



REPÚBLICA DE PANAMÁ

UNIVERSIDAD INTERNACIONAL DE CIENCIA Y TECNOLOGÍA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN Y TECNOLOGÍA

**PLATAFORMA PARA LA GESTIÓN Y SEGUIMIENTO DE SOLICITUDES
ESTUDIANTILES EN LA UNIVERSIDAD UNICYT**

PROYECTO DE TRABAJO PARA OPTAR AL GRADO DE MAGISTER

**MAGÍSTER EN INGENIERÍA DE SOFTWARE CON ÉNFASIS EN APLICACIONES
EMPRESARIALES BAJO ENTORNO WEB**

Tutor: JOSÉ A. MC LEAN R

Autor: ORIEL ARTURO CEDEÑO MUÑOZ

Ciudad de Panamá, 16 de marzo de 2026



REPÚBLICA DE PANAMÁ

UNIVERSIDAD INTERNACIONAL DE CIENCIA Y TECNOLOGÍA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN Y TECNOLOGÍA

**PLATAFORMA PARA LA GESTIÓN Y SEGUIMIENTO DE SOLICITUDES
ESTUDIANTILES EN LA UNIVERSIDAD UNICYT**

PROYECTO DE TRABAJO PARA OPTAR AL GRADO DE MAGISTER

**MAGÍSTER EN INGENIERÍA DE SOFTWARE CON ÉNFASIS EN APLICACIONES
EMPRESARIALES BAJO ENTORNO WEB**

Autor: ORIEL ARTURO CEDEÑO MUÑOZ

Ciudad de Panamá, 16 de marzo de 2026



Ciudad de Panamá, 17 de marzo de 2026

Profesor

Dr. Héctor Mazurkiewicz

Coordinador del Comité de Titulación de Estudios de Grado y Postgrado

Presente.

En mi carácter de Tutor del Proyecto de Graduación de Maestría Profesional, presentado por el (la) estudiante Oriel A. Cedeño, para optar al grado de, Magíster en Ingeniería de Software con énfasis en Aplicaciones Empresariales Bajo Entorno Web considero que el trabajo: reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del Jurado examinador que se designe.

Atentamente,



José A. Mc Lean R.

Línea de Investigación: Ingeniería de software y gestión de procesos.



UNIVERSIDAD INTERNACIONAL DE CIENCIA Y TECNOLOGÍA
FACULTAD DE CIENCIAS DE LA COMPUTACIÓN Y TECNOLOGÍA

INFORME DE ACTIVIDADES DE TUTORÍA OPCIÓN DE TITULACIÓN DE PROYECTO DE GRADUACIÓN DE MAESTRÍA PROFESIONAL

Programa de Maestría: Magíster En Ingeniería De Software Con Énfasis En Aplicaciones Empresariales Bajo Entorno Web

Estudiante: Oriel Arturo Cedeño Muñoz

Tutor: Prof. José A. Mc Lean

Título del trabajo de grado: Plataforma Para La Gestión Y Seguimiento De Solicitudes Estudiantiles En La Universidad UNICYT

Línea de Investigación: Ingeniería de software y gestión de procesos.

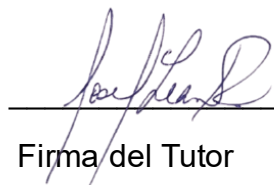
SESIÓN	FECHA	HORA REUNIÓN.	ASPECTO TRATADO	OBSERVACIÓN
1	22/11/2025	4pm.	Definición y contextualización del problema.	
2	6/12/2025	4pm.	Revisión de antecedentes y bases teóricas.	
3	13/12/2025	4pm.	Análisis del modelo actual (AS-IS) y requisitos.	
4	22/12/2025	4pm.	Análisis del modelo actual (AS-IS) y requisitos.	
5	10/01/2026	4pm.	Diseño de arquitectura	

			bajo patrón MVC.	
6	24/01/2026	4pm.	Avance de implementación del prototipo en PHP.	
7	7/02/2026	4pm.	Pruebas funcionales y validación del sistema.	
8	21/02/2026	4pm.	Revisión final de conclusiones y bibliografía.	

Titulo definitivo:

Plataforma Para La Gestión Y Seguimiento De Solicitudes Estudiantiles En La Universidad UNICYT

Comentarios finales acerca de la investigación: Declaramos que las especificaciones anteriores representan el proceso de dirección del trabajo de grado arriba mencionado.



Firma del Tutor



Firma del Estudiante

DEDICATORIA

A Dios, por guiar mi camino y darme la fortaleza para llegar a la meta.

A mis padres, por su amor infinito y por ser el pilar sobre el cual construí este sueño.

Y a mis sobrinos Camila, Nelly, Samil y Noely para que este logro le sirva de inspiración y sepa que con esfuerzo todo es posible.

AGRADECIMIENTO

En primer lugar, deseo expresar mi profunda gratitud a Dios, por ser mi luz y mi guía y darme la salud y sabiduría necesarias para culminar esta etapa profesional.

A mis padres, por su sacrificio constante y su apoyo incondicional. Gracias por enseñarme el valor del esfuerzo y por ser mi refugio en los momentos de mayor presión; este título es tan suyo como mío.

De manera especial, a mi sobrina Camila, cuya alegría y ternura fueron el bálsamo perfecto en mis días de cansancio. Su presencia me recordó siempre la importancia de dejar un legado de superación.

Contenido

RESUMEN.....	13
ABSTRACT	14
INTRODUCCIÓN.....	15
CAPÍTULO I	17
PLANTEAMIENTO DEL PROBLEMA.....	17
1.1 Contextualización de la problemática.....	18
1.2 Descripción del problema.....	18
1.3 Formulación del problema.....	19
1.4 Objetivo general	19
1.5 Objetivos específicos	20
1.6 Justificación	20
1.7 Delimitación del estudio.....	21
CAPÍTULO II	23
MARCO TEÓRICO	23
2.1 Antecedentes.....	24
2.1.1 Antecedentes internacionales	24
2.1.2 Antecedentes nacionales	25
2.2 Bases teóricas.....	27
2.2.1 Transformación digital en instituciones de educación superior	27
2.2.2 Gestión de procesos administrativos y modelado de flujos	29
2.2.3 Sistemas de gestión de solicitudes (Ticketing Systems)	31
2.2.4 Arquitectura de software en aplicaciones web	33
2.2.5 Patrón Modelo–Vista–Controlador (MVC).....	35
2.2.6 Seguridad en sistemas web y control de acceso	37
2.2.7 Trazabilidad e indicadores de gestión.....	39
2.3 Bases legales.....	41
CAPÍTULO III	44
MARCO METODOLÓGICO.....	44
3.1 Enfoque y tipo de investigación.....	45
3.2 Diseño metodológico del proyecto.....	46

3.3 Fases de desarrollo del sistema	47
3.4 Técnicas de análisis utilizadas	48
3.5 Herramientas y entorno tecnológico.....	49
3.6 Criterios de validación del sistema	50
CAPÍTULO IV.....	53
DISEÑO E IMPLEMENTACIÓN DEL SISTEMA PROPUESTO	53
4.1 Análisis del proceso actual (Modelo AS-IS).....	54
4.2 Modelo propuesto (Modelo TO-BE).....	56
4.3 Arquitectura del sistema	59
4.3.1 Arquitectura general	59
4.3.2 Separación por capas.....	61
4.3.3 Seguridad estructural.....	63
4.3.4 Justificación de la arquitectura adoptada	65
4.4 Diseño de base de datos	66
4.4.1 Modelo de entidad-relación	66
4.4.2 Descripción de tablas principales	68
4.4.3 Integridad y relaciones	70
4.4.4 Justificación del modelo relacional adoptado	72
4.5 Diseño de interfaz	72
4.5.3 Diseño responsive.....	75
4.5.4 Justificación del enfoque adoptado	75
4.6 Implementación de módulos.....	76
4.6.1 Módulo de autenticación y control de acceso	76
4.6.2 Módulo de gestión de solicitudes.....	77
4.6.3 Módulo de asignación y gestión de estados	80
4.6.4 Módulo de historial y trazabilidad	81
4.6.5 Módulo de reportes e indicadores	81
4.6.6 Consideraciones técnicas generales	82
4.7 Pruebas funcionales y validación.....	83
4.7.1 Validación de autenticación y control de roles.....	83
4.7.2 Validación del flujo de solicitudes	84

4.7.3 Validación de historial y trazabilidad	85
4.7.4 Validaciones de seguridad básica	86
4.7.5 Resultados de validación	86
4.8 Resultados obtenidos	87
4.8.1 Comparación entre modelo AS-IS y modelo TO-BE	87
4.8.2 Cumplimiento de objetivos específicos	88
4.8.3 Impacto operativo	88
4.8.4 Valor técnico del prototipo	89
CAPÍTULO V	90
CONCLUSIONES Y RECOMENDACIONES	90
5.1 Conclusiones generales	91
5.2 Conclusiones específicas	91
5.3 Aportes del proyecto	93
5.4 Limitaciones del estudio	93
5.5 Recomendaciones	94
Referencias Bibliográficas	95

Lista de Tablas

Tabla 1. Comparación entre los Modelos AS-IS, TO-BE	87
---	----

Lista de Figuras

Figura 1. Flujo AS-IS del proceso actual de gestión de solicitudes en UNICYT.....	55
Figura 2. Modelo TO-BE del proceso de gestión de solicitudes estudiantiles en UNICYT.	57
Figura 3. Diagrama de arquitectura general Modelo–Vista–Controlador (MVC) implementada en el sistema.....	60
Figura 4. Carpetas del Proyecto	62
Figura 5. Implementación del token CSRF en formulario.....	65
Figura 6. Diagrama ER generado desde phpMyAdmin	67
Figura 7. Estructura de tablas en phpMyAdmin.	71
Figura 8. Menú Principal de los Roles en la Plataforma	75
Figura 9. Pantalla de Login o Inicio de Sesión de la Plataforma	77
Figura 10. Pantalla de Creación de Nueva solicitud	77
Figura 11. Pantalla de Listado de Solicitudes.....	78
Figura 12. Formulario de Nueva Solicitud.....	79
Figura 13. Detalle de Solicitudes Creadas.....	79
Figura 14. Detalle de Solicitud y Cambio de Estado de Solicitud	80
Figura 15. Historial de la Solicitud.....	81
Figura 16. Dashboard con métricas	82
Figura 17. Validación de control de acceso por roles.	84
Figura 18. <i>Pantalla Completa del detalle de una Solicitud</i>	85



REPÚBLICA DE PANAMÁ

UNIVERSIDAD INTERNACIONAL DE CIENCIA Y TECNOLOGÍA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN Y TECNOLOGÍA

**PLATAFORMA PARA LA GESTIÓN Y SEGUIMIENTO DE SOLICITUDES
ESTUDIANTILES EN LA UNIVERSIDAD UNICYT**

Tutora: José McLean

Autor: Oriel Cedeño

RESUMEN

La presente investigación trata el asunto de la administración de trámites académicos en la Universidad Internacional de Ciencia y Tecnología (UNICYT). Esta se distingue por su dependencia del correo electrónico como medio primordial para brindar atención, lo que produce restricciones en la trazabilidad, control y evaluación del proceso administrativo. La meta principal fue crear una plataforma web con el fin de gestionar y monitorear de forma estructurada las solicitudes de los estudiantes, con el propósito de mejorar la supervisión e implementación institucional de los trámites. Se basa la investigación en los principios de gestión de procesos organizacionales, diseño de aplicaciones web e ingeniería del software. La metodología utilizada fue de tipo aplicado con un enfoque tecnológico. Se organizó en etapas: análisis del procedimiento actual, diseño arquitectónico, modelado de base de datos, implementación del prototipo y validación a través de pruebas funcionales.

Como resultado, se creó un sistema operativo en PHP con arquitectura Modelo-Vista-Controlador (MVC) y base de datos MySQL. Este sistema posibilita la creación de tickets únicos, el rastreo por estados, el registro histórico de acciones y la visualización de indicadores esenciales para la gestión. Se llega a la conclusión de que el método sugerido convierte un proceso informal en un sistema estructurado, medible y trazable, lo cual ayuda a modernizar la administración de instituciones.

Palabras claves: Gestión de peticiones, plataforma en línea, arquitectura del tipo MVC, seguimiento y transformación digital.



REPUBLIC OF PANAMA

INTERNATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY

FACULTY OF COMPUTER SCIENCE AND TECHNOLOGY

**PLATFORM FOR THE MANAGEMENT AND MONITORING OF STUDENT
REQUESTS AT UNICYT UNIVERSITY**

Tutor: José McLean
Author: Oriel Cedeño

ABSTRACT

This research addresses the issue of managing academic procedures at the International University of Science and Technology (UNICYT). It is characterized by its reliance on email as the primary means of providing support, which results in limitations in traceability, control, and evaluation of the administrative process. The main goal was to create a web platform in order to manage and monitor student requests in a structured way, aiming to improve the institutional supervision and implementation of procedures. The research is based on the principles of organizational process management, web application design, and software engineering. The methodology used was applied with a technological approach. It was organized in stages: analysis of the current procedure, architectural design, database modeling, prototype implementation, and validation through functional testing. As a result, an operating system was created in PHP with a Model-View-Controller (MVC) architecture and a MySQL database. This system enables the creation of unique tickets, tracking by status, historical logging of actions, and visualization of key management indicators. It is concluded that the suggested method transforms an informal process into a structured, measurable, and traceable system, which helps modernize the management of institutions.

Keywords: request management, online platform, MVC architecture, monitoring and digital transformation.

INTRODUCCIÓN

La transformación digital se ha convertido en un factor estratégico para las instituciones de educación superior, ya que permite optimizar procesos internos y mejorar la calidad de los servicios ofrecidos a los estudiantes. En el caso de las universidades privadas, donde la experiencia del estudiante influye directamente en la percepción institucional, resulta necesario implementar soluciones tecnológicas que estructuren y controlen los procesos administrativos.

La Universidad Internacional de Ciencia y Tecnología (UNICYT) gestiona múltiples solicitudes estudiantiles relacionadas con trámites académicos y administrativos, tales como constancias, certificaciones, reclamos, procesos de matrícula, pagos y otras solicitudes especiales. Actualmente, el principal canal de atención es el correo electrónico institucional, complementado con interacciones presenciales y comunicaciones entre distintas áreas administrativas. Sin embargo, no existe un sistema centralizado que permita registrar, asignar, monitorear y auditar el ciclo completo de cada solicitud.

La dependencia del correo electrónico como mecanismo principal de gestión genera fragmentación de la información, ausencia de identificadores únicos por trámite y limitaciones en la trazabilidad del proceso. Además, dificulta la generación de indicadores de desempeño que permitan evaluar tiempos de respuesta, volumen de solicitudes y productividad por área. Desde la perspectiva de la ingeniería de software y la gestión de procesos, esta situación evidencia la necesidad de estructurar el flujo de atención mediante una plataforma especializada.

En respuesta a esta problemática, el presente Proyecto de Graduación propone el diseño y desarrollo de una plataforma web para la gestión y seguimiento de solicitudes estudiantiles en la UNICYT. La solución busca centralizar los trámites en un entorno digital único, incorporar asignación por roles, establecer estados de seguimiento, mantener un historial de solicitudes y generar reportes que faciliten el control y la toma de decisiones administrativas.

El proyecto se enmarca en una investigación aplicada orientada al análisis del proceso actual y al desarrollo de un prototipo funcional implementado en tecnología PHP. Se prioriza una arquitectura estructurada del sistema, la definición clara de módulos funcionales y la incorporación de criterios de seguridad, control de acceso y trazabilidad.

El documento se organiza en seis capítulos: el primero presenta el planteamiento del problema; el segundo desarrolla el marco teórico; el tercero describe la metodología adoptada; el cuarto expone el diseño y desarrollo técnico de la plataforma; y finalmente se presentan las conclusiones y recomendaciones derivadas del estudio.

CAPÍTULO I

PLANTEAMIENTO DEL PROBLEMA

1.1 Contextualización de la problemática

Las instituciones de educación superior gestionan de manera constante solicitudes académicas y administrativas realizadas por sus estudiantes. Estas solicitudes forman parte del funcionamiento operativo regular de la universidad y abarcan trámites como la emisión de constancias, certificaciones, procesos de matrícula, reclamos académicos, solicitudes de inclusión o retiro de asignaturas, validaciones de pagos, entre otros. La eficiencia en la atención de estos procesos influye directamente en la experiencia del estudiante y en la percepción institucional.

En la Universidad Internacional de Ciencia y Tecnología (UNICYT), el principal canal de recepción de estas solicitudes es el correo electrónico institucional. Aunque la universidad dispone de un sistema de matrícula y de la plataforma Moodle para la gestión académica, no cuenta con una herramienta tecnológica específica que permita administrar, registrar y dar seguimiento estructurado al ciclo completo de las solicitudes estudiantiles. Como consecuencia, la gestión de cada trámite depende del intercambio de correos electrónicos entre estudiantes y distintas áreas administrativas, lo que configura un proceso descentralizado y con bajo nivel de estandarización.

1.2 Descripción del problema

Desde la perspectiva de la ingeniería de software y la gestión de procesos organizacionales, el modelo actual presenta limitaciones operativas relevantes. No existe un identificador único o número de ticket que permita diferenciar formalmente cada solicitud. Tampoco se dispone de un sistema centralizado que permita al estudiante consultar el estado de su trámite en tiempo real. La asignación de responsables se realiza

mediante el reenvío manual de correos electrónicos entre áreas, lo que dificulta el control del flujo del proceso.

Asimismo, no se generan reportes automáticos que permitan medir tiempos de respuesta, volumen de solicitudes por categoría o desempeño por área administrativa. La trazabilidad del proceso depende del historial individual de correos electrónicos, lo que limita la capacidad de auditoría y control institucional. Estas condiciones incrementan la probabilidad de retrasos, duplicidad de información, pérdida de seguimiento y ausencia de indicadores consolidados de gestión.

El problema central no radica únicamente en el uso del correo electrónico como canal de recepción, sino en la inexistencia de un sistema estructurado que permita modelar, registrar y controlar el ciclo completo de atención bajo criterios de trazabilidad, asignación formal de responsabilidades y generación de indicadores de desempeño.

1.3 Formulación del problema

Ante la situación descrita, surge la siguiente interrogante:

¿De qué manera el diseño y desarrollo de una plataforma web de gestión de solicitudes estudiantiles puede mejorar la eficiencia operativa, la trazabilidad del proceso y el control institucional en la Universidad Internacional de Ciencia y Tecnología (UNICYT)?

1.4 Objetivo general

Desarrollar una plataforma web para la gestión y seguimiento de solicitudes estudiantiles en la Universidad Internacional de Ciencia y Tecnología (UNICYT), con el

propósito de centralizar los trámites, optimizar el flujo operativo y fortalecer los mecanismos de control y trazabilidad institucional.

1.5 Objetivos específicos

1. Analizar el flujo actual de gestión de solicitudes estudiantiles.
2. Identificar los requerimientos funcionales y no funcionales del sistema propuesto.
3. Diseñar la arquitectura técnica de la plataforma.
4. Modelar la base de datos y los componentes estructurales del sistema.
5. Implementar un prototipo funcional utilizando tecnología PHP.
6. Incorporar mecanismos de autenticación, roles y control de acceso.
7. Integrar módulos de seguimiento, historial y generación de reportes.
8. Validar funcionalmente la solución desarrollada.

1.6 Justificación

La implementación de una plataforma especializada permitirá reducir la dependencia del correo electrónico como único mecanismo de gestión de solicitudes y establecer un entorno digital estructurado que garantice la identificación única por trámite, la asignación clara de responsables, el seguimiento del estado en tiempo real, el registro histórico de interacciones y la generación de indicadores de gestión.

Desde la perspectiva técnica, el proyecto contribuye a la modernización de los procesos administrativos mediante la aplicación de principios de ingeniería de software orientados a la eficiencia, escalabilidad y control. Asimismo, se alinea con las tendencias actuales de transformación digital en educación superior y fortalece la capacidad institucional para ofrecer servicios administrativos más organizados, medibles y auditables.

En este sentido, el proyecto busca transformar un proceso operativo basado en comunicaciones dispersas en un sistema estructurado, trazable y controlado, sustentado en principios de arquitectura de software y gestión organizacional.

1.7 Delimitación del estudio

El presente proyecto se delimita al diseño y desarrollo de una plataforma web para la gestión y seguimiento de solicitudes estudiantiles en la Universidad Internacional de Ciencia y Tecnología (UNICYT). La propuesta se enfoca específicamente en la estructuración digital del flujo administrativo relacionado con la recepción, asignación, seguimiento y cierre de trámites académicos y administrativos realizados por los estudiantes.

El alcance del sistema comprende el modelado del proceso actual, la definición de requerimientos funcionales y no funcionales, el diseño de la arquitectura bajo el patrón Modelo-Vista-Controlador (MVC) y la implementación de un prototipo funcional desarrollado en PHP. La plataforma incluirá mecanismos de autenticación, control de acceso por roles, registro de solicitudes, estados de seguimiento, historial de interacciones y generación de reportes básicos.

El estudio no contempla la integración con sistemas institucionales existentes, tales como el sistema de matrícula o la plataforma Moodle, ni incluye la implementación en ambiente productivo institucional. Tampoco abarca análisis estadísticos basados en instrumentos de recolección de datos, ya que el enfoque del proyecto es técnico-aplicado y se centra en el diseño y desarrollo de la solución.

La validación del sistema se limitará a pruebas funcionales del prototipo desarrollado, con el propósito de verificar el cumplimiento de los requerimientos definidos.

CAPÍTULO II

MARCO TEÓRICO

2.1 Antecedentes

2.1.1 Antecedentes internacionales

La transformación digital en las instituciones de educación superior ha sido ampliamente abordada en estudios recientes, especialmente a partir de los cambios acelerados por la pandemia y la consolidación de modelos híbridos de gestión universitaria. De acuerdo con el Banco Interamericano de Desarrollo (BID), la digitalización universitaria no solo implica la virtualización del aprendizaje, sino también la modernización de los procesos administrativos mediante sistemas estructurados que permitan eficiencia, trazabilidad y generación de datos para la toma de decisiones (Lustosa Rosario et al., 2021).

En el mismo sentido, la Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO, 2021) destaca que la transformación digital en educación superior debe abarcar tanto la dimensión académica como la administrativa, promoviendo la implementación de plataformas tecnológicas que optimicen la gestión interna y fortalezcan la gobernanza institucional.

Asimismo, estudios publicados en revistas académicas sobre transformación digital organizacional indican que la adopción de sistemas de gestión estructurados mejora significativamente la eficiencia operativa y la coordinación interdepartamental. Verhoef et al. (2021) señalan que la transformación digital organizacional requiere rediseñar procesos internos bajo criterios de integración tecnológica y generación de valor institucional, lo cual resulta especialmente relevante en entornos complejos como las universidades.

Por otra parte, la Organización para la Cooperación y el Desarrollo Económicos (OCDE, 2020) resalta que las instituciones que implementan soluciones digitales integradas logran mayor transparencia en sus procesos administrativos y mejor capacidad para medir desempeño mediante indicadores estructurados. En el contexto universitario, esto se traduce en sistemas capaces de registrar solicitudes, asignar responsabilidades y generar métricas de gestión.

Desde la perspectiva de arquitectura de sistemas, el diseño de plataformas administrativas requiere estructuras modulares que faciliten la mantenibilidad y escalabilidad del software. Sommerville (2020) sostiene que la aplicación de principios de ingeniería de software y patrones arquitectónicos estructurados resulta esencial para garantizar la calidad y sostenibilidad de sistemas organizacionales.

En conjunto, estos antecedentes internacionales evidencian una tendencia clara hacia la digitalización integral de los procesos universitarios, respaldando la pertinencia del diseño y desarrollo de una plataforma web de gestión de solicitudes estudiantiles como una solución alineada con prácticas contemporáneas de modernización institucional.

2.1.2 Antecedentes nacionales

En Panamá, se han establecido instrumentos públicos que buscan modernizar la entrega de servicios y la administración institucional como medio para promover la transformación digital. La Agenda Digital Nacional 2021, impulsada por la Autoridad Nacional para la Innovación Gubernamental, es un referente esencial. Esta agenda establece directrices con el fin de progresar en digitalización, utilización estratégica de

datos y robustecimiento de habilidades tecnológicas a nivel nacional (Autoridad Nacional para la Innovación Gubernamental [AIG], 2021). Como la modernización de procesos administrativos necesita estructuras digitales para centralizar los trámites, estandarizar los flujos, registrar las interacciones y producir información que ayude a tomar decisiones, este enfoque es relevante para las instituciones de educación superior.

La modernización institucional y la tecnología se han incluido como componentes importantes en el Plan Estratégico de Educación 2025-2029 del Ministerio de Educación, con el fin de planificar el sistema educativo (Ministerio de Educación [MEDUCA], 2025). Si bien este plan está dirigido al sistema educativo nacional en su totalidad, crea un contexto institucional que subraya la importancia de potenciar las habilidades digitales y fomentar procesos más articulados y eficaces. Esto se ajusta con la tendencia de digitalización que también afecta a las universidades, principalmente en su gestión de los servicios para los alumnos.

Desde la perspectiva de la normativa, el país tiene un marco jurídico particular para manejar los datos personales. Los principios, derechos y deberes que se aplican al procesamiento de datos personales están definidos por la Ley N.º 81 de 2019; esta ley también incluye requerimientos vinculados a la seguridad y confidencialidad de la información (República de Panamá, 2019). Después, el Decreto Ejecutivo N.º 285 de 2021 regula la Ley 81, desarrollando detalles operativos sobre su implementación. Esto subraya la importancia de establecer controles técnicos y organizativos en plataformas digitales que manejen información personal (República de Panamá, 2021). Esto significa, en la práctica, que los sistemas universitarios que gestionan las solicitudes y

documentación de los estudiantes deben incluir el control de acceso, la trazabilidad y las medidas de seguridad conforme a la legislación actual.

Además, la Autoridad Nacional de Transparencia y Acceso a la Información ha emitido guías enfocadas en el cumplimiento de las normativas sobre protección de datos, que incluyen consejos acerca de principios, responsabilidades y acciones que se deben tomar en relación al manejo de información personal dentro de las organizaciones (Autoridad Nacional de Transparencia y Acceso a la Información [ANTAI], 2024). Esta clase de lineamientos brinda pautas valiosas para los proyectos de digitalización institucional, porque guían acerca de las prácticas adecuadas y del cumplimiento, sobre todo en los procedimientos administrativos donde se guardan y procesan datos de alumnos.

Finalmente, investigaciones académicas que incluyen evidencia empírica sobre el contexto panameño han mostrado brechas vinculadas al acceso y uso de tecnología, así como impactos diferenciados según el tipo de institución, lo cual refleja desafíos reales de adopción digital en el país (Cubilla-Bonnetier et al., 2023). Aunque estos estudios se enfocan en el ámbito educativo en general, su aporte contextual refuerza que los procesos de transformación digital requieren soluciones estructuradas, sostenibles y con adecuada gestión de información, especialmente en instituciones que buscan mejorar la experiencia del usuario y la eficiencia operativa.

2.2 Bases teóricas

2.2.1 Transformación digital en instituciones de educación superior

La transformación digital en las instituciones de educación superior (IES) se ha establecido como un proceso estratégico con el objetivo de cambiar la forma en que las

universidades manejan, organizan y ofrecen sus servicios administrativos y académicos. La transformación digital, a diferencia de la mera digitalización, que consiste en convertir procesos analógicos a formatos electrónicos, implica una completa reorganización de los modelos operativos de las instituciones a través del empleo sistemático de tecnologías de información (Verhoef et al., 2021).

Este proceso no se restringe solamente a la inclusión de plataformas de enseñanza virtual en el ámbito universitario, sino que incluye, además, la actualización de procesos administrativos internos. La mejora de la gestión institucional a través de la creación de datos, la optimización de flujos administrativos y la integración de sistemas deben formar parte de la transformación digital en educación superior, según indica el Banco Interamericano de Desarrollo (Lustosa Rosario et al., 2021). Esto significa que se debe transitar de esquemas de comunicación desarticulados a sistemas organizados que faciliten la trazabilidad, la supervisión y el análisis del rendimiento.

La OCDE (2020) señala que las instituciones que implementan soluciones digitales integradas obtienen una mayor capacidad de reacción ante lo que los estudiantes exigen, así como una eficiencia más alta en sus operaciones. Concretamente, la puesta en marcha de plataformas digitales para la gestión simplifica la uniformización de procesos, la reducción del tiempo dedicado a la atención y el fortalecimiento de datos en contextos centralizados.

Desde un punto de vista organizacional, la transformación digital significa reinventar los procesos de acuerdo con estándares de integración tecnológica y creación de valor a nivel institucional. Verhoef et al. (2021) afirman que es necesario realizar modificaciones estructurales en la forma en que se coordinan las funciones internas para

llevar a cabo una digitalización efectiva; esto es particularmente importante en instituciones complejas como las universidades, donde diferentes departamentos colaboran en la administración de los servicios para el alumnado.

Además, según la UNESCO (2021), la gobernanza universitaria en la era digital tiene que basarse en sistemas que posibiliten el monitoreo y evaluación del rendimiento institucional a través de indicadores precisos y datos verificables. Por lo tanto, la habilidad para registrar, monitorear y evaluar procesos administrativos se vuelve un componente esencial para mejorar la transparencia y la calidad del servicio.

En este marco, la implementación de plataformas digitales para la gestión de solicitudes estudiantiles se alinea con las tendencias internacionales de transformación digital en educación superior. La centralización de trámites, la asignación estructurada de responsabilidades y la generación de métricas operativas representan manifestaciones concretas de un proceso de modernización institucional sustentado en principios tecnológicos y organizacionales.

2.2.2 Gestión de procesos administrativos y modelado de flujos

La administración de procesos administrativos es un elemento fundamental en el funcionamiento de las instituciones de educación superior, porque facilita la organización, coordinación y supervisión de las acciones que apoyan la entrega de servicios académicos. Según Dumas et al. (2018), un proceso organizacional es un grupo de actividades vinculadas que convierten insumos en resultados a través de una serie lógica de acciones con el propósito de alcanzar un objetivo concreto. A pesar de que esta definición proviene del campo de la gestión empresarial, su uso en entornos

universitarios es especialmente importante para el estudio de los servicios y trámites estudiantiles.

La gestión por procesos se vuelve más relevante en el contexto de la digitalización, ya que es necesario detectar, registrar y mejorar los flujos de trabajo existentes. Hammer (2015) sostiene que no es posible conseguir la mejora organizacional solamente a través de la automatización tecnológica; es necesario entender el proceso actual (modelo AS-IS) y crear un proceso mejorado (modelo TO-BE) que suprima redundancias, disminuya los plazos y defina responsabilidades claras. A pesar de que esta visión es anterior a la etapa más reciente, sigue siendo un referente esencial en las investigaciones actuales sobre el rediseño organizacional.

El modelado de procesos facilita la representación gráfica y conceptual del flujo de actividades, los participantes implicados y los puntos de decisión en un sistema. Esta práctica ayuda a detectar en entornos digitales dependencias innecesarias, tareas repetidas y cuellos de botella. Según vom Brocke y Mendling (2018), la perspectiva basada en procesos brinda una visión organizada que optimiza la coordinación entre departamentos y posibilita la creación de indicadores de rendimiento acordes a las metas organizacionales.

En el contexto universitario, la falta de un sistema formal para gestionar procesos administrativos puede provocar que la información esté fragmentada, que se dependa de comunicaciones informales y que sea difícil medir los resultados. Con el fin de formalizar responsabilidades, fijar estados concretos y conservar un registro histórico de cada interacción, la estandarización del flujo de atención a través de plataformas digitales fortalece la trazabilidad institucional (OECD, 2020).

El rediseño de procesos administrativos a través de sistemas web, desde un punto de vista técnico, necesita que el flujo operativo se traduzca en elementos funcionales del software. Esto significa reconocer eventos iniciales, actividades intermedias, decisiones, roles de los participantes y estados finales. Después, estos se convierten en requisitos funcionales, casos de uso y esquemas de base de datos. Por lo tanto, el modelado de procesos no es solamente una tarea organizativa, sino que representa además un periodo esencial en la ingeniería de requisitos y en el diseño de sistemas.

En este escenario, el estudio del procedimiento de gestión de solicitudes estudiantiles en curso y la propuesta de un flujo ordenado son pasos cruciales para el desarrollo de la plataforma web que se propone en este proyecto. Esto posibilita pasar de un modelo operativo descentralizado a uno que es controlado, organizado y cuantificable.

2.2.3 Sistemas de gestión de solicitudes (Ticketing Systems)

Los sistemas de gestión de solicitudes o *ticketing systems* son plataformas orientadas a controlar, de forma estructurada, el ciclo de vida de requerimientos que ingresan a una organización. En lugar de manejar trámites como conversaciones dispersas (por ejemplo, hilos de correos), un ticketing system centraliza cada solicitud en un registro único, permitiendo su clasificación, asignación a responsables, seguimiento por estados, y cierre con evidencia. Este enfoque responde a principios de gestión de servicios, donde la trazabilidad y el control del trabajo son condiciones necesarias para garantizar continuidad operativa y mejora de la calidad del servicio (AXELOS, 2019).

En ITIL 4, desde la gestión de servicios, propone prácticas concretas (por ejemplo service desk, incident management y service request management) que exigen que las

solicitudes se manejen a través de flujos medibles y controlados. Esto normalmente se lleva a cabo mediante herramientas que documentan los casos, organizan la atención y retienen el historial. En este contexto, un sistema de tickets sirve como una herramienta tecnológica para implementar esas prácticas, ya que transforma cada interacción en un acontecimiento que se puede rastrear dentro de un proceso evaluable y gestionable (AXELOS, 2019).

Simultáneamente, la norma ISO/IEC 20000-1:2018 (estándar global para sistemas de gestión de servicios) impone condiciones que permiten a una organización planificar, ofrecer, controlar, evaluar y optimizar sus servicios a través de un sistema formal. Cuando las solicitudes se registran y gestionan en un sistema que posibilita evidencias, indicadores y control de manera constante, es más sencillo aplicar esta lógica, que requiere la habilidad de monitorear, medir y revisar el sistema de gestión de servicios (International Organization for Standardization [ISO], 2018). En consecuencia, un sistema de tickets proporciona la base instrumental para recopilar datos operacionales y convertirlos en información útil para la administración del servicio.

La trazabilidad es un componente esencial en estos sistemas. No es suficiente con "recibir y responder": es fundamental poder reconstruir lo que pasó con la solicitud (quién se encargó de ella, cuándo varió su estado, qué información se pidió, qué evidencias se anexaron y cuál fue la resolución). Esta trazabilidad permite la mejora continua, el control interno y la auditoría; también posibilita la creación de indicadores como el tiempo de primera respuesta, el tiempo de resolución y la carga laboral por área o categoría (ISO, 2018).

Desde el punto de vista funcional, muchos sistemas de tickets incluyen tareas de clasificación y enrutamiento (asignación automática o semiautomática al equipo apropiado), ya que la gran cantidad de solicitudes puede hacer que el triage manual sea ineficiente. Según la literatura académica más reciente, los tickets suelen necesitar una clasificación temática para optimizar el direccionamiento y acelerar la atención en situaciones reales de soporte; incluso se han investigado métodos de automatización fundamentados en aprendizaje automático para ayudar con este proceso (Zangari et al., 2023). Esto evidencia que la estructura de un ticket (categoría, prioridad, estado, responsable, historial) no es meramente "una idea de sistema", sino una modalidad de operación que se ha investigado en escenarios de servicio.

Por último, la evidencia aplicada también respalda que instaurar un sistema de tickets en contextos pequeños o específicos posibilita la organización de la atención, la reducción de pérdidas informativas y la creación de registros operativos valiosos para análisis y gestión. Por ejemplo, se han publicado investigaciones de caso acerca de cómo se implementaron sistemas de ticketing en centros técnicos, lo cual demostró ventajas en el control del trabajo y en la disposición del servicio (Yamaoka, 2019). Estos principios se reflejan en una mejora directa en el ámbito universitario: cada proceso de los estudiantes pasa a ser un caso que puede rastrearse, medirse y gestionarse por roles, lo cual disminuye la dependencia de correos electrónicos dispersos y robustece el control institucional.

2.2.4 Arquitectura de software en aplicaciones web

La arquitectura de software detalla la estructura básica de un sistema, lo que incluye sus elementos, las conexiones entre ellos y los principios que orientan su

evolución y diseño. La arquitectura, de acuerdo con Bass, Clements y Kazman (2022), es la suma de las decisiones estructurales significativas que establecen cómo un sistema cumple con características de calidad como la seguridad, el rendimiento y la mantenibilidad. Esta dimensión es crucial en las aplicaciones web institucionales, porque una arquitectura inadecuadamente establecida puede poner en riesgo la habilidad del sistema de evolucionar de forma controlada.

La arquitectura, desde un punto de vista ingenieril, no solo se ocupa de organizar el código, sino que también crea un marco para manejar la complejidad del sistema. Según Richards y Ford (2020), una arquitectura apropiada facilita la toma de decisiones conscientes sobre acoplamiento, cohesión y distribución de responsabilidades, lo que impide el crecimiento desorganizado o improvisado del sistema. Una base arquitectónica bien definida disminuye los riesgos técnicos y simplifica el mantenimiento en entornos administrativos, donde los sistemas suelen añadir nuevos módulos, reglas de negocio y roles a lo largo del tiempo.

En el terreno normativo, el estándar ISO/IEC/IEEE 42010:2022 establece los lineamientos para describir arquitecturas de software y sistemas. Según este estándar, la arquitectura tiene que ser documentada con vistas estructuradas que hagan posible su análisis y evaluación (ISO/IEC/IEEE, 2022). En proyectos institucionales, la claridad en la estructura del sistema y el seguimiento de las decisiones técnicas son factores esenciales para que sean sostenibles; por ello, este enfoque es importante.

Además, se puede valorar la calidad del producto software de acuerdo con el modelo ISO/IEC 25010, que establece atributos como la confiabilidad, la eficiencia en el rendimiento, la mantenibilidad, la usabilidad y la seguridad (ISO/IEC, 2023). Estas

categorías brindan un marco teórico para fundamentar decisiones arquitectónicas vinculadas a la separación en capas, validación de datos, control de acceso y registro de actividades, sobre todo en sistemas que manejan información delicada.

La separación por capas o el modelo MVC (Modelo-Vista-Controlador) es uno de los métodos más empleados en aplicaciones web. Este método fomenta la distribución de responsabilidades entre el control de flujo, la presentación y la lógica del negocio. Este principio promueve la mantenibilidad y disminuye el acoplamiento entre componentes, de modo que las modificaciones de la interfaz no impacten directamente en la lógica interna del sistema. La implementación de una arquitectura modular es especialmente relevante en sistemas administrativos, que necesitan unificar funciones como la autenticación, el manejo de solicitudes, el historial y los informes dentro de una estructura lógica.

En resumen, la arquitectura de software en las aplicaciones web es el marco técnico que posibilita transformar un procedimiento organizativo en una solución estructurada y evaluable. La decisión de implementar una arquitectura organizada para la creación del prototipo de este proyecto se basa en el apoyo teórico proveniente de atributos de calidad, normas internacionales y perspectivas arquitectónicas consolidadas.

2.2.5 Patrón Modelo–Vista–Controlador (MVC)

MVC, que es un patrón modelo-vista-controlador, es una arquitectura de diseño muy empleada para crear aplicaciones web y dividir las responsabilidades en el sistema. La aplicación se divide en tres partes esenciales según este patrón: la Vista, que es responsable de presentar la información al usuario; el Controlador, que funciona como

intermediario coordinando las solicitudes y la interacción entre vista y modelo; y el Modelo, que gestiona los datos y se encarga de la lógica del negocio (Gamma et al., 1995; Richards & Ford, 2020).

A pesar de que el patrón MVC fue establecido inicialmente en el contexto de la programación orientada a objetos, su uso en entornos web se ha afianzado como una práctica estandarizada para asegurar la mantenibilidad y modularidad. Según Richards y Ford (2020), los patrones arquitectónicos no solamente estructuran el código, sino que también disminuyen el acoplamiento e impulsan la evolución del sistema a lo largo de un periodo extenso. Esta división estructural es particularmente importante en aplicaciones institucionales, donde se pueden agregar nuevas funcionalidades y módulos a lo largo del tiempo.

Desde el punto de vista de la ingeniería de software, un principio fundamental para optimizar la calidad del sistema es la división de responsabilidades. La modificabilidad y la mantenibilidad son características fundamentales de un software, según lo establece la norma ISO/IEC 25010 (2023). Al posibilitar que las alteraciones en la interfaz no tengan un impacto en la lógica del negocio y que la actualización de la base de datos no exija modificar la presentación, el patrón MVC colabora directamente con estas características.

La implementación manual del patrón MVC, en el marco concreto del desarrollo en PHP, posibilita estructurar el proyecto sin tener que recurrir a frameworks externos y conservar un control absoluto sobre la organización lógica y de código. De acuerdo con Sommerville (2020), la arquitectura tiene que ajustarse a las exigencias establecidas y al

contexto del proyecto, dándole prioridad a la coherencia técnica y a la claridad estructural por encima de adoptar automáticamente herramientas sofisticadas.

De la misma manera, dado que cada componente tiene una función bien definida dentro del sistema, el empleo de MVC hace más fácil la incorporación de mecanismos para validar datos, rastrear y controlar el acceso. La vista puede restringirse a la representación de formularios y estados; el modelo puede unificar las reglas de negocio y las consultas seguras a la base de datos; y el controlador tiene la capacidad de gestionar permisos y autenticación. Esta organización disminuye los peligros relacionados con cambios futuros y hace que el código sea más legible.

Así, la elección del patrón Modelo-Vista-Controlador como fundamento arquitectónico del proyecto actual está respaldada por principios sólidos de ingeniería de software, que buscan asegurar modularidad, mantenibilidad y control estructural en el proceso de desarrollo de la plataforma para gestionar las solicitudes de los estudiantes.

2.2.6 Seguridad en sistemas web y control de acceso

La seguridad constituye uno de los atributos fundamentales en el diseño de sistemas web institucionales, especialmente cuando estos gestionan información personal y académica. En entornos universitarios, las plataformas administrativas procesan datos sensibles relacionados con identidad, historial académico y trámites oficiales, lo que exige mecanismos de protección adecuados para garantizar confidencialidad, integridad y disponibilidad de la información.

Desde el punto de vista de la calidad del software, el modelo ISO/IEC 25010 (2023) considera la seguridad como una propiedad fundamental del producto. Esta incluye subatributos como la integridad, la autenticidad, la confidencialidad y la

responsabilidad. Es necesario incorporar estos elementos desde la etapa de diseño arquitectónico, para que la seguridad no se considere como un componente adicional después.

La autenticación de usuarios es uno de los procesos esenciales en las aplicaciones web, que consiste en la verificación de la identidad del usuario que intenta acceder al sistema. Adicionalmente, la autorización establece qué acciones cada usuario tiene permitido ejecutar en el sistema, generalmente a través de un esquema de permisos y roles. La implementación apropiada del control de acceso es esencial para evitar las vulnerabilidades relacionadas con la exposición de datos o el aumento indebido de privilegios, de acuerdo con OWASP (2021).

El diseño fundamentado en roles habilita la división de responsabilidades dentro de la aplicación, otorgando permisos particulares a cada perfil, ya sea estudiante, administrador o personal administrativo. Esta organización no solo optimiza la estructura interna del sistema, sino que además solidifica el control institucional y la gobernanza al limitar las operaciones delicadas a los usuarios autorizados.

En Panamá, la Ley N.º 81 de 2019 es la que regula la seguridad en el tratamiento de datos personales, y se basa en principios como el consentimiento, la legalidad, el propósito, la proporción y la seguridad en el manejo de los datos personales. La normativa requiere que las entidades adopten acciones organizativas y técnicas para salvaguardar la información que tienen bajo su custodia. El Decreto Ejecutivo N.º 285 de 2021 también regula la ley mencionada, estableciendo reglas sobre el acceso, la protección y el almacenamiento de información en sistemas digitales.

Por lo tanto, cualquier plataforma web institucional que maneje información de estudiantes tiene que incluir controles de autorización por roles, autenticación, validación de entradas, un manejo seguro de sesiones y mecanismos para registrar actividades (bitácora). Estas prácticas no solo están alineadas con los estándares globales de ingeniería de software, sino que también aseguran la conformidad con las regulaciones en el contexto legal de Panamá.

El diseño del sistema propuesto, que incorpora estos principios, ayuda a garantizar que la plataforma de gestión de solicitudes estudiantiles sea no solo efectiva, sino además segura, fiable y acorde con las normativas legales actuales.

2.2.7 Trazabilidad e indicadores de gestión

La trazabilidad en sistemas organizacionales se refiere a la habilidad de documentar y reconstruir el trayecto total de un proceso, desde su comienzo hasta su término, identificando los resultados, los eventos y las personas responsables. En ambientes digitales, esta habilidad depende de la disponibilidad de registros estructurados que registren cada movimiento realizado en un trámite. La norma ISO 9001:2015, según la perspectiva de gestión de calidad, dicta que las organizaciones deben preservar documentación adecuada para asegurar el control sobre los procesos y ofrecer prueba de su desempeño (International Organization for Standardization [ISO], 2015). Este principio se manifiesta en las aplicaciones web institucionales mediante la inclusión de bitácoras, historiales y estados de seguimiento que faciliten la comprobación del progreso de cada solicitud.

La trazabilidad permite un análisis sistemático del rendimiento de la organización, además de registrar los eventos. Los datos operativos pueden convertirse en información

estratégica cuando los procesos se digitalizan y se organizan en sistemas informáticos. Según Davenport y Harris (2017), las entidades que incorporan análisis en sus procedimientos consiguen transformar datos transaccionales en saber para tomar decisiones fundamentadas en evidencia. Esto supone, en el marco administrativo de una universidad, calcular indicadores como la carga laboral por área, la tasa de resolución, el volumen de solicitudes por categoría y los tiempos medios de respuesta.

La digitalización institucional en la educación superior ha mostrado que puede mejorar la habilidad de monitoreo y evaluación interna. La OECD (2021) indica que la urgencia de contar con datos estructurados para apoyar las decisiones organizacionales y los procedimientos de rendición de cuentas ha aumentado a medida que se acelera la transformación digital en las universidades. En este contexto, los sistemas que combinan métricas y trazabilidad hacen posible detectar cuellos de botella, perfeccionar los flujos laborales y aumentar la eficiencia en términos operativos.

La experiencia del usuario también se ve afectada de manera directa por la trazabilidad digital. La transparencia del proceso mejora y la incertidumbre disminuye cuando se le permite al solicitante ver el estado actualizado de su trámite y el registro de las acciones ejecutadas. Desde el punto de vista de la gestión, la unión del registro estructurado y la creación de indicadores hace que el sistema sea una herramienta no solo operativa sino también estratégica, ya que brinda evidencias medibles del rendimiento institucional.

Por lo tanto, la incorporación de indicadores de gestión y mecanismos de trazabilidad en la plataforma que se propone está basada en principios establecidos de gobernanza fundamentada en datos, analítica organizacional y gestión de calidad

(Davenport & Harris, 2017; ISO, 2015; OECD, 2021). Estos elementos posibilitan que se respalde de manera técnica la inclusión de reportes, bitácoras e historial como partes fundamentales del sistema.

2.3 Bases legales

En Panamá, la legislación específica para la protección de datos personales establece los derechos, principios, obligaciones y procedimientos que se aplican a la gestión de información personal en medios digitales o físicos. La norma fundamental que regula este asunto en la nación es la Ley N.º 81 de Protección de Datos Personales, promulgada el 26 de marzo del año 2019. La República de Panamá aprobó esta ley con el propósito de asegurar que los datos personales sean tratados conforme a principios como seguridad, veracidad, confidencialidad, legalidad, finalidad y proporcionalidad. Además, busca garantizar que se respeten los derechos esenciales de las personas al procesar su información personal (Ley N.º 81 de 2019; consulte también Protección de datos personales en Panamá, 2024).

Según la Ley N.º 81 de 2019, los datos personales son aquellos que permiten identificar a una persona natural o que identifican a esta, como su dirección, nombres, cédula y correos electrónicos, entre otros datos vinculados con la vida privada (Protección de datos personales en Panamá, 2024). Esta legislación también establece que la gestión de datos sensibles —los cuales tienen el potencial de revelar detalles más personales o reservados del individuo— requiere medidas de protección más rigurosas

y, en algunas circunstancias, la autorización explícita del propietario para su tratamiento, divulgación o transferencia (Ley N.º 81 de 2019; Data protection policy, 2022).

El Decreto Ejecutivo N.º 285 del 28 de mayo de 2021 fue el que reglamentó la Ley N.º 81, la cual se aplica en Panamá y cuyas obligaciones y procedimientos son definidos por este decreto para clarificar su alcance. El Decreto Ejecutivo 285 establece los requisitos organizativos y técnicos para las personas a cargo del tratamiento de datos personales. También define el proceso para ejercer los derechos de los dueños de esos datos y otorga la facultad a la Autoridad Nacional de Transparencia y Acceso a la Información (ANTAI) como entidad supervisora responsable de garantizar que se cumpla con la normativa (Decreto Ejecutivo N° 285 de 2021; Reglamentación de Ley 81 sobre Protección de Datos Personales, 2021).

La legislación panameña en cuanto a la protección de datos personales es de carácter general y se aplica obligatoriamente a toda persona natural o jurídica, ya sea pública o privada, que maneje información personal dentro del territorio panameño o posea bases de datos en el país, incluso si los titulares de los datos son extranjeros (RSM Panamá, 2022; Law 81 of 2019 overview, 2021). Este método territorial y material garantiza que los datos personales sean tratados bajo cualquier modalidad, sea física o electrónica, con base en criterios de control, transparencia y seguridad. Esto es así sin importar la nacionalidad de la persona o el tipo de entidad que maneje los datos.

El Oficial de Protección de Datos Personales es el encargado de organizar la implementación de medidas de seguridad y el ejercicio de derechos por parte de los titulares, según se establece en el mismo decreto y las regulaciones complementarias (Reglamentación Ley 81 sobre Protección de Datos Personales, 2021). Este rol es

esencial para asegurar que las entidades implementen acciones técnicas y organizativas apropiadas, como el cifrado, la gestión de acceso, el registro de actividades y políticas internas, para cumplir con sus obligaciones legales en cuanto a la protección de datos.

La importancia de este marco legal para sistemas web que manejan información de estudiantes, como la plataforma sugerida en este proyecto, estriba en que impone la adopción de acciones destinadas a garantizar la confidencialidad, la seguridad y la trazabilidad de los datos personales que se almacenan y procesan. La implementación de controles de acceso, permisos claramente establecidos, registros de actividad y mecanismos de autenticación seguros es una demostración tangible de cumplimiento normativo, asegurando que el procesamiento de datos en la plataforma no solo sea funcionalmente correcto sino también conforme con la Ley N.º 81 del 2019 y su reglamento por Decreto Ejecutivo N.º 285 del 2021.

CAPÍTULO III

MARCO METODOLÓGICO

3.1 Enfoque y tipo de investigación

El presente proyecto se enmarca en un enfoque aplicado, orientado a la resolución de un problema específico identificado en el contexto administrativo de la Universidad Internacional de Ciencia y Tecnología (UNICYT). La investigación aplicada se caracteriza por la utilización del conocimiento teórico para diseñar soluciones prácticas a situaciones concretas, en este caso, la optimización del proceso de gestión de solicitudes estudiantiles mediante el desarrollo de una plataforma web especializada.

En cuanto a su tipo, el estudio es de carácter descriptivo–propositivo. Es descriptivo porque analiza y documenta el funcionamiento actual del proceso de atención de solicitudes estudiantiles, identificando sus características operativas, limitaciones estructurales y áreas de mejora. Es propositivo porque, a partir de dicho análisis, se diseña y desarrolla una solución tecnológica que busca reorganizar el flujo administrativo bajo criterios de trazabilidad, control y eficiencia.

El diseño metodológico es no experimental, ya que no se manipulan variables ni se aplican tratamientos controlados sobre una población específica. El proyecto se centra en el análisis del proceso existente y en la construcción de un prototipo funcional que modela una propuesta de mejora estructural.

Desde la perspectiva de la ingeniería de software, el proyecto adopta un enfoque de desarrollo tecnológico incremental, en el cual se parte del estudio del proceso actual para traducirlo en requerimientos funcionales y no funcionales, que posteriormente son transformados en componentes estructurales del sistema.

3.2 Diseño metodológico del proyecto

La metodología que se aplica en este proyecto está basada en un enfoque estructurado de desarrollo de software, enfocado en crear una solución tecnológica a partir del estudio de un proceso administrativo ya existente. Este estudio se enfoca en la ingeniería de sistemas, a diferencia de las investigaciones experimentales o estadísticas. Su objetivo principal es convertir un flujo operacional en un modelo digital estructurado.

El proceso metodológico se llevó a cabo de forma secuencial y progresiva, comenzando con el análisis del flujo actual de gestión de solicitudes por parte de los estudiantes y terminando con la puesta en marcha de un prototipo práctico. Esta perspectiva permitió entender las interacciones entre los actores, detectar puntos críticos y convertir estos elementos en exigencias técnicas específicas.

Desde una perspectiva técnica, el diseño del proyecto se dividió en etapas concretas que incluyen: evaluación del proceso actual, especificación de requisitos, diseño arquitectónico, modelado de base de datos, desarrollo del sistema y validación funcional. Cada etapa alimenta a la siguiente, asegurando que el problema detectado y la solución elaborada sean consistentes.

La metodología adoptada sigue los principios de la ingeniería de software, que priorizan la separación de tareas, la organización modular y la coordinación entre los requisitos y los elementos del sistema. Así, el proyecto no se reduce a la programación de una aplicación; incluye también un esquema lógico que posibilita justificar cada módulo implementado desde un punto de vista técnico.

Este enfoque asegura que la solución sugerida no sea solo una herramienta tecnológica, sino un modelo estructurado para optimizar el proceso administrativo, basado en análisis previos y desarrollo sistemático.

3.3 Fases de desarrollo del sistema

El desarrollo del sistema propuesto se estructuró en fases organizadas de manera secuencial, con el propósito de garantizar coherencia entre el análisis del problema identificado y la solución tecnológica implementada. Estas fases permitieron transformar un proceso administrativo informal en un modelo digital estructurado.

La fase inicial fue el análisis del proceso vigente (modelo AS-IS). En esta fase, se analizó cómo se manejan hoy en día las solicitudes de los estudiantes, reconociendo los canales empleados, la interacción entre áreas administrativas y los puntos críticos del flujo operativo. Este análisis posibilitó entender la falta de mecanismos formales para identificar de manera única, trazar y generar indicadores.

En la segunda fase se definieron los requisitos del sistema. Los requisitos funcionales y no funcionales que la plataforma debía satisfacer se definieron a partir del análisis anterior. La generación de informes, la creación de solicitudes, el seguimiento por estados, la asignación por roles y el historial de interacciones fueron algunos de los requerimientos funcionales. Con respecto a los requisitos no funcionales, se establecieron parámetros relacionados con la seguridad, el control de acceso, la organización modular y el mantenimiento.

La fase número tres consistió en el diseño del sistema a nivel arquitectónico. Durante esta fase, se estableció la estructura principal de la aplicación, utilizando una

arquitectura que sigue el patrón MVC (Modelo-Vista-Controlador), con una distinción nítida entre la lógica del negocio, el control de flujo y la presentación. Igualmente, la base de datos fue modelada teniendo en cuenta las entidades, relaciones y restricciones requeridas para asegurar la trazabilidad y la integridad.

La puesta en práctica del prototipo funcional fue la cuarta etapa. El desarrollo se llevó a cabo en PHP puro, con una arquitectura MVC manual y empleando un ambiente de servidor local y una base de datos MySQL. En esta fase se desarrollaron los módulos establecidos en los requisitos, que abarcan gestión de roles, autenticación, seguimiento por estados, generación de informes básicos y registro de solicitudes.

Por último, la validación funcional del sistema fue la quinta fase. En esta fase se realizaron exámenes con el objetivo de comprobar la correcta operación de los módulos implementados, el flujo operativo coherente, la apropiada asignación de permisos por rol y la producción de datos organizados para indicadores de gestión.

La segmentación en etapas posibilitó que el problema detectado, el diseño conceptual y la solución elaborada permanecieran alineados, asegurando así la coherencia técnica a lo largo del proceso metodológico.

3.4 Técnicas de análisis utilizadas

Se utilizaron métodos de análisis con el objetivo de entender el actual flujo operativo en la gestión de solicitudes estudiantiles y convertirlo en un modelo estructurado que se puede implementar digitalmente para desarrollar este proyecto.

La técnica principal utilizada fue la observación técnica del proceso existente. Esta observación permitió identificar la secuencia de actividades involucradas en la recepción, gestión y resolución de solicitudes, así como los actores que intervienen en cada etapa. El análisis se centró en la forma en que las solicitudes son recibidas, reenviadas, gestionadas y cerradas, evidenciando la ausencia de mecanismos formales de identificación, seguimiento y registro estructurado.

Además, se efectuó un estudio de los procesos administrativos con la finalidad de estructurar el flujo operativo. Este análisis permitió detectar los puntos de partida, las decisiones intermedias, las responsabilidades asignadas y los estados finales del proceso. La propuesta de un flujo mejorado, que está en línea con los principios de trazabilidad y control, se desarrolló a partir de esta evaluación.

La finalidad del proyecto fue modelar y mejorar un proceso operativo a través de una solución tecnológica estructurada, por lo que las técnicas empleadas no incluyeron la manipulación de variables ni la aplicación de herramientas estadísticas. Por lo tanto, el análisis realizado permitió la creación de una base técnica firme para determinar los requerimientos y diseñar la arquitectura del sistema.

3.5 Herramientas y entorno tecnológico

La creación del prototipo funcional se llevó a cabo mediante el uso de tecnologías web de código abierto, que fueron escogidas por su compatibilidad, accesibilidad y sencillez para ser implementadas en contextos institucionales.

PHP fue el lenguaje principal de desarrollo, que se implementó utilizando una arquitectura manual basada en el patrón Modelo-Vista-Controlador (MVC). Esta

selección posibilitó la estructuración del proyecto con una distinción nítida entre la lógica de negocio, el control del flujo de solicitudes y la capa de presentación, lo que asegura que el código sea mantenible y esté organizado por módulos.

MySQL fue el sistema de gestión de bases de datos relacional que se utilizó para manejar los datos, ya que es estable, cuenta con una extensa documentación y es compatible con las aplicaciones web que se desarrollan en PHP. Se modeló la base de datos tomando en cuenta entidades vinculadas a usuarios, roles, registros históricos, solicitudes y estados, garantizando así la trazabilidad y la integridad referencial.

Utilizando XAMPP, que incluye Apache, PHP y MySQL en un entorno integrado que permite la ejecución y las pruebas del sistema, se configuró el ambiente de desarrollo a través de un servidor local. Este ambiente posibilitó la simulación de condiciones reales de operación antes de que fuera posible su implementación en un servidor institucional.

Para la interfaz de usuario se utilizaron tecnologías estándar, como HTML5 y CSS3, junto con estilos estructurados que posibilitan una presentación clara y eficiente. Con el fin de ser sencillo, claro y navegable, se diseñó la estructura visual.

El empleo de herramientas abiertas y ampliamente utilizadas asegura que la solución sugerida sea ejecutable sin la necesidad de licencias propias, lo que disminuye gastos y simplifica su adaptación futura en el contexto institucional.

3.6 Criterios de validación del sistema

Se llevaron a cabo pruebas funcionales para comprobar que los requisitos establecidos en la etapa de análisis se cumplieran, lo que permitió validar el sistema desarrollado. El proyecto se enfoca en desarrollar un prototipo funcional, por lo que la

validación no estuvo dirigida a las mediciones estadísticas, sino a la verificación técnica del desempeño del sistema ante los escenarios anticipados.

En primer lugar, se llevaron a cabo ensayos para crear y gestionar solicitudes, asegurándose de que el sistema produjera un identificador único por trámite y que cada solicitud pudiera avanzar adecuadamente a través de los estados establecidos. Se verificó la coherencia del flujo de operaciones, garantizando que las transiciones entre estados fueran el resultado de acciones válidas según el rol del usuario.

En segundo lugar, se llevaron a cabo pruebas de control de acceso basado en roles y de autenticación. Estas pruebas validaron que cada perfil (administrador, personal administrativo, estudiante) tuviera acceso únicamente a funciones permitidas, evitando así el acceso a información confidencial o alteraciones inapropiadas.

También se verificó la persistencia de los datos en la base de datos relacional, asegurando que los registros fueran íntegros, las relaciones entre entidades correctas y el historial de interacciones almacenado apropiadamente. Esto posibilitó que cada solicitud en el sistema tuviera una trazabilidad total.

Por último, se llevaron a cabo pruebas para crear informes básicos, comprobando que el sistema fuera capaz de consolidar información sobre el volumen de solicitudes, los estados y la actividad por usuario. Estas pruebas corroboraron que el sistema es factible como instrumento de asistencia a la gestión administrativa.

Los criterios de validación implementados aseguran que el prototipo funcional satisface las metas fijadas y ofrece una respuesta lógica al problema detectado, lo cual

refuerza la solución como una propuesta operativamente factible y estructurada técnicamente.

CAPÍTULO IV

DISEÑO E IMPLEMENTACIÓN DEL SISTEMA PROPUESTO

4.1 Análisis del proceso actual (Modelo AS-IS)

Antes de diseñar e implementar la plataforma sugerida, se llevó a cabo un estudio sobre el flujo presente de gestión de solicitudes estudiantiles en la Universidad Internacional de Ciencia y Tecnología (UNICYT). Esta evaluación permitió que se detectara la estructura operativa actual y las restricciones relacionadas con el modelo que se basa mayormente en el empleo del correo electrónico institucional.

Hoy en día, los alumnos envían a las direcciones de correo electrónico institucional las solicitudes académicas y administrativas, que incluyen cosas como constancias, reclamaciones, validaciones de pago o procesos de matrícula. El personal administrativo reenvía la solicitud manualmente a los departamentos pertinentes, dependiendo del tipo de trámite, una vez que se ha recibido. Este proceso no tiene un identificador único estructurado que posibilite seguir formalmente cada solicitud en un sistema centralizado.

El seguimiento del estado de los trámites está supeditado a la memoria operativa del personal participante y al historial personal de correos electrónicos. No hay un sistema automatizado para verificar el estado de la solicitud ni un registro institucional consolidado que documente las acciones llevadas a cabo en cada caso. Por lo tanto, la capacidad de rastrear el proceso es limitada y se complica la producción de indicadores objetivos como el promedio de tiempo para responder o la cantidad de peticiones por categoría.

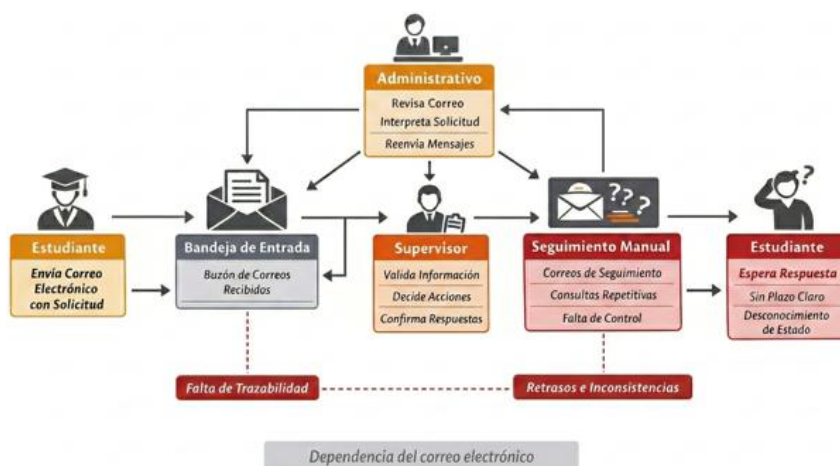
Desde una perspectiva de ingeniería de software y gestión de procesos, el modelo actual presenta las siguientes características:

- Flujo descentralizado.
- Ausencia de número de ticket.
- Asignación manual mediante reenvío de correos.
- Falta de registro estructurado de estados.
- Dificultad para generar métricas institucionales.

Estas condiciones incrementan el riesgo de retrasos, duplicidad de información y falta de visibilidad sobre el estado real de los trámites.

Figura 1.

Flujo AS-IS del proceso actual de gestión de solicitudes en UNICYT.



Nota. Diagrama que ilustra el proceso actual de gestión de solicitudes en UNICYT, caracterizado por la descentralización y la dependencia exclusiva del correo electrónico institucional, lo que genera falta de trazabilidad y retrasos operativos. Elaboración propia (2026).

El análisis del modelo AS-IS permitió establecer la necesidad de diseñar una solución tecnológica que estructure el proceso bajo criterios de centralización, trazabilidad y control.

4.2 Modelo propuesto (Modelo TO-BE)

Se creó un modelo operativo propuesto (TO-BE) con el objetivo de estructurar la administración de las solicitudes estudiantiles a través de una plataforma web centralizada, todo basado en el análisis del proceso actual. El propósito primordial del nuevo modelo es convertir un flujo de datos descentralizado que se basa en correos electrónicos en un sistema formal de seguimiento, registro y control institucional.

En el modelo sugerido, cada requerimiento creado por un alumno se anota en la plataforma y se le asigna de manera automática un identificador único (número de ticket). Este identificador posibilita hacer un seguimiento del procedimiento desde su creación hasta su finalización, lo que asegura la trazabilidad total de todo el proceso. El sistema centraliza toda la información en una base de datos organizada, a diferencia del modelo anterior, que seguía dependiendo del historial de correos.

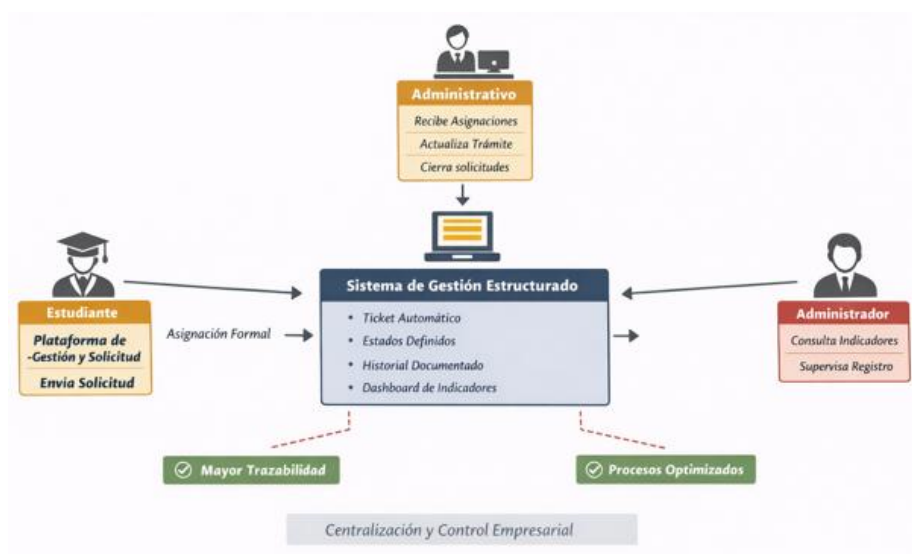
El modelo TO-BE se estructura operativamente de la forma siguiente:

- El alumno ingresa al sistema a través de la autenticación segura.
- Genera una petición eligiendo la categoría correspondiente y añadiendo los documentos necesarios.
- El sistema, de manera automática, crea un número de ticket y anota la fecha y el horario en que fue creado.

- La bandeja administrativa muestra la solicitud.
- Un usuario que tiene un rol administrativo puede alterar el estado del procedimiento y designar al responsable.
- El historial guarda un registro de cada cambio de estado.
- El alumno tiene la posibilidad de verificar en tiempo real el estado de su solicitud.
- Cuando se ha atendido la solicitud, se marca como cerrada, manteniendo el registro histórico completo.

Figura 2.

Modelo TO-BE del proceso de gestión de solicitudes estudiantiles en UNICYT.



Nota. Representación del modelo propuesto mediante una plataforma web centralizada. Se observa la automatización en la generación de tickets, definición de estados y la integración de un dashboard para el control institucional. Elaboración propia (2026)

El modelo sugerido incluye estados bien definidos que facilitan la visualización del ciclo de vida del trámite. Esta definición, al ser estructurada, disminuye la ambigüedad y optimiza la comunicación entre los participantes. Además, la inclusión de roles distintos (de administrador, administrativo y estudiante) asegura que haya un control de acceso y una separación de funciones dentro del sistema.

El modelo TO-BE permite la creación de indicadores básicos de gestión, como el total de solicitudes por cada categoría, el número de trámites que están pendientes y la cantidad procesada por cada usuario administrativo, desde la perspectiva del control institucional. La falta de datos estructurados impedía que estos indicadores se generaran con el modelo anterior.

Además, el modelo nuevo posibilita guardar un historial pormenorizado de las acciones efectuadas en cada solicitud, como cambios de estado, asignaciones y registros de intervención. Este elemento mejora la trazabilidad y brinda pruebas objetivas del proceso, lo cual hace más sencillo el análisis y la auditoría posteriores.

El modelo TO-BE muestra las siguientes ventajas en comparación con el modelo AS-IS:

- Centralización de la información.
- Identificación única por solicitud.
- Seguimiento en tiempo real.
- Registro estructurado de acciones.
- Control de acceso por roles.

- Capacidad de generación de métricas.

El diseño del modelo que se propone no solamente satisface una necesidad de tipo operativo, sino también se ajusta a los principios de transformación digital institucional, gestión de procesos y arquitectura estructurada, asegurando así una sólida base técnica para la puesta en marcha del sistema creado.

4.3 Arquitectura del sistema

El sistema desarrollado fue diseñado bajo una arquitectura estructurada orientada a la separación de responsabilidades y control modular del código. La implementación adopta el patrón Modelo–Vista–Controlador (MVC) en su versión manual utilizando PHP puro, sin el uso de frameworks externos, con el propósito de mantener control total sobre la estructura y el flujo de ejecución del sistema.

La arquitectura definida permite organizar el sistema en componentes independientes que interactúan de manera controlada, facilitando la mantenibilidad, escalabilidad y claridad estructural del código.

4.3.1 Arquitectura general

El sistema utiliza un esquema de Front Controller, donde todas las solicitudes HTTP son canalizadas a través de un punto único de entrada (`index.php`). Este componente central se encarga de interpretar la ruta solicitada y dirigir la petición al controlador correspondiente.

El flujo general de ejecución es el siguiente:

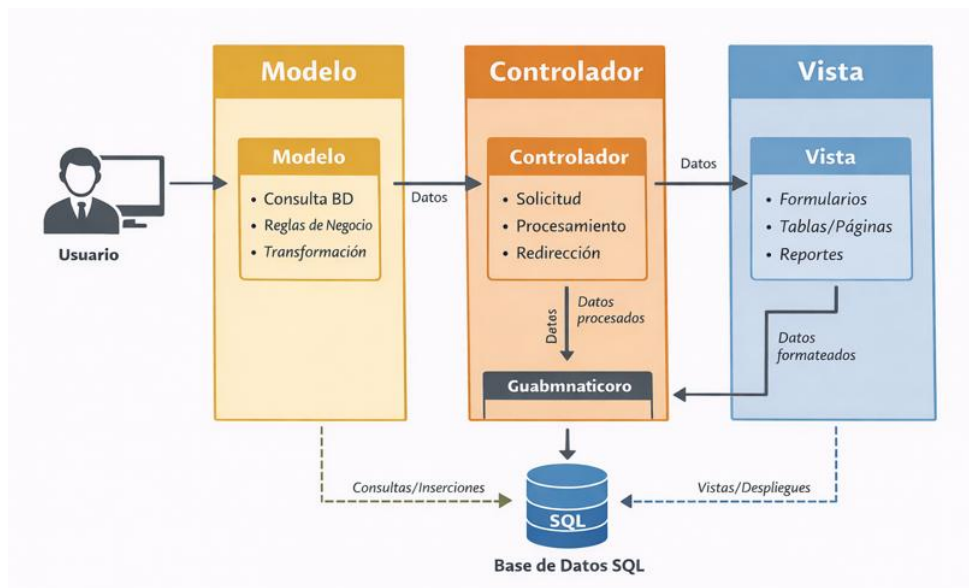
- El usuario realiza una solicitud HTTP.

- El Front Controller recibe la petición.
- El Router identifica la ruta y el método HTTP.
- Se invoca el controlador correspondiente.
- El controlador interactúa con el modelo.
- El modelo consulta o actualiza la base de datos mediante PDO.
- El controlador envía los datos a la vista.
- La vista renderiza la respuesta al usuario.

De esta forma posibilita la separación de la lógica de negocio con la capa de presentación, lo cual disminuye la dependencia directa entre los componentes.

Figura 3.

Diagrama de arquitectura general Modelo–Vista–Controlador (MVC) implementada en el sistema.



Nota. Diagrama de la arquitectura Modelo–Vista–Controlador implementada de forma manual en el sistema. Muestra el flujo de datos desde la solicitud del usuario hasta la consulta en la base de datos SQL y su posterior renderización. Elaboración propia (2026).

4.3.2 Separación por capas

La arquitectura implementada se organiza en las siguientes capas:

Capa de Presentación (Views)

Responsable de la interfaz visual del sistema.

Incluye:

- Formularios de autenticación
- Dashboard
- Creación y visualización de solicitudes
- Reportes e indicadores

Se utiliza HTML5, CSS3 y Bootstrap por CDN, manteniendo un diseño minimalista y académico.

Capa de Control (Controllers)

Encargada de:

- Gestionar las solicitudes HTTP
- Validar datos de entrada
- Aplicar reglas de negocio

- Controlar el flujo entre modelos y vistas

Cada módulo funcional posee su propio controlador (por ejemplo: AuthController, SolicitudesController, DashboardController).

Capa de Modelo (Models)

Responsable de la interacción con la base de datos.

Se implementa acceso mediante PDO utilizando consultas preparadas (prepared statements), lo cual reduce riesgos de inyección SQL y fortalece la seguridad del sistema.

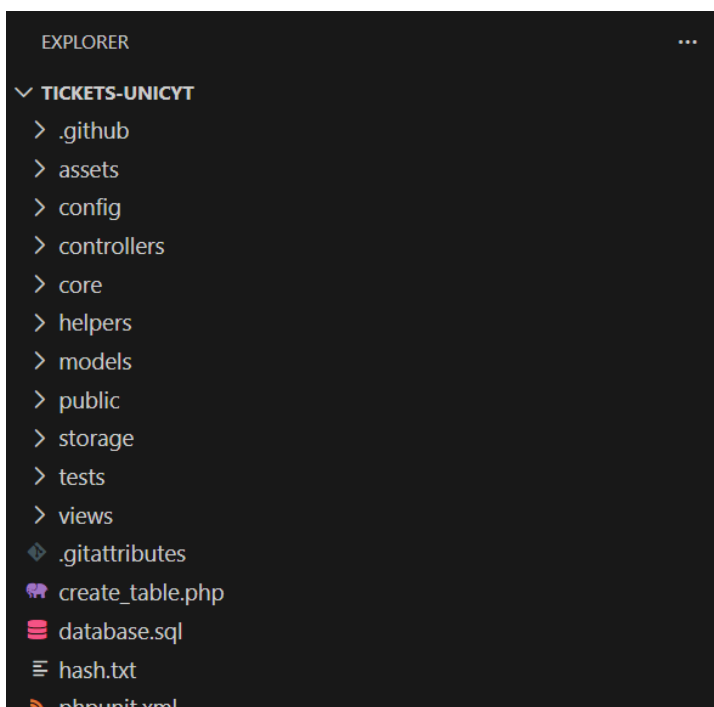
Capa de Datos

La base de datos MySQL almacena:

- Usuarios
- Categorías
- Solicitudes
- Historial
- Registros de auditoría

El modelo relacional permite garantizar integridad referencial y trazabilidad.

Figura 4.
Carpetas del Proyecto



Nota. Captura de pantalla del entorno de desarrollo que muestra la organización modular del código fuente, estructurado bajo el patrón MVC para facilitar la mantenibilidad y escalabilidad del software. Elaboración propia (2026).

4.3.3 Seguridad estructural

El sistema incorpora mecanismos básicos de seguridad alineados con buenas prácticas de desarrollo web:

Autenticación

- Uso de sesiones PHP seguras.
- Almacenamiento de contraseñas utilizando funciones de hash seguro (password_hash).
- Verificación de credenciales antes de permitir acceso al sistema.

Control de acceso por roles

El sistema implementa tres roles principales:

- Estudiante
- Administrativo
- Administrador

Cada rol posee permisos diferenciados, limitando acciones según responsabilidades definidas.

Protección contra inyección SQL

Todas las consultas a base de datos utilizan prepared statements mediante PDO, evitando la concatenación directa de variables en sentencias SQL.

Protección CSRF

Los formularios críticos incorporan tokens de verificación para prevenir ataques de tipo Cross-Site Request Forgery.

Figura 5.
Implementación del token CSRF en formulario

```

1  <?php
2  namespace Core;
3
4  class Csrft
5  {
6      public static function generateToken()
7      {
8          if (empty($_SESSION['csrf_token'])) {
9              $_SESSION['csrf_token'] = bin2hex(random_bytes(32));
10             }
11             return $_SESSION['csrf_token'];
12         }
13
14         public static function validateToken($token)
15         {
16             if (!empty($_SESSION['csrf_token']) && hash_equals($_SESSION['csrf_t
17                 return true;
18             }
19             return false;
20         }
21     }

```

Nota. Fragmento de código en PHP que muestra la lógica de generación y validación de tokens de seguridad para prevenir ataques de falsificación de solicitudes entre sitios (CSRF). Elaboración propia (2026).

4.3.4 Justificación de la arquitectura adoptada

La selección de una arquitectura MVC manual se basa en los siguientes estándares:

- Control absoluto del flujo interno.
- Entendimiento integral de la estructura del sistema.
- Evitar depender de marcos externos.
- Exhibir habilidad de ingeniería en la fabricación del prototipo.
- Promover la posibilidad de escalar en el futuro y el mantenimiento.

Esta arquitectura posibilita que el sistema se expanda de manera modular, añadiendo módulos nuevos sin impactar directamente la estructura ya existente, lo que disminuye el riesgo de acoplamiento excesivo.

4.4 Diseño de base de datos

El sistema sugerido está basado en un modelo relacional que se implementa en MySQL, el cual fue creado para asegurar la integridad de los datos, su trazabilidad y una organización estructurada. El modelo de datos fue creado teniendo en cuenta los requisitos funcionales detectados y la necesidad de mantener la coherencia en el registro y seguimiento de las peticiones hechas por los alumnos.

El diseño incluye entidades principales que simbolizan registros históricos, usuarios, solicitudes y categorías, y determina relaciones precisas entre ellas a través de claves foráneas y primarias.

4.4.1 Modelo de entidad-relación

Las entidades principales, según el modelo entidad–relación (ER) del sistema, son las siguientes:

- Usuarios: almacena los datos de todos los usuarios del sistema (alumnos, personal administrativo y administrador).
- Categorías: clasificar las distintas clases de solicitudes.
- Solicitudes: Se refiere a cada gestión que se produce en el sistema.
- Historial: documenta todas las acciones que se han llevado a cabo respecto a cada solicitud.

- Audit_logs: guarda eventos significativos del sistema para propósitos de supervisión y monitoreo.

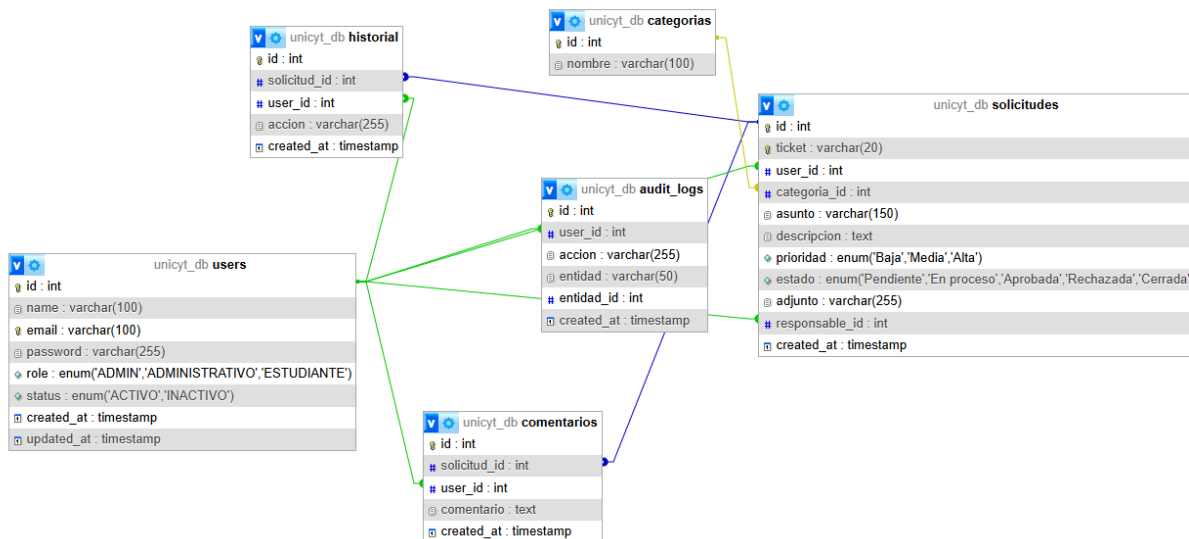
La relación central del modelo gira en torno a la entidad Solicitudes, la cual se vincula:

- Con Users (usuario creador y usuario asignado).
- Con Categorías (tipo de trámite).
- Con Historial (registro de cambios).
- Con Audit_logs (acciones críticas del sistema).

Este diseño permite reconstruir el ciclo de vida completo de cada solicitud

Figura 6.

Diagrama ER generado desde phpMyAdmin



Nota. Modelo relacional de la base de datos generado en phpMyAdmin. Se detallan las entidades principales (usuarios, solicitudes, historial, categorías y auditoría) y sus relaciones para garantizar la integridad referencial. Elaboración propia (2026).

4.4.2 Descripción de tablas principales

Tabla: users

Almacena los datos de autenticación y rol del usuario.

Campos principales:

- id (clave primaria)
- nombre
- email
- password
- rol
- created_at

La inclusión del campo *rol* permite implementar control de acceso basado en permisos diferenciados.

Tabla: categorias

Permite clasificar las solicitudes según su naturaleza administrativa.

Campos principales:

- id

- nombre
- descripcion

Esta estructura facilita la generación de indicadores por tipo de trámite.

Tabla: solicitudes

Entidad central del sistema.

Campos principales:

- id
- ticket (identificador único)
- user_id (creador)
- categoria_id
- descripcion
- estado
- asignado_a
- created_at
- updated_at

El campo *ticket* permite identificar cada solicitud de manera única.

El campo *estado* define el ciclo de vida del trámite.

Tabla: historial

Registra cada intervención realizada sobre una solicitud.

Campos principales:

- id
- solicitud_id
- usuario_id
- accion
- fecha

Esta tabla es fundamental para la trazabilidad completa del proceso.

Tabla: audit_logs

Registra eventos críticos del sistema, como inicios de sesión o modificaciones relevantes.

Campos principales:

- id
- usuario_id
- accion
- fecha

Este componente fortalece el control institucional y la capacidad de auditoría.

4.4.3 Integridad y relaciones

El diseño incorpora:

- Claves primarias auto incrementales.
- Claves foráneas entre solicitudes y usuarios.
- Relaciones entre solicitudes y categorías.
- Relaciones entre historial y solicitudes.

La implementación de claves foráneas permite mantener integridad referencial, evitando registros huérfanos y garantizando coherencia estructural en la base de datos.

Adicionalmente, se incluyeron índices en campos de búsqueda frecuente como:

- ticket
- estado
- categoria_id
- usuario_id

Estos índices mejoran el rendimiento en consultas administrativas y generación de reportes.

Figura 7.

Estructura de tablas en phpMyAdmin.

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
☐ audit_logs	Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_0900_ai_ci	32.0 KB	-
☐ categorias	Examinar Estructura Buscar Insertar Vaciar Eliminar	5	InnoDB	utf8mb4_0900_ai_ci	16.0 KB	-
☐ comentarios	Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_0900_ai_ci	48.0 KB	-
☐ historial	Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_0900_ai_ci	48.0 KB	-
☐ solicitudes	Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_0900_ai_ci	80.0 KB	-
☐ users	Examinar Estructura Buscar Insertar Vaciar Eliminar	3	InnoDB	utf8mb4_0900_ai_ci	32.0 KB	-
6 tablas	Número de filas	8	InnoDB	utf8mb4_0900_ai_ci	256.0 KB	0 B

Nota. Captura de pantalla del gestor de base de datos que detalla la lista de tablas (users, solicitudes, historial, etc.) y sus características técnicas como el motor de almacenamiento InnoDB. Elaboración propia (2026).

4.4.4 Justificación del modelo relacional adoptado

El modelo relacional fue seleccionado debido a:

- Necesidad de integridad estructural.
- Relaciones claras entre entidades.
- Requerimiento de trazabilidad.
- Facilidad para generar consultas agregadas e indicadores.

La estructura implementada permite garantizar consistencia en el registro de solicitudes y proporciona una base sólida para la generación de métricas de desempeño institucional.

4.5 Diseño de interfaz

La interfaz del sistema fue diseñada con una perspectiva académica y minimalista, dando mayor importancia a la funcionalidad, la claridad estructural y la coherencia visual

que a los componentes decorativos. El propósito fue asegurar que el usuario encontrara la plataforma intuitiva, sin que su experiencia se viera saturada con elementos superfluos.

Se implementaron normas de jerarquía visual, modularización y simpleza desde el punto de vista del diseño funcional. Cada vista fue diseñada para que el usuario pudiera identificar fácilmente las acciones principales disponibles, como crear una solicitud, consultar el estado, revisar el historial o manejar los trámites administrativos.

El sistema emplea Bootstrap 5 a través de CDN, añadiendo estilos propios, lo que permitió que componentes como formularios, botones, tablas y tarjetas informativas mantuvieran la uniformidad. Esta decisión técnica permite que el sistema responda mejor, garantizando que sea compatible con pantallas de diferentes tamaños sin la necesidad de configuraciones adicionales complicadas.

4.5.1 Principios de diseño adoptados

El diseño de interfaz se fundamenta en los siguientes principios:

- Claridad en la navegación.
- Reducción de elementos distractores.
- Consistencia visual entre módulos.
- Separación clara entre información y acciones.
- Uso moderado del color para diferenciar estados.

Los estados de las solicitudes se muestran visualmente a través de etiquetas o indicadores que son fácilmente reconocibles (como "Aprobada", "Pendiente", "Rechazada", "En proceso"), lo cual posibilita que el usuario pueda entender con rapidez el estado de su trámite.

4.5.2 Estructura de navegación

La navegación del sistema se organiza según el rol del usuario autenticado.

Para el estudiante:

- Acceso a dashboard personal.
- Opción de crear nueva solicitud.
- Visualización del historial propio.
- Consulta del estado de cada trámite.

Para el administrativo:

- Acceso a bandeja general.
- Filtros por estado y categoría.
- Asignación de solicitudes.
- Cambio de estados.

Para el administrador:

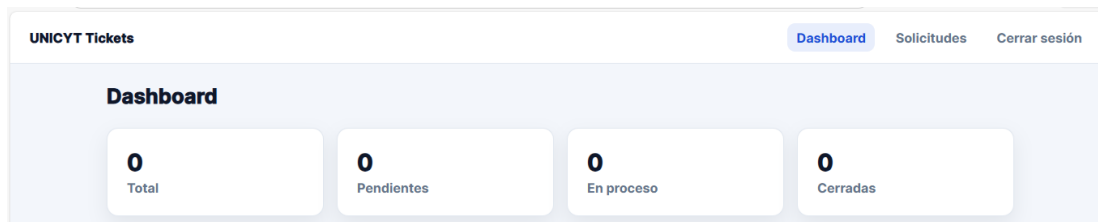
- Acceso a métricas globales.
- Visualización de bitácora de auditoría.

- Gestión general del sistema.

Esta estructura posibilita la conservación de un control de acceso basado en roles, previniendo que un usuario tenga acceso a opciones que no le pertenecen.

Figura 8.

Menú Principal de los Roles en la Plataforma



Nota. Vista de la barra de navegación superior que se adapta según el perfil del usuario, permitiendo el acceso segregado a módulos de solicitudes y administración de usuarios. Elaboración propia (2026).

4.5.3 Diseño responsive

La interfaz se creó para que pueda emplearse en una variedad de dispositivos, incluyendo computadoras de escritorio, laptops y dispositivos móviles. Bootstrap posibilita que columnas y componentes se reorganicen de forma automática, garantizando su funcionalidad y legibilidad sin poner en riesgo la experiencia del usuario.

El rendimiento también se ve favorecido por el diseño minimalista, porque este disminuye la carga de recursos gráficos no esenciales y mantiene un tiempo de respuesta apropiado.

4.5.4 Justificación del enfoque adoptado

La interfaz se creó para que pueda emplearse en una variedad de dispositivos, incluyendo computadoras de escritorio, laptops y dispositivos móviles. Bootstrap

posibilita que columnas y componentes se reorganicen de forma automática, garantizando su funcionalidad y legibilidad sin poner en riesgo la experiencia del usuario.

El rendimiento también se ve favorecido por el diseño minimalista, porque este disminuye la carga de recursos gráficos no esenciales y mantiene un tiempo de respuesta apropiado.

4.6 Implementación de módulos

El sistema fue implementado en módulos funcionales, que fueron organizados de acuerdo con la arquitectura MVC que se utilizó. Cada módulo está diseñado para cumplir un conjunto particular de responsabilidades en el flujo general de gestión de solicitudes estudiantiles, lo que posibilita una organización del sistema que es modular y coherente.

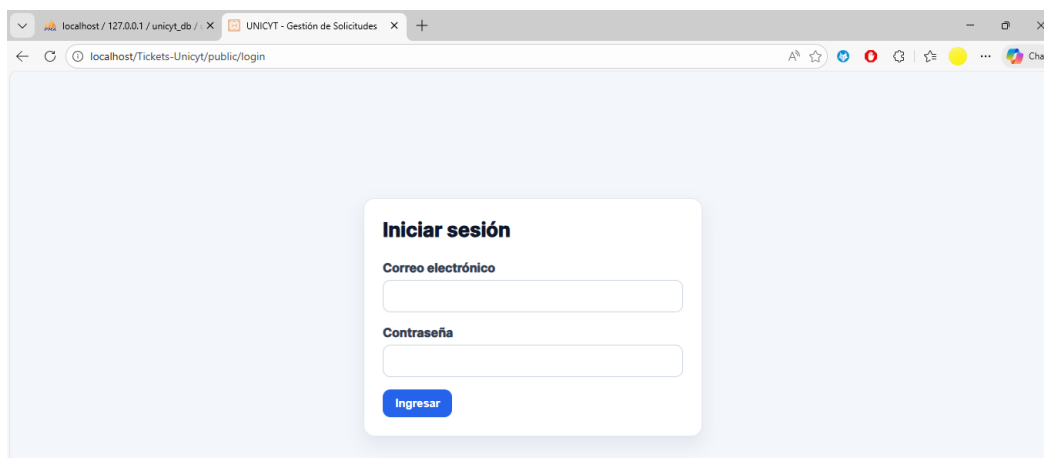
El desarrollo modular permite que el código sea más fácil de mantener, que las responsabilidades estén separadas y que se puedan hacer ampliaciones a futuro sin alterar la estructura general.

4.6.1 Módulo de autenticación y control de acceso

Este módulo gestiona el proceso de inicio y cierre de sesión, validación de credenciales y establecimiento de sesiones seguras. Las contraseñas se almacenan utilizando mecanismos de hash seguro, garantizando protección de información sensible.

Adicionalmente, el sistema implementa control de acceso basado en roles, diferenciando permisos entre estudiantes, personal administrativo y administrador. Esta segregación evita accesos indebidos y asegura que cada usuario pueda interactuar únicamente con las funcionalidades correspondientes a su perfil.

Figura 9.
Pantalla de Login o Inicio de Sesión de la Plataforma



Nota. Interfaz del módulo de autenticación de la plataforma. El diseño es minimalista y académico, enfocado en la seguridad mediante el uso de sesiones seguras y validación de credenciales. Elaboración propia (2026).

4.6.2 Módulo de gestión de solicitudes

Este módulo es el centro funcional del sistema. Facilita que los alumnos generen nuevas peticiones, elijan categorías, anexen documentos y detallen lo que requieren.

Cada solicitud se guarda en la base de datos con un estado inicial específico y recibe automáticamente un número único de ticket. El sistema ofrece la posibilidad de ver las solicitudes de manera estructurada, a través de listas organizadas por categoría y estado.

Figura 10.
Pantalla de Creación de Nueva solicitud

The screenshot shows a web browser window with the URL `localhost/Tickets-Unicyt/public/solicitudes/create`. The page title is 'UNICYT Tickets'. The navigation bar includes 'Dashboard', 'Solicitudes', and 'Cerrar sesión'. The main content area is titled 'Nueva solicitud' and contains the following form elements:

- Categoría:** A dropdown menu with the selected option 'Seleccione'.
- Asunto:** A text input field.
- Descripción:** A larger text area for detailed description.
- Prioridad:** A dropdown menu with the selected option 'Media'.
- Adjunto (PDF/JPG/PNG max 5MB):** A file upload field.

Nota. Interfaz del módulo de usuario que permite al estudiante iniciar un trámite seleccionando una categoría predefinida (como Matrícula o Reclamos) y asignando un nivel de prioridad al requerimiento. El diseño prioriza la facilidad de uso y la captura estructurada de datos iniciales. Elaboración propia (2026).

Figura 11.
Pantalla de Listado de Solicitudes

The screenshot shows a web browser window with the URL `localhost/Tickets-Unicyt/public/solicitudes?categoria_id=&estado=`. The page title is 'UNICYT Tickets'. The navigation bar includes 'Dashboard', 'Solicitudes', and 'Cerrar sesión'. The main content area is titled 'Solicitudes' and includes a 'Nueva solicitud' button. Below the title are two filter dropdowns: 'Categoría' (set to 'Todas') and 'Estado' (set to 'Todos'), followed by a 'Filtrar' button. Below the filters is a table with the following columns: TICKET, ASUNTO, CATEGORÍA, PRIORIDAD, ESTADO, and FECHA.

Nota. Bandeja de entrada con funciones de filtrado por categoría y estado, permitiendo una visualización organizada de los trámites activos e históricos. Elaboración propia (2026).

Figura 12.
Formulario de Nueva Solicitud

Nueva solicitud

Categoría
Matricula

Asunto
Matricular Materias

Descripción
Hola, requiero matricular la materia de Bases de Datos.

Prioridad
Media

Adjunto (PDF/JPG/PNG max 5MB)
Elegir archivo No se eligió ningún archivo

[Crear solicitud](#) [Cancelar](#)

Nota. Vista detallada de los campos de entrada donde el sistema valida la información obligatoria, como el asunto y la descripción técnica del trámite. Este componente asegura que cada petición cuente con el sustento necesario antes de ser procesada por el departamento administrativo. Elaboración propia (2026).

Figura 13.
Detalle de Solicitudes Creadas

UNICYT Tickets [Dashboard](#) [Solicitudes](#) [Cerrar sesión](#)

Solicitud creada con éxito

Solicitudes [Nueva solicitud](#)

Categoría Todas **Estado** Todos [Filtrar](#)

TICKET	ASUNTO	CATEGORÍA	PRIORIDAD	ESTADO	FECHA
TKT-20260301-7459	Matricular Materias	Matricula	Media	Pendiente	2026-03-01 11:19:00 Ver

Nota. Interfaz del formulario de creación de trámites donde el estudiante selecciona la categoría, prioridad y adjunta documentación de respaldo en formatos PDF o imagen. Elaboración propia (2026).

4.6.3 Módulo de asignación y gestión de estados

El personal administrativo tiene la capacidad de asignar solicitudes a personas responsables concretas y también puede cambiar su estado a medida que el procedimiento progresa. Los estados definidos posibilitan la representación del ciclo de vida del trámite, lo que hace más fácil el seguimiento y la supervisión.

Cada vez que hay un cambio de estado, se crea un registro en el historial vinculado con la solicitud, lo que garantiza una trazabilidad total.

Figura 14.

Detalle de Solicitud y Cambio de Estado de Solicitud

The screenshot shows a web browser window with the URL `localhost/Tickets-Unity/public/solicitudes/show?id=1`. The page title is "UNICYT Tickets" and it includes navigation links for "Dashboard", "Solicitudes", and "Cerrar sesión".

The main content area is titled "Detalle de solicitud" and is divided into two panels:

- Información:** A table displaying ticket details.

TICKET	TKT-20260301-7459
ESTUDIANTE	Juan Estudiante
CATEGORÍA	Matricula
ASUNTO	Matricular Materias
DESCRIPCIÓN	Hola, requiero matricular la materia de Bases de Datos.
PRIORIDAD	Media
ESTADO	Pendiente
RESPONSABLE	Sin asignar
- Acciones:** A panel for managing the ticket's state.
 - Estado:** A dropdown menu currently set to "Pendiente".
 - Responsable:** A dropdown menu currently set to "Sin asignar".
 - Guardar cambios:** A blue button to save the changes.

Nota. Vista expandida de un ticket individual que permite al personal administrativo asignar responsables y actualizar el estado del trámite en tiempo real. Elaboración propia (2026).

4.6.4 Módulo de historial y trazabilidad

Este módulo documenta todas las acciones ejecutadas sobre cada solicitud. Contiene datos sobre el usuario responsable, la fecha y la hora, así como la acción realizada.

La transparencia del proceso se fortalece con la presencia de este componente, así como la posibilidad de reconstruir el procedimiento completo desde su inicio hasta su finalización.

Figura 15.

Historial de la Solicitud



FECHA	USUARIO	ACCIÓN
2026-03-01 11:19:00	Juan Estudiante	Solicitud creada

[← Volver a solicitudes](#)

© 2026 UNICYT

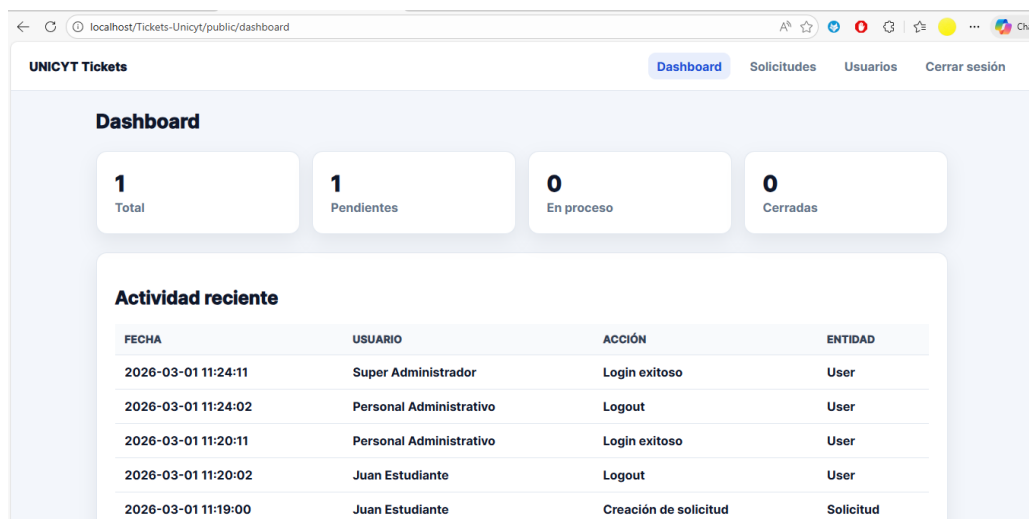
Nota. Vista detallada del módulo de trazabilidad, donde se registran de forma cronológica todas las acciones, usuarios responsables y cambios de estado realizados sobre un trámite específico. Elaboración propia (2026).

4.6.5 Módulo de reportes e indicadores

El sistema incluye un módulo que permite visualizar métricas fundamentales, como la cantidad de trámites pendientes, el total administrado y cuántas solicitudes hay por categoría.

Estas métricas se producen mediante consultas estructuradas en la base de datos, lo que posibilita transformar los registros operativos en información valiosa para la toma de decisiones administrativas.

Figura 16.
Dashboard con métricas



Nota. Panel de control administrativo que visualiza indicadores de gestión en tiempo real, tales como el total de solicitudes pendientes, en proceso y cerradas, facilitando la toma de decisiones. Elaboración propia (2026).

4.6.6 Consideraciones técnicas generales

Se llevó a cabo la implementación empleando PHP puro con arquitectura MVC manual, accediendo a la base de datos por medio de PDO y consultas preestablecidas. Cada elemento puede funcionar de manera independiente gracias a la estructura modular adoptada, lo que disminuye el acoplamiento y favorece la mejora futura.

4.7 Pruebas funcionales y validación

Después de terminar la implementación del sistema, se llevaron a cabo pruebas manuales para comprobar que los módulos desarrollados funcionaban correctamente y que se cumplían los requisitos establecidos en el Capítulo I.

Las pruebas se orientaron a evaluar:

- Autenticación y control de acceso.
- Gestión de solicitudes.
- Asignación y cambio de estados.
- Registro de historial.
- Generación de indicadores.
- Funcionamiento de la bitácora de auditoría.
- Validaciones de seguridad básica.

4.7.1 Validación de autenticación y control de roles

Se verificó que el sistema permita el acceso únicamente a usuarios registrados y que las credenciales sean validadas correctamente antes de iniciar sesión.

Asimismo, se comprobó que los permisos estén correctamente restringidos según el rol del usuario:

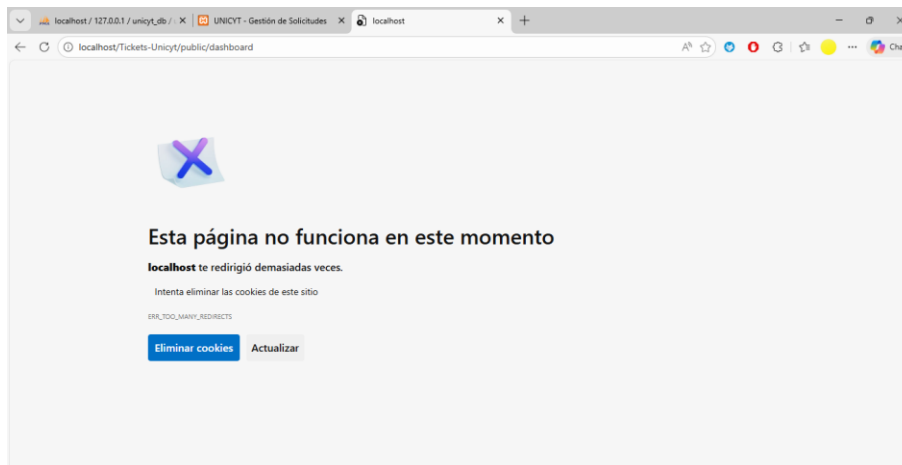
- El estudiante puede crear y consultar únicamente sus propias solicitudes.
- El personal administrativo puede visualizar y gestionar solicitudes asignadas.

- El administrador posee acceso global y visualización de métricas y bitácora.

Se realizaron pruebas intentando acceder a rutas restringidas sin permisos adecuados, confirmando que el sistema bloquea dichos accesos.

Figura 17.

Validación de control de acceso por roles.



Nota. Evidencia de las pruebas de seguridad donde el sistema bloquea y redirige intentos de acceso no autorizados a rutas restringidas de la plataforma. Elaboración propia (2026).

4.7.2 Validación del flujo de solicitudes

Se ejecutaron pruebas completas del ciclo de vida de una solicitud:

1. Creación de solicitud por parte del estudiante.
2. Generación automática del número de ticket.
3. Visualización en bandeja administrativa.
4. Asignación a responsable.

5. Cambio de estado.

6. Cierre del trámite.

Se verificó que cada transición quedara registrada en el historial correspondiente y que la información se mantuviera consistente en la base de datos.

Figura 18.

Pantalla Completa del detalle de una Solicitud.

Detalle de solicitud

Información

TICKET	TKT-20260301-7459
ESTUDIANTE	Juan Estudiante
CATEGORÍA	Matrícula
ASUNTO	Matricular Materias
DESCRIPCIÓN	Hola, requiero matricular la materia de Bases de Datos.
PRIORIDAD	Medio
ESTADO	Pendiente
RESPONSABLE	Sin asignar

Acciones

Estado: Pendiente

Responsable: Sin asignar

[Guardar cambios](#)

Historial

FECHA	USUARIO	ACCIÓN
2026-03-01 11:19:00	Juan Estudiante	Solicitud creada

[Volver a solicitudes](#)

Nota. Vista integral de la gestión de un ticket que unifica la información general del estudiante, los controles de asignación para el personal administrativo y el panel de historial cronológico. Esta interfaz es el núcleo de la trazabilidad del sistema, permitiendo reconstruir el ciclo de vida completo del trámite. Elaboración propia (2026).

4.7.3 Validación de historial y trazabilidad

Se comprobó que cada acción relevante sobre una solicitud generara un registro en la tabla de historial, incluyendo:

- Usuario que realizó la acción.
- Fecha y hora.

- Descripción del evento.

Esta validación permitió confirmar la capacidad del sistema para reconstruir el recorrido completo del trámite.

4.7.4 Validaciones de seguridad básica

Se realizaron pruebas orientadas a identificar posibles vulnerabilidades operativas, verificando que:

- Las contraseñas se almacenarán cifradas.
- Las consultas a base de datos utilizaran consultas preparadas.
- Los formularios incorporaran mecanismos de protección CSRF.
- No fuera posible ejecutar acciones sin sesión activa.

Estas pruebas confirmaron que el sistema incorpora medidas básicas de seguridad alineadas con buenas prácticas de desarrollo web.

4.7.5 Resultados de validación

Las pruebas funcionales demostraron que el sistema cumple con los requerimientos establecidos, garantizando:

- Registro estructurado de solicitudes.
- Seguimiento en tiempo real.
- Trazabilidad completa.
- Control de acceso por roles.
- Generación de métricas básicas.

- Registro de auditoría institucional.

El sistema se comportó de manera estable durante las pruebas ejecutadas, sin presentar errores críticos que comprometieran su funcionamiento general.

4.8 Resultados obtenidos

Pudimos convertir el modelo de operación anterior, que se fundamentaba en la comunicación a través de correo electrónico, en una plataforma organizada para gestionar y hacer seguimiento a las solicitudes de los estudiantes gracias a la implementación del sistema. Los resultados logrados pueden ser examinados tomando como base la comparación entre el modelo TO-BE y el modelo AS-IS, así como en función de si se cumplieron las metas establecidas.

4.8.1 Comparación entre modelo AS-IS y modelo TO-BE

Antes de ser implementado, el proceso se distinguía por la falta de rastreo estructurado, la descentralización de la información y la falta de identificadores únicos. El seguimiento se basaba en el intercambio manual de correos electrónicos, y no había un método formal para crear indicadores de rendimiento.

Tras la implementación del sistema, se lograron los siguientes cambios estructurales:

Tabla 1. Comparación entre los Modelos AS-IS, TO-BE

Aspecto	Modelo AS-IS	Modelo TO-BE
Registro de solicitudes	Correo electrónico	Plataforma centralizada
Identificación	Sin número único	Ticket automático
Seguimiento	Manual	Estados definidos
Asignación	Reenvío de correos	Asignación dentro del sistema

Historial	No estructurado	Registro completo por solicitud
Métricas	No disponibles	Dashboard con indicadores
Auditoría	No consolidada	Bitácora visible

Nota. Cuadro comparativo que resume las mejoras estructurales entre el modelo actual (AS-IS) basado en correo y el modelo propuesto (TO-BE) basado en la plataforma centralizada. Elaboración propia (2026).

4.8.2 Cumplimiento de objetivos específicos

En relación con los objetivos definidos en el Capítulo I, se obtuvieron los siguientes resultados:

- Se analizó el flujo actual de gestión de solicitudes, identificando debilidades estructurales.
- Se diseñó una arquitectura técnica basada en el patrón MVC manual.
- Se modeló una base de datos relacional con integridad referencial.
- Se implementó un prototipo funcional en PHP.
- Se incorporaron mecanismos de autenticación y control de acceso por roles.
- Se desarrolló un módulo de seguimiento con estados definidos.
- Se integró un sistema de historial y bitácora de auditoría.
- Se habilitó la generación de indicadores básicos de gestión.

4.8.3 Impacto operativo

Desde el punto de vista operativo, el sistema permite:

- Centralizar la información de solicitudes.

- Reducir la dependencia del correo electrónico.
- Mejorar la visibilidad del estado de los trámites.
- Facilitar la supervisión administrativa.
- Proporcionar evidencia documental estructurada.

La incorporación de métricas básicas permite transformar registros operativos en información útil para la toma de decisiones internas, fortaleciendo la gobernanza institucional.

4.8.4 Valor técnico del prototipo

La habilidad para crear un sistema sin la necesidad de frameworks externos, utilizando principios de separación de responsabilidades, seguridad básica y modularidad, se evidencia en el desarrollo manual basado en arquitectura MVC.

La estructura del prototipo no solo sirve para demostrar, sino que también es escalable y permite añadir nuevas funciones, como la integración con otros sistemas institucionales, la generación de informes avanzados o las notificaciones automáticas.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones generales

Este proyecto posibilitó la creación y puesta en marcha de una plataforma web organizada para manejar y monitorear las peticiones de los alumnos en la Universidad Internacional de Ciencia y Tecnología (UNICYT). La solución desarrollada fue una respuesta directa a los problemas detectados, que se distinguieron por la falta de un sistema centralizado, la imposibilidad de rastrear información y la escasa capacidad para producir indicadores de gestión.

La puesta en marcha del sistema evidenció que se puede convertir un procedimiento operacional basado en comunicaciones informales por correo electrónico en un ambiente digital con estructura, que incluye registro ordenado, distribución formal de responsabilidades y seguimiento en tiempo real. Desde un punto de vista técnico, la implementación manual de una arquitectura MVC posibilitó que el desarrollo se organizara de forma modular, asegurando la separación de responsabilidades, la posibilidad de mantenimiento y el control del flujo interno del sistema.

La plataforma desarrollada, en el plano institucional, representa un fundamento firme para robustecer la administración universitaria. Esto es posible gracias a que facilita la consolidación de información, la creación de indicadores básicos y el aumento de la visibilidad del estado de los trámites.

5.2 Conclusiones específicas

La evaluación del flujo presente de gestión de solicitudes posibilitó detectar deficiencias estructurales relacionadas con la falta de un sistema centralizado, la carencia de trazabilidad formal y la imposibilidad de crear indicadores objetivos del

rendimiento. Este diagnóstico fue esencial para respaldar, desde el punto de vista técnico, la necesidad de una solución digital estructurada.

Con base en ese análisis, se creó una estructura técnica fundamentada en el patrón de Modelo-Vista-Controlador (MVC) manual, lo cual permitió que el sistema estuviera organizado conforme a los principios de control del flujo interno, modularidad y división de responsabilidades. Esta elección de arquitectura aseguró que la estructura fuera clara y permitió que los diferentes módulos funcionales se implementaran de manera gradual.

Además, se diseñó una base de datos relacional que une entidades relacionadas con usuarios, categorías, solicitudes e historial, garantizando la integridad referencial y la posibilidad de reconstruir el ciclo de vida de cada trámite. El agregar claves foráneas y registros históricos permitió establecer la trazabilidad como un elemento fundamental del sistema.

El prototipo funcional creado en PHP satisface las exigencias establecidas, incluyendo la autenticación segura, el control de acceso por roles, la creación automática de tickets y el registro organizado de acciones. Estos componentes evidencian la factibilidad técnica de la solución sugerida.

En última instancia, la incorporación de métricas esenciales y la visualización de indicadores demuestran que la digitalización del procedimiento no solo optimiza la organización operativa, sino que además posibilita instrumentos para el análisis y

monitoreo institucional que antes no existían. En general, estos hallazgos demuestran que se lograron las metas concretas establecidas en el proyecto.

5.3 Aportes del proyecto

El proyecto contribuye en tres aspectos fundamentales:

Contribución técnica:

Muestra cómo se aplican, en la creación de un sistema administrativo que funcione, los principios de gestión de datos, arquitectura de software y seguridad básica.

Contribución institucional:

Proporciona un instrumento que puede ayudar a actualizar los procedimientos administrativos de las universidades, mejorando la capacidad de seguimiento y control organizacional.

Contribución metodológica:

Combina principios teóricos de administración de calidad, transformación digital y sistemas de ticketing en una solución específica y funcional.

5.4 Limitaciones del estudio

El sistema creado es un prototipo funcional que no se implementó en un entorno institucional real de producción. De igual modo, no se llevó a cabo una evaluación longitudinal del efecto operativo ni una validación formal con los usuarios finales en el interior de la universidad.

Además, el sistema no fue incorporado con las plataformas académicas ya existentes, lo que supone una posibilidad de mejora a futuro.

5.5 Recomendaciones

Por la parte técnica, se sugiere tener en cuenta mejoras futuras como la integración con sistemas académicos de la institución, la ampliación del módulo de indicadores y reportes, así como la implementación de notificaciones automáticas.

Desde el punto de vista institucional, se recomienda que la puesta en marcha de la plataforma vaya acompañada de políticas claras sobre su uso, formación para los empleados administrativos y una definición formal de los flujos internos de atención.

Por último, se aconseja llevar a cabo una evaluación de la posibilidad de poner en práctica el sistema en un entorno real para determinar su efecto sobre los tiempos de respuesta, la satisfacción del usuario y la eficiencia administrativa.

Referencias Bibliográficas

- Lustosa Rosario, A. C., Yaacov, B. B., Franco Segura, C., Arias Ortiz, E., Heredero, E., Botero, J., Brothers, P., Payva, T., & Spies, M. (2021). *Transformación digital en la educación superior en América Latina y el Caribe*. Banco Interamericano de Desarrollo. <https://doi.org/10.18235/0003829>
- Organisation for Economic Co-operation and Development. (2020). *The state of higher education: Digitalisation and student experience*. OECD Publishing. <https://doi.org/10.1787/83c41957-en>
- UNESCO. (2021). *Reimagining our futures together: A new social contract for education*. UNESCO Publishing. <https://unesdoc.unesco.org/ark:/48223/pf0000379707>
- Verhoef, P. C., Broekhuizen, T., Bart, Y., Bhattacharya, A., Dong, J. Q., Fabian, N., & Haenlein, M. (2021). Digital transformation: A multidisciplinary reflection and research agenda. *Journal of Business Research*, 122, 889–901. <https://doi.org/10.1016/j.jbusres.2019.09.022>
- Sommerville, I. (2020). *Engineering software products: An introduction to modern software engineering*. Pearson.
- Autoridad Nacional para la Innovación Gubernamental. (2021). *Agenda Digital Nacional 2021*. <https://aig.gob.pa/descargas/2019/06/agenda-digital-2021-aig.pdf>

Autoridad Nacional de Transparencia y Acceso a la Información. (2024). *Guía para cumplir la normativa sobre protección de datos personales (sector privado)*.

<https://antai.gob.pa/wp-content/uploads/2024/01/GUIA-SECTOR-PRIVADO-1.pdf>

Cubilla-Bonnetier, D., Grajales-Barrios, M., Ortega-Espinosa, A., Puertas, L., & De León

Sautú, N. (2023). Unequal literacy development and access to online education in public versus private Panamanian schools during COVID-19 pandemic.

Frontiers in Education, 8, Artículo 989872.

<https://doi.org/10.3389/feduc.2023.989872>

Ministerio de Educación. (2025). *Plan Estratégico de Educación 2025–2029*.

https://www.meduca.gob.pa/wp-content/uploads/2025/07/Plan_Estrategico_2025-2029.pdf

Ley N.º 81 de 26 de marzo de 2019. Sobre protección de datos personales. *Gaceta Oficial Digital* No. 28743-A. 29 de marzo de 2019.

https://www.gacetaoficial.gob.pa/pdfTemp/28743_A/GacetaNo_28743a_20190329.pdf

Decreto Ejecutivo N.º 285 de 28 de mayo de 2021. Que reglamenta la Ley 81 de 2019 sobre protección de datos personales. *Gaceta Oficial Digital* No. 29296-A. 28 de mayo de 2021. https://www.gacetaoficial.gob.pa/pdfTemp/29296_A/85281.pdf

UNESCO. (2021). *Reimagining our futures together: A new social contract for education*. UNESCO Publishing.

- Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2018). *Fundamentals of business process management* (2.^a ed.). Springer. <https://doi.org/10.1007/978-3-662-56509-4>
- Hammer, M. (2015). What is business process management? En J. vom Brocke & M. Rosemann (Eds.), *Handbook on business process management 1* (pp. 3–16). Springer. https://doi.org/10.1007/978-3-642-45100-3_1
- vom Brocke, J., & Mendling, J. (Eds.). (2018). *Business process management cases: Digital innovation and business transformation in practice*. Springer. <https://doi.org/10.1007/978-3-319-58307-5>
- AXELOS. (2019). *ITIL Foundation: ITIL 4 edition*. TSO (The Stationery Office).
- International Organization for Standardization. (2018). ISO/IEC 20000-1:2018 Information technology — Service management — Part 1: Service management system requirements.
- amaoka, H. (2019). Case study of implementing an IT service desk ticketing system at small computer center. En *Proceedings of the 2019 3rd International Conference on Information System and Data Mining* (pp. 58–62). ACM. <https://doi.org/10.1145/3347709.3347820>
- Zangari, A., Marcuzzo, M., Schiavinato, M., Gasparetto, A., & Albarelli, A. (2023). Ticket automation: An insight into current research with applications to multi-level classification scenarios. *Expert Systems with Applications*, 225, Artículo 119984. <https://doi.org/10.1016/j.eswa.2023.119984>

Bass, L., Clements, P., & Kazman, R. (2022). *Software architecture in practice* (4.^a ed.). Addison-Wesley.

Richards, M., & Ford, N. (2020). *Fundamentals of software architecture: An engineering approach*. O'Reilly Media.

International Organization for Standardization & Institute of Electrical and Electronics Engineers. (2022). *ISO/IEC/IEEE 42010:2022 Systems and software engineering — Architecture description*.

International Organization for Standardization. (2023). *ISO/IEC 25010:2023 Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — System and software quality models*.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.

OWASP Foundation. (2021). *OWASP Top 10: The ten most critical web application security risks*. <https://owasp.org/www-project-top-ten/>

Ley N.º 81 de 26 de marzo de 2019. Sobre protección de datos personales. *Gaceta Oficial Digital*, No. 28743-A (29 de marzo de 2019).

Decreto Ejecutivo N.º 285 de 28 de mayo de 2021. Que reglamenta la Ley 81 de 2019 sobre protección de datos personales. *Gaceta Oficial Digital*, No. 29296-A (28 de mayo de 2021).

Davenport, T. H., & Harris, J. G. (2017). *Competing on analytics: The new science of winning* (Ed. actualizada). Harvard Business Review Press.

International Organization for Standardization. (2015). *ISO 9001:2015 Quality management systems — Requirements*.

Ley N.º 81 de 26 de marzo de 2019. Sobre protección de datos personales. *Gaceta Oficial Digital No. 28743-A*. 29 de marzo de 2019.

https://www.gacetaoficial.gob.pa/pdfTemp/28743_A/GacetaNo_28743a_20190329.pdf

Decreto Ejecutivo N.º 285 de 28 de mayo de 2021. Que reglamenta la Ley 81 de 2019 sobre protección de datos personales. *Gaceta Oficial Digital No. 29296-A*. 28 de mayo de 2021. https://www.gacetaoficial.gob.pa/pdfTemp/29296_A/85281.pdf

Protección de datos personales en Panamá. (2024, 15 de agosto). En *Wikipedia*.

https://es.wikipedia.org/wiki/Protecci%C3%B3n_de_datos_personales

RSM Panamá. (2022). *Ley 81 protección de datos personales*.

<https://www.rsm.global/panama/es/insights/special-reports/ley-81-proteccion-de-datos-personales>

MASCO Services Corp. (2022). *Data protection policy*.

<https://maservicescorp.com/en/data-protection-policy/>