



# UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO



**FACULTAD DE INGENIERÍA CIVIL, DE SISTEMAS Y  
ARQUITECTURA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**TESIS**

Plataforma virtual para mejorar la gestión académica en la institución  
educativa particular San Isidro - Zaña, Lambayeque

**PARA OBTENER EL TÍTULO PROFESIONAL DE:**

Ingeniero de sistemas

**Autores:**

Bach. Luis Edgardo Acuña Guevara

Bach. Gerson Anthony Oliva Remigio

**Asesor:**

Dr. Ing. Jesús Bernardo Olavarría Paz - 0000-0001-6078-5223

**Lambayeque, Perú**

**2026**



# UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO



**FACULTAD DE INGENIERÍA CIVIL, DE SISTEMAS Y  
ARQUITECTURA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**TESIS**

Plataforma virtual para mejorar la gestión académica en la institución  
educativa particular San Isidro - Zaña, Lambayeque

**PARA OBTENER EL TÍTULO PROFESIONAL DE:**

Ingeniero de sistemas

**APROBADO POR EL SIGUIENTE JURADO**



**DR. ING. ERNESTO KARLO CELI AREVALO**  
PRESIDENTE



**DR. ING. REGIS JORGE ALBERTO DIAZ-PLAZA**  
SECRETARIO



**DRA. ING. MARIA DE LOS ANGELES GUZMAN VALLE**  
VOCAL



# UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO



**FACULTAD DE INGENIERÍA CIVIL, DE SISTEMAS Y  
ARQUITECTURA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

## **TESIS**

Plataforma virtual para mejorar la gestión académica en la institución  
educativa particular San Isidro - Zaña, Lambayeque

**PARA OBTENER EL TÍTULO PROFESIONAL DE:**

Ingeniero de sistemas

DR. ING. JESUS BERNARDO OLAVARRIA PAZ  
ASESOR

BACH. LUIS EDGARDO ACUÑA GUEVARA  
BACH. EN INGENIERÍA DE SISTEMAS

BACH. GERSON ANTHONY OLIVA REMIGIO  
BACH. EN INGENIERÍA DE SISTEMAS



**ACTA DE SUSTENTACIÓN  
 N° 635-2026-UI-FICSA**



Siendo las 12:00 m horas del día 19 de febrero del 2026, se reunieron de manera presencial los miembros de jurado de la tesis titulada: **“PLATAFORMA VIRTUAL PARA MEJORAR LA GESTIÓN ACADÉMICA EN LA INSTITUCIÓN EDUCATIVA PARTICULAR SAN ISIDRO - ZAÑA, LAMBAYEQUE”** con código N° IS\_V\_2025\_031, designado por Resolución Decanal N° 625-2025-UNPRG-FICSA con la finalidad de Evaluar y Calificar la sustentación de la tesis antes mencionada, conformado por los siguientes docentes:

<b>DR. ING. ERNESTO KARLO CELI AREVALO</b>	<b>PRESIDENTE</b>
<b>DR. ING. REGIS JORGE ALBERTO DIAZ PLAZA</b>	<b>SECRETARIO</b>
<b>DRA. ING. MARIA DE LOS ANGELES GUZMAN VALLE</b>	<b>VOCAL</b>

Asesorado por **DR. ING. JESUS BERNARDO OLAVARRIA PAZ.**

El acto de sustentación fue autorizado por **OFICIO VIRTUAL N° 37-2026-UIFICSA**, la tesis fue presentada y sustentada por los Bachilleres: **LUIS EDGARDO ACUÑA GUEVARA Y GERSON ANTHONY OLIVA REMIGIO**, tuvo una duración de 105 minutos. Después de la sustentación, y absueltas las preguntas y observaciones de los miembros del jurado; se procedió a la calificación respectiva:

	NUMERO	LETRAS	CALIFICATIVO
<b>LUIS EDGARDO ACUÑA GUEVARA</b>	<u>18</u>	<u>Dieciocho</u>	<u>MUY BUENO</u>
<b>GERSON ANTHONY OLIVA REMIGIO</b>	<u>18</u>	<u>Dieciocho</u>	<u>MUY BUENO</u>

Por lo que quedan **APTOS** para obtener el Título Profesional de **INGENIERO DE SISTEMAS** de acuerdo con la Ley Universitaria 30220 y la normatividad vigente de la Facultad de Ingeniería Civil De Sistemas y de Arquitectura de la Universidad Nacional Pedro Ruiz Gallo.

Siendo las 1:45 p.m. Se dio por concluido el presente acto académico, dándose conformidad al presente acto, con la firma de los miembros del jurado.

**DR. ING. ERNESTO KARLO CELI AREVALO**  
 PRESIDENTE

**DR. ING. REGIS JORGE ALBERTO DIAZ PLAZA**  
 SECRETARIO

**DRA. ING. MARIA DE LOS ANGELES GUZMAN VALLE**  
 VOCAL

**DR. ING. JESUS BERNARDO OLAVARRIA PAZ**  
 ASESOR



## CONSTANCIA DE VERIFICACION DE ORIGINALIDAD

Yo **Dr. Ing. Jesús Bernardo Olavarría Paz** usuario revisor de Tesis titulado:

**“PLATAFORMA VIRTUAL PARA MEJORAR LA GESTIÓN ACADÉMICA EN LA INSTITUCIÓN EDUCATIVA PARTICULAR SAN ISIDRO - ZAÑA, LAMBAYEQUE”**,

Cuyo autor (es) son:


- **Luis Edgardo Acuña Guevara; con DNI N° 76665698**
- **Gerson Anthony Oliva Remigio; con DNI N° 73390528;**

declaro que la evaluación realizada por el Programa informático, ha arrojado un porcentaje de similitud 17%, verificables en el Resumen del Reporte automatizado de similitudes que se acompaña.

El suscrito (a) analizó reporte y concluyó que cada una de las coincidencias detectadas dentro del porcentaje de similitud permitido no constituyen plagio y que el documento cumple con la integridad científica y con las normas para el uso de citas y referencias establecidas en los protocolos respectivos,

Se cumple con adjuntar el Recibo Digital a efectos de la trazabilidad respectiva del proceso.

Lambayeque, 06 de febrero del 2026

  
.....  
**Dr. Ing. Jesús Bernardo Olavarría Paz**  
DNI 40388780

Adjunta:

- *Resumen de Reporte automatizado de similitudes*
- *Recibo digital*

# Tesis

## INFORME DE ORIGINALIDAD

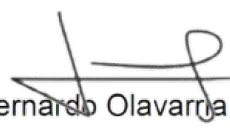
<b>17</b> %	<b>14</b> %	<b>7</b> %	<b>7</b> %
INDICE DE SIMILITUD	FUENTES DE INTERNET	PUBLICACIONES	TRABAJOS DEL ESTUDIANTE

## FUENTES PRIMARIAS


<b>1</b>	<b>hdl.handle.net</b> Fuente de Internet	<b>4</b> %
<b>2</b>	<b>Daza Rojas, John Jairo. "Modelo de Computación en la Nube Para el Telemonitoreo de Signos Vitales en Tiempo Real", Universidad Distrital Francisco José de Caldas (Colombia)</b> Publicación	<b>2</b> %
<b>3</b>	<b>repositorio.ucv.edu.pe</b> Fuente de Internet	<b>1</b> %
<b>4</b>	<b>revistapensamientotransformacional.editorialpiensadif</b> Fuente de Internet	<b>&lt;1</b> %
<b>5</b>	<b>Submitted to Universidad Nacional Pedro Ruiz Gallo</b> Trabajo del estudiante	<b>&lt;1</b> %
<b>6</b>	<b>repositorio.uwiener.edu.pe</b> Fuente de Internet	<b>&lt;1</b> %
<b>7</b>	<b>repositorio.unprg.edu.pe</b> Fuente de Internet	<b>&lt;1</b> %
<b>8</b>	<b>Submitted to Universidad Francisco de Vitoria</b> Trabajo del estudiante	<b>&lt;1</b> %
<b>9</b>	<b>Submitted to Ilerna Online</b> Trabajo del estudiante	<b>&lt;1</b> %
<b>10</b>	<b>www.coursehero.com</b> Fuente de Internet	<b>&lt;1</b> %
<b>11</b>	<b>Submitted to Universidad Nacional Jorge Basadre Grohmann</b> Trabajo del estudiante	<b>&lt;1</b> %
<b>12</b>	<b>Submitted to Universidad Pablo de Olavide</b> Trabajo del estudiante	<b>&lt;1</b> %

Ing. Jesús Bernardo Olavarría Paz  
Asesor

13	Submitted to Universidad Andina del Cusco Trabajo del estudiante	<1 %
14	Submitted to Universidad San Ignacio de Loyola Trabajo del estudiante	<1 %
15	repositorio.uladech.edu.pe Fuente de Internet	<1 %
16	Submitted to Universidad Católica Boliviana "San Pablo" Trabajo del estudiante	<1 %
17	Submitted to Universidad Tecnológica del Perú Trabajo del estudiante	<1 %
18	repositorio.uss.edu.pe Fuente de Internet	<1 %
19	Submitted to Ilerna Online Blackboard Trabajo del estudiante	<1 %
20	Alex Pacheco, Renato Yupanqui, Diana Mogrovejo, Joel Garay, Yrene Uribe-Hernández. "Impact of digitization on educational management: Results of the introduction of a learning management system in a traditional school context", Computers in Human Behavior Reports, 2025 Publicación	<1 %
21	Submitted to Universitat Politècnica de València Trabajo del estudiante	<1 %
22	repositorio.uch.edu.pe Fuente de Internet	<1 %
23	repositorio.uct.edu.pe Fuente de Internet	<1 %
24	Submitted to Universidad TecMilenio Trabajo del estudiante	<1 %
25	issuu.com Fuente de Internet	<1 %
26	repositorio.continental.edu.pe Fuente de Internet	<1 %

  
 Ing. Jesús Bernardo Olavarría Paz  
 Asesor

27	<a href="http://alicia.concytec.gob.pe">alicia.concytec.gob.pe</a> Fuente de Internet	<1 %
28	Submitted to ucb Trabajo del estudiante	<1 %
29	<a href="http://www.repositorio.usac.edu.gt">www.repositorio.usac.edu.gt</a> Fuente de Internet	<1 %
30	<a href="http://es.scribd.com">es.scribd.com</a> Fuente de Internet	<1 %
31	<a href="http://repositorioacademico.upc.edu.pe">repositorioacademico.upc.edu.pe</a> Fuente de Internet	<1 %
32	"Identificación de fuentes contaminantes en suelos y polvos de calle: el caso de Copiapó, Chile", Pontificia Universidad Católica de Chile, 2022 Publicación	<1 %
33	Submitted to Universidad Politécnica de Madrid Trabajo del estudiante	<1 %
34	<a href="http://dspace.esPOCH.edu.ec">dspace.esPOCH.edu.ec</a> Fuente de Internet	<1 %
35	<a href="http://idus.us.es">idus.us.es</a> Fuente de Internet	<1 %
36	<a href="http://prezi.com">prezi.com</a> Fuente de Internet	<1 %
37	<a href="http://repositorio.upla.edu.pe">repositorio.upla.edu.pe</a> Fuente de Internet	<1 %
38	<a href="http://www.mef.gob.pe">www.mef.gob.pe</a> Fuente de Internet	<1 %
39	<a href="http://atsgestion.net">atsgestion.net</a> Fuente de Internet	<1 %
40	<a href="http://1library.co">1library.co</a> Fuente de Internet	<1 %
41	<a href="http://repositorio.uap.edu.pe">repositorio.uap.edu.pe</a> Fuente de Internet	<1 %
42	<a href="http://repositorio.unibe.edu.ec">repositorio.unibe.edu.ec</a> Fuente de Internet	<1 %

  
 Ing. Jesús Bernardo Olavarría Paz  
 Asesor

43	<a href="https://repositorio.unp.edu.pe">repositorio.unp.edu.pe</a> Fuente de Internet	<1 %
44	Dyer, Karen. "The Relationship Between Resilience and Adaptability as Demonstrated by Global Tourism Leaders in the United States and Ecuador.", Saint Mary of the Woods College Publicación	<1 %
45	Submitted to Universidad Andina Nestor Caceres Velasquez Trabajo del estudiante	<1 %
46	Submitted to Universidad Cesar Vallejo Trabajo del estudiante	<1 %
47	Submitted to Universidad Internacional Isabel I de Castilla Trabajo del estudiante	<1 %
48	<a href="http://uvadoc.uva.es">uvadoc.uva.es</a> Fuente de Internet	<1 %
49	<a href="http://yhon-alejandro.blogspot.com">yhon-alejandro.blogspot.com</a> Fuente de Internet	<1 %
50	<a href="http://digibug.ugr.es">digibug.ugr.es</a> Fuente de Internet	<1 %
51	<a href="http://ecosur.repositorioinstitucional.mx">ecosur.repositorioinstitucional.mx</a> Fuente de Internet	<1 %
52	<a href="https://repositorio.uncp.edu.pe">repositorio.uncp.edu.pe</a> Fuente de Internet	<1 %
53	Olivares, Victor Manuel Garcia. "Geography, Genomics, and Evolutionary Success in an Insular Environment", Universidad de La Laguna (Canary Islands, Spain), 2022 Publicación	<1 %
54	Submitted to Universidad Camilo José Cela Trabajo del estudiante	<1 %
55	Submitted to Universidad Politécnica del Perú Trabajo del estudiante	<1 %
56	Submitted to Universidad de Navarra Trabajo del estudiante	<1 %

  
 Ing. Jesús Bernardo Olavarría Paz  
 Asesor





## Recibo digital

Este recibo confirma que su trabajo ha sido recibido por Turnitin. A continuación podrá ver la información del recibo con respecto a su entrega.

La primera página de tus entregas se muestra abajo.

Autor de la entrega: Gerson Oliva  
Título del ejercicio: Quick Submit  
Título de la entrega: Tesis  
Nombre del archivo: INFORME\_FINAL.pdf  
Tamaño del archivo: 21.52M  
Total páginas: 164  
Total de palabras: 17,297  
Total de caracteres: 120,069  
Fecha de entrega: 06-feb-2026 07:43a. m. (UTC-0500)  
Identificador de la entrega: 2872482013

 UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO  
FACULTAD DE INGENIERÍA CIVIL, DE SISTEMAS Y ARQUITECTURA  
UNIDAD DE INVESTIGACIÓN 

**“PLATAFORMA VIRTUAL PARA MEJORAR LA GESTIÓN ACADÉMICA EN LA INSTITUCIÓN EDUCATIVA PARTICULAR SAN ISIDRO - ZAÑA, LAMBAYEQUE”**

Tesis presentada para optar el título profesional de Ingeniero de Sistemas

**Autores**  
Luis Edgardo Acuña Guevara  
Gerson Anthony Oliva Remigio

**Asesor**  
Dr. Ing. Jesús Bernardo Olavarría Paz

**Línea de investigación**  
Ingenierías y Tecnologías

**Lugar de ejecución**  
Zaña, Chiclayo, Lambayeque


**Duración estimada del proyecto**  
06 meses

**Fecha de inicio**  
01/06/2025

**Fecha de término**  
30/11/2025

---

1

  
Dr. Ing. Jesús B. Olavarría Paz  
Asesor

## DEDICATORIA

A Dios, por brindarme salud, fortaleza y sabiduría para afrontar cada reto durante mi formación profesional; a mi familia, por su apoyo constante, su confianza y por inculcarme desde siempre el valor del esfuerzo y la perseverancia; y a mí mismo, por la constancia, el aprendizaje autodidacta y la determinación para desarrollar una solución tecnológica orientada a mejorar la educación, reafirmando mi compromiso con la innovación y el impacto social.

*Luis Edgardo Acuña Guevara*

A quienes me han acompañado durante este camino, quienes empezaron y quienes aún siguen apoyándome. Su confianza y consejos han convertido en mí la calidad de persona y profesional del cual estoy orgulloso. Papá y mamá esto es por y para ustedes, cada paso diario es pensando en ustedes dos.

*Gerson Anthony Oliva Remigio*

## **AGRADECIMIENTO**

Agradecemos en primer lugar a Dios, por habernos brindado salud, sabiduría y la fortaleza necesaria para culminar nuestra etapa de estudiantes y hoy desempeñándose como profesionales hasta llegar al día de hoy cerrando esa etapa que empezaron en las aulas.

A nuestras familias, por su amor incondicional, paciencia y constante apoyo en todo sentido, que fueron fundamentales durante este proceso.

A nuestros docentes que tuvimos durante el transcurso de toda nuestra carrera como estudiantes y por sobre todo a nuestro asesor Dr. Ing. Jesús Bernardo Olavarría Paz, por su dedicación, preocupación demostrada, su guía, orientación académica y por compartir con nosotros sus conocimientos con compromiso y profesionalismo.

A nuestros amigos y compañeros, por su compañía, palabras de aliento y colaboración en los momentos más desafiantes durante todo este camino.

Finalmente, a todas las personas que, directa o indirectamente, contribuyeron a la realización de este trabajo de investigación, les extendemos nuestro más profundo agradecimiento.

## RESUMEN

El presente proyecto de investigación tuvo como objetivo desarrollar una plataforma virtual para mejorar la gestión académica en la Institución Educativa Particular San Isidro, ubicada en el distrito de Zaña, región Lambayeque. La institución presentaba limitaciones en el manejo de la información académica, tales como el registro manual de datos, el seguimiento del desempeño estudiantil y la gestión de actividades escolares, lo que dificultaba la eficiencia administrativa y la toma de decisiones oportunas.

Ante esta problemática, se propuso el diseño e implementación de una plataforma académica multiplataforma denominada **EDUNOVA**, orientada a integrar los procesos académicos y administrativos en un solo sistema. La solución fue desarrollada utilizando tecnologías modernas como PostgreSQL, Node.js, Express y React para el entorno web, y Flutter para el acceso desde dispositivos móviles, permitiendo el uso remoto por parte de estudiantes, docentes y administradores.

El desarrollo del sistema se realizó bajo la metodología ágil Scrum, lo que permitió una implementación progresiva, flexible y alineada a las necesidades reales de la institución. La plataforma incluye módulos para la gestión de estudiantes, docentes, cursos, programación de clases, evaluaciones digitales, registro de calificaciones, distribución de material educativo y seguimiento académico.

Para validar el correcto funcionamiento del sistema, se aplicó un checklist de funcionalidades basado en el backlog del proyecto y un instrumento de validación con escala Likert, dirigido a especialistas de la institución educativa. Los resultados evidenciaron un alto nivel de conformidad respecto a la funcionalidad, usabilidad y utilidad del sistema, demostrando que la plataforma cumple con los objetivos propuestos.

Finalmente, se concluye que la plataforma desarrollada contribuye significativamente a optimizar la gestión académica, fortalecer el seguimiento del desempeño estudiantil y mejorar la calidad del servicio educativo en la Institución Educativa Particular San Isidro.

**Palabras clave:** gestión académica, sistema académico, plataforma virtual, sistema web, validación de software, educación.

## ABSTRACT

The present research project aimed to develop a virtual platform to improve academic management at the Private Educational Institution San Isidro, located in the district of Zaña, Lambayeque region. The institution faced limitations in academic information management, such as manual data recording, insufficient student performance tracking, and difficulties in managing academic activities, which affected administrative efficiency and decision-making processes.

In response to this problem, the design and implementation of a multiplatform academic system called **EDUNOVA** was proposed, integrating academic and administrative processes into a single platform. The solution was developed using modern technologies such as PostgreSQL, Node.js, Express, and React for the web environment, and Flutter for mobile access, enabling remote use by students, teachers, and administrators.

The system was developed under the Scrum agile methodology, allowing a progressive, flexible, and needs-oriented implementation. The platform includes modules for student and teacher management, course administration, class scheduling, digital assessments, grade recording, educational material distribution, and academic performance monitoring.

To verify the proper functioning of the system, a functionality checklist based on the project backlog and a validation instrument using a Likert scale were applied, involving an institutional specialist. The results showed a high level of agreement regarding the system's functionality, usability, and usefulness, confirming that the platform meets the proposed objectives.

In conclusion, the developed virtual platform significantly contributes to improving academic management, strengthening student performance monitoring, and enhancing the quality of educational services at the Private Educational Institution San Isidro.

**Keywords:** academic management, academic system, virtual platform, web system, software validation, education.

## ÍNDICE

<b>INTRODUCCIÓN.....</b>	<b>22</b>
<b>1. DISEÑO TEÓRICO.....</b>	<b>23</b>
1.1. Planteamiento de la investigación.....	23
1.1.1. Síntesis de la situación problemática.....	23
1.1.2. Formulación del problema de investigación.....	24
1.1.3. Objetivo general.....	24
1.1.4. Objetivos específicos.....	24
1.1.5. Glosario de términos.....	25
1.2. Diseño teórico.....	28
1.2.1. Antecedentes.....	28
1.2.2. Antecedentes a nivel nacional.....	28
1.2.3. Antecedentes a nivel regional.....	29
1.2.4. Aporte de la presente investigación.....	29
1.2.5. Bases teóricas.....	29
Registro de actividades escolares.....	30
Control de actividades escolares (calificaciones y asistencia).....	30
<b>2. DISEÑO METODOLÓGICO.....</b>	<b>32</b>
2.1. Procedimiento a seguir en la investigación.....	32
2.1.1. ¿Por qué es descriptiva?.....	32
2.1.2. ¿Por qué es propositiva?.....	32
2.1.3. ¿Por qué es no experimental?.....	32
2.2. Método de la investigación.....	32
2.2.1. Pretest.....	33
2.2.2. Intervención.....	33
2.2.3. Postest.....	33
2.3. Tabla de operacionalización de variable.....	34
2.4. Población y muestra.....	35
2.5. Técnicas, instrumentos, equipos y materiales.....	35
2.5.1. Técnica.....	35
2.5.2. Instrumento.....	35
2.5.3. Matriz de Validación.....	36
2.5.4. Criterios de Interpretación.....	36
2.6. Especificación y diseño técnico.....	37
2.6.1. Backlog del producto.....	37
2.6.2. Lista de requisitos funcionales y no funcionales.....	40
2.6.3. Casos de uso (diagramas UML).....	44
2.6.4. Descripción de los casos de uso.....	52
2.7. Actividades y recursos.....	71
2.7.1. Cronograma.....	71
2.7.2. Presupuesto.....	72
<b>3. DESARROLLO DE LA SOLUCIÓN.....</b>	<b>73</b>
3.1. Arquitectura del sistema.....	73
3.2. Diagrama de arquitectura del sistema.....	73

3.3. Arquitectura frontend.....	75
3.3.1. Características del frontend.....	75
3.3.2. Organización del frontend.....	75
3.4. Arquitectura del backend.....	77
3.4.1. Responsabilidades del backend.....	77
3.4.2. Organización del backend.....	77
3.5. Base de datos y persistencia.....	79
3.5.1. Modelo de datos.....	79
3.5.2. Acceso a datos.....	79
3.5.3. Base de datos relacional.....	80
3.6. Comunicación en tiempo real.....	81
3.7. Seguridad y control de accesos.....	81
3.8. Tecnologías y librerías utilizadas.....	81
3.9. Justificación de la arquitectura adoptada.....	83
3.10. Despliegue y entorno de ejecución.....	83
3.11. Código fuente.....	84
3.11.1. Funcionalidad 1: Registro de estudiantes en año académico.....	84
3.11.2. Funcionalidad 2: Registro de asistencia diaria de estudiantes.....	92
3.11.3. Funcionalidad 3: Registro de pagos del estudiante.....	97
3.11.4. Funcionalidad 4: Registro de notas del estudiante.....	98
3.12. Interfaz del sistema.....	99
3.12.1. Landing page.....	99
3.12.2. Plataforma educativa.....	106
<b>4. RESULTADOS.....</b>	<b>137</b>
4.1. Validación de la plataforma educativa propuesta.....	137
4.1.1. Objetivo de la validación.....	137
4.1.2. Aplicación del checklist de funcionalidades y del instrumento de validación....	137
4.1.3. Matriz de validación del sistema.....	149
VARIABLE 1: REGISTRO DE ACTIVIDADES ESCOLARES.....	149
VARIABLE 2: CONTROL DE ACTIVIDADES ESCOLARES.....	150
VARIABLE 3: SEGUIMIENTO DE ACTIVIDADES ESCOLARES.....	151
4.2. Resultados de la validación.....	152
<b>5. DISCUSIÓN DE RESULTADOS.....</b>	<b>154</b>
<b>6. CONCLUSIONES.....</b>	<b>155</b>
<b>7. RECOMENDACIONES.....</b>	<b>156</b>
<b>8. REFERENCIAS.....</b>	<b>157</b>
<b>9. ANEXOS.....</b>	<b>159</b>
ANEXO 1: CHECKLIST TÉCNICO DE VERIFICACIÓN DEL BACKLOG LA PLATAFORMA EDUCATIVA.....	159
ANEXO 2: INSTRUMENTO DE VALIDACIÓN DE LA PLATAFORMA EDUCATIVA.....	167
ANEXO 3: MATRIZ DE VALIDACIÓN DE LA PLATAFORMA EDUCATIVA.....	170

## ÍNDICE DE TABLAS

Tabla I: Tabla de operacionalización de variable (Fuente: Elaboración propia).....	34
Tabla II: Tabla de Backlog Usuario Administrador - Docente (Fuente: Elaboración propia).....	38
Tabla III: Tabla de Backlog Usuario Estudiante (Fuente: Elaboración propia).....	38
Tabla IV: Tabla de Backlog Usuario Visitante (Fuente: Elaboración propia).....	39
Tabla V: RF-01 - Registro de estudiantes (Fuente: Elaboración propia).....	40
Tabla VI: RF-02 - Inicio de sesión (Fuente: Elaboración propia).....	40
Tabla VII: RF-03 - Cierre de sesión (Fuente: Elaboración propia).....	40
Tabla VIII: RF-04 - Gestión de asistencia (Fuente: Elaboración propia).....	40
Tabla IX : RF-05 - Gestión de perfiles de usuario (Fuente: Elaboración propia).....	40
Tabla X: RF-06 - Matrícula en año académico (Fuente: Elaboración propia).....	40
Tabla XI: RF-07 - Visualización de clases (Fuente: Elaboración propia).....	40
Tabla XII: RF-08 - Resolución de cuestionarios (Fuente: Elaboración propia).....	41
Tabla XIII: RF-09 - Generación de notas (Fuente: Elaboración propia).....	41
Tabla XIV: RF-10 - Visualización de notas (Fuente: Elaboración propia).....	41
Tabla XV: RF-11 - Gestión de usuarios (admin) (Fuente: Elaboración propia).....	41
Tabla XVI: RF-12 - Gestión de contenidos (admin/docente) (Fuente: Elaboración propia).....	41
Tabla XVII: RF-13 - Notificaciones internas (Fuente: Elaboración propia).....	41
Tabla XVIII: RF-14 - Soporte multimedia (Fuente: Elaboración propia).....	41
Tabla XIX: RF-15 - Evaluación automatizada (Fuente: Elaboración propia).....	41
Tabla XX: RF-16 - Visualización de progreso (Fuente: Elaboración propia).....	42
Tabla XXI: RF-17 - Chat de soporte o mensajería interna (Fuente: Elaboración propia).....	42
Tabla XXII: RF-18 - Acceso restringido (Fuente: Elaboración propia).....	42
Tabla XXIII: RF-19 - Cierre de año escolar (Fuente: Elaboración propia).....	42
Tabla XXIV: RNF-01 - Disponibilidad del sistema (Fuente: Elaboración propia).....	42
Tabla XXV: RNF-02 - Escalabilidad (Fuente: Elaboración propia).....	42
Tabla XXVI: RNF-03 - Accesibilidad (Fuente: Elaboración propia).....	42
Tabla XXVII: RNF-04 - Rendimiento (Fuente: Elaboración propia).....	43
Tabla XXVIII: RNF-05 - Seguridad de la información (Fuente: Elaboración propia).....	43
Tabla XXIX: RNF-06 - Imagen institucional (Fuente: Elaboración propia).....	43
Tabla XXX: RNF-07 - Usabilidad (Fuente: Elaboración propia).....	43
Tabla XXXI: RNF-08 - Respaldo automático de datos (Fuente: Elaboración propia).....	43
Tabla XXXII: RNF-09 - Integridad de datos (Fuente: Elaboración propia).....	43
Tabla XXXIII: RNF-10 - Arquitectura de API REST (Fuente: Elaboración propia).....	43
Tabla XXXIV: Actor - Administrador de sistema (Fuente: Elaboración propia).....	44
Tabla XXXV: Actor - Administrador principal (Fuente: Elaboración propia).....	44
Tabla XXXVI: Actor - Docente (Fuente: Elaboración propia).....	44
Tabla XXXVII: Actor - Estudiante (Fuente: Elaboración propia).....	44
Tabla XXXVIII: CU-01 - Iniciar sesión (Fuente: Elaboración propia).....	52
Tabla XXXIX: CU-02 - Cierre de sesión (Fuente: Elaboración propia).....	52
Tabla XL: CU-03 - Editar datos de usuario (Fuente: Elaboración propia).....	53
Tabla XLI: CU-04 - Matrícula de nuevo estudiante en año académico (Fuente: Elaboración propia).....	53
Tabla XLII: CU-05 - Matrícula de estudiante de reingreso en año académico (Fuente:	

Elaboración propia).....	54
Tabla XLIII: CU-06 - Visualizar clases en formato diapositiva (Fuente: Elaboración propia)....	54
Tabla XLIV: CU-07 - Resolver cuestionario (Fuente: Elaboración propia).....	55
Tabla XLV: CU-08 - Ver calificaciones (Fuente: Elaboración propia).....	55
Tabla XLVI: CU-09 - Subir contenido educativo (Fuente: Elaboración propia).....	56
Tabla XLVII: CU-10 - Gestionar eventos institucionales (Fuente: Elaboración propia).....	56
Tabla XLVIII: CU-11 - Generar reportes académicos (Fuente: Elaboración propia).....	57
Tabla XLIX: CU-12 - Administrar usuarios (Fuente: Elaboración propia).....	57
Tabla L: CU-13 - Cambiar contraseña (Fuente: Elaboración propia).....	58
Tabla LI: CU-14 - Gestionar horarios de clases (Fuente: Elaboración propia).....	58
Tabla LII: CU-15 - Visualizar perfil (Fuente: Elaboración propia).....	59
Tabla LIII: CU-16 - Gestionar cursos (Fuente: Elaboración propia).....	59
Tabla LIV: CU-17 - Enviar notificaciones (Fuente: Elaboración propia).....	60
Tabla LV: CU-18 - Registrar asistencia diaria (Fuente: Elaboración propia).....	60
Tabla LVI: CU-19 - Ver historial de asistencia del estudiante (Fuente: Elaboración propia).....	61
Tabla LVII: CU-20 - Validar año académico aprobado (Fuente: Elaboración propia).....	61
Tabla LVIII: CU-21 - Gestionar categorías y conceptos de pago (Fuente: Elaboración propia)..	62
Tabla LIX: CU-22 - Asignar docentes por año académico (Fuente: Elaboración propia).....	62
Tabla LX: CU-23 - Participar en el foro académico (Fuente: Elaboración propia).....	63
Tabla LXI: CU-24- Gestionar sesiones en vivo (Fuente: Elaboración propia).....	63
Tabla LXII: CU-25 - Gestionar pagos del estudiante (Fuente: Elaboración propia).....	64
Tabla LXIII: CU-26 - Gestionar documentos del estudiante (Fuente: Elaboración propia).....	64
Tabla LXIV: CU-27 - Mantenimiento de áreas (asignaturas, documentos, niveles, grados) (Fuente: Elaboración propia).....	65
Tabla LXV: CU-28 - Filtrar ciclos (Fuente: Elaboración propia).....	65
Tabla LXVI: CU-29 - Ver historial de asistencia del estudiante (Fuente: Elaboración propia)...	66
Tabla LXVII: CU-30 - Editar notas (Fuente: Elaboración propia).....	66
Tabla LXVIII: CU-31 - Eliminar alumno (Fuente: Elaboración propia).....	67
Tabla LXIX: CU-32 - Listar grados y ver pagos del estudiante (Fuente: Elaboración propia)....	67
Tabla LXX: CU-33 - Gestionar estados de alumnos (Fuente: Elaboración propia).....	68
Tabla LXXI: CU-34 - Ver dashboard administrativo (Fuente: Elaboración propia).....	68
Tabla LXXII: CU-35 - Ver dashboard del estudiante (Fuente: Elaboración propia).....	69
Tabla LXXIII: CU-36 - Visualizar calendario de eventos (Fuente: Elaboración propia).....	69
Tabla LXXIV: CU-37 - Gestionar malla curricular (Fuente: Elaboración propia).....	70
Tabla LXXV: CU-38 - Gestionar roles y permisos (Fuente: Elaboración propia).....	70
Tabla LXXVI: Cronograma de desarrollo (Fuente: Elaboración propia).....	71
Tabla LXXVII: Presupuesto del Proyecto (Fuente: Elaboración propia).....	72
Tabla LXXVIII: Tabla de tecnologías usadas en el proyecto (Fuente: Elaboración propia).....	82
Tabla LXXIX: Matriz: Registro de Actividades Escolares (Fuente: Elaboración propia).....	149
Tabla LXXX: Matriz: Control de Actividades Escolares (Fuente: Elaboración propia).....	150
Tabla LXXXI: Matriz: Seguimiento de Actividades Escolares (Fuente: Elaboración propia)..	151
Tabla LXXXII: Tabla de resultados de la validación (Fuente: Elaboración propia).....	152

## ÍNDICE DE FIGURAS

Figura I: Banner principal de pantalla de inicio (Fuente: Elaboración propia).....	99
Figura II: Sección “Sobre Nosotros” (Fuente: Elaboración propia).....	100
Figura III: Sección “Niveles Educativos” (Fuente: Elaboración propia).....	101
Figura IV: Sección “Metodología y Valores” (Fuente: Elaboración propia).....	102
Figura V: Sección “Noticias y Eventos” (Fuente: Elaboración propia).....	103
Figura VI: Sección “Contáctanos” (Fuente: Elaboración propia).....	104
Figura VII: Login de ingreso a la Plataforma Educativa (Fuente: Elaboración propia).....	105
Figura VIII: Dashboard Administrativo - Tema oscuro (Fuente: Elaboración propia).....	106
Figura IX: Dashboard Administrativo - Tema claro (Fuente: Elaboración propia).....	107
Figura X: Categorías de Pago - Gestión de conceptos de pago (Fuente: Elaboración propia)...	108
Figura XI: Año Académico - Gestión de alumnos (Fuente: Elaboración propia).....	109
Figura XII: Año Académico - Gestión Académica (Fuente: Elaboración propia).....	110
Figura XIII: Año Académico - Horarios de clase (Fuente: Elaboración propia).....	111
Figura XIV: Año Académico - Asignación de Docentes (Fuente: Elaboración propia).....	112
Figura XV: Gestión de Asistencia (Fuente: Elaboración propia).....	113
Figura XVI: Gestión de Asistencia - Registro de Asistencia (Fuente: Elaboración propia).....	114
Figura XVII: Gestión de Asistencia - Historial de Asistencia (Fuente: Elaboración propia)....	115
Figura XVIII: Foro Académico (Fuente: Elaboración propia).....	116
Figura XIX: Clases en Vivo (Fuente: Elaboración propia).....	117
Figura XX: Malla Curricular - Gestión de asignaturas (Fuente: Elaboración propia).....	118
Figura XXI: Malla Curricular - Edición de contenido multimedia (Fuente: Elaboración propia)...	119
Figura XXII: Malla Curricular - Visualización de contenido multimedia (Fuente: Elaboración propia).....	120
Figura XXIII: Gestión de Notificaciones (Fuente: Elaboración propia).....	121
Figura XXIV: Mantenedores: Asignaturas, Documentos, Grados, Niveles, Eventos (Fuente: Elaboración propia).....	122
Figura XXV: Reportes - Administrativos (Fuente: Elaboración propia).....	123
Figura XXVI: Reportes - Académicos (Fuente: Elaboración propia).....	124
Figura XXVII: Usuarios y Permisos - Usuarios (Fuente: Elaboración propia).....	125
Figura XXVIII: Usuarios y Permisos - Tipos de usuario (Fuente: Elaboración propia).....	126
Figura XXIX: Usuarios y Permisos - Permisos (Fuente: Elaboración propia).....	127
Figura XXX: Dashboard Docente (Fuente: Elaboración propia).....	128
Figura XXXI: Dashboard Estudiante (Fuente: Elaboración propia).....	129
Figura XXXII: Mis Cursos (Fuente: Elaboración propia).....	130
Figura XXXIII: Foro Académico (Fuente: Elaboración propia).....	131
Figura XXXIV: Clases en Vivo (Fuente: Elaboración propia).....	132
Figura XXXV: Mis Pagos (Fuente: Elaboración propia).....	133
Figura XXXVI: Mis Calificaciones (Fuente: Elaboración propia).....	134
Figura XXXVII: Mis Notificaciones (Fuente: Elaboración propia).....	135
Figura XXXVIII: Chat AI (Fuente: Elaboración propia).....	136
Figura XXXVIII: ANEXO 1 - Checklist parte 1 (Fuente: Elaboración propia).....	138
Figura XXXIX: ANEXO 1 - Checklist parte 2 (Fuente: Elaboración propia).....	139

Figura XL: ANEXO 1 - Checklist parte 3 (Fuente: Elaboración propia).....	140
Figura XLI: ANEXO 1 - Checklist parte 4 (Fuente: Elaboración propia).....	141
Figura XLII: ANEXO 1 - Checklist parte 5 (Fuente: Elaboración propia).....	142
Figura XLIII: ANEXO 1 - Checklist parte 6 (Fuente: Elaboración propia).....	143
Figura XLIV: ANEXO 1 - Checklist parte 7 (Fuente: Elaboración propia).....	144
Figura XLV: ANEXO 1 - Checklist parte 8 (Fuente: Elaboración propia).....	145
Figura XLVI: ANEXO 2 - Instrumento de validación en escala Likert parte 1 (Fuente: Elaboración propia).....	146
Figura XLVII: ANEXO 2 - Instrumento de validación en escala Likert parte 2 (Fuente: Elaboración propia).....	147
Figura XLVIII: ANEXO 2 - Instrumento de validación en escala Likert parte 3 (Fuente: Elaboración propia).....	148

## ÍNDICE DE ILUSTRACIONES

Ilustración I: Diagrama de actores (Fuente: Elaboración propia).....	45
Ilustración II: Casos de Uso del Administrador del sistema (Fuente: Elaboración propia).....	47
Ilustración III: Casos de Uso del Administrador principal (Fuente: Elaboración propia).....	49
Ilustración IV: Casos de Uso del Docente (Fuente: Elaboración propia).....	50
Ilustración V: Casos de Uso del Estudiante (Fuente: Elaboración propia).....	51
Ilustración VI: Diagrama de arquitectura de software (Fuente: Elaboración propia).....	74
Ilustración VII: Estructura de directorios del frontend (Fuente: Elaboración propia).....	76
Ilustración VIII: Estructura de directorios del backend (Fuente: Elaboración propia).....	78
Ilustración IX: Modelo Entidad Relación (Fuente: Elaboración propia).....	80
Ilustración X: Código fuente de registro de estudiante (Fuente: Elaboración propia).....	84
Ilustración XI: Código fuente de modelo User - campos (Fuente: Elaboración propia).....	85
Ilustración XII: Código fuente de modelo User - relaciones (Fuente: Elaboración propia).....	86
Ilustración XIII: Código fuente de modelo Student - campos (Fuente: Elaboración propia).....	86
Ilustración XIV: Código fuente de modelo Student - relaciones (Fuente: Elaboración propia).....	87
Ilustración XV: Código fuente de modelo PaymentConcept - campos (Fuente: Elaboración propia).....	87
Ilustración XVI: Código fuente de modelo PaymentConcept - relaciones (Fuente: Elaboración propia).....	88
Ilustración XVII: Código fuente de modelo RegisteredPayment - campos (Fuente: Elaboración propia).....	88
Ilustración XVIII: Código fuente de modelo RegisteredPayment - relaciones (Fuente: Elaboración propia).....	89
Ilustración XIX: Código fuente de modelo SyllabuSubjectGrade - campos (Fuente: Elaboración propia).....	89
Ilustración XX: Código fuente de modelo SyllabuSubjectGrade - relaciones (Fuente: Elaboración propia).....	90
Ilustración XXI: Código fuente de modelo Course - campos (Fuente: Elaboración propia).....	90
Ilustración XXII: Código fuente de modelo Course - relaciones (Fuente: Elaboración propia).....	91
Ilustración XXIII: Código fuente del método del controlador Student (Fuente: Elaboración propia).....	91

propia).....	91
Ilustración XXIV: Código fuente del registro de asistencia diaria (Fuente: Elaboración propia).....	92
Ilustración XXV: Código fuente del modelo DailyAttendance (Fuente: Elaboración propia).....	93
Ilustración XXVI: Código fuente de consulta de asistencia de estudiante (Fuente: Elaboración propia).....	94
Ilustración XXVII: Código fuente del modelo AcademicCycle - campos (Fuente: Elaboración propia).....	95
Ilustración XXVIII: Código fuente del modelo AcademicCycle - relaciones (Fuente: Elaboración propia).....	96
Ilustración XXIX: Código fuente de consulta de asistencia de estudiante (Fuente: Elaboración propia).....	96
Ilustración XXX: Código fuente de los métodos del controlador de RegisteredPayment (Fuente: Elaboración propia).....	97
Ilustración XXXI: Código fuente del registro y consulta de pagos del estudiante (Fuente: Elaboración propia).....	97
Ilustración XXXII: Código fuente del registro y consulta de notas del estudiante por curso (Fuente: Elaboración propia).....	98
Ilustración XXXIII: Código fuente del método del controlador de Courses (Fuente: Elaboración propia).....	98

## INTRODUCCIÓN

En la actualidad, el uso de las Tecnologías de la Información y Comunicación (TIC) es fundamental para fortalecer los procesos educativos y administrativos en las instituciones educativas. La gestión académica, en particular, requiere herramientas tecnológicas que permitan organizar y acceder a la información de manera eficiente y confiable, contribuyendo a la mejora del servicio educativo.

La Institución Educativa Particular San Isidro, ubicada en el distrito de Zaña, región Lambayeque, presenta una realidad problemática relacionada con la gestión de procesos académicos como el registro de estudiantes y docentes, el control de calificaciones, el seguimiento del desempeño académico y la distribución de materiales educativos. Estos procesos se realizan de forma parcial o manual, generando demoras y limitaciones en el acceso oportuno a la información.

Ante esta situación, surge el problema de investigación orientado a determinar cómo una plataforma virtual puede mejorar la gestión académica en la Institución Educativa Particular San Isidro. La investigación se enmarca dentro de la línea de desarrollo de sistemas de información, considerando el contexto social y educativo actual, donde resulta imprescindible el uso de soluciones tecnológicas que optimicen los procesos institucionales.

La investigación se justifica desde el ámbito tecnológico, al proponer una solución que automatiza los procesos académicos; institucional, al mejorar la organización y el control de la información; y social, al facilitar el acceso a la información académica por parte de docentes y estudiantes.

El objetivo general es desarrollar una plataforma virtual para mejorar la gestión académica en la Institución Educativa Particular San Isidro, siendo los objetivos específicos el análisis de los procesos actuales, el diseño del sistema, el desarrollo de los módulos funcionales y la validación de su funcionamiento.

El informe se estructura en capítulos que comprenden el planteamiento del problema, el marco teórico, el diseño metodológico, el desarrollo del sistema, la validación, los resultados, las conclusiones y las recomendaciones.

## **1. DISEÑO TEÓRICO**

### **1.1. Planteamiento de la investigación**

#### **1.1.1. Síntesis de la situación problemática**

La Institución Educativa Particular San Isidro, ubicada en el distrito de Zaña, presenta limitaciones significativas en la gestión de sus procesos académicos, lo cual repercute en la eficiencia administrativa y en la calidad del servicio educativo. Actualmente, el registro de estudiantes, docentes y cursos se realiza de manera manual, a través de documentos físicos y procedimientos dispersos. Este sistema incrementa el riesgo de errores en la inscripción, duplicidad de datos y pérdida de información, lo que genera retrasos en la actualización y disponibilidad de los registros académicos.

El control de calificaciones y asistencia también enfrenta serias deficiencias. Al no contar con un sistema centralizado, los docentes deben consolidar notas y reportes manualmente, lo que provoca demoras en la entrega de resultados y dificulta el acceso oportuno a la información académica. Esta situación afecta la transparencia de los procesos de evaluación y limita la capacidad de los estudiantes y sus familias para conocer con prontitud el rendimiento académico.

De igual forma, la institución carece de un mecanismo formal que permita realizar un seguimiento académico integral. No existe una herramienta que facilite a los docentes monitorear de manera sistemática el progreso individual de cada estudiante a lo largo del período escolar. La ausencia de esta funcionalidad impide identificar con anticipación dificultades en el aprendizaje, limita la retroalimentación oportuna y restringe la toma de decisiones pedagógicas basadas en datos.

Estas deficiencias en el registro, control y seguimiento académico generan una sobrecarga de trabajo administrativo, reducen la eficiencia institucional y afectan la calidad del proceso formativo. Por ello, surge la necesidad de implementar una plataforma tecnológica que permita digitalizar y centralizar la información, integrando estos procesos clave en un solo sistema. De esta manera, se busca optimizar la gestión académica, reducir errores, agilizar el acceso a la información y fortalecer el acompañamiento al estudiante, contribuyendo a la mejora continua de la institución educativa.

### **1.1.2. Formulación del problema de investigación**

¿De qué manera una plataforma virtual mejora el registro, control y seguimiento de actividades escolares en la Institución Educativa Particular San Isidro, ubicada en el distrito de Zaña, región Lambayeque?

### **1.1.3. Objetivo general**

Desarrollar una plataforma virtual para mejorar el registro, control y seguimiento de actividades escolares en la Institución Educativa Particular San Isidro, ubicada en el distrito de Zaña, región Lambayeque.

### **1.1.4. Objetivos específicos**

- Diseñar e implementar un módulo de registro digital de estudiantes, docentes y cursos, que permita reducir errores administrativos y asegurar la disponibilidad de información actualizada en la Institución Educativa Particular San Isidro.
- Desarrollar un sistema centralizado para el control de calificaciones y asistencia, con el fin de agilizar la consolidación de notas y mejorar el acceso oportuno a la información académica por parte de docentes y estudiantes.
- Incorporar un módulo de seguimiento académico, que facilite a los docentes monitorear el progreso individual de los estudiantes y apoyar la toma de decisiones pedagógicas basadas en datos.
- Integrar un chat con inteligencia artificial como tutor virtual, orientado a brindar asistencia personalizada a los estudiantes en el uso de la plataforma y en la resolución de dudas académicas.
- Implementar un foro académico en línea, que promueva la interacción entre docentes y estudiantes, fomentando el aprendizaje colaborativo y la construcción de comunidad educativa.
- Evaluar el impacto de la plataforma en los procesos de registro, control y seguimiento académico, para determinar su efectividad en la optimización de la gestión escolar en la institución.

### 1.1.5. Glosario de términos

**Aplicativo móvil:**

Aplicación informática diseñada para ejecutarse en dispositivos móviles como teléfonos inteligentes o tabletas, que permite a los usuarios acceder a funcionalidades del sistema académico desde cualquier lugar con conexión a internet.

**Backlog:**

Listado priorizado de funcionalidades, tareas y requerimientos del sistema, utilizado como base para el desarrollo y validación del software.

**Base de datos:**

Conjunto organizado de datos almacenados electrónicamente, que permite el registro, consulta y gestión eficiente de la información académica.

**Calificación cualitativa:**

Sistema de evaluación basado en letras (AD, A, B, C) que representa el nivel de logro del estudiante según criterios establecidos por la institución educativa.

**Calificación cuantitativa:**

Sistema de evaluación numérica, generalmente expresado en una escala de 0 a 20, utilizado para medir el rendimiento académico del estudiante.

**Checklist:**

Lista de verificación utilizada para comprobar el cumplimiento de funcionalidades y tareas del sistema según lo establecido en el backlog del proyecto.

**Escala Likert:**

Instrumento de medición que permite evaluar el nivel de aceptación o conformidad de un sistema mediante una escala de cinco niveles, desde “Muy en desacuerdo” hasta “Muy de acuerdo”.

**Flutter:**

Framework de desarrollo multiplataforma creado por Google, utilizado para la creación de aplicaciones móviles con una sola base de código.

**Gestión académica:**

Conjunto de procesos relacionados con la administración, control y seguimiento de actividades educativas, tales como matrícula, evaluación, asistencia y rendimiento estudiantil.

**EDUNOVA:**

Nombre proveniente de la frase “Educación Innova” asignado a la plataforma educativa virtual desarrollada en el presente proyecto, orientada a mejorar la gestión académica mediante la integración de módulos de registro, control y seguimiento de actividades escolares, accesible desde entornos web.

**Instrumento de validación:**

Herramienta diseñada para evaluar el funcionamiento, calidad y aceptación del sistema desarrollado, aplicada a expertos mediante criterios previamente definidos.

**Interoperabilidad:**

Capacidad de un sistema informático para intercambiar información y funcionar de manera integrada con otros sistemas o plataformas.

**Matriz de validación:**

Tabla que organiza y consolida los resultados obtenidos del instrumento de validación, permitiendo su análisis e interpretación.

**Metodología Scrum:**

Metodología ágil de desarrollo de software basada en iteraciones cortas e incrementales, orientada a la entrega continua de valor y adaptación a cambios.

**Multiplataforma:**

Característica de un sistema que permite su uso en distintos dispositivos y sistemas operativos, como computadoras y dispositivos móviles.

**Node.js:**

Entorno de ejecución para JavaScript en el lado del servidor, utilizado para el desarrollo del backend del sistema.

**Plataforma virtual:**

Sistema informático accesible vía internet que permite la gestión y desarrollo de actividades académicas de forma digital.

**PostgreSQL:**

Sistema de gestión de bases de datos relacional de código abierto, utilizado para el almacenamiento de la información del sistema académico.

**React:**

Biblioteca de JavaScript utilizada para el desarrollo de interfaces de usuario dinámicas y modernas.

**Roles de usuario:**

Conjunto de permisos y funcionalidades asignadas a los diferentes tipos de usuarios del sistema, tales como administrador, docente y estudiante.

**SIAGIE:**

Sistema de Información de Apoyo a la Gestión de la Institución Educativa, plataforma del Estado peruano que centraliza la información académica oficial de las instituciones educativas.

**Sistema académico:**

Conjunto de módulos y funcionalidades que permiten gestionar integralmente los procesos educativos y administrativos de una institución.

**Validación del sistema:**

Proceso mediante el cual se verifica que el sistema desarrollado cumple con los requerimientos funcionales y objetivos establecidos en el proyecto.

## **1.2. Diseño teórico**

### **1.2.1. Antecedentes**

Diversas investigaciones en el Perú han evidenciado la necesidad de optimizar los procesos de gestión académica mediante el uso de plataformas tecnológicas.

### **1.2.2. Antecedentes a nivel nacional**

En este sentido, Allcca (2020) desarrolló una propuesta de implementación de un sistema web de gestión académica en el Instituto de Educación Superior Tecnológico Público Santo Domingo de Guzmán – Ayacucho, cuyo objetivo fue mejorar los procesos administrativos y académicos a través de un sistema digital. La investigación concluyó que la institución requería con urgencia la adopción de herramientas tecnológicas para agilizar la administración de información y reducir tiempos de procesamiento.

Por otro lado, Asencio y Carranza (2020) implementaron un sistema de información web en la Institución Educativa N.º 88049 – Cascajal Bajo, sustentado en la metodología SCRUM y el framework PHP-Laravel. Los autores señalan que el sistema permitió a los padres de familia acceder de forma ágil al avance académico de los estudiantes, solucionando así las demoras en la entrega de información.

Flores y Guillén (2019) diseñaron un aplicativo web para la gestión académica de la Institución Educativa Privada Nuestra Señora de Fátima – Callao. Su investigación, basada en la metodología ICONIX, concluyó que la herramienta permitió disminuir significativamente el tiempo de generación de reportes y mejorar el rendimiento académico de los estudiantes, alcanzando un 99.65% de efectividad en el seguimiento académico.

Asimismo, Esperilla (2019) trabajó en un sistema multiplataforma para la Institución Educativa Jireh – Pachacamac (Lima), destacando la importancia del acceso móvil en la gestión educativa. Su estudio concluyó que la implementación de un sistema accesible desde diversos dispositivos optimiza los procesos de matrícula, evaluación y comunicación, reduciendo la brecha digital en sectores con limitado acceso a computadoras de escritorio.

En conjunto, estas investigaciones nacionales coinciden en que la incorporación de plataformas web o multiplataforma contribuye a mejorar la eficiencia administrativa y

la calidad educativa, sentando las bases para propuestas que integren tecnologías aún más modernas y accesibles.

### **1.2.3. Antecedentes a nivel regional**

En la región norte del Perú, distintas investigaciones han abordado la problemática de la gestión académica en instituciones educativas públicas y privadas. Se han identificado limitaciones en el control de matrículas, evaluaciones y seguimiento académico cuando estos procesos se realizan de forma manual, generando insatisfacción entre estudiantes, docentes y padres de familia.

Los estudios realizados en Piura y Tumbogrande muestran que la implementación de sistemas académicos web permitió mejorar la organización institucional, centralizar la información y brindar un servicio educativo más eficiente y confiable. Asimismo, se resalta que estas soluciones han facilitado la labor docente al proporcionar herramientas digitales que reducen tiempos de trabajo y fortalecen la comunicación con la comunidad educativa.

### **1.2.4. Aporte de la presente investigación**

Si bien los trabajos revisados evidencian resultados positivos, ninguno de ellos integra de manera completa el uso de tecnologías modernas como React, Node.js, PostgreSQL y Flutter para garantizar un sistema académico verdaderamente multiplataforma. Además, el presente proyecto añade un componente innovador: la implementación de un chatbot con inteligencia artificial que permitirá a los estudiantes resolver dudas de manera inmediata, potenciando la interacción y el acompañamiento académico en tiempo real.

### **1.2.5. Bases teóricas**

El diseño teórico de la investigación busca sustentar las variables relacionadas con la gestión académica en la Institución Educativa Particular San Isidro, distrito de Zaña, región Lambayeque. Estas variables son: registro de actividades escolares, control de actividades escolares, seguimiento de actividades escolares y la implementación de una plataforma virtual. Cada una de ellas será definida conceptualmente, delimitada operativamente y vinculada a los fundamentos teóricos que las sustentan, a fin de establecer su relevancia dentro del problema de investigación.

## **Registro de actividades escolares**

En la actualidad, el registro de estudiantes, docentes y cursos en la institución se realiza de forma manual mediante documentos físicos, lo que genera errores frecuentes, duplicidad de datos y pérdida de información.

Definición teórica: El registro escolar se entiende como el proceso de almacenamiento organizado de datos académicos que permite la consulta, actualización y preservación de información de manera segura.

Evaluación: Puede medirse cuantitativamente en términos de tiempo de registro, número de errores cometidos, duplicados o pérdida de datos, y cualitativamente en función de la satisfacción del usuario respecto a la rapidez y confiabilidad del sistema.

Fundamento teórico: La literatura sobre gestión educativa señala que la digitalización de los registros mejora la eficiencia y la precisión, fortaleciendo la confiabilidad institucional en la administración académica.

## **Control de actividades escolares (calificaciones y asistencia)**

El proceso de control en la institución depende de consolidaciones manuales de notas y asistencia, lo cual genera demoras en la entrega de resultados y acceso tardío a información clave.

Definición teórica: El control académico se refiere a la supervisión sistemática de la asistencia y el rendimiento estudiantil, asegurando la transparencia y la objetividad en los procesos de evaluación.

Evaluación: Puede analizarse por la rapidez en la generación de reportes, disponibilidad de resultados en tiempo real y confiabilidad de los registros consolidados.

Fundamento teórico: Los sistemas de información académica (SIA o SIS) son ampliamente reconocidos por centralizar procesos como el registro de calificaciones y asistencia, permitiendo emitir reportes inmediatos y facilitar la gestión de la evaluación escolar.

## **Seguimiento de actividades escolares**

Actualmente, la institución no cuenta con herramientas que permitan realizar un seguimiento integral del progreso académico de los estudiantes. Ello impide detectar tempranamente dificultades de aprendizaje y limita la retroalimentación pedagógica.

Definición teórica: El seguimiento académico consiste en el monitoreo continuo del desempeño estudiantil, con el objetivo de identificar patrones, prevenir rezagos y ofrecer acompañamiento oportuno.

Evaluación: Se mide mediante frecuencia de actualización de los reportes, identificación de alertas tempranas, análisis de tendencias en el rendimiento y acciones de retroalimentación derivadas del sistema.

Fundamento teórico: Los enfoques de learning analytics y minería de datos educativos respaldan que el seguimiento digitalizado permite toma de decisiones pedagógicas basadas en evidencia, lo que fortalece la calidad educativa.

## **Plataforma virtual**

La carencia de un sistema digital centralizado en la IEP San Isidro limita la integración de los procesos de registro, control y seguimiento.

Definición teórica: Una plataforma virtual es un sistema digital accesible desde diversos dispositivos (computadoras, tablets y smartphones) que integra procesos de gestión académica y pedagógica en un entorno unificado.

Evaluación: Se mide según usabilidad (facilidad de uso, interfaz intuitiva), accesibilidad desde distintos dispositivos, integración de módulos académicos y escalabilidad del sistema.

Fundamento teórico: Las plataformas educativas virtuales (LMS o entornos virtuales de aprendizaje) han demostrado ser herramientas que mejoran la comunicación, reducen la carga administrativa y fortalecen la interacción docente-estudiante, especialmente cuando se diseñan bajo un enfoque multiplataforma que garantiza acceso ubicuo.

## **2. DISEÑO METODOLÓGICO**

### **2.1. Procedimiento a seguir en la investigación**

La presente investigación adopta un diseño no experimental, de tipo descriptivo–propositivo, con enfoque mixto (cuantitativo y cualitativo).

#### **2.1.1. ¿Por qué es descriptiva?**

Se describe la situación actual de la gestión académica en la Institución Educativa Particular San Isidro, analizando de manera detallada las variables registro de actividades escolares, control de actividades escolares y seguimiento de actividades escolares. Esto implica caracterizar sus procesos, identificar deficiencias y documentar cómo se desarrollan antes de la intervención tecnológica.

#### **2.1.2. ¿Por qué es propositiva?**

A partir del diagnóstico, se propone e implementa una plataforma virtual como solución tecnológica para optimizar las variables mencionadas. La propuesta no se limita a describir, sino que plantea y ejecuta un modelo de mejora concreto, evaluando su impacto.

#### **2.1.3. ¿Por qué es no experimental?**

No se manipulan deliberadamente las variables independientes en un entorno controlado, sino que se observa y analiza el fenómeno en su contexto natural. La intervención (implementación de la plataforma) se realiza sobre la realidad existente, sin alterar las condiciones externas de la institución.

### **2.2. Método de la investigación**

Se empleará el método de pretest–posttest con un solo grupo, evaluando las variables registro, control y seguimiento de actividades escolares antes y después de la implementación de la plataforma virtual.

### **2.2.1. Pretest**

- Aplicación de instrumentos de medición (encuestas, análisis de logs, entrevistas) para obtener datos cuantitativos y cualitativos sobre el estado actual de las variables.
- Indicadores: tiempo promedio de registro, número de errores, rapidez en generación de reportes, nivel de satisfacción de usuarios.

### **2.2.2. Intervención**

- Desarrollo e implementación de la plataforma virtual multiplataforma con módulos de registro, control y seguimiento académico.

### **2.2.3. Postest**

- Reaplicación de los mismos instrumentos para medir cambios en los indicadores.
- Comparación de resultados pre y post implementación para determinar el impacto de la propuesta.

### 2.3. Tabla de operacionalización de variable

Variable	Definición conceptual	Definición operacional	Dimensiones	Indicadores	Escala de medición
Registro de actividades escolares	Proceso de recopilar, organizar y almacenar datos académicos como matrículas, notas, asistencia y actividades escolares en una plataforma digital, garantizando trazabilidad y disponibilidad de información (Ayala Pari, 2022).	Uso del sistema para digitalizar y centralizar el registro académico, reduciendo errores y pérdida de datos.	<ul style="list-style-type: none"> <li>- Rapidez del registro</li> <li>- Disponibilidad de información</li> </ul>	<ul style="list-style-type: none"> <li>- Tiempo promedio de registro</li> <li>- Nivel de accesibilidad a los registros</li> </ul>	Ordinal Likert: 1: Muy en desacuerdo 2: Desacuerdo 3: Ni en desacuerdo ni de acuerdo 4: De acuerdo 5: Muy de acuerdo
Control de actividades escolares	Conjunto de acciones que permiten supervisar y verificar el cumplimiento de las actividades académicas mediante reportes, alertas o consultas en el sistema, asegurando la calidad de los procesos educativos (Damasio Jara, 2021).	Consolidación de notas y asistencia mediante la plataforma, generando reportes centralizados y accesibles.	<ul style="list-style-type: none"> <li>- Gestión de calificaciones</li> <li>- Gestión de asistencia</li> <li>- Acceso a reportes</li> </ul>	<ul style="list-style-type: none"> <li>- Tiempo de consolidación de notas</li> <li>- Nivel de satisfacción</li> </ul>	Ordinal Likert: 1: Muy en desacuerdo 2: Desacuerdo 3: Ni en desacuerdo ni de acuerdo 4: De acuerdo 5: Muy de acuerdo
Seguimiento de actividades escolares	Actividad de monitoreo continuo del desempeño académico de los estudiantes para identificar avances, dificultades y brindar retroalimentación oportuna a través de una plataforma digital (Calderón Bailón, 2021).	Uso del sistema para generar alertas tempranas y reportes de progreso individual de los estudiantes.	<ul style="list-style-type: none"> <li>- Progreso académico</li> <li>- Alertas tempranas</li> </ul>	<ul style="list-style-type: none"> <li>- Nivel de avance académico registrado</li> <li>- Alertas generadas</li> </ul>	Ordinal Likert: 1: Muy en desacuerdo 2: Desacuerdo 3: Ni en desacuerdo ni de acuerdo 4: De acuerdo 5: Muy de acuerdo

Tabla I: Tabla de operacionalización de variable (Fuente: Elaboración propia)

## 2.4. Población y muestra

La funcionalidad de la plataforma virtual se validará con 55 usuarios (45 estudiantes, 8 docentes y 2 expertos), estos usuarios forman parte de los distintos niveles dentro de la institución educativa, los expertos estarán conformado por coordinadores o tutores quienes darán conformidad de la recopilación de datos.

La muestra garantiza diversidad en edad, rol y familiaridad con tecnologías digitales.

## 2.5. Técnicas, instrumentos, equipos y materiales

La validación del sistema académico desarrollado se realizó con la finalidad de comprobar el correcto funcionamiento de las funcionalidades propuestas y su contribución a la mejora de la gestión académica en la Institución Educativa Particular San Isidro. Para ello, se empleó un proceso de validación por juicio de experto, considerando que el proyecto no contempla una implementación en producción, sino la demostración funcional del sistema.

### 2.5.1. Técnica

**Juicio de Experto:** La validación del sistema se realizó mediante el juicio de un experto perteneciente a la institución educativa, seleccionado por su experiencia en gestión académica y conocimiento del contexto institucional. El experto evaluó cada uno de los ítems del instrumento, asignando un valor de acuerdo con su percepción sobre el cumplimiento y funcionamiento de las funcionalidades del sistema.

### 2.5.2. Instrumento

El instrumento de validación utilizado fueron 2: Checklist de funcionalidades del backlog que permite verificar la implementación funcional del sistema y además un cuestionario estructurado con escala Likert de cinco niveles, diseñado en función de las variables de estudio: **registro de actividades escolares, control de actividades escolares** y **seguimiento de actividades escolares**. Dicho instrumento permitió evaluar de manera sistemática las funcionalidades del sistema académico desde la perspectiva experta y de los roles involucrados: administrador, docente y estudiante.

Cada ítem del instrumento fue formulado en relación directa con las funcionalidades implementadas en el sistema, tales como el registro académico, la gestión de usuarios,

el control de evaluaciones, la generación de reportes, el seguimiento del rendimiento académico y el acceso a la plataforma desde entornos web, incluyendo el uso de un chatbot con inteligencia artificial para la atención de consultas académicas.

La escala de valoración empleada fue la siguiente:

- 1: Muy en desacuerdo
- 2: En desacuerdo
- 3: Ni de acuerdo ni en desacuerdo
- 4: De acuerdo
- 5: Muy de acuerdo

### **2.5.3. Matriz de Validación**

A partir de los resultados obtenidos del instrumento de validación, se elaborará una matriz de validación, la cual permitirá organizar y analizar la información de manera estructurada. Esta matriz relaciona las variables, dimensiones, indicadores e ítems del instrumento, junto con los puntajes obtenidos y el nivel de validación alcanzado por cada uno.

La matriz de validación permitirá evidenciar el grado de cumplimiento de los requerimientos funcionales del sistema académico, facilitando la interpretación de los resultados y su correspondencia con los objetivos específicos del proyecto.

### **2.5.4. Criterios de Interpretación**

Para la interpretación de los resultados se establecieron los siguientes niveles de validación, considerando el puntaje obtenido en la escala Likert:

- Nivel alto: 4.21 – 5.00
- Nivel medio: 3.41 – 4.20
- Nivel bajo: 2.61 – 3.40
- Nivel deficiente: 1.00 – 2.60

Dado que la validación fue realizada por un único experto, el puntaje obtenido para cada ítem corresponde directamente al valor asignado en el instrumento de validación.

## 2.6. Especificación y diseño técnico

### 2.6.1. Backlog del producto

Backlog de requerimientos para el desarrollo de la plataforma Edunova							
Nota: Comenzaremos con el Diseño de las interfaces para tener mapeadas las funcionalidades completas tanto para el usuario administrador como para todos los usuarios (clientes y negocio)							
ID	Tema	Como un ...	Necesito	Así podré	Notas	Prioridad	Status
PWA0001	Plataforma web	Usuario Administrador	Módulo Año académico	Crear,editar, eliminar año académico	Crear,editar, eliminar año académico	Crear,editar, eliminar año académico	PENDIENTE
PWA0002	Plataforma web	Usuario Administrador		Ver alumnos matriculados y poder matricular alumnos en el año académico	Ver pagos del alumno, poder agregar conceptos de pago, ver documentos y datos del alumno(grado, estado, etc)		PENDIENTE
PWA0003	Plataforma web	Usuario Administrador		Agregar tutor a la matricula de alumno			PENDIENTE
PWA0004	Plataforma web	Usuario Administrador		Cerrar año académico	Cerrar un año académico, debe abrir un siguiente año, para que los alumnos que aprobaron pasen al siguiente grado en el año siguiente		PENDIENTE
PWA0005	Plataforma web	Usuario Administrador	Módulo Asignaturas	Crear, editar eliminar asignaturas	Nombre, grado, tipo, imagen		PENDIENTE
PWA0006	Plataforma web	Usuario Administrador	Módulo Malla curricular	Crear, editar, eliminar malla curricular	Nombre		PENDIENTE
PWA0007	Plataforma web	Usuario Administrador		Agregar cursos a malla y poder eliminarlos	Asignar curso por grados y nombre de curso a mostrarse		PENDIENTE
PWA0008	Plataforma web	Usuario Administrador	Módulo Grados	Crear, editar , eliminar grados	Nombre		PENDIENTE
PWA0009	Plataforma web	Usuario Administrador		Asignar nivel educativo a grados			PENDIENTE
PWA0010	Plataforma web	Usuario Administrador	Módulo Nivel educativo	Crear, editar , eliminar nivel educativo			PENDIENTE
PWA0011	Plataforma web	Usuario Administrador	Módulo Multimedia	Gestionar por malla curricular	Una malla debe tener varios cursos por grados		PENDIENTE
PWA0012	Plataforma web	Usuario Administrador y Usuario Docente		Ver todos los cursos de una malla curricular, agregar botones de editar contenido y ver contenido del curso	Poder filtrar por grado		PENDIENTE
PWA0013	Plataforma web	Usuario Administrador y Usuario Docente		Módulo de editar contenido de curso, poder registrar sesiones por semana	Habilitar la creación de semanas por curso, tener en cuenta los tipos de sesiones(clase, cuestionario, examen)		PENDIENTE
PWA0014	Plataforma web	Usuario Administrador y Usuario Docente		Poder editar sesión, el nombre de la sesión, poder editar contenido de la sesión según los esquemas establecidos por la librería de formularios	Validar al cambio tipo de sesión, de clase a cuestionario o examen no se podría o no estaría habilitado		PENDIENTE
PWA0015	Plataforma web	Usuario Administrador y Usuario Docente		Módulo de ver contenido	Se muestra menu lateral con las semanas y sus respectivas sesiones guardadas y el contenido respectivo de cada sesión		PENDIENTE
PWA0016	Plataforma web	Usuario Administrador	Módulo Asignar docente	Módulo para asignar docente a curso por año académico abierto	Analizar bien el requerimiento y en qué servirá la data para futuro		PENDIENTE
PWA0017	Plataforma web	Usuario Administrador y Usuario Docente	Módulo Seguimiento	Poder filtrar alumnos por año académico y por grado de secundaria			PENDIENTE
PWA0018	Plataforma web	Usuario Administrador y Usuario Docente		Mostrar lista detalla de cursos por alumno	Mostrar Promedio, Nota de Conducta, Nota de Cuestionario, Nota de Trabajo Final, Recuperacion. Además se debe mostrar un porcentaje de avance del curso según el contenido total del curso		PENDIENTE

PWA0019	Plataforma web	Usuario Administrador y Usuario Docente	Módulo Seguimiento académico	Validar un botón de activar examen de recuperación si cumple regla de nota mínima requerida			PENDIENTE
PWA0020	Plataforma web	Usuario Administrador y Usuario Docente		Mostrar progreso de sesiones por curso	Mostrar de forma detallada el proceso de avance de clases, cuestionarios, exámenes y mostrar notas respectivas y fecha que se realizó		PENDIENTE
PWA0021	Plataforma web	Usuario Administrador y Usuario Docente		Habilitar formulario para editar de manera manual notas de alumnos	Validar acceso solo para algunos usuarios		PENDIENTE
PWA0022	Plataforma web	Usuario Administrador y Usuario Docente	Módulo Clases en vivo	Habilitar crud para guardar, editar clases en vivo	Enlaces para zoom u otra plataforma		PENDIENTE
PWA0023	Plataforma web	Usuario Administrador y Usuario Docente		Visualizar clases en vivo creadas			PENDIENTE
PWA0024	Plataforma web	Usuario Administrador y Usuario Docente	Módulo Notificaciones	Realizar crud de notificaciones	Por grado y por colegio - solo para año académico vigente		PENDIENTE
PWA0025	Plataforma web	Usuario Administrador y Usuario Docente	Módulo Foro académico	Administración de post de alumnos o docentes para su discusión en tiempo real	Manejo de notificaciones en tiempo real		PENDIENTE
PWA0026	Plataforma web	Usuario Administrador y Usuario Docente	Módulo Asistencias	Realizar registro de asistencia diaria por grado y nivel académico			
PWA0027	Plataforma web	Usuario Administrador y Usuario Docente		Visualizar registro de asistencia de manera detallada			PENDIENTE
PWA0028	Plataforma web	Usuario Administrador		Realizar crud de categoría de eventos			
PWA0029	Plataforma web	Usuario Administrador	Módulo Eventos	Realizar crud de eventos para poder gestionarlos	Estos eventos se mostrarán a los estudiantes y en el landing page de la institución		PENDIENTE
PWA0030	Plataforma web	Usuario Administrador	Módulo Reportes	Realizar descarga de informes administrativos y académicos. Reporte de estudiantes por grado Reporte de asistencia mensual Reporte de notas por grado Reporte de notas por estudiante			PENDIENTE
PWA0031	Plataforma web	Usuario Administrador	Módulo Dashboard	Resumen de datos importantes acerca de matrículas, asistencias, registros académicos y financieros	Visualización al iniciar sesión		PENDIENTE

Tabla II: Tabla de Backlog Usuario Administrador - Docente (Fuente: Elaboración propia)

Backlog de requerimientos para el desarrollo de la plataforma Edunova							
Nota: Comenzaremos con el Diseño de las interfaces para tener mapeadas las funcionalidades completas tanto para el usuario cliente como para todos los usuarios (clientes y negocio)							
ID	Tema	Como un ...	Necesito	Así podré	Notas	Prioridad	Status
PWE0001	Plataforma web	Usuario Estudiante	Módulo Cursos	Ver contenido del curso	Organizar contenido del curso por semanas, visualizar sesiones y cuestionarios		PENDIENTE
PWE0002	Plataforma web	Usuario Estudiante		Ver notas y avance del curso	Cuestionarios, trabajo final, conducta, promedio final. Avance de curso y mostrar resultado de notas y evaluación		PENDIENTE
PWE0003	Plataforma web	Usuario Estudiante	Módulo Clases en vivo	Ver clases extra y agregar material adjunto	Clases complementarios a cursos		PENDIENTE
PWE0004	Plataforma web	Usuario Estudiante	Módulo Mi perfil	Mostrar datos del alumno	Datos personales y datos académicos		PENDIENTE
PWE0005	Plataforma web	Usuario Estudiante		Poder actualizar datos			PENDIENTE
PWE0006	Plataforma web	Usuario Estudiante		Poder actualizar foto de perfil			PENDIENTE
PWE0007	Plataforma web	Usuario Estudiante		Actualizar contraseña			PENDIENTE
PWE0008	Plataforma web	Usuario Estudiante	Módulo Mis pagos	Visualizar pagos por año académico	Mostrar concepto, monto, estado y link de voucher		PENDIENTE
PWE0009	Plataforma web	Usuario Estudiante		Opción para enviar voucher			PENDIENTE
PWE0010	Plataforma web	Usuario Estudiante	Módulo Historial Académico	Mostrar notas de cursos por año académico	Mostrar todas las notas		PENDIENTE
PWE0011	Plataforma web	Usuario Estudiante		Poder descargar libreta de notas			PENDIENTE
PWE0012	Plataforma web	Usuario Estudiante	Módulo Mis notificaciones	Notificaciones de sistema en tiempo real			PENDIENTE
PWE0013	Plataforma web	Usuario Estudiante	Módulo Chatbot	Chatbot de IA generativa para ayuda en planificación de estudios y resolución de dudas educativas			PENDIENTE
PWE0014	Plataforma web	Usuario Estudiante	Módulo Foro académico	Ver de post de alumnos o docentes para su discusión en tiempo real	Manejo de notificaciones en tiempo real		PENDIENTE
PWE0014	Plataforma web	Usuario Estudiante	Módulo Dashboard	Resumen de asistencia y rendimiento académico, visualizar horario de clases por día y métricas globales del estudiante	Visualización al iniciar sesión		PENDIENTE

Tabla III: Tabla de Backlog Usuario Estudiante (Fuente: Elaboración propia)

Backlog de requerimientos para el desarrollo de la plataforma Edunova							
ID	Tema	Como un ...	Necesito	Así podré	Notas	Prioridad	Status
PWV0001	Plataforma web	Usuario Visitante	Banner de inicio	Mostrar carrusel de fotos relevantes de la institución			PENDIENTE
PWV0002	Plataforma web	Usuario Visitante	Sección Header	Definición de los menús del header	Se manejará los menús: Sobre nosotros, Niveles, Metodología, Noticias		PENDIENTE
PWV0003	Plataforma web	Usuario Visitante		Mostrar logo de institución			PENDIENTE
PWV0004	Plataforma web	Usuario Visitante		Mostrar botón de acceso para el login hacia la plataforma			PENDIENTE
PWV0005	Plataforma web	Usuario Visitante		Un menu desplegable en vista de dispositivo móvil	Para mejor experiencia de navegación sobre la landing		PENDIENTE
PWV0006	Plataforma web	Usuario Visitante	Sección Sobre nosotros	Mostrar un resumen sobre la identidad de la institución mediante un diseño amigable			PENDIENTE
PWV0007	Plataforma web	Usuario Visitante	Sección niveles	Dar a conocer los distintos niveles educativos que ofrece la institución educativa			PENDIENTE
PWV0008	Plataforma web	Usuario Visitante	Sección Metodología y Valores	Informar sobre la forma de trabajo de la institución			PENDIENTE
PWV0009	Plataforma web	Usuario Visitante	Sección Testimonios	Dar a conocer casos de éxito tanto de padres de familia así como de alumnos, que ayuden a reforzar el mensaje de la institución			PENDIENTE
PWV0010	Plataforma web	Usuario Visitante	Sección Noticias y Eventos	Informar sobre eventos importantes que realiza la institución			PENDIENTE
PWV0011	Plataforma web	Usuario Visitante	Sección Contáctanos	Formulario breve que permita conectar a padres familia u otros interesados, con la institución			PENDIENTE
PWV0012	Plataforma web	Usuario Visitante	Sección Footer	Sección mostrando breve mensaje, redes sociales, ítem de menús, formas de contacto			PENDIENTE

Tabla IV: Tabla de Backlog Usuario Visitante (Fuente: Elaboración propia)

## 2.6.2. Lista de requisitos funcionales y no funcionales

### Requisitos funcionales

RF-01	Registro de estudiantes
Permite registrar nuevos usuarios proporcionando su información personal.	

*Tabla V: RF-01 - Registro de estudiantes (Fuente: Elaboración propia)*

RF-02	Inicio de sesión
Permite que usuarios registrados accedan a la plataforma.	

*Tabla VI: RF-02 - Inicio de sesión (Fuente: Elaboración propia)*

RF-03	Cierre de sesión
Permite a los usuarios cerrar su sesión de forma segura.	

*Tabla VII: RF-03 - Cierre de sesión (Fuente: Elaboración propia)*

RF-04	Gestión de asistencia
Permite gestionar la asistencia de los estudiantes.	

*Tabla VIII: RF-04 - Gestión de asistencia (Fuente: Elaboración propia)*

RF-05	Gestión de perfiles de usuario
Permite a los usuarios editar su información personal y cambiar contraseña.	

*Tabla IX: RF-05 - Gestión de perfiles de usuario (Fuente: Elaboración propia)*

RF-06	Matrícula en año académico
Permite que el administrador matricule estudiantes en un año académico disponible.	

*Tabla X: RF-06 - Matrícula en año académico (Fuente: Elaboración propia)*

RF-07	Visualización de clases
Los estudiantes pueden ver las clases asignadas a su ciclo (diapositivas, videos, etc).	

*Tabla XI: RF-07 - Visualización de clases (Fuente: Elaboración propia)*

RF-08	Resolución de cuestionarios
Permite resolver cuestionarios interactivos y enviar respuestas.	

*Tabla XII: RF-08 - Resolución de cuestionarios (Fuente: Elaboración propia)*

RF-09	Generación de notas
El sistema genera las calificaciones de los estudiantes en función de sus respuestas.	

*Tabla XIII: RF-09 - Generación de notas (Fuente: Elaboración propia)*

RF-10	Visualización de notas
Los estudiantes pueden consultar sus calificaciones por curso.	

*Tabla XIV: RF-10 - Visualización de notas (Fuente: Elaboración propia)*

RF-11	Gestión de usuarios (admin)
El administrador puede crear, editar o eliminar cuentas de estudiantes y docentes.	

*Tabla XV: RF-11 - Gestión de usuarios (admin) (Fuente: Elaboración propia)*

RF-12	Gestión de contenidos (admin/docente)
Docentes o administradores pueden subir clases, cuestionarios y recursos.	

*Tabla XVI: RF-12 - Gestión de contenidos (admin/docente) (Fuente: Elaboración propia)*

RF-13	Notificaciones internas
Los usuarios reciben alertas sobre eventos académicos, foro académico o mensajes de la dirección del colegio.	

*Tabla XVII: RF-13 - Notificaciones internas (Fuente: Elaboración propia)*

RF-14	Soporte multimedia
Soporta carga y reproducción de imágenes, PDF, audio y video para clases.	

*Tabla XVIII: RF-14 - Soporte multimedia (Fuente: Elaboración propia)*

RF-15	Evaluación automatizada
Corrige automáticamente los cuestionarios.	

*Tabla XIX: RF-15 - Evaluación automatizada (Fuente: Elaboración propia)*

RF-16	Visualización de progreso
El estudiante puede ver su avance en los cursos (porcentaje completado).	

*Tabla XX: RF-16 - Visualización de progreso (Fuente: Elaboración propia)*

RF-17	Foro académico
Comunicación directa con docentes o soporte a través de un foro académico con publicaciones y respuestas.	

*Tabla XXI: RF-17 - Chat de soporte o mensajería interna (Fuente: Elaboración propia)*

RF-18	Acceso restringido
Solo usuarios autenticados pueden acceder a las funcionalidades.	

*Tabla XXII: RF-18 - Acceso restringido (Fuente: Elaboración propia)*

RF-19	Cierre de año académico
Finalización de año académico y preinscripción de alumnos aprobados en el siguiente año.	

*Tabla XXIII: RF-19 - Cierre de año escolar (Fuente: Elaboración propia)*

### **Requisitos no funcionales**

RNF-01	Disponibilidad del sistema
El sistema estará disponible 99.5% del tiempo (excepto en mantenimiento programado).	

*Tabla XXIV: RNF-01 - Disponibilidad del sistema (Fuente: Elaboración propia)*

RNF-02	Escalabilidad
Debe poder soportar un aumento progresivo en la cantidad de estudiantes y docentes.	

*Tabla XXV: RNF-02 - Escalabilidad (Fuente: Elaboración propia)*

RNF-03	Accesibilidad
El administrador puede crear, editar o eliminar cuentas de estudiantes y docentes.	

*Tabla XXVI: RNF-03 - Accesibilidad (Fuente: Elaboración propia)*

RNF-04	Rendimiento
Las páginas deben cargar en menos de 3 segundos con conexión estándar.	

*Tabla XXVII: RNF-04 - Rendimiento (Fuente: Elaboración propia)*

RNF-05	Seguridad de la información
La información personal de los usuarios debe almacenarse cifrada.	

*Tabla XXVIII: RNF-05 - Seguridad de la información (Fuente: Elaboración propia)*

RNF-06	Imagen institucional
Desarrollo de landing page para dar a conocer la institución educativa.	

*Tabla XXIX: RNF-06 - Imagen institucional (Fuente: Elaboración propia)*

RNF-07	Usabilidad
La plataforma será intuitiva para estudiantes de educación primaria y secundaria, además del personal de la institución educativa	

*Tabla XXX: RNF-07 - Usabilidad (Fuente: Elaboración propia)*

RNF-08	Respaldo automático de datos
Se realizará un backup automático de la base de datos.	

*Tabla XXXI: RNF-08 - Respaldo automático de datos (Fuente: Elaboración propia)*

RNF-09	Integridad de datos
El sistema implementará eliminación lógica (soft delete) para preservar el historial de datos. Las operaciones críticas como matriculación y registro de pagos utilizarán transacciones de base de datos para garantizar la consistencia.	

*Tabla XXXII: RNF-09 - Integridad de datos (Fuente: Elaboración propia)*

RNF-10	Arquitectura de API REST
El backend implementará una arquitectura API RESTful con endpoints estandarizados, respuestas en formato JSON, códigos de estado HTTP apropiados y paginación en listados, facilitando la integración con aplicaciones móviles y sistemas externos.	

*Tabla XXXIII: RNF-10 - Arquitectura de API REST (Fuente: Elaboración propia)*

### 2.6.3. Casos de uso (diagramas UML)

Los casos de uso describen la serie de acciones que deben llevar a cabo los usuarios, también conocidos como actores, para completar una tarea específica dentro del sistema.

#### Actores

Los actores en los casos de uso son entidades externas al sistema (personas, otros sistemas o dispositivos) que interactúan con él para lograr un objetivo específico. Representan los roles que usan o se comunican con el sistema.

Actor	Administrador de sistema
Descripción	Usuario con rol de administrador, responsable de administrar y supervisar el funcionamiento general de la aplicación.

*Tabla XXXIV: Actor - Administrador de sistema (Fuente: Elaboración propia)*

Actor	Administrador principal
Descripción	Usuario con rol de administrador, responsable de administrar y supervisar el funcionamiento general de la aplicación.

*Tabla XXXV: Actor - Administrador principal (Fuente: Elaboración propia)*

Actor	Docente
Descripción	Usuario con rol de administrador, responsable de administrar y supervisar el funcionamiento general de la aplicación.

*Tabla XXXVI: Actor - Docente (Fuente: Elaboración propia)*

Actor	Estudiante
Descripción	Usuario con rol de administrador, responsable de administrar y supervisar el funcionamiento general de la aplicación.

*Tabla XXXVII: Actor - Estudiante (Fuente: Elaboración propia)*

## Diagramas de casos de uso

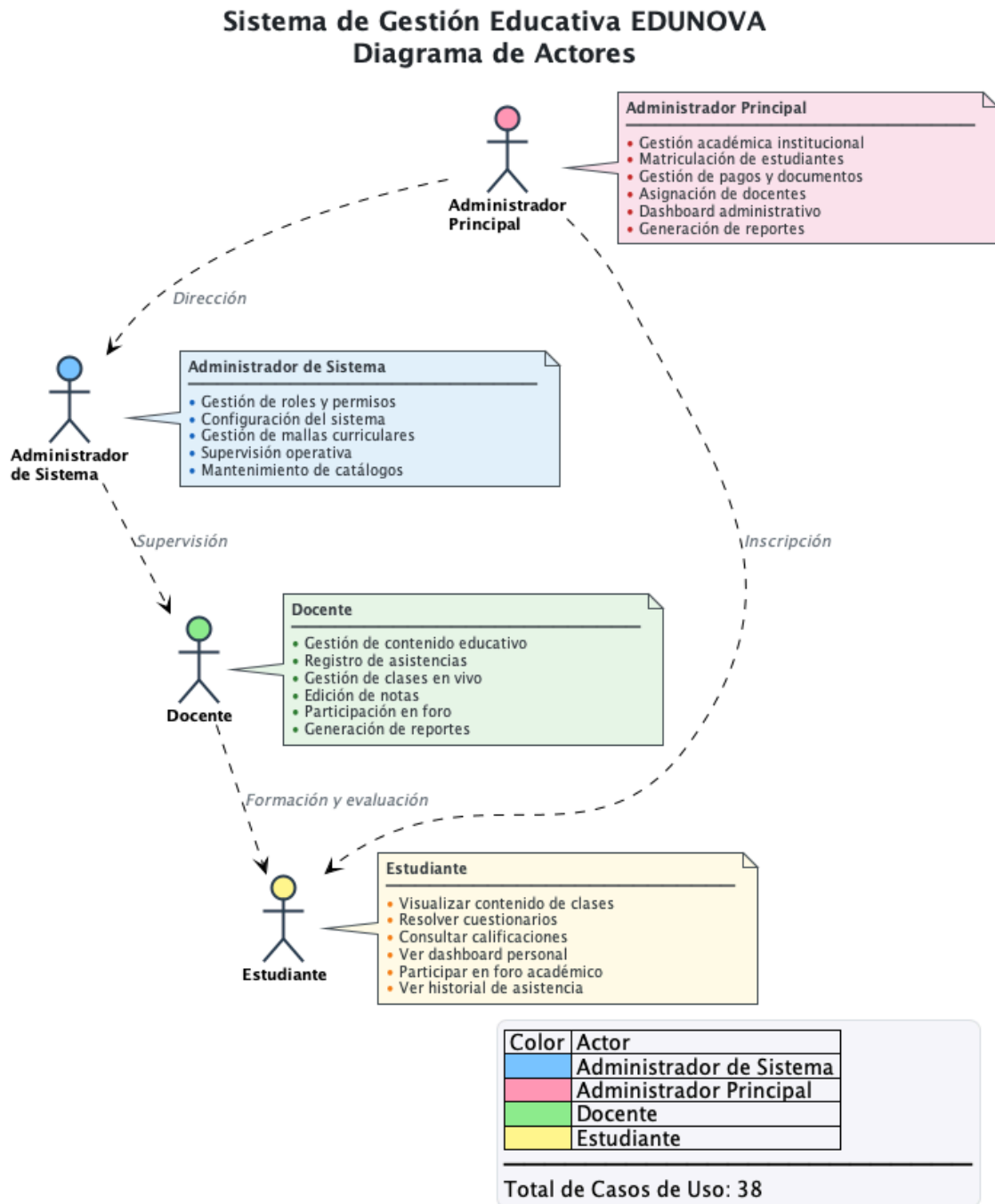
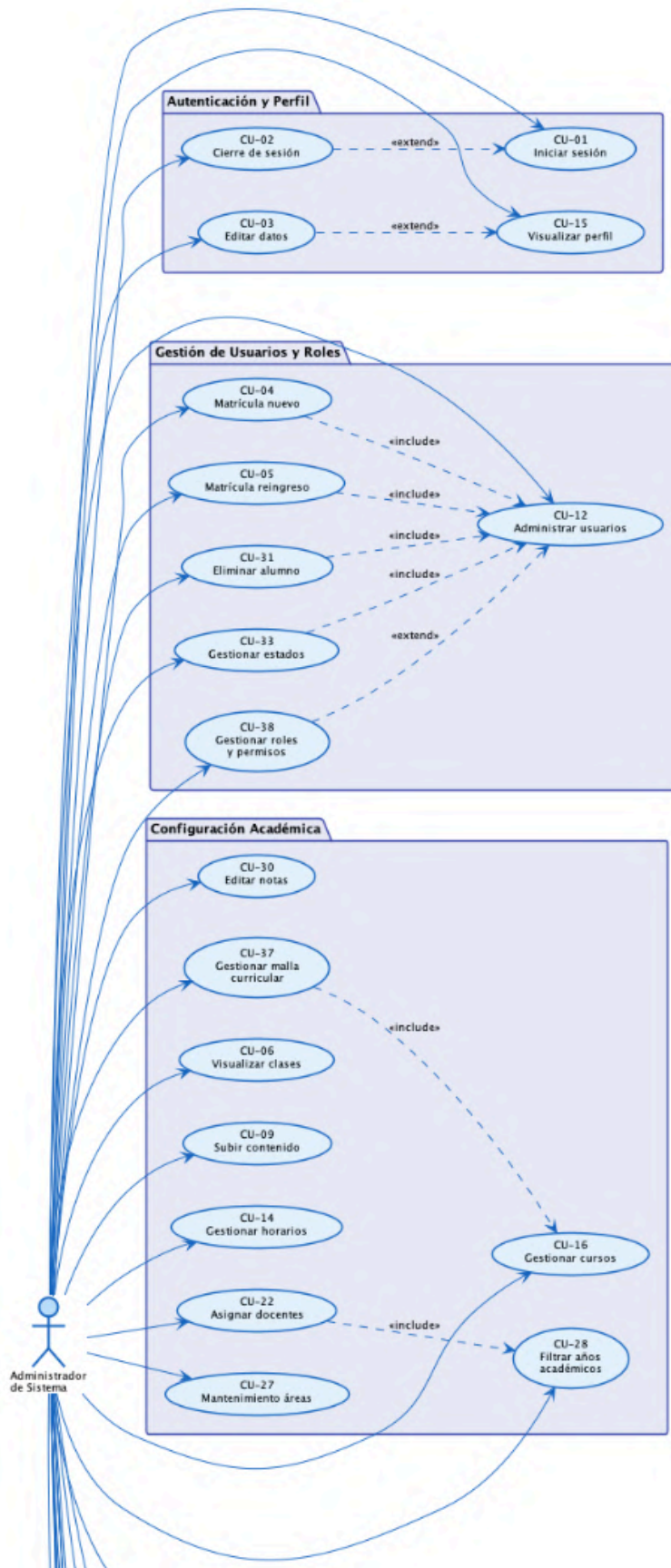


Ilustración I: Diagrama de actores (Fuente: Elaboración propia)

## Descripción de los casos de uso



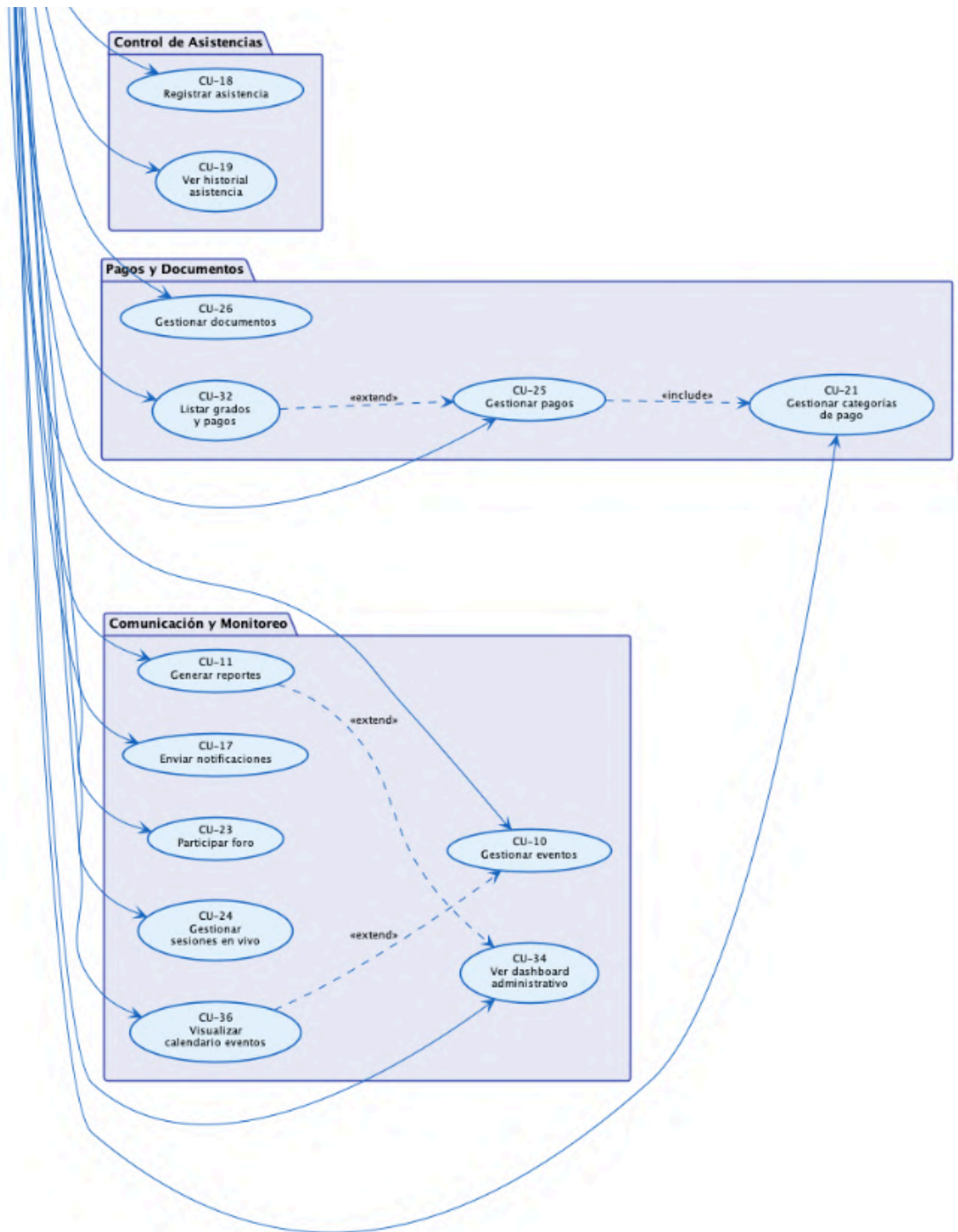
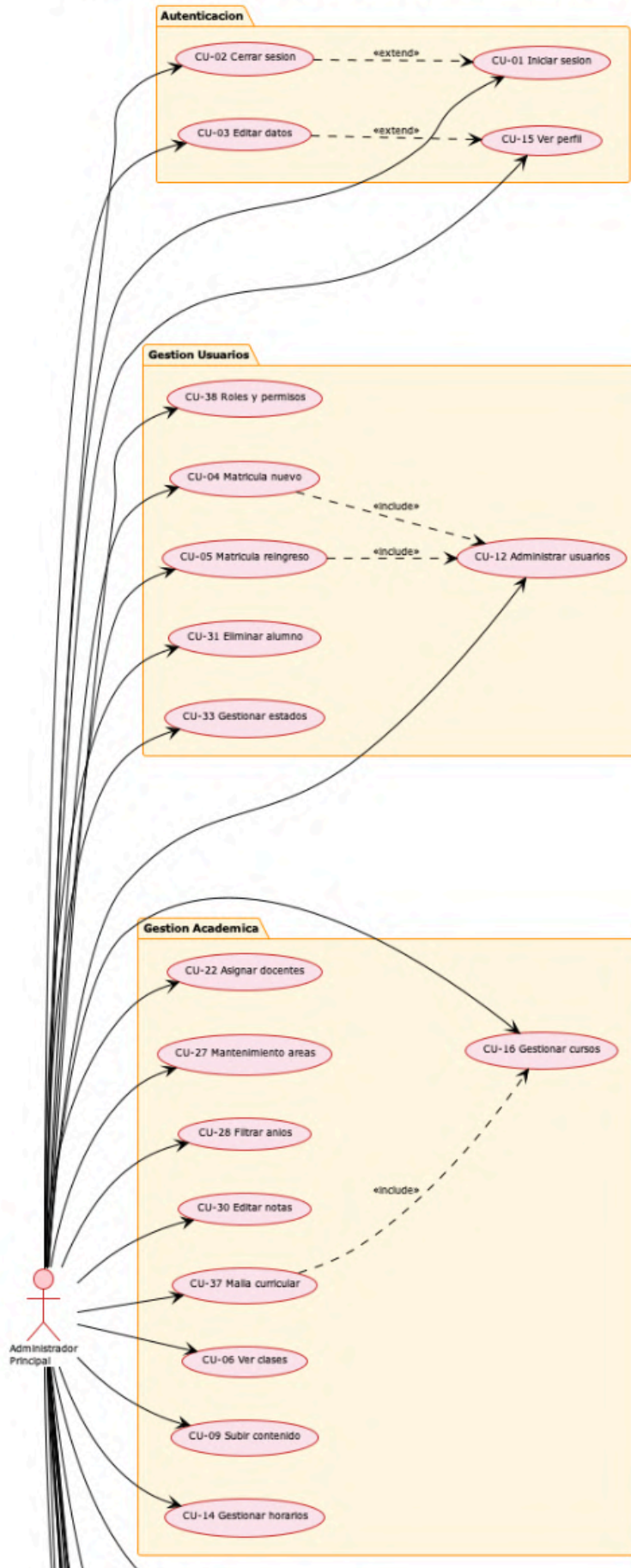
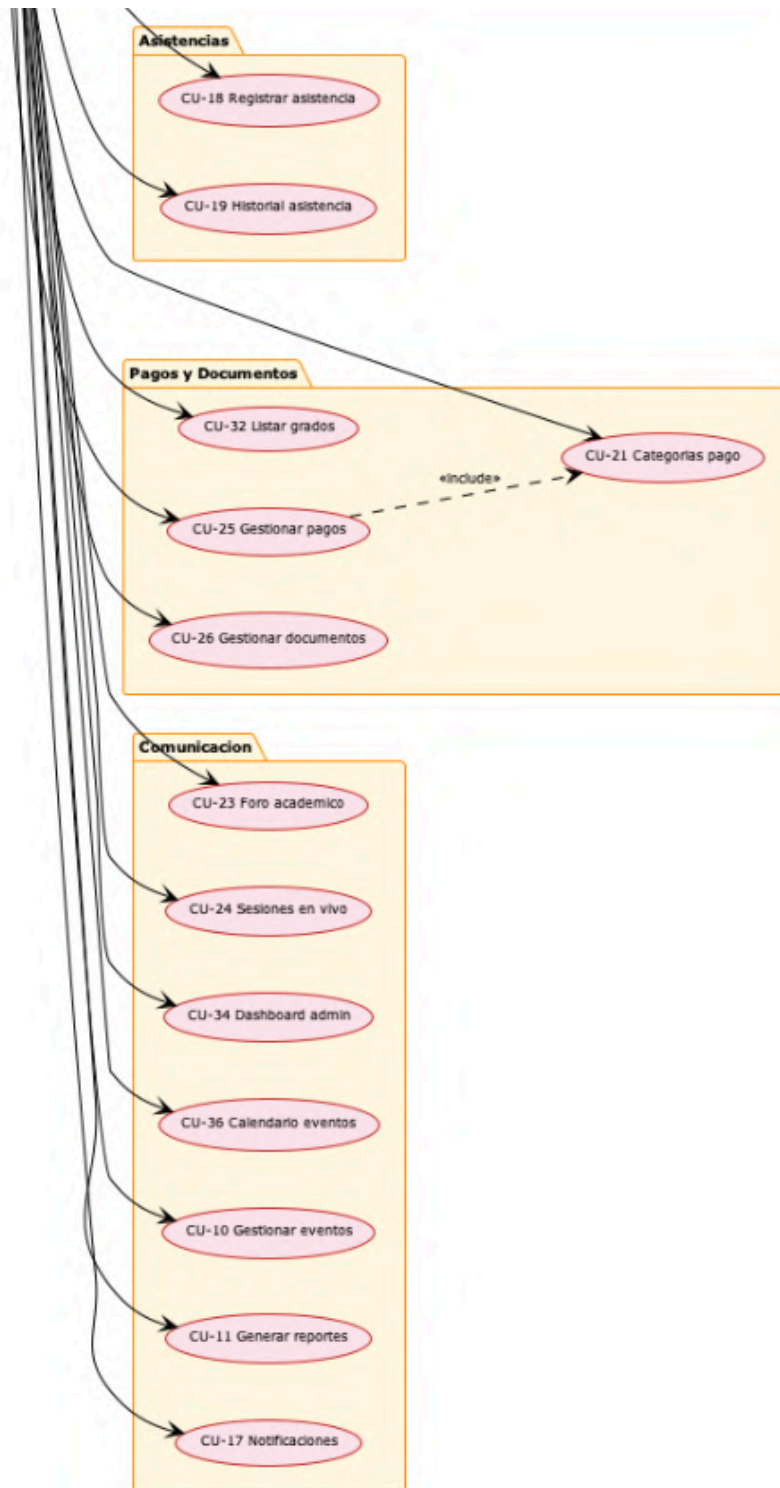


Ilustración II: Casos de Uso del Administrador del sistema (Fuente: Elaboración propia)





*Ilustración III: Casos de Uso del Administrador principal (Fuente: Elaboración propia)*

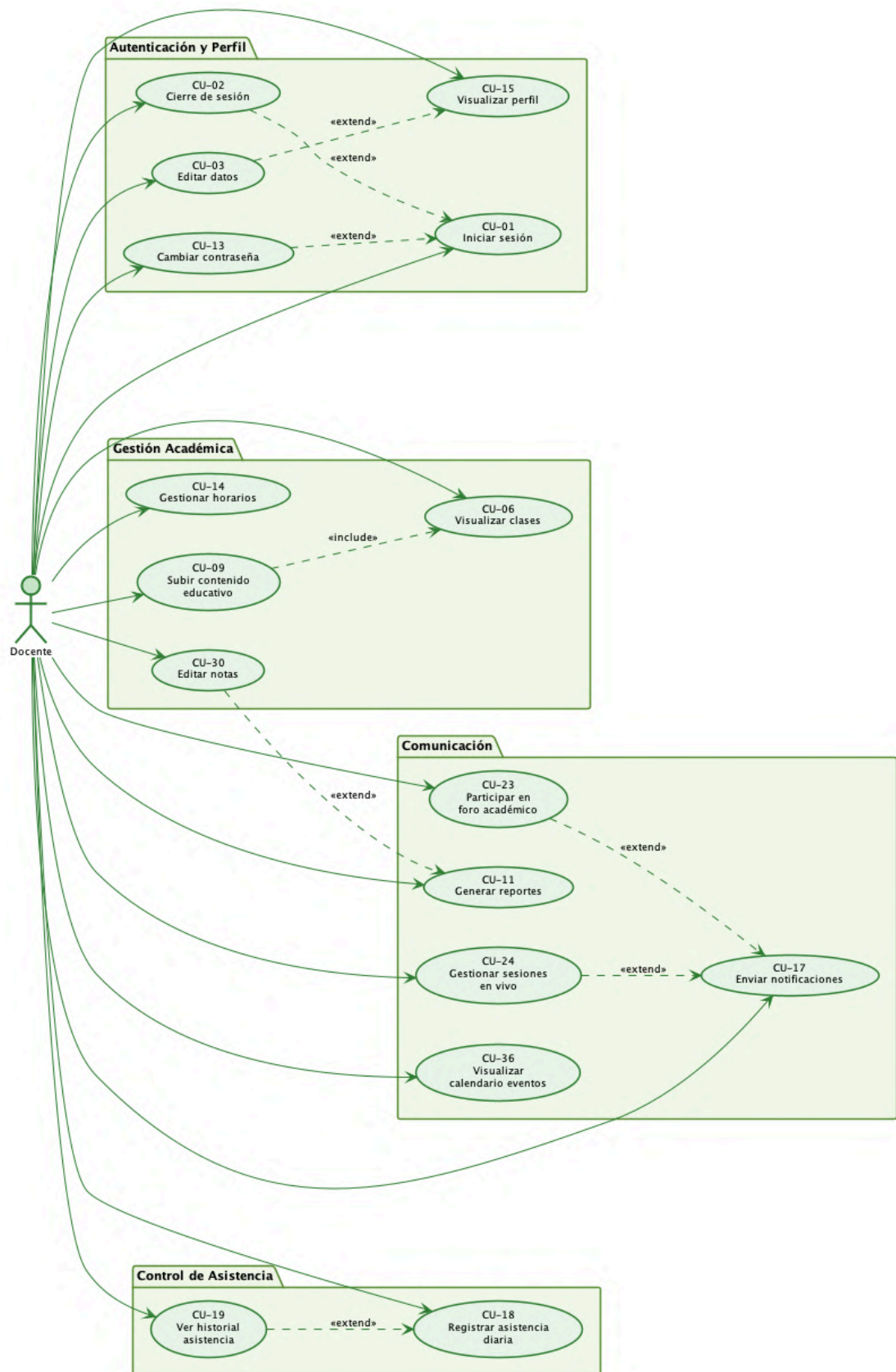


Ilustración IV: Casos de Uso del Docente (Fuente: Elaboración propia)

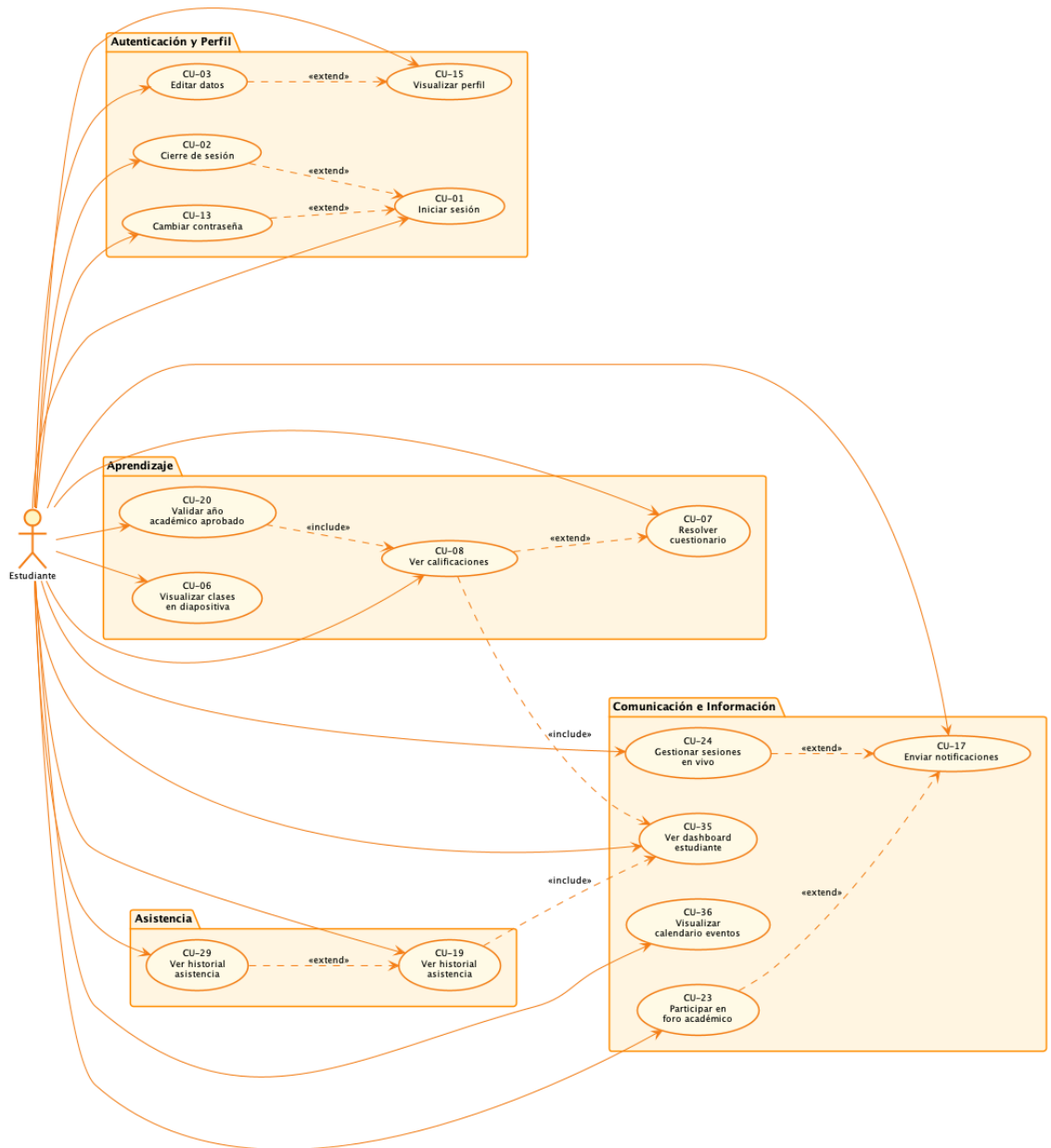


Ilustración V: Casos de Uso del Estudiante (Fuente: Elaboración propia)

#### 2.6.4. Descripción de los casos de uso

CU-01	Iniciar sesión
Descripción	Los usuarios se autentican en el sistema mediante un formulario con campos de correo electrónico y contraseña.
Actores	Estudiante, Docente, Administrador
Dependencias	Ninguna
Precondición	El usuario debe estar registrado en el sistema y no haber iniciado sesión.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Accede a la pantalla de inicio de sesión.</li> <li>2. Introduce correo y contraseña.</li> <li>3. Hace clic en “Iniciar sesión”.</li> </ol>
Postcondición	El usuario queda autenticado en el sistema.
Excepciones	<ul style="list-style-type: none"> <li>▪ Usuario o contraseña incorrecta. (Muestra mensaje de alerta)</li> </ul>
Importancia	Alta

Tabla XXXVIII: CU-01 - Iniciar sesión (Fuente: Elaboración propia)

CU-02	Cierre de sesión
Descripción	Permite al usuario autenticado cerrar su sesión en el sistema.
Actores	Estudiante, Docente, Administrador
Dependencias	Ninguna
Precondición	El usuario debe estar autenticado.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Hace clic en el botón de cerrar sesión.</li> <li>2. El sistema lo redirige como usuario anónimo.</li> </ol>
Postcondición	El usuario queda desautenticado.
Excepciones	<ul style="list-style-type: none"> <li>▪ Error de red al cerrar sesión. (Muestra mensaje de error)</li> </ul>
Importancia	Alta

Tabla XXXIX: CU-02 - Cierre de sesión (Fuente: Elaboración propia)

CU-03	Editar datos de usuario
Descripción	Permite al usuario modificar su información personal, incluyendo su nombre, correo y avatar.
Actores	Estudiante, Docente, Administrador
Dependencias	Ninguna
Precondición	El usuario debe estar autenticado.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Entra a ajustes de cuenta.</li> <li>2. Modifica los campos deseados.</li> <li>3. Guarda los cambios.</li> </ol>
Postcondición	La información del usuario queda actualizada en el sistema.
Excepciones	<ul style="list-style-type: none"> <li>▪ Datos inválidos o faltantes. (Muestra mensaje de error)</li> </ul>
Importancia	Media

Tabla XL: CU-03 - Editar datos de usuario (Fuente: Elaboración propia)

CU-04	Matrícula de nuevo estudiante en año académico
Descripción	Permite al administrador registrar un estudiante nuevo en un año académico disponible.
Actores	Administrador
Dependencias	Ninguna
Precondición	Debe existir al menos un año académico en estado activo o preinscripción.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Accede a la sección de año académico.</li> <li>2. Selecciona un año académico y entrar a módulo gestionar alumnos.</li> <li>3. Ingresar datos necesarios para matricular al estudiante en la sección matrícula.</li> </ol>
Postcondición	El estudiante queda matriculado en el año académico.
Excepciones	<ul style="list-style-type: none"> <li>▪ El año académico ya no está disponible.</li> <li>▪ Datos incompletos.</li> </ul>
Importancia	Alta

Tabla XLI: CU-04 - Matrícula de nuevo estudiante en año académico (Fuente: Elaboración propia)

CU-05	Matrícula de estudiante de reingreso en año académico
Descripción	Permite al administrador registrar un estudiante de reingreso en un año académico disponible.
Actores	Administrador
Dependencias	Ninguna
Precondición	<ul style="list-style-type: none"> <li>▪ Debe existir al menos un año académico en estado activo o preinscripción.</li> <li>▪ Debe existir el estudiante previamente.</li> </ul>
Secuencia normal	<ol style="list-style-type: none"> <li>1. Accede a la sección de año académico.</li> <li>2. Selecciona un año académico y entrar a módulo gestionar alumnos.</li> <li>3. Ingresar datos necesarios para matricular al estudiante en la sección reingreso.</li> </ol>
Postcondición	El usuario queda autenticado en el sistema.
Excepciones	<ul style="list-style-type: none"> <li>▪ El año académico ya no está disponible.</li> <li>▪ Datos incompletos.</li> </ul>
Importancia	Alta

*Tabla XLII: CU-05 - Matrícula de estudiante de reingreso en año académico (Fuente: Elaboración propia)*

CU-06	Visualizar clases en formato diapositiva
Descripción	Permite al estudiante visualizar las clases del año en formato de diapositiva multimedia.
Actores	Estudiante, Docente, Administrador
Dependencias	Debe existir contenido multimedia disponible para el curso.
Precondición	El estudiante debe estar matriculado en un año académico activo.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Accede a la plataforma.</li> <li>2. Selecciona un curso matriculado.</li> <li>3. Visualiza las diapositivas.</li> </ol>
Postcondición	El estudiante revisa el contenido de la clase.
Excepciones	<ul style="list-style-type: none"> <li>▪ El contenido no está disponible o no se carga.</li> <li>▪ Error en el reproductor.</li> </ul>
Importancia	Alta

*Tabla XLIII: CU-06 - Visualizar clases en formato diapositiva (Fuente: Elaboración propia)*

CU-07	Resolver cuestionario
Descripción	Permite al estudiante rendir un cuestionario con preguntas de evaluación.
Actores	Estudiante
Dependencias	El cuestionario debe estar habilitado por el docente.
Precondición	El estudiante debe estar matriculado y haber accedido a la clase.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Ingresa a la plataforma a la sección de cursos.</li> <li>2. Ingresa al cuestionario de una clase específica.</li> <li>3. Responde todas las preguntas.</li> <li>4. Envía el cuestionario.</li> </ol>
Postcondición	El sistema guarda y califica automáticamente el cuestionario.
Excepciones	▪ Cuestionario inactivo.
Importancia	Alta

*Tabla XLIV: CU-07 - Resolver cuestionario (Fuente: Elaboración propia)*

CU-08	Ver calificaciones
Descripción	Permite al estudiante revisar sus notas obtenidas en los cuestionarios.
Actores	Estudiante
Dependencias	Deben existir evaluaciones calificadas.
Precondición	El estudiante debe estar autenticado y haber resuelto algún cuestionario.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Ingresa al módulo de calificaciones.</li> <li>2. Visualiza las notas por curso.</li> </ol>
Postcondición	El estudiante conoce su rendimiento académico.
Excepciones	▪ No hay notas registradas aún.
Importancia	Alta

*Tabla XLV: CU-08 - Ver calificaciones (Fuente: Elaboración propia)*

CU-09	Subir contenido educativo
Descripción	Permite a los docentes cargar diapositivas multimedia al curso.
Actores	Docente, Administrador
Dependencias	Debe existir un curso o clase asignada.
Precondición	El usuario debe estar autenticado y tener permisos en el curso.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Accede a la malla curricular.</li> <li>2. Accede al curso a la edición de la malla curricular.</li> <li>3. Crea o edita contenido, creando sesiones o cuestionarios dentro de unidades, módulos o semanas.</li> </ol>
Postcondición	El contenido queda disponible para los estudiantes.
Excepciones	<ul style="list-style-type: none"> <li>▪ Formato de archivo no válido.</li> <li>▪ Error en la subida.</li> </ul>
Importancia	Alta

*Tabla XLVI: CU-09 - Subir contenido educativo (Fuente: Elaboración propia)*

CU-10	Gestionar eventos institucionales
Descripción	Permite crear, editar y eliminar eventos del calendario institucional.
Actores	Administrador
Dependencias	Ninguna
Precondición	Usuario autenticado con permisos de administración.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al módulo de eventos.</li> <li>2. Crear nuevo evento con título, fecha, ubicación.</li> <li>3. Definir audiencia (público, estudiantes, docentes).</li> <li>4. Publicar evento.</li> </ol>
Postcondición	El evento queda visible en el calendario.
Excepciones	<ul style="list-style-type: none"> <li>▪ Campos obligatorios vacíos.</li> <li>▪ Fecha inválida.</li> </ul>
Importancia	Media

*Tabla XLVII: CU-10 - Gestionar eventos institucionales (Fuente: Elaboración propia)*

CU-11	Generar reportes académicos
Descripción	Permite a los docentes y administradores generar reportes por estudiante, curso o ciclo.
Actores	Docente, Administrador
Dependencias	Debe existir información académica registrada.
Precondición	El usuario debe estar autenticado y tener permisos.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Ingresar al módulo de reportes.</li> <li>2. Seleccionar criterios.</li> <li>3. Generar reporte.</li> <li>4. Descargar o imprimir.</li> </ol>
Postcondición	Se genera un documento con datos académicos.
Excepciones	<ul style="list-style-type: none"> <li>▪ No hay datos disponibles.</li> <li>▪ Error al generar PDF/Excel.</li> </ul>
Importancia	Alta

*Tabla XLVIII: CU-11 - Generar reportes académicos (Fuente: Elaboración propia)*

CU-12	Administrar usuarios
Descripción	Permite al administrador registrar, editar o eliminar usuarios del sistema.
Actores	Administrador
Dependencias	Debe tener acceso al panel de administración.
Precondición	El administrador debe estar autenticado.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Ingresar a la administración.</li> <li>2. Seleccionar el módulo de usuarios.</li> <li>3. Registrar/editar/eliminar usuario.</li> </ol>
Postcondición	El sistema actualiza la información de los usuarios.
Excepciones	<ul style="list-style-type: none"> <li>▪ Datos incompletos.</li> <li>▪ Error de validación.</li> <li>▪ No se puede eliminar usuario con notas activas.</li> </ul>
Importancia	Alta

*Tabla XLIX: CU-12 - Administrar usuarios (Fuente: Elaboración propia)*

CU-13	Cambiar contraseña
Descripción	Permite al usuario modificar su contraseña desde su perfil.
Actores	Estudiante
Dependencias	Debe estar autenticado.
Precondición	Ingresar con contraseña actual.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Accede al perfil.</li> <li>2. Selecciona “Cambiar contraseña”.</li> <li>3. Ingresa contraseña actual y nueva.</li> <li>4. Guarda cambios.</li> </ol>
Postcondición	La contraseña es modificada.
Excepciones	<ul style="list-style-type: none"> <li>▪ Contraseña actual incorrecta.</li> <li>▪ No cumple con validaciones de seguridad.</li> </ul>
Importancia	Media

*Tabla L: CU-13 - Cambiar contraseña (Fuente: Elaboración propia)*

CU-14	Gestionar horarios de clases
Descripción	Permite crear y gestionar horarios de clases por grado, asignando docentes, asignaturas y aulas.
Actores	Administrador
Dependencias	Docentes asignados a cursos
Precondición	Usuario autenticado. Docentes y cursos configurados.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al módulo de horarios.</li> <li>2. Seleccionar grado y ciclo académico.</li> <li>3. Crear bloques horarios con asignatura, docente y aula.</li> <li>4. Guardar horario.</li> </ol>
Postcondición	El horario queda disponible para consulta.
Excepciones	<ul style="list-style-type: none"> <li>▪ Conflicto de horarios (docente o aula).</li> <li>▪ Horario superpuesto.</li> </ul>
Importancia	Alta

*Tabla LI: CU-14 - Gestionar horarios de clases (Fuente: Elaboración propia)*

CU-15	Visualizar perfil
Descripción	Permite al usuario ver sus datos personales y académicos.
Actores	Todos los usuarios
Dependencias	Debe estar autenticado.
Precondición	El usuario ha iniciado sesión.
Secuencia normal	1. Ingresa al menú de perfil. 2. Visualiza datos.
Postcondición	Puede ver sus datos sin modificarlos.
Excepciones	▪ Error al cargar datos.
Importancia	Media

*Tabla LII: CU-15 - Visualizar perfil (Fuente: Elaboración propia)*

CU-16	Gestionar cursos
Descripción	Permite al administrador crear, editar o eliminar cursos en el sistema.
Actores	Administrador
Dependencias	Debe existir al menos un año académico registrado.
Precondición	El administrador debe estar autenticado.
Secuencia normal	1. Acceder al módulo de asignaturas. 2. Crear/editar/eliminar curso. 3. Guardar cambios.
Postcondición	Los cursos quedan disponibles en el sistema.
Excepciones	▪ Curso asignado no puede eliminarse. ▪ Datos incompletos.
Importancia	Alta

*Tabla LIII: CU-16 - Gestionar cursos (Fuente: Elaboración propia)*

CU-17	Enviar notificaciones
Descripción	Permite enviar mensajes personalizados a los estudiantes filtrados por grado o nivel educativo.
Actores	Administrador, Docente, Estudiante
Dependencias	Grados y niveles registrados
Precondición	Usuario autenticado
Secuencia normal	1. Seleccionar filtro (grado/nivel). 2. Redactar mensaje. 3. Enviar notificación.
Postcondición	Estudiantes reciben la notificación
Excepciones	<ul style="list-style-type: none"> <li>▪ No hay estudiantes en el filtro</li> <li>▪ Error de envío</li> </ul>
Importancia	Alta

*Tabla LIV: CU-17 - Enviar notificaciones (Fuente: Elaboración propia)*

CU-18	Registrar asistencia diaria
Descripción	Permite al docente o administrador registrar la asistencia diaria de los estudiantes por grado.
Actores	Docente, Administrador
Dependencias	Estudiantes matriculados en el grado
Precondición	Usuario autenticado con permisos sobre el grado. Debe existir un año académico activo.
Secuencia normal	1. Acceder al módulo de asistencias. 2. Seleccionar grado y fecha. 3. Marcar estado de asistencia (presente, ausente, tardanza). 4. Guardar registro.
Postcondición	La asistencia queda registrada en el sistema.
Excepciones	<ul style="list-style-type: none"> <li>▪ Asistencia ya registrada para esa fecha.</li> <li>▪ Error al guardar datos.</li> </ul>
Importancia	Alta

*Tabla LV: CU-18 - Registrar asistencia diaria (Fuente: Elaboración propia)*

CU-19	Ver historial de asistencia del estudiante
Descripción	Permite visualizar el historial completo de asistencias de un estudiante por período.
Actores	Estudiante, Docente, Administrador
Dependencias	App instalada o acceso web responsive.
Precondición	Asistencias registradas
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al módulo de asistencias o perfil del estudiante.</li> <li>2. Seleccionar rango de fechas.</li> <li>3. Visualizar historial con estadísticas.</li> </ol>
Postcondición	Se muestra el historial de asistencias.
Excepciones	<ul style="list-style-type: none"> <li>▪ No hay registros en el período seleccionado.</li> </ul>
Importancia	Alta

Tabla LVI: CU-19 - Ver historial de asistencia del estudiante (Fuente: Elaboración propia)

CU-20	Validar año académico aprobado
Descripción	Permite al sistema verificar si un estudiante ha aprobado todos los cursos de un ciclo.
Actores	Sistema
Dependencias	Notas finales registradas.
Precondición	El estudiante debe haber completado los cursos.
Secuencia normal	<ol style="list-style-type: none"> <li>1. El sistema revisa las notas.</li> <li>2. Compara contra umbral de aprobación.</li> <li>3. Marca el ciclo como aprobado o no.</li> </ol>
Postcondición	El ciclo queda aprobado o pendiente.
Excepciones	<ul style="list-style-type: none"> <li>▪ Nota faltante en algún curso.</li> <li>▪ Error de lógica en validación.</li> </ul>
Importancia	Alta

Tabla LVII: CU-20 - Validar año académico aprobado (Fuente: Elaboración propia)

CU-21	Gestionar categorías y conceptos de pago
Descripción	Permite al administrador crear, editar o eliminar categorías y conceptos de pago (matrícula, pensión, recuperación, etc).
Actores	Administrador
Dependencias	Año académico creado
Precondición	Estar autenticado
Secuencia normal	1. Acceder a sección de pagos. 2. Crear categoría. 3. Asociar conceptos y montos.
Postcondición	Categorías y conceptos registrados en el sistema.
Excepciones	<ul style="list-style-type: none"> <li>▪ Conceptos duplicados</li> <li>▪ Campos incompletos</li> </ul>
Importancia	Alta

*Tabla LVIII: CU-21 - Gestionar categorías y conceptos de pago (Fuente: Elaboración propia)*

CU-22	Asignar docentes por año académico
Descripción	Permite al administrador asignar docentes a cursos específicos dentro de un año académico.
Actores	Administrador
Dependencias	Docentes y cursos creados, año académico activo
Precondición	Usuario autenticado
Secuencia normal	1. Seleccionar ciclo y curso. 2. Buscar docente. 3. Asignar docente.
Postcondición	Curso tiene un docente asignado.
Excepciones	<ul style="list-style-type: none"> <li>▪ Docente ya asignado a otro curso en mismo horario</li> <li>▪ Curso inexistente</li> </ul>
Importancia	Alta

*Tabla LIX: CU-22 - Asignar docentes por año académico (Fuente: Elaboración propia)*

CU-23	Participar en el foro académico
Descripción	Permite a los estudiantes y docentes realizar publicaciones, responder dudas y marcar respuestas correctas.
Actores	Estudiante, Docente, Administrador
Dependencias	Curso y ciclo activos
Precondición	Usuario autenticado y con matrícula activa
Secuencia normal	1. Crear post. 2. Visualizar respuestas. 3. Responder o marcar correcto.
Postcondición	La información se publica en el foro.
Excepciones	<ul style="list-style-type: none"> <li>▪ Post duplicado</li> <li>▪ Publicación sin aprobar (pendiente)</li> </ul>
Importancia	Alta

*Tabla LX: CU-23 - Participar en el foro académico (Fuente: Elaboración propia)*

CU-24	Gestionar sesiones en vivo
Descripción	Permite al docente crear enlaces a clases en vivo (Jitsi Meet) y al alumno acceder a ellas.
Actores	Estudiante, Docente, Administrador
Dependencias	Curso asignado y activo
Precondición	Usuario autenticado y curso asignado
Secuencia normal	1. Crear sesión con enlace. 2. Estudiante accede desde su interfaz.
Postcondición	Sesión registrada y accesible en la fecha/hora indicada.
Excepciones	<ul style="list-style-type: none"> <li>▪ Enlace inválido</li> <li>▪ Fecha pasada</li> </ul>
Importancia	Alta

*Tabla LXI: CU-24- Gestionar sesiones en vivo (Fuente: Elaboración propia)*

CU-25	Gestionar pagos del estudiante
Descripción	Permite registrar pagos del alumno, además de agregar pagos por otros conceptos.
Actores	Administrador
Dependencias	Alumnos matriculados
Precondición	Usuario autenticado
Secuencia normal	1. Buscar estudiante. 2. Registrar pago 3. Guardar.
Postcondición	Información se guarda en el perfil del estudiante.
Excepciones	<ul style="list-style-type: none"> <li>▪ Formato inválido</li> <li>▪ Archivo mayor al límite</li> </ul>
Importancia	Alta

*Tabla LXII: CU-25 - Gestionar pagos del estudiante (Fuente: Elaboración propia)*

CU-26	Gestionar documentos del estudiante
Descripción	Permite registrar y subir documentos del alumno como DNI, certificados, etc.
Actores	Administrador
Dependencias	Alumnos matriculados
Precondición	Usuario autenticado
Secuencia normal	1. Buscar estudiante. 2. Registrar documento. 3. Guardar.
Postcondición	Información se guarda en el perfil del estudiante.
Excepciones	<ul style="list-style-type: none"> <li>▪ Formato inválido</li> <li>▪ Archivo mayor al límite</li> </ul>
Importancia	Alta

*Tabla LXIII: CU-26 - Gestionar documentos del estudiante (Fuente: Elaboración propia)*

CU-27	Mantenimiento de áreas (asignaturas, documentos, niveles, grados)
Descripción	Permite realizar el CRUD de estructuras como asignaturas, niveles, grados y documentos requeridos.
Actores	Administrador
Dependencias	Ninguna
Precondición	Usuario autenticado
Secuencia normal	1. Acceder al módulo de mantenimiento. 2. Crear/editar/eliminar registros.
Postcondición	Información estructural actualizada.
Excepciones	<ul style="list-style-type: none"> <li>▪ Duplicados</li> <li>▪ Datos incompletos</li> </ul>
Importancia	Alta

*Tabla LXIV: CU-27 - Mantenimiento de áreas (asignaturas, documentos, niveles, grados)  
(Fuente: Elaboración propia)*

CU-28	Filtrar años académicos
Descripción	Permite realizar filtros por años académicos para control de alumnos y docentes.
Actores	Administrador
Dependencias	Ninguna
Precondición	Usuario autenticado
Secuencia normal	1. Acceder al módulo de año académico. 2. Seleccionar año académico.
Postcondición	Información estructural actualizada.
Excepciones	<ul style="list-style-type: none"> <li>▪ Error al cargar datos.</li> </ul>
Importancia	Media

*Tabla LXV: CU-28 - Filtrar ciclos (Fuente: Elaboración propia)*

CU-29	Ver historial de asistencia del estudiante
Descripción	Permite visualizar el historial completo de asistencias de un estudiante por período.
Actores	Estudiante, Docente, Administrador
Dependencias	Asistencias registradas
Precondición	Usuario autenticado.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al módulo de asistencias o perfil del estudiante.</li> <li>2. Seleccionar rango de fechas.</li> <li>3. Visualizar historial con estadísticas.</li> </ol>
Postcondición	Se muestra el historial de asistencias.
Excepciones	<ul style="list-style-type: none"> <li>▪ No hay registros en el período seleccionado.</li> </ul>
Importancia	Alta

*Tabla LXVI: CU-29 - Ver historial de asistencia del estudiante (Fuente: Elaboración propia)*

CU-30	Editar notas
Descripción	Permite el ingreso y edición de notas de los alumnos.
Actores	Administradores, Docentes
Dependencias	Existencia de notas del curso
Precondición	Usuario autenticado
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al módulo de año académico.</li> <li>2. Seleccionar gestión académica de algún año académico.</li> <li>3. Edición de notas del alumno.</li> </ol>
Postcondición	Información estructural actualizada.
Excepciones	<ul style="list-style-type: none"> <li>▪ Error al cargar datos.</li> <li>▪ Error de lógica en validación.</li> </ul>
Importancia	Alta

*Tabla LXVII: CU-30 - Editar notas (Fuente: Elaboración propia)*

CU-31	Eliminar alumno
Descripción	Permite al administrador eliminar alumnos del año académico.
Actores	Administrador
Dependencias	Debe tener acceso al panel de administración.
Precondición	El administrador debe estar autenticado.
Secuencia normal	1. Ingresar al módulo año escolar. 2. Seleccionar el listado de alumnos. 3. Eliminar alumno.
Postcondición	El sistema actualiza la información de los usuarios.
Excepciones	<ul style="list-style-type: none"> <li>▪ Datos incompletos.</li> <li>▪ Error de validación.</li> <li>▪ No se puede eliminar alumno con notas activas.</li> </ul>
Importancia	Alta

*Tabla LXVIII: CU-31 - Eliminar alumno (Fuente: Elaboración propia)*

CU-32	Listar grados y ver pagos de estudiantes
Descripción	Permite listar grados del año académico además verificar los pagos realizados por el estudiante.
Actores	Administrador
Dependencias	Alumnos matriculados
Precondición	Usuario autenticado
Secuencia normal	1. Ingresar al módulo año académico. 2. Acceder al listado de alumnos por grados 3. Ver pagos por estudiante.
Postcondición	La información se guarda en el perfil del estudiante.
Excepciones	<ul style="list-style-type: none"> <li>▪ Formato inválido</li> <li>▪ Archivo mayor al límite</li> </ul>
Importancia	Alta

*Tabla LXIX: CU-32 - Listar grados y ver pagos del estudiante (Fuente: Elaboración propia)*

CU-33	Gestionar estados de alumnos
Descripción	Permite al administrador controlar los estados de los alumnos como activos o no, sin necesidad de eliminar datos.
Actores	Administrador
Dependencias	Alumno activo
Precondición	Usuario autenticado
Secuencia normal	<ol style="list-style-type: none"> <li>1. Ingresar al módulo año académico.</li> <li>2. Acceder al listado de alumnos por grados</li> <li>3. Cambiar estado de estudiante.</li> </ol>
Postcondición	Sesión registrada y accesible en la fecha/hora indicada.
Excepciones	<ul style="list-style-type: none"> <li>▪ Datos incompletos.</li> <li>▪ Error de validación.</li> </ul>
Importancia	Alta

Tabla LXX: CU-33 - Gestionar estados de alumnos (Fuente: Elaboración propia)

CU-34	Ver dashboard administrativo
Descripción	Permite al administrador visualizar métricas generales: matrícula, asistencia, pagos y rendimiento académico.
Actores	Administrador
Dependencias	Datos registrados en el sistema
Precondición	Usuario autenticado con rol de administrador.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al dashboard.</li> <li>2. Visualizar KPIs principales.</li> <li>3. Filtrar por ciclo, nivel o grado.</li> <li>4. Revisar alertas y estudiantes en riesgo.</li> </ol>
Postcondición	El administrador obtiene visión general del sistema.
Excepciones	<ul style="list-style-type: none"> <li>▪ Error al cargar métricas.</li> </ul>
Importancia	Alta

Tabla LXXI: CU-34 - Ver dashboard administrativo (Fuente: Elaboración propia)

CU-35	Ver dashboard del estudiante
Descripción	Permite al estudiante visualizar su resumen académico, asistencia, pagos y progreso de aprendizaje.
Actores	Estudiante
Dependencias	Estudiante matriculado
Precondición	Usuario autenticado con rol de estudiante.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al dashboard.</li> <li>2. Visualizar promedio general.</li> <li>3. Ver estado de asistencia.</li> <li>4. Verificar pagos pendientes.</li> </ol>
Postcondición	El estudiante conoce su situación académica actual.
Excepciones	<ul style="list-style-type: none"> <li>▪ Error al cargar datos.</li> </ul>
Importancia	Alta

*Tabla LXXII: CU-35 - Ver dashboard del estudiante (Fuente: Elaboración propia)*

CU-36	Visualizar calendario de eventos
Descripción	Permite al estudiante visualizar su resumen académico, asistencia, pagos y progreso de aprendizaje.
Actores	Estudiante
Dependencias	Estudiante matriculado
Precondición	Usuario autenticado con rol de estudiante.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al dashboard o landing page</li> </ol>
Postcondición	El estudiante conoce su situación académica actual.
Excepciones	<ul style="list-style-type: none"> <li>▪ Error al cargar datos.</li> </ul>
Importancia	Alta

*Tabla LXXIII: CU-36 - Visualizar calendario de eventos (Fuente: Elaboración propia)*

CU-37	Gestionar malla curricular
Descripción	Permite crear y configurar mallas curriculares, asociando asignaturas a grados específicos.
Actores	Administrador
Dependencias	Asignaturas y grados creados
Precondición	Usuario autenticado con permisos de administración.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder al módulo de mallas curriculares.</li> <li>2. Crear nueva malla o editar existente.</li> <li>3. Asociar asignaturas por grado.</li> <li>4. Guardar configuración.</li> </ol>
Postcondición	La malla curricular queda configurada.
Excepciones	<ul style="list-style-type: none"> <li>▪ Asignatura ya asociada al grado.</li> <li>▪ Datos incompletos.</li> </ul>
Importancia	Alta

*Tabla LXXIV: CU-37 - Gestionar malla curricular (Fuente: Elaboración propia)*

CU-38	Gestionar roles y permisos
Descripción	Permite crear roles personalizados y asignar permisos específicos a cada rol.
Actores	Administrador
Dependencias	Permisos definidos en el sistema
Precondición	Usuario autenticado con rol de superadministrador.
Secuencia normal	<ol style="list-style-type: none"> <li>1. Acceder a gestión de roles.</li> <li>2. Crear/editar rol.</li> <li>3. Asignar permisos al rol.</li> <li>4. Guardar cambios.</li> </ol>
Postcondición	El rol queda configurado con los permisos seleccionados.
Excepciones	<ul style="list-style-type: none"> <li>▪ Rol en uso no puede eliminarse.</li> <li>▪ Nombre de rol duplicado.</li> </ul>
Importancia	Alta

*Tabla LXXV: CU-38 - Gestionar roles y permisos (Fuente: Elaboración propia)*

## 2.7. Actividades y recursos

### 2.7.1. Cronograma

Nº	Objetivo	Mes 1				Mes 2				Mes 3				Mes 4				Mes 5				Mes 6			
		S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4	S1	S2	S3	S4
<b>1 Diseñar e implementar un módulo de registro digital de estudiantes, docentes y cursos</b>																									
1.1	Levantamiento de requerimientos con directivos y docentes.	X	X	X																					
1.2	Análisis y diseño de la base de datos para el registro académico.			X	X	X																			
1.3	Programación del módulo de registro digital.					X	X	X	X	X	X	X	X												
1.4	Pruebas piloto con datos reales de matrícula.													X	X	X	X								
<b>2 Desarrollar un sistema centralizado para el control de calificaciones y asistencia.</b>																									
2.1	Análisis de los procesos actuales de calificación y asistencia.	X	X	X																					
2.2	Desarrollo de interfaces para la gestión de notas y asistencia.			X	X	X	X	X	X																
2.3	Implementación de reportes automáticos y filtros de búsqueda.							X	X	X	X	X													
2.4	Capacitación inicial a los docentes.													X	X										
<b>3 Incorporar un módulo de seguimiento académico.</b>																									
3.1	Definición de métricas de rendimiento académico.	X	X	X	X																				
3.2	Desarrollo del módulo de seguimiento con reportes gráficos.				X	X	X	X	X	X	X														
3.3	Implementación de alertas tempranas en la plataforma.									X	X	X													
3.4	Validación con docentes sobre retroalimentación generada.													X	X	X	X								
<b>4 Integrar un chat con inteligencia artificial como tutor virtual.</b>																									
4.1	Integración de API de inteligencia artificial (ej. OpenAI o similar).									X	X	X													
4.2	Configuración de respuestas predefinidas para asistencia en la plataforma.											X													
4.3	Pruebas de interacción con estudiantes.													X	X	X	X								
4.4	Ajustes según retroalimentación recibida.															X	X								
<b>5 Implementar un foro académico en línea.</b>																									
5.1	Diseño del módulo de foros (categorías y permisos).						X	X																	
5.2	Desarrollo de funcionalidades de interacción (comentarios, respuestas, notificaciones).								X	X															
5.3	Integración del foro con la base de datos centralizada.										X	X													
5.4	Capacitación de estudiantes y docentes en su uso.													X	X	X									
<b>6 Evaluar el impacto de la plataforma en los procesos académicos.</b>																									
6.1	Aplicación de encuestas a estudiantes y docentes.																	X	X						
6.2	Análisis estadístico de los resultados obtenidos.																			X	X				
6.3	Elaboración de informe de impacto.																						X		
6.4	Presentación de resultados a la institución.																							X	

Tabla LXXVI: Cronograma de desarrollo (Fuente: Elaboración propia)

## 2.7.2. Presupuesto

### Presupuesto del proyecto

DESCRIPCIÓN	CANTIDAD	PRECIO UNITARIO	PRECIO TOTAL
MacBook Pro M1 Pro 2021	01	S/ 7,000.00	S/ 7,000.00
Acer Nitro Gamer i5 10ma	01	S/ 3,000.00	S/ 3,000.00
PC Ryzen 5 3600, 16GB, GTX 1050	01	S/ 3,200.00	S/ 3,200.00
Hosting	06	S/ 61	S/ 366
Servicio internet (x2)	06	S/ 220.00	S/ 1,320.00
Servicio de luz (x2)	06	S/ 180.00	S/ 1,080.00
Pasajes traslados al colegio	20	S/15	S/ 300
<b>TOTAL</b>			<b>S/ 16,466.00</b>

*Tabla LXXVII: Presupuesto del Proyecto (Fuente: Elaboración propia)*

### **3. DESARROLLO DE LA SOLUCIÓN**

El presente capítulo describe de manera detallada la arquitectura de software y el diseño técnico del sistema, una plataforma web educativa orientada a la gestión académica de instituciones educativas. Se detallan los componentes principales del sistema, las tecnologías utilizadas, la organización interna del frontend y backend, la persistencia de datos, los mecanismos de seguridad y las decisiones técnicas adoptadas durante el desarrollo.

#### **3.1. Arquitectura del sistema**

El sistema ha sido diseñado bajo una arquitectura cliente-servidor basada en servicios, lo que permite una clara separación de responsabilidades, alta mantenibilidad y escalabilidad. La solución está compuesta por tres capas principales:

- Capa de Presentación (Frontend Web)
- Capa de Lógica de Negocio (Backend API REST)
- Capa de Persistencia de Datos (Base de Datos Relacional)

Estas capas se comunican entre sí mediante protocolos HTTP/HTTPS, utilizando el estándar REST para el intercambio de información en formato JSON. La arquitectura adoptada facilita la evolución del sistema, permitiendo la incorporación de nuevos módulos, servicios o clientes (por ejemplo, aplicaciones móviles) sin afectar el núcleo del sistema.

#### **3.2. Diagrama de arquitectura del sistema**

La arquitectura general del sistema se representa mediante un diagrama en el que se identifican los actores principales (administrador, estudiante y usuario público), el frontend web desarrollado en React, el backend basado en Node.js y Express, y la base de datos PostgreSQL. El flujo de información se inicia cuando el usuario interactúa con la interfaz web; estas solicitudes son procesadas por la API REST del backend, la cual ejecuta la lógica de negocio correspondiente y accede a la base de datos para consultar o persistir información. Finalmente, el backend retorna la respuesta al frontend para su visualización.

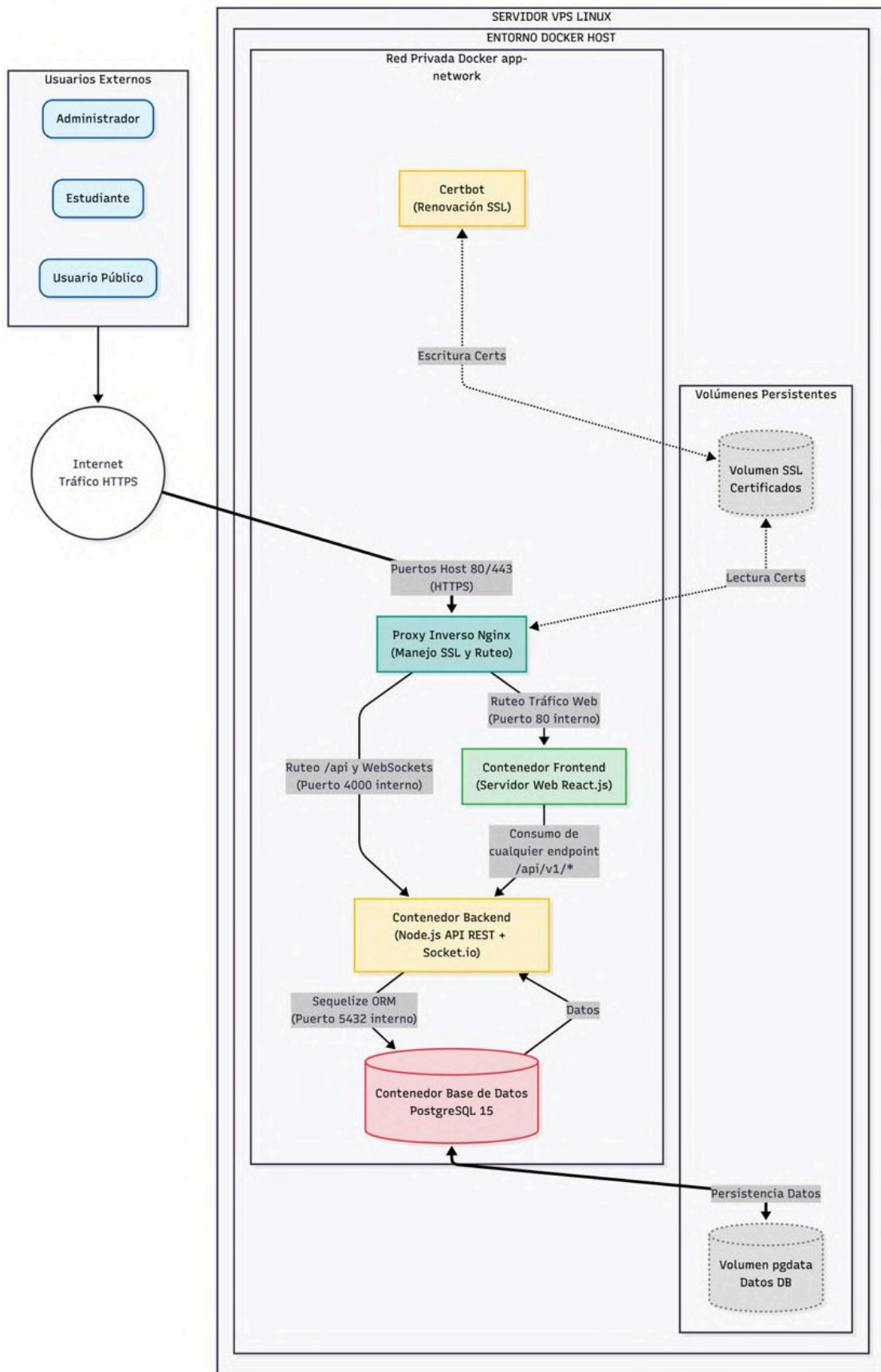


Ilustración VI: Diagrama de arquitectura de software (Fuente: Elaboración propia)

### **3.3. Arquitectura frontend**

El frontend fue desarrollado utilizando React.js, una biblioteca JavaScript para la construcción de interfaces de usuario basadas en componentes reutilizables.

#### **3.3.1. Características del frontend**

- Arquitectura basada en componentes funcionales
- Navegación tipo SPA (Single Page Application)
- Manejo de estado mediante Context API y hooks personalizados
- Comunicación con el backend mediante API REST
- Diseño responsivo para múltiples dispositivos

#### **3.3.2. Organización del frontend**

El proyecto frontend se organiza en módulos claramente definidos, tales como componentes, páginas, layouts, servicios de API, contextos y utilidades, lo que permite una alta cohesión y bajo acoplamiento entre funcionalidades.

Además, se implementaron layouts diferenciados para los distintos roles del sistema: administrador, estudiante y sitio público.

```

1  edunova-frontend/
2  └─ public/           # Archivos públicos
3     └─ index.html    # HTML principal
4     └─ manifest.json # Configuración PWA
5     └─ robots.txt    # Configuración SEO
6     └─ img/          # Imágenes públicas
7         └─ container/
8         └─ landing/
9         └─ logo/
10 └─ src/             # Código fuente
11     └─ api/         # Servicios de API
12         └─ auth.js  # Autenticación
13         └─ course.js # Gestión de cursos
14         └─ student.js # Funciones de estudiantes
15         └─ aiAssistant.js # Asistente IA
16         └─ ...      # Otros servicios
17     └─ assets/     # Recursos estáticos
18         └─ jpg/
19         └─ png/
20         └─ svg/
21     └─ components/ # Componentes reutilizables
22         └─ Admin/  # Componentes de administración
23             └─ AcademicCycle/
24             └─ Course/
25             └─ Users/
26             └─ ... # 24+ componentes
27         └─ Student/ # Componentes de estudiantes
28             └─ Dashboard/
29             └─ MyCourses/
30             └─ MyGrades/
31             └─ ... # 18+ componentes
32     └─ Web/         # Componentes del sitio público
33         └─ Banner/
34         └─ Courses/
35         └─ Reviews/
36         └─ ...      # 18+ componentes
37     └─ Shared/     # Componentes compartidos
38         └─ BasicModal/
39         └─ CustomTable/
40         └─ ...
41     └─ ui/         # Componentes UI genéricos
42     └─ contexts/   # Context API de React
43         └─ AuthContext.js
44         └─ MediaContext.js
45         └─ PermissionsContext.js
46     └─ hooks/      # Hooks personalizados
47         └─ useAuth.js
48         └─ usePermissions.js
49     └─ layouts/    # Layouts de página
50         └─ AdminLayout/
51         └─ StudentLayout/
52         └─ ClientLayout/
53     └─ pages/      # Páginas principales
54         └─ admin/
55         └─ student/
56         └─ web/
57     └─ router/     # Configuración de rutas
58         └─ AdminRouter.js
59         └─ StudentRouter.js
60         └─ WebRouter.js
61         └─ RouterManager.js
62     └─ scss/       # Estilos globales
63         └─ colors.scss
64         └─ variables.scss
65         └─ modals.scss
66         └─ ...
67     └─ utils/      # Utilidades y constantes
68         └─ constants.js
69         └─ socket.js
70         └─ token.js
71         └─ ...
72     └─ App.js      # Componente raíz
73     └─ index.js    # Punto de entrada
74     └─ index.scss  # Estilos principales
75     └─ package.json # Dependencias del proyecto
76     └─ tailwind.config.js # Configuración de Tailwind CSS
77     └─ postcss.config.js # Configuración de PostCSS
78     └─ docker-compose.yml # Configuración Docker Compose
79     └─ Dockerfile   # Dockerfile de producción
80     └─ nginx.conf   # Configuración de Nginx
81     └─ README.md    # Documentación del proyecto

```

Ilustración VII: Estructura de directorios del frontend (Fuente: Elaboración propia)

### **3.4. Arquitectura del backend**

El backend del sistema fue desarrollado utilizando Node.js junto con el framework Express.js, siguiendo una arquitectura RESTful.

#### **3.4.1. Responsabilidades del backend**

- Gestión de la lógica de negocio
- Autenticación y autorización de usuarios
- Exposición de endpoints REST versionados
- Validación de datos
- Comunicación en tiempo real mediante WebSockets

#### **3.4.2. Organización del backend**

El backend se encuentra organizado en capas que incluyen controladores, servicios, modelos, rutas y middlewares, lo que facilita la escalabilidad y el mantenimiento del sistema.

Se implementó un sistema de versión de API (v1), permitiendo futuras actualizaciones sin afectar a los clientes existentes.

```

1  edunova-backend/
2  └─ app.js           # Configuración principal de Express
3  └─ index.js        # Punto de entrada del servidor
4  └─ socket.js       # Configuración de Socket.io
5  └─ constants.js    # Constantes de la aplicación
6  └─ migrateData.js  # Script de migración de datos
7  └─ package.json    # Dependencias y scripts
8  └─ Dockerfile      # Configuración de imagen Docker
9  └─ docker-compose.local.yml # Docker Compose para desarrollo híbrido
10 └─ README.md       # Documentación del proyecto
11 └─ config/         # Configuración de base de datos
12     └─ config.js   # Configuración de entornos
13     └─ database.js # Configuración de Sequelize
14 └─ controllers/   # Controladores de rutas
15     └─ auth.js     # Autenticación y autorización
16     └─ user.js     # Gestión de usuarios
17     └─ student.js  # Gestión de estudiantes
18     └─ course.js   # Gestión de cursos
19     └─ subject.js  # Gestión de materias
20     └─ grade.js    # Gestión de grados
21     └─ syllabu.js  # Gestión de sílabos
22     └─ payment*.js # Gestión de pagos
23     └─ post.js     # Sistema de posts/foros
24     └─ notification.js # Sistema de notificaciones
25     └─ room.js     # Salas virtuales
26     └─ ...
27 └─ models/        # Modelos de Sequelize
28 └─ migrations/    # Migraciones de base de datos
29 └─ seeders/       # Datos iniciales
30 └─ router/        # Definición de rutas
31 └─ services/      # Lógica de negocio
32 └─ middlewares/   # Middlewares personalizados
33     └─ authenticated.js # Middleware de autenticación
34     └─ authorize.js  # Middleware de autorización
35 └─ utils/         # Utilidades
36     └─ jwt.js       # Manejo de tokens JWT
37     └─ image.js     # Procesamiento de imágenes
38     └─ ubigeo-reniec.js # Datos geográficos
39 └─ uploads/       # Archivos subidos

```

*Ilustración VIII: Estructura de directorios del backend (Fuente: Elaboración propia)*

### **3.5. Base de datos y persistencia**

Para la persistencia de la información se utilizó PostgreSQL, una base de datos relacional de alto rendimiento y confiabilidad.

#### **3.5.1. Modelo de datos**

La base de datos almacena información relacionada con:

- Usuarios y roles
- Estudiantes
- Cursos, materias y sílabos
- Ciclos académicos y sesiones
- Pagos y conceptos de pago
- Foros, publicaciones y notificaciones
- Documentos institucionales

Se aplicaron principios de normalización para garantizar la integridad y consistencia de los datos.

#### **3.5.2. Acceso a datos**

El acceso a la base de datos se realiza mediante el ORM Sequelize, lo que permite abstraer las consultas SQL y facilita la portabilidad y mantenimiento del sistema.

### 3.5.3. Base de datos relacional

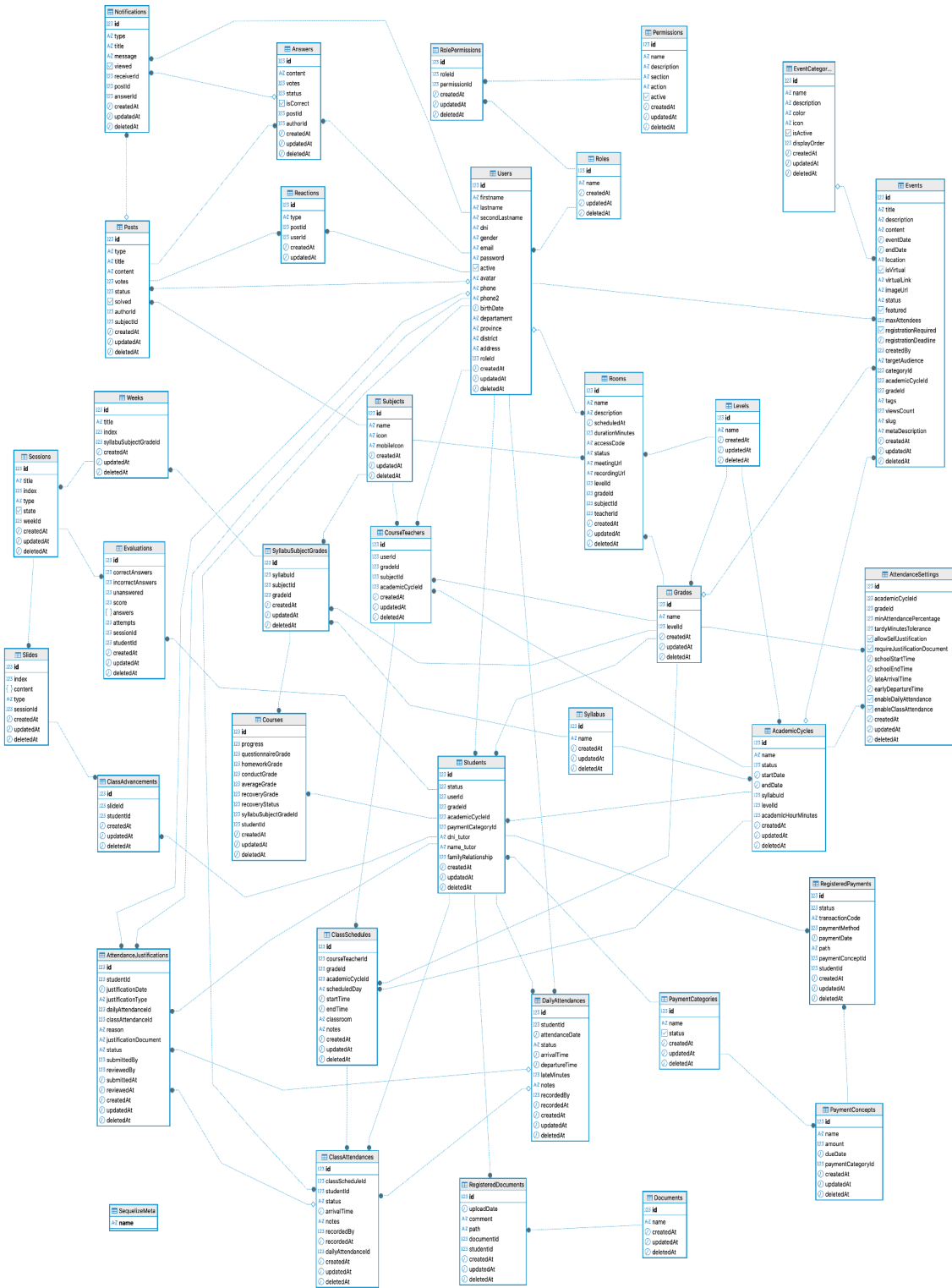


Ilustración IX: Modelo Entidad Relación (Fuente: Elaboración propia)

### 3.6. Comunicación en tiempo real

El sistema incorpora funcionalidades de comunicación en tiempo real mediante Socket.io, permitiendo:

- Envío de notificaciones instantáneas
- Actualización en tiempo real de foros académicos

Este enfoque mejora la experiencia del usuario y fomenta la interacción dentro de la plataforma.

### 3.7. Seguridad y control de accesos

El sistema implementa diversos mecanismos de seguridad, entre los cuales destacan:

- Autenticación basada en JWT (JSON Web Tokens)
- Encriptación de contraseñas mediante bcryptjs
- Middlewares de autenticación y autorización
- Control de acceso por roles
- Configuración de políticas CORS

Estos mecanismos garantizan que solo los usuarios autorizados puedan acceder a los recursos del sistema.

### 3.8. Tecnologías y librerías utilizadas

A continuación, se detallan las principales tecnologías y herramientas empleadas en el desarrollo del sistema:

<b>Categoría</b>	<b>Tecnología / Herramienta</b>	<b>Descripción</b>
Frontend	React.js	Construcción de interfaces web
Frontend	React Router	Navegación SPA
Frontend	Tailwind CSS	Estilización rápida y responsiva

Frontend	Semantic UI React	Componentes UI estilizados
Formularios	React Hook Form	Gestión eficiente de formularios
Validación	Yup	Validación de datos
Multimedia	React Quill	Editor de texto enriquecido
Multimedia	React Player	Reproducción multimedia
Visualización	Recharts	Gráficos interactivos
Backend	Node.js	Entorno de ejecución
Backend	Express.js	Framework API REST
Base de Datos	PostgreSQL	Persistencia de datos
ORM	Sequelize	Mapeo objeto-relacional
Seguridad	JSON Web Token (JWT)	Autenticación
Seguridad	bcryptjs	Encriptación de contraseñas
Seguridad	Helmet	Protección de la aplicación mediante configuración de seguridad HTTP.
Seguridad	CORS	Control de acceso seguro entre frontend y backend.
Configuración	Dotenv	Gestión de variables de entorno y configuración sensible.
Tiempo real	Socket.io	Comunicación en tiempo real
Contenedores	Docker	Entornos aislados
Control de versiones	Git	Gestión de código fuente

*Tabla LXXVIII: Tabla de tecnologías usadas en el proyecto (Fuente: Elaboración propia)*

### **3.9. Justificación de la arquitectura adoptada**

La arquitectura seleccionada responde a criterios de escalabilidad, seguridad, mantenibilidad y costo. El uso de tecnologías open source permite reducir costos de licenciamiento, mientras que la separación en capas facilita el mantenimiento y la incorporación de nuevas funcionalidades.

Asimismo, la adopción de una API REST desacoplada permite que el sistema pueda ser extendido fácilmente hacia aplicaciones móviles u otros clientes en el futuro.

### **3.10. Despliegue y entorno de ejecución**

El sistema puede ser desplegado utilizando contenedores Docker, lo que garantiza consistencia entre los entornos de desarrollo y producción. Se definieron configuraciones específicas para desarrollo, producción y entornos híbridos, optimizando el flujo de trabajo del proyecto.

Este enfoque permite una mayor portabilidad, facilidad de despliegue y escalabilidad del sistema.

## 3.11. Código fuente

### 3.11.1. Funcionalidad 1: Registro de estudiantes en año académico

```
1 static async createStudent(studentData, transaction) {
2   const dataUser = {
3     firstname: studentData.firstname,
4     lastname: studentData.lastname,
5     secondLastname: studentData.secondLastname,
6     dni: studentData.dni,
7     gender: studentData.gender,
8     email: studentData.email,
9     phone: studentData.phone,
10    phone2: studentData.phone2,
11    birthDate: studentData.birthDate,
12    departament: studentData.departament,
13    province: studentData.province,
14    district: studentData.district,
15    address: studentData.address,
16    roleId: 3,
17    active: true,
18    avatar: studentData.avatar,
19    password: bcrypt.hashSync("edunova2024", bcrypt.genSaltSync(10)),
20  };
21  const storedUser = await User.create(dataUser, { transaction });
22  const storedStudent = await Student.create(
23    {
24      status: 1,
25      userId: storedUser.id,
26      gradeId: studentData.gradeId,
27      academicCycleId: studentData.academicCycleId,
28      paymentCategoryId: studentData.paymentCategoryId,
29      dni_tutor: studentData.dni_tutor,
30      name_tutor: studentData.name_tutor,
31      familyRelationship: studentData.familyRelationship,
32    },
33    { transaction }
34  );
35  const paymentConcepts = await PaymentConcept.findAll({
36    where: {
37      paymentCategoryId: studentData.paymentCategoryId,
38      deletedAt: null,
39    },
40    attributes: { exclude: ["createdAt", "updatedAt", "deletedAt"] },
41    order: [["createdAt", "ASC"]],
42  });
43  const registeredPayments = paymentConcepts.map((concept) => ({
44    paymentConceptId: concept.id,
45    studentId: storedStudent.id,
46  }));
47  await RegisteredPayment.bulkCreate(registeredPayments, {
48    transaction,
49  });
50  const syllabuSubjectGrades = await SyllabuSubjectGrade.findAll({
51    where: {
52      deletedAt: null,
53      syllabuId: studentData.syllabuId,
54      gradeId: studentData.gradeId,
55    },
56    attributes: { exclude: ["createdAt", "updatedAt", "deletedAt"] },
57    order: [["createdAt", "ASC"]],
58  });
59  const courses = syllabuSubjectGrades.map((syllabuSubjectGrade) => ({
60    syllabuSubjectGradeId: syllabuSubjectGrade.id,
61    studentId: storedStudent.id,
62  }));
63  await Course.bulkCreate(courses, { transaction });
64  return storedStudent;
65 }
```

Ilustración X: Código fuente de registro de estudiante (Fuente: Elaboración propia)

```

1  const { DataTypes, Model } = require("sequelize");
2
3  class User extends Model {
4    /**
5     * Initializes the User model.
6     *
7     * @param {Object} sequelize - The Sequelize instance.
8     * @returns {Object} - The User model.
9     */
10   static init(sequelize) {
11     return super.init(
12       {
13         id: {
14           type: DataTypes.INTEGER,
15           primaryKey: true,
16           autoIncrement: true,
17         },
18         firstname: {
19           type: DataTypes.STRING(100),
20           allowNull: true,
21         },
22         lastname: {
23           type: DataTypes.STRING(100),
24           allowNull: true,
25         },
26         secondLastname: {
27           type: DataTypes.STRING(100),
28           allowNull: true,
29         },
30         dni: {
31           type: DataTypes.STRING(15),
32           allowNull: true,
33         },
34         gender: {
35           type: DataTypes.STRING(20),
36           allowNull: true,
37         },
38         email: {
39           type: DataTypes.STRING,
40           unique: true,
41           allowNull: false,
42         },
43         password: {
44           type: DataTypes.STRING,
45           allowNull: false,
46         },
47         active: {
48           type: DataTypes.BOOLEAN,
49           allowNull: false,
50           defaultValue: false,
51         },
52         avatar: {
53           type: DataTypes.STRING,
54           allowNull: true,
55         },
56         phone: {
57           type: DataTypes.STRING(15),
58           allowNull: true,
59         },
60         phone2: {
61           type: DataTypes.STRING(15),
62           allowNull: true,
63         },
64         birthDate: {
65           type: DataTypes.DATE,
66           allowNull: true,
67         },
68         department: {
69           type: DataTypes.STRING(100),
70           allowNull: true,
71         },
72         province: {
73           type: DataTypes.STRING(100),
74           allowNull: true,
75         },
76         district: {
77           type: DataTypes.STRING(100),
78           allowNull: true,
79         },
80         address: {
81           type: DataTypes.STRING,
82           allowNull: true,
83         },
84         roleId: {
85           type: DataTypes.INTEGER,
86           allowNull: false,
87           references: {
88             model: "Roles",
89             key: "id",
90           },
91         },
92       },
93       {
94         sequelize,
95         modelName: "User",
96         schema: "edunova_db",
97         timestamps: true,
98         paranoid: true,
99       }
100    );
101  }

```

*Ilustración XI: Código fuente de modelo User - campos (Fuente: Elaboración propia)*

```

102
103 /**
104  * Initializes the User associations.
105  *
106  * @param {Object} models - The database models.
107  * @returns {void}
108  */
109 static associate(models) {
110   this.belongsTo(models.Role, { foreignKey: "roleId" });
111   this.hasMany(models.Student, {
112     foreignKey: "userId",
113   });
114   this.hasMany(models.Post, {
115     foreignKey: "authorId",
116   });
117
118   // Attendance-related associations
119   this.hasMany(models.DailyAttendance, {
120     foreignKey: "recordedBy",
121   });
122   this.hasMany(models.ClassAttendance, {
123     foreignKey: "recordedBy",
124   });
125   this.hasMany(models.AttendanceJustification, {
126     foreignKey: "submittedBy",
127   });
128   this.hasMany(models.AttendanceJustification, {
129     foreignKey: "reviewedBy",
130   });
131 }
132 }
133
134 module.exports = User;

```

*Ilustración XII: Código fuente de modelo User - relaciones (Fuente: Elaboración propia)*

```

1 const { DataTypes, Model } = require("sequelize");
2
3 class Student extends Model {
4   /**
5    * Initializes the Student model.
6    *
7    * @param {Object} sequelize - The Sequelize instance.
8    * @returns {Object} - The Student model.
9    */
10  static init(sequelize) {
11    return super.init(
12      {
13        id: {
14          type: DataTypes.INTEGER,
15          primaryKey: true,
16          autoIncrement: true,
17        },
18        status: {
19          type: DataTypes.INTEGER,
20          allowNull: false,
21          defaultValue: 1,
22        },
23        userId: {
24          type: DataTypes.INTEGER,
25          allowNull: false,
26          references: {
27            model: "Users",
28            key: "id",
29          },
30        },
31        gradeId: {
32          type: DataTypes.INTEGER,
33          allowNull: false,
34          references: {
35            model: "Grades",
36            key: "id",
37          },
38        },
39        academicCycleId: {
40          type: DataTypes.INTEGER,
41          allowNull: false,
42          references: {
43            model: "AcademicCycles",
44            key: "id",
45          },
46        },
47        paymentCategoryId: {
48          type: DataTypes.INTEGER,
49          allowNull: false,
50          references: {
51            model: "PaymentCategories",
52            key: "id",
53          },
54        },
55        dni_tutor: {
56          type: DataTypes.STRING(100),
57          allowNull: false,
58        },
59        name_tutor: {
60          type: DataTypes.STRING(100),
61          allowNull: false,
62        },
63        familyRelationship: {
64          type: DataTypes.INTEGER,
65          allowNull: false,
66        },
67      },
68      {
69        sequelize,
70        modelName: "Student",
71        schema: "esunova_db",
72        timestamps: true,
73        paranoid: true,
74      }
75    );
76  }

```

*Ilustración XIII: Código fuente de modelo Student - campos (Fuente: Elaboración propia)*

```

77
78 /**
79  * Initializes the Student associations.
80  *
81  * @param {Object} models - The database models.
82  * @returns {void}
83  */
84 static associate(models) {
85   this.belongsTo(models.User, {
86     foreignKey: "userId",
87     as: "User"
88   });
89   this.belongsTo(models.Grade, {
90     foreignKey: "gradeId",
91     as: "Grade"
92   });
93   this.belongsTo(models.AcademicCycle, {
94     foreignKey: "academicCycleId",
95   });
96   this.belongsTo(models.PaymentCategory, {
97     foreignKey: "paymentCategoryId",
98   });
99   this.hasMany(models.RegisteredPayment, {
100    foreignKey: "studentId",
101  });
102  this.hasMany(models.RegisteredDocument, {
103    foreignKey: "studentId",
104  });
105  this.hasMany(models.Course, {
106    foreignKey: "studentId",
107  });
108
109  // Attendance-related associations
110  this.hasMany(models.DailyAttendance, {
111    foreignKey: "studentId",
112    as: "dailyAttendances"
113  });
114  this.hasMany(models.ClassAttendance, {
115    foreignKey: "studentId",
116  });
117  this.hasMany(models.AttendanceJustification, {
118    foreignKey: "studentId",
119  });
120 }
121 }
122
123 module.exports = Student;

```

*Ilustración XIV: Código fuente de modelo Student - relaciones (Fuente: Elaboración propia)*

```

1  const { DataTypes, Model } = require("sequelize");
2
3  class PaymentConcept extends Model {
4    /**
5     * Initializes the PaymentConcept model.
6     *
7     * @param {Object} sequelize - The Sequelize instance.
8     * @returns {Object} - The PaymentConcept model.
9     */
10   static init(sequelize) {
11     return super.init(
12       {
13         id: {
14           type: DataTypes.INTEGER,
15           primaryKey: true,
16           autoIncrement: true,
17         },
18         name: {
19           type: DataTypes.STRING(100),
20           allowNull: false,
21         },
22         amount: {
23           type: DataTypes.DOUBLE,
24           allowNull: false,
25           defaultValue: 0.0,
26         },
27         dueDate: {
28           type: DataTypes.DATE,
29           allowNull: true,
30         },
31         paymentCategoryId: {
32           type: DataTypes.INTEGER,
33           allowNull: false,
34           references: {
35             model: "PaymentCategories",
36             key: "id",
37           },
38         },
39       },
40       {
41         sequelize,
42         modelName: "PaymentConcept",
43         schema: "edunova_db",
44         timestamps: true,
45         paranoid: true,
46       }
47     );
48   }

```

*Ilustración XV: Código fuente de modelo PaymentConcept - campos (Fuente: Elaboración propia)*

```

49
50 /**
51  * Initializes the PaymentConcept associations.
52  *
53  * @param {Object} models - The database models.
54  * @returns {void}
55  */
56 static associate(models) {
57   this.belongsTo(models.PaymentCategory, {
58     foreignKey: "paymentCategoryId",
59   });
60   this.hasMany(models.RegisteredPayment, {
61     foreignKey: "paymentConceptId",
62   });
63 }
64 }
65
66 module.exports = PaymentConcept;

```

Ilustración XVI: Código fuente de modelo PaymentConcept - relaciones (Fuente: Elaboración propia)

```

1  const { DataTypes, Model } = require("sequelize");
2
3  class RegisteredPayment extends Model {
4    /**
5     * Initializes the RegisteredPayment model.
6     *
7     * @param {Object} sequelize - The Sequelize instance.
8     * @returns {Object} - The RegisteredPayment model.
9     */
10   static init(sequelize) {
11     return super.init(
12       {
13         id: {
14           type: DataTypes.INTEGER,
15           primaryKey: true,
16           autoIncrement: true,
17         },
18         status: {
19           type: DataTypes.INTEGER,
20           allowNull: false,
21           defaultValue: 1,
22         },
23         transactionCode: {
24           type: DataTypes.STRING(100),
25           allowNull: true,
26         },
27         paymentMethod: {
28           type: DataTypes.INTEGER,
29           allowNull: true,
30         },
31         paymentDate: {
32           type: DataTypes.DATE,
33           allowNull: true,
34         },
35         path: {
36           type: DataTypes.STRING,
37           allowNull: true,
38         },
39         paymentConceptId: {
40           type: DataTypes.INTEGER,
41           allowNull: false,
42           references: {
43             model: "PaymentConcepts",
44             key: "id",
45           },
46         },
47         studentId: {
48           type: DataTypes.INTEGER,
49           allowNull: false,
50           references: {
51             model: "Students",
52             key: "id",
53           },
54         },
55       },
56       {
57         sequelize,
58         modelName: "RegisteredPayment",
59         schema: "edunova_db",
60         timestamps: true,
61         paranoid: true,
62       }
63     );
64   }

```

Ilustración XVII: Código fuente de modelo RegisteredPayment - campos (Fuente: Elaboración propia)

```

65
66  /**
67   * Initializes the RegisteredPayment associations.
68   *
69   * @param {Object} models - The database models.
70   * @returns {void}
71   */
72  static associate(models) {
73    this.belongsTo(models.PaymentConcept, {
74      foreignKey: "paymentConceptId",
75      as: "PaymentConcept"
76    });
77    this.belongsTo(models.Student, {
78      foreignKey: "studentId",
79      as: "Student"
80    });
81  }
82 }
83
84 module.exports = RegisteredPayment;
85

```

*Ilustración XVIII: Código fuente de modelo RegisteredPayment - relaciones (Fuente: Elaboración propia)*

```

1  const { DataTypes, Model } = require("sequelize");
2
3  class SyllabuSubjectGrade extends Model {
4    /**
5     * Initializes the SyllabuSubjectGrade model.
6     *
7     * @param {Object} sequelize - The Sequelize instance.
8     * @returns {Object} - The SyllabuSubjectGrade model.
9     */
10   static init(sequelize) {
11     return super.init(
12       {
13         id: {
14           type: DataTypes.INTEGER,
15           primaryKey: true,
16           autoIncrement: true,
17         },
18         syllabuId: {
19           type: DataTypes.INTEGER,
20           allowNull: false,
21           references: {
22             model: "Syllabus", // Nombre de la tabla de syllabus
23             key: "id",
24           },
25         },
26         subjectId: {
27           type: DataTypes.INTEGER,
28           allowNull: false,
29           references: {
30             model: "Subjects", // Nombre de la tabla de subjects
31             key: "id",
32           },
33         },
34         gradeId: {
35           type: DataTypes.INTEGER,
36           allowNull: false,
37           references: {
38             model: "Grades", // Nombre de la tabla de grades
39             key: "id",
40           },
41         },
42       },
43       {
44         sequelize,
45         modelName: "SyllabuSubjectGrade",
46         schema: "edunova_db",
47         timestamps: true,
48         paranoid: true,
49       }
50     );
51   }

```

*Ilustración XIX: Código fuente de modelo SyllabuSubjectGrade - campos (Fuente: Elaboración propia)*

```

52
53 /**
54  * Initializes the SyllabuSubjectGrade associations.
55  *
56  * @param {Object} models - The database models.
57  * @returns {void}
58  */
59 static associate(models) {
60   this.belongsTo(models.Syllabu, {
61     foreignKey: "syllabuId",
62   });
63   this.belongsTo(models.Subject, {
64     foreignKey: "subjectId",
65     as: "subject"
66   });
67   this.belongsTo(models.Grade, {
68     foreignKey: "gradeId",
69   });
70   this.hasMany(models.Week, {
71     foreignKey: "syllabuSubjectGradeId",
72   });
73   this.hasMany(models.Course, {
74     foreignKey: "syllabuSubjectGradeId",
75   });
76 }
77 }
78
79 module.exports = SyllabuSubjectGrade;
80

```

*Ilustración XX: Código fuente de modelo SyllabuSubjectGrade - relaciones (Fuente: Elaboración propia)*

```

1  const { DataTypes, Model } = require("sequelize");
2
3  class Course extends Model {
4    /**
5     * Initializes the Course model.
6     *
7     * @param {Object} sequelize - The Sequelize instance.
8     * @returns {Object} - The Course model.
9     */
10   static init(sequelize) {
11     return super.init(
12       {
13         id: {
14           type: DataTypes.INTEGER,
15           primaryKey: true,
16           autoIncrement: true,
17         },
18         progress: {
19           type: DataTypes.DOUBLE,
20           allowNull: true,
21           defaultValue: 0.0,
22         },
23         questionnaireGrade: {
24           type: DataTypes.DOUBLE,
25           allowNull: true,
26           defaultValue: 0.0,
27         },
28         homeworkGrade: {
29           type: DataTypes.DOUBLE,
30           allowNull: true,
31           defaultValue: 0.0,
32         },
33         conductGrade: {
34           type: DataTypes.DOUBLE,
35           allowNull: true,
36           defaultValue: 0.0,
37         },
38         averageGrade: {
39           type: DataTypes.DOUBLE,
40           allowNull: true,
41           defaultValue: 0.0,
42         },
43         recoveryGrade: {
44           type: DataTypes.DOUBLE,
45           allowNull: true,
46           defaultValue: 0.0,
47         },
48         recoveryStatus: {
49           type: DataTypes.INTEGER,
50           allowNull: true,
51           defaultValue: 1,
52         },
53         syllabuSubjectGradeId: {
54           type: DataTypes.INTEGER,
55           allowNull: false,
56           references: {
57             model: "SyllabuSubjectGrades",
58             key: "id",
59           },
60         },
61         studentId: {
62           type: DataTypes.INTEGER,
63           allowNull: false,
64           references: {
65             model: "Students",
66             key: "id",
67           },
68         },
69       },
70       {
71         sequelize, "Course",
72         schema: "eduneve_db",
73         timestamps: true,
74         paranoid: true,
75       }
76     );
77   }
78 }

```

*Ilustración XXI: Código fuente de modelo Course - campos (Fuente: Elaboración propia)*

```

79
80 /**
81  * Initializes the Course associations.
82  *
83  * @param {Object} models - The database models.
84  * @returns {void}
85  */
86 static associate(models) {
87   this.belongsTo(models.SyllabusSubjectGrade, {
88     foreignKey: "syllabusSubjectGradeId",
89     as: "SyllabusSubjectGrade"
90   });
91   this.belongsTo(models.Student, {
92     foreignKey: "studentId",
93     as: "Student"
94   });
95 }
96 }
97
98 module.exports = Course;
99

```

Ilustración XXII: Código fuente de modelo Course - relaciones (Fuente: Elaboración propia)

```

1 static async createStudent(req, res) {
2   const studentData = req.body;
3   // Verificar que los campos obligatorios estén presentes
4   if (
5     !studentData.firstname ||
6     !studentData.lastname ||
7     !studentData.dni ||
8     !studentData.email
9   ) {
10    return res.status(400).json({ message: "Faltan campos obligatorios" });
11  }
12  if (req.files?.avatar) {
13    studentData.avatar = image.getFilePath(req.files.avatar);
14  }
15  const t = await sequelize.transaction();
16  try {
17    const storedStudent = await StudentService.createStudent(studentData, t);
18    await t.commit();
19    res.status(201).json(storedStudent);
20  } catch (error) {
21    await t.rollback();
22    BaseController.handleError(res, "Error al crear usuario.", error);
23  }
24 }

```

Ilustración XXIII: Código fuente del método del controlador Student (Fuente: Elaboración propia)

### 3.11.2. Funcionalidad 2: Registro de asistencia diaria de estudiantes

```
1 static async recordDailyAttendanceByGrade(  
2     attendanceDate,  
3     attendances,  
4     recordedBy  
5 ) {  
6     const transaction = await sequelize.transaction();  
7  
8     try {  
9         const results = [];  
10  
11        for (const attendance of attendances) {  
12            // Buscar si ya existe un registro para este estudiante en esta fecha  
13            const existingRecord = await DailyAttendance.findOne({  
14                where: {  
15                    studentId: attendance.studentId,  
16                    attendanceDate,  
17                    deletedAt: null,  
18                },  
19                transaction,  
20            });  
21  
22            let attendanceRecord;  
23  
24            if (existingRecord) {  
25                // Si existe, actualizar el registro  
26                await existingRecord.update(  
27                    {  
28                        status: attendance.status,  
29                        arrivalTime: attendance.arrivalTime,  
30                        departureTime: attendance.departureTime,  
31                        lateMinutes: attendance.lateMinutes || 0,  
32                        notes: attendance.notes,  
33                        recordedBy,  
34                    },  
35                    { transaction }  
36                );  
37  
38                attendanceRecord = existingRecord;  
39            } else {  
40                // Si no existe, crear un nuevo registro  
41                attendanceRecord = await DailyAttendance.create(  
42                    {  
43                        studentId: attendance.studentId,  
44                        attendanceDate,  
45                        status: attendance.status,  
46                        arrivalTime: attendance.arrivalTime,  
47                        departureTime: attendance.departureTime,  
48                        lateMinutes: attendance.lateMinutes || 0,  
49                        notes: attendance.notes,  
50                        recordedBy,  
51                    },  
52                    { transaction }  
53                );  
54            }  
55  
56            results.push(attendanceRecord);  
57        }  
58  
59        await transaction.commit();  
60        return results;  
61    } catch (error) {  
62        await transaction.rollback();  
63        throw error;  
64    }  
65 }
```

Ilustración XXIV: Código fuente del registro de asistencia diaria (Fuente: Elaboración propia)

```

1  const { DataTypes, Model } = require("sequelize");
2
3  class DailyAttendance extends Model {
4    /**
5     * Initializes the DailyAttendance model.
6     *
7     * @param {Object} sequelize - The Sequelize instance.
8     * @returns {Object} - The DailyAttendance model.
9     */
10   static init(sequelize) {
11     return super.init(
12       {
13         id: {
14           type: DataTypes.INTEGER,
15           primaryKey: true,
16           autoIncrement: true,
17         },
18         studentId: {
19           type: DataTypes.INTEGER,
20           allowNull: false,
21           references: {
22             model: "Students",
23             key: "id",
24           },
25         },
26         attendanceDate: {
27           type: DataTypes.DATEONLY,
28           allowNull: false,
29         },
30         status: {
31           type: DataTypes.STRING(20),
32           allowNull: false,
33           comment: "presente, ausente, tardanza, justificado, salida_temprana",
34         },
35         arrivalTime: {
36           type: DataTypes.TIME,
37           allowNull: true,
38         },
39         departureTime: {
40           type: DataTypes.TIME,
41           allowNull: true,
42         },
43         lateMinutes: {
44           type: DataTypes.INTEGER,
45           allowNull: false,
46           defaultValue: 0,
47         },
48         notes: {
49           type: DataTypes.TEXT,
50           allowNull: true,
51         },
52         recordedBy: {
53           type: DataTypes.INTEGER,
54           allowNull: false,
55           references: {
56             model: "Users",
57             key: "id",
58           },
59         },
60         recordedAt: {
61           type: DataTypes.DATE,
62           allowNull: false,
63           defaultValue: DataTypes.NOW,
64         },
65       },
66       {
67         sequelize,
68         modelName: "DailyAttendance",
69         schema: "edunova_db",
70         timestamps: true,
71         paranoid: true,
72         indexes: [
73           {
74             unique: true,
75             fields: ["studentId", "attendanceDate"],
76             name: "daily_attendances_student_date_unique",
77           },
78           {
79             fields: ["attendanceDate"],
80             name: "daily_attendances_date_idx",
81           },
82           {
83             fields: ["status"],
84             name: "daily_attendances_status_idx",
85           },
86         ],
87       }
88     );
89   }
90 }
91 /**
92 * Initializes the DailyAttendance associations.
93 *
94 * @param {Object} models - The database models.
95 * @returns {void}
96 */
97 static associate(models) {
98   this.belongsTo(models.Student, {
99     foreignKey: "studentId",
100    as: "student"
101  });
102   this.belongsTo(models.User, {
103     foreignKey: "recordedBy",
104   });
105   this.hasMany(models.ClassAttendance, {
106     foreignKey: "dailyAttendanceId",
107   });
108   this.hasMany(models.AttendanceJustification, {
109     foreignKey: "dailyAttendanceId",
110   });
111 }
112 }
113
114 module.exports = DailyAttendance;

```

*Ilustración XXV: Código fuente del modelo DailyAttendance (Fuente: Elaboración propia)*

```

1  static async getStudentAttendanceHistoryByCycle(
2      studentId,
3      academicCycleId,
4      filters = {}
5  ) {
6      const { startDate, endDate } = filters;
7
8      const whereClause = {
9          studentId,
10         deletedAt: null,
11     };
12
13     // Construir includeClause para filtrar por ciclo académico
14     const includeClause = [
15         {
16             model: Student,
17             as: "student",
18             where: { academicCycleId },
19             attributes: ["id", "status"],
20             include: [
21                 {
22                     model: User,
23                     as: "User",
24                     attributes: ["firstname", "lastname", "secondLastname", "dni"],
25                 },
26                 {
27                     model: AcademicCycle,
28                     attributes: ["id", "name", "startDate", "endDate"],
29                 },
30             ],
31         },
32     ];
33
34     if (startDate && endDate) {
35         whereClause.attendanceDate = {
36             [Op.between]: [startDate, endDate],
37         };
38     }
39
40     // Obtener estadísticas generales
41     const statistics = await this.getStudentAttendanceStatistics(studentId, {
42         startDate,
43         endDate,
44         academicCycleId,
45     });
46
47     // Obtener todos los registros sin paginación
48     const rows = await DailyAttendance.findAll({
49         where: whereClause,
50         include: includeClause,
51         order: [["attendanceDate", "DESC"]],
52     });
53
54     return {
55         attendanceRecords: rows,
56         statistics,
57         studentInfo:
58             rows.length > 0
59             ? {
60                 student: rows[0].student,
61                 academicCycle: rows[0].student.AcademicCycle,
62             }
63             : null,
64     };
65 }

```

Ilustración XXVI: Código fuente de consulta de asistencia de estudiante (Fuente: Elaboración propia)

```

1  const { DataTypes, Model } = require("sequelize");
2
3  class AcademicCycle extends Model {
4    /**
5     * Initializes the AcademicCycle model.
6     *
7     * @param {Object} sequelize - The Sequelize instance.
8     * @returns {Object} - The AcademicCycle model.
9     */
10   static init(sequelize) {
11     return super.init(
12       {
13         id: {
14           type: DataTypes.INTEGER,
15           primaryKey: true,
16           autoIncrement: true,
17         },
18         name: {
19           type: DataTypes.STRING(100),
20           allowNull: false,
21         },
22         status: {
23           type: DataTypes.INTEGER,
24           allowNull: false,
25           defaultValue: 1,
26         },
27         startDate: {
28           type: DataTypes.DATE,
29           allowNull: false,
30         },
31         endDate: {
32           type: DataTypes.DATE,
33           allowNull: false,
34         },
35         syllabuId: {
36           type: DataTypes.INTEGER,
37           allowNull: false,
38           references: {
39             model: "Syllabus",
40             key: "id",
41           },
42         },
43         levelId: {
44           type: DataTypes.INTEGER,
45           allowNull: false,
46           references: {
47             model: "Levels",
48             key: "id",
49           },
50         },
51         academicHourMinutes: {
52           type: DataTypes.INTEGER,
53           allowNull: false,
54           defaultValue: 45,
55           comment:
56             "Cantidad de minutos que dura una hora académica en este ciclo",
57         },
58       },
59       {
60         sequelize,
61         modelName: "AcademicCycle",
62         schema: "edunova_db",
63         timestamps: true,
64         paranoid: true,
65       }
66     );
67   }

```

*Ilustración XXVII: Código fuente del modelo AcademicCycle - campos (Fuente: Elaboración propia)*

```

68
69  /**
70   * Initializes the AcademicCycle associations.
71   *
72   * @param {Object} models - The database models.
73   * @returns {void}
74   */
75  static associate(models) {
76    this.belongsTo(models.Syllabu, { foreignKey: "syllabuId" });
77    this.hasMany(models.Student, {
78      foreignKey: "academicCycleId",
79    });
80    this.belongsTo(models.Level, { foreignKey: "levelId" });
81
82    // Attendance-related associations
83    this.hasMany(models.AttendanceSettings, {
84      foreignKey: "academicCycleId",
85    });
86  }
87 }
88
89 module.exports = AcademicCycle;
90

```

Ilustración XXVIII: Código fuente del modelo AcademicCycle - relaciones (Fuente: Elaboración propia)

```

1  static async recordDailyAttendanceByGrade(req, res) {
2    const { gradeId } = req.params;
3    const { academicCycleId, attendanceDate, attendances, recordedBy } = req.body;
4    try {
5      const result = await AttendanceService.recordDailyAttendanceByGrade(
6        gradeId,
7        academicCycleId,
8        attendanceDate,
9        attendances,
10       recordedBy
11     );
12     res.status(201).json({
13       msg: "Asistencia diaria registrada exitosamente",
14       data: result
15     });
16   } catch (error) {
17     BaseController.handleError(res, "Error al registrar asistencia diaria.", error);
18   }
19 }
20
21 static async getStudentAttendanceHistoryByCycle(req, res) {
22   const { studentId, academicCycleId } = req.params;
23   const { startDate, endDate, page = 1, pageSize = 50 } = req.query;
24
25   try {
26     const attendanceHistory = await AttendanceService.getStudentAttendanceHistoryByCycle(
27       studentId,
28       academicCycleId,
29       { startDate, endDate, page: parseInt(page), pageSize: parseInt(pageSize) }
30     );
31     res.status(200).json({
32       msg: "Historial de asistencia por ciclo académico obtenido exitosamente",
33       data: attendanceHistory
34     });
35   } catch (error) {
36     BaseController.handleError(res, "Error al obtener historial de asistencia por ciclo académico.", error);
37   }
38 }

```

Ilustración XXIX: Código fuente de consulta de asistencia de estudiante (Fuente: Elaboración propia)

### 3.11.3. Funcionalidad 3: Registro de pagos del estudiante

```
1 static async createRegisteredPayment(req, res) {
2   const registeredPaymentData = req.body;
3   try {
4     const storedRegisteredPayment =
5       await RegisteredPaymentService.createRegisteredPayment(
6         registeredPaymentData
7       );
8     res.status(201).json(storedRegisteredPayment);
9   } catch (error) {
10    BaseController.handleError(res, "Error al crear categoría de pago.", error);
11  }
12 }
13
14 static async getRegisteredPayments(req, res) {
15   const { studentId } = req.params;
16   try {
17     const registeredPaymentList =
18       await RegisteredPaymentService.getRegisteredPayments(studentId);
19     res.status(200).send(registeredPaymentList);
20   } catch (error) {
21     BaseController.handleError(res, "Error al obtener pagos del estudiante.", error);
22   }
23 }
```

Ilustración XXX: Código fuente del métodos del controlador de RegisteredPayment (Fuente: Elaboración propia)

```
1 static async createRegisteredPayment(registeredPaymentData) {
2   return await RegisteredPayment.create(registeredPaymentData);
3 }
4
5 static async getRegisteredPayments(studentId) {
6   return await RegisteredPayment.findAll({
7     where: {
8       studentId: studentId,
9       deletedAt: null,
10    },
11    order: [[{ model: PaymentConcept, as: "PaymentConcept" }, "id", "ASC"]],
12    attributes: {
13      exclude: ["createdAt", "updatedAt", "deletedAt"],
14    },
15    include: [
16      {
17        model: PaymentConcept,
18        as: "PaymentConcept",
19        where: { deletedAt: null },
20        attributes: {
21          exclude: ["updatedAt", "deletedAt"],
22        },
23        order: [["createdAt", "ASC"]],
24        required: true,
25      },
26      {
27        model: Student,
28        as: "Student",
29        where: { deletedAt: null },
30        attributes: {
31          exclude: ["updatedAt", "deletedAt"],
32        },
33        order: [["createdAt", "ASC"]],
34        required: true,
35      },
36    ],
37  });
38 }
```

Ilustración XXXI: Código fuente del registro y consulta de pagos del estudiante (Fuente: Elaboración propia)

### 3.11.4. Funcionalidad 4: Registro de notas del estudiante

```
1 static async updateCourseGrades(courseId, gradesToUpdate) {
2     const course = await Course.findByPk(courseId);
3
4     if (!course) {
5         throw new Error("Curso no encontrado");
6     }
7
8     await course.update(gradesToUpdate);
9     return course;
10 }
11
12 static async getCoursesByStudentAndSubject(
13     syllabuSubjectGradeIds,
14     studentIds
15 ) {
16     const syllabuIds = Array.isArray(syllabuSubjectGradeIds)
17         ? syllabuSubjectGradeIds
18         : [syllabuSubjectGradeIds];
19     const students = Array.isArray(studentIds) ? studentIds : [studentIds];
20
21     // Si no hay ids válidos, devolver array vacío
22     if (syllabuIds.length === 0 || students.length === 0) return [];
23
24     return await Course.findAll({
25         where: {
26             syllabuSubjectGradeId: syllabuIds,
27             studentId: students,
28             deletedAt: null,
29         },
30         attributes: ["studentId", "syllabuSubjectGradeId", "averageGrade"],
31         raw: true,
32     });
33 }
```

Ilustración XXXII: Código fuente del registro y consulta de notas del estudiante por curso (Fuente: Elaboración propia)

```
1 static async updateManualGrades(req, res) {
2     const { id } = req.params;
3     const { questionnaireGrade, homeworkGrade, conductGrade, averageGrade } = req.body;
4
5     try {
6         const course = await CourseService.getCourseById(id);
7
8         if (!course) {
9             return res.status(404).send({ msg: "Curso no encontrado" });
10        }
11
12        const gradesToUpdate = {};
13        if (questionnaireGrade !== undefined) gradesToUpdate.questionnaireGrade = questionnaireGrade;
14        if (homeworkGrade !== undefined) gradesToUpdate.homeworkGrade = homeworkGrade;
15        if (conductGrade !== undefined) gradesToUpdate.conductGrade = conductGrade;
16        if (averageGrade !== undefined) gradesToUpdate.averageGrade = averageGrade;
17
18        if (Object.keys(gradesToUpdate).length === 0) {
19            return res.status(400).send({ msg: "No se proporcionaron calificaciones para actualizar" });
20        }
21
22        await CourseService.updateCourseGrades(id, gradesToUpdate);
23
24        const updatedCourse = await CourseService.getCourseById(id);
25        res.status(200).send(updatedCourse);
26    } catch (error) {
27        BaseController.handleError(res, "Error al actualizar calificaciones manuales.", error);
28    }
29 }
```

Ilustración XXXIII: Código fuente del método del controlador de Courses (Fuente: Elaboración propia)

## 3.12. Interfaz del sistema

### 3.12.1. Landing page



Figura I: Banner principal de pantalla de inicio (Fuente: Elaboración propia)

## Sobre Nosotros

### Nuestra Historia

Fundado en 1998, I.E.P. San Isidro ha sido un pilar de excelencia educativa en Zaña por más de dos décadas. Nuestra institución nació con la visión de ofrecer una educación integral que combine los valores tradicionales con metodologías pedagógicas efectivas, preparando a nuestros estudiantes para los desafíos del futuro.

### Nuestro Compromiso

A lo largo de los años, hemos mantenido nuestro compromiso con la formación académica rigurosa y el desarrollo personal de cada estudiante. Nuestro enfoque tradicional se complementa con una atención personalizada, reconociendo que cada niño y adolescente tiene talentos y necesidades únicas que merecen ser atendidas con dedicación y profesionalismo.

 **Excelencia Académica**  
Programas educativos de alta calidad

 **Valores Tradicionales**  
Formación integral y ética

 **Atención Personalizada**  
Cada estudiante es único

 **Metodologías Efectivas**  
Técnicas pedagógicas modernas




**25+**  
Años de excelencia educativa



Figura II: Sección “Sobre Nosotros” (Fuente: Elaboración propia)

## Niveles Educativos

Ofrecemos una educación integral y de calidad en todos nuestros niveles, adaptada a las necesidades específicas de cada etapa del desarrollo.




**Inicial**  
3-5 años

Enfocado en el desarrollo emocional y social a través del juego guiado. Nuestros pequeños aprenden en un ambiente seguro y estimulante que fomenta la curiosidad y sienta las bases para el aprendizaje futuro.

- Desarrollo emocional
- Juego guiado
- Ambiente seguro
- Curiosidad natural

[Conoce más](#)



**Primaria**  
6-11 años

Construimos bases sólidas en matemáticas, lectura y ciencias. Nuestro programa de primaria combina la rigurosidad académica con actividades que desarrollan habilidades sociales y valores fundamentales.

- Matemáticas
- Lectura
- Ciencias
- Valores fundamentales

[Conoce más](#)



**Secundaria**  
12-16 años

Formación académica avanzada y preparación para la universidad. Nuestros estudiantes de secundaria desarrollan pensamiento crítico, habilidades de investigación y la disciplina necesaria para el éxito universitario.

- Pensamiento crítico
- Investigación
- Preparación universitaria
- Disciplina académica

[Conoce más](#)



Figura III: Sección “Niveles Educativos” (Fuente: Elaboración propia)

**I.E.P. SAN ISIDRO**    Sobre Nosotros    Niveles    Metodología    Noticias    Plataforma Educativa

## Metodología y Valores

Nuestro enfoque educativo combina métodos tradicionales con técnicas modernas para formar estudiantes integrales y preparados para los desafíos del futuro.

### Educación Tradicional

Mantenemos un enfoque educativo tradicional con disciplina y compromiso, valorando el esfuerzo y la perseverancia como pilares del aprendizaje.

### Valores Fundamentales

Inculcamos respeto, liderazgo y responsabilidad en cada estudiante, formando no solo buenos alumnos sino también excelentes ciudadanos.

### Clases Presenciales

Ofrecemos educación 100% presencial, priorizando la interacción directa entre profesores y alumnos para un aprendizaje más efectivo y personalizado.

### Nuestros Valores Fundamentales

- Respeto
- Responsabilidad
- Excelencia
- Compromiso

En I.E.P. San Isidro, creemos que la educación va más allá de los conocimientos académicos. Formamos personas íntegras con valores sólidos que les permitirán enfrentar los desafíos de la vida con determinación y ética.

↑

Figura IV: Sección “Metodología y Valores” (Fuente: Elaboración propia)

## Noticias y eventos

Mantente informado sobre los últimos eventos y actividades en nuestra institución educativa.

**Calendario de Eventos** < Enero 2026 >

Lun	Mar	Mié	Jue	Vie	Sáb	Dom
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15 Sesión de Lectura ...	16	17	18
19	20	21	22	23	24 Actividad de Integ...	25
26	27 Campaña de Salu...	28	29	30	31	

### Próximos eventos




- 
**Actividad de Integración Familiar - Enero 2026**
📅 24 de enero de 2026
  
 Jornada recreativa para estudiantes y familias
 [→](#)
- 
**Campaña de Salud y Bienestar - Enero 2026**
📅 27 de enero de 2026
  
 Jornada de promoción de hábitos saludables
 [→](#)
- 
**Sesión de Lectura Compartida - Febrero 2026**
📅 15 de febrero de 2026
  
 Club de lectura para estudiantes y padres
 [→](#)

Figura V: Sección “Noticias y Eventos” (Fuente: Elaboración propia)


ESTAMOS PARA ATENDERTE


## Contáctanos


¿Tienes alguna pregunta sobre nuestra institución? Estamos aquí para ayudarte.


### Información de contacto

Completa el formulario y te responderemos a la brevedad.

 [informes@iepsanisidro.edu.pe](mailto:informes@iepsanisidro.edu.pe)

 +51 991 234 567

 Nombre completo

 Correo electrónico

 Mensaje


 Enviar mensaje



Figura VI: Sección “Contáctanos” (Fuente: Elaboración propia)



### Iniciar sesión

**Correo electrónico**

**Contraseña**

[Entrar](#)






**I.E.P. SAN ISIDRO**  
Excelencia educativa

Formando líderes para triunfar desde 1998.  
Comprometidos con una educación de  
calidad y valores firmes para las futuras  
generaciones.



#### Contacto

-  Calle Independencia 765, Zaña
-  [info@sanisidro.edu.pe](mailto:info@sanisidro.edu.pe)
-  (074) 123-4567

#### Enlaces rápidos

- [Inicio](#)
- [Sobre Nosotros](#)
- [Niveles](#)
- [Metodología](#)
- [Noticias](#)

#### Horario de atención

Lunes - Viernes: **7:30 - 15:30**

¿Tienes alguna consulta? Estamos para ayudarte.

[Contáctanos aquí →](#)

Figura VII: Login de ingreso a la Plataforma Educativa (Fuente: Elaboración propia)

### 3.12.2. Plataforma educativa

#### Rol administrativo

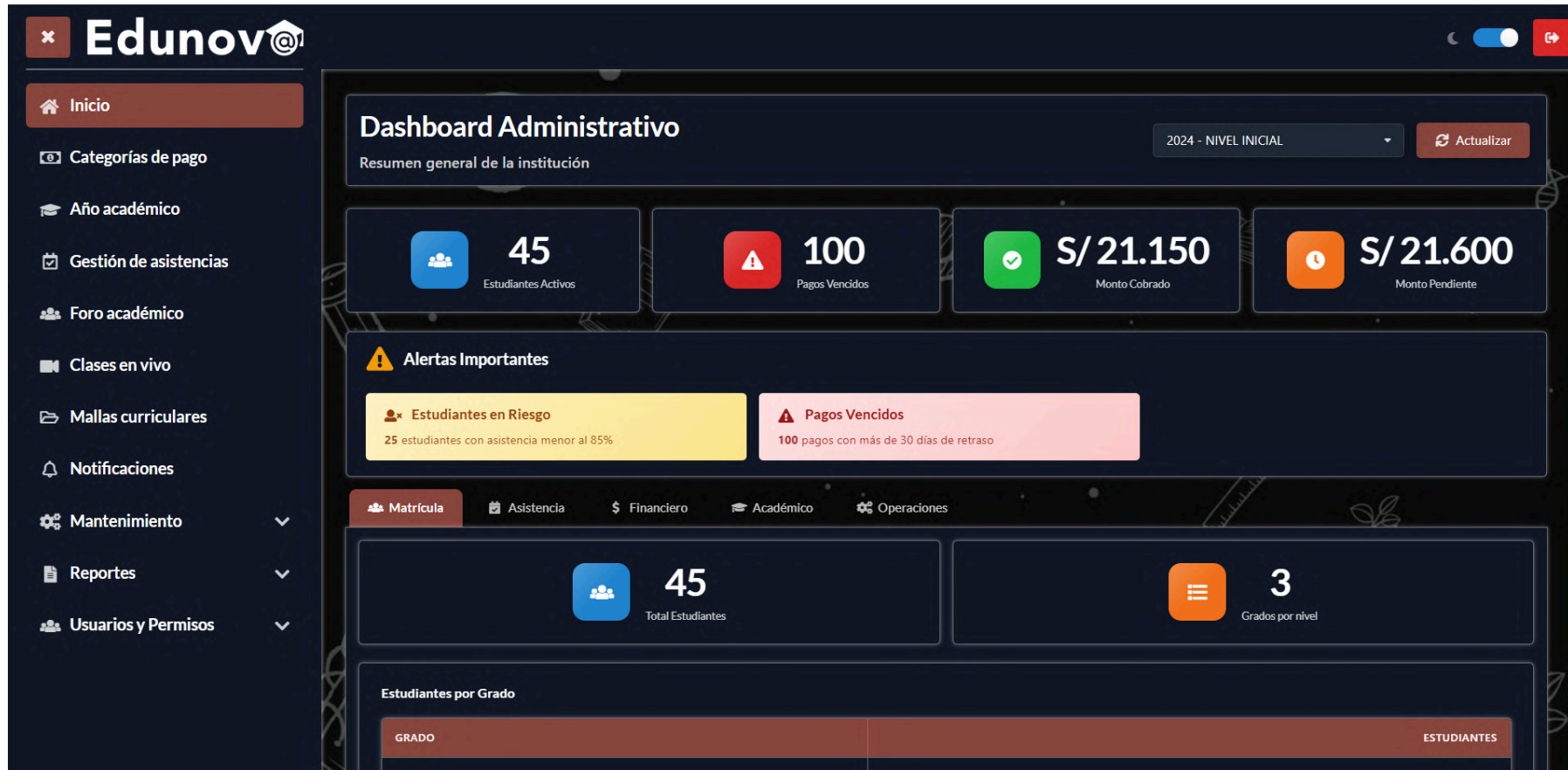


Figura VIII: Dashboard Administrativo - Tema oscuro (Fuente: Elaboración propia)

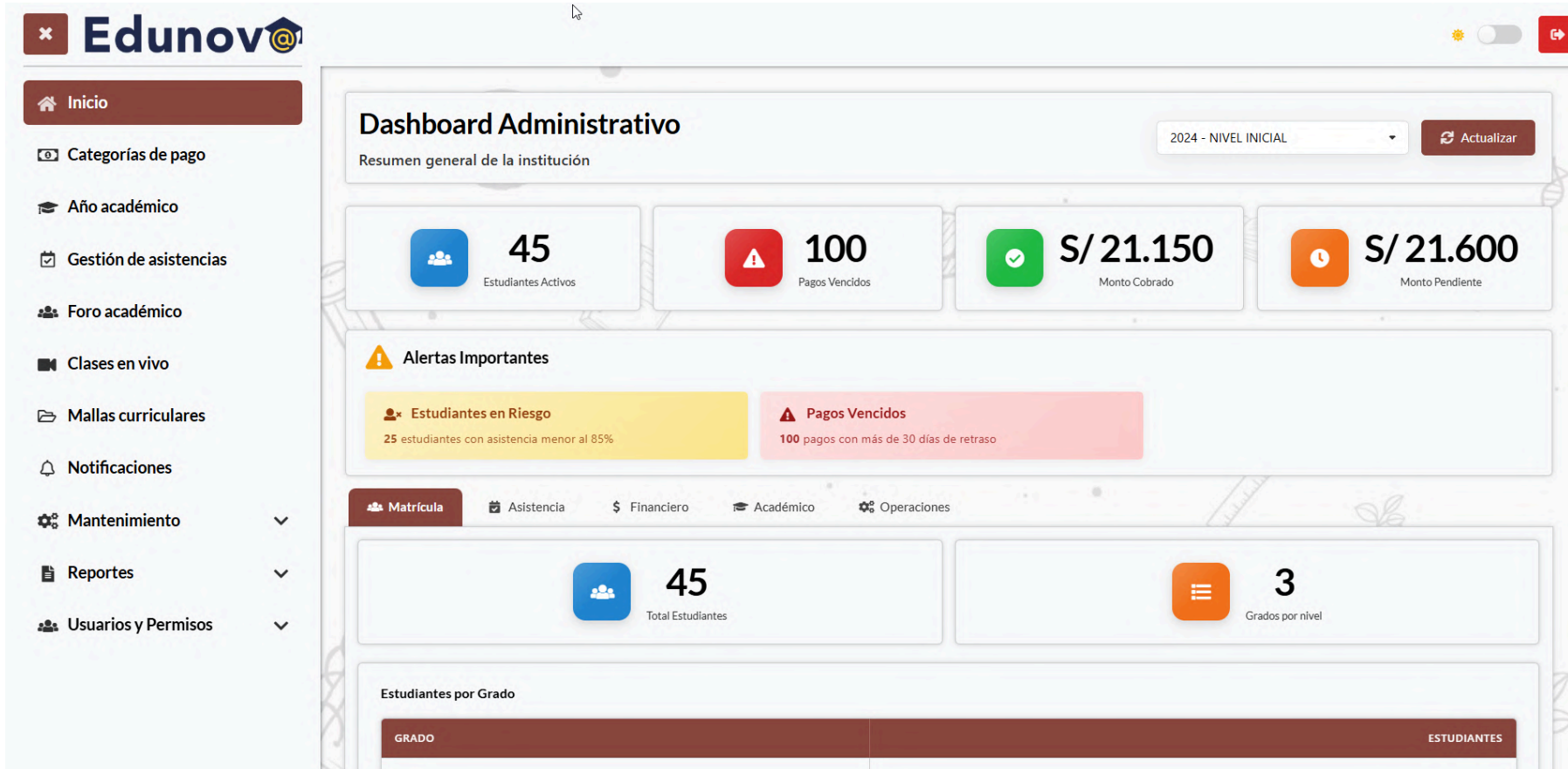


Figura IX: Dashboard Administrativo - Tema claro (Fuente: Elaboración propia)

The screenshot displays the Edunov@ interface with a modal window titled "Gestionar conceptos de pago de: 2024". The modal contains a form with the following fields:

- Nombre de concepto de pago:** A text input field.
- S/. Monto:** A numeric input field.
- Fecha de vencimiento:** A date picker set to "dd/mm/aaaa".

Below the form is a button labeled "Agregar concepto de pago". Underneath is a table with the following data:

CONCEPTO DE PAGO	MONTO	FECHA DE VENCIMIENTO	ACCIONES
Matrícula	S/. 150.00	14/02/2024	[Edit] [Delete]
Cuota Marzo	S/. 80.00	30/03/2024	[Edit] [Delete]
Cuota Abril	S/. 80.00	29/04/2024	[Edit] [Delete]
Cuota Mayo	S/. 80.00	30/05/2024	[Edit] [Delete]
Cuota Junio	S/. 80.00	29/06/2024	[Edit] [Delete]
Cuota Julio	S/. 80.00	30/07/2024	[Edit] [Delete]
Cuota Agosto	S/. 80.00	30/08/2024	[Edit] [Delete]
Cuota Septiembre	S/. 80.00	29/09/2024	[Edit] [Delete]
Cuota Octubre	S/. 80.00	30/10/2024	[Edit] [Delete]

Figura X: Categorías de Pago - Gestión de conceptos de pago (Fuente: Elaboración propia)



















































**Edunov**

Inicio  
Categorías de pago  
**Año académico**  
Gestión de asistencias  
Foro académico  
Clases en vivo  
Mallas curriculares  
Notificaciones  
Mantenimiento  
Reportes  
Usuarios y Permisos

**Estudiantes matriculados en Año académico 2025 - NIVEL PRIMARIA**

PRIMERO SEGUNDO TERCERO CUARTO QUINTO SEXTO Matricular estudiante

Buscar por nombre... Buscar por DNI... Buscar por teléfono... Buscar Limpiar filtros

APELLIDOS Y NOMBRES	DNI	CORREO ELECTRÓNICO	TELÉFONO	CATEGORÍA DE PAGO	ESTADO	ACCIONES
García Ortega Sofía	10000015	estudiante66@edunova.com	900000015	2025	ACTIVO	    
García Molina Manuel	10000016	estudiante67@edunova.com	900000016	2025	ACTIVO	    
García Ramírez Patricia	10000017	estudiante68@edunova.com	900000017	2025	ACTIVO	    
García Ortiz Jorge	10000018	estudiante69@edunova.com	900000018	2025	ACTIVO	    
García Núñez Lucía	10000019	estudiante70@edunova.com	900000019	2025	ACTIVO	    
García Castillo Ricardo	10000020	estudiante71@edunova.com	900000020	2025	ACTIVO	    
García Iglesias Valentina	10000021	estudiante72@edunova.com	900000021	2025	ACTIVO	    
García Guerrero Roberto	10000022	estudiante73@edunova.com	900000022	2025	ACTIVO	    
García Méndez Andrea	10000023	estudiante74@edunova.com	900000023	2025	ACTIVO	    
García Cortés Fernando	10000024	estudiante75@edunova.com	900000024	2025	ACTIVO	    

Mostrando 1-10 de 15 registros Registros por página: 10 << < 1 2 > >>

Figura XI: Año Académico - Gestión de alumnos (Fuente: Elaboración propia)

**Edunov**

Inicio

Categorías de pago

**Año académico**

Gestión de asistencias

Foro académico

Clases en vivo

Mallas curriculares

Notificaciones

Mantenimiento

Reportes

Usuarios y Permisos

Estudiantes matriculados en Año académico 2025 - NIVEL INICIAL

3 AÑOS 4 AÑOS 5 AÑOS

Buscar por nombre... Buscar por DNI... Buscar por teléfono... Buscar Limpiar filtros

APELLIDOS Y NOMBRES	DNI	REGISTRO ACADÉMICO						
		ASIGNATURA	PROGRESO	CUESTIONARIOS	TRABAJO FINAL	CONDUCTA	RECUPERACIÓN	PROMEDIO
Fernández Ruiz Eva	10000195	Religión	50%	15	13	12	-	13
		Ciencia, ambiente y salud	31%	11	17	15	-	14
		Ciencias sociales	71%	17	14	13	-	15
		Idiomas	47%	14	15	15	-	15
		Educación para el trabajo	7%	17	10	16	-	14
		Matemática	39%	12	16	18	-	15
		Comunicación	65%	13	18	12	-	14
		Educación física	73%	11	19	13	-	14
		Arte y cultura	66%	13	13	11	-	9
		Desarrollo personal y ciudadano	91%	20	20	16	-	19

Figura XII: Año Académico - Gestión Académica (Fuente: Elaboración propia)

**Edunov**

Inicio | Categorías de pago | Año académico | Gestión de asistencias | Foro académico | Clases en vivo | Mallas curriculares | Notificaciones | Mantenimiento | Reportes | Usuarios y Permisos

3 AÑOS | 4 AÑOS | 5 AÑOS

+ Nuevo horario para 3 AÑOS

Hora	Lunes	Martes	Miércoles	Jueves	Viernes
08:00 - 08:50	<b>Religión</b> Docente: Profesor Inicial Aula: Aula Inicial A Religión - 3 AÑOS (2025) - Ses...	<b>Educación para el trabajo</b> Docente: Profesor Inicial Aula: Aula Inicial A Educación para el trabajo - 3 ...	<b>Educación física</b> Docente: Profesor Inicial Aula: Aula Inicial A Educación física - 3 AÑOS (202...	<b>Ciencias sociales</b> Docente: Profesor Inicial Aula: Aula Inicial A Ciencias sociales - 3 AÑOS (20...	<b>Comunicación</b> Docente: Profesor Inicial Aula: Aula Inicial A Comunicación - 3 AÑOS (2025) ...
08:50 - 09:40	<b>Comunicación</b> Docente: Profesor Inicial Aula: Aula Inicial A Comunicación - 3 AÑOS (2025) ...	<b>Religión</b> Docente: Profesor Inicial Aula: Aula Inicial A Religión - 3 AÑOS (2025) - Ses...	<b>Educación para el trabajo</b> Docente: Profesor Inicial Aula: Aula Inicial A Educación para el trabajo - 3 ...	<b>Arte y cultura</b> Docente: Profesor Inicial Aula: Aula Inicial A Arte y cultura - 3 AÑOS (2025)...	<b>Ciencias sociales</b> Docente: Profesor Inicial Aula: Aula Inicial A Ciencias sociales - 3 AÑOS (20...
09:40 - 10:30	<b>Ciencias sociales</b> Docente: Profesor Inicial Aula: Aula Inicial A Ciencias sociales - 3 AÑOS (20...	<b>Comunicación</b> Docente: Profesor Inicial Aula: Aula Inicial A Comunicación - 3 AÑOS (2025) ...	<b>Religión</b> Docente: Profesor Inicial Aula: Aula Inicial A Religión - 3 AÑOS (2025) - Ses...	<b>Matemática</b> Docente: Profesor Inicial Aula: Aula Inicial A Matemática - 3 AÑOS (2025) - S...	<b>Arte y cultura</b> Docente: Profesor Inicial Aula: Aula Inicial A Arte y cultura - 3 AÑOS (2025)...
10:30 - 10:45	<b>RECREO</b>				
10:45 - 11:35	<b>Arte y cultura</b> Docente: Profesor Inicial Aula: Aula Inicial A Arte y cultura - 3 AÑOS (2025)...	<b>Idiomas</b> Docente: Profesor Inicial Aula: Aula Inicial A Idiomas - 3 AÑOS (2025) - Sesi...	<b>Comunicación</b> Docente: Profesor Inicial Aula: Aula Inicial A Comunicación - 3 AÑOS (2025) ...	<b>Ciencia, ambiente y salud</b> Docente: Profesor Inicial Aula: Aula Inicial A Ciencia, ambiente y salud - 3 ...	<b>Matemática</b> Docente: Profesor Inicial Aula: Aula Inicial A Matemática - 3 AÑOS (2025) - S...
11:35 - 12:25	<b>Matemática</b> Docente: Profesor Inicial Aula: Aula Inicial A Matemática - 3 AÑOS (2025) - S...	+ Agregar	<b>Idiomas</b> Docente: Profesor Inicial Aula: Aula Inicial A Idiomas - 3 AÑOS (2025) - Sesi...	<b>Comunicación</b> Docente: Profesor Inicial Aula: Aula Inicial A Comunicación - 3 AÑOS (2025) ...	<b>Ciencia, ambiente y salud</b> Docente: Profesor Inicial Aula: Aula Inicial A Ciencia, ambiente y salud - 3 ...
12:25 - 13:15	<b>Ciencia, ambiente y salud</b> Docente: Profesor Inicial Aula: Aula Inicial A Ciencia, ambiente y salud - 3 ...	<b>Matemática</b> Docente: Profesor Inicial Aula: Aula Inicial A Matemática - 3 AÑOS (2025) - S...	+ Agregar	<b>Idiomas</b> Docente: Profesor Inicial Aula: Aula Inicial A Idiomas - 3 AÑOS (2025) - Sesi...	<b>Comunicación</b> Docente: Profesor Inicial Aula: Aula Inicial A Comunicación - 3 AÑOS (2025) ...
13:15 - 14:00	<b>ALMUERZO</b>				
14:00 - 14:50	<b>Educación física</b> Docente: Profesor Inicial Aula: Aula Inicial A Educación física - 3 AÑOS (202...	<b>Ciencia, ambiente y salud</b> Docente: Profesor Inicial Aula: Aula Inicial A Ciencia, ambiente y salud - 3 ...	<b>Matemática</b> Docente: Profesor Inicial Aula: Aula Inicial A Matemática - 3 AÑOS (2025) - S...	+ Agregar	<b>Idiomas</b> Docente: Profesor Inicial Aula: Aula Inicial A Idiomas - 3 AÑOS (2025) - Sesi...

Figura XIII: Año Académico - Horarios de clase (Fuente: Elaboración propia)

**Edunov**

**Asignar docentes en ciclo 2025**

Grado académico: 4 AÑOS

Asignatura: [Selecione]

Docente: [Selecione] **Asignar docente**

GRADO	ASIGNATURA	DOCENTE	ACCIONES
4 AÑOS	Religión	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Ciencia, ambiente y salud	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Ciencias sociales	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Idiomas	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Educación para el trabajo	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Matemática	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Comunicación	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Educación física	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Arte y cultura	Inicial Cuatro Profesor	[Icono de borrar]
4 AÑOS	Desarrollo personal y ciudadano	Inicial Cuatro Profesor	[Icono de borrar]

Figura XIV: Año Académico - Asignación de Docentes (Fuente: Elaboración propia)

**Edunov**

Inicio

Categorías de pago

Año académico

**Gestión de asistencias**

Foro académico

Clases en vivo

Mallas curriculares

Notificaciones

Mantenimiento

Reportes

Usuarios y Permisos

Asistencia General - 2025

PRIMERO SEGUNDO **TERCERO** CUARTO QUINTO SEXTO

+ Registrar asistencia masiva - TERCERO

DNI	ESTUDIANTE	CORREO ELECTRÓNICO	ESTADO	ACCIONES
10000045	García Blanco Fernanda	estudiante96@edunova.com	Activo	Ver Asistencia + Registrar
10000046	García Suárez Eduardo	estudiante97@edunova.com	Activo	Ver Asistencia + Registrar
10000047	García Rubio Julia	estudiante98@edunova.com	Activo	Ver Asistencia + Registrar
10000048	García Delgado Álvaro	estudiante99@edunova.com	Activo	Ver Asistencia + Registrar
10000049	García Morales Sara	estudiante100@edunova.com	Activo	Ver Asistencia + Registrar
10000050	García Marín Rubén	estudiante101@edunova.com	Activo	Ver Asistencia + Registrar
10000051	García Medina Clara	estudiante102@edunova.com	Activo	Ver Asistencia + Registrar
10000052	García Santos Emilio	estudiante103@edunova.com	Activo	Ver Asistencia + Registrar
10000053	García Garrido Alba	estudiante104@edunova.com	Activo	Ver Asistencia + Registrar
10000054	García Lozano Marcos	estudiante105@edunova.com	Activo	Ver Asistencia + Registrar

Figura XV: Gestión de Asistencia (Fuente: Elaboración propia)

**Registro Masivo de Asistencia - 3 AÑOS**

Fecha (Solo días hábiles: Lunes a Viernes) \*  
 30/12/2025

Acciones Masivas  
 Todos Presentes Todos Ausentes

Seleccionar todos los estudiantes Seleccionados: 0 | Presentes: 0 | Ausentes: 0 | Tardanzas: 0 | Justificados: 0

	ESTUDIANTE	DNI	ESTADO	HORA DE LLEGADA	NOTAS
<input type="checkbox"/>	Fernández Ruiz Eva	10000195	Presente	07:30	Notas...
<input type="checkbox"/>	Fernández Moreno Ignacio	10000196	Presente	07:30	Notas...
<input type="checkbox"/>	Fernández Álvarez Alicia	10000197	Presente	07:30	Notas...
<input type="checkbox"/>	Fernández Gutiérrez Martín	10000198	Presente	07:30	Notas...
<input type="checkbox"/>	Fernández Navarro Claudia	10000199	Presente	07:30	Notas...
<input type="checkbox"/>	Fernández Domínguez Gabriel	10000200	Presente	07:30	Notas...
<input type="checkbox"/>	Fernández Vázquez Lidia	10000201	Presente	07:30	Notas...
<input type="checkbox"/>	Fernández Serrano Santiago	10000202	Presente	07:30	Notas...

Cancelar Registrar Asistencia de 0 estudiantes seleccionados

Figura XVI: Gestión de Asistencia - Registro de Asistencia (Fuente: Elaboración propia)

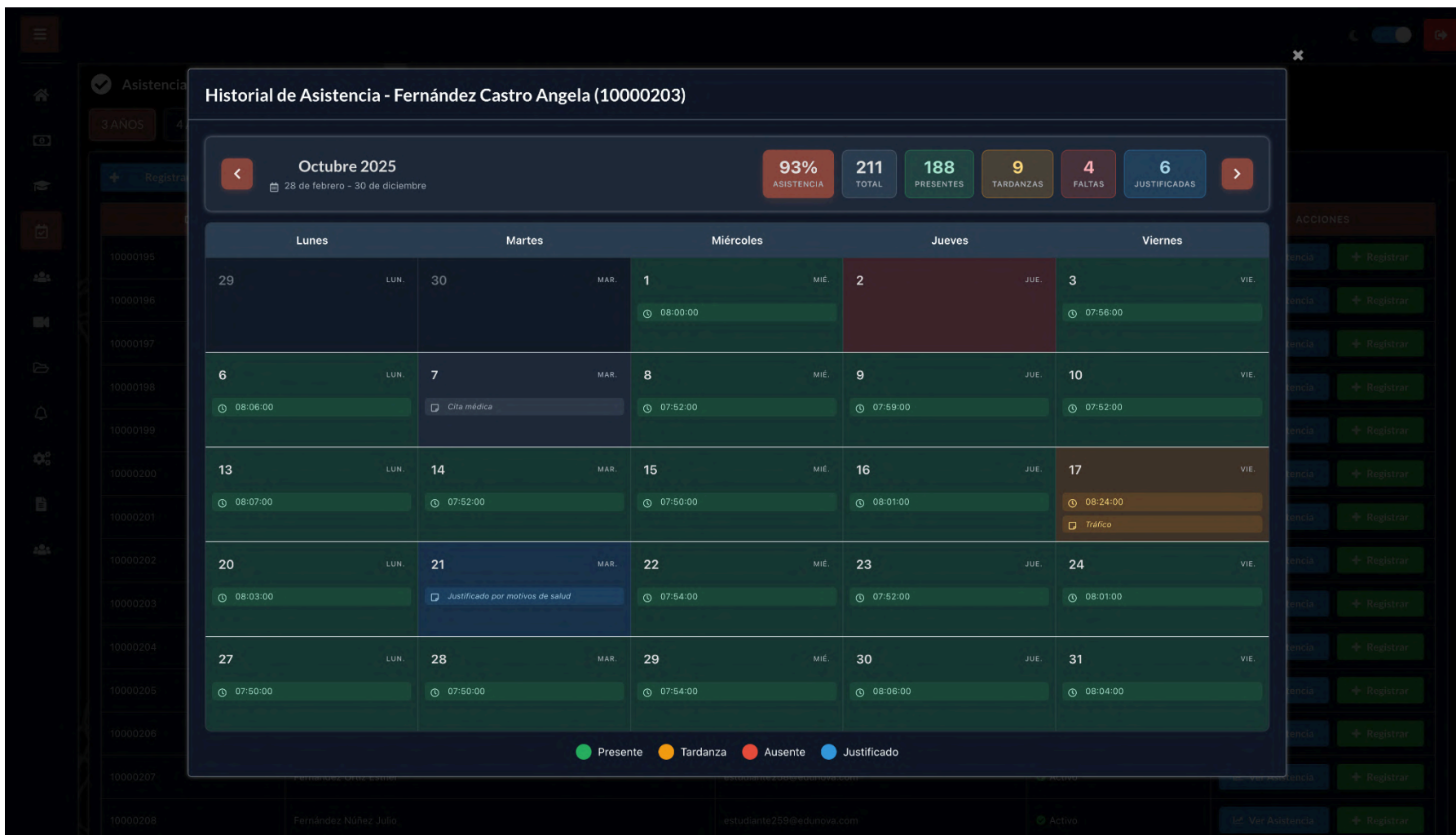


Figura XVII: Gestión de Asistencia - Historial de Asistencia (Fuente: Elaboración propia)

The screenshot displays the Edunov forum interface. On the left is a dark sidebar with navigation options: Inicio, Categorías de pago, Año académico, Gestión de asistencias, **Foro académico** (highlighted), Clases en vivo, Mallas curriculares, Notificaciones, Mantenimiento, Reportes, and Usuarios y Permisos. The main content area is titled 'Detalle de la publicación' and features a post by María García García about writing an argumentative essay. Below the post are two responses. The first response, from 'User Three Teacher', provides a detailed structure for an argumentative essay and is marked as 'Correcta'. The second response, from 'Ana García Sánchez', offers tips on using logical connectors and is also marked as 'Correcta'. At the bottom, there is a 'Respuesta' section with a text input field and an 'Añadir respuesta' button.

Figura XVIII: Foro Académico (Fuente: Elaboración propia)

**Edunov**

Inicio

Categorías de pago

Año académico

Gestión de asistencias

Foro académico

**Clases en vivo**

Mallas curriculares

Notificaciones

Mantenimiento

Reportes

Usuarios y Permisos

Nueva clase en vivo

Selecciona estado

Buscar por nombre o descripción...

Buscar

Limpiar filtros

NOMBRE	DESCRIPCIÓN	FECHA Y HORA	DURACIÓN (MIN)	ACCESO	CLASE EN VIVO
Comunicación - Comprensión Lectora	Técnicas de comprensión lectora y análisis de textos	23/01/2026 12:58 pm	45	COMM01	<a href="https://meet.jit.si/comunicacion-comprension-lectora-1768931882832">https://meet.jit.si/comunicacion-comprension-lectora-1768931882832</a>
Historia del Perú - Época Colonial	Análisis de la época colonial en el Perú y sus principales características	22/01/2026 12:58 pm	90	HIST01	<a href="https://meet.jit.si/historia-peru-colonial-1768931882832">https://meet.jit.si/historia-peru-colonial-1768931882832</a>
Matemáticas Básicas - Álgebra	Clase de introducción al álgebra para estudiantes de primer año	21/01/2026 12:58 pm	60	MATH01	<a href="https://meet.jit.si/matematicas-basicas-algebra-1768931882832">https://meet.jit.si/matematicas-basicas-algebra-1768931882832</a>
Tutoría Matemáticas Avanzadas	Sesión de tutoría para resolver dudas de matemáticas avanzadas	20/01/2026 12:58 pm	30	TUT001	<a href="https://meet.jit.si/tutoria-matematicas-avanzadas-1768931882832">https://meet.jit.si/tutoria-matematicas-avanzadas-1768931882832</a>
Reunión General de Profesores	Reunión mensual para coordinar actividades académicas	19/01/2026 12:58 pm	120	MEET01	<a href="https://meet.jit.si/reunion-profesores-1768931882832">https://meet.jit.si/reunion-profesores-1768931882832</a>

Mostrando 1-5 de 5 registros    Registros por página: 10

<< < 1 > >>

Figura XIX: Clases en Vivo (Fuente: Elaboración propia)

**Gestionar asignaturas de malla curricular: 2024**

Selecciona un grado: 5 AÑOS ×      Selecciona una asignatura: [Dropdown]      Agregar asignatura y grado

5 AÑOS	Religión	[Eliminar]
5 AÑOS	Ciencia, ambiente y salud	[Eliminar]
5 AÑOS	Ciencias sociales	[Eliminar]
5 AÑOS	Idiomas	[Eliminar]
5 AÑOS	Educación para el trabajo	[Eliminar]
5 AÑOS	Matemática	[Eliminar]
5 AÑOS	Comunicación	[Eliminar]
5 AÑOS	Educación física	[Eliminar]
5 AÑOS	Arte y cultura	[Eliminar]
5 AÑOS	Desarrollo personal y ciudadano	[Eliminar]

Figura XX: Malla Curricular - Gestión de asignaturas (Fuente: Elaboración propia)



**Edunov**

- Inicio
- Categorías de pago
- Año académico
- Gestión de asistencias
- Foro académico
- Clases en vivo
- Mallas curriculares**
- Notificaciones
- Mantenimiento
- Reportes
- Usuarios y Permisos

**Religión**


- S1: La creación del mundo
    - ¿Cómo Dios creó al mundo?
    - Reflexionamos
    - Cuestionario
  - S2: El bautismo
  - S3: La Semana Santa
  - S4: Reforzando la Cuaresma
  - S5: Jesucristo, Inspiración de Amo
  - S6: El Trabajo: Medio de Progreso
  - S7: Significado de la misa
  - S8: Los sacramentos
  - S9: La comunidad cristiana
  - S10: El prójimo
  - S11: La promesa de salvación
  - S12: Los patriarcas
  - S13: Los jueces
  - S14: Los reyes
  - S15: Los profetas
  - S16: La oración
  - S17: Encarnación del hijo de Dios
  - S18: María: pureza y humildad
  - S19: Tiempo de adviento
  - S20: Navidad
  - S21: TRABAJO FINAL - RELIGIÓN

**Contenido de sesión: ¿Cómo Dios creó al mundo?**

**GÉNESIS (Primeros capítulos)**

La historia de la creación del mundo se encuentra en Génesis, el primer libro de la Biblia. Esta historia bíblica es importante porque introduce a Dios como el creador del mundo y como el único Dios. La creencia más básica del cristianismo nos dice que:

"Dios es creador de todas las cosas "



Anterior

Página 1 de 9

Siguiente

Figura XXII: Malla Curricular - Visualización de contenido multimedia (Fuente: Elaboración propia)

FECHA	TIPO	TITULO	MENSAJE	ESTADO	ACCIONES
23/02/2025 10:45 am	GENERAL	Actualización del sistema	El sistema se actualizará con nuevas funcionalidades esta noche.	NO LEIDA	✓ ✖
22/02/2025 07:30 am	CURSO	Evaluación docente	Es momento de evaluar a tus profesores del semestre anterior.	NO LEIDA	✓ ✖
21/02/2025 04:15 am	FORO	Tutorial disponible	Nuevo tutorial: "Cómo usar efectivamente el foro estudiantil".	NO LEIDA	✓ ✖
20/02/2025 11:30 am	PAGO	Reembolso procesado	Tu solicitud de reembolso ha sido procesada exitosamente.	LEIDA	✓ ✖
19/02/2025 08:45 am	CALIFICACIÓN	Proyecto grupal	Tu grupo ha sido asignado para el proyecto final de Gestión.	LEIDA	✓ ✖
18/02/2025 05:20 am	GENERAL	Graduación 2025	Ya están abiertas las inscripciones para la ceremonia de graduación.	NO LEIDA	✓ ✖
17/02/2025 12:15 pm	CURSO	Certificado listo	Tu certificado del curso de Excel Avanzado está listo.	NO LEIDA	✓ ✖
16/02/2025 09:45 am	FORO	Nuevo moderador	Has sido nombrado moderador del foro de Ciencias.	LEIDA	✓ ✖
15/02/2025 06:30 am	PAGO	Bono estudiantil	Has recibido un bono de \$50 por tu excelente rendimiento.	LEIDA	✓ ✖
14/02/2025 03:15 am	CALIFICACIÓN	Recuperación disponible	Puedes presentar recuperación para el examen de Álgebra.	NO LEIDA	✓ ✖

Mostrando 1-10 de 40 registros    Registros por página: 10

Figura XXIII: Gestión de Notificaciones (Fuente: Elaboración propia)

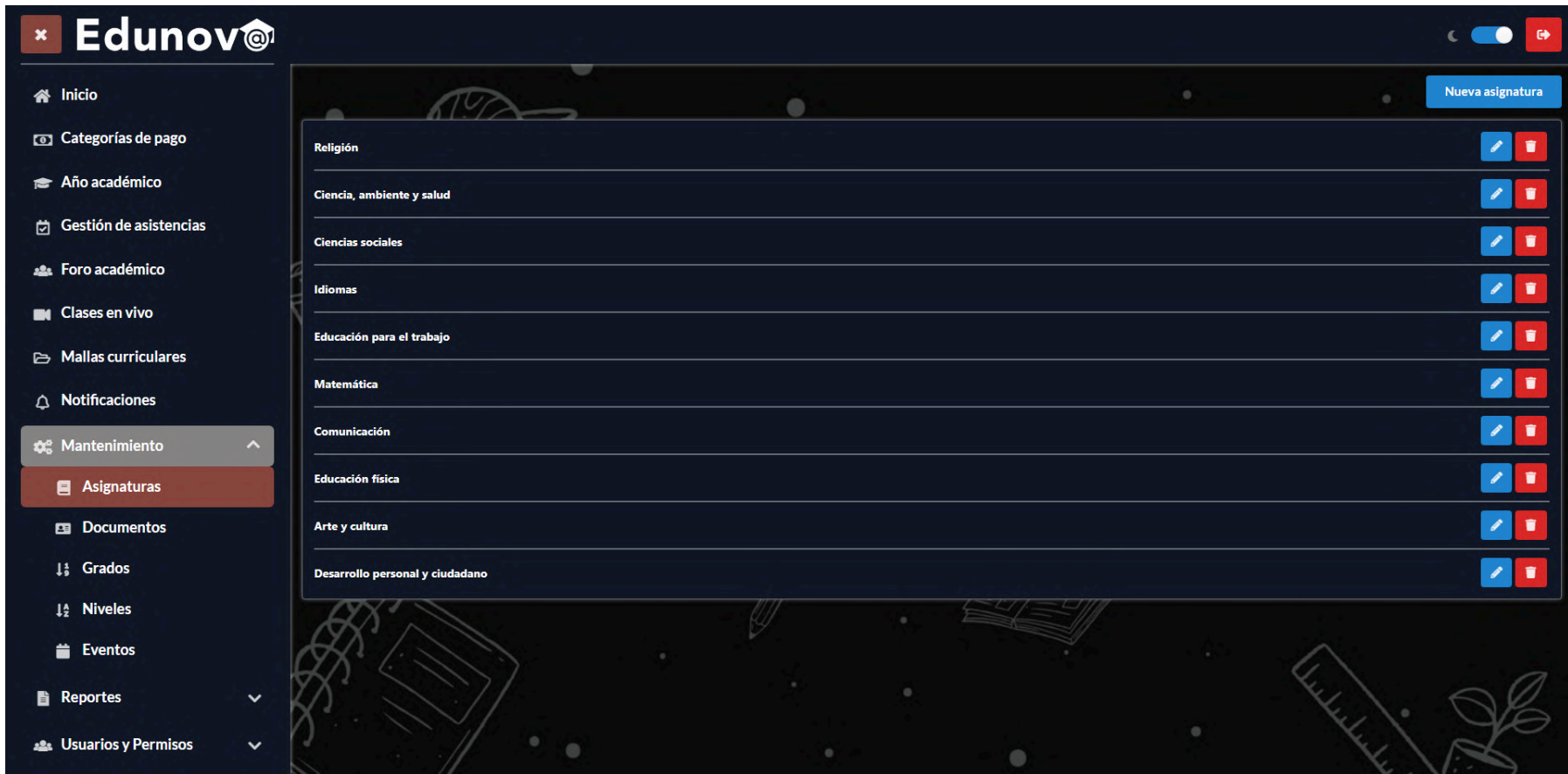


Figura XXIV: Mantenedores: Asignaturas, Documentos, Grados, Niveles, Eventos (Fuente: Elaboración propia)

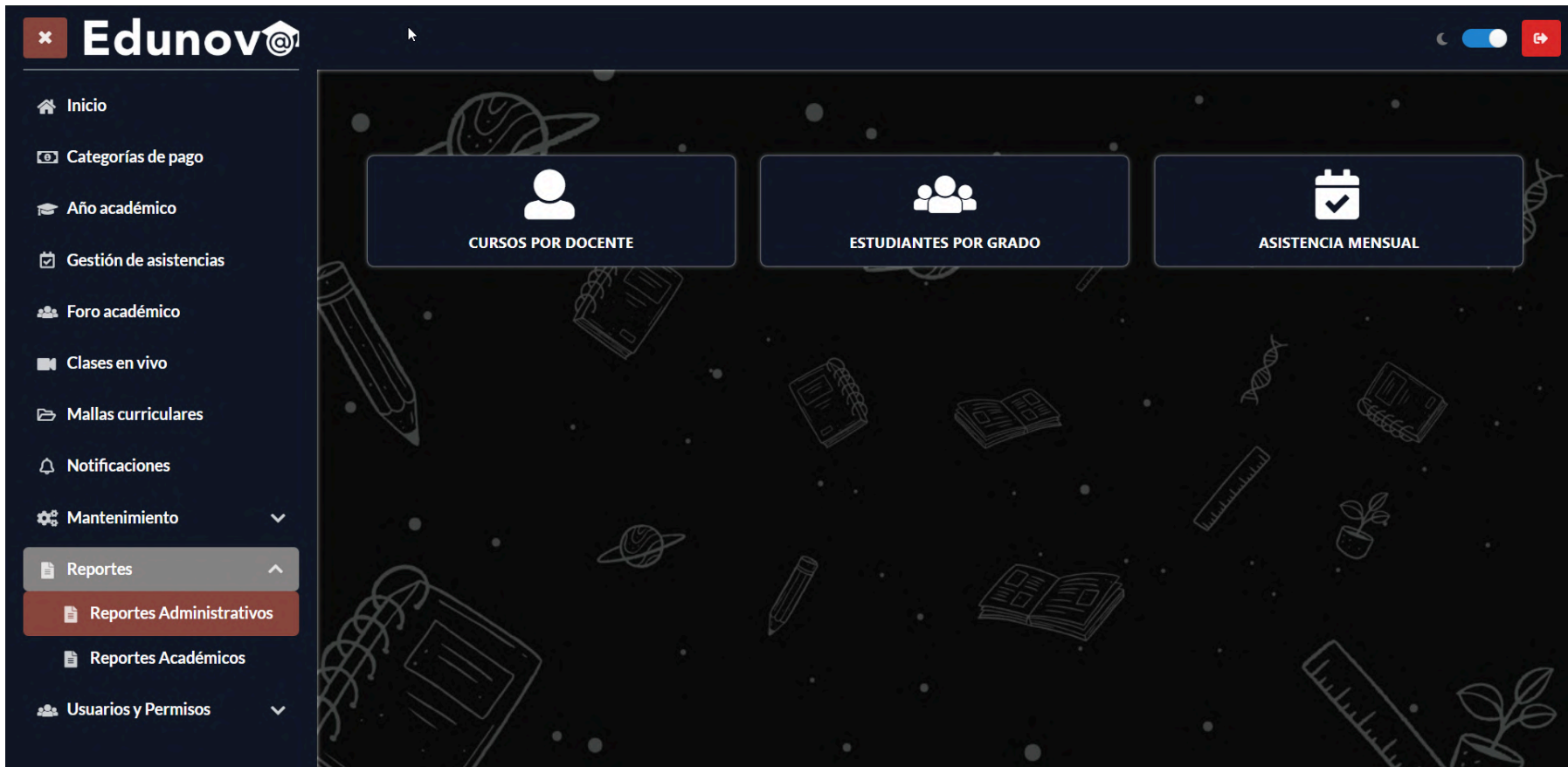


Figura XXV: Reportes - Administrativos (Fuente: Elaboración propia)

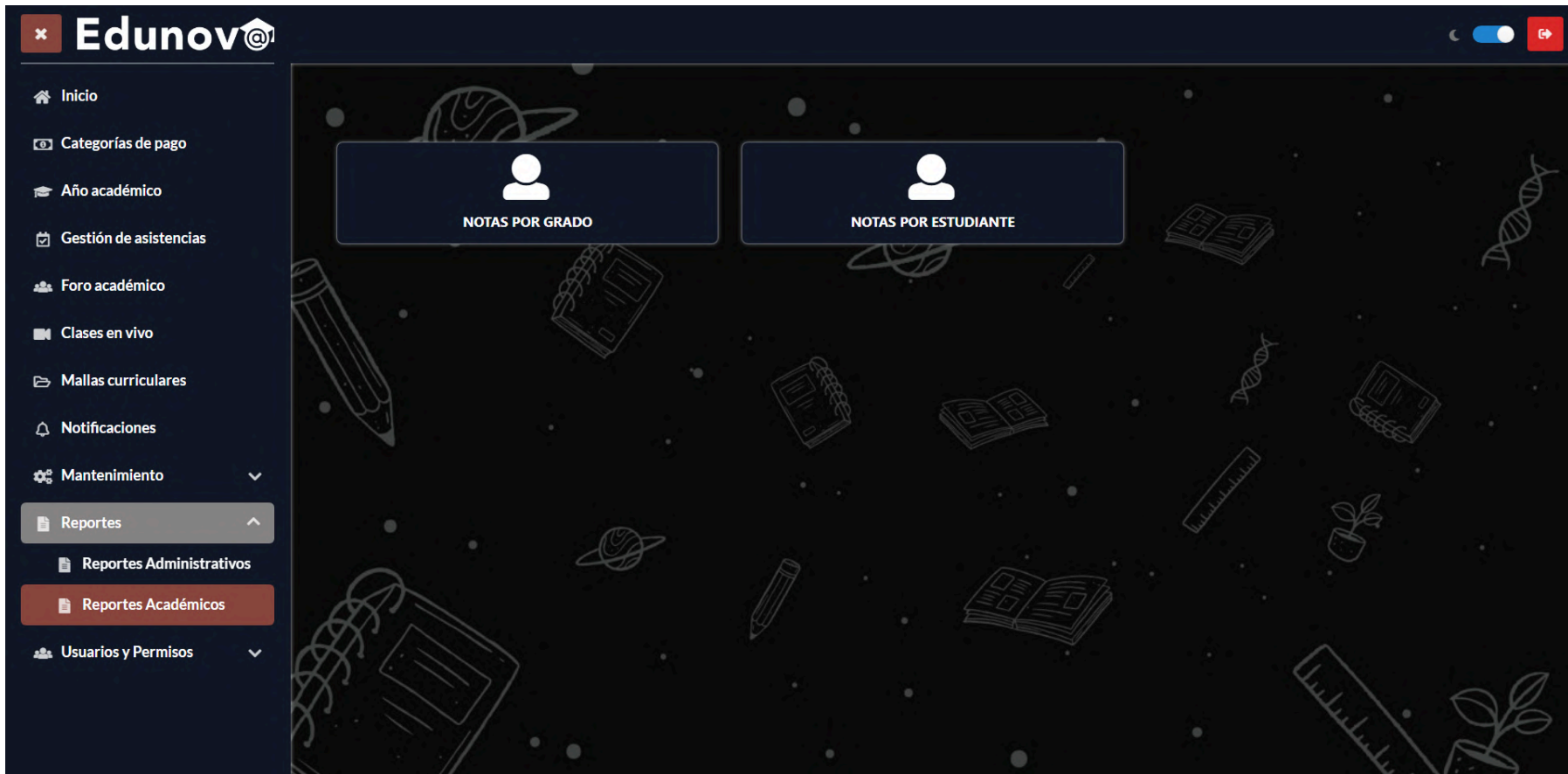


Figura XXVI: Reportes - Académicos (Fuente: Elaboración propia)

**Edunov**

- Inicio
- Categorías de pago
- Año académico
- Gestión de asistencias
- Foro académico
- Clases en vivo
- Mallas curriculares
- Notificaciones
- Mantenimiento
- Reportes
- Usuarios y Permisos**
  - Usuarios**
  - Tipos de usuario
  - Permisos

Nuevo usuario

Buscar por nombre... | Buscar por DNI... | Buscar por teléfono... | Buscar | Limpiar filtros

AVATAR	NOMBRE COMPLETO	DNI	EMAIL	TELÉFONO	ESTADO	ACCIONES
	User Two Admin	87654321	user2@example.com	0987654321	ACTIVO	
	User Three Teacher	11223344	user3@example.com	1122334455	ACTIVO	
	User Four Teacher	44332211	user4@example.com	5544332211	ACTIVO	
	Parent One Guardian	55667788	parent1@example.com	5566778899	ACTIVO	
	Parent Two Guardian	88776655	parent2@example.com	9988776655	ACTIVO	
	Profesor Inicial Tres	56000037	teacher.inicial3@edunova.com	9600000370	ACTIVO	
	Profesor Inicial Cuatro	56000038	teacher.inicial4@edunova.com	9600000380	ACTIVO	
	Profesor Inicial Cinco	56000039	teacher.inicial5@edunova.com	9600000390	ACTIVO	
	Profesor Primaria Primero	56000040	teacher.primaria1@edunova.com	9600000400	ACTIVO	
	Admin User Admin	12345678	admin@gmail.com	1234567890	ACTIVO	

Mostrando 1-10 de 245 registros | Registros por página: 10 | << < 1 2 3 4 5 ... 25 > >>

Figura XXVII: Usuarios y Permisos - Usuarios (Fuente: Elaboración propia)

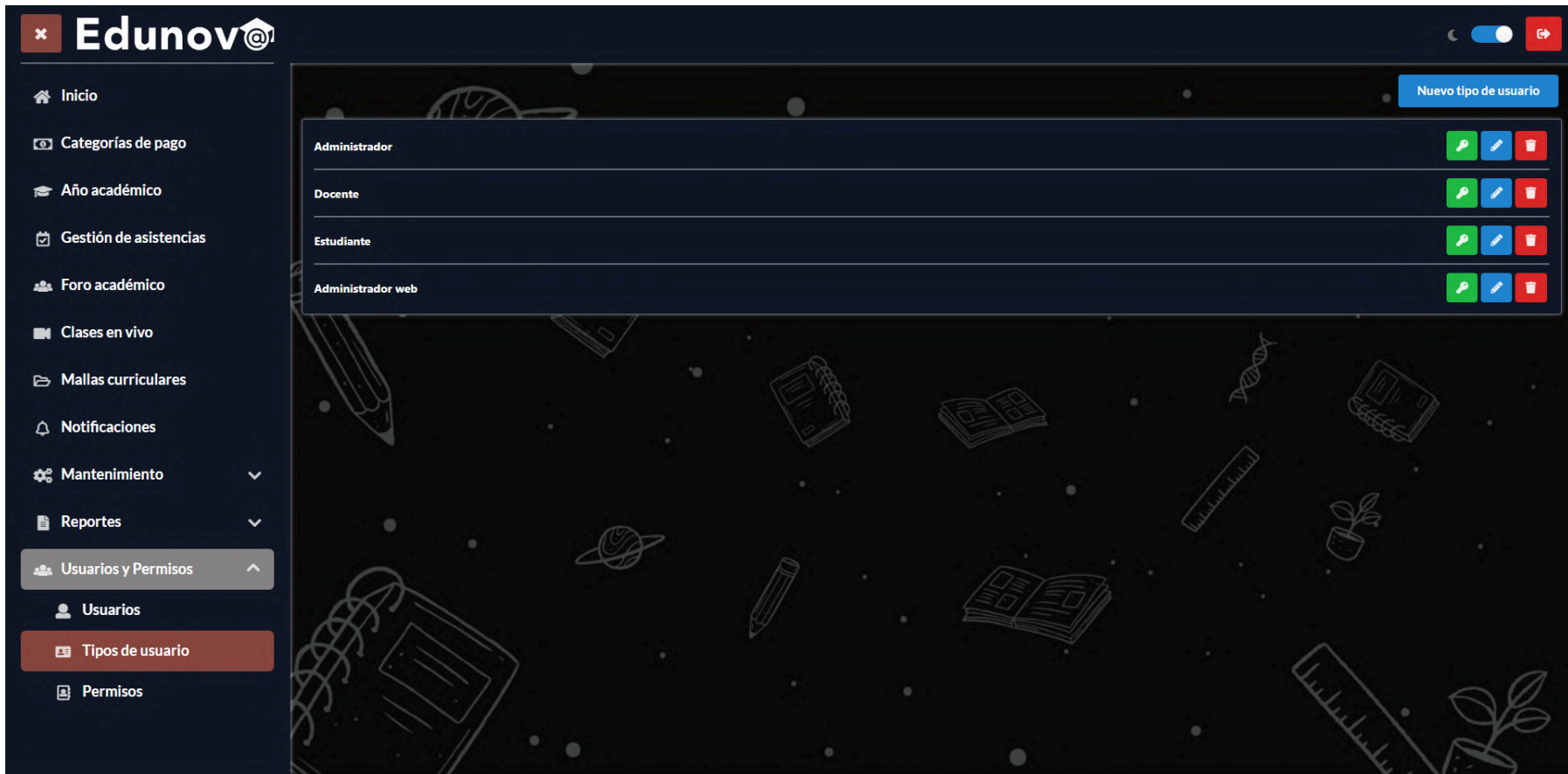


Figura XXVIII: Usuarios y Permisos - Tipos de usuario (Fuente: Elaboración propia)

The screenshot displays the Edunov web application interface. On the left is a dark sidebar with navigation items: Inicio, Categorías de pago, Año académico, Gestión de asistencias, Foro académico, Clases en vivo, Mallas curriculares, Notificaciones, Mantenimiento, Reportes, Usuarios y Permisos (highlighted), Usuarios, Tipos de usuario, and Permisos (highlighted in red). The main content area has a dark background and features a dropdown menu at the top left labeled 'Selecciona una sección'. A blue button labeled 'Nuevo Permiso' is in the top right. The main area contains a list of permissions, each with a title, section, description, and edit/delete icons.

Acción	Sección	Descripción	Editar	Eliminar
Listar usuarios	Usuarios	Listar usuarios registrados	✎	🗑️
Crear usuario	Usuarios	Crear usuario	✎	🗑️
Actualizar usuario	Usuarios	Actualizar usuario registrado	✎	🗑️
Eliminar usuario	Usuarios	Eliminar usuario registrado	✎	🗑️
Activar / Desactivar usuario	Usuarios	Activar o desactivar usuario registrado	✎	🗑️
Listar tipos de usuario	Tipos de usuario	Listar tipos de usuario registrados	✎	🗑️
Crear tipo de usuario	Tipos de usuario	Crear tipo de usuario	✎	🗑️
Actualizar tipo de usuario	Tipos de usuario	Actualizar tipo de usuario registrado	✎	🗑️
Eliminar tipo de usuario	Tipos de usuario	Eliminar tipo de usuario registrado	✎	🗑️

Figura XXIX: Usuarios y Permisos - Permisos (Fuente: Elaboración propia)

## Rol docente

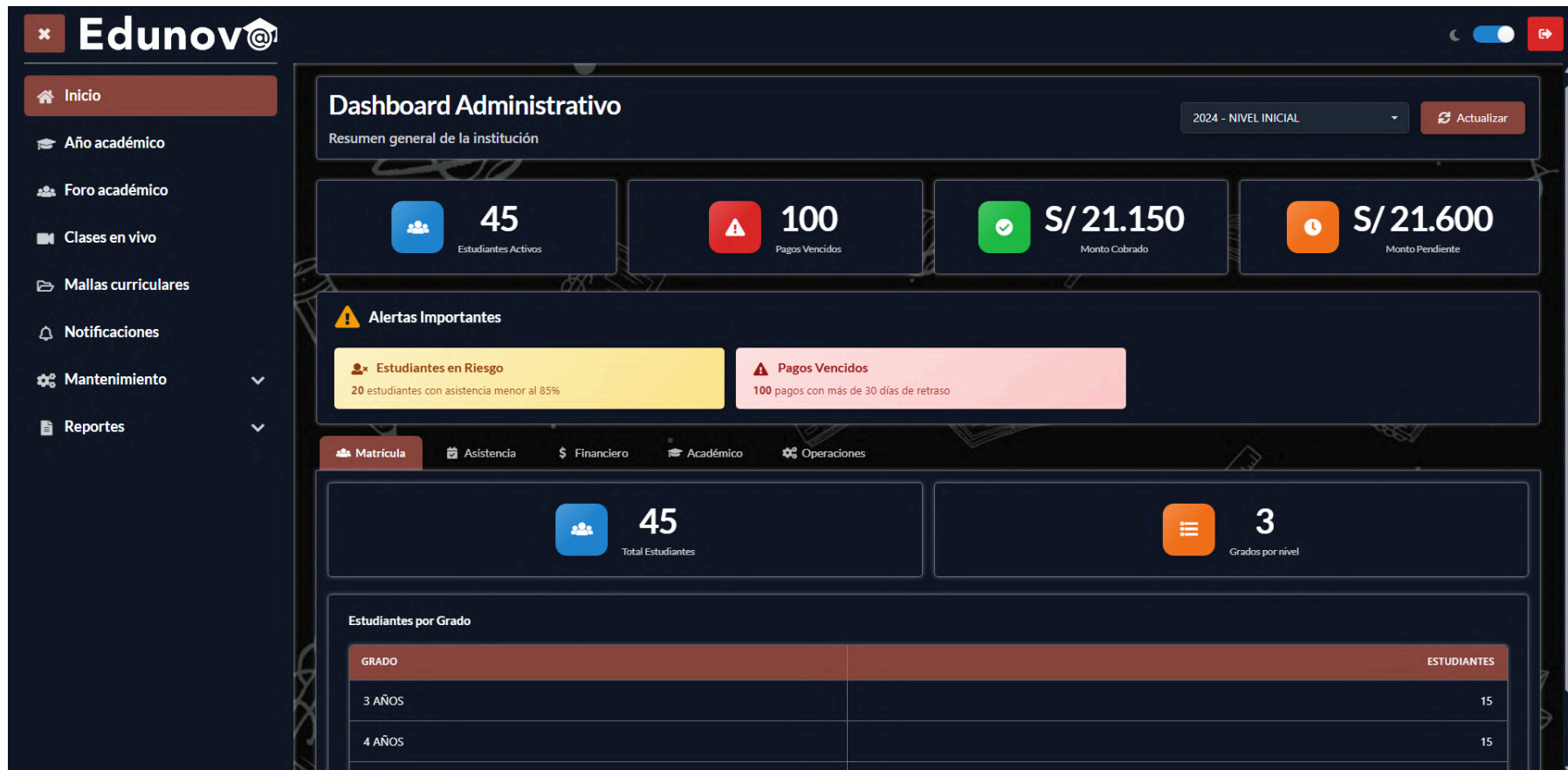


Figura XXX: Dashboard Docente (Fuente: Elaboración propia)

- Como se puede observar en la barra lateral de la izquierda, el docente comparte ciertos accesos como el administrador pero en menor medida.

## Rol estudiante

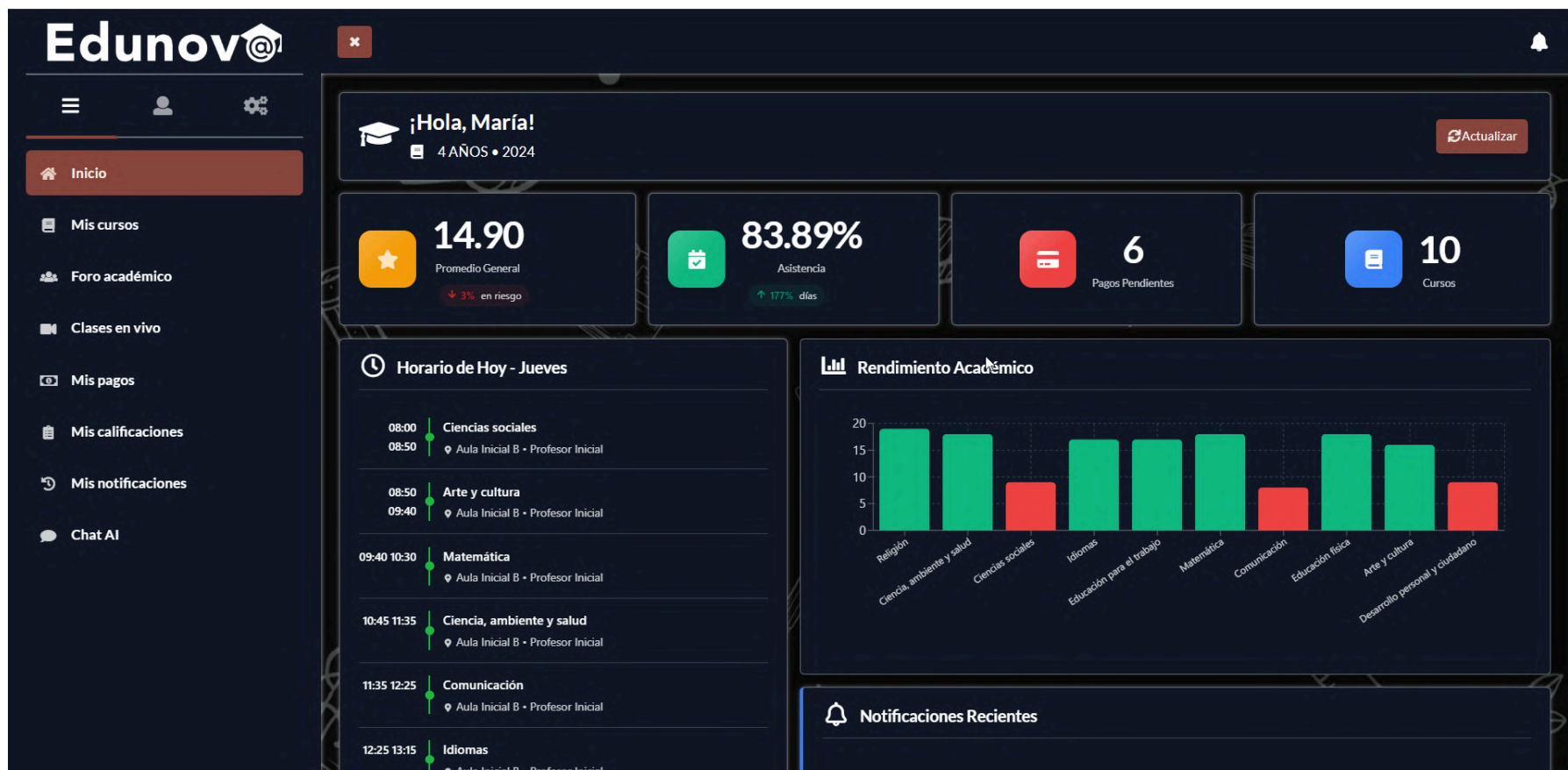


Figura XXXI: Dashboard Estudiante (Fuente: Elaboración propia)

**Edunov**

## Mis Cursos

Aquí puedes ver todos tus cursos actuales, sus profesores y tu progreso en cada uno.

Curso	Profesor	Progreso del curso	Ver notas	Ir al curso
Religión	Prof. Profesor Inicial	1%	Ver notas	Ir al curso
Ciencia, ambiente y salud	Prof. Profesor Inicial	58%	Ver notas	Ir al curso
Ciencias sociales	Prof. Profesor Inicial	14%	Ver notas	Ir al curso
Idiomas	Prof. Profesor Inicial	20%	Ver notas	Ir al curso
Educación para el trabajo	Prof. Profesor Inicial	12%	Ver notas	Ir al curso
Matemática	Prof. Profesor Inicial	65%	Ver notas	Ir al curso

Figura XXXII: Mis Cursos (Fuente: Elaboración propia)

Figura XXXIII: Foro Académico (Fuente: Elaboración propia)

**Edunov**

## Clases en Vivo

Accede a tus clases virtuales programadas, en progreso y finalizadas. Haz clic en una fila para ver detalles o unirte si la clase está activa.

Programada  [Limpiar filtros](#)

NOMBRE	DESCRIPCIÓN	FECHA Y HORA	DURACIÓN	PROFESOR	ESTADO	ACCIÓN
Matemáticas Básicas - Álgebra	Clase de introducción al álgebra para estudiantes de primer año	21/01/2026 12:58	60	User Three	PROGRAMADA	
Historia del Perú - Época Colonial	Análisis de la época colonial en el Perú y sus principales características	22/01/2026 12:58	90	User Four	PROGRAMADA	
Comunicación - Comprensión Lectora	Técnicas de comprensión lectora y análisis de textos	23/01/2026 12:58	45	Profesor Inicial	PROGRAMADA	

Mostrando 1-3 de 3 registros    Registros por página:     << < 1 > >>

Figura XXXIV: Clases en Vivo (Fuente: Elaboración propia)

**Edunov**

## Mis Pagos

Aquí puedes ver todos tus pagos de tu año académico seleccionado, además de tu comprobante subido en cada pago.

Año académico: 2024 ✕ 🔄 Limpiar filtros

CONCEPTO DE PAGO	MONTO	CÓDIGO DE TRANSACCIÓN	FECHA DE VENCIMIENTO	FECHA DE PAGO	MÉTODO DE PAGO	ESTADO	ACCIONES
Matrícula	S/. 150.00	TXN000001	14/02/2024	09/02/2024	EFFECTIVO	REALIZADO	👁️
Cuota Marzo	S/. 80.00	TXN000002	30/03/2024	14/03/2024	YAPE	REALIZADO	👁️
Cuota Abril	S/. 80.00	TXN000003	29/04/2024	14/04/2024	PLIN	REALIZADO	👁️
Cuota Mayo	S/. 80.00	TXN000004	30/05/2024	14/05/2024	TRANSFERENCIA BANCARIA	REALIZADO	👁️
Cuota Junio	S/. 80.00	TXN000005	29/06/2024	14/06/2024	OTROS	REALIZADO	👁️
Cuota Julio	S/. 80.00		30/07/2024	N/A	SIN DEFINIR	PENDIENTE	👁️
Cuota Agosto	S/. 80.00		30/08/2024	N/A	SIN DEFINIR	PENDIENTE	👁️
Cuota Septiembre	S/. 80.00		29/09/2024	N/A	SIN DEFINIR	PENDIENTE	👁️
Cuota Octubre	S/. 80.00		30/10/2024	N/A	SIN DEFINIR	PENDIENTE	👁️
Cuota Noviembre	S/. 80.00		29/11/2024	N/A	SIN DEFINIR	PENDIENTE	👁️
Cuota Diciembre	S/. 80.00		30/12/2024	N/A	SIN DEFINIR	PENDIENTE	👁️

Figura XXXV: Mis Pagos (Fuente: Elaboración propia)

**Edunov** ✕ 🔔

☰ 👤 ⚙️

- Inicio
- Mis cursos
- Foro académico
- Clases en vivo
- Mis pagos
- Mis calificaciones**
- Mis notificaciones
- Chat AI

## Mis Calificaciones

Aquí puedes ver todos tus calificaciones de tu año académico y grado seleccionado, además de tu historial académico.

Año académico: **NIVEL INICIAL - 2025** ✕ 🔄 Limpiar filtros

ASIGNATURA	PROGRESO	CUESTIONARIOS	TRABAJO FINAL	CONDUCTA	RECUPERACIÓN	PROMEDIO	ESTADO	ACCIONES
Religión	71 %	10	17	19	-	15	APROBADO	📄
Ciencia, ambiente y salud	32 %	12	11	10	-	11	APROBADO	📄
Ciencias sociales	34 %	14	13	16	-	14	APROBADO	📄
Idiomas	17 %	12	16	17	-	15	APROBADO	📄
Educación para el trabajo	36 %	20	15	14	-	16	APROBADO	📄
Matemática	74 %	20	18	16	-	18	APROBADO	📄
Comunicación	1 %	13	20	10	-	14	APROBADO	📄
Educación física	18 %	20	18	18	-	19	APROBADO	📄
Arte y cultura	62 %	13	10	11	-	11	APROBADO	📄
Desarrollo personal y ciudadano	47 %	18	11	17	-	15	APROBADO	📄

Figura XXXVI: Mis Calificaciones (Fuente: Elaboración propia)

**Edunov**

## Mis Notificaciones

Aquí puedes ver todas tus notificaciones del sistema, incluyendo información sobre calificaciones, pagos, cursos y eventos importantes de tu proceso educativo.

Filtrar por tipo: [v]    Filtrar por estado: [v]    Buscar por título o mensaje... [v]    Limpiar filtros [v]    Marcar todas como leídas [v]

FECHA	TIPO	TÍTULO	MENSAJE	ESTADO	ACCIONES
22/01/2026 08:01 pm	FORO	Respuesta rechazada	Tu respuesta en la publicación "asdasd" ha sido rechazada por el docente.	NO LEÍDA	[✓] [✖]
22/01/2026 08:01 pm	FORO	Respuesta incorrecta	Tu respuesta en la publicación "asdasd" ha sido marcada como incorrecta.	NO LEÍDA	[✓] [✖]
22/01/2026 08:01 pm	FORO	Respuesta correcta	Tu respuesta en la publicación "asdasd" ha sido marcada como correcta.	NO LEÍDA	[✓] [✖]
20/01/2026 04:50 pm	FORO	Publicación resuelta	La publicación "asdasd" ha sido resuelta.	NO LEÍDA	[✓] [✖]
20/01/2026 04:50 pm	FORO	Respuesta aprobada	Tu respuesta en la publicación "asdasd" ha sido aprobada y se muestra en el foro.	NO LEÍDA	[✓] [✖]
20/01/2026 04:49 pm	FORO	Publicación aprobada	La publicación "asdasd" ha sido aprobada y se muestra en el foro.	NO LEÍDA	[✓] [✖]

Mostrando 1-6 de 6 registros    Registros por página: 10    << < 1 > >>

Figura XXXVII: Mis Notificaciones (Fuente: Elaboración propia)

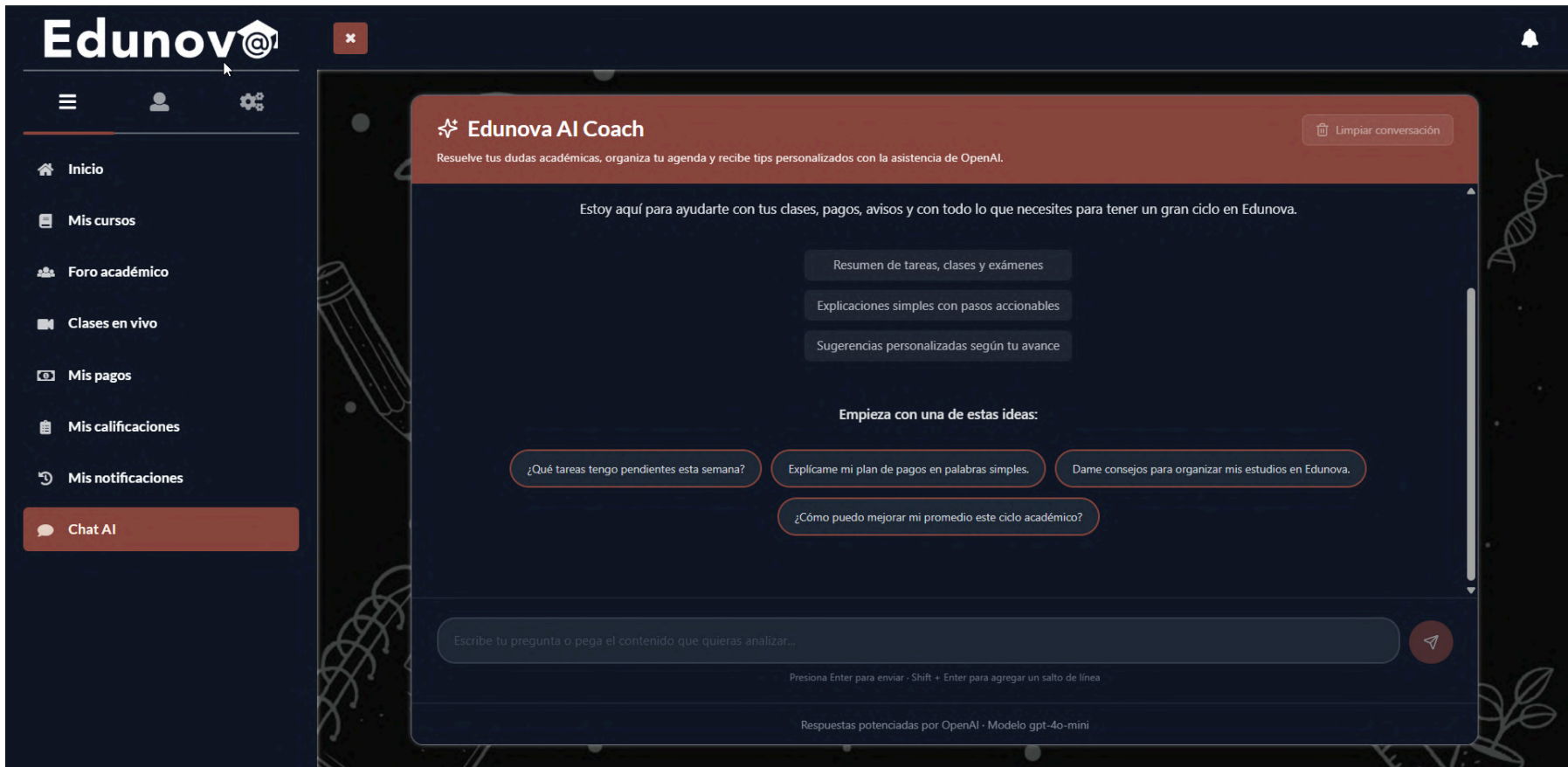


Figura XXXVIII: Chat AI (Fuente: Elaboración propia)

## **4. RESULTADOS**

### **4.1. Validación de la plataforma educativa propuesta**

#### **4.1.1. Objetivo de la validación**

El presente apartado tiene como objetivo validar el correcto funcionamiento del sistema académico propuesto, verificando el cumplimiento de las funcionalidades definidas en el backlog y los módulos del sistema, mediante la aplicación de un instrumento de validación basado en checklist y escala Likert, evaluado por expertos.

#### **4.1.2. Aplicación del checklist de funcionalidades y del instrumento de validación**

Con el propósito de validar el sistema desarrollado, se aplicaron dos herramientas de evaluación, las cuales permiten una verificación objetiva y estructurada de las funcionalidades implementadas.

##### **Checklist de funcionalidades del sistema**

El checklist de funcionalidades fue elaborado a partir del backlog del proyecto, considerando todas las tareas y requerimientos funcionales definidos durante el desarrollo del sistema.

Este checklist permitió verificar si cada funcionalidad:

- Fue implementada correctamente.
- Se encuentra operativa.
- Cumple con el propósito para el cual fue diseñada.

La evaluación se realizó a un experto, miembro de la institución educativa para que cerciore cada punto planteado en nuestro Anexo 1.

Cada ítem fue marcado según su estado de cumplimiento, permitiendo corroborar que el sistema cumple con las funcionalidades planteadas en el diseño del proyecto.

**ANEXO 1: CHECKLIST TÉCNICO DE VERIFICACIÓN DEL BACKLOG LA PLATAFORMA EDUCATIVA**

El presente checklist tiene como finalidad verificar el cumplimiento de las tareas definidas en el Product Backlog de la plataforma educativa, permitiendo constatar si cada funcionalidad fue implementada correctamente. Este instrumento será utilizado como evidencia de funcionamiento del sistema.

**Escala de verificación**

✓ = Implementado | ✘ = No implementado | O = Implementado parcialmente

**PERFIL ADMINISTRATIVO**

**MÓDULO: AUTENTICACIÓN Y SEGURIDAD**

Nº	Funcionalidad	Estado	Observaciones
1	Inicio de sesión con credenciales.	✓	
2	Cierre de sesión seguro.	✓	
3	Control de acceso por roles.	✓	
4	Protección de rutas y vistas.	✓	

**MÓDULO: USUARIOS Y PERMISOS**

Nº	Funcionalidad	Estado	Observaciones
1	Registro de permisos.	✓	
2	Edición y eliminación de permisos.	✓	
3	Registro de tipo de usuarios.	✓	
4	Edición y eliminación de tipos usuarios.	✓	
5	Gestión de permisos por tipo de usuario.	✓	
6	Registro usuarios.	✓	
7	Edición y eliminación de usuarios.	✓	

*Figura XXXVIII: ANEXO 1 - Checklist parte 1 (Fuente: Elaboración propia)*

**MÓDULO: MANTENIMIENTO**

Nº	Funcionalidad	Estado	Observaciones
1	Registro de asignaturas.	✓	
2	Edición y eliminación de asignaturas.	✓	
3	Registro de documentos.	✓	
4	Edición y eliminación de documentos.	✓	
5	Registro de grados.	✓	
6	Edición y eliminación de grados.	✓	
7	Registro de niveles.	✓	
8	Edición y eliminación de niveles.	✓	
9	Registro de categoría de eventos.	✓	
10	Edición y eliminación de categorías de eventos.	✓	
11	Registro de eventos.	✓	
12	Edición y eliminación de eventos.	✓	

**MÓDULO: DASHBOARD ADMINISTRATIVO**

Nº	Funcionalidad	Estado	Observaciones
1	Filtros por años académicos.	✓	
2	Alertas de asistencia de estudiantes.	✓	
3	Información de asistencias.	✓	
4	Gestión de ingresos, cobros pendientes, montos cobrados, etc.	✓	

Figura XXXIX: ANEXO 1 - Checklist parte 2 (Fuente: Elaboración propia)

**MÓDULO: CATEGORÍAS DE PAGO**

Nº	Funcionalidad	Estado	Observaciones
1	Registro de categorías de pago.	✓	
2	Edición y eliminación de categorías de pago.	✓	
3	Creación y asignación de conceptos de pago.	✓	
4	Edición y eliminación de conceptos de pago.	✓	

**MÓDULO: MALLAS CURRICULARES**

Nº	Funcionalidad	Estado	Observaciones
1	Registro de mallas curriculares.	✓	
2	Edición y eliminación de mallas curriculares.	✓	
3	Asignación de asignaturas a la malla curricular.	✓	
4	Creación de contenido multimedia relacionado a cada asignatura.	✓	

**MÓDULO: NOTIFICACIONES**

Nº	Funcionalidad	Estado	Observaciones
1	Visualización de notificaciones en tiempo real.	✓	

**MÓDULO: FORO ACADÉMICO**

Nº	Funcionalidad	Estado	Observaciones
1	Creación de foros.	✓	
2	Edición y eliminación de foros.	✓	
3	Gestión de respuestas en foro.	✓	
4	Interacción en tiempo real con otros usuarios, reacciones y respuestas.	○	

*Figura XL: ANEXO 1 - Checklist parte 3 (Fuente: Elaboración propia)*

**MÓDULO: CLASES EN VIVO**

Nº	Funcionalidad	Estado	Observaciones
1	Creación de clases en vivo.	✓	
2	Edición y eliminación de clases en vivo.	✓	

**MÓDULO: AÑO ACADÉMICO**

Nº	Funcionalidad	Estado	Observaciones
1	Creación de año académico.	✓	
2	Edición y eliminación de año académico.	✓	
3	Gestión de estudiantes (matrícula y reingreso).	✓	
4	Seguimiento de la gestión académica.	✓	Poder mostrar notas de manera literal
5	Creación de los horarios académicos.	✓	
6	Armado de horarios.	✓	Añadir gestión de horario de docentes
7	Asignación de docentes	✓	

**MÓDULO: GESTIÓN DE ASISTENCIAS**

Nº	Funcionalidad	Estado	Observaciones
1	Marcaje de asistencia.	✓	Añadir gestión de asistencia de docentes
2	Registro manual e individual de asistencia.	✓	
3	Registro de asistencia masiva	✓	

**MÓDULO: REPORTE**

Nº	Funcionalidad	Estado	Observaciones
1	Reportes administrativos.	✓	
2	Reportes académicos.	✓	

Figura XLI: ANEXO 1 - Checklist parte 4 (Fuente: Elaboración propia)

## PERFIL DOCENTE

### MÓDULO: AUTENTICACIÓN Y SEGURIDAD

Nº	Funcionalidad	Estado	Observaciones
1	Inicio de sesión con credenciales.	✓	
2	Cierre de sesión seguro.	✓	
3	Protección de rutas y vistas.	✓	

### MÓDULO: MANTENIMIENTO

Nº	Funcionalidad	Estado	Observaciones
1	Registro de asignaturas.	✓	
2	Edición y eliminación de asignaturas.	✓	
3	Registro de grados.	✓	
4	Edición y eliminación de grados.	✓	
5	Registro de niveles.	✓	
6	Edición y eliminación de niveles.	✓	

### MÓDULO: MALLAS CURRICULARES

Nº	Funcionalidad	Estado	Observaciones
1	Registro de mallas curriculares.	✓	
2	Edición y eliminación de mallas curriculares.	✓	
3	Asignación de asignaturas a malla curricular.	✓	
4	Creación de contenido multimedia relacionado a cada asignatura.	✓	

### MÓDULO: NOTIFICACIONES

Nº	Funcionalidad	Estado	Observaciones
1	Visualización de notificaciones en tiempo real.	✓	

Figura XLII: ANEXO 1 - Checklist parte 5 (Fuente: Elaboración propia)

**MÓDULO: FORO ACADÉMICO**

Nº	Funcionalidad	Estado	Observaciones
1	Creación de foros.	✓	
2	Edición y eliminación de foros.	✓	
3	Gestión de respuestas en foro.	✓	
4	Interacción en tiempo real con otros usuarios, reacciones y respuestas.	○	

**MÓDULO: CLASES EN VIVO**

Nº	Funcionalidad	Estado	Observaciones
1	Creación de clases en vivo.	✓	
2	Edición y eliminación de clases en vivo.	✓	

**MÓDULO: GESTIÓN DE ASISTENCIAS**

Nº	Funcionalidad	Estado	Observaciones
1	Marcaje de asistencia.	✓	
2	Registro manual e individual de asistencia.	✓	
3	Registro de asistencia masiva	✓	

**MÓDULO: REPORTES**

Nº	Funcionalidad	Estado	Observaciones
1	Reportes administrativos.	✓	
2	Reportes académicos.	✓	

Figura XLIII: ANEXO 1 - Checklist parte 6 (Fuente: Elaboración propia)

**PERFIL ESTUDIANTE**

**MÓDULO: AUTENTICACIÓN Y SEGURIDAD**

Nº	Funcionalidad	Estado	Observaciones
1	Inicio de sesión con credenciales.	✓	
2	Cierre de sesión seguro.	✓	
3	Protección de rutas y vistas.	✓	

**MÓDULO: MI PERFIL**

Nº	Funcionalidad	Estado	Observaciones
1	Facilidad para actualizar datos.	✓	
2	Actualización de contraseña.	✓	

**MÓDULO: MIS CURSOS**

Nº	Funcionalidad	Estado	Observaciones
1	Visualización general de cursos.	✓	
2	Visualización de contenido de cada curso por semana, incluido cuestionarios.	✓	
3	Ver avance de cada curso y notas del mismo.	✓	

**MÓDULO: MIS NOTIFICACIONES**

Nº	Funcionalidad	Estado	Observaciones
1	Visualización de notificaciones en tiempo real.	✓	

*Figura XLIV: ANEXO 1 - Checklist parte 7 (Fuente: Elaboración propia)*

**MÓDULO: FORO ACADÉMICO**

Nº	Funcionalidad	Estado	Observaciones
1	Creación de foros.	✓	
2	Visualización de respuestas en foro.	✓	
3	Interacción en tiempo real con otros usuarios, reacciones y respuestas.	○	

**MÓDULO: CLASES EN VIVO**

Nº	Funcionalidad	Estado	Observaciones
1	Ver lista de clases en vivo.	✓	
2	Acceder a las clases en vivo.	✓	

**MÓDULO: MIS PAGOS**

Nº	Funcionalidad	Estado	Observaciones
1	Visualización de pagos realizados por año académico.	✓	

**MÓDULO: MIS NOTIFICACIONES**

Nº	Funcionalidad	Estado	Observaciones
1	Visualización de calificaciones por año académico.	✓	

**MÓDULO: CHAT AI**

Nº	Funcionalidad	Estado	Observaciones
1	Acceso a chat con un agente de inteligencia artificial para resolver dudas académicas.	✓	



*Norma Tineo Calderón*

Directora Norma Tineo Calderón  
Experto

Figura XLV: ANEXO 1 - Checklist parte 8 (Fuente: Elaboración propia)

## Instrumento de Validación con escala Likert

De manera complementaria, se aplicó un instrumento de validación basado en la escala Likert (revisar Anexo 2), con el objetivo de evaluar la calidad del sistema desde la perspectiva de un experto.

El instrumento estuvo estructurado en función de tres variables, previamente definidas en la investigación, y fue aplicado considerando cinco niveles de valoración explicados al inicio de esta investigación, que permiten medir el grado de conformidad del evaluador respecto al sistema.

### ANEXO 2: INSTRUMENTO DE VALIDACIÓN DE LA PLATAFORMA EDUCATIVA

Objetivo: Validar las funcionalidades de la plataforma educativa de acuerdo a las variables Registro, Control y Seguimiento de actividades escolares, utilizando una escala Likert.

#### Escala Likert

1 = Muy en desacuerdo | 2 = En desacuerdo | 3 = Ni de acuerdo ni en desacuerdo | 4 = De acuerdo | 5 = Muy de acuerdo

#### VARIABLE 1: REGISTRO DE ACTIVIDADES ESCOLARES

Rol	Módulo	Ítem del checklist técnico	Ítem	Escala (1-5)
Administrador	Gestión de usuarios	Registrar estudiantes	El sistema permite registrar correctamente la información de los estudiantes.	5
Administrador	Gestión de usuarios	Registrar docentes	El sistema permite registrar correctamente la información de los docentes.	5
Administrador	Gestión académica	Registrar cursos y asignaturas	El sistema permite registrar cursos y asignaturas sin errores.	5
Docente	Asistencia	Registrar asistencia	El sistema permite registrar la asistencia de los estudiantes de manera eficiente.	4
Docente	Evaluaciones	Registrar calificaciones	El sistema permite registrar calificaciones de forma clara y organizada.	4
Docente	Contenido	Subir materiales educativos	El sistema permite registrar y almacenar materiales educativos digitales.	5
Estudiante	Plataforma	Registro de actividades académicas	El sistema registra correctamente las actividades académicas realizadas por el estudiante.	5

Figura XLVI: ANEXO 2 - Instrumento de validación en escala Likert parte 1 (Fuente: Elaboración propia)

**VARIABLE 2: CONTROL DE ACTIVIDADES ESCOLARES**

Rol	Módulo	Ítem del checklist técnico	Ítem	Escala (1-5)
Administrador	Reportes	Visualizar reportes académicos	El sistema permite controlar la información académica mediante reportes claros.	4
Administrador	Reportes	Exportar reportes	El sistema permite exportar reportes académicos cuando se requiere.	5
Docente	Evaluaciones	Visualizar notas registradas	El sistema permite verificar las calificaciones registradas de los estudiantes.	5
Docente	Asistencia	Consultar asistencias	El sistema permite controlar la asistencia de los estudiantes oportunamente.	5
Estudiante	Académico	Visualizar notas	El sistema permite al estudiante consultar sus calificaciones de forma clara.	5
Estudiante	Académico	Visualizar horarios y cursos	El sistema permite al estudiante consultar sus cursos y horarios asignados.	5

*Figura XLVII: ANEXO 2 - Instrumento de validación en escala Likert parte 2 (Fuente: Elaboración propia)*

**VARIABLE 3: SEGUIMIENTO DE ACTIVIDADES ESCOLARES**

Rol	Módulo	Ítem del checklist técnico	Ítem	Escala (1-5)
Docente	Seguimiento	Visualizar progreso académico	El sistema permite realizar seguimiento al progreso académico de los estudiantes.	5
Docente	Comunicación	Enviar mensajes y avisos	El sistema facilita la retroalimentación entre docente y estudiante.	4
Estudiante	Seguimiento	Visualizar avance académico	El sistema permite al estudiante conocer su avance académico.	5
Estudiante	Comunicación	Recibir notificaciones	El sistema notifica oportunamente sobre actividades y evaluaciones.	4
Estudiante	Chatbot IA	Consultar dudas académicas	El sistema permite resolver dudas académicas mediante el chatbot con IA.	4



Directora Norma Tineo Calderón  
Experto

*Figura XLVIII: ANEXO 2 - Instrumento de validación en escala Likert parte 3 (Fuente: Elaboración propia)*

### 4.1.3. Matriz de validación del sistema

Una vez aplicada la evaluación, los resultados obtenidos fueron organizados en una matriz de validación, la cual permitió consolidar de manera ordenada las respuestas del experto (Anexo 3). Esta matriz facilita el análisis sistemático de los datos recolectados y permite identificar el nivel de aceptación y conformidad del sistema académico desarrollado.

**VARIABLE 1: REGISTRO DE ACTIVIDADES ESCOLARES**

Variable	Dimensión	Indicador	Ítem	Escala	Puntaje	Nivel
<b>Registro de Actividades Escolares</b>	Gestión de usuarios	Registrar estudiantes	El sistema permite registrar correctamente la información de los estudiantes.	1 – 5	5	Alto
	Gestión de usuarios	Registrar docentes	El sistema permite registrar correctamente la información de los docentes.	1 – 5	5	Alto
	Gestión académica	Registrar cursos y asignaturas	El sistema permite registrar cursos y asignaturas sin errores.	1 – 5	5	Alto
	Asistencia	Registrar asistencia	El sistema permite registrar la asistencia de los estudiantes de manera eficiente.	1 – 5	4	Medio
	Evaluaciones	Registrar calificaciones	El sistema permite registrar calificaciones de forma clara y organizada.	1 – 5	4	Medio
	Contenido	Subir materiales educativos	El sistema permite registrar y almacenar materiales educativos digitales.	1 – 5	5	Alto
	Plataforma	Registro de actividades académicas	El sistema registra correctamente las actividades académicas realizadas por el estudiante.	1 – 5	5	Alto

*Tabla LXXIX: Matriz: Registro de Actividades Escolares (Fuente: Elaboración propia)*

**VARIABLE 2: CONTROL DE ACTIVIDADES ESCOLARES**

Variable	Dimensión	Indicador	Ítem	Escala	Puntaje	Nivel
<b>Control de Actividades Escolares</b>	Reportes	Visualizar reportes académicos	El sistema permite controlar la información académica mediante reportes claros.	1 – 5	4	Medio
	Reportes	Exportar reportes	El sistema permite exportar reportes académicos cuando se requiere.	1 – 5	5	Alto
	Evaluaciones	Visualizar notas registradas	El sistema permite verificar las calificaciones registradas de los estudiantes.	1 – 5	5	Alto
	Asistencia	Consultar asistencias	El sistema permite controlar la asistencia de los estudiantes oportunamente.	1 – 5	5	Alto
	Académico	Visualizar notas	El sistema permite al estudiante consultar sus calificaciones de forma clara.	1 – 5	5	Alto
	Académico	Visualizar horarios y cursos	El sistema permite al estudiante consultar sus cursos y horarios asignados.	1 – 5	5	Alto

*Tabla LXXX: Matriz: Control de Actividades Escolares (Fuente: Elaboración propia)*

**VARIABLE 3: SEGUIMIENTO DE ACTIVIDADES ESCOLARES**

Variable	Dimensión	Indicador	Ítem	Escala	Puntaje	Nivel
Seguimiento de Actividades Escolares	Seguimiento	Visualizar progreso académico	El sistema permite realizar seguimiento al progreso académico de los estudiantes.	1 – 5	5	Alto
	Comunicación	Enviar mensajes y avisos	El sistema facilita la retroalimentación entre docente y estudiante.	1 – 5	4	Medio
	Seguimiento	Visualizar avance académico	El sistema permite al estudiante conocer su avance académico.	1 – 5	5	Alto
	Comunicación	Recibir notificaciones	El sistema notifica oportunamente sobre actividades y evaluaciones.	1 – 5	4	Medio
	Chatbot IA	Consultar dudas académicas	El sistema permite resolver dudas académicas mediante el chatbot con IA.	1 – 5	4	Medio

*Tabla LXXXI: Matriz: Seguimiento de Actividades Escolares (Fuente: Elaboración propia)*

## 4.2. Resultados de la validación

Variable	Puntaje promedio	Nivel de validación
Registro de actividades escolares	4.7	Alto
Control de actividades escolares	4.8	Alto
Seguimiento de actividades escolares	4.4	Alto
<b>Promedio general</b>	<b>4.63</b>	<b>Alto</b>

*Tabla LXXXII: Tabla de resultados de la validación (Fuente: Elaboración propia)*

En el presente apartado se exponen los resultados obtenidos a partir de la aplicación del instrumento de validación con escala Likert, cuyos datos fueron organizados y consolidados mediante la matriz de validación, permitiendo evaluar el grado de cumplimiento y aceptación del sistema académico desarrollado para la Institución Educativa Particular San Isidro.

La validación se realizó considerando tres variables fundamentales del sistema: Registro de actividades escolares, Control de actividades escolares y Seguimiento de actividades escolares. Cada una de estas variables fue evaluada por el experto designado, utilizando una escala Likert de cinco niveles, donde 1 representa “Muy en desacuerdo” y 5 representa “Muy de acuerdo”.

Los resultados obtenidos se resumen en la *Tabla LXXXII*, donde se observa que la variable *Registro de actividades escolares* alcanzó un puntaje promedio de 4.7, lo que corresponde a un nivel de validación alto. Este resultado evidencia que el sistema permite registrar y organizar la información académica de manera precisa, accesible y confiable, reduciendo errores y mejorando la disponibilidad de los datos.

Asimismo, la variable *Control de actividades escolares* obtuvo un puntaje promedio de 4.8, siendo la más alta entre las variables evaluadas, y clasificándose también en un nivel de validación alto. Este resultado demuestra que la plataforma facilita de forma eficiente la gestión de calificaciones, asistencia y reportes académicos, contribuyendo a un mejor control de las actividades escolares por parte de la institución.

Por su parte, la variable *Seguimiento de actividades escolares* alcanzó un puntaje promedio de 4.4, lo que igualmente corresponde a un nivel de validación alto. Esto indica que el sistema permite realizar un monitoreo adecuado del desempeño académico de los estudiantes, así como generar alertas y brindar retroalimentación oportuna para apoyar el proceso educativo.

Finalmente, el promedio general de validación del sistema fue de 4.63, ubicándose en un nivel alto, lo que confirma que la plataforma virtual desarrollada cumple satisfactoriamente con los requerimientos funcionales y académicos establecidos en el proyecto. Estos resultados evidencian que el sistema es técnicamente adecuado, funcional y pertinente para mejorar la gestión académica en la Institución Educativa Particular San Isidro.

## 5. DISCUSIÓN DE RESULTADOS

Los resultados obtenidos en la presente investigación evidencian que la plataforma virtual desarrollada contribuye de manera significativa a la mejora de la gestión académica en la Institución Educativa Particular San Isidro. A continuación, se discuten los resultados en función de los objetivos planteados, contrastándolos con los antecedentes y aportes teóricos considerados.

En relación con el registro de actividades escolares, los resultados muestran un nivel de validación alto, con un puntaje promedio de 4.7, lo que indica que la plataforma permite una gestión más precisa, rápida y accesible de la información académica. Estos hallazgos coinciden con lo señalado por Trejo Alvarado (2021), quien concluye que la digitalización de los registros académicos reduce errores, optimiza el almacenamiento de información y mejora la disponibilidad de los datos para los actores educativos. Asimismo, Torres Argomedo (2023) resalta que el uso de plataformas virtuales fortalece la trazabilidad y confiabilidad de los registros escolares, aspectos que también se evidencian en el presente estudio.

Respecto al control de las actividades escolares, los resultados reflejan el mayor puntaje promedio 4.8, evidenciando que el sistema facilita eficazmente la gestión de calificaciones, asistencia y generación de reportes académicos. Estos resultados son coherentes con los hallazgos de Maza Ancajima (2022), quien afirma que los sistemas web de gestión académica permiten centralizar la información y fortalecer los procesos de supervisión y control institucional. De igual manera, Trejo Alvarado (2021) señala que el acceso oportuno a reportes académicos favorece la toma de decisiones por parte de las autoridades educativas, lo cual se ve reflejado en el funcionamiento del sistema validado.

En cuanto al seguimiento de las actividades escolares, se obtuvo un puntaje promedio de 4.4, correspondiente a un nivel de validación alto, lo que demuestra que la plataforma posibilita el monitoreo continuo del desempeño académico de los estudiantes y la generación de alertas y reportes de progreso. Estos resultados guardan concordancia con lo expuesto por Torres Argomedo (2023), quien sostiene que el seguimiento académico sistematizado permite identificar oportunamente dificultades de aprendizaje y fortalece la retroalimentación entre docentes y estudiantes. Asimismo,

Maza Ancajima (2022) destaca que el seguimiento permanente del rendimiento académico contribuye a mejorar la calidad del proceso educativo, aspecto evidenciado en la presente investigación.

Finalmente, el promedio general de validación 4.63 confirma que la plataforma virtual cumple con los criterios funcionales y operativos definidos, respaldando su utilidad como herramienta tecnológica para la gestión académica. En este sentido, los resultados obtenidos coinciden con las conclusiones generales de los estudios revisados, los cuales señalan que la implementación de sistemas de gestión académica basados en plataformas digitales genera mejoras sustanciales en la organización, control y seguimiento de los procesos educativos (Trejo Alvarado, 2021; Maza Ancajima, 2022; Torres Argomedo, 2023).

## **6. CONCLUSIONES**

Se concluye que el desarrollo de la plataforma virtual cumple con el objetivo general del proyecto, al constituirse como una solución efectiva para mejorar la gestión académica en la Institución Educativa Particular San Isidro. Este resultado coincide con lo reportado en investigaciones previas sobre sistemas de gestión académica, donde se evidencia que la digitalización de los procesos académicos contribuye a una administración institucional más eficiente y organizada.

El sistema implementado permitió fortalecer el proceso de registro de actividades escolares, garantizando una gestión más confiable, ordenada y accesible de la información académica. Este hallazgo es consistente con otros estudios similares, los cuales concluyen que el uso de plataformas virtuales reduce errores en el manejo de datos y mejora la disponibilidad de la información para la toma de decisiones.

La plataforma desarrollada demostró ser una herramienta eficiente para el control de las actividades escolares, facilitando la gestión de calificaciones, asistencia y reportes académicos. En concordancia con otras tesis revisadas, se evidencia que los sistemas académicos digitales optimizan los procesos de control y seguimiento, fortaleciendo el

rol de los docentes y directivos en la supervisión del desempeño académico.

Se concluye que el sistema favorece el seguimiento continuo del desempeño académico de los estudiantes, permitiendo un monitoreo oportuno que mejora la interacción entre docentes y estudiantes. A diferencia de algunos trabajos previos que se enfocan únicamente en el registro de información, el presente proyecto integra herramientas de seguimiento que aportan un mayor valor al proceso educativo.

## **7. RECOMENDACIONES**

Se recomienda extender la plataforma virtual mediante el desarrollo de un aplicativo móvil, recomendación que guarda relación con estudios previos que destacan la importancia de la accesibilidad móvil para mejorar la participación y el uso continuo de los sistemas académicos.

Se sugiere adaptar y parametrizar el sistema para su posible implementación en otras instituciones educativas, considerando diferentes modelos de gestión académica y esquemas de evaluación. Esta recomendación coincide con otros trabajos que proponen la escalabilidad de los sistemas como un factor clave para su sostenibilidad.

Se recomienda ampliar y mejorar progresivamente las funcionalidades del sistema, incorporando nuevos módulos de acuerdo con las necesidades institucionales, lo cual concuerda con las recomendaciones planteadas en investigaciones similares orientadas a la mejora continua de los sistemas de información académica.

Entre una de las funcionalidades, se recomienda el desarrollo del módulo exclusivo para padres de familia, donde ellos podrán tener toda la información disponible correspondiente a sus hijos o apoderados.

Se propone, como trabajo futuro, integrar funcionalidades avanzadas como analítica educativa o asistentes virtuales, ampliando lo sugerido en otros estudios que enfatizan la

necesidad de incorporar tecnologías emergentes para fortalecer la toma de decisiones pedagógicas.

Finalmente, se recomienda evaluar la viabilidad de una futura integración con el sistema estatal SIAGIE, lo cual complementa las recomendaciones de otras investigaciones que resaltan la importancia de la interoperabilidad con plataformas oficiales para garantizar coherencia y validez de la información académica.

## 8. REFERENCIAS

Ayala Pari, E. R. (2022). Diseño de un sistema web para registro académico y administrativo para la institución educativa pública Señor de los Milagros Ranchaj en el año 2018 [Tesis de pregrado, Universidad Católica Los Ángeles de Chimbote]. Repositorio ULADECH. <https://hdl.handle.net/20.500.13032/26644>

Damasio Jara, C. A. (2021). Implementación de una plataforma web para el control académico de los alumnos de secundaria en el colegio 6069 Pachacútec de Villa el Salvador en la ciudad de Lima 2019 [Tesis de pregrado, Universidad Tecnológica del Perú]. Repositorio UTP. <https://hdl.handle.net/20.500.12867/5097>

Calderón Bailón, R. H. (2021). Implementación de un sistema de información para la mejora en el seguimiento académico de la Institución Educativa Particular Ciencias Siglo XXI – Puente Piedra [Tesis de pregrado, Universidad César Vallejo]. Repositorio UCV. <https://hdl.handle.net/20.500.12692/83113>

Maza Ancajima, C. M. A. (2023). Implementación de un sistema web de gestión académica para el Instituto Superior Tecnológico Privado San Martín de Porres, Tambogrande – Piura; 2022 [Tesis de licenciatura, Universidad Católica Los Ángeles de Chimbote]. Universidad Católica Los Ángeles de Chimbote. Repositorio ULADECH. <https://hdl.handle.net/20.500.13032/32440>

Trejo Alvarado, J. A. (2020). Entorno virtual de aprendizaje en la gestión académica en el Colegio “Los Andes”- Huaraz, 2019 [Tesis de maestría, Universidad Peruana de Ciencias e Informática]. Universidad Peruana de Ciencias e Informática. Repositorio UPCI <http://repositorio.upci.edu.pe/handle/upci/166>

Torres Argomedo, L. (2019). Plataforma virtual para mejorar el rendimiento en una asignatura del plan curricular de la Escuela de Tecnologías de la Información, SENATI [Tesis de maestría, Universidad Peruana Cayetano Heredia]. Repositorio Institucional UNCH. <https://hdl.handle.net/20.500.12866/7726>

Gonzales Mijahuanca, J. L., & Peña Alama, Y. D. P. (2022). Sistema multiplataforma para automatizar el proceso administrativo de la Institución Educativa Privada “Sócrates” Piura - 2021 [Tesis de licenciatura, Universidad César Vallejo]. Universidad César Vallejo. <https://hdl.handle.net/20.500.12692/99429>

## 9. ANEXOS

### ANEXO 1: CHECKLIST TÉCNICO DE VERIFICACIÓN DEL BACKLOG LA PLATAFORMA EDUCATIVA

El presente checklist tiene como finalidad verificar el cumplimiento de las tareas definidas en el Product Backlog de la plataforma educativa, permitiendo constatar si cada funcionalidad fue implementada correctamente. Este instrumento será utilizado como evidencia de funcionamiento del sistema.

#### Escala de verificación

✓ = Implementado | ✗ = No implementado | O = Implementado parcialmente

#### PERFIL ADMINISTRATIVO

##### MÓDULO: AUTENTICACIÓN Y SEGURIDAD

N°	Funcionalidad	Estado	Observaciones
1	Inicio de sesión con credenciales.		
2	Cierre de sesión seguro.		
3	Control de acceso por roles.		
4	Protección de rutas y vistas.		

##### MÓDULO: USUARIOS Y PERMISOS

N°	Funcionalidad	Estado	Observaciones
1	Registro de permisos.		
2	Edición y eliminación de permisos.		
3	Registro de tipo de usuarios.		
4	Edición y eliminación de tipos usuarios.		

5	Gestión de permisos por tipo de usuario.		
6	Registro usuarios.		
7	Edición y eliminación de usuarios.		

#### MÓDULO: MANTENIMIENTO

N°	Funcionalidad	Estado	Observaciones
1	Registro de asignaturas.		
2	Edición y eliminación de asignaturas.		
3	Registro de documentos.		
4	Edición y eliminación de documentos.		
5	Registro de grados.		
6	Edición y eliminación de grados.		
7	Registro de niveles.		
8	Edición y eliminación de niveles.		
9	Registro de categoría de eventos.		
10	Edición y eliminación de categorías de eventos.		
11	Registro de eventos.		
12	Edición y eliminación de eventos.		

#### MÓDULO: DASHBOARD ADMINISTRATIVO

N°	Funcionalidad	Estado	Observaciones
1	Filtros por años académicos.		
2	Alertas de asistencia de estudiantes.		
3	Información de asistencias.		

4	Gestión de ingresos, cobros pendientes, montos cobrados, etc.		
---	---	--	--

#### MÓDULO: CATEGORÍAS DE PAGO

N°	Funcionalidad	Estado	Observaciones
1	Registro de categorías de pago.		
2	Edición y eliminación de categorías de pago.		
3	Creación y asignación de conceptos de pago.		
4	Edición y eliminación de conceptos de pago.		

#### MÓDULO: MALLAS CURRICULARES

N°	Funcionalidad	Estado	Observaciones
1	Registro de mallas curriculares.		
2	Edición y eliminación de mallas curriculares.		
3	Asignación de asignaturas a la malla curricular.		
4	Creación de contenido multimedia relacionado a cada asignatura.		

#### MÓDULO: NOTIFICACIONES

N°	Funcionalidad	Estado	Observaciones
1	Visualización de notificaciones en tiempo real.		

#### MÓDULO: FORO ACADÉMICO

N°	Funcionalidad	Estado	Observaciones
1	Creación de foros.		
2	Edición y eliminación de foros.		
3	Gestión de respuestas en foro.		
4	Interacción en tiempo real con otros usuarios, reacciones y respuestas.		

#### MÓDULO: CLASES EN VIVO

N°	Funcionalidad	Estado	Observaciones
1	Creación de clases en vivo.		
2	Edición y eliminación de clases en vivo.		

#### MÓDULO: AÑO ACADÉMICO

N°	Funcionalidad	Estado	Observaciones
1	Creación de año académico.		
2	Edición y eliminación de año académico.		
3	Gestión de estudiantes (matrícula y reingreso).		
4	Seguimiento de la gestión académica.		
5	Creación de los horarios académicos.		
6	Armado de horarios.		
7	Asignación de docentes		

#### MÓDULO: GESTIÓN DE ASISTENCIAS

N°	Funcionalidad	Estado	Observaciones
1	Marcaje de asistencia.		
2	Registro manual e individual de asistencia.		

3	Registro de asistencia masiva		
---	-------------------------------	--	--

### MÓDULO: REPORTE

N°	Funcionalidad	Estado	Observaciones
1	Reportes administrativos.		
2	Reportes académicos.		

### PERFIL DOCENTE

#### MÓDULO: AUTENTICACIÓN Y SEGURIDAD

N°	Funcionalidad	Estado	Observaciones
1	Inicio de sesión con credenciales.		
2	Cierre de sesión seguro.		
3	Protección de rutas y vistas.		

#### MÓDULO: MANTENIMIENTO

N°	Funcionalidad	Estado	Observaciones
1	Registro de asignaturas.		
2	Edición y eliminación de asignaturas.		
3	Registro de grados.		
4	Edición y eliminación de grados.		
5	Registro de niveles.		
6	Edición y eliminación de niveles.		

#### MÓDULO: MALLAS CURRICULARES

N°	Funcionalidad	Estado	Observaciones
1	Registro de mallas curriculares.		
2	Edición y eliminación de mallas curriculares.		

3	Asignación de asignaturas a malla curricular.		
4	Creación de contenido multimedia relacionado a cada asignatura.		

#### MÓDULO: NOTIFICACIONES

N°	Funcionalidad	Estado	Observaciones
1	Visualización de notificaciones en tiempo real.		

#### MÓDULO: FORO ACADÉMICO

N°	Funcionalidad	Estado	Observaciones
1	Creación de foros.		
2	Edición y eliminación de foros.		
3	Gestión de respuestas en foro.		
4	Interacción en tiempo real con otros usuarios, reacciones y respuestas.		

#### MÓDULO: CLASES EN VIVO

N°	Funcionalidad	Estado	Observaciones
1	Creación de clases en vivo.		
2	Edición y eliminación de clases en vivo.		

#### MÓDULO: GESTIÓN DE ASISTENCIAS

N°	Funcionalidad	Estado	Observaciones
1	Marcaje de asistencia.		
2	Registro manual e individual de asistencia.		
3	Registro de asistencia masiva		

#### MÓDULO: REPORTES

<b>N°</b>	<b>Funcionalidad</b>	<b>Estado</b>	<b>Observaciones</b>
<b>1</b>	<b>Reportes administrativos.</b>		
<b>2</b>	<b>Reportes académicos.</b>		

## PERFIL ESTUDIANTE

### MÓDULO: AUTENTICACIÓN Y SEGURIDAD

Nº	Funcionalidad	Estado	Observaciones
1	Inicio de sesión con credenciales.		
2	Cierre de sesión seguro.		
3	Protección de rutas y vistas.		

### MÓDULO: MI PERFIL

Nº	Funcionalidad	Estado	Observaciones
1	Facilidad para actualizar datos.		
2	Actualización de contraseña.		

### MÓDULO: MIS CURSOS

Nº	Funcionalidad	Estado	Observaciones
1	Visualización general de cursos.		
2	Visualización de contenido de cada curso por semana, incluido cuestionarios.		
3	Ver avance de cada curso y notas del mismo.		

### MÓDULO: MIS NOTIFICACIONES

Nº	Funcionalidad	Estado	Observaciones
1	Visualización de notificaciones en tiempo real.		

**MÓDULO: FORO ACADÉMICO**

N°	Funcionalidad	Estado	Observaciones
1	Creación de foros.		
2	Visualización de respuestas en foro.		
3	Interacción en tiempo real con otros usuarios, reacciones y respuestas.		

**MÓDULO: CLASES EN VIVO**

N°	Funcionalidad	Estado	Observaciones
1	Ver lista de clases en vivo.		
2	Acceder a las clases en vivo.		

**MÓDULO: MIS PAGOS**

N°	Funcionalidad	Estado	Observaciones
1	Visualización de pagos realizados por año académico.		

**MÓDULO: MIS NOTIFICACIONES**

N°	Funcionalidad	Estado	Observaciones
1	Visualización de calificaciones por año académico.		

**MÓDULO: CHAT AI**

N°	Funcionalidad	Estado	Observaciones
1	Acceso a chat con un agente de inteligencia artificial para resolver dudas académicas.		

---

Directora Norma Tineo Calderón  
Exp

## ANEXO 2: INSTRUMENTO DE VALIDACIÓN DE LA PLATAFORMA EDUCATIVA

Objetivo: Validar las funcionalidades de la plataforma educativa de acuerdo a las variables Registro, Control y Seguimiento de actividades escolares, utilizando una escala Likert.

### Escala Likert

1 = Muy en desacuerdo | 2 = En desacuerdo | 3 = Ni de acuerdo ni en desacuerdo | 4 = De acuerdo | 5 = Muy de acuerdo

### VARIABLE 1: REGISTRO DE ACTIVIDADES ESCOLARES

Rol	Módulo	Ítem del checklist técnico	Ítem	Escala (1-5)
Administrador	Gestión de usuarios	Registrar estudiantes	El sistema permite registrar correctamente la información de los estudiantes.	
Administrador	Gestión de usuarios	Registrar docentes	El sistema permite registrar correctamente la información de los docentes.	
Administrador	Gestión académica	Registrar cursos y asignaturas	El sistema permite registrar cursos y asignaturas sin errores.	
Docente	Asistencia	Registrar asistencia	El sistema permite registrar la asistencia de los estudiantes de manera eficiente.	
Docente	Evaluaciones	Registrar calificaciones	El sistema permite registrar calificaciones de forma clara y organizada.	
Docente	Contenido	Subir materiales educativos	El sistema permite registrar y almacenar materiales educativos digitales.	
Estudiante	Plataforma	Registro de actividades académicas	El sistema registra correctamente las actividades académicas realizadas por el estudiante.	

**VARIABLE 2: CONTROL DE ACTIVIDADES ESCOLARES**

<b>Rol</b>	<b>Módulo</b>	<b>Ítem del checklist técnico</b>	<b>Ítem</b>	<b>Escala (1-5)</b>
Administrador	Reportes	Visualizar reportes académicos	El sistema permite controlar la información académica mediante reportes claros.	
Administrador	Reportes	Exportar reportes	El sistema permite exportar reportes académicos cuando se requiere.	
Docente	Evaluaciones	Visualizar notas registradas	El sistema permite verificar las calificaciones registradas de los estudiantes.	
Docente	Asistencia	Consultar asistencias	El sistema permite controlar la asistencia de los estudiantes oportunamente.	
Estudiante	Académico	Visualizar notas	El sistema permite al estudiante consultar sus calificaciones de forma clara.	
Estudiante	Académico	Visualizar horarios y cursos	El sistema permite al estudiante consultar sus cursos y horarios asignados.	

**VARIABLE 3: SEGUIMIENTO DE ACTIVIDADES ESCOLARES**

<b>Rol</b>	<b>Módulo</b>	<b>Ítem del checklist técnico</b>	<b>Ítem</b>	<b>Escala (1-5)</b>
Docente	Seguimiento	Visualizar progreso académico	El sistema permite realizar seguimiento al progreso académico de los estudiantes.	
Docente	Comunicación	Enviar mensajes y avisos	El sistema facilita la retroalimentación entre docente y estudiante.	
Estudiante	Seguimiento	Visualizar avance académico	El sistema permite al estudiante conocer su avance académico.	
Estudiante	Comunicación	Recibir notificaciones	El sistema notifica oportunamente sobre actividades y evaluaciones.	
Estudiante	Chatbot IA	Consultar dudas académicas	El sistema permite resolver dudas académicas mediante el chatbot con IA.	

---

Directora Norma Tineo Calderón  
Experto

### ANEXO 3: MATRIZ DE VALIDACIÓN DE LA PLATAFORMA EDUCATIVA

Objetivo: Relacionar las variables – dimensiones – indicadores – ítems – resultados, para demostrar que el sistema cumple con lo propuesto.

#### VARIABLE 1: REGISTRO DE ACTIVIDADES ESCOLARES

Variable	Dimensión	Indicador	Ítem	Escala	Puntaje	Nivel
<b>Registro de Actividades Escolares</b>	Gestión de usuarios	Registrar estudiantes	El sistema permite registrar correctamente la información de los estudiantes.	1 – 5		
	Gestión de usuarios	Registrar docentes	El sistema permite registrar correctamente la información de los docentes.	1 – 5		
	Gestión académica	Registrar cursos y asignaturas	El sistema permite registrar cursos y asignaturas sin errores.	1 – 5		
	Asistencia	Registrar asistencia	El sistema permite registrar la asistencia de los estudiantes de manera eficiente.	1 – 5		
	Evaluaciones	Registrar calificaciones	El sistema permite registrar calificaciones de forma clara y organizada.	1 – 5		
	Contenido	Subir materiales educativos	El sistema permite registrar y almacenar materiales educativos digitales.	1 – 5		
	Plataforma	Registro de actividades académicas	El sistema registra correctamente las actividades académicas realizadas por el estudiante.	1 – 5		

## VARIABLE 2: CONTROL DE ACTIVIDADES ESCOLARES

Variable	Dimensión	Indicador	Ítem	Escala	Puntaje	Nivel
<b>Control de Actividades Escolares</b>	Reportes	Visualizar reportes académicos	El sistema permite controlar la información académica mediante reportes claros.	1 – 5		
	Reportes	Exportar reportes	El sistema permite exportar reportes académicos cuando se requiere.	1 – 5		
	Evaluaciones	Visualizar notas registradas	El sistema permite verificar las calificaciones registradas de los estudiantes.	1 – 5		
	Asistencia	Consultar asistencias	El sistema permite controlar la asistencia de los estudiantes oportunamente.	1 – 5		
	Académico	Visualizar notas	El sistema permite al estudiante consultar sus calificaciones de forma clara.	1 – 5		
	Académico	Visualizar horarios y cursos	El sistema permite al estudiante consultar sus cursos y horarios asignados.	1 – 5		

**VARIABLE 3: SEGUIMIENTO DE ACTIVIDADES ESCOLARES**

Variable	Dimensión	Indicador	Ítem	Escala	Puntaje	Nivel
<b>Seguimiento de Actividades Escolares</b>	Seguimiento	Visualizar progreso académico	El sistema permite realizar seguimiento al progreso académico de los estudiantes.	1 – 5		
	Comunicación	Enviar mensajes y avisos	El sistema facilita la retroalimentación entre docente y estudiante.	1 – 5		
	Seguimiento	Visualizar avance académico	El sistema permite al estudiante conocer su avance académico.	1 – 5		
	Comunicación	Recibir notificaciones	El sistema notifica oportunamente sobre actividades y evaluaciones.	1 – 5		
	Chatbot IA	Consultar dudas académicas	El sistema permite resolver dudas académicas mediante el chatbot con IA.	1 – 5		