada duplicada en documento o user name ya registrado luego violación de llave foránea y violación de check constraint luego en el próximo módulo se revisó que facultad esperando mapeen correctamente al código HTTP y mensaje correspondiente

FacultyService:

Verifica que el crud esté completo de facultades

AuthService

Verifica la autenticación y búsqueda y gestión de jwt

GraduatesService

Verifica el ciclo de vida completo de un egresado, con énfasis en el orden de eliminación para evitar violaciones de FK

Figura 14. Ordenamiento carpetas para Docker

```
Test Suites: 4 passed, 4 total
Tests:      35 passed, 35 total
Snapshots:  0 total
Time:       5.508 s, estimated 7 s
Ran all test suites.
```

Fuente: Autor

De esta manera se logro verificar el correcto funcionamiento de los puntos y las logicas encontradas dentro de lo anterior menten nombrado juntando 35 pruebas junto con la documnetacion en swagger se puede afirmar la completitud el sistema de la refactorizacion de codigo logrando demosrtar las habilidades en el desarrollo y la refactorizacion correcta de un sistema de esta manera asi mismo se teniea encuentra la manera correcta en la cual se deseaba loagrar llegar a resolver los probleams y las nesesidades del software.

Describir el montaje de la aplicación (esta sección es opcional y depende del alcance del proyecto)

El despliegue a producción se dio en un servido en la universidad mas específicamente en uno montado en la red oculta de la universidad Santo tomas Seccional tunja en el edificio de Santo Domingo en una red llamada ESTUDIO_TV en el cual fue montado en la ip 172.21.1.80 desde allí se gestiono el despliegue a producción dando le acceso a un maquina la cual poseía un Linux mint se clono el repositorio y se realizo el despliegue gracias a Docker y al Docker compose montado desde allí se realizaron pruebas para verificar su funcionamiento. Así mismo para completar con un despliegue continuo se generó un github Actions selfhosted para que de esta manera desde fuera de la universidad y por medio del repositorio se pueda poner en producción :

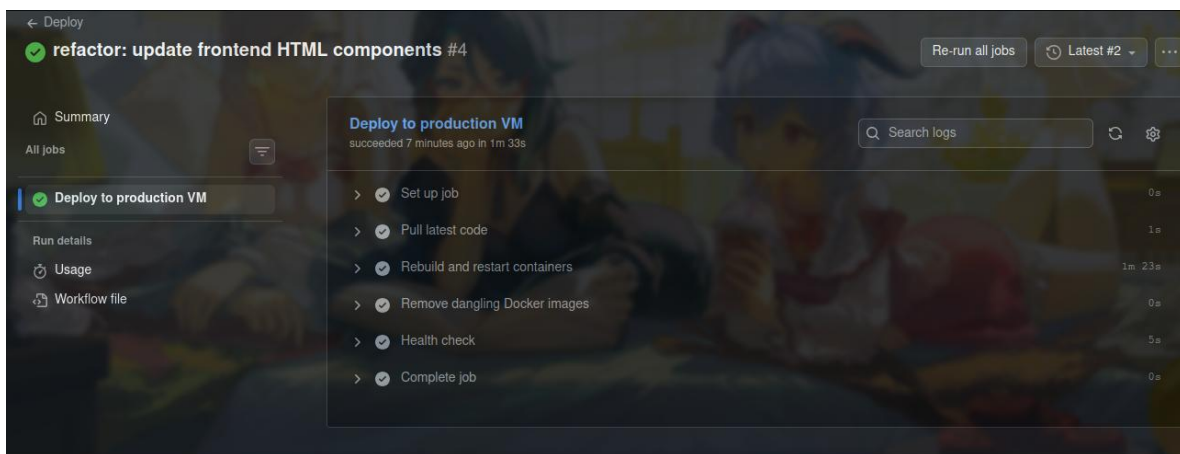
Figura

```
[+] Running 8/8
✓ egresados-frontend Built 0.0s
✓ egresados-backend Built 0.0s
✓ egresados-images Built 0.0s
✓ Container postgres Running 0.0s
✓ Container pgadmin Running 0.0s
✓ Container backend Started 11.4s
✓ Container images Started 11.4s
✓ Container frontend Started 1.3s
```

Fuente: Autor

De esta manera se genero el CICD con el fin de lograr verificar el estado de la puesta en produccion del software dando como resultado un estado donde vemos que la maquina viryal proporcuonada por la universidad realica el pull request junto con reconstruir los container nesesarios para el funcionamiento evitando tocar los contenedores donde se encuentra la infromacion db y los datos almacenados en el servidor de imagenes luego remover la informacion de los docker image realizzando un limpiado seguro y verificando el estado de los docker y con ello pudiendo verificar el estado de la puetsa en produccion luego de esto se agregron noticias proporcionadas por el ingeniero diego vela para comensar a llenar los datos nesesarios

Figura 18.



Fuente: Autor

Podemos ver cómo fue funcionando a través de las actualizaciones de cada uno de los cambios realizados para la mejora del sistema

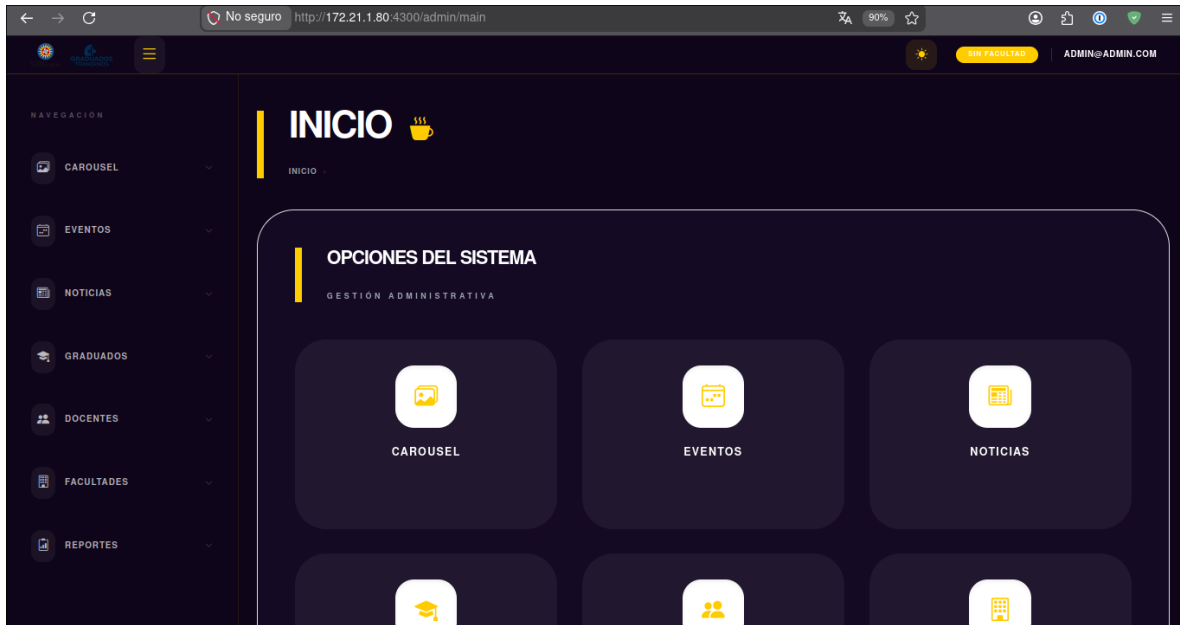
Figura19.



Fuente: Autor

Se revisó el aplicativo ya montado en el servidor demostrando estar funcional

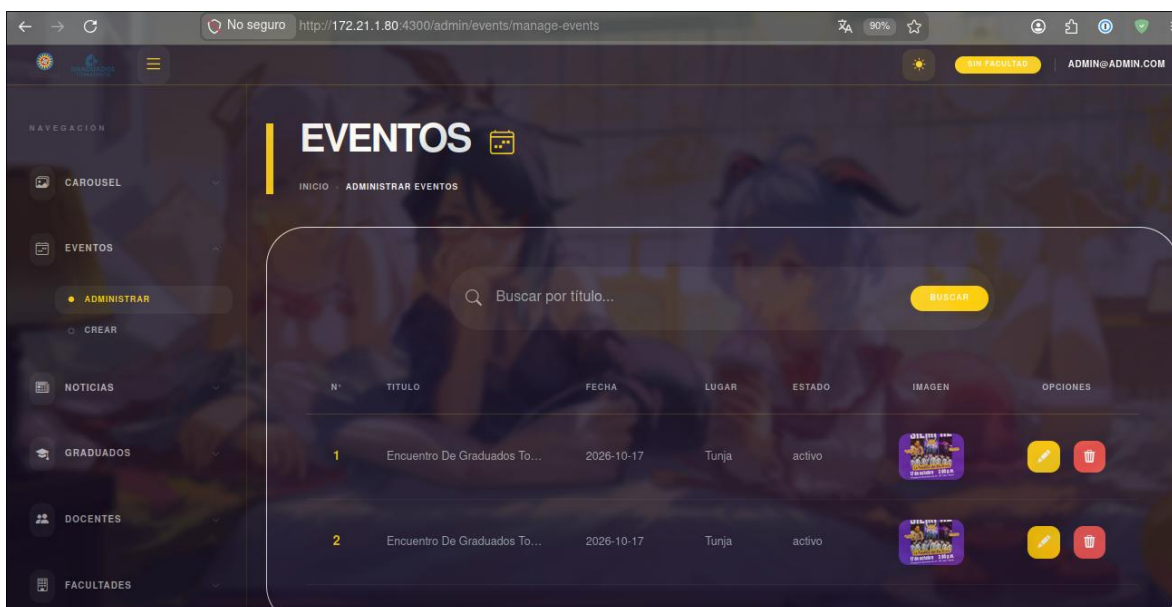
Figura 20.



Fuente: Autor

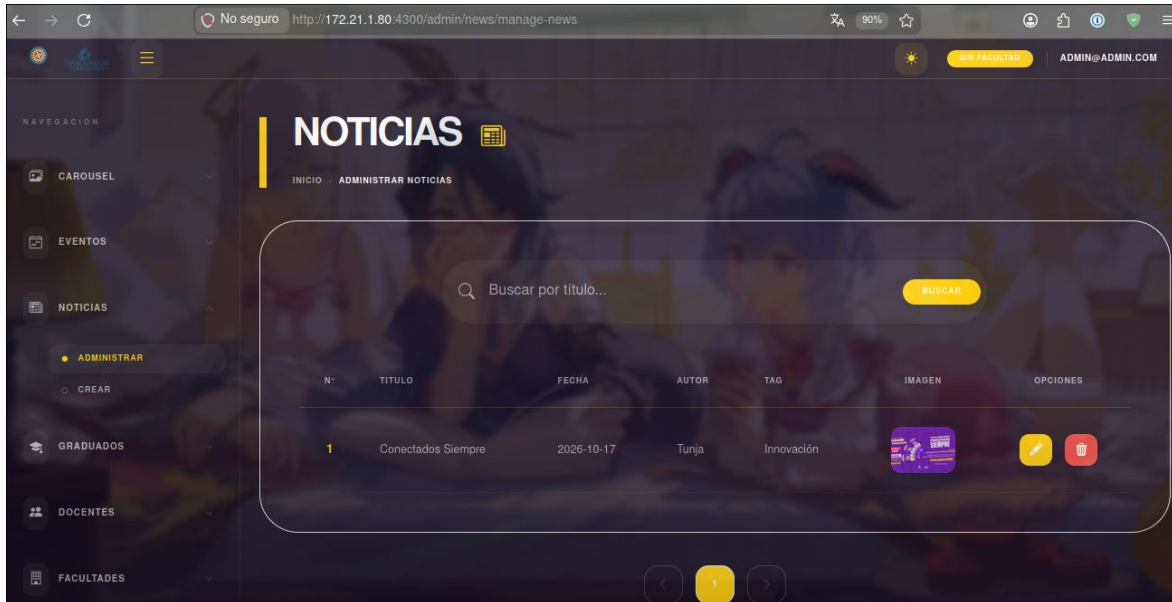
Y junto con la ingesta de los datos aplicados en noticias y eventos de un evento próximo a petición del ingeniero Diego Alejandro Vela Beltrá, demostrando el correcto funcionamiento del servidor de imágenes y de los cambios gráficos realizados.

Figura 21.



Fuente: Autor

Figura 22.



Asimismo se verificaron las demás interfaces donde se muestra el correcto funcionamiento del programa

8. CONCLUSIONES Y ANÁLISIS DE RESULTADOS

Se logró rediseñar e implementar una reforma estructural del sistema de seguimiento a graduados, que originalmente estaba circunscrito a la Facultad de Ingeniería de Sistemas, y escalarla a nivel institucional, de carácter multifacultad, en la Universidad Santo Tomás Seccional Tunja, consolidando una herramienta que centraliza la gestión de graduados y contribuye al aseguramiento de la calidad académica.

Se realizó de manera adecuada la identificación de los requerimientos funcionales y no funcionales, dando paso a determinar correctamente el alcance multifacultad del sistema preservando todo el stack de Angular, NestJS y PostgreSQL

Dada la automatización de los procesos de integración y despliegue continuo mediante Docker y GitHub Actions, y la configuración de un runner autohostado en la infraestructura institucional, se permitió desplegar el sistema en los servidores sin exponer puertos ni modificar el firewall de la red, lo que resultó de gran ayuda . Junto con una orquestación de Docker Compose que garantice la reproducibilidad entre desarrollo y producción.

En conjunto, el proyecto muestra la evolución del sistema hacia plataformas intocables mediante una estrategia de refactorización basada en el análisis de la estructura anterior y de los requerimientos de la universidad . La solución dada no solo resuelve las limitaciones de la versión anterior, sino que también deja sentadas las bases técnicas para incorporar progresivamente mejoras a disposición de la universidad.

9. REFERENCIAS

1. A. Olowolayemo, K. Harun and T. Mantoro, "University Based Job Recommender & Alumni System," 2018 International Conference on Computing, Engineering, and Design (ICCED), Bangkok, Thailand, 2018, pp. 212-217, doi: 10.1109/ICCED.2018.00049.
2. J. C. Escobar Aplicativo web para la gestión de seguimiento a graduados de la Facultad de Ingeniería de Sistemas de la Universidad Santo Tomás Seccional Tunja. [online]. Disponible en: [referencia](#)
3. Ministerio de Educación Nacional. (2019). Decreto 1330 de 2019. Artículo 2.5.3.2.3.1.5. Programa de graduados. MEN. <https://www.suin-juriscol.gov.co/viewDocument.asp?ruta=Decretos/30019930>
4. A. Mukherjee, A. Roy, M. K. Lath, A. Ghosal, and D. Sengupta, "Centralized Alumni Management System (CAMS) - A Prototype Proposal," 2019 Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, 2019, pp. 967-971, doi: 10.1109/AICAI.2019.8701383.
5. Martin, R. C. (2008). Clean Code: A Handbook of Agile Software Craftsmanship. Disponible en : [referencia](#)
6. A. Rista, L. Toti and E. Xhaferri, "Design and Implementation of an Alumni Management System," 2023 4th International Conference on Communications, Information, Electronic and Energy Systems (CIEES), Plovdiv, Bulgaria, 2023, pp. 1-4, doi: 10.1109/CIEES58940.2023.10378784.
7. Inc, D. (2026, 25 marzo). Why use Compose? Docker Documentation. <https://docs.docker.com/compose/intro/features-uses/>
8. Palomba, F., Bavota, G., Penta, M. et al. On the diffuseness and the impact on maintainability of code smells: a large scale empirical investigation. Empir Software Eng 23, 1188–1221 (2018). <https://doi.org/10.1007/s10664-017-9535-z>
9. Consejo Nacional de Acreditación. (2020). Acuerdo CESU 02 de 2020. Modelo de Acreditación en Alta Calidad. CNA. https://www.cna.gov.co/1779/articles-402848_documento.pdf
10. Bass, L., Clements, P., & Kazman, R. (2013). Software Architecture in Practice. disponible en : [referencia](#)

11. G. Shetty and T. Sharma, "Mapping Code Smells and Refactorings Accurately: Insights from an Empirical Study," 2025 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), Honolulu, HI, USA, 2025, pp. 1-11, doi: 10.1109/ESEM64174.2025.00044.
12. E. Zabardast, J. Gonzalez-Huerta and D. Šmite, "Refactoring, Bug Fixing, and New Development Effect on Technical Debt: An Industrial Case Study," 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Portoroz, Slovenia, 2020, pp. 376-384, doi: 10.1109/SEAA51224.2020.00068.
13. Y. Kataoka, T. Imai, H. Andou and T. Fukaya, "A quantitative evaluation of maintainability enhancement by refactoring," International Conference on Software Maintenance, 2002. Proceedings., Montreal, QC, Canada, 2002, pp. 576-585, doi: 10.1109/ICSM.2002.1167822.
14. S. A. M. Rizvi and Z. Khanam, "A methodology for refactoring legacy code," 2011 3rd International Conference on Electronics Computer Technology, Kanyakumari, India, 2011, pp. 198-200, doi: 10.1109/ICECTECH.2011.5942080.
15. Fowler, M. (2002). Patterns of Enterprise Application Architecture. Addison-Wesley. Disponible en: [referencia](#)
16. Fowler, M. (2018). Refactoring: Improving the Design of Existing Code (2.^a ed.). Addison-Wesley Professional. Bass, L., Clements, P., & Kazman, R. (2021). Software Architecture in Practice (4.^a ed.). Addison-Wesley Professional. disponible en: [referencia](#)
17. E. Trichkova-Kashamova, "Applying the ISO/IEC 25010 Quality Models to an Assessment Approach for Information Systems," 2021 12th National Conference with International Participation (ELECTRONICA), Sofia, Bulgaria, 2021, pp. 1-4, doi: 10.1109/ELECTRONICA52725.2021.9513662.
18. A. Nandi, P. Ghosh, S. Mandal, A. Ghosh, A. K. Das and S. Dutta, "Hybrid Multi-Entity Reinforcement Learning Approach for Intelligent, Scalable Career Path Recommendations in Alumni Networks," 2025 13th International Conference on Intelligent Embedded, MicroElectronics, Communication and Optical Networks (IEMECON), Jaipur, India, 2025, pp. 1-6, doi: 10.1109/IEMECON69302.2025.11365848.

10. ANEXOS

Anexo 1. Ficha Técnica

1. Servidor

Componente	Minimo	Recomendado
Prosesador	2 vCPU / núcleos x64	4 vCPU / núcleos x64
Memoria RAM	4 GB	8 GB
Sistema operativo	Linux x64 (Ubuntu 22.04 LTS)	Linux x64 (Ubuntu 22.04 LTS)
Red	Conexión saliente a GitHub	Igual + IP fija o DNS interno
Software base	Docker Engine 24+, Docker Compose v2, Git	Igual + monitoreo de recursos

2. Estación de desarrollo

Componente	Minimo	Recomendado
Prosesador	4 núcleos x64	6-8 núcleos x64
Memoria RAM	8 GB	16 GB
Sistema operativo	Windows 10/11, macOS o Linux x64	Igual
Red	Conexión saliente a GitHub	Igual + IP fija o DNS interno
Software base	Docker Desktop, Node 20, pnpm 10.17, Git	Igual + editor (VS Code)

Anexo 2. Manual de Usuario

Debido a la posible necesidad de uso este documento también lo encuentra en :

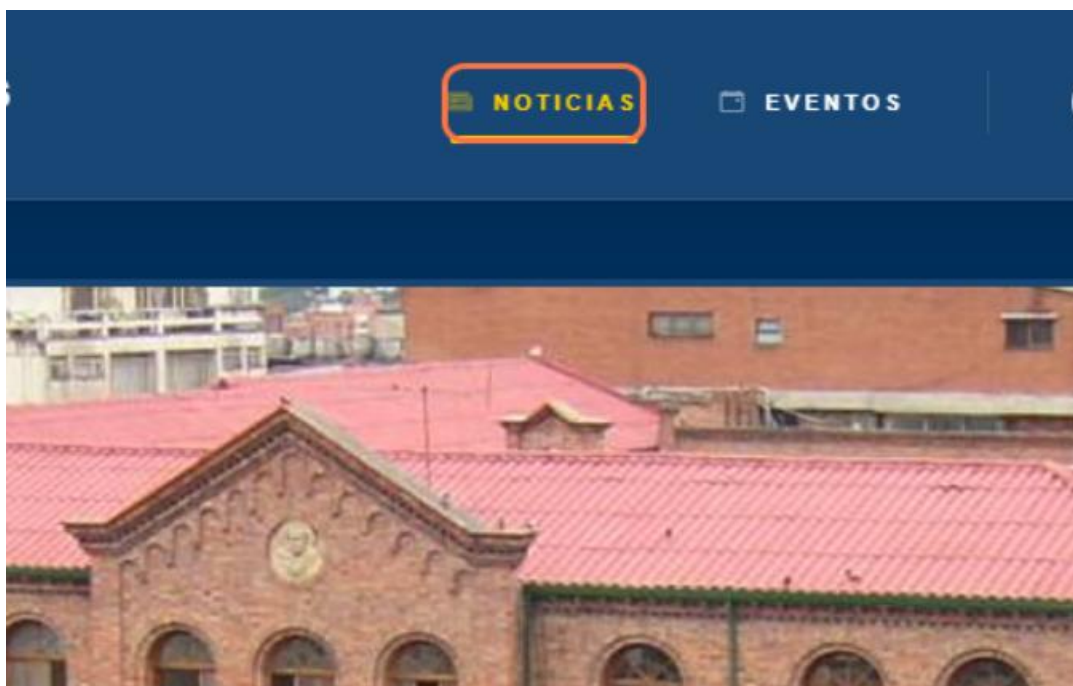
<https://drive.google.com/file/d/1Dwt4NyqmL0oxTCtZy1M7ubpDwWGQjtED/view?usp=sharing>

1. Acceso a la página

La web se encuentra en la ip en el servidor de institucional ESTUDIO_TV en:
172.21.1.80:4300

2.1 Pantalla principal

En la pantalla principal, el botón de Noticias lleva al apartado de noticias publicadas en la plataforma.

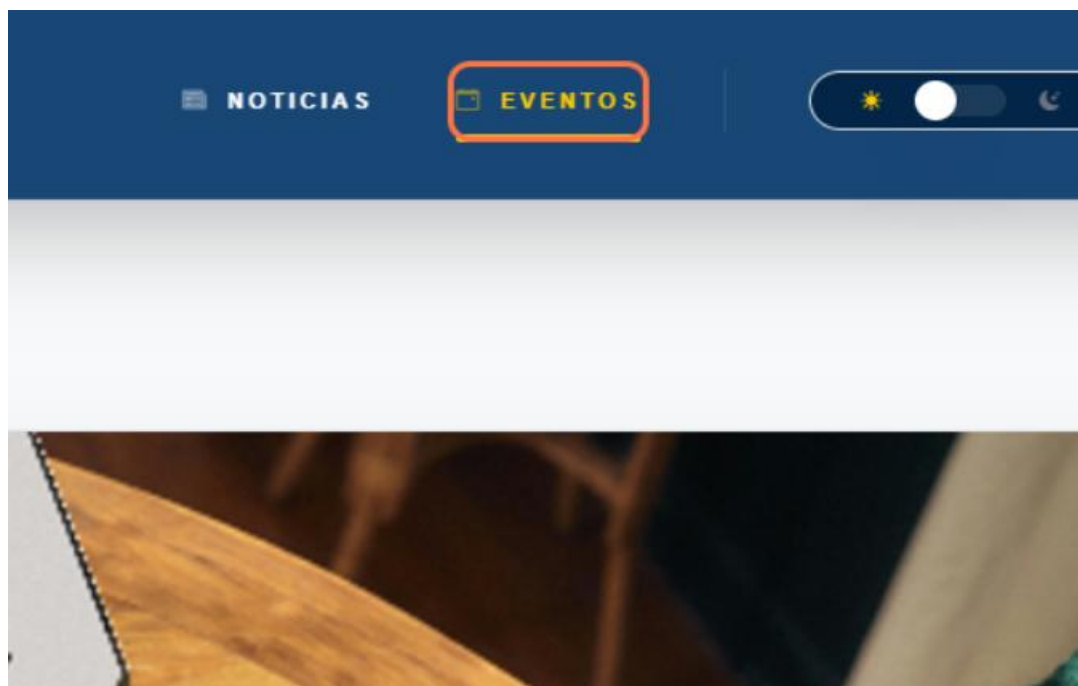


A continuación se muestran, en la sección de noticias , los artículos disponibles; además, al hacer clic en el botón **“Leer artículo completo”**, se mostrarán todos.

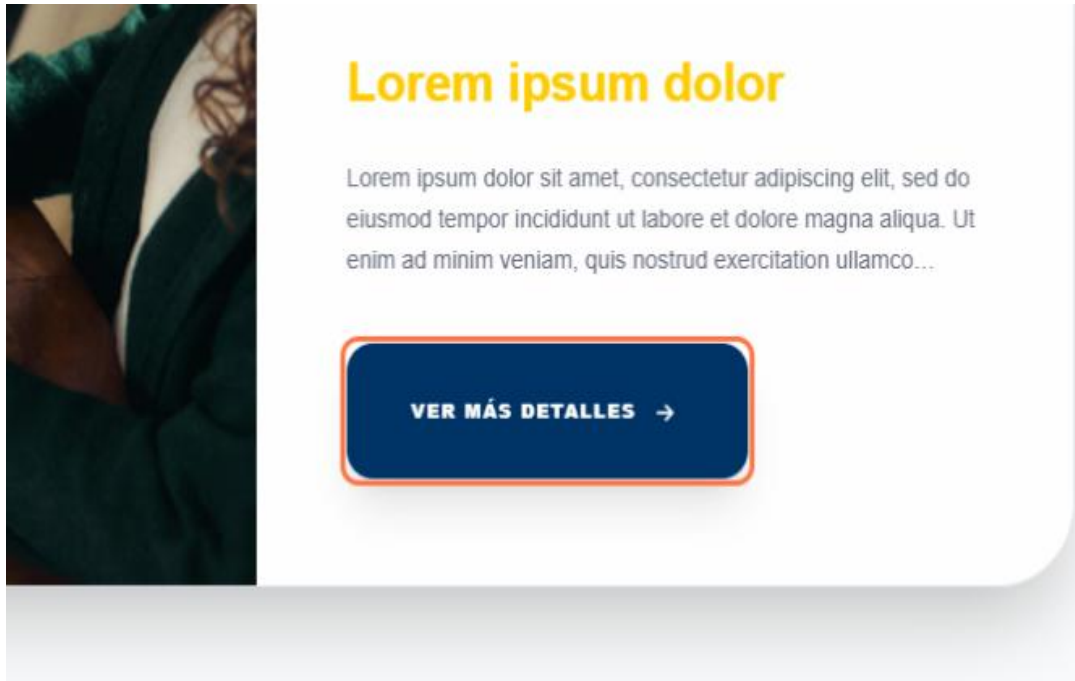


En esta misma página, se encuentra el botón de “**Evento**”.

Eventos: los eventos que has registrado en la plataforma.

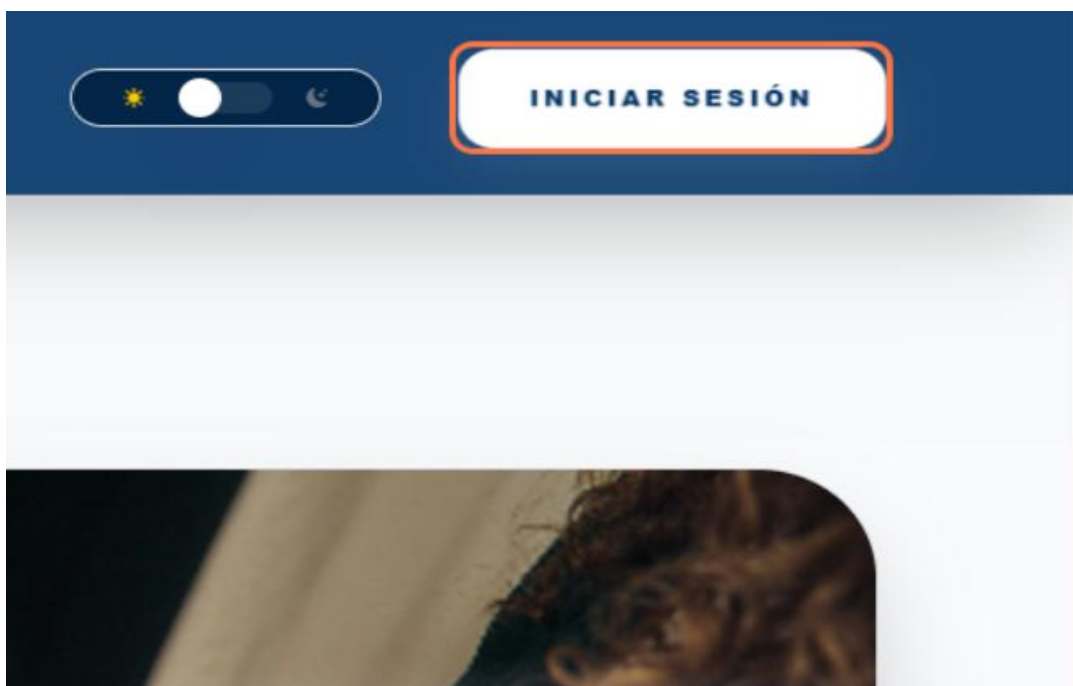


Para ver los detalles de un evento, haga clic en “Ver más detalles”. Esto mostrará la información completa de cada evento.



2.2 Inicio de sesión

Para iniciar sesión, haga clic en el botón de inicio de sesión en la esquina superior derecha de la pantalla principal.



Ingrese su nombre de usuario, que consiste en el primer nombre + el carácter @ + el primer apellido (en minúsculas y con la inicial del segundo apellido). El usuario puede consultarlo en Detalle de egresado o en Editar docente, según el tipo de cuenta.



Ingrese su contraseña. Se genera automáticamente en este formato: \$Ab + número de documento y puede modificarse posteriormente en la edición de usuario.



UNIVERSIDAD SANTO TOMÁS

USUARIO

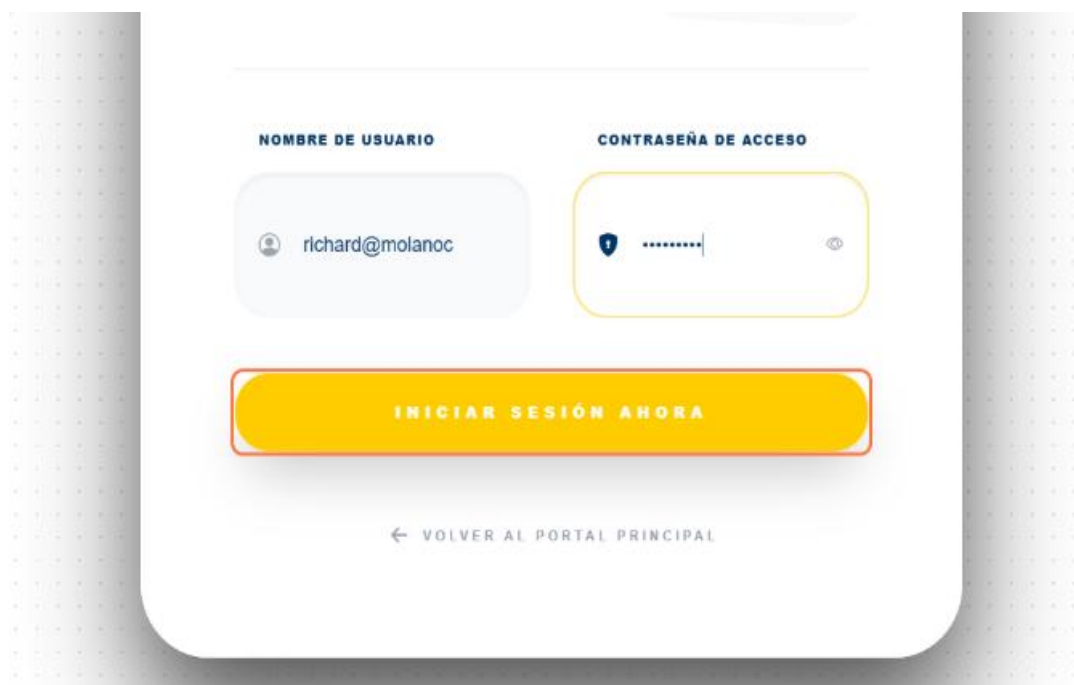
CONTRASEÑA DE ACCESO

d@molanoc

INICIAR SESIÓN AHORA

The image shows a login form for Universidad Santo Tomás. It has two input fields: 'USUARIO' (Username) and 'CONTRASEÑA DE ACCESO' (Access Password). The username field contains 'd@molanoc'. The password field is highlighted with a red and yellow border and contains a masked password '.....' with a toggle icon. Below the fields is a blue button labeled 'INICIAR SESIÓN AHORA'.

Haga clic en el botón Iniciar sesión para acceder al panel administrativo.



NOMBRE DE USUARIO

CONTRASEÑA DE ACCESO

richard@molanoc

INICIAR SESIÓN AHORA

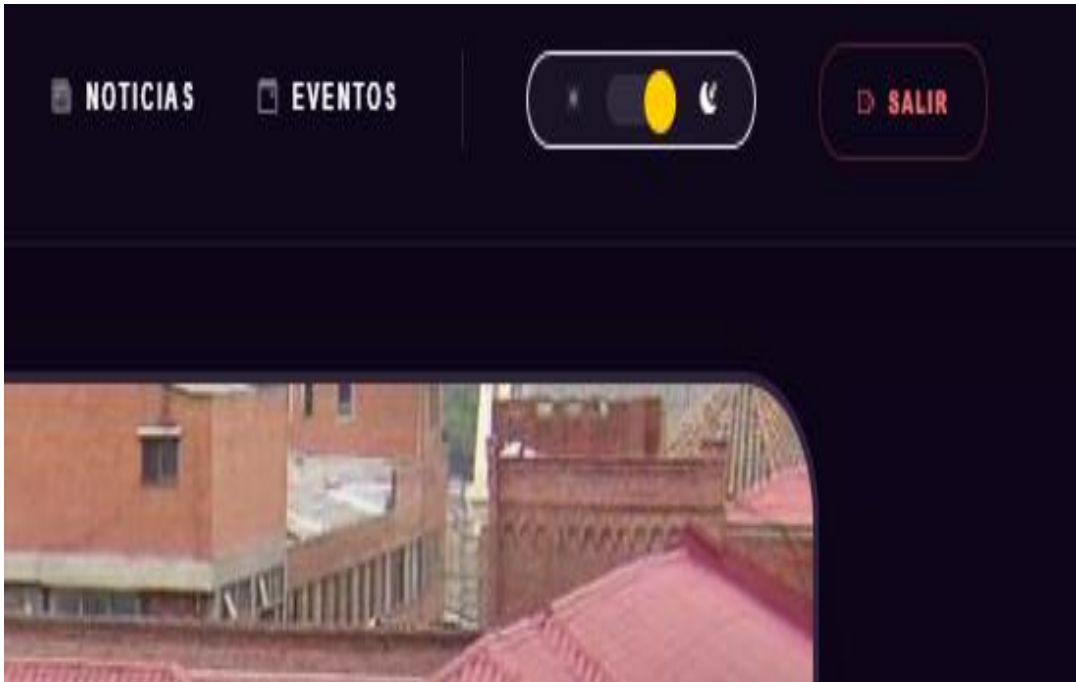
← VOLVER AL PORTAL PRINCIPAL

The image shows a login form for Universidad Santo Tomás. It has two input fields: 'NOMBRE DE USUARIO' (Username) and 'CONTRASEÑA DE ACCESO' (Access Password). The username field contains 'richard@molanoc'. The password field is highlighted with a yellow border and contains a masked password '.....' with a toggle icon. Below the fields is a yellow button labeled 'INICIAR SESIÓN AHORA'. At the bottom, there is a link labeled '← VOLVER AL PORTAL PRINCIPAL'.

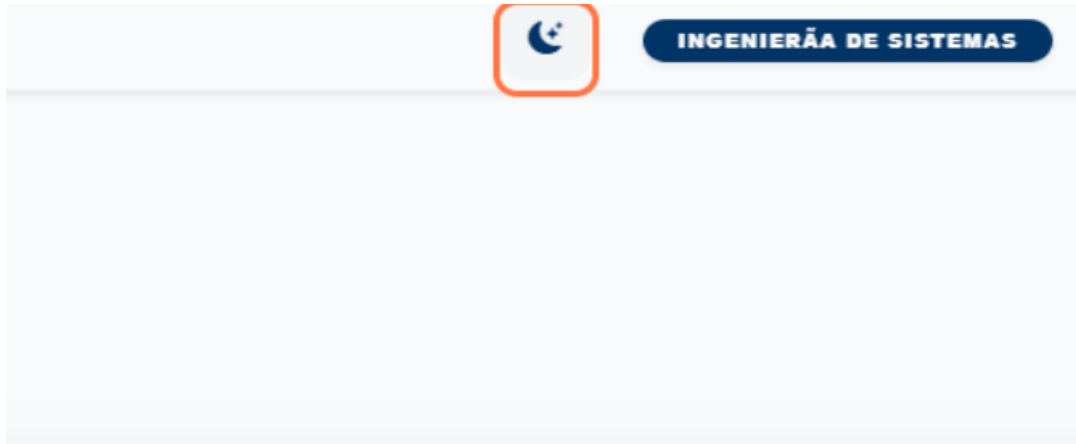
3. Panel administrativo

3.1 Cambio de tema (modo claro / oscuro)

Puedes alternar el estilo entre claro y oscuro, usando el botón ubicado en la parte superior de la vista del panel.



Al seleccionar un tema, la interfaz lo aplica de inmediato.



3.2 Información del usuario y filtrado por facultad

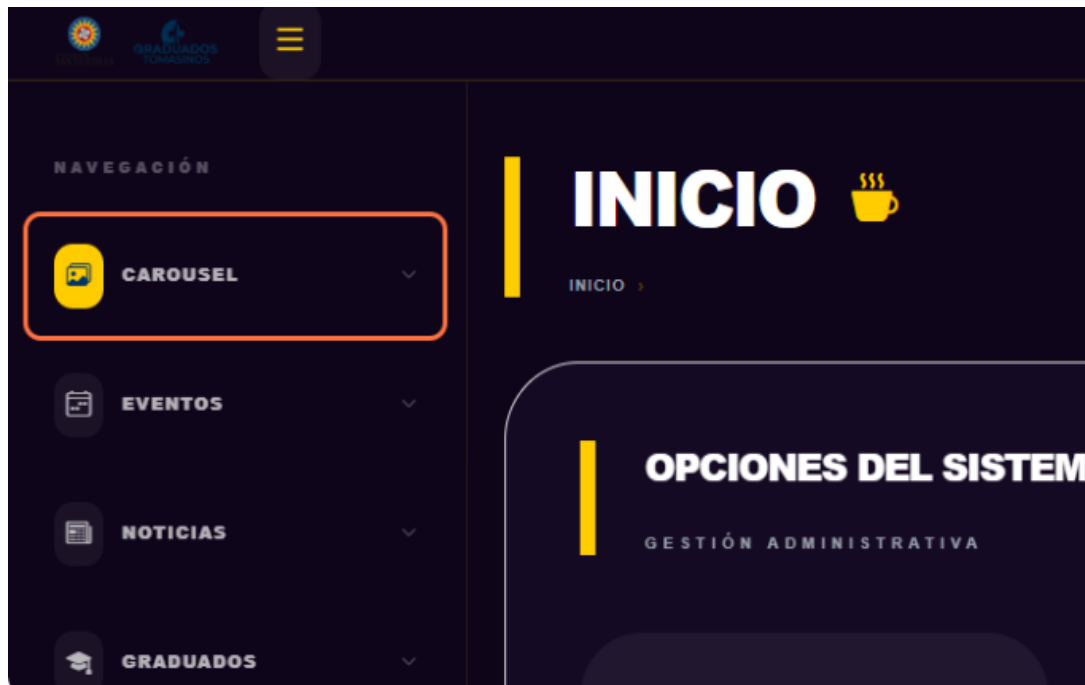
El nombre del usuario y la facultad a la que pertenece se muestran en la parte superior de la pantalla. La facultad filtra automáticamente toda la información a la que el usuario tiene acceso.



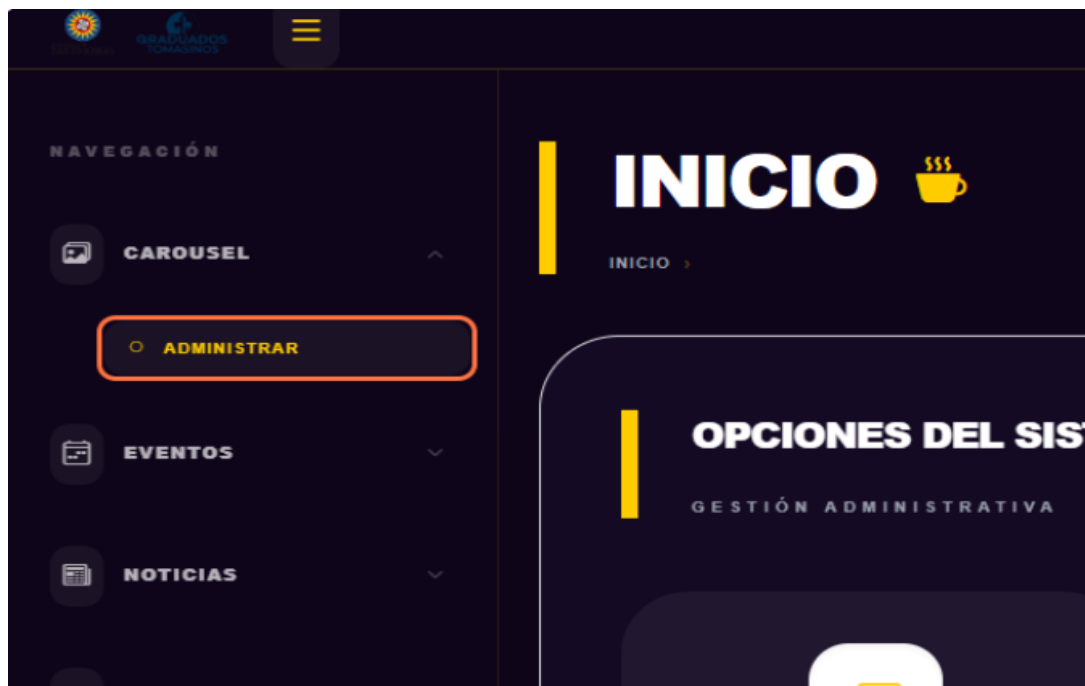
4. Gestión del carrusel

El **Carrusel** administra las imágenes que se muestran debajo de su pantalla de inicio en el sitio público.

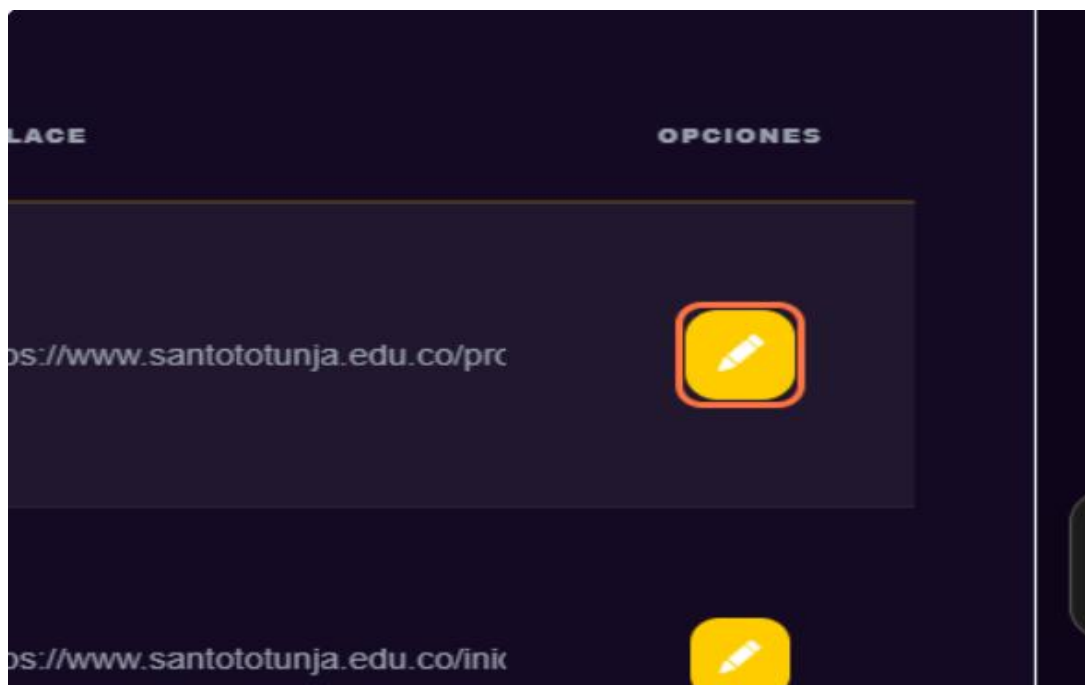
Ve al menú lateral y selecciona el módulo Carrusel.



Selecciona **“Administrar”** para usar la función de administración de imágenes.

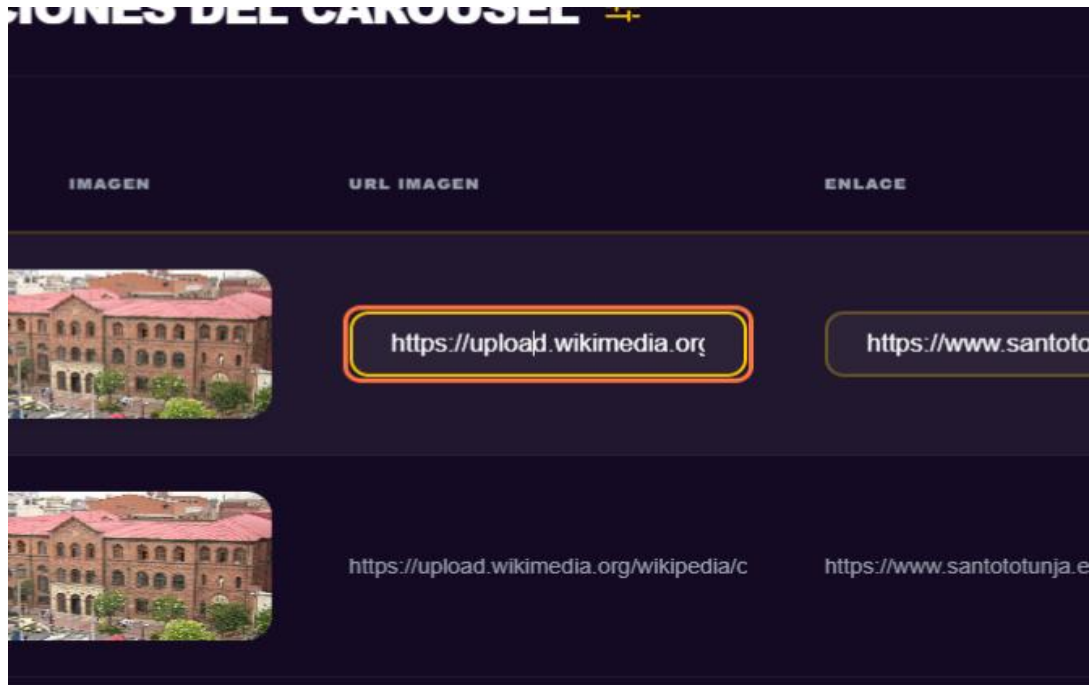


En esta vista, haga clic en el ícono del lápiz para editar la información de cada imagen del carrusel: la imagen y el enlace al que redirige.



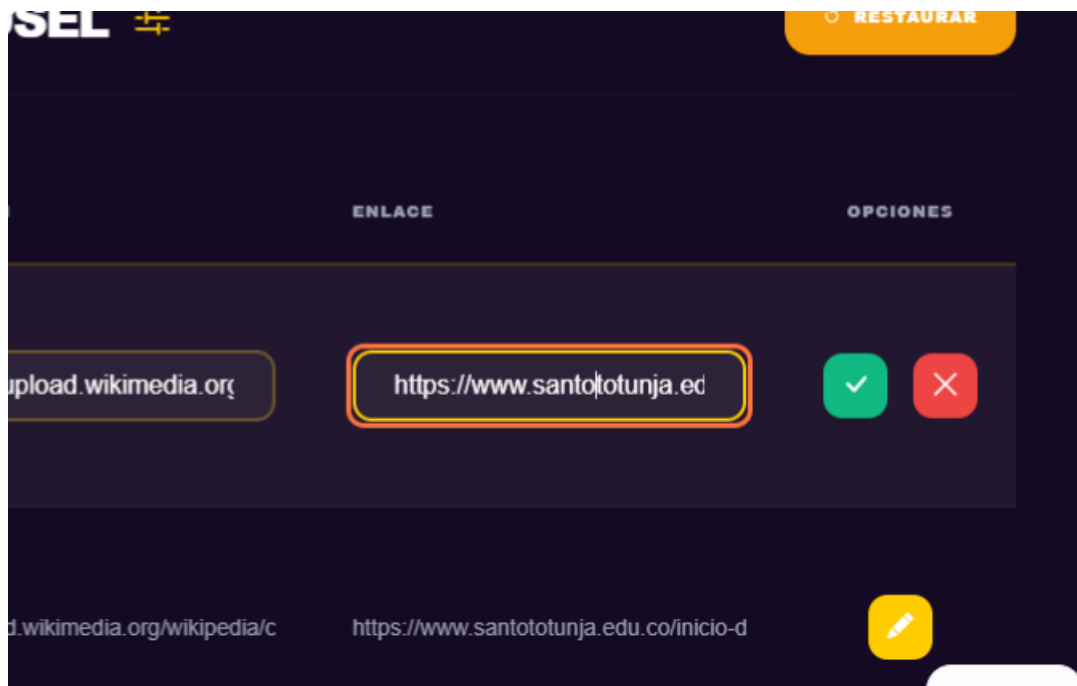
4.1 Editar URL de la imagen

Ahora, para cambiar la imagen, solo necesitas editar el campo del archivo de imagen.



4.2 Editar hipervínculo

Para modificar el enlace de destino, edita el campo del hipervínculo.



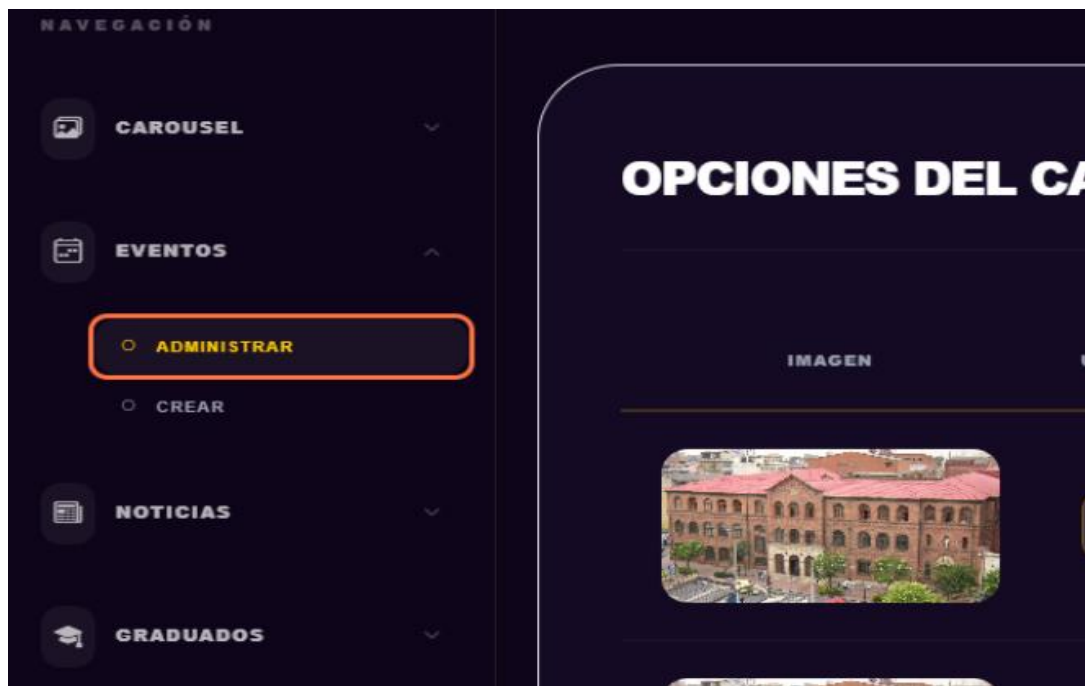
5. Gestión de eventos

Desde el menú lateral, accede al módulo de **Eventos**.



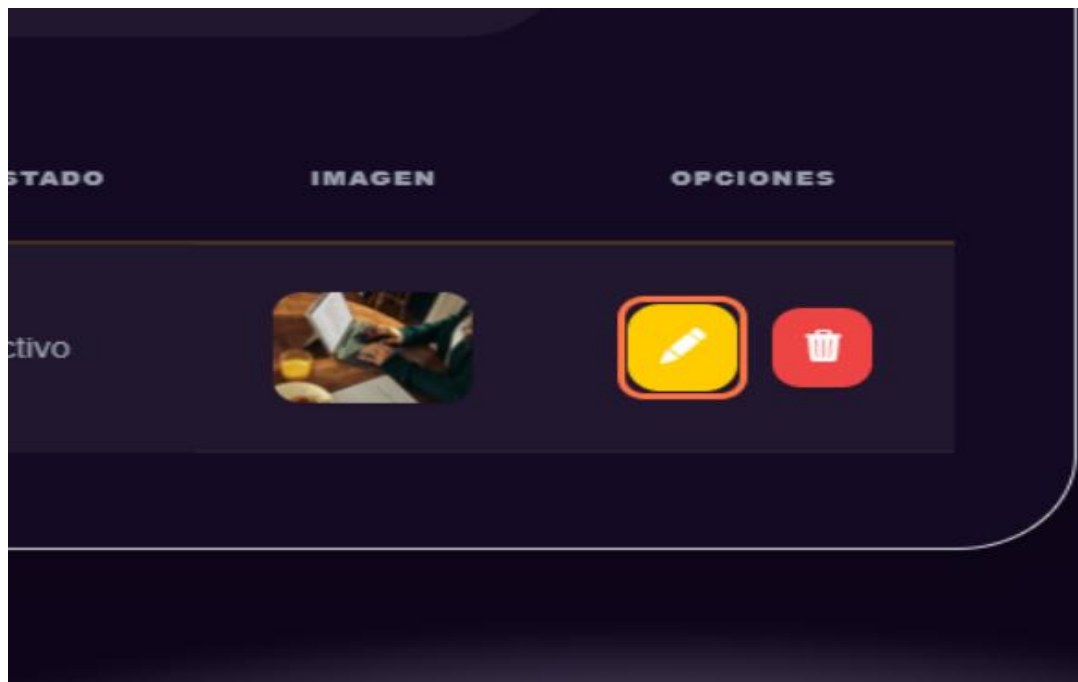
5.1 Administrar eventos

Elija la opción «Administrar»→ Aquí puede editar o eliminar los eventos registrados.



5.2 Editar un evento

Haga clic en el botón con el ícono de lápiz para editar el evento seleccionado.



Realiza las ediciones que necesites en el formulario de edición.

ADMINISTRAR EVENTOS

INICIO · ADMINISTRAR EVENTOS · DETALLES EVENTO

IMAGEN



CLICK PARA CAMBIAR IMAGEN

FORMULARIO

TÍTULO

Lorem ipsum dolor

FECHA DEL EVENTO

02/05/2026

PÁRRAFO 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

PÁRRAFO 2

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

PÁRRAFO 3

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum

UBICACIÓN

Lorem ipsum dolor

ESTADO

Activo

Para configurar o cambiar la imagen de evento, haga clic en la imagen actual en el apartado "Escoger imagen".

IMAGEN



CLICK PARA CAMBIAR IMAGEN

TÍTULO

Lorem ips

PÁRRAFO

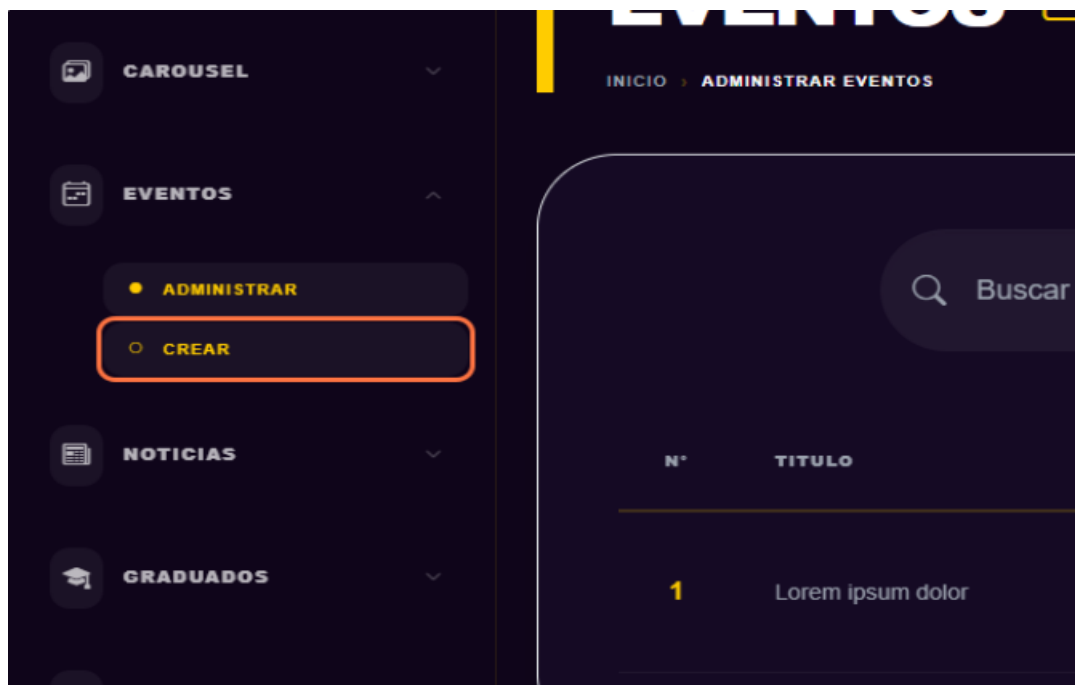
Lorem ips
eiusmod t
enim ad n
nisi ut alic

PÁRRAFO

nisi ut alic
in reprehe
nulla paria
sunt in cu

5.3 Crear un evento

El formulario para crear un evento se muestra al hacer clic en “Crear”.



Diligencie el formulario con la información del nuevo evento.

A screenshot of a form titled 'CREAR EVENTO'. The form contains several input fields: 'TITULO' with placeholder 'Titulo del evento', 'FECHA DEL EVENTO' with placeholder 'mm/dd/yyyy' and a calendar icon, 'PÁRRAFO 1' with placeholder 'Primer párrafo del contenido', 'PÁRRAFO 2' with placeholder 'Segundo párrafo (opcional)', and 'PÁRRAFO 3' with placeholder 'Tercer párrafo (opcional)'. Below these are 'UBICACIÓN' with placeholder 'Lugar del evento' and 'ESTADO' with a dropdown menu showing '--Seleccione una opción--'. At the bottom is an 'IMAGEN' section with a 'Choose File' button and the text 'No file chosen'. A large green button at the very bottom is labeled 'PUBLICAR EVENTO'.

Selecciona una imagen y haz clic en “Escoger imagen” para asociarla al evento.

UBICACIÓN

Lugar del evento

Este campo es requerido.

ESTADO

--Seleccione una opción--

Este campo es requerido.

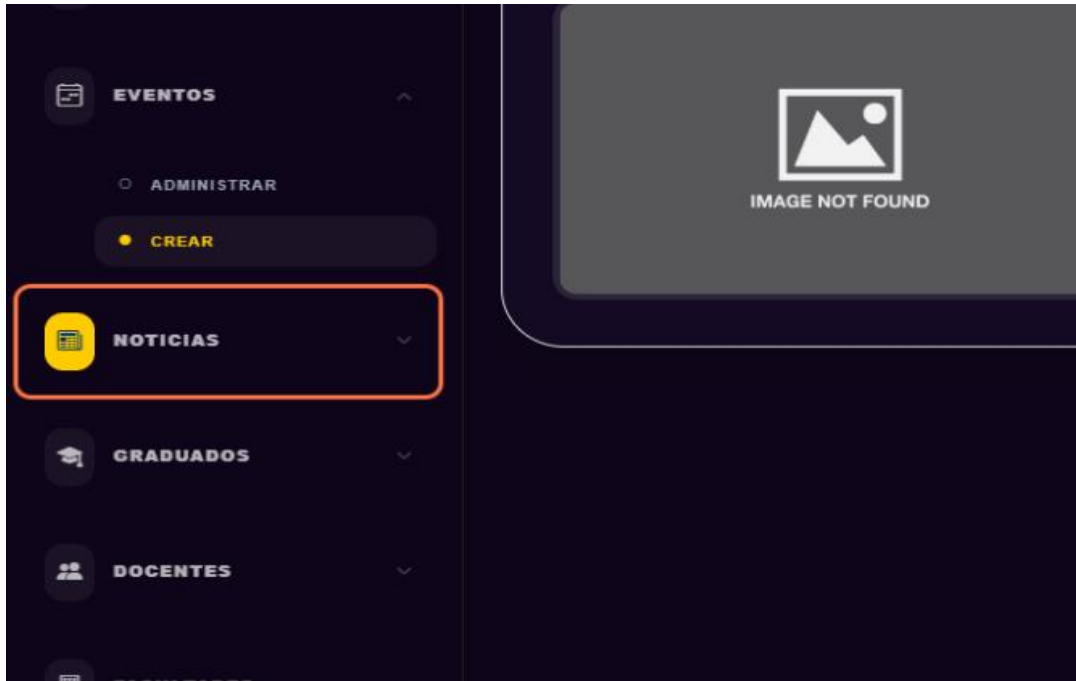
IMAGEN

Choose File No file chosen

PUBLICAR EVENTO

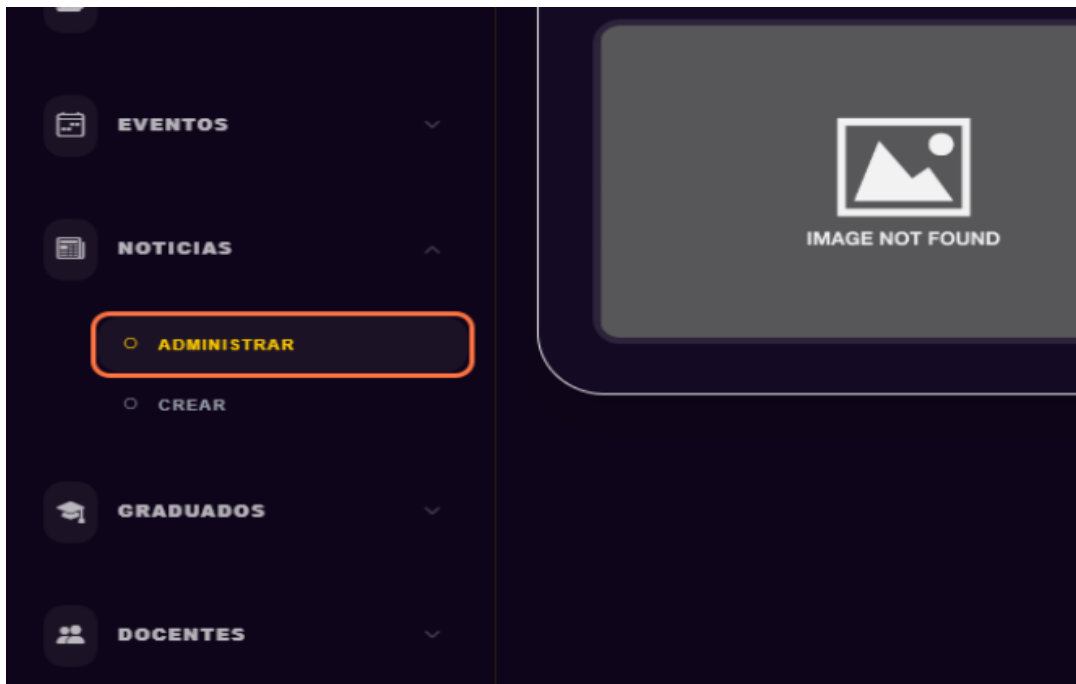
6. Gestión de noticias

Al acceder al menú lateral, expande la sección de Noticias.



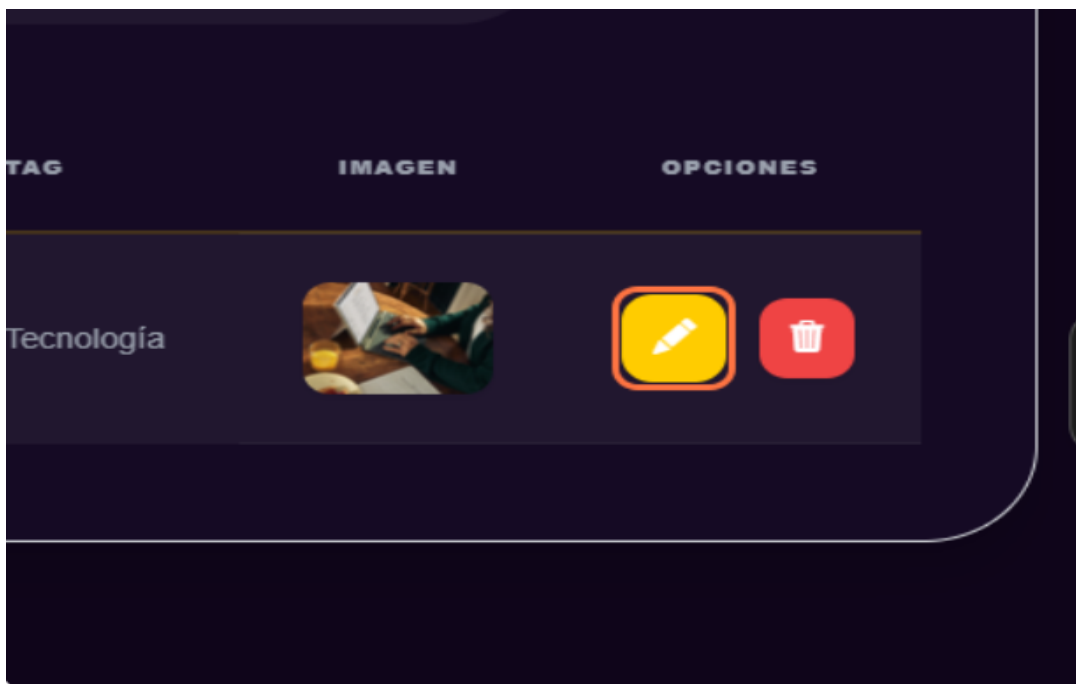
6.1 Administrar noticias

Elija la opción “**Administrar**” para editar o consultar las noticias registradas en la plataforma.



6.2 Editar una noticia

Haga clic en Editar (ícono de lápiz) para modificar o ver los elementos de noticias registrados en la plataforma.



Diligencie el formulario de edición según corresponda.

NOTICIAS

IMAGEN

CLICK PARA CAMBIAR IMAGEN

DETALLES NOTICIA

TÍTULO: Ejemplo de Títulos

FECHA DE PUBLICACIÓN: 02/19/2026

PÁRRAFO 1

PÁRRAFO 2

PÁRRAFO 3

AUTOR: Lorem Ipsum es simplemente

TAG: Tecnología

ACTUALIZAR

ELIMINAR

Para reemplazar la imagen de la noticia, haz clic en la imagen de la sección "Escoger imagen".

IMAGEN

CLICK PARA CAMBIAR IMAGEN

TÍTULO

>Lorem ipsum

PÁRRAFO

>Lorem ipsum
eiusmod
enim ad
nisi ut alic

PÁRRAFO

nisi ut alic
in reprehe
nulla para
sunt in cu

Haz clic en Actualizar para guardar los cambios

impresor (N. del T. persona que se dedica a la imprenta)

AUTOR	TAG
Lorem Ipsum es simplemer	Tecnología

ACTUALIZAR

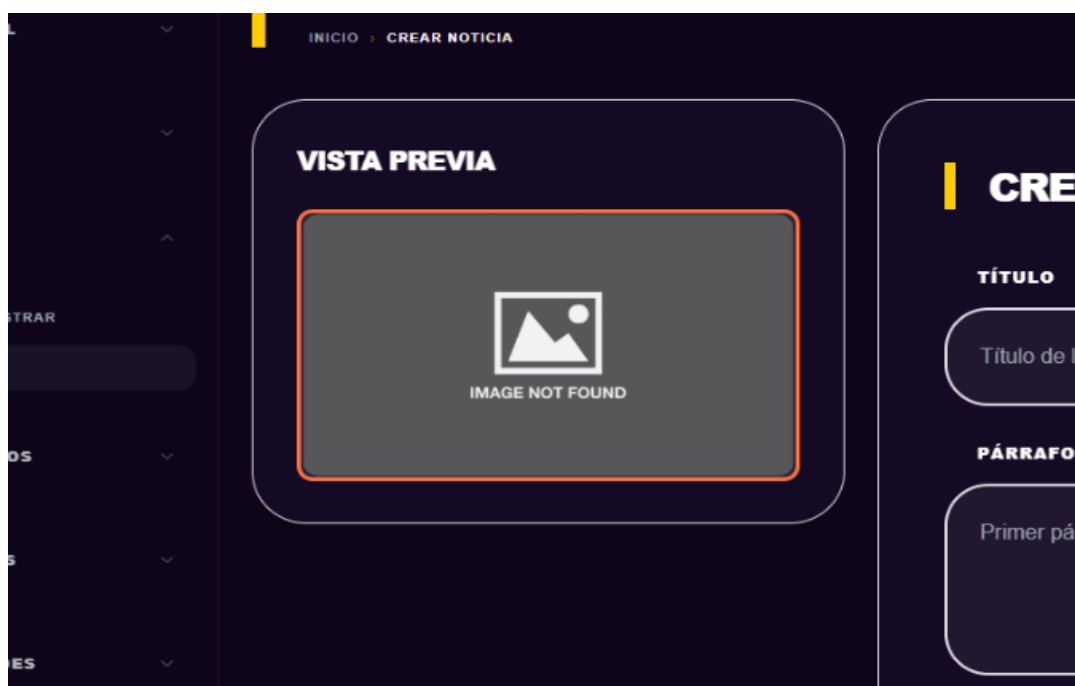
ELIMINAR

6.3 Crear una noticia

Elige **“Crea”** para redactar un artículo de noticias.



Para asociar una imagen, haga clic sobre la imagen o usa el botón “Adjuntar” archivo al final del formulario.



Diligencie el formulario con la información de la noticia.

NOTICIAS

INICIO CREAR NOTICIA

VISTA PREVIA



IMAGE NOT FOUND

CREAR NOTICIA

TÍTULO

Título de la noticia

FECHA DE PUBLICACIÓN

nov04yyyy

PÁRRAFO 1

Primer párrafo del contenido

PÁRRAFO 2

Segundo párrafo (opcional)

PÁRRAFO 3

Tercer párrafo (opcional)

AUTOR

Nombre del autor

TAG

-Selecciona una etiqueta-

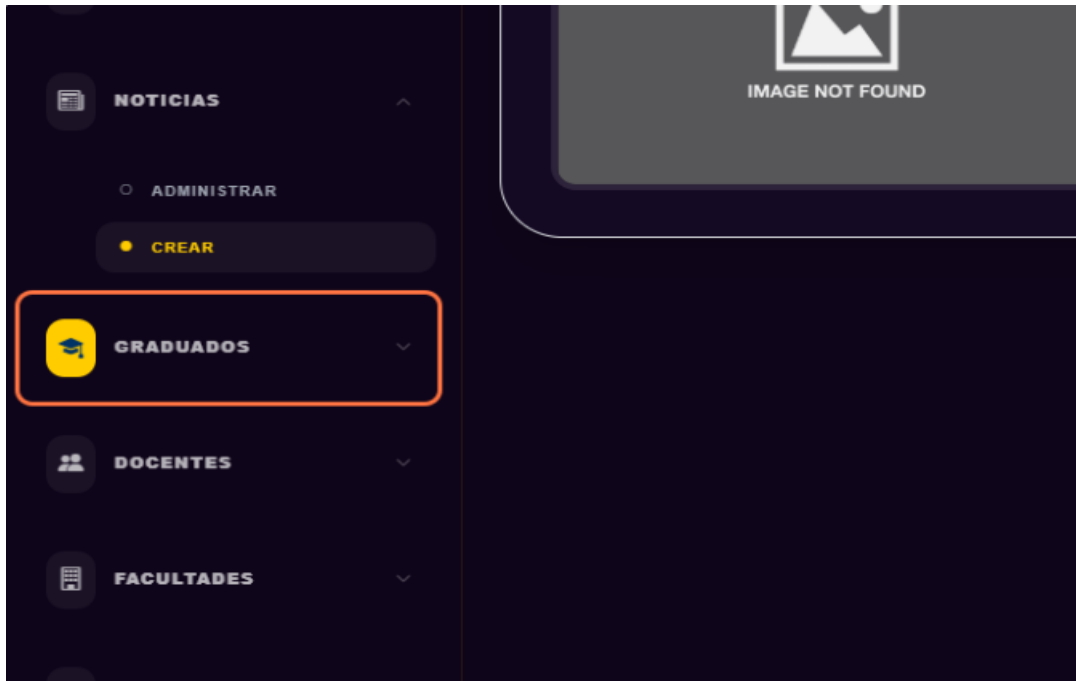
IMAGEN

Choose file

No file chosen

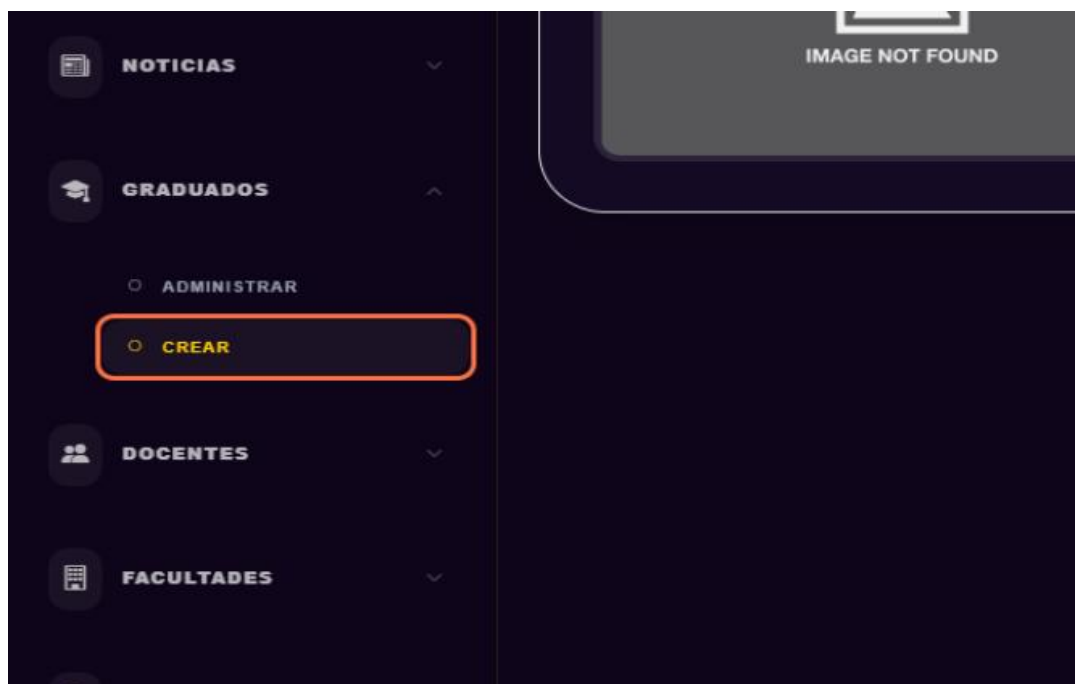
7. Gestión de graduados

Acceda al módulo de Graduados desde el menú lateral. Aquí puede ver, actualizar o crear un graduado.



7.1 Crear un graduado

Elija la opción “Crear” → “Graduados”.



Diligencie el formulario con la información del graduado.

INFORMACIÓN EGRESADO

FACULTAD
--Seleccione una facultad--

NOMBRE	APellido
CEdULA	PERIODO DE ADMISION
PERIODO DE EGRESO	FECHA DE GRADO
CEdULA	CORREO PERSONAL
RESIDENCIA	GÉNERO
URL LINKEDIN	URL CVLAE
TRABAJO	POSGRADO

GUARDAR INFORMACIÓN

Si el graduado tiene más de un empleo o un título de posgrado, haga clic en el botón amarillo **“Agregar otro”** para replicar según sea necesario y completar los campos correspondientes.

Formulario para agregar trabajo y posgrado. El formulario está dividido en dos secciones: 'TRABAJO' y 'POSGRADO'. La sección 'TRABAJO' contiene campos para 'CARGO', 'SPECIALIDAD', 'EMPRESA', 'PAIS' y 'CIUDAD'. La sección 'POSGRADO' contiene campos para 'TIPO', 'TÍTULO OBTENIDO' y 'UNIVERSIDAD'. Cada sección tiene un botón 'AGREGAR OTRO' con un ícono de flecha hacia arriba.

7.2 Administrar graduados

Elija la opción «Administrar» para ver los graduados creados.

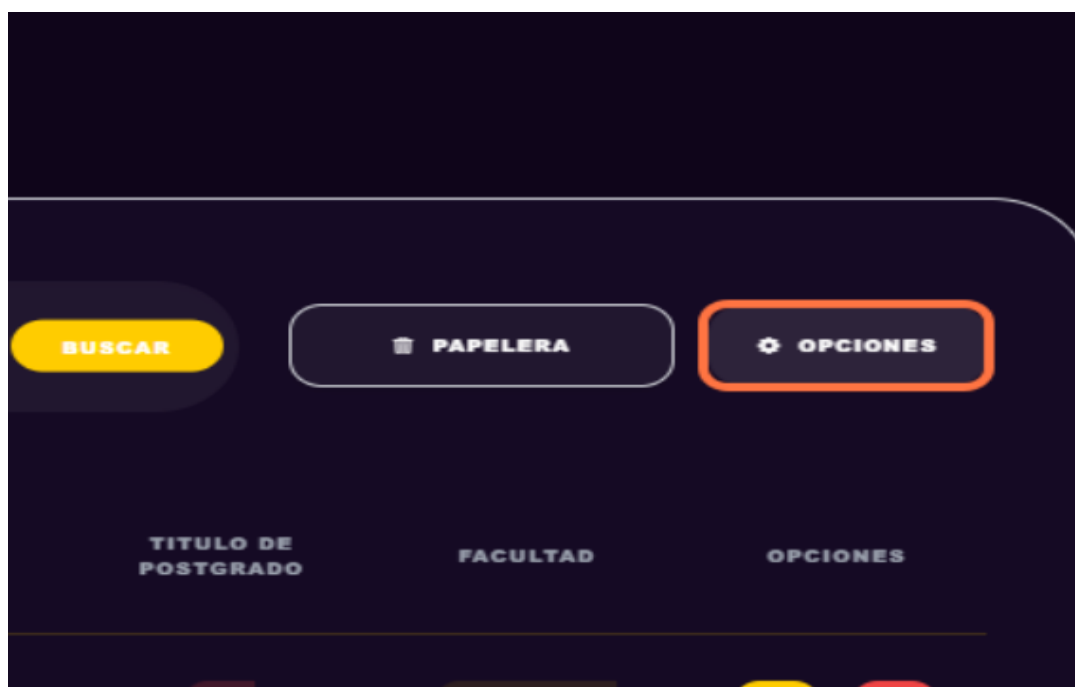
Interfaz de administración de graduados. A la izquierda, un menú lateral con opciones: 'EVENTOS', 'NOTICIAS', 'GRADUADOS' (seleccionado), 'ADMINISTRAR' (destacado con un recuadro naranja) y 'CREAR'. A la derecha, un formulario con campos para 'RESIDENCIA', 'URL LINKEDIN' y 'TRABAJO'. Los campos 'RESIDENCIA' y 'URL LINKEDIN' tienen mensajes de error: 'Este campo es requerido.'.

La lista especifica la facultad a la que está afiliada y el estado de empleo (empleado o sin empleo), con botones de acción asociados.



7.3 Importar / Exportar graduados (CSV)

En el botón “**Opciones**” se encuentran las funcionalidades de importación y exportación de datos.



Tanto la importación como la exportación operan con archivos en formato CSV.



7.4 Editar un graduado

Elija la opción “**Editar**” para modificar la información de un graduado.



Cambie o edite los detalles según corresponda.

En caso de un cambio de contraseña, vaya a Información del usuario, donde puede ver y administrar las credenciales del graduado también.

ADMINISTRACIÓN

INFORMACIÓN DEL USUARIO

USUARIO

asd.asdasdx123

CONTRASEÑA

FACULTAD

Ingeniería De Sistemas

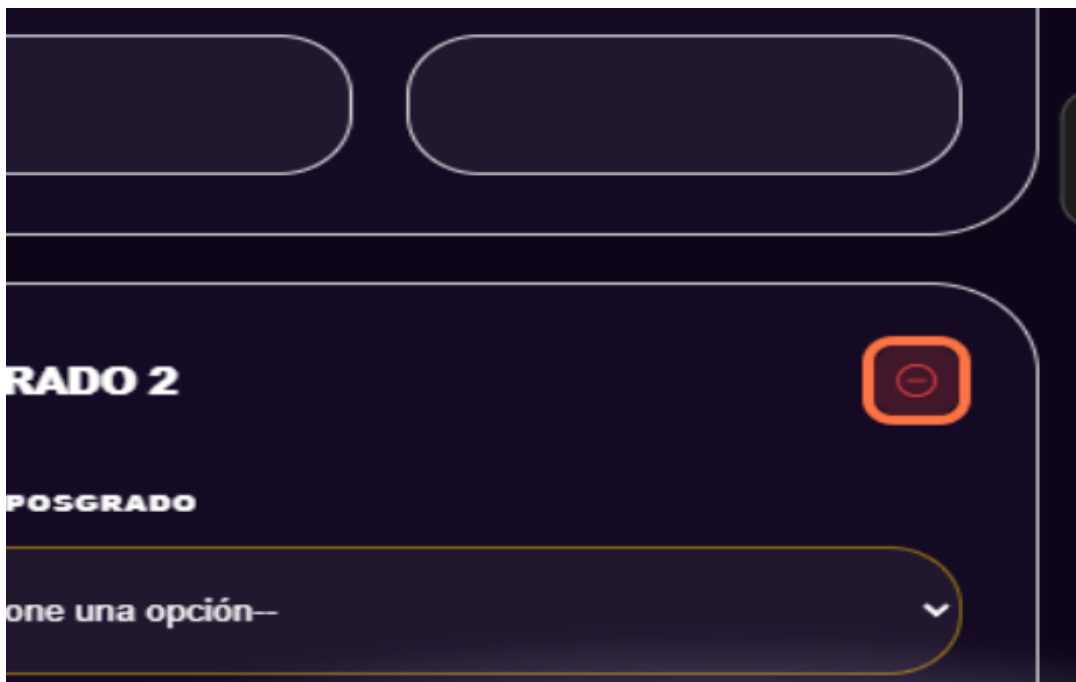
7.4.1 Empleos y posgrados

Si agrega otra calificación de posgrado, elija la opción **"Agregar otro posgrado"**.



The image shows a dark-themed user interface. On the left, there is a vertical sidebar with several rounded rectangular buttons, each containing a small square icon. The main area on the right contains a form. At the top of the form is a dark rounded rectangle with a small eye icon in the top right corner. Below this is a section labeled 'FACULTAD' in white uppercase letters. Underneath is a rounded rectangle containing the text 'Ingeniería De Sistemas' and a small downward arrow on the right. Below that is a prominent button with a yellow circle icon containing a white exclamation mark, followed by the text 'AGREGAR OTRO POSGRADO' in white uppercase letters. The button has a dashed orange border. Below this button is a section labeled 'POSGRADO' in white uppercase letters, preceded by a vertical yellow bar. Underneath is a section labeled 'TIPO DE POSGRADO' in white uppercase letters, followed by a rounded rectangle containing the text 'Elija una opción--' and a small downward arrow on the right.

Haga clic en el botón rojo para eliminar cualquier título de posgrado o empleo que haya agregado.



The image shows a dark-themed user interface. At the top, there are two rounded rectangular buttons. Below them is a section labeled 'POSGRADO 2' in white uppercase letters. To the right of this label is a red rounded square button with a white minus sign inside. Below the 'POSGRADO 2' label is a section labeled 'POSGRADO' in white uppercase letters. Underneath is a rounded rectangle containing the text 'Elija una opción--' and a small downward arrow on the right.

Agregue empleos haciendo clic en el botón correspondiente y completando los campos según corresponda.

The image shows a dark-themed user profile form. On the left is a sidebar with menu items: "EQUIPOS", "ADMINISTRAR", "CREAR", "CATEGORÍAS", "RESULTADOS", and "REPORTES". The main form area contains several input fields and dropdown menus. The fields are organized as follows:

- Personal Information:** A text field containing "asdasdasd" and a dropdown menu set to "Femenino".
- Social Media:** Two text fields labeled "URL LINKEDIN" and "URL CVLAC", both containing "asdasdasd".
- Education:** Two sections. The "TRABAJO" section has a dropdown menu currently showing "Aplica", which is highlighted with an orange border. The "POSGRADO" section has a text field containing "Aplica".

Ahora, para agregar otro título de posgrado, repita el mismo proceso tocando "**Agregar otro posgrado.**"

The image shows a dark-themed web interface for adding a new job. On the left is a sidebar with several menu items, each with a small icon. The main content area is divided into sections. At the top, there is a search bar with a magnifying glass icon. Below it is a section titled "FACULTAD" with a dropdown menu currently showing "Ingeniería De Sistemas". Underneath this is a prominent button with a yellow exclamation mark icon and the text "AGREGAR OTRO TRABAJO", which is highlighted with a dashed orange border. Below the button is a section titled "TRABAJO" with a yellow vertical bar icon to its left. Under the "TRABAJO" section is a label "CARGO" followed by an empty input field.

8. Gestión de docentes

Acceda al módulo de Docentes desde el menú lateral.

The screenshot displays a web application interface. On the left is a dark sidebar with a vertical menu. The menu items are: 'NOTICIAS' (with a calendar icon), 'GRADUADOS' (with a graduation cap icon), 'DOCENTES' (with a group of people icon, highlighted with an orange rectangle), 'FACULTADES' (with a document icon), and 'REPORTES' (with a bar chart icon). Below 'GRADUADOS' are two sub-options: 'ADMINISTRAR' and 'CREAR'. The main content area on the right is a form for adding a new teacher. It contains several input fields: 'NOMBRE' (Name) with the value 'Jose Maria', 'APELLIDO' (Last Name) with the value 'Zalas', 'DOCUMENTO DE IDENTIDAD' (ID Document) with the value '123123123123', 'PERIODO DE EGRESO' (Graduation Period) with the value '05/16/2026', and 'FECHA' (Date) with the value '05/23'. The form is styled with a dark background and light-colored text and borders.

8.1 Administrar docentes

Elija la opción “**Administrar**” para gestionar a los usuarios docentes según sea necesario.

NOTICIAS

GRADUADOS

DOCENTES

ADMINISTRAR

CREAR

FACULTADES

REPORTES

INFORMACION

NOMBRE

Jose Maria

DOCUMENTO DE IDENTIDAD

123123123123

PERIODO DE EGRESO

05/16/2026

El resultado muestra cada docente agregado con su cargo y facultad, seguido de las opciones “Editar” y “Eliminar”.

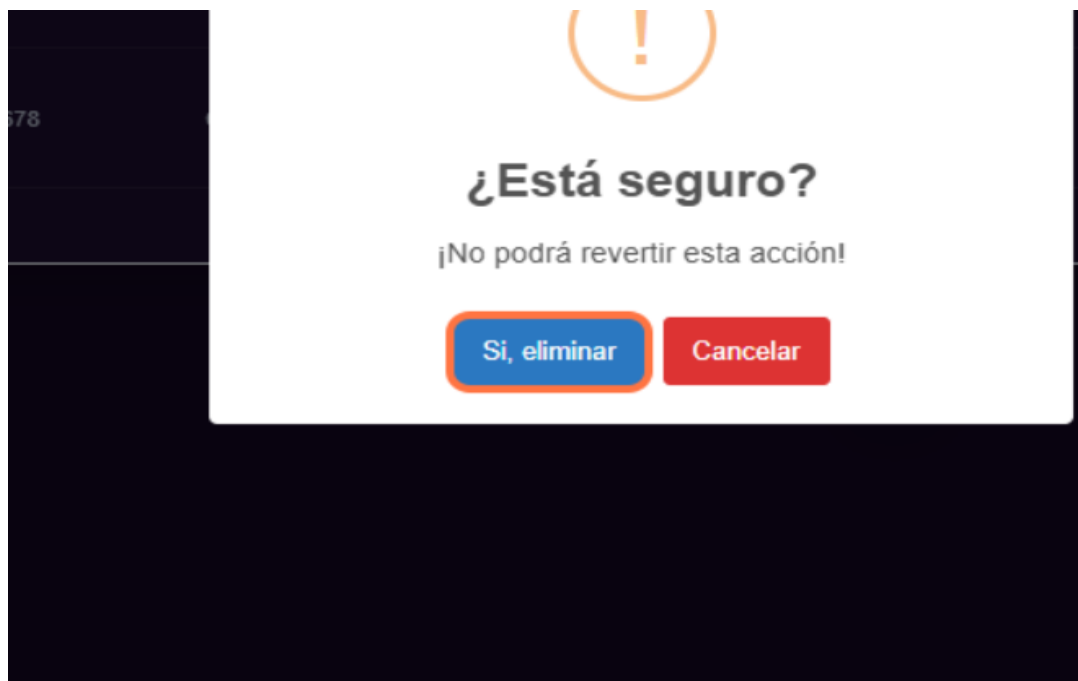
	CARGO	FACULTAD	OPCIONES
il.com	Decano	INGENIERÍA DE SISTEMAS	 
sta.edu.co	Docente de Planta	INGENIERÍA DE SISTEMAS	 

8.2 Eliminar un docente o decano

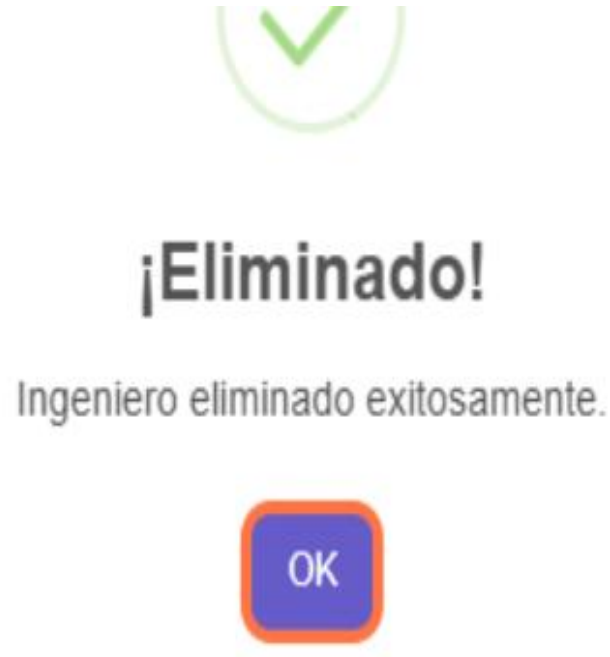
Para eliminar a un docente o a un decano, utilice el botón correspondiente.



El sistema mostrará una advertencia que confirmará la acción a ejecutar.



Se mostrará un mensaje indicando que el usuario se eliminó al hacer clic en confirmar.

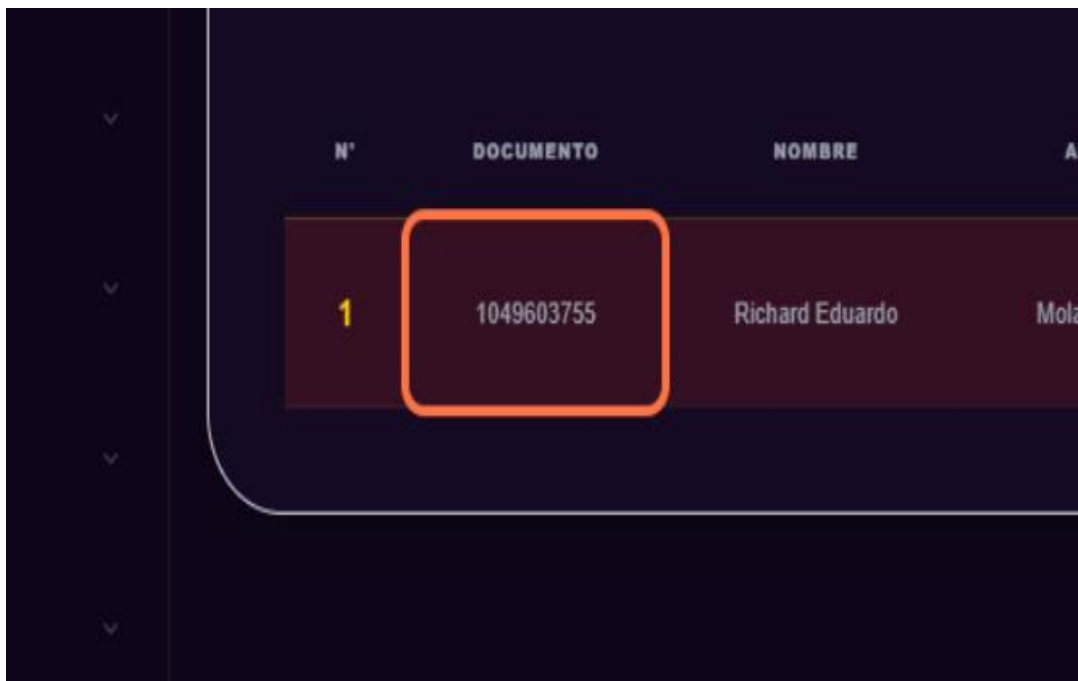


8.3 Papelera y restauración

Puede restaurar usuarios eliminados por error desde la Papelera de reciclaje.



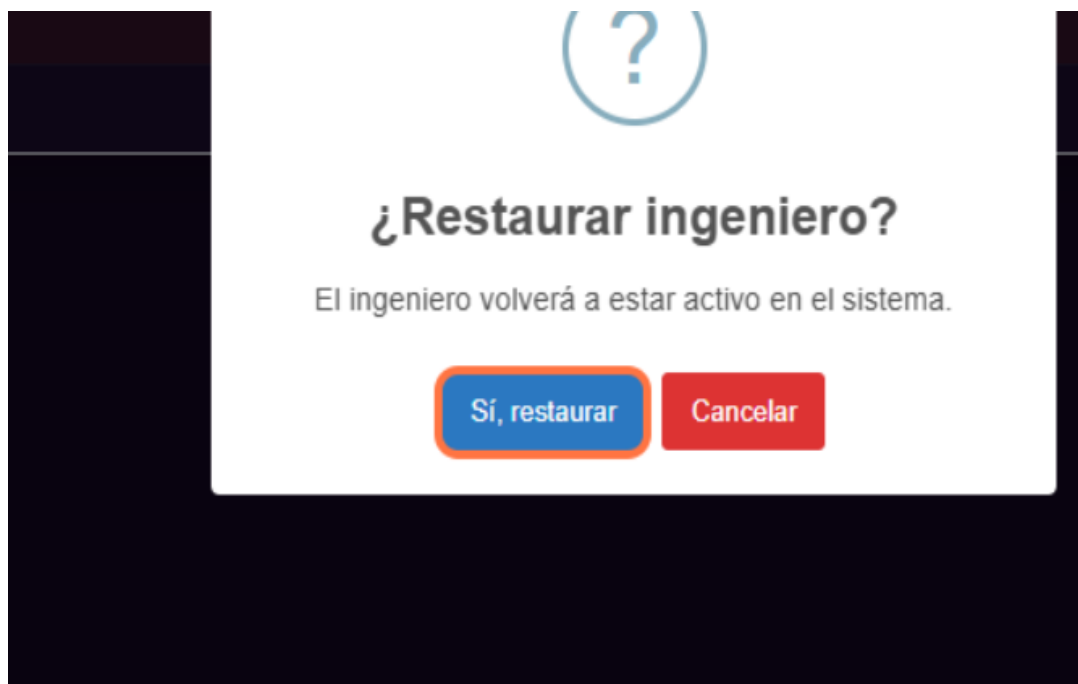
Se mostrarán los datos e información del usuario eliminado.



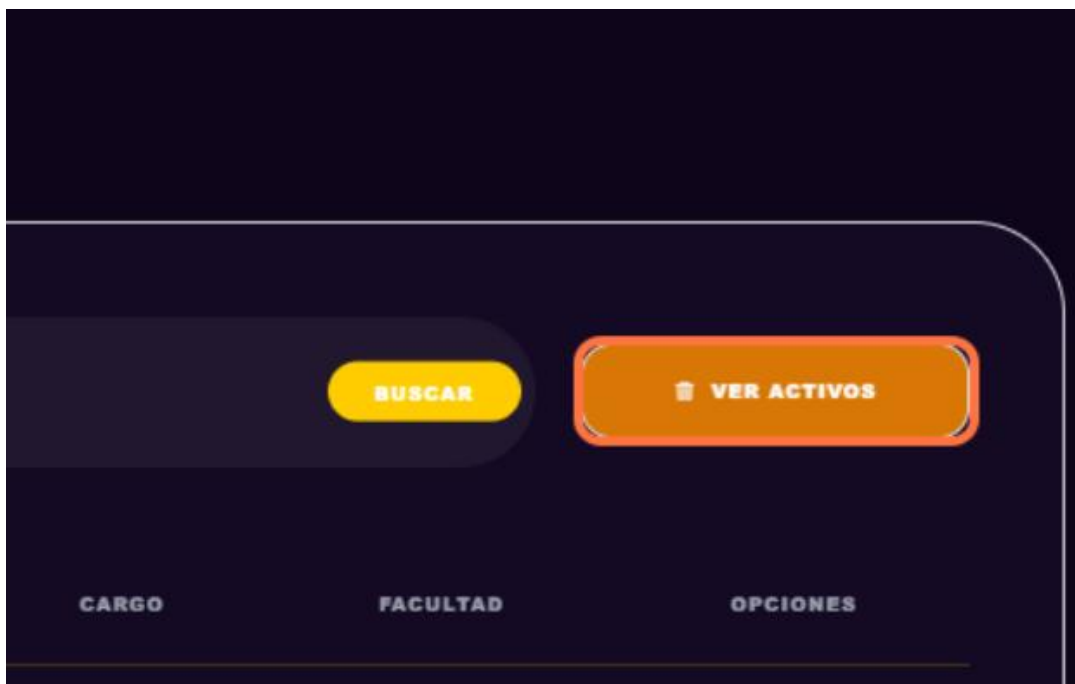
Restaurelo haciendo clic en el botón verde.



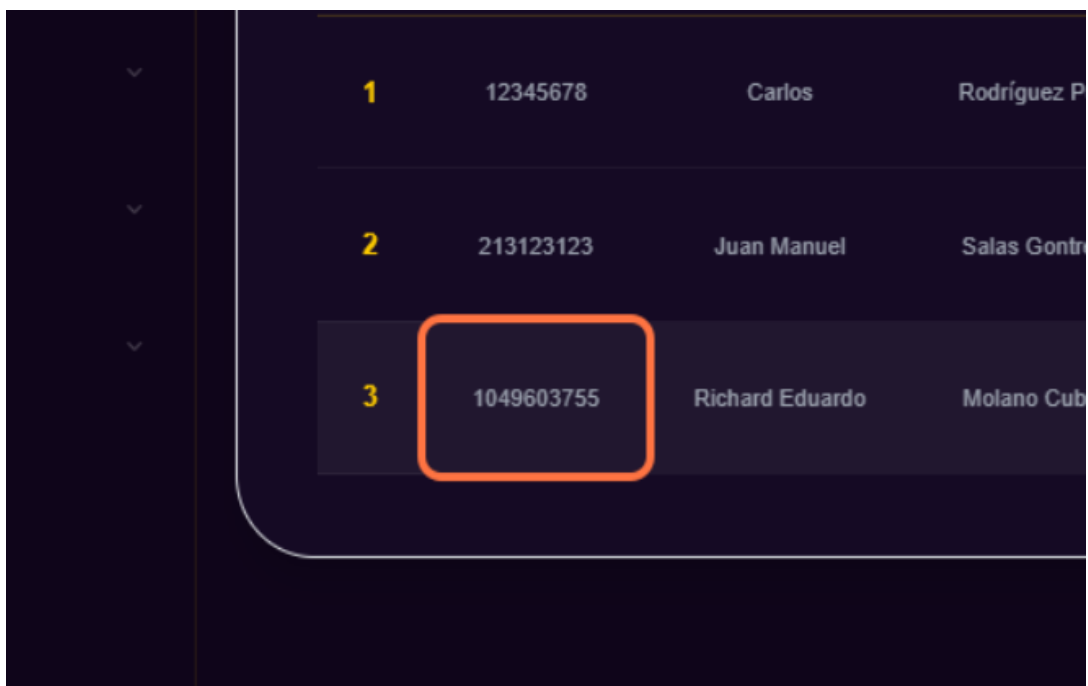
El sistema informará que el usuario será restaurado.



Al volver a la vista de la Papelera de reciclaje, lo llevará de regreso a Activos.

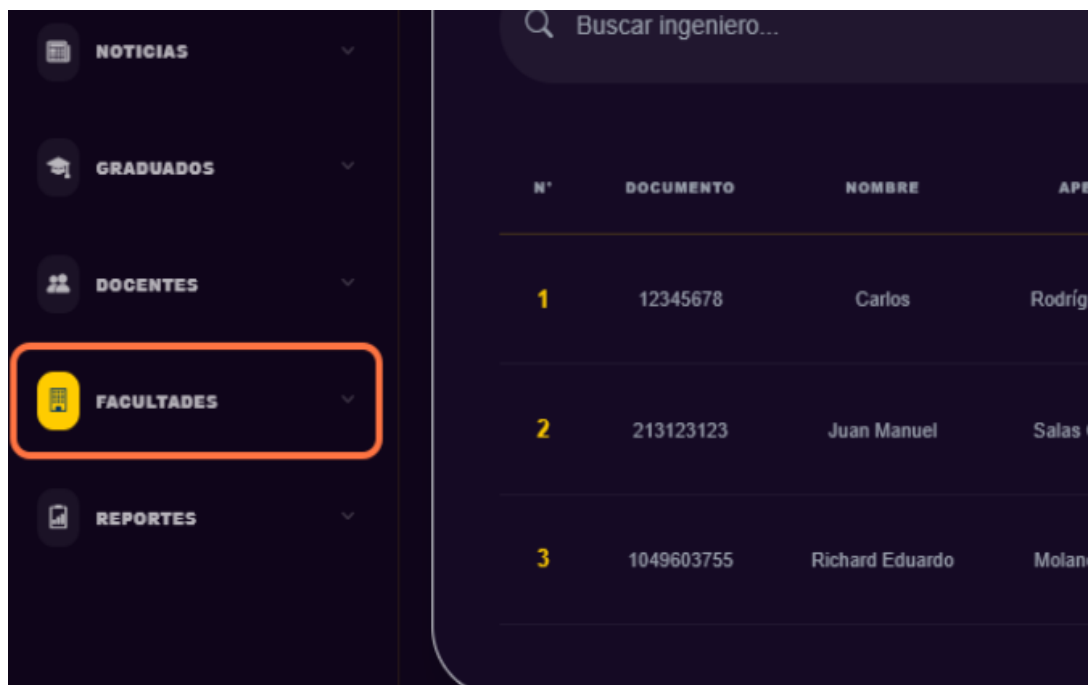


Nuevamente, el usuario eliminado aparecerá en la lista activa.



9. Gestión de facultades

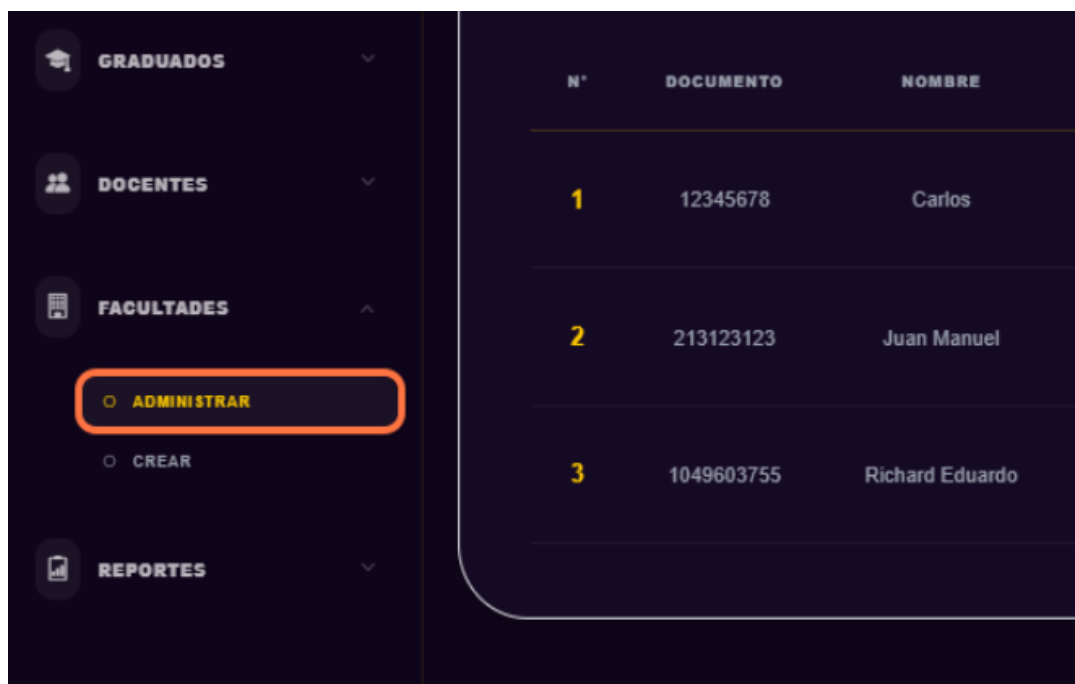
Navegue desde el menú lateral hasta el módulo de Facultades. Esta página mostrará las facultades a las que tiene acceso (según sea decano o administrador).



N°	DOCUMENTO	NOMBRE	APELLIDO
1	12345678	Carlos	Rodríguez
2	213123123	Juan Manuel	Salas
3	1049603755	Richard Eduardo	Molano

9.1 Administrar facultades

Ahora, puede gestionar las facultades activas desde allí.



The screenshot shows a web application interface. On the left is a dark sidebar with navigation options: GRADUADOS, DOCENTES, FACULTADES, ADMINISTRAR (highlighted with an orange box), CREAR, and REPORTE. The main area displays a table with three columns: N°, DOCUMENTO, and NOMBRE. The table contains three rows of data.

N°	DOCUMENTO	NOMBRE
1	12345678	Carlos
2	213123123	Juan Manuel
3	1049603755	Richard Eduardo

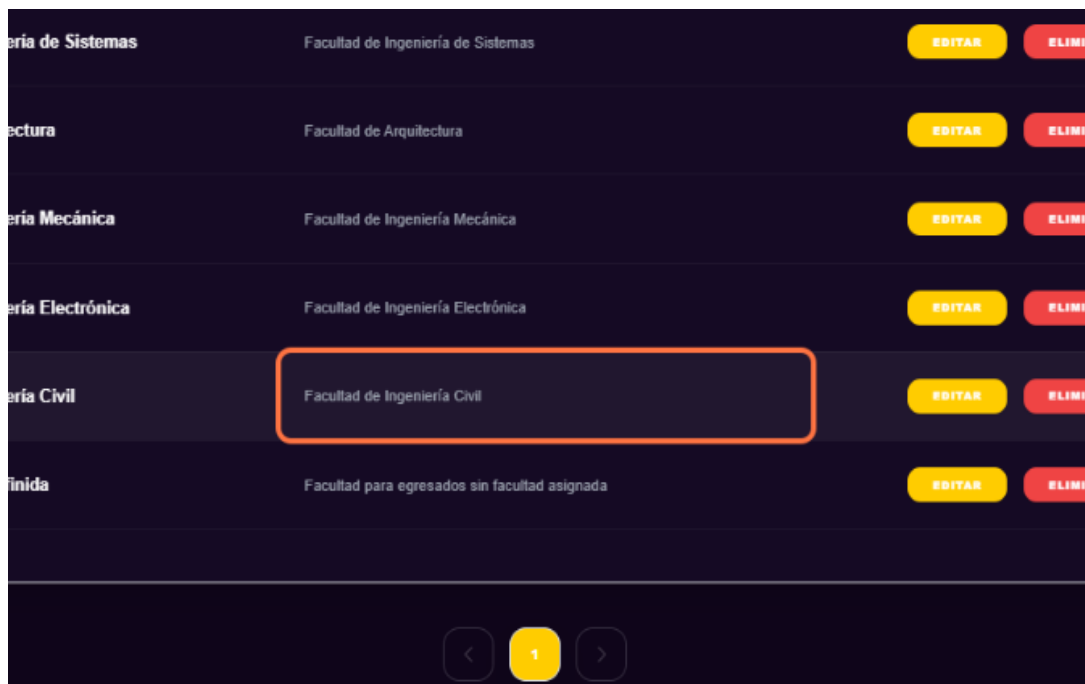
La vista muestra la facultad y su respectiva carrera.



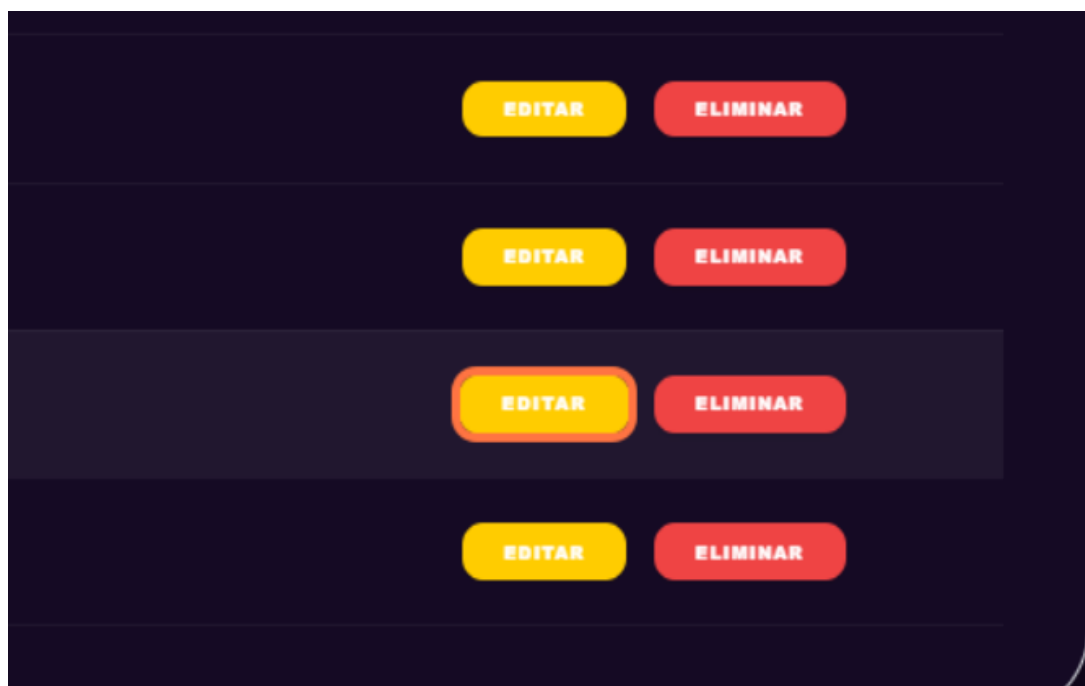
The screenshot shows a web application interface displaying a table with three columns: an index number, the career name, and the faculty name. The table contains five rows of data. The row for 'Ingeniería Civil' is highlighted with an orange box.

2	Arquitectura	Facultad de Arquitectura
3	Ingeniería Mecánica	Facultad de Ingeniería Mecánica
4	Ingeniería Electrónica	Facultad de Ingeniería Electrónica
5	Ingeniería Civil	Facultad de Ingeniería Civil
6	No Definida	Facultad para egresados sin facultad as

Vista complementaria de facultades.



Cada registro permite “**Editar**” y “**Eliminar**” la facultad según sea necesario.



9.2 Editar una facultad

En “**Editar**”, puede modificar el nombre y la descripción de la facultad.

TALLE FACULTAD

ADMINISTRAR FACULTADES

← **INFORMACIÓN DE LA FACULTAD**

NOMBRE

Ingeniería Civil

DESCRIPCIÓN

Facultad de Ingeniería Civil

GUARDAR CAMBIOS

Recibirá una notificación que confirme los cambios.

← **INFORMACIÓN DE LA FACULTAD**

NOMBRE

Ingeniería Civil

DESCRIPCIÓN

Facultad de Ingeniería Civil

GUARDAR CAMBIOS

9.3 Crear una facultad

Elija la opción **“Crear”** para registrar nuevas facultades que podrán asignarse a nuevos usuarios.

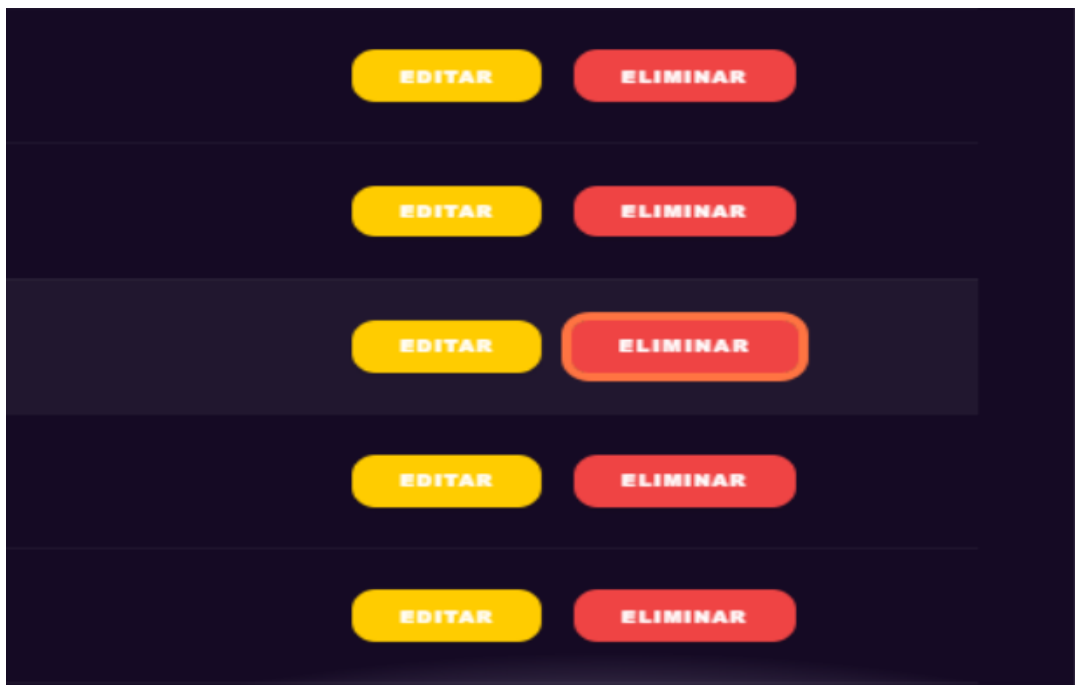


Diligencie el formulario indicando el nombre y la descripción de la facultad.

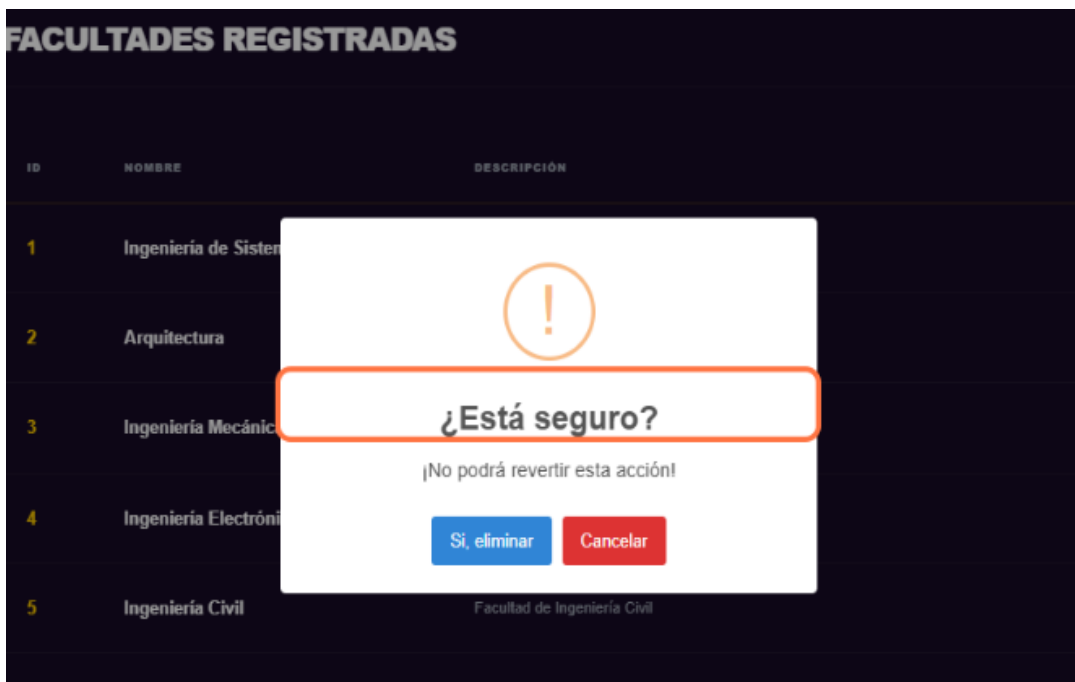
The image shows a form titled 'INFORMACIÓN FACULTAD' with two main sections. The first section, 'NOMBRE DE LA FACULTAD', contains a text input field with the placeholder text 'Ej: Ingeniería de Sistemas' and a small icon to its right. The second section, 'DESCRIPCIÓN', contains a larger text area with the placeholder text 'Descripción breve de la facultad'.

9.4 Eliminar una facultad

Si desea eliminarla, se mostrará una pantalla de confirmación en la que se le preguntará si está seguro de que desea eliminar la facultad.



Vista resultante tras la eliminación.

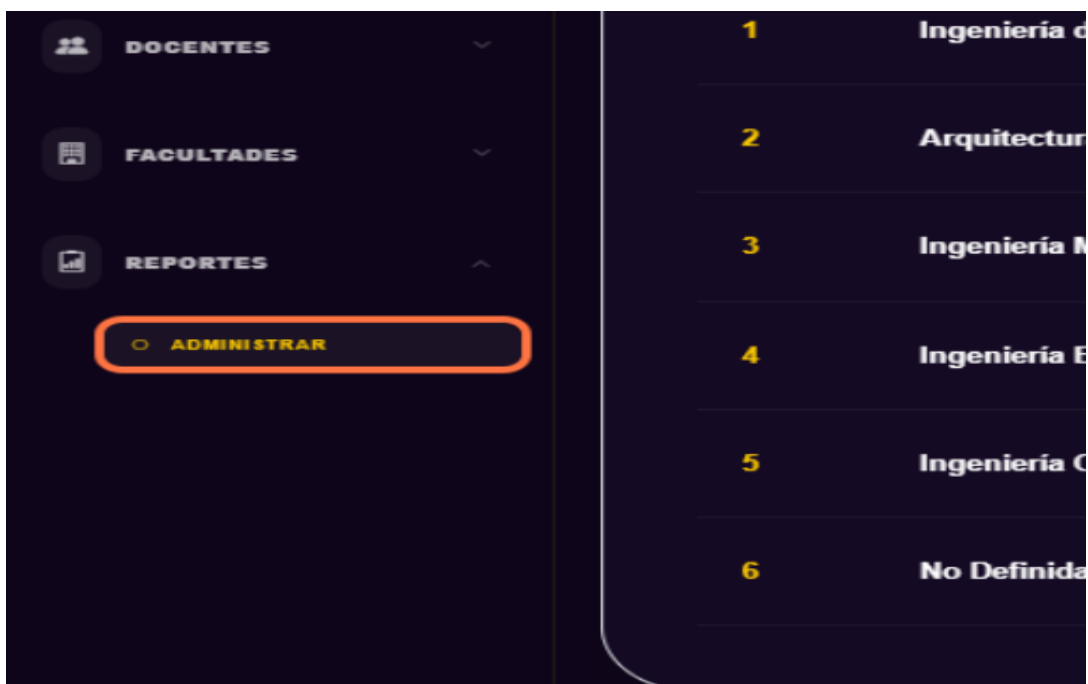


10. Reportes y estadísticas

En el apartado de Reportes, puedes acceder según el permiso que te haya sido otorgado y tu facultad.



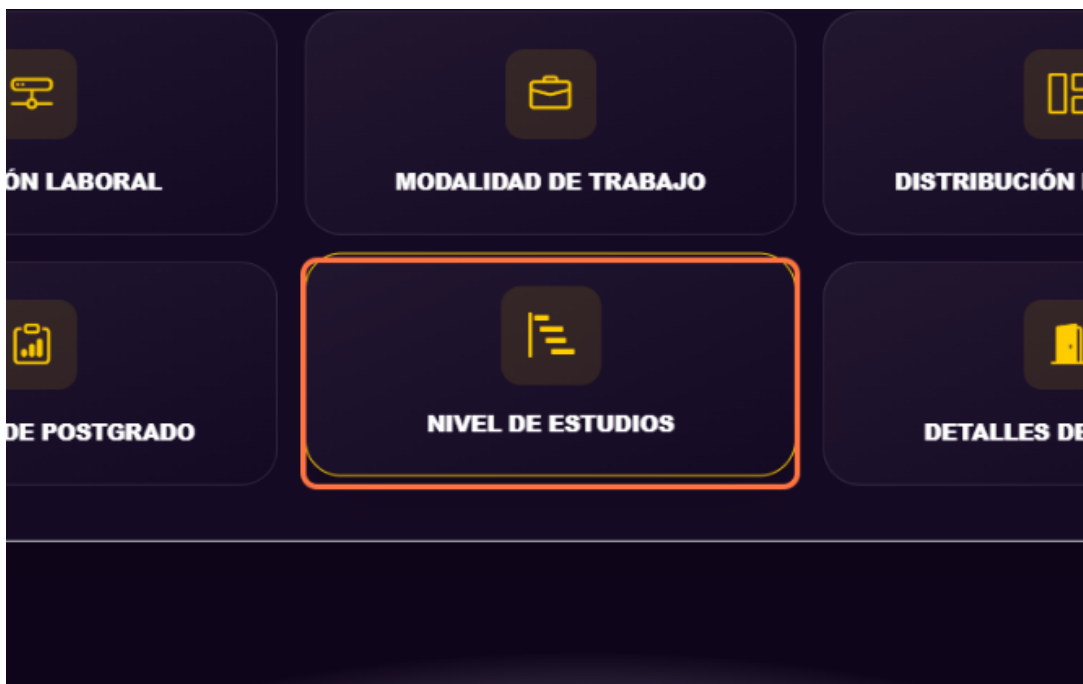
Elija la opción **“Administrar”** para acceder a los reportes.



Se mostrarán los seis módulos de reportes disponibles. Elige cuál necesitas consultar.



Al hacer clic en cualquier módulo, puedes ver el gráfico correspondiente al caso de estudio seleccionado.



Este gráfico contendrá estadísticas sobre los graduados de tu facultad



Anexo 3. Evidencias Reunión Actores clave

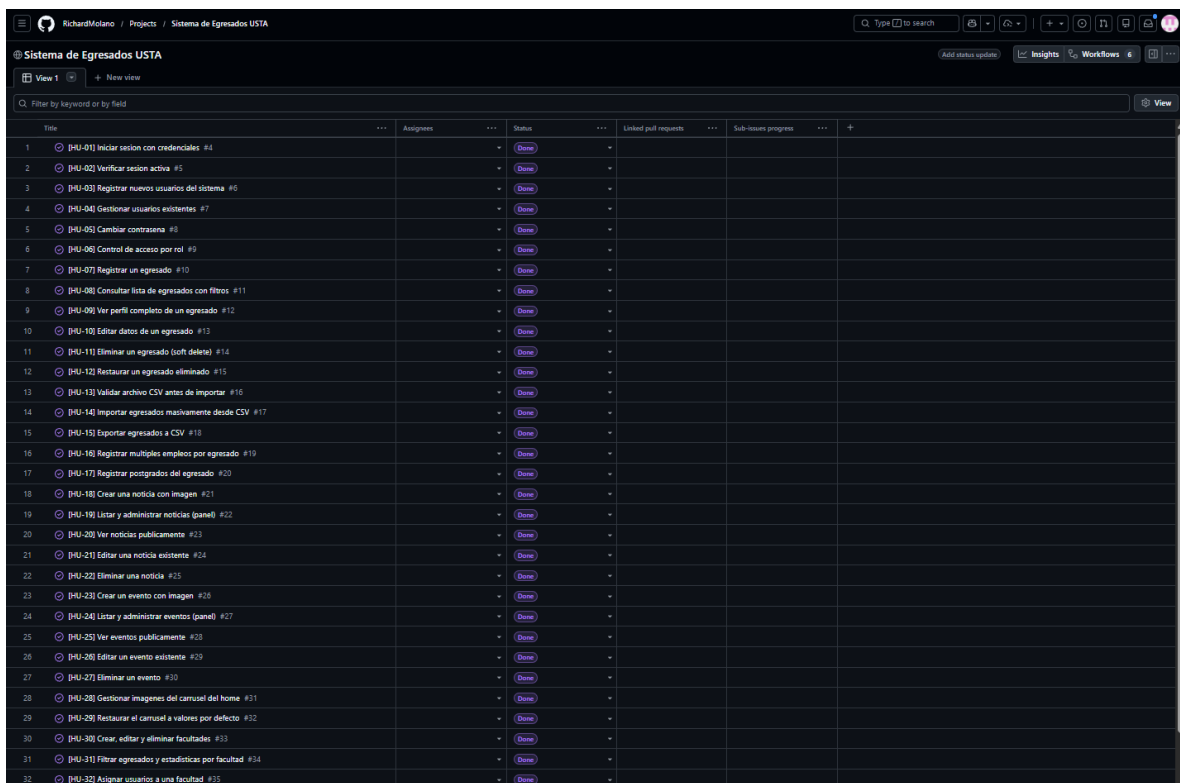
Los entregables de la reunión con el ingeniero Diego Alejandro Vela Beltrá quien gracias a su posición clave como coordinador de graduados y el graduado de ingeniería de sistemas Julián Camilo Escobar Araque también autor del primer prototipo del sistema a seguimiento a egresados para la facultad de sistemas se encuentran en un documento de síntesis de transcripción en:

<https://drive.google.com/file/d/1kD0gkjeMqjbCTguG50x3QnmWL1P2IsC7/view?usp=sharing>

Anexo 4. GitHub

El anexo al repositorio de GitHub se encuentra en privado debido a las conexiones que tiene con el servidor institucional; adjunto imágenes y su acceso se encuentra brindado al servidor de la institución, junto al ingeniero Diego Alejandro Vela Beltrá.

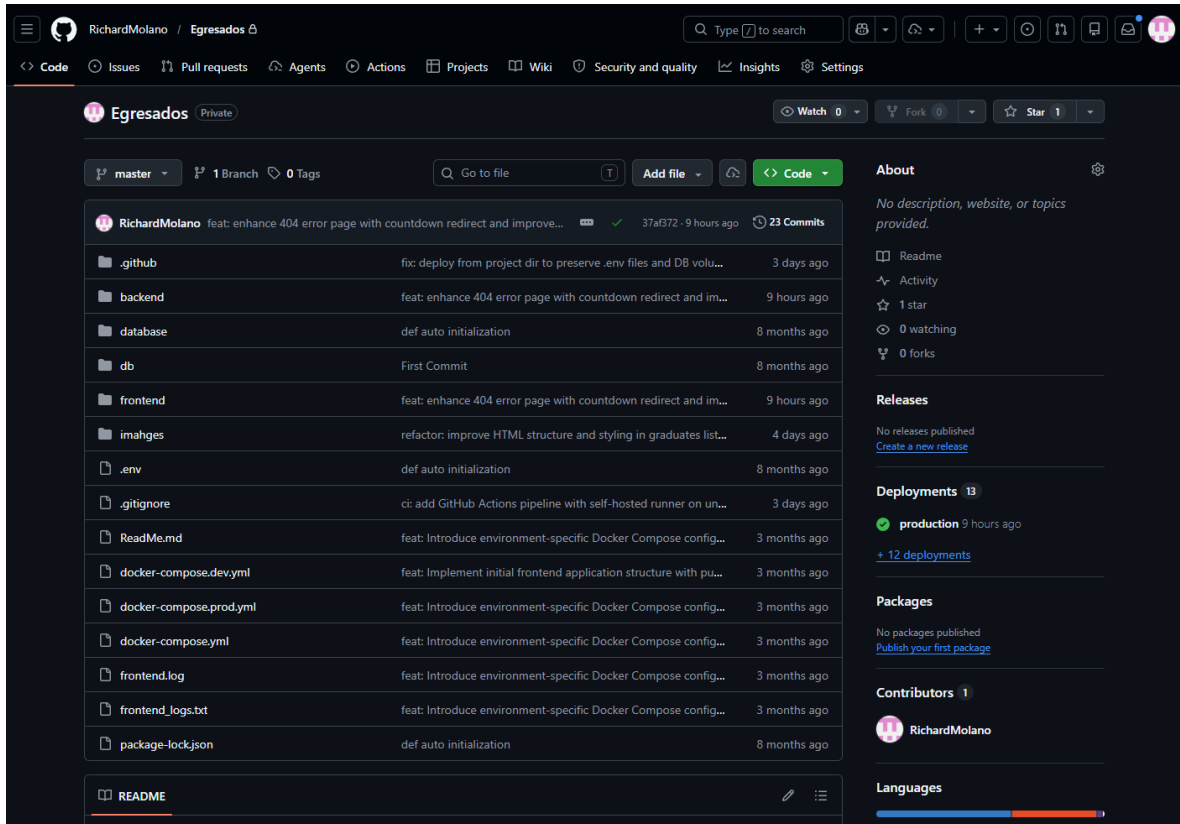
Figura Backlog



Title	Assignee	Status	Linked pull request	Sub-issues progress	
1 [MJ-01] Iniciar sesión con credenciales #4		Done			
2 [MJ-02] Verificar sesión activa #5		Done			
3 [MJ-03] Registrar nuevos usuarios del sistema #6		Done			
4 [MJ-04] Gestionar usuarios existentes #7		Done			
5 [MJ-05] Cambiar contraseña #8		Done			
6 [MJ-06] Control de acceso por rol #9		Done			
7 [MJ-07] Registrar un egresado #10		Done			
8 [MJ-08] Consultar lista de egresados con filtros #11		Done			
9 [MJ-09] Ver perfil completo de un egresado #12		Done			
10 [MJ-10] Editar datos de un egresado #13		Done			
11 [MJ-11] Eliminar un egresado (soft delete) #14		Done			
12 [MJ-12] Restaurar un egresado eliminado #15		Done			
13 [MJ-13] Validar archivo CSV antes de importar #16		Done			
14 [MJ-14] Importar egresados masivamente desde CSV #17		Done			
15 [MJ-15] Exportar egresados a CSV #18		Done			
16 [MJ-16] Registrar múltiples empleos por egresado #19		Done			
17 [MJ-17] Registrar postgrados del egresado #20		Done			
18 [MJ-18] Crear una noticia con imagen #21		Done			
19 [MJ-19] Listar y administrar noticias (panel) #22		Done			
20 [MJ-20] Ver noticias publicamente #23		Done			
21 [MJ-21] Editar una noticia existente #24		Done			
22 [MJ-22] Eliminar una noticia #25		Done			
23 [MJ-23] Crear un evento con imagen #26		Done			
24 [MJ-24] Listar y administrar eventos (panel) #27		Done			
25 [MJ-25] Ver eventos publicamente #28		Done			
26 [MJ-26] Editar un evento existente #29		Done			
27 [MJ-27] Eliminar un evento #30		Done			
28 [MJ-28] Gestionar imágenes del carrusel del home #31		Done			
29 [MJ-29] Restaurar el carrusel a valores por defecto #32		Done			
30 [MJ-30] Crear, editar y eliminar facultades #33		Done			
31 [MJ-31] Filtrar egresados y estadísticas por facultad #34		Done			
32 [MJ-32] Asignar usuarios a una facultad #35		Done			

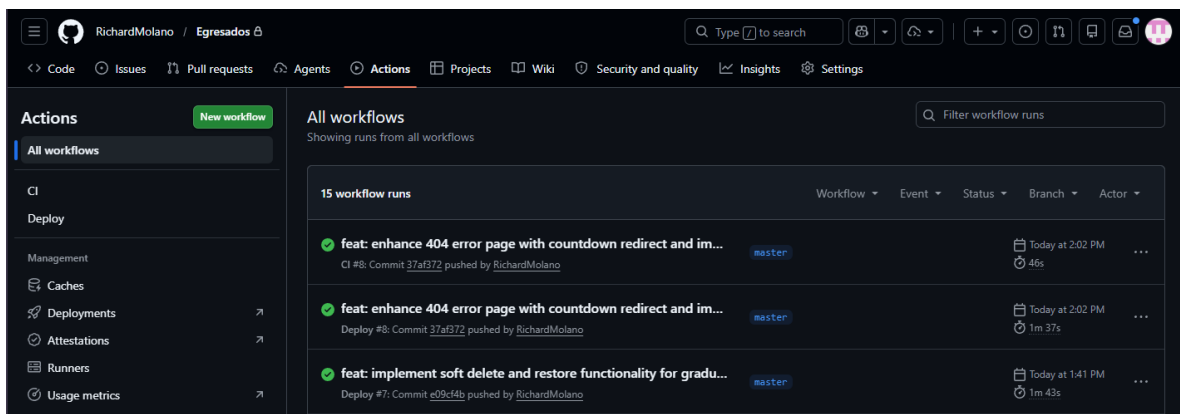
Fuente Autor

Figura GitHub Repositorio



Fuente Autor

Figura Github Actions



Fuente Autor