

**UNIVERSIDAD NACIONAL DE SAN CRISTÓBAL DE  
HUAMANGA**

**FACULTAD DE INGENIERÍA DE MINAS, GEOLOGÍA Y CIVIL**

**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



**TESIS:**

**Aplicación web para la gestión de matrícula de estudiantes en la  
Institución Educativa Nuestra Señora de Guadalupe, 2025**

Para optar el título profesional de:  
**INGENIERO DE SISTEMAS**

PRESENTADO POR:  
**Bach. Victor Alberto LOPEZ AMIQUERO**

ASESOR:  
**Mg. Richard ZAPATA CASAVARDE**

**AYACUCHO - PERÚ**

**2025**

## DEDICATORIA

Con infinita gratitud dedico este trabajo a mi madre, por esforzarse en todo momento para darme lo mejor, por ser mi apoyo y fuerza inquebrantable. A mi abuelo Ercilio, quien con su ejemplo, sabiduría y cariño se convirtió en mi maestro y ejemplo. Mi abuela Segundina que desde el cielo me acompaña e ilumina, sé que desde allá arriba te sientes orgullosa y satisfecha al verme convertido en el profesional que siempre quisiste. A mis tías Karina y Beatriz, porque mis primeros aprendizajes fueron gracias a ustedes, agradezco su apoyo constante y consejos con la que me guiaron en cada etapa de mi vida. A mi hermana Anghely, por ser mi fuente de inspiración, alegría y motivo de superación cada día.

Este logro también es suyo, porque sin ustedes, nada de esto habría sido posible.

## AGRADECIMIENTOS

Toda mi gratitud a mi familia por ser el pilar fundamental en cada etapa de mi vida.

Agradecido con mi universidad, mi alma máter, por haberme brindado la oportunidad de formarme profesionalmente y por ser el espacio donde adquirí los conocimientos, valores y experiencias que han contribuido a mi crecimiento personal, académico y profesional.

Agradecer a los docentes de la Escuela Profesional de Ingeniería de Sistemas, por su dedicación, esfuerzo y compromiso en la formación de profesionales competentes, y por haber sido parte esencial de mi desarrollo profesional.

Al Mg. Ing. Richard Zapata Casaverde, mi asesor de tesis, por su orientación y apoyo durante el desarrollo de este trabajo de investigación.

Agradecer al director de la Institución Educativa Nuestra Señora de Guadalupe, Lic. Lucho Alfredo Yupanqui Ataucusi, por su amabilidad y disposición al permitirme realizar esta investigación en su institución, contribuyendo de manera significativa al logro de los objetivos planteados.

## RESUMEN

En la actualidad, muchas instituciones educativas enfrentan dificultades en la gestión de sus procesos administrativos, especialmente en lo relacionado con la matrícula de estudiantes. Tal es el caso de la Institución Educativa “Nuestra Señora de Guadalupe”, donde este proceso se realiza de manera manual mediante hojas de cálculo, generando pérdida de información, duplicidad de registros, errores en el control de pagos y un uso ineficiente del tiempo del personal administrativo.

Por ello, la presente investigación tiene como objetivo determinar en qué medida el uso de la aplicación web desarrollado mediante el marco de trabajo Scrum, mejorará la gestión de matrícula de estudiantes, con el desarrollo de un prototipo funcional que sirva como base para una futura implementación.

La investigación fue de tipo observacional, retrospectiva, transversal y descriptiva, con enfoque cuantitativo y diseño no experimental. Para el desarrollo se aplicó el marco de trabajo Scrum, empleando como instrumento principal de recolección de datos la ficha de registro.

Como resultado, se desarrolló una aplicación web mediante el marco de trabajo Scrum, la validación realizada con el usuario final evidenció un 100% de cumplimiento funcional, confirmando que la aplicación responde adecuadamente a los requerimientos establecidos. Asimismo, se estimaron los resultados ante una futura implementación, mejorando significativamente, concluyendo que, con el uso de la aplicación los procesos de: gestión académica, gestión de información del estudiante y control de pago de pensiones, serán un 93.33%, 71.43% y 99.05% más rápidos respectivamente.

**Palabras clave:** Aplicación web, gestión de matrícula, PHP, Laravel, arquitectura monolítica, automatización, MVC, Scrum, procesos administrativos, institución educativa.

## ABSTRACT

Currently, many educational institutions face difficulties in managing their administrative processes, especially those related to student enrollment. This is the case of the “Nuestra Señora de Guadalupe” Educational Institution, where enrollment management is carried out manually using spreadsheets, leading to information loss, record duplication, payment control errors, and inefficient use of administrative staff time.

Therefore, this research aims to design a web application for enrollment management to optimize and automate the institution’s administrative processes, presenting a functional prototype as a foundation for future implementation.

The research was observational, retrospective, cross-sectional, and descriptive, with a quantitative approach and a non-experimental design. The Scrum framework was applied during development, and a data registration form was used as the main data collection instrument.

As a result, a web application was developed using Scrum framework. Validation carried out with the end user showed 100% functional compliance, confirming that the application adequately meets the established requirements. Furthermore, estimates for a future implementation indicate a significant improvement, concluding that the processes of academic management, student information management, and tuition payment control would be 93.33%, 71.43%, and 99.05% faster, respectively.

**Keywords:** Web application, enrollment management, PHP, Laravel, monolithic architecture, automation, MVC, Scrum, administrative processes, educational institution.

## Contenido

DEDICATORIA.....	ii
AGRADECIMIENTOS.....	iii
RESUMEN.....	iv
ABSTRACT.....	v
CAPÍTULO I PLANTEAMIENTO DE LA INVESTIGACIÓN.....	1
1.1. Planteamiento del problema.....	1
1.2. Formulación del problema.....	2
1.1.1. Problema general.....	2
1.1.2. Problemas específicos.....	2
1.3. Justificación y delimitación de la investigación.....	2
1.1.3. Importancia.....	2
1.1.4. Justificación.....	3
1.1.5. Delimitación de la investigación.....	3
1.4. Objetivos.....	3
1.1.6. Objetivo general.....	3
1.1.7. Objetivos específicos.....	3
CAPÍTULO II MARCO TEÓRICO.....	5
1.5. Antecedentes de la investigación.....	5
2.1.1. A nivel internacional.....	5
2.1.2. A nivel nacional.....	7
1.6. Marco teórico.....	9
2.1.3. La institución.....	9
2.1.4. Aplicación Web.....	11
2.1.5. Matrícula.....	11
2.1.6. Gestión Académica.....	11
2.1.7. Gestión de Información.....	12
2.1.8. Política y Gestión de Cobros.....	12
2.1.9. Proceso de Desarrollo del Software.....	12
2.1.10. Scrum.....	14
2.1.11. Modelo Vista Controlador (MVC).....	21
2.1.12. Programación Orientada a Objetos.....	22
2.1.13. Arquitectura Monolítica.....	25

2.1.14.	PHP.....	25
2.1.15.	JavaScript.....	26
2.1.16.	Servidor Apache.....	26
2.1.17.	Laravel.....	26
2.1.18.	Tailwind CSS.....	28
2.1.19.	Base de Datos.....	28
2.1.20.	Sistema de administración de bases de Datos (DBMS).....	29
2.1.21.	Computación en la Nube.....	30
2.1.22.	Git.....	30
2.1.23.	Azure DevOps.....	30
2.1.24.	Visual Studio Code.....	31
2.1.25.	Pruebas unitarias.....	31
CAPÍTULO III MATERIALES Y MÉTODOS.....		32
1.7.	Tipo y nivel de investigación.....	32
3.1.1.	Tipo de investigación.....	32
3.1.2.	Nivel de investigación.....	32
1.8.	Diseño de investigación.....	33
1.9.	Variables.....	33
3.1.3.	Definición conceptual de las variables.....	33
3.1.4.	Operacionalización de las variables.....	34
1.10.	Población y muestra.....	34
3.1.5.	Población.....	34
3.1.6.	Muestra.....	35
1.11.	Técnicas e instrumentos de investigación.....	35
3.1.7.	Técnicas.....	35
3.1.8.	Instrumentos.....	35
1.12.	Procedimientos.....	36
3.1.9.	Técnicas de Procesamientos de datos.....	36
3.1.10.	Herramientas para el tratamiento de datos.....	36
3.1.11.	Técnicas de análisis de datos.....	37
3.1.12.	Técnicas para aplicar el marco de trabajo Scrum.....	37
CAPÍTULO IV RESULTADOS Y DISCUSIÓN.....		39
1.13.	Entregables.....	39
4.1.1.	Definiendo roles de trabajo.....	39

4.1.2.	Definiendo el product backlog .....	40
1.14.	Sprint 0 .....	44
4.1.3.	Sprint planning .....	44
4.1.4.	Sprint backlog.....	45
4.1.5.	Arquitectura de la aplicación.....	47
4.1.6.	Resultados del sprint.....	47
1.15.	Sprint 1 .....	49
4.1.7.	Sprint planning .....	49
4.1.8.	Sprint backlog.....	51
4.1.9.	Historias de usuario .....	52
4.1.10.	Taskboard .....	60
4.1.11.	Burndown chart .....	66
4.1.12.	Daily scrum.....	67
4.1.13.	Sprint review.....	68
4.1.14.	Despliegue del sprint .....	69
4.1.15.	Criterios de aceptación .....	70
4.1.16.	Pruebas unitarias.....	73
4.1.17.	Resultados del sprint.....	75
4.1.18.	Sprint retrospective.....	83
1.16.	Sprint 2 .....	84
4.1.19.	Sprint planning .....	84
4.1.20.	Sprint backlog.....	86
4.1.21.	Historias de usuario .....	88
4.1.22.	Taskboard .....	101
4.1.23.	Burndown chart .....	108
4.1.24.	Daily scrum.....	109
4.1.25.	Sprint review.....	110
4.1.26.	Despliegue del sprint .....	111
4.1.27.	Criterios de aceptación .....	112
4.1.28.	Pruebas unitarias.....	115
4.1.29.	Resultados del sprint.....	119
4.1.30.	Sprint retrospective.....	137
1.17.	Sprint 3 .....	138
4.1.31.	Sprint planning .....	138

4.1.32.	Sprint backlog.....	141
4.1.33.	Historias de usuario .....	143
4.1.34.	Taskboard .....	154
4.1.35.	Burndown chart .....	161
4.1.36.	Daily scrum.....	162
4.1.37.	Sprint review.....	164
4.1.38.	Despliegue del sprint .....	166
4.1.39.	Criterios de aceptación .....	167
4.1.40.	Pruebas unitarias.....	171
4.1.41.	Resultados del sprint.....	172
4.1.42.	Sprint retrospective.....	178
1.18.	Resultados.....	180
4.1.43.	Resultados de funcionalidad de la aplicación.....	180
4.1.44.	Resultados derivados de la funcionalidad de la aplicación .....	182
CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES .....		188
1.19.	Conclusiones.....	188
1.20.	Recomendaciones .....	189
BIBLIOGRAFÍA .....		190
ANEXOS .....		195

## Lista de tablas

<b>Tabla 1.</b> Operacionalización de la variable.....	34
<b>Tabla 2.</b> Herramientas utilizadas para el tratamiento de datos. ....	36
<b>Tabla 3.</b> Entregables de Scrum. ....	38
<b>Tabla 4.</b> Roles de Scrum.....	39
<b>Tabla 5.</b> Product Backlog .....	40
<b>Tabla 6.</b> Sprint planning del sprint 0. ....	44
<b>Tabla 7.</b> Sprint backlog del sprint 0.....	46
<b>Tabla 8.</b> Sprint planning del sprint 1 .....	50
<b>Tabla 9.</b> Sprint backlog del sprint 1.....	51
<b>Tabla 10.</b> HU-SEG-001 Inicio de sesión .....	52
<b>Tabla 11.</b> HU-SEG-002 Gestión de roles y permisos.....	53
<b>Tabla 12.</b> HU-SEG-003 Gestión de usuarios.....	55
<b>Tabla 13.</b> HU-SEG-004 Restricciones de acceso en el sistema.....	60
<b>Tabla 14.</b> Estimación de trabajo en horas del sprint 1 .....	61
<b>Tabla 15.</b> Taskboard del sprint 1 .....	62
<b>Tabla 16.</b> Burndown chart del sprint 1 .....	66
<b>Tabla 17.</b> Daily scrum del sprint 1.....	67
<b>Tabla 18.</b> Sprint review del sprint 1.....	68
<b>Tabla 19.</b> Despliegue del sprint 1 .....	69
<b>Tabla 20.</b> Criterios de aceptación del sprint 1 .....	70
<b>Tabla 21.</b> Prueba unitaria - sprint 1 .....	73
<b>Tabla 22.</b> Sprint retrospective del sprint 1 .....	83
<b>Tabla 23.</b> Sprint planning del sprint 2 .....	85
<b>Tabla 24.</b> Sprint backlog del sprint 2.....	86
<b>Tabla 25.</b> HU-CONF-001 Gestión de ciclo académico .....	88
<b>Tabla 26.</b> HU-CONF-002 gestión de secciones .....	91
<b>Tabla 27.</b> HU-CONF-003 gestión de niveles .....	92
<b>Tabla 28.</b> HU-CONF-004 gestión de grados .....	94
<b>Tabla 29.</b> HU-CONF-005 gestión de tipos de matrícula .....	95
<b>Tabla 30.</b> HU-CONF-006 gestión de descuentos .....	97
<b>Tabla 31.</b> HU-CONF-007 gestión de tipo de operación.....	98
<b>Tabla 32.</b> HU-CONF-008 gestión de métodos de pago.....	100
<b>Tabla 33.</b> Estimación de trabajo en horas del sprint 2.....	101
<b>Tabla 34.</b> Taskboard del sprint 2 .....	102
<b>Tabla 35.</b> Daily scrum del sprint 2.....	109
<b>Tabla 36.</b> Sprint review del sprint 2.....	110
<b>Tabla 37.</b> Despliegue del sprint 2 .....	111
<b>Tabla 38.</b> Criterios de aceptación sprint 2 .....	112
<b>Tabla 39.</b> Prueba unitaria - sprint 2 .....	115
<b>Tabla 40.</b> Sprint retrospective del sprint 2.....	137
<b>Tabla 41.</b> Sprint planning del sprint 3 .....	139
<b>Tabla 42.</b> Product backlog del sprint 3 .....	141
<b>Tabla 43.</b> HU-OPER-001 registro de matrícula .....	143
<b>Tabla 44.</b> HU-OPER-002 editar matrícula .....	144
<b>Tabla 45.</b> HU-OPER-003 pago de pensiones .....	148

<b>Tabla 46.</b>	HU-OPER-004 anular operación.....	150
<b>Tabla 47.</b>	HU-OPER-005 gestión de deudas y mora.....	151
<b>Tabla 48.</b>	HU-ACA-001 reporte de estudiantes .....	152
<b>Tabla 49.</b>	HU-ACA-002 filtrar estudiantes por ciclo. ....	153
<b>Tabla 50.</b>	HU-ACA-003 reporte de estudiantes con pensiones vencidas. ....	153
<b>Tabla 51.</b>	Estimación de trabajo en horas del sprint 3.....	154
<b>Tabla 52.</b>	Taskboard del sprint 3 .....	155
<b>Tabla 53.</b>	Daily scrum - sprint 3 .....	162
<b>Tabla 54.</b>	Sprint review del sprint 3.....	165
<b>Tabla 55.</b>	Despliegue del sprint 3. ....	166
<b>Tabla 56.</b>	Criterios de aceptación del sprint 3 .....	167
<b>Tabla 57.</b>	Prueba unitaria - sprint 3 .....	171
<b>Tabla 58.</b>	Sprint retrospective del sprint 3.....	178
<b>Tabla 59.</b>	Resultados de evaluación de funcionalidad de la aplicación.....	180
<b>Tabla 60.</b>	Tiempos registrados en la ejecución de los procesos evaluados. ....	182
<b>Tabla 61.</b>	Resultados comparativos en la gestión de matrícula. ....	183

## Lista de figuras

<b>Figura 1.</b> Organigrama de la Institución Educativa Nuestra Señora de Guadalupe .....	10
<b>Figura 2.</b> Modelo en cascada .....	14
<b>Figura 3.</b> Características deseadas de los roles centrales de Scrum. ....	16
<b>Figura 4.</b> Pila del sprint .....	19
<b>Figura 5.</b> El ciclo iterativo en SCRUM .....	20
<b>Figura 6.</b> Organización del MVC .....	22
<b>Figura 7.</b> Jerarquía de herencia de personal académico .....	24
<b>Figura 8.</b> Repositorio de la aplicación configurado en Azure DevOps .....	47
<b>Figura 9.</b> Estructura inicial del proyecto. ....	48
<b>Figura 10.</b> Wireframe de la vista de inicio de sesión. ....	53
<b>Figura 11.</b> Wireframe de la vista listado de roles. ....	54
<b>Figura 12.</b> Wireframe de la vista de creación de roles. ....	55
<b>Figura 13.</b> Wireframe de la vista de listado de usuarios. ....	57
<b>Figura 14.</b> Wireframe de la vista de creación de usuarios - paso 1: Datos personales.....	57
<b>Figura 15.</b> Wireframe de la vista de creación de usuarios - paso 2: Documento. ....	58
<b>Figura 16.</b> Wireframe de la vista de creación de usuarios - paso 3: Contacto de emergencia. ....	58
<b>Figura 17.</b> Wireframe de la vista de creación de usuarios - paso 4: Contacto. ....	59
<b>Figura 18.</b> Wireframe de la vista de creación de usuarios - paso 5: Asignación de rol. ....	59
<b>Figura 19.</b> Resultado de las pruebas unitarias de autenticación de usuarios. ....	73
<b>Figura 20.</b> Resultado de las pruebas unitarias de gestión de roles. ....	74
<b>Figura 21.</b> Resultado de las pruebas unitarias de gestión de usuarios. ....	74
<b>Figura 22.</b> Modelo físico de base de datos - Gestión de persona .....	75
<b>Figura 23.</b> Modelo físico de base de datos - Módulo seguridad.....	76
<b>Figura 24.</b> Modelo físico de base de datos - Módulo operación .....	77
<b>Figura 25.</b> Modelo físico de base de datos - Módulo académico y configuración.....	78
<b>Figura 26.</b> Interfaz de inicio de sesión de la aplicación. ....	79
<b>Figura 27.</b> Interfaz de la página principal de rol. ....	79
<b>Figura 28.</b> Interfaz de creación de rol.....	80
<b>Figura 29.</b> Interfaz de creación de rol - asignación de permisos. ....	80
<b>Figura 30.</b> Interfaz de edición de rol. ....	81
<b>Figura 31.</b> Interfaz de editar rol - asignación de permisos. ....	81
<b>Figura 32.</b> Interfaz de eliminado de rol. ....	82
<b>Figura 33.</b> Interfaz de listado de usuarios.....	82
<b>Figura 34.</b> Wireframe de la vista de listado de ciclo académico. ....	89
<b>Figura 35.</b> Wireframe de la vista de creación de ciclo académico - paso 1. ....	89
<b>Figura 36.</b> Wireframe de creación de ciclo académico -- paso 2. ....	90
<b>Figura 37.</b> Wireframe de creación de ciclo académico -- paso 3. ....	90
<b>Figura 38.</b> Wireframe de la vista de listado de secciones.....	91
<b>Figura 39.</b> Wireframe de creación y edición de sección. ....	92
<b>Figura 40.</b> Wireframe de listado de niveles.....	93
<b>Figura 41.</b> Wireframe de creación y edición de niveles. ....	93
<b>Figura 42.</b> Wireframe de listado de grados. ....	94
<b>Figura 43.</b> Wireframe de creación y edición de grados.....	95
<b>Figura 44.</b> Wireframe de listado de tipos de matrícula. ....	96
<b>Figura 45.</b> Wireframe de creación y edición de tipo de matrícula. ....	96

<b>Figura 46.</b> Wireframe de listado de descuentos.....	97
<b>Figura 47.</b> Wireframe de creación y edición de descuento. ....	98
<b>Figura 48.</b> Wireframe de listado de tipos de operación.....	99
<b>Figura 49.</b> Wireframe de creación y edición de tipo de operación.....	99
<b>Figura 50.</b> Wireframe de listado de métodos de pago. ....	100
<b>Figura 51.</b> Wireframe de creación y edición métodos de pago. ....	101
<b>Figura 52.</b> Burndown chart del sprint 2.....	108
<b>Figura 53.</b> Resultado de las pruebas unitarias de gestión de ciclos académicos. ....	116
<b>Figura 54.</b> Resultado de las pruebas unitarias de gestión de secciones.....	116
<b>Figura 55.</b> Resultado de las pruebas unitarias de gestión de niveles.....	117
<b>Figura 56.</b> Resultado de las pruebas unitarias de gestión de grados. ....	117
<b>Figura 57.</b> Resultado de las pruebas unitarias de gestión de tipos de matrícula. ....	117
<b>Figura 58.</b> Resultado de las pruebas unitarias de gestión de descuentos.....	118
<b>Figura 59.</b> Resultado de las pruebas unitarias de gestión de tipos de operación.....	118
<b>Figura 60.</b> Resultado de las pruebas unitarias de gestión de métodos de pago. ....	118
<b>Figura 61.</b> Interfaz de listado de ciclo académico del módulo configuración.....	119
<b>Figura 62.</b> Interfaz de creación de ciclo académico – paso 1: ciclo académico.....	120
<b>Figura 63.</b> Interfaz de creación de ciclo académico – paso 2: configuración financiera.....	120
<b>Figura 64.</b> Interfaz de creación de ciclo académico – paso 3: Descuentos. ....	121
<b>Figura 65.</b> Interfaz de edición de ciclo académico – paso 1: ciclo académico.....	121
<b>Figura 66.</b> Interfaz de edición de ciclo académico – paso 2: configuración financiera. ....	122
<b>Figura 67.</b> Interfaz de edición de ciclo académico – paso 3: descuentos. ....	122
<b>Figura 68.</b> Interfaz de eliminado de ciclo académico.....	123
<b>Figura 69.</b> Interfaz de listado de tipo de matrícula.....	123
<b>Figura 70.</b> Interfaz de creación de tipo de matrícula.....	124
<b>Figura 71.</b> Interfaz de edición de tipo de matrícula.....	124
<b>Figura 72.</b> Interfaz de eliminado de tipo de matrícula.....	125
<b>Figura 73.</b> Interfaz de listado de sección.....	125
<b>Figura 74.</b> Interfaz de creación de sección.....	126
<b>Figura 75.</b> Interfaz de edición de sección.....	126
<b>Figura 76.</b> Interfaz de eliminado de sección.....	127
<b>Figura 77.</b> Interfaz de listado de grado.....	127
<b>Figura 78.</b> Interfaz de creación de grado.....	128
<b>Figura 79.</b> Interfaz de edición de grado.....	128
<b>Figura 80.</b> Interfaz de eliminado de grado.....	129
<b>Figura 81.</b> Interfaz de listado de nivel.....	129
<b>Figura 82.</b> Interfaz de creación de nivel.....	130
<b>Figura 83.</b> Interfaz de edición de nivel.....	130
<b>Figura 84.</b> Interfaz de eliminado de nivel.....	131
<b>Figura 85.</b> Interfaz de listado de descuento.....	131
<b>Figura 86.</b> Interfaz de creación de descuento.....	132
<b>Figura 87.</b> Interfaz de edición de descuento.....	132
<b>Figura 88.</b> Interfaz de eliminado de descuento.....	133
<b>Figura 89.</b> Interfaz de listado de tipo de operación.....	133
<b>Figura 90.</b> Interfaz de creación de tipo de operación.....	134
<b>Figura 91.</b> Interfaz de edición de tipo de operación.....	134
<b>Figura 92.</b> Interfaz de eliminado de tipo de operación.....	135

<b>Figura 93.</b>	Interfaz de listado de método de pago. ....	135
<b>Figura 94.</b>	Interfaz de creación de método de pago.....	136
<b>Figura 95.</b>	Interfaz de edición de método de pago. ....	136
<b>Figura 96.</b>	Interfaz de eliminado de método de pago. ....	137
<b>Figura 97.</b>	Wireframe de registro y edición de matrícula - paso 1.....	145
<b>Figura 98.</b>	Wireframe de registro y edición de matrícula paso 2. ....	145
<b>Figura 99.</b>	Wireframe de registro y edición de matrícula paso 3. ....	146
<b>Figura 100.</b>	Wireframe de registro y edición de matrícula paso 4. ....	146
<b>Figura 101.</b>	Wireframe de registro y edición de matrícula paso 5. ....	147
<b>Figura 102.</b>	Wireframe de registro y edición de matrícula paso 6. ....	147
<b>Figura 103.</b>	Wireframe de pago de pensión - paso 1.....	149
<b>Figura 104.</b>	Wireframe de pago de pensión - paso 2.....	149
<b>Figura 105.</b>	Wireframe de pago de pensión - paso 3.....	150
<b>Figura 106.</b>	Wireframe de listado de estudiantes matriculados.....	152
<b>Figura 107.</b>	Burndown chart del sprint 3.....	161
<b>Figura 108.</b>	Resultado de las pruebas unitarias de gestión de matrícula. ....	171
<b>Figura 109.</b>	Interfaz del módulo de operaciones. ....	172
<b>Figura 110.</b>	Interfaz de creación de matrícula - paso 1: datos personales.....	172
<b>Figura 111.</b>	Interfaz de creación de matrícula - paso 2: documento. ....	173
<b>Figura 112.</b>	Interfaz de creación de matrícula - paso 3: contacto de emergencia. ....	173
<b>Figura 113.</b>	Interfaz de creación de matrícula – paso 4: contacto.....	174
<b>Figura 114.</b>	Interfaz de creación de matrícula - paso 5: datos académicos. ....	174
<b>Figura 115.</b>	Interfaz de creación de matrícula - paso 6: datos de facturación.....	175
<b>Figura 116.</b>	Interfaz de creación de matrícula - paso 7: resumen.....	175
<b>Figura 117.</b>	Interfaz de pago de pensiones – paso 1.....	176
<b>Figura 118.</b>	Interfaz de pago de pensiones. - paso 2. ....	176
<b>Figura 119.</b>	Interfaz de pago de pensiones - paso 3. ....	177
<b>Figura 120.</b>	Interfaz de anular operación.....	177
<b>Figura 121.</b>	Reporte de estudiantes matriculados.....	178
<b>Figura 122.</b>	Situación actual en la gestión de matrícula. ....	183
<b>Figura 123.</b>	Indicador 1: Tiempo promedio para la gestión académica. ....	184
<b>Figura 124.</b>	Indicador 2: Tiempo promedio para la gestión de información del estudiante..	185
<b>Figura 125.</b>	Indicador 3: Tiempo promedio para el control de pagos. ....	186
<b>Figura 126.</b>	Porcentaje de reducción de tiempo con el uso de la aplicación. ....	187

## CAPÍTULO I

### PLANTEAMIENTO DE LA INVESTIGACIÓN

#### 1.1. Planteamiento del problema

La Institución Educativa “Nuestra Señora de Guadalupe”, ubicada en el distrito de Jesús Nazareno, provincia de Huamanga, región Ayacucho, con dirección en Av. Abraham Valdelomar N.º 1270, enfrenta serias dificultades en la gestión de matrícula de sus estudiantes. Actualmente, todos los procesos relacionados con el registro de información, el control de pagos de pensiones y la gestión académica se realizan de forma manual, utilizando hojas de cálculo (Microsoft Excel), cuadernos físicos y recibos escritos. Este enfoque tradicional genera una alta dependencia del trabajo humano, incrementando el riesgo de errores y pérdida de información clave.

El colegio actualmente cuenta con 247 estudiantes, en promedio hay entre 250 a 260 estudiantes por año. Lo que implica la cantidad de registros de matrícula por año. El tiempo promedio de matrícula por estudiante es de 15 a 20 minutos, la secretaria registra información como datos personales, contacto de emergencia, datos de los padres, datos académicos entre otros. Sumada a este proceso también se presenta el pago de las pensiones, el colegio para el año 2025 tiene como día máximo para pagar la pensión correspondiente al mes anterior, el día 5 de cada mes, pasado el día se aplica una mora. A partir de este día la secretaria valida los pagos de los estudiantes para notificar a los padres de familia de quienes aún no pagaron, esta tarea suele demorar entre 6 a 8 horas.

Un caso que comúnmente sucede es del estudiante con pensiones retrasadas, cuando regulariza sus pensiones el cálculo de las moras la secretaria lo realiza de manera manual, muchas veces cayendo en errores de cálculo. Esta práctica de realizar los procesos de matrícula y pagos de manera manual ha caído en varias oportunidades en errores de registro, errores de validación de pagos, registros duplicados, pérdida de información, entre otros. Por la misma naturaleza del trabajo y carga laboral del personal encargado.

Las causas principalmente recaen en que no existe un asesoramiento profesional que permita dar soluciones tecnológicas para mejorar o automatizar estos procesos, la nula iniciativa de quienes dirigen la institución y las limitaciones presupuestarias, son factores que dan como resultado al problema encontrado.

De continuar con la misma manera en que manejan sus procesos, las consecuencias en la institución son claras, pérdida creciente de dinero, pérdida de información valiosa y la reputación ante los padres de familia.

Frente a esta situación, el presente proyecto tiene como objetivo determinar en qué medida el uso de la aplicación web desarrollado mediante el marco de trabajo Scrum, mejorará la gestión de matrícula de estudiantes, mediante el prototipo se podrá evidenciar la mejora de la eficiencia administrativa, reducción de los errores humanos y la optimización el control de la información educativa y financiera.

## **1.2. Formulación del problema**

### **1.1.1. Problema general**

¿En qué medida el uso de la aplicación web desarrollado mediante el marco de trabajo Scrum, mejorará la gestión de matrícula de estudiantes en la Institución Educativa “Nuestra Señora De Guadalupe”?

### **1.1.2. Problemas específicos**

- a) ¿En qué medida el uso de la aplicación web reducirá el tiempo en la gestión académica?
- b) ¿En qué medida el uso de la aplicación web reducirá el tiempo en la gestión de información del estudiante?
- c) ¿En qué medida el uso de la aplicación web reducirá el tiempo en el control de pago de pensiones?

## **1.3. Justificación y delimitación de la investigación**

### **1.1.3. Importancia**

La gestión de matrícula constituye un proceso fundamental dentro de las instituciones educativas, ya que de su correcta ejecución depende la organización académica y administrativa de cada periodo escolar. Cuando este proceso se realiza de manera manual, mediante registros en papel u hojas de cálculo, se incrementa el riesgo de errores, duplicidad de información y pérdida de datos, lo que afecta la eficiencia del trabajo administrativo y la satisfacción de los usuarios.

La automatización de este procedimiento mediante una aplicación web permitirá reducir de manera significativa el tiempo destinado a la gestión de matrícula y al control de pagos, optimizando el uso de los recursos humanos, mejorando la exactitud de los registros y elevando la calidad del servicio ofrecido a la comunidad educativa.

#### **1.1.4. Justificación**

La necesidad de desarrollar una aplicación web para la gestión de matrícula, surge de las deficiencias detectadas en el sistema manual actualmente utilizado por la institución: duplicidad de registros, errores en la consolidación de pagos y una carga laboral excesiva para el personal administrativo, quienes invierten entre 6 a 8 horas mensuales en la validación de información.

La propuesta de una aplicación web permitirá centralizar los datos, automatizar tareas repetitivas, mejorar la integridad de la información y como consecuencia reducir el tiempo en la gestión de matrícula.

#### **1.1.5. Delimitación de la investigación**

La presente investigación se delimita al desarrollo de una aplicación web para la gestión de matrícula en la Institución Educativa “Nuestra Señora de Guadalupe”.

Si bien la institución realiza diversos procesos administrativos y académicos como control de asistencia, gestión de cursos, evaluación y reportes de calificaciones, en este estudio se prioriza el proceso de matrícula por su alto impacto en la eficiencia administrativa y por ser el punto de partida de la gestión académica.

### **1.4. Objetivos**

#### **1.1.6. Objetivo general**

Determinar en qué medida el uso de la aplicación web desarrollado mediante el marco de trabajo scrum, mejorará la gestión de matrícula de estudiantes en la Institución Educativa “Nuestra Señora De Guadalupe”.

#### **1.1.7. Objetivos específicos**

- a) Determinar en qué medida el uso de la aplicación web reducirá el tiempo en la gestión académica.

- b) Determinar en qué medida el uso de la aplicación web reducirá el tiempo en la gestión de información del estudiante.
- c) Determinar en qué medida el uso de la aplicación web reducirá el tiempo en el control de pago de pensiones.

## **CAPÍTULO II**

### **MARCO TEÓRICO**

#### **1.5. Antecedentes de la investigación**

##### **2.1.1. A nivel internacional**

Recalde y Zurita (2020) en su tesis titulada “Sistema web para la gestión académica y administrativa de la empresa de capacitación profesional DIENAV”, Ecuador. Tuvo como objetivo implementar un Sistema Web para la Gestión Académica y Administrativa de Empresa de Capacitación Profesional DIENAV con la finalidad de permitir una gestión eficiente de los procesos académicos y administrativos. El estudio empleó dos clases de investigación de campo y bibliográfico. Esto permitió la validación en campo y un uso real de la aplicación. Como resultado, se obtuvo una aplicación web funcional e implementada. Se observó que las encuestas aplicadas incluían preguntas poco relacionadas con los objetivos específicos de la investigación. El presente trabajo busca tener en cuenta esta limitación mediante instrumentos precisos y alineados con los objetivos de la investigación.

Calamani (2020) en su tesis “Sistema de Información para el Registro y Seguimiento Académico” en la Academia de Formación Profesional “Bela”, institución dedicada a la formación en estilismo y estética facial. El objetivo principal fue implementar un sistema web que optimizara la gestión académica mediante el registro y seguimiento de estudiantes. Para el análisis y diseño se aplicó la metodología UWE (UML-based Web Engineering), y el desarrollo se realizó utilizando PHP, facilitando interfaces amigables para los usuarios. La implementación del sistema mejoró la administración de la información académica, proporcionando datos oportunos y precisos para la toma de decisiones institucionales. Sin embargo, una limitación del estudio fue la falta de evaluación cuantitativa del impacto del sistema en la eficiencia administrativa, lo que dificulta medir objetivamente las mejoras alcanzadas. En contraste, la presente investigación busca superar esta debilidad mediante la inclusión de indicadores de rendimiento y métricas específicas que permitan evaluar el impacto del sistema en la gestión de matrículas y control de pensiones en instituciones educativas escolares.

Pulido y Martínez (2022) desarrollaron la tesis titulada “Desarrollo de una aplicación web para la inscripción de cursos libres del Departamento de Computación de la UNAN-LEÓN, mediante la utilización del framework Laravel Livewire de PHP”. El estudio tuvo como

propósito implementar una solución tecnológica que optimizara el proceso de inscripción a cursos libres, permitiendo a los estudiantes registrar sus datos y seleccionar los cursos de forma digital. El enfoque metodológico fue proyectivo, y se aplicaron tecnologías modernas como PHP con Laravel Livewire y bases de datos MySQL. Como resultado, se logró desarrollar una aplicación web de acuerdo a las necesidades planteadas. Sin embargo, una limitación del trabajo fue la escasa inclusión de indicadores cuantitativos que muestren de forma concreta los beneficios obtenidos.

Freire y Sánchez (2023) desarrollaron el proyecto titulado “Desarrollo de una aplicación web para la sistematización de la gestión de los horarios de docentes, estudiantes y aulas para la Universidad Técnica de Cotopaxi – Extensión La Maná”. El objetivo fue diseñar e implementar un sistema web que automatice la planificación y asignación de horarios, a fin de optimizar el uso de los recursos académicos y mejorar la organización interna. La investigación combinó los enfoques bibliográfico de campo y aplicado, y utilizó como instrumento una guía de entrevistas. Como resultado, se logró mejorar la coordinación entre docentes y estudiantes, reducir los conflictos de horario, y facilitar la toma de decisiones administrativas. No obstante, una limitación del estudio fue su enfoque exclusivo en la gestión de horarios, sin abordar otros procesos clave como la matrícula, los pagos o la gestión de información estudiantil, lo cual restringe la aplicabilidad del sistema en instituciones que requieren soluciones integrales.

Franco y Cerezo (2024) desarrollaron el trabajo titulado “Desarrollo de un sistema de gestión para el control de estudiantes en los cursos del Centro de Emprendimiento y Optimización Municipal de Samborondón (CEYOM)”, con el objetivo de mejorar el proceso de inscripción y seguimiento de estudiantes mediante una aplicación web personalizada. Utilizando tecnologías como HTML5, CSS3, PHP, MySQL y el framework Scrum para el desarrollo ágil, el sistema automatizó tareas anteriormente realizadas en hojas de Excel, lo que permitió reducir errores manuales, agilizar los registros y mejorar la experiencia del usuario. El estudio combinó enfoques cualitativos y cuantitativos, aplicando encuestas y entrevistas para definir requerimientos funcionales. Si bien el sistema mostró avances importantes en términos de eficiencia operativa y usabilidad, una limitación fue la ausencia de indicadores cuantitativos posteriores a la implementación que permitan medir con claridad el impacto real del sistema sobre variables como la reducción de tiempo, errores o satisfacción del usuario.

### **2.1.2. A nivel nacional**

Dioses, et al., (2022) realizaron un estudio Titulado: Implementación de un sistema web, utilizando metodología Scrum para mejorar los procesos de matrícula y control de pagos en la Institución Educativa Particular El Triunfo. El objetivo del estudio fue Implementar un sistema WEB, utilizando metodología SCRUM para mejorar el proceso de matrícula y control de pagos en la Institución Educativa El Triunfo. Tuvo como muestra 20 personas entre: Padres de familia, profesores y alumnos. El diseño que se utilizó fue no experimental, el instrumento utilizado encuesta. Obteniendo los resultados 90% indica de estar Muy de acuerdo con la implementación de este nuevo sistema utilizando SCRUM, seguido de un 10% indica No estar Seguros sobre esta nueva implementación en la Institución Educativa El Triunfo. Si bien los resultados muestran la exitosa implementación del sistema, la manera en que se aplica la encuesta y el contenido de esta no enriquece al lineamiento del objetivo de la investigación.

Alburqueque (2022) realizó el estudio titulado “Implementación de un sistema de matrícula web en la I.E. Ignacio Merino - Piura; 2020”, en la ciudad de Piura, Perú. El objetivo del estudio fue implementar un sistema de matrícula web en la institución educativa con el fin de mejorar los procesos administrativos. La investigación, de tipo cuantitativo con diseño no experimental y corte transversal, tuvo como muestra a cinco directivos, utilizando como instrumento un cuestionario. Los resultados indicaron que, en la dimensión “nivel de satisfacción con el sistema actual”, el 80% del personal administrativo manifestó estar en desacuerdo con el sistema vigente. Respecto a la segunda dimensión, “nivel de satisfacción con la propuesta de mejora”, el 100% expresó su conformidad con la implementación del nuevo sistema de matrícula. Si bien los resultados muestran una aceptación favorable hacia la propuesta tecnológica, existen algunas limitaciones metodológicas relevantes. Por un lado, la muestra fue muy reducida y homogénea (solo cinco directivos), lo que limita la validez externa de los resultados y restringe la posibilidad de generalizar los hallazgos a toda la comunidad educativa. Por otro lado, no se explica de forma clara cómo los datos recolectados mediante el cuestionario influyeron en la implementación y evaluación funcional del sistema web, ni se detallan pruebas técnicas o métricas de desempeño del software. Estas omisiones reducen la rigurosidad del proceso de validación de la propuesta.

Rugel (2022) desarrolló un sistema web para optimizar la gestión de matrículas en la Institución Educativa Inmaculada Concepción de Tumbes. El estudio identificó problemas administrativos como retrasos en la elaboración de reportes, duplicidad de datos y ampliación innecesaria del tiempo de clases debido a procesos manuales ineficientes. La investigación, de tipo cuantitativo con diseño no experimental y corte transversal, incluyó a 23 participantes entre personal administrativo, directivos, APAFA y padres de familia. Los resultados mostraron que el 60.87% de los encuestados no estaban satisfechos con el sistema actual de gestión de matrícula, y el 91.30% consideraron necesaria la implementación de un sistema web. Aunque la propuesta tecnológica abordó las necesidades inmediatas de automatización, sin embargo la desconexión entre el objetivo general con los específicos puede afectar la coherencia del estudio, ya que no se profundiza la variable en estudio.

Chiuca y Laguna (2023) desarrollaron un sistema web para optimizar la gestión de matrículas en la Institución Educativa Mártir José Olaya, ubicada en Musho, Áncash. El proyecto tuvo como objetivo reemplazar los procesos manuales y descentralizados mediante una plataforma digital que automatizara el registro de estudiantes, la asignación de secciones y la generación de reportes académicos. Utilizando tecnologías como PHP, MySQL y HTML5, y aplicando la metodología ágil Scrum, lograron reducir los tiempos de inscripción y minimizar errores administrativos. La validación del sistema se realizó mediante encuestas a usuarios y análisis comparativos antes y después de su implementación. Aunque el sistema mejoró significativamente la eficiencia operativa, una limitación fue la falta de integración con otros sistemas institucionales, como la gestión de pagos o el seguimiento académico a largo plazo.

Perez (2023) desarrolló un sistema web para optimizar la gestión del proceso de matrícula en un colegio privado de Trujillo. La investigación identificó problemas administrativos como duplicidad de registros, errores en cálculos y demoras en la elaboración de reportes, lo que afectaba la eficiencia del proceso de matrícula. Utilizando un diseño de investigación preexperimental, se aplicaron pruebas estadísticas como la de Wilcoxon para evaluar la efectividad del sistema implementado. Los resultados mostraron mejoras significativas en la reducción de tiempo, costos y errores en el proceso de matrícula. Sin embargo, una limitación del estudio fue la falta de integración del sistema con otros módulos institucionales.

## **1.6. Marco teórico**

### **2.1.3. La institución**

La Institución Educativa "Nuestra Señora de Guadalupe" es un colegio privado ubicado en el distrito de Jesús Nazareno, en la provincia de Huamanga, región Ayacucho, Perú. Ofrece educación integral en los niveles de inicial, primaria y secundaria, enfocándose en formar estudiantes con sólidos valores éticos, académicos y sociales. La institución busca proporcionar una enseñanza de calidad, adaptada a las necesidades de su comunidad y alineada con las demandas educativas modernas.

#### **2.1.3.1. Misión y Visión**

##### **Misión**

Somos una institución con personal capacitado y motivado que brinda servicio educativo integral y de calidad, con carácter humanístico y científico que orienta a formar ciudadanos responsables y comprometidos en la sociedad.

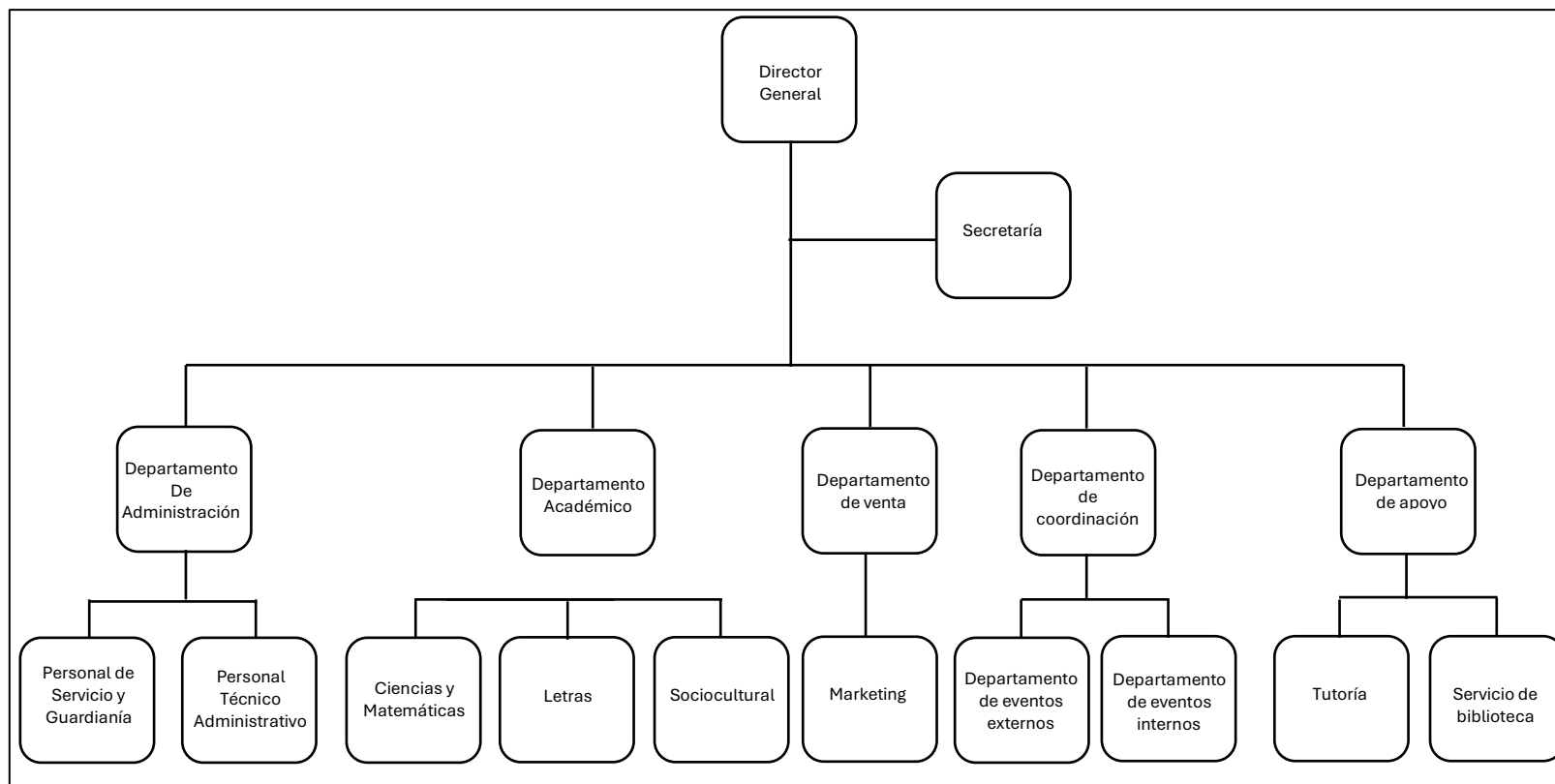
##### **Visión**

Ser una organización educativa que mejore los estándares de calidad de vida y así lograr una cultura de excelencia en todas sus integrantes.

#### **2.1.3.2. Organigrama**

**Figura 1.**

*Organigrama de la Institución Educativa Nuestra Señora de Guadalupe*



*Nota.* Elaboración propia con base en el organigrama institucional de la Institución Educativa Nuestra Señora de Guadalupe (fotografía tomada en secretaría, 2024).

#### **2.1.4. Aplicación Web**

Según (Pressman, 2010) las aplicaciones web conocidas como "webapps", pertenecen a una categoría de software orientadas a la red, inicialmente estas aplicaciones funcionaban como páginas informativas, en archivos de hipertexto, con contenido básico texto e imágenes. Con el auge de las webs 2.0 las aplicaciones fueron evolucionando a plataformas robustas, pudiéndose integrar con bases de datos empresariales, negocios y organizaciones.

Una aplicación web es un sistema basado en el modelo cliente-servidor, donde el cliente está representado por el navegador web y el servidor por un servidor web. Ambos se comunican mediante el protocolo HTTP, el cual está estandarizado, por lo que no requiere ser desarrollado desde cero por los programadores. HTTP forma parte de la familia de protocolos TCP/IP utilizada en Internet, lo que facilita la conexión entre sistemas heterogéneos y el intercambio de información entre diferentes computadoras. En el modelo OSI, este protocolo opera en la capa de aplicación, correspondiente al nivel 7 (Luján Mora, 2002).

#### **2.1.5. Matrícula**

La matrícula, dentro del ámbito educativo, se entiende como el proceso mediante el cual un estudiante se inscribe de manera formal en un programa académico. Este trámite puede realizarse tanto para quienes continúan sus estudios como por aquellos que se incorporan por primera vez al sistema educativo. Su propósito principal es recopilar la información personal, académica y administrativa que exige cada institución. En la práctica, el proceso suele incluir el llenado de formularios, la entrega de documentos, el pago correspondiente y el cumplimiento de los plazos establecidos (Red Educa, 2024).

#### **2.1.6. Gestión Académica**

Según (Blanco & Quesada, 2014) La gestión académica es la columna vertebral en toda institución educativa, tiene por objetivo coordinar diversos procesos y actividades. Coordinar los planes de estudio, metodologías pedagógicas, actividades de aprendizaje y actividades académicas. Su importancia radica en el impacto que tiene sobre la calidad institucional, al influir en aspectos como la evaluación del rendimiento estudiantil, la actualización de los programas formativos y la adaptación de los currículos a las demandas actuales.

### **2.1.7. Gestión de Información**

Según (Turban et al., 2018) la gestión de la información es la parte medular en toda organización, ya que a través de este proceso se usa o aprovecha de la mejor manera el uso del recurso principal, que es la información. Mediante diferentes metodologías, métodos, estrategias y empleando diversas tecnologías se asegura que la información esté protegida, accesible y organizada de manera óptima. La clave importante para alcanzar el éxito en toda organización es la calidad y disponibilidad de la información.

### **2.1.8. Política y Gestión de Cobros**

Según una publicación de (Centre Metal Lúrgic) la política y gestión de cobros es muy importante dentro de la administración de tesorería de una organización. Ya que los ingresos de facturación representan una fuente importante de financiamiento y la adecuada administración de estos procesos permite responder rápidamente a las necesidades de la organización.

### **2.1.9. Proceso de Desarrollo del Software**

El proceso de desarrollo de software es una estructura conformada por distintas actividades, acciones y tareas. Esto permite que el software sea desarrollado con calidad y entregado en plazos determinados. Esta estructura contine cinco actividades principales. Comunicación, planeación, modelado, construcción y despliegue. Cada actividad responde a una necesidad específica en el proceso de desarrollo, desde la interacción con el cliente para la recolección de requerimientos, hasta la entrega final y la retroalimentación. Este marco o estructura del proceso no es ley que se tiene que cumplir tal cual. Sino una es una guía flexible que permite a los equipos seleccionar las tareas más adecuadas para sus necesidades específicas, promoviendo la mejora continua y la adaptabilidad a diferentes contextos (Pressman, 2010).

Por otra parte (Sommerville, 2011) define el proceso del software como un conjunto de actividades relacionadas entre sí, que abarcan la especificación, el diseño, la validación y la evolución del sistema. Además, incorpora aspectos como los productos intermedios, los roles del equipo y las condiciones necesarias para cada fase. El autor distingue entre procesos planificados y procesos ágiles, señalando que la elección de uno u otro depende de las características del proyecto y del entorno en el que se desarrolla.

### **2.1.9.1. Modelo Cascada**

Según (Sommerville, 2011), el modelo cascada organiza las actividades del desarrollo en una secuencia lineal, donde cada fase depende de la finalización de la anterior este modelo se conoce como “modelo en cascada” o ciclo de vida del software. Las principales etapas del modelo incluyen:

#### **a) Análisis**

En esta etapa inicial, se recopilan los requisitos del sistema mediante consultas a los usuarios y partes interesadas. Los servicios, las restricciones y los objetivos del sistema se especifican de manera detallada. Estos requerimientos servirán como base para todo el desarrollo y serán documentados como una especificación formal del sistema, lo que ayuda a las expectativas del cliente con el producto final.

#### **b) Diseño**

Una vez que los requisitos están claramente definidos, el siguiente paso es la creación de un diseño detallado del sistema. Esto incluye tanto el diseño del sistema general como el diseño específico del software. El diseño del sistema aborda cómo se estructurarán y se conectarán los componentes del hardware y software, mientras que el diseño del software implica la identificación de las principales abstracciones y relaciones dentro del sistema, lo que permite que los desarrolladores comiencen a estructurar el software de manera coherente.

#### **c) Implementación**

En esta fase, el diseño se traduce en código. Los desarrolladores programan las distintas unidades o módulos de manera independiente y luego aplican pruebas unitarias para verificar su correcto funcionamiento. Detectar y corregir errores en esta etapa ahorra tiempo y evita problemas en etapas posteriores.

#### **d) Integración y prueba del sistema**

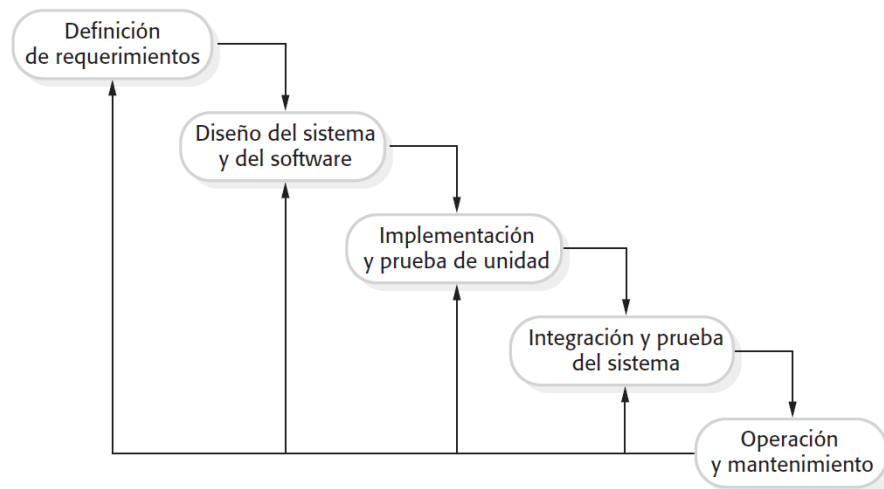
Una vez verificados los módulos, se integran para formar el sistema completo. Pues se realizan pruebas que confirman que todos los componentes interactúan de forma adecuada y que el sistema cumple con los requisitos iniciales. Esta verificación garantiza que el software funcione como un todo coherente.

### e) Operación y mantenimiento

Esta es la última fase del ciclo de vida del software, donde el sistema se pone en producción y se utiliza en un entorno real. Durante esta fase, se realizan tareas de mantenimiento, que incluyen la corrección de errores no detectados durante las pruebas, la mejora continua de la implementación y la actualización del sistema según sea necesario para satisfacer nuevos requisitos. Esta fase puede durar mucho tiempo y representa una porción significativa del ciclo de vida del software, ya que el mantenimiento es continuo, especialmente a medida que surgen nuevas necesidades del cliente o cambios tecnológicos.

#### Figura 2.

*Modelo en cascada*



*Nota.* Adaptado de *Ingeniería de Software* por (p. 30) por Ian Sommerville, 2011. PEARSON EDUCACIÓN.

#### 2.1.10. Scrum

Scrum es un marco de trabajo ágil, que permite a equipos y organizaciones desarrollar soluciones adaptables para problemas complejos aplicando un enfoque iterativo e incremental. Se fundamenta en el empirismo y en los principios del pensamiento Lean, que priorizan la experiencia, observación y la eliminación de desperdicios. A través de ciclos de trabajo llamados sprints, el equipo evalúa su progreso, adapta y mejora de forma continua su rendimiento, manteniendo la transparencia y centrando sus esfuerzos en la creación de valor (Schwaber & Sutherland, 2020).

### **2.1.10.1. Roles Scrum**

De acuerdo con (Satpathy, 2017), los roles de Scrum se dividen en dos categorías, roles centrales y no centrales. Los roles centrales están comprendidos por el Product Owner, el Scrum Master y el Equipo Scrum.

#### **a) Product Owner**

El Product Owner tiene la responsabilidad de asegurar que el proyecto aporte el mayor valor posible al negocio. Este rol actúa como la voz del cliente y se encarga de priorizar los requisitos según las necesidades del usuario y los objetivos del proyecto. Además, asegura que la justificación del negocio se mantenga a lo largo del desarrollo. En proyectos más grandes, puede haber variantes del rol, como un Program Product Owner para programas o un Portfolio Product Owner para portafolios, que asumen responsabilidades similares en esos contextos.

#### **b) Scrum Master**

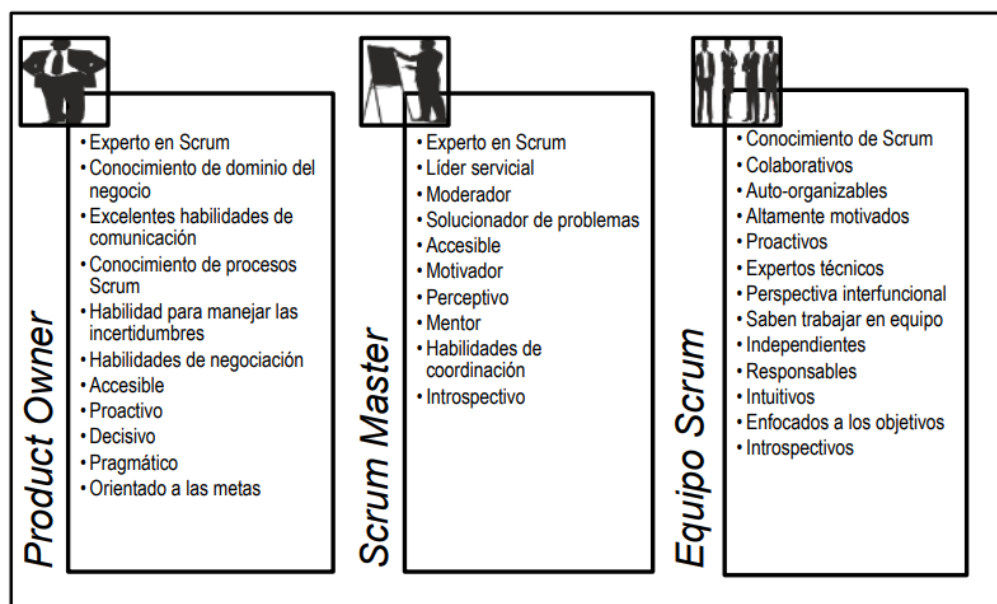
El Scrum Master actúa como un líder servicial dentro del Equipo Scrum, tiene por función principal moderar, facilitar y guiar las interacciones del equipo como coach y motivador. Este rol asegura un entorno de trabajo productivo, protegiendo al equipo de influencias externas, eliminando obstáculos que puedan dificultar su desempeño y velando por el cumplimiento de los principios, procesos y aspectos fundamentales del marco Scrum.

#### **c) Equipo Scrum**

El Equipo Scrum, también llamado equipo de desarrollo, es el grupo responsable de crear el producto, servicio o resultado esperado del proyecto. Está compuesto por personas que trabajan de manera colaborativa en las historias de usuario definidas en el Sprint Backlog, con el propósito de desarrollar los entregables acordados dentro del marco del Sprint.

**Figura 3.**

*Características deseadas de los roles centrales de Scrum.*



*Nota.* Adaptado de Una guía para el cuerpo de conocimiento de SCRUM (Guía SBok™) (p. 56) por Tridibesh Satpathy, 2017. SCRUMstudy™.

### 2.1.10.2. Eventos Scrum

Los eventos en Scrum están contenidos dentro del Sprint, que actúa como un marco para organizarlos. Estos eventos cumplen ciertos estándares para poder evitar reuniones improvisadas o innecesarias, lo que ocasiona pérdida y retrasos en el proyecto. Cada evento es una oportunidad formal para inspeccionar y adaptar los artefactos, esto garantiza la transparencia en el proceso. Realizándose siempre en el mismo lugar y horario para favorecer la organización del equipo (Schwaber & Sutherland, 2020).

#### a. Sprint

El Sprint es el ciclo central de Scrum, tiene una duración fija de un máximo de un mes, donde se desarrollan todas las actividades necesarias para alcanzar el objetivo del producto, incluyendo planificación, reuniones diarias, revisión y retrospectiva. Durante el desarrollo del Sprint, debe evitarse cualquier cambio que pueda poner en riesgo con el cumplimiento del objetivo, se refina el trabajo pendiente según sea necesario, y se puede ajustar el alcance en colaboración con el product owner. Los Sprints más cortos reducen riesgos y permiten ciclos de

aprendizaje más rápidos. En caso de que el objetivo se vuelva obsoleto, el product owner puede cancelar el Sprint.

### **b. Sprint Planning**

El sprint planning es el primer evento de un sprint, en este evento el equipo define el objetivo del sprint, selecciona todas las tareas del product backlog, evalúa su capacidad de desarrollo y planea como completarlos. La planificación responde a tres preguntas ¿qué valor aportará el sprint?, ¿qué se puede lograr? Y ¿cómo se llevará a cabo el trabajo? El resultado del sprint planning es el sprint backlog, un plan que organiza las tareas en incrementos manejables. La duración del sprint planning depende de la duración de un sprint, con un máximo de ocho horas para sprints de un mes.

### **c. Daily Meeting**

Es una reunión breve, de un tiempo máximo de 15 minutos, que se realiza diariamente en un lugar y hora fija durante el sprint. Su propósito principal es inspeccionar el progreso hacia el objetivo del sprint, ajustando el sprint backlog según sea necesario, enfocándose en las tareas pendientes y priorizando un plan accionable para el día siguiente. En este evento solo participan los desarrolladores, aunque el product owner o scrum master pueden participar si están trabajando activamente en los elementos del sprint. Este evento fomenta la autogestión, la comunicación, identifica impedimentos y elimina la necesidad de otras reuniones innecesarias.

### **d. Sprint Review**

En esta reunión, el equipo presenta los resultados del sprint a las partes interesadas. No se trata de una simple exposición, sino de un espacio colaborativo para analizar lo logrado, obtener retroalimentación y ajustar el product backlog de acuerdo con los cambios o nuevas oportunidades. La sesión puede durar hasta cuatro horas para sprints de un mes.

### **e. Retrospective**

Este evento se realiza al finalizar el Sprint y permite al equipo reflexionar sobre su desempeño, identificar áreas de mejora y definir acciones concretas para el siguiente ciclo. Se analizan los procesos, herramientas y dinámicas de trabajo, priorizando los cambios con mayor impacto. Puede durar hasta tres horas en sprints de una duración de un mes.

### **2.1.10.3. Artefactos Scrum**

Para (Palacio, 2024) los artefactos son elementos fundamentales en el marco scrum, diseñados para facilitar la interacción entre los roles y los eventos, asegurando que los objetivos del proyecto se alcancen de manera efectiva. Los principales artefactos incluyen:

#### **a. Product Backlog (Pila del Producto)**

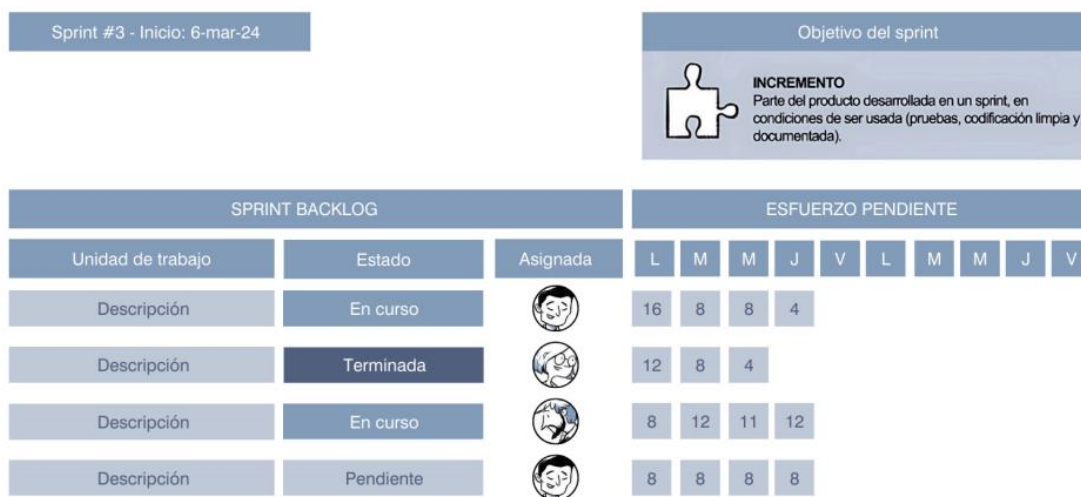
La pila del producto (Product Backlog), es una lista priorizada y dinámica que reúne los requisitos necesarios para desarrollar el producto, incluyendo funcionalidades, correcciones y mejoras. Cada elemento se organiza en forma de historias de usuario, las cuales representan necesidades específicas de los clientes y usuarios. Este artefacto evoluciona constantemente y es gestionado por el Product Owner, quien prioriza y detalla los elementos según su valor y urgencia. Las historias preparadas para un Sprint deben cumplir con criterios claros para garantizar su viabilidad. Este artefacto sirve como una herramienta clave para la planificación, la colaboración del equipo y la adaptación a cambios durante el desarrollo del proyecto.

#### **b. Sprint Backlog (Pila del Sprint)**

La pila del sprint (Sprint Backlog), es una lista de las tareas necesarias para completar el incremento de un Sprint. En ella, las historias de usuario se dividen en unidades de trabajo de tamaño adecuado para poder seguir el progreso diario y detectar posibles riesgos o problemas sin complicar la gestión. Durante la planificación del Sprint, el equipo colabora en su creación, asignando un esfuerzo estimado para cada unidad, utilizando métodos como los puntos o el tiempo ideal. El equipo de desarrollo es el único que puede modificarla durante el Sprint. Este artefacto proporciona una visualización clara del avance, por lo que se recomienda que esté accesible en un tablero físico o herramienta colaborativa, permitiendo que todos los miembros del equipo consulten el estado de las tareas y ajusten sus esfuerzos según sea necesario, siempre alineado con el objetivo del Sprint.

## Figura 4.

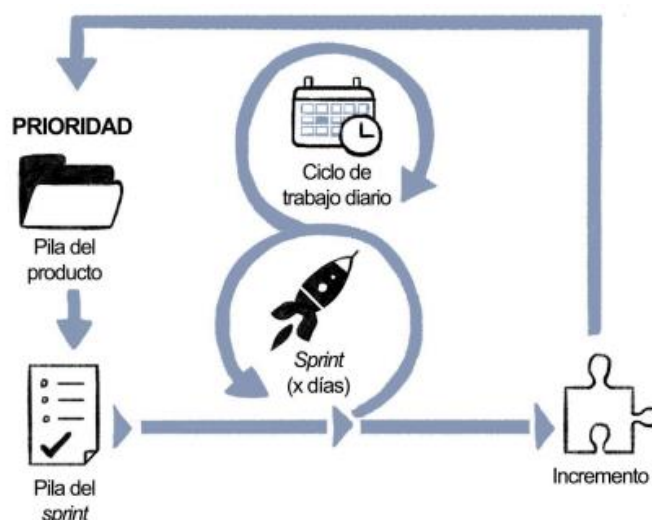
### Pila del sprint



Nota. Adaptado de *Scrum Master* (P. 32) por Marta Palacio, 2024. Scrum Manager®.

### c. Incremento

El incremento en Scrum, es la parte del producto desarrollada durante un Sprint que está lista para ser entregada al cliente, es decir, que está completa, probada y operativa. No se considera un incremento a prototipos, módulos o elementos pendientes de pruebas. Cada elemento de la pila del producto debe corresponder a funcionalidades entregables, y se espera que se produzca un incremento al final de cada Sprint. Sin embargo, el primer Sprint, conocido como "Sprint cero", puede ser una excepción, ya que se dedica a actividades previas como el diseño o la validación de herramientas. El incremento solo se considera "terminado" cuando se cumplen todos los criterios de calidad establecidos en la "definición de hecho", la cual debe ser compartida y entendida por todo el equipo.

**Figura 5.***El ciclo iterativo en SCRUM*

*Nota.* Adaptado de *Scrum Master* (P. 29) por Marta Palacio, 2024. Scrum Manager®.

#### **2.1.10.4. Unidades de Trabajo**

Según (Palacio, 2024) la unidad de trabajo es el componente que el equipo de Scrum utiliza para organizar y gestionar las tareas dentro de un sprint. Entre las unidades más comunes se encuentran las historias de usuario, las historias técnicas y las tareas.

##### **a. Historias de usuario**

Una historia de usuario es una descripción breve y clara del valor que un producto o servicio debe ofrecer al usuario final, escrita desde su punto de vista. Se crea durante las interacciones entre el equipo de desarrollo y el propietario del producto para asegurar una comprensión común, sirviendo como un recordatorio de lo discutido en esas conversaciones.

##### **b. Historia técnica**

Una historia técnica es similar a una historia de usuario, pero se enfoca en funcionalidades que no son directamente visibles para el usuario final. Esto incluye tareas como refactorizaciones o la implementación de medidas de seguridad, que aunque esenciales para el funcionamiento del sistema, no son percibidas por el usuario en su interacción directa con el producto.

### c. Tarea

Una tarea es una unidad de trabajo más detallada y pequeña en comparación con las historias de usuario y técnicas. Son actividades específicas que deben completarse para cumplir con los requisitos de una historia, como escribir código, diseñar interfaces o realizar pruebas.

#### 2.1.10.5. Técnicas de estimación y control en Scrum

- **Burndown Chart**

El Burndown Chart es una herramienta visual usada en la gestión ágil de proyectos, permite mostrar cuánto trabajo falta por hacer en función del tiempo. En este gráfico, el eje vertical representa la cantidad de tareas pendientes, mientras que el eje horizontal muestra el tiempo planificado para el sprint o el proyecto. Dentro del gráfico se dibuja una línea ideal, que refleja el ritmo de avance esperado, y otra línea real, que muestra el progreso que el equipo va logrando realmente. Comparar ambas líneas permite detectar retrasos, ajustar el ritmo de trabajo y tomar decisiones a tiempo para cumplir con los objetivos planteados (DoneTonic, 2025).

- **Planning Poker**

El Planning Poker es una técnica de estimación, utilizada en metodologías ágiles, permite determinar el esfuerzo de las tareas dentro del product backlog. Esta dinámica fomenta la participación de todos los miembros del equipo y busca alcanzar un consenso grupal sobre las estimaciones. Generalmente siguiendo la secuencia de Fibonacci, que representan el nivel de esfuerzo estimado (Cohn, 2025).

#### 2.1.11. Modelo Vista Controlador (MVC)

El patrón arquitectónico Modelo-Vista-Controlador (MVC) organiza un sistema en tres componentes lógicos independientes que interactúan para gestionar datos, presentación e interacción. Según (Sommerville, 2011), este patrón separa la presentación de los datos de la lógica de negocio y la interacción del usuario, promoviendo una estructura más modular y flexible. Se encarga de gestionar los datos del sistema y las operaciones relacionadas con ellos. Es el núcleo de la lógica de negocio.

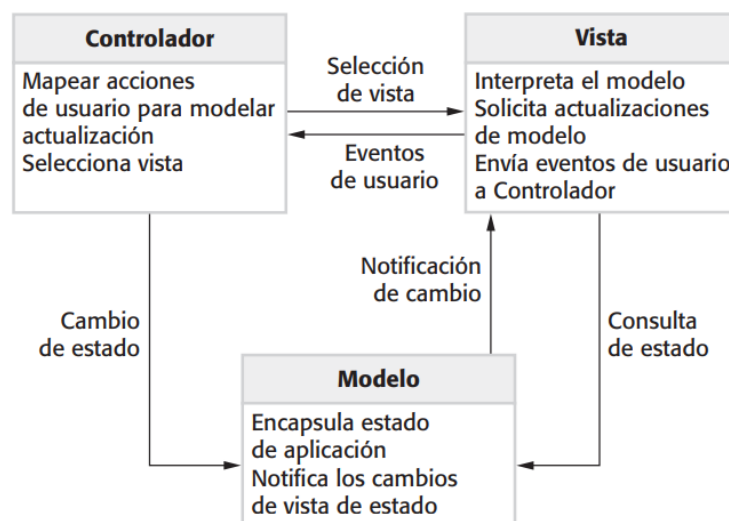
- Vista: Define y controla la presentación de los datos al usuario, asegurando que la interfaz sea comprensible y efectiva.

- Controlador: Gestiona las interacciones del usuario, como clics o pulsaciones de teclas, y las traduce en comandos para el Modelo y la Vista.

Esta arquitectura facilita la reutilización y el mantenimiento del código, permitiendo que cada componente evolucione de manera independiente, lo que es especialmente útil en aplicaciones complejas y dinámicas.

**Figura 6.**

*Organización del MVC*



*Nota.* Adaptado de *Ingeniería de Software* (p. 156), por Ian Sommerville, 2011. PEARSON EDUCACIÓN.

### 2.1.12. Programación Orientada a Objetos

Según (Joyanes Aguilar & Zahonero Martinez, 2011), la programación orientada a objetos (POO) es un paradigma de programación que divide el programa en módulos o componentes llamados objetos, cada uno de ellos tienen sus propios datos y funcionalidad. Estos objetos son autosuficientes que se comunican entre sí, de esta manera un sistema se hace mantenible y escalable. Este paradigma promueve la encapsulación y reutilización de código. La POO organiza los componentes del sistema en clases, estas clases actúan como plantillas para la creación de objetos, este proceso denominado instanciación. Como paradigma la POO se basa en cuatro pilares. La abstracción, encapsulamiento, polimorfismo y herencia.

Del modo (Flórez Fernández, 2012) resalta que la POO mediante la abstracción permite representar un sistema en objetos, ayudando a resolver problemas complejos. Ambos autores resaltan la importancia de la modularización, reutilización de código y la creación de aplicaciones más robustas y estructuradas.

#### **2.1.12.1. Abstracción**

La abstracción es un concepto clave en la programación, se refiere a la capacidad de centrarse solo en los aspectos más importantes o significativos de un problema, ignorando los detalles innecesarios. Es fundamental para representar información de manera que sea comprensible para el usuario, usando tipos definidos por el usuario o diseñando clases que implementan interfaces. Una clase como en Java o C++, permite traducir una abstracción en un tipo que combina datos y métodos para manipular esa información. Tiene distintos niveles, facilitando la gestión de la complejidad de los sistemas del mundo real. Es así que este pilar ayuda a entender lo que hace un sistema sin necesidad de conocer cómo lo hace, como en el ejemplo de la televisión, aunque no sepamos cómo funciona internamente, conocemos cómo usarla a través de su interfaz, como los botones o el control remoto (Joyanes Aguilar & Zahonero Martinez, 2011).

#### **2.1.12.2. Encapsulamiento**

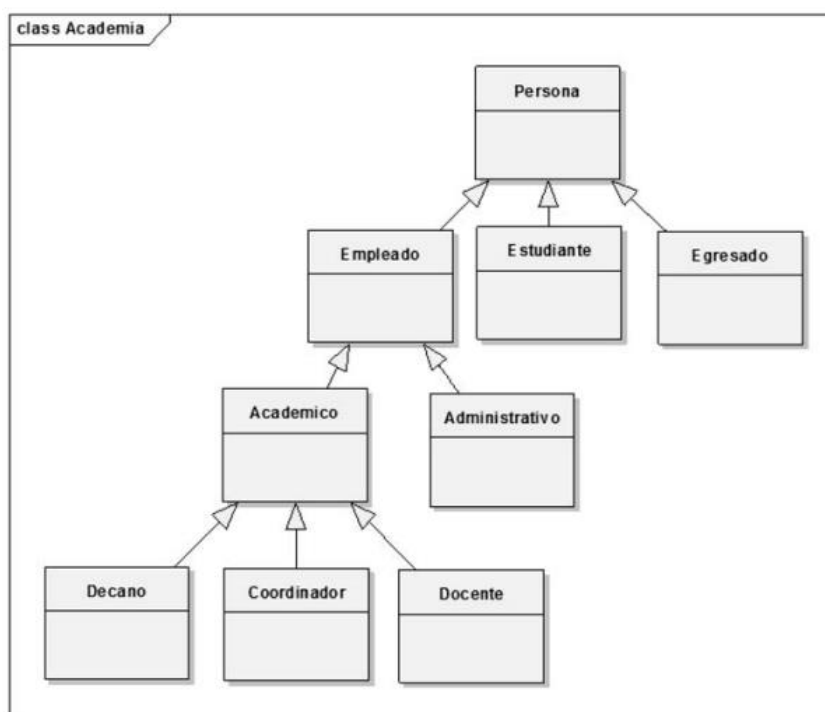
El encapsulamiento u ocultación de datos es un principio fundamental en la programación orientada a objetos. Consiste en agrupar datos y operaciones relacionadas bajo una misma unidad, aumentando la cohesión dentro del sistema. Los objetos con características y comportamientos similares se agrupan en clases, que encapsulan tanto los datos como las operaciones. Este proceso no solo organiza, sino que también oculta los detalles internos de un objeto, permitiendo que interactúe con el mundo exterior únicamente a través de una interfaz. Por ejemplo, una televisión o una lavadora tienen interfaces, como botones o mandos, que permiten al usuario controlar las funciones sin necesidad de entender cómo funcionan internamente. Esta separación entre la interfaz (lo que se puede hacer) y la implementación (cómo se hace) permite modificar la funcionalidad de un objeto sin afectar el resto del sistema, garantizando así su estabilidad (Joyanes Aguilar & Zahonero Martinez, 2011).

### 2.1.12.3. Herencia

La herencia es un concepto muy importante en la programación orientada a objetos, permite crear nuevas clases basadas en clases ya existentes, con el objetivo de reutilizar el código previamente desarrollado. Generando una jerarquía de clases dentro de la aplicación, donde la clase derivada hereda los atributos y métodos de la clase base. La clase derivada puede añadir nuevos atributos y métodos o modificar los existentes para especializar el comportamiento heredado, permitiendo la extensión y personalización de las funcionalidades sin necesidad de reescribir código (Flórez Fernández, 2012).

#### *Figura 7.*

*Jerarquía de herencia de personal académico*



*Nota.* Adaptado de *Programación Orientada a Objetos usando JAVA* (p. 120), Por Héctor Flórez. ECOE EDICIONES.

### 2.1.12.4. Polimorfismo

El polimorfismo es una característica clave de la programación orientada a objetos, permite a una función u operación tener el mismo nombre en diferentes clases, pero actuar de manera distinta en cada una. El polimorfismo implica que una operación puede ser interpretada y ejecutada de forma específica por cada objeto que la invoca, dependiendo de su clase. El

polimorfismo se realiza en tiempo de ejecución, lo que significa que el tipo de operación a ejecutar no se determina hasta que el programa está en ejecución. Esta propiedad facilita la reutilización de código y la flexibilidad en el desarrollo de sistemas, ya que permite que el mismo método sea utilizado para diferentes tipos de objetos, adaptándose a las características de cada uno. Además, en algunos lenguajes como Java o C++, el polimorfismo puede incluir la sobrecarga de métodos y operadores, lo que permite usar los mismos operadores o funciones para operar sobre diferentes tipos de datos (Joyanes Aguilar & Zahonero Martínez, 2011).

#### **2.1.12.5. Clase**

Una clase se describe como un tipo abstracto de dato que encapsula tanto atributos como métodos. Los atributos representan las características o propiedades del concepto abstracto, mientras que los métodos se refieren a los servicios o funcionalidades asociados a dicho concepto. En otras palabras, una clase implementa un modelo abstracto de un objeto, agrupando las características y comportamientos que definen ese objeto en un sistema (Flórez Fernández, 2012).

#### **2.1.12.6. Objeto**

(Flórez Fernández, 2012), menciona que “un objeto es la referencia e instancia de una clase. Al crear una referencia se asigna un espacio de memoria dinámica al objeto, pero no es utilizable. Al crear la instancia, el objeto es utilizable” (p. 28).

#### **2.1.13. Arquitectura Monolítica**

Una arquitectura monolítica es un modelo de desarrollo de software tradicional que organiza todas las funciones empresariales en un único código base. Los componentes del sistema están estrechamente interconectados debido a los mecanismos internos de intercambio de datos. Esto hace que los cambios en la arquitectura sean complicados y requieran mucho tiempo, ya que incluso pequeñas modificaciones pueden afectar significativamente a múltiples áreas del código (AWS, 2024).

#### **2.1.14. PHP**

PHP Es un lenguaje de programación interpretado, es decir su código no se compila sino se procesa en un servidor. La característica más importante de PHP es su facilidad de integrarse

con HTML. Es especialmente utilizado en el desarrollo web para añadir dinamismo a los sitios web. (PHP Group, 2024).

#### **2.1.15. JavaScript**

Es un lenguaje de programación interpretado, es decir no requiere ser compilado. Está diseñado para ejecutarse en un navegador. Utilizado principalmente para ejecutar acciones en el lado del cliente, aunque también es posible utilizar JavaScript del lado del servidor. Su principal objetivo es añadir interactividad a las páginas, proporcionando mayor vistosidad. JavaScript está definido como un lenguaje de programación orientada a objetos que se basa en prototipos. Se caracteriza por ser débilmente tipado, no hace falta especificar el tipo de dato que se almacena tal y como sucede con otros lenguajes. Es muy utilizado para recibir y enviar información del servidor junto con otras tecnologías. (Mozilla Developer Network, 2024).

#### **2.1.16. Servidor Apache**

El servidor Apache HTTP es un software de código abierto ampliamente utilizado para ofrecer servicios web bajo el protocolo HTTP. Desde su lanzamiento en 1995, se ha consolidado como una de las plataformas más estables y seguras para la distribución de contenido en línea. Una de sus principales características es su arquitectura modular, esta característica permite ajustar su funcionamiento a distintos entornos y necesidades específicas. Puede ejecutarse tanto en sistemas operativos UNIX como en Windows, lo que ha favorecido su expansión y soporte comunitario. (Apache HTTP Server Project, 2024).

#### **2.1.17. Laravel**

Es un framework de código abierto escrito en PHP, se utiliza para el desarrollo de aplicaciones y servicios web. Tiene por filosofía "desarrollar código PHP de forma elegante y simple". Está diseñado para simplificar tareas comunes en los proyectos web, como el manejo de autenticación, enrutamiento, sesiones y almacenamiento en caché. Laravel ofrece herramientas avanzadas como su ORM Eloquent para la gestión de bases de datos, migraciones, y un potente sistema de plantillas. Este enfoque lo hace adecuado tanto para aplicaciones pequeñas como para sistemas empresariales complejos (Laravel, 2024).

### **2.1.17.1. Rutas y Controladores**

En Laravel, las rutas y controladores trabajan juntos para gestionar las solicitudes HTTP de manera eficiente. Las rutas definen las URLs que el usuario solicita y asignan las acciones correspondientes, como métodos de controladores o funciones anónimas, facilitando la navegación y el enrutamiento de las solicitudes. Las rutas pueden incluir parámetros dinámicos y restricciones para gestionar las entradas de manera más flexible. Los controladores son clases que agrupan la lógica para manejar estas solicitudes. Cada controlador contiene métodos que responden a las rutas, separando la lógica de negocio de la presentación y permitiendo una estructura de código más limpia y escalable. Laravel también incluye controladores de recursos, que proporcionan métodos predefinidos para gestionar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) en los modelos de base de datos, optimizando la interacción con los datos (Laravel, 2024).

### **2.1.17.2. Eloquent ORM (Object-Relational Mapping)**

Eloquent es un sistema de mapa objeto-relacional integrado en Laravel, que permite interactuar con las bases de datos mediante objetos en lugar de escribir consultas SQL manualmente. Esto hace que la manipulación de datos sea más sencilla y coherente con el enfoque orientado a objetos (Laravel, 2024).

### **2.1.17.3. Migraciones**

Las migraciones en Laravel permiten modificar la estructura de la base de datos de manera controlada y versionada. Con esta característica facilita el trabajo en el equipo, asegurando que todos los desarrolladores mantengan la misma estructura de base de datos en diferentes entornos (Laravel, 2024).

### **2.1.17.4. Blade Templating Engine**

Blade es el motor de plantillas de Laravel que permite crear vistas dinámicas utilizando una sintaxis sencilla. Blade permite incluir lógica en las vistas de forma limpia y eficiente, permitiendo el manejo de datos en las páginas web (Laravel, 2024).

#### **2.1.17.5. Artisan CLI**

Artisan es la interfaz de línea de comandos de Laravel, proporciona comandos útiles para automatizar tareas como la creación de controladores y diversas clases, la ejecución de migraciones y la limpieza de la base de datos. Estas funcionalidades hacen que el desarrollo sea productivo y eficiente (Laravel, 2024).

#### **2.1.18. Tailwind CSS**

Tailwind es un framework de CSS que se caracteriza por centrarse en la utilidad antes que en los estilos predeterminados. A diferencia de otros como Bootstrap o Bulma, no incluye componentes ya diseñados, sino que ofrece un conjunto amplio de clases que sirven como base para estructurar y dar estilo a los elementos. Gracias a esto, el desarrollador puede construir interfaces personalizadas de manera rápida y flexible. Su enfoque no impone un diseño específico, lo que permite crear resultados únicos adaptados a las necesidades de cada proyecto. (Tailwind Labs, 2024).

#### **2.1.19. Base de Datos**

Según (Mannino, 2007), las bases de datos contienen datos persistentes, esto significa que los datos se almacenan en sistemas de almacenamiento permanente como discos magnéticos. Estos datos pueden ser consultados y utilizados continuamente por diversas áreas de la organización. Las bases de datos también permiten la compartición de información entre múltiples usuarios, lo que permite su uso simultáneo en distintas funciones dentro de la organización, como la gestión de empleados o el procesamiento de pedidos. Las bases de datos están diseñadas para interrelacionar datos almacenados en diferentes entidades, lo que facilita el procesamiento de información compleja y la toma de decisiones.

Por su parte (Kendall & Kendall, 2011) una base de datos es un sistema estructurado que centraliza la información para su uso compartido de manera eficiente. Está diseñada para permitir el acceso simultáneo de múltiples usuarios y aplicaciones, garantizando la organización, el almacenamiento y la recuperación de datos de forma efectiva. Este enfoque minimiza redundancias y asegura que la información sea accesible y útil para las operaciones de una organización.

### **2.1.20. Sistema de administración de bases de Datos (DBMS)**

(Kendall & Kendall, 2011) definen “Software que organiza los datos en una base de datos; provee la capacidad de almacenamiento, organización y recuperación de los datos” (p. 563).

Un sistema de administración de bases de datos (DBMS) es un conjunto de componentes, diseñados para facilitar la creación, uso y mantenimiento de bases de datos. Los DBMS en sus inicios se enfocaban en ofrecer un almacenamiento eficiente y una recuperación rápida de datos. Con el paso del tiempo y gracias a la demanda del mercado y la innovación en los productos, estos sistemas han evolucionado para proporcionar una variedad más amplia de funcionalidades, como la incorporación, almacenamiento, difusión, mantenimiento, recuperación y formato de datos. Esta evolución ha aumentado la complejidad de los DBMS, haciendo que se requiera un extenso tiempo de estudio y uso para dominar un DBMS específico. Debido a que continúan evolucionando, es importante que los profesionales se mantengan actualizados sobre sus características, ya que cada nueva versión suele integrar mejoras y nuevas funcionalidades (Mannino, 2007).

#### **2.1.20.1. MySQL**

MySQL es un sistema de gestión de bases de datos relacional (RDBMS) de código abierto, que permite almacenar, organizar y administrar grandes volúmenes de información de manera eficiente, mediante tablas relacionadas entre sí, lo que facilita el manejo de los datos de forma rápida y flexible. Se caracteriza por ser una herramienta escalable, confiable y fácil de usar, haciéndola muy utilizada tanto en aplicaciones web como en entornos locales. MySQL trabaja con el lenguaje SQL, permitiendo realizar consultas y manipular los datos de manera sencilla. El sistema puede adaptarse a proyectos de distinto tamaño, desde pequeños desarrollos hasta aplicaciones empresariales de gran escala (MySQL, 2024).

#### **2.1.20.2. Datagrip**

DataGrip es un IDE de administración de bases de datos desarrollada por JetBrains. Permite conectarse, gestionar y trabajar con distintos sistemas de bases de datos de forma práctica y eficiente. Los usuarios pueden crear consultas SQL, editar datos, visualizar esquemas y depurar scripts. Estas funcionalidades facilitan tanto las tareas básicas como otros procesos más complejos. Es compatible con varios motores de bases de datos, entre ellos MySQL,

PostgreSQL, SQL Server y Oracle, entre otros. Tiene por ventaja mejorar el flujo de trabajo gracias a funciones como la introspección y el autocompletado de código, que agilizan la escritura y reducen errores durante el desarrollo (JetBrains, 2024).

#### **2.1.21. Computación en la Nube**

La computación en la nube, también conocida como Cloud Computing, es un modelo de uso de la tecnología que permite acceder a diferentes recursos informáticos a través de internet. De acuerdo con (Joyanes Aguilar, 2012), este enfoque ofrece a las organizaciones la posibilidad de utilizar servicios como almacenamiento, virtualización, centros de datos o software como servicio (SaaS), sin tener que administrar directamente una infraestructura física. Gracias a este modelo, las empresas pueden gestionar su información de manera más flexible y eficiente, ya que la nube permite ajustar los recursos según la demanda y reducir los costos operativos. La computación en la nube ha cambiado la forma en que las organizaciones distribuyen y utilizan la información, aprovechando tecnologías existentes para mejorar su infraestructura tecnológica. Hoy en día la computación en la nube resulta muy importante dentro del entorno digital, porque permite pagar solo por los servicios que se usan y adaptarse fácilmente a los cambios y necesidades de un mercado cada vez más dinámico.

#### **2.1.22. Git**

Git es un sistema de control de versiones de código abierto, se utiliza para llevar el registro de los cambios en los proyectos de desarrollo de software. Se caracteriza por ser rápido, eficiente y por adaptarse a proyectos de cualquier tamaño. Una de sus principales ventajas es que funciona de manera distribuida, esto quiere decir que cada integrante del equipo puede tener una copia completa del repositorio, permitiendo trabajar sin depender siempre de una conexión al servidor. También ofrece la posibilidad de crear ramas locales fácilmente y administrar diferentes versiones del código sin complicaciones. Gracias a estas funciones, Git se ha convertido en una herramienta muy útil para el trabajo colaborativo, ya que permite integrar y combinar aportes de varios desarrolladores sin perder el control de los cambios realizados (Git, 2024).

#### **2.1.23. Azure DevOps**

Azure DevOps es una plataforma creada por Microsoft que reúne varias herramientas pensadas para cubrir todas las etapas del desarrollo de software. Con esta plataforma los equipos

pueden planificar, programar, probar y entregar aplicaciones de una forma más rápida y organizada. Entre sus principales servicios están Azure Boards, que se usa para la gestión ágil de proyectos; Azure Pipelines, que permite automatizar la integración y entrega continua; Azure Repos, donde se guardan los repositorios Git, Azure Test Plans que ayuda a realizar pruebas planificadas o exploratorias; y Azure Artifacts, que se utiliza para manejar paquetes y dependencias. Una de las ventajas de Azure DevOps es que puede trabajar con diferentes lenguajes, plataformas y servicios en la nube. Se puede integrar fácilmente con otras herramientas muy usadas, como GitHub o Kubernetes, lo que mejora la colaboración entre los miembros del equipo y facilita la gestión del proyecto (Microsoft, 2024).

#### **2.1.24. Visual Studio Code**

Visual Studio Code es un editor de código fuente moderno, ligero y personalizable, creado para facilitar el trabajo con distintos lenguajes de programación. Tiene un diseño modular permitiendo a los desarrolladores adaptar el entorno de acuerdo con sus necesidades, mediante la instalación de extensiones. Entre sus principales funciones, se encuentran la depuración de código, resaltado de sintaxis y la integración con sistemas de control de versiones como Git. Es compatible con diversos entornos, tanto en equipos de escritorio como en la web, esto lo convierte en una herramienta flexible y accesible. Visual Studio Code ofrece la posibilidad de incorporar múltiples herramientas adicionales a través de su sistema de extensiones, optimizando así las tareas de codificación, prueba y depuración (Microsoft, 2024).

#### **2.1.25. Pruebas unitarias**

Una prueba unitaria es el proceso en el desarrollo de software en el cual se prueba el funcionamiento o una unidad de código. La unidad de código es la parte más pequeña que se puede probar de un software, puede ser una función u otro método. Esto ayuda a garantizar la calidad del código siendo un proceso integral en el desarrollo de software (AWS, 2025).

##### **2.1.25.1. PUnit**

PHPUnit es un framework de testing o pruebas unitarias para el lenguaje PHP, permite ejecutar tests unitarios de manera sencilla, el framework utiliza assertions para comprobar que el funcionamiento del código es el esperado (phpunit, 2025).

## CAPÍTULO III

### MATERIALES Y MÉTODOS

#### 1.7. Tipo y nivel de investigación

##### 3.1.1. Tipo de investigación

La investigación observacional consiste en recopilar información sin alterar las condiciones del entorno o intervenir en las variables involucradas. Los datos se registran tal como se presentan en su estado natural (Haro Sarango et al., 2024). El estudio es observacional porque no existe intervención sobre la variable gestión de matrícula.

La investigación retrospectiva se enfoca en analizar información ya registrada o eventos previos, utilizando estos datos para identificar patrones o resultados pasados (Haro Sarango et al., 2024). El estudio es retrospectivo porque se emplea registros históricos de matrícula ya existentes en la institución.

Según (Hernández Sampieri et al., 2006), la investigación transversal se lleva a cabo en un único punto en el tiempo, donde se examinan las variables tal como se presentan sin intervenir ni modificar el entorno. El estudio es transversal porque se analiza la variable gestión de matrícula en un momento específico.

El tipo de investigación descriptiva se orienta hacia la identificación detallada de atributos o cualidades de diferentes fenómenos, como personas, grupos o procesos, permitiendo describir su comportamiento o estado en un entorno determinado (Hernández Sampieri et al., 2006). El estudio es descriptivo porque se analiza y describe la variable gestión académica.

##### 3.1.2. Nivel de investigación

Según (Bernal, 2006), el nivel de investigación descriptivo tiene como finalidad principal analizar y registrar las características de un fenómeno o proceso tal y como se presentan, sin intervenir en ellos. Este nivel resulta adecuado para casos como la gestión de matrícula en colegios, ya que facilita la observación y documentación de los procesos existentes, permitiendo describir sus particularidades sin explorar relaciones de causa y efecto.

El nivel de investigación del presente estudio es descriptivo, porque analiza y documenta las características actuales de la gestión de matrícula, describiendo el sistema manual utilizado.

Esto permite identificar sus necesidades y limitaciones. Proporcionando una base sólida para diseñar un prototipo de software que automatice los procesos.

## **1.8. Diseño de investigación**

(Hernández Sampieri et al., 2006) señalan que, en los diseños no experimentales, el investigador no interviene ni modifica las variables estudiadas. En este enfoque, los fenómenos se examinan tal como se presentan de manera natural, permitiendo analizarlos sin la imposición de condiciones o manipulaciones controladas.

El presente estudio es no experimental porque la variable gestión académica se analiza tal como se presenta en su estado actual, sin aplicar ningún tipo de manipulación o experimento.

## **1.9. Variables**

### **3.1.3. Definición conceptual de las variables**

#### **Variable de estudio**

**Gestión de matrícula:** Es el proceso administrativo integral que se encarga de registrar, organizar y controlar la inscripción de estudiantes en una institución educativa. Incluyendo el seguimiento de pago de pensiones.

#### **Variables descriptivas**

**Gestión académica:** Es el conjunto de actividades, procedimientos y herramientas que permiten planificar, organizar y controlar los aspectos relacionados con la estructura educativa de una institución, como los niveles, grados, secciones y asignaciones de estudiantes. En el contexto del sistema propuesto, la gestión académica comprende la administración de secciones y la generación de reportes que facilitan la coordinación interna de los procesos académicos.

**Gestión de información del estudiante:** Se refiere al proceso sistemático de recopilación, almacenamiento, actualización y consulta de los datos personales y académicos de los estudiantes.

**Control de pagos:** Es el proceso mediante el cual se registran, verifican y supervisan las transacciones económicas relacionadas con las pensiones y otros conceptos de pago de los estudiantes. Implica el seguimiento de los pagos realizados y la aplicación de moras.

### 3.1.4. Operacionalización de las variables

(Hernández Sampieri et al., 2006) definen “Actividades u operaciones que deben realizarse para medir una variable” (p. 146).

**Tabla 1.**

*Operacionalización de la variable.*

Variable	Dimensiones	Indicador	Unidad de medida
Gestión de matrícula	Gestión académica	Tiempo promedio para la gestión de académica.	minutos.
	Gestión de información del estudiante	Tiempo promedio para la gestión de información del estudiante.	minutos.
	Control de pagos	Tiempo promedio para el control de pagos.	minutos

*Nota.* La tabla muestra la operacionalización de la variable.

### 1.10. Población y muestra

#### 3.1.5. Población

Según (Hernández Sampieri et al., 2006), la población se entiende como el conjunto completo de elementos que comparten ciertas características específicas que son objeto de estudio en una investigación, es decir, es el grupo del que se pretende obtener datos o realizar un análisis.

La población está compuesta por los procesos gestión académica, gestión de información del estudiante y control de pago de pensiones en la Institución Educativa "Nuestra Señora De Guadalupe" durante el año 2025.

### **3.1.6. Muestra**

Según (Hernández Sampieri et al., 2006), cuando la población es pequeña o se tiene acceso a todos los elementos de la población, es posible utilizar un muestreo censal, en el cual se estudian todos los elementos de la población sin necesidad de realizar una selección muestral.

Dado que la población en esta investigación está conformada por un total de 3 procesos (gestión académica, gestión de información del estudiante y control de pago de pensiones), y considerando que se tiene acceso directo a todos los procesos involucrados, se optará por un muestreo censal.

## **1.11. Técnicas e instrumentos de investigación**

### **3.1.7. Técnicas**

Según (Hernández Sampieri et al., 2006) una técnica de investigación es un conjunto de procedimientos y métodos que se aplican de manera sistemática para recoger, analizar y procesar la información necesaria para resolver un problema de investigación. Las técnicas varían según el tipo de estudio, pero su objetivo común es transformar los datos obtenidos en conocimientos útiles y válidos para las conclusiones de la investigación. Las técnicas se seleccionan de acuerdo con los objetivos de la investigación y el tipo de datos que se requieren.

Análisis documental, permite obtener y sistematizar información de fuentes secundarias, como registros de matrícula, informes institucionales y otros documentos relacionados con la gestión académica y administrativa.

Entrevista, Esta técnica busca establecer una comunicación directa con las personas involucradas que son la fuente de información.

Encuesta, Es una técnica de recolección de información que consiste en la aplicación de un cuestionario que contiene una serie de preguntas estructuradas.

### **3.1.8. Instrumentos**

La ficha de registro permitió obtener y sistematizar información de fuentes secundarias, como informes institucionales y otros documentos relacionados con la gestión académica.

Guía de entrevista, este instrumento permitió recopilar información para entender el proceso y lógica del negocio, como también permitió identificar diversos problemas que existía en los procesos.

Cuestionario, este instrumento permitió validar la funcionalidad de la aplicación por un usuario final.

## 1.12. Procedimientos

### 3.1.9. Técnicas de Procesamientos de datos

Para procesar las fichas de registro y guía de entrevista, que nos indican los requerimientos de la aplicación web y funcionalidad del software, se realizará mediante las distintas fases del marco de trabajo Scrum.

### 3.1.10. Herramientas para el tratamiento de datos

Los datos e información obtenida a través de los instrumentos fueron tratados utilizando las siguientes herramientas que se describen en la tabla.

**Tabla 2.**

*Herramientas utilizadas para el tratamiento de datos.*

Herramienta	Descripción
PHP	Lenguaje de scripting de código abierto del lado del servidor, diseñado para el desarrollo web, que ejecuta código en el servidor para generar contenido dinámico y enviarlo al navegador del usuario.
Laravel	Framework PHP de código abierto y gratuito diseñado para el desarrollo rápido y organizado de aplicaciones web, utilizando el patrón de arquitectura Modelo-Vista-Controlador (MVC).
Tailwind CSS	Framework CSS basado en clases de utilidad que permite crear interfaces de usuario web personalizadas directamente en el HTML.
MySQL	MySQL es un sistema de gestión de bases de datos relacionales (RDBMS) de código abierto, ampliamente utilizado para almacenar y administrar datos en un formato de tablas con filas y columnas, utilizando el lenguaje SQL.
DataGrip	DataGrip es un Entorno de Desarrollo Integrado (IDE) de JetBrains diseñado para desarrolladores SQL, que permite consultar, crear y administrar bases

	de datos locales, de servidor o en la nube, como PostgreSQL, MySQL y SQL Server.
Visual Studio Code	Visual Studio Code es un editor de código fuente ligero pero eficaz que se ejecuta en el escritorio y está disponible para Windows, macOS y Linux.
Git	Git es un sistema de control de versiones distribuido de código abierto que permite a los desarrolladores rastrear cambios en el código fuente y colaborar en proyectos de forma eficiente.
Azure DevOps	Plataforma en la nube de Microsoft Learn que proporciona herramientas para el desarrollo de software.

---

*Nota.* Esta tabla se detalla las herramientas utilizadas para el desarrollo de la aplicación web.

### **3.1.11. Técnicas de análisis de datos**

Los datos e información recolectada a través de la ficha de registro y guía de entrevista permitieron analizar y comprender los distintos exigencias y características de la aplicación. Esto permitió definir los distintos requerimientos funcionales para posteriormente iniciar con el desarrollo bajo las fases del marco de trabajo Scrum.

### **3.1.12. Técnicas para aplicar el marco de trabajo Scrum**

El marco de trabajo Scrum está orientada al desarrollo incremental e iterativo de software, promueve la entrega continua de valor mediante la colaboración y la adaptación contante a los cambios.

Scrum se estructura a través de roles, eventos y artefactos, las cuales permiten organizar el trabajo y controlar el avance del proyecto de manera transparente y medible. Los principales roles definidos son el Product Owner, el Scrum Master y el Development Team, quienes interactúan de forma iterativa durante cada Sprint, buscando alcanzar los objetivos establecidos en el Product Backlog.

Cada Sprint representa un ciclo de trabajo con una duración determinada, en el cual se desarrollan un conjunto de historias de usuario seleccionadas según su prioridad y valor de negocio. Al término de cada iteración, se realiza una revisión de los resultados obtenidos, permitiendo la retroalimentación y mejora continua del proceso.

**Tabla 3.***Entregables de Scrum.*

Tarea	Entregable	Técnica	Responsable
Identificación de requisitos funcionales y no funcionales.	Lista inicial de requisitos del sistema.	Recopilar requerimientos a través de la entrevista y análisis documental.	Product Owner
Elaboración del Product Backlog	Product Backlog	Revisión de las historias de usuario, elaborar el producto backlog y priorización con Planning Poker.	Scrum Team
Planificación del Sprint	Sprint Backlog	Sprint Planning	Scrum Master
Evaluación de resultados del Sprint	Incremento funcional	Sprint Review	Scrum Team y Product Owner
Análisis de mejoras y ajustes al proceso	Registro de lecciones aprendidas y plan de mejora.	Sprint Retrospective	Scrum Master
Control del rendimiento y avance general	Gráfico de avance real vs. esperado del Sprint	Burn Down Chart	Product owner

*Nota.* La tabla muestra los entregables necesarios para el desarrollo de la aplicación web.

## CAPÍTULO IV

### RESULTADOS Y DISCUSIÓN

#### 1.13. Entregables

##### 4.1.1. Definiendo roles de trabajo

El marco de trabajo Scrum contempla tres roles principales: Product Owner, Scrum Master y Development Team. Durante el desarrollo del proyecto, el rol de Product Owner fue asumido por el director de la institución Lic. Lucho Alfredo Yupanqui Ataucusi, quien brindó la orientación necesaria respecto a los requerimientos funcionales y prioridades del sistema. Por otro lado, los roles de Scrum Master y Development Team fueron desempeñados por el tesista.

**Tabla 4.**

*Roles de Scrum*

Rol	Responsabilidades	Responsable
Product Owner	Definir los requerimientos funcionales del sistema, priorizar el product backlog, establecer los objetivos de cada sprint y validar las funcionalidades desarrolladas.	Lic. Lucho Alfredo Yupanqui Ataucusi
Scrum Master	Planificar las iteraciones, controlar el cumplimiento de las tareas, asegurar la correcta aplicación de la metodología Scrum y gestionar posibles incidencias durante el desarrollo.	Bach: Víctor López Amiquero
Development Team	Analizar, diseñar, codificar, probar e implementar los módulos del sistema web para la gestión de matrícula, asegurando su correcto funcionamiento y calidad técnica.	Bach: Víctor López Amiquero

*Nota.* La tabla presenta los roles scrum, así como las responsabilidades asumidas por cada integrante del equipo.

#### 4.1.2. Definiendo el product backlog

**Tabla 5.**

*Product Backlog*

ID de la historia	Descripción	Módulo	Estado	Prioridad	Esfuerzo
HU-SEG-001	Como usuario del sistema, quiero iniciar sesión con mi usuario y contraseña para acceder a mis funciones.	Seguridad	Propuesto	Alta	8
HU-SEG-002	Como administrador, quiero crear y administrar roles configurando sus permisos, para controlar qué acciones pueden realizar los usuarios en el sistema.	Seguridad	Propuesto	Alta	13
HU-SEG-003	Como administrador, quiero gestionar usuarios mediante un formulario por pasos, que permita registrar sus datos personales, documentos, contactos y asignarles un rol dentro del sistema, para asegurar que accedan únicamente con los permisos adecuados.	Seguridad	Propuesto	Alta	8
HU-SEG-004	Como usuario con rol de soporte, quiero visualizar solo los menús y funciones que me permitan mis permisos para evitar acceder a áreas restringidas.	Seguridad	Propuesto	Alta	3
HU-CONF-001	Como administrador, quiero registrar ciclos académicos mediante un proceso por pasos, para organizar la matrícula y su gestión financiera.	Configuración	Propuesto	Alta	5

HU-CONF-002	Como administrador, quiero gestionar secciones para organizar a los estudiantes dentro de las aulas.	Configuración	Propuesto	Alta	2
HU-CONF-003	Como administrador, quiero gestionar niveles para organizar a los estudiantes por niveles escolares (inicial, primaria y secundaria).	Configuración	Propuesto	Alta	2
HU-CONF-004	Como administrador, quiero gestionar grados para organizar a los estudiantes dentro de la institución.	Configuración	Propuesto	Alta	3
HU-CONF-005	Como administrador, quiero configurar tipos de matrícula (regular, extraordinaria, traslado) para adaptarme a las necesidades de la institución.	Configuración	Propuesto	Media	2
HU-CONF-006	Como administrador, quiero registrar descuentos (hermano, puntualidad, beca) para aplicarlos a los pagos de matrícula y pensiones.	Configuración	Propuesto	Media	2
HU-CONF-007	Como administrador, quiero configurar tipos de operación (matrícula, pago de pensión) registrar correctamente las transacciones financieras.	Configuración	Propuesto	Media	2
HU-CONF-008	Como administrador, quiero configurar métodos de pago (efectivo, tarjeta, transferencia) para registrar correctamente las transacciones financieras.	Configuración	Propuesto	Media	2
HU-OPER-001	Como secretaria, quiero registrar la matrícula de un estudiante mediante un proceso por pasos, para almacenar su información personal, académica y de pago.	Operación	Propuesto	Alta	13

HU-OPER-002	Como secretaria, quiero editar los datos de una matrícula ya registrada mediante un proceso similar al registro, para corregir errores o actualizar información del estudiante, contacto o datos financieros.	Operación	Propuesto	Alta	8
HU-OPER-003	Como secretaria, quiero registrar el pago de pensiones de un estudiante mediante un proceso por pasos (búsqueda, selección de pensiones y facturación), para llevar un control financiero actualizado y confiable.	Operación	Propuesto	Alta	13
HU-OPER-004	Como administrador, quiero anular operaciones registradas en el sistema (matrículas o pagos de pensiones) en casos justificados, de modo que se conserve el historial, pero no se consideren como activas en los reportes.	Operación	Propuesto	Alta	5
HU-OPER-005	Como administrador, quiero que el sistema calcule de manera automática la mora diaria en caso de retraso en el pago de pensiones, en base a la configuración financiera del ciclo académico, para reflejar correctamente la deuda del estudiante y mantener información precisa para la gestión de cobros.	Operación	Propuesto	Alta	8
HU-ACA-001	Como director, quiero visualizar un listado de estudiantes matriculados para obtener información general del ciclo académico.	Académico	Propuesto	Media	3

HU-ACA-002	Como secretaria, quiero filtrar estudiantes por ciclo académico para obtener información de un año específico.	Académico	Propuesto	Media	2
HU-ACA-003	Como administrador, quiero visualizar estudiantes con pensiones vencidas para realizar el seguimiento de cobros.	Académico	Propuesto	Media	5

---

*Nota.* La tabla presenta el product backlog el cual contiene todas las historias de usuario de los módulos: seguridad, configuración, operación y académico.

## 1.14. Sprint 0

### 4.1.3. Sprint planning

El objetivo principal del Sprint 0 fue establecer el entorno de desarrollo, la estructura base del proyecto y las herramientas necesarias para iniciar el proceso de implementación ágil del sistema.

**Tabla 6.**

*Sprint planning del sprint 0.*

Reunión de planificación	
Fecha	26/05/2025
Participantes	<ul style="list-style-type: none"> <li>• Product owner</li> <li>• Scrum team</li> <li>• Scrum master</li> </ul>
Artefactos	<ul style="list-style-type: none"> <li>• Product backlog</li> <li>• Backlog de sprint</li> </ul>
Agenda	<ul style="list-style-type: none"> <li>• Definición del entorno tecnológico a utilizar (framework, base de datos, herramientas).</li> <li>• Configuración del repositorio de control de versiones.</li> <li>• Creación del proyecto base y estructura inicial.</li> </ul>
Objetivos	<ul style="list-style-type: none"> <li>• Configurar el entorno de desarrollo (Laravel, MySQL, Git, VS Code).</li> <li>• Crear la estructura base del proyecto con el patrón MVC.</li> <li>• Conectar el sistema a la base de datos.</li> <li>• Dejar operativo el entorno para iniciar el Sprint 1.</li> </ul>

*Nota.* La tabla resume la reunión de planificación para el desarrollo del sprint 0.

#### **4.1.4. Sprint backlog**

El Sprint 0 tuvo como finalidad preparar el entorno de trabajo y establecer la arquitectura base del proyecto. Las tareas realizadas permitieron contar con un entorno configurado y un repositorio operativo, asegurando las condiciones necesarias para iniciar el desarrollo iterativo de funcionalidades a partir del Sprint 1.

**Tabla 7.***Sprint backlog del sprint 0*

ID de la tarea	Descripción	Sprint	Prioridad	Esfuerzo
T0-01	Instalar y configurar el entorno de desarrollo (PHP, Composer, Laravel, MySQL, VS Code).	0	Alta	2
T0-02	Crear el repositorio de control de versiones en Azure DevOps y definir ramas principales.	0	Alta	2
T0-03	Generar la estructura base del proyecto Laravel con el patrón MVC.	0	Alta	2
T0-04	Configurar el archivo .env y establecer la conexión con la base de datos MySQL.	0	Alta	2
T0-05	Diseñar la arquitectura general del sistema y definir los módulos principales.	0	Alta	2
T0-06	Crear la base de datos inicial	0	Alta	2
T0-07	Instalar el paquete de autenticación Laravel Jetstream	0	Alta	2
T0-08	Definir el flujo de trabajo y políticas de commits (frecuencia, nomenclatura).	0	Alta	2
T0-09	Documentar los pasos de instalación y configuración del entorno.	0	Alta	2

*Nota.* La tabla muestra las tareas planificadas del sprint backlog correspondiente al sprint 0.

#### 4.1.5. Arquitectura de la aplicación

La aplicación tiene una arquitectura monolítica, es decir todos los componentes de la aplicación se agrupa en un código base, incluyendo la vista, lógica de negocio y el acceso a la base de datos.

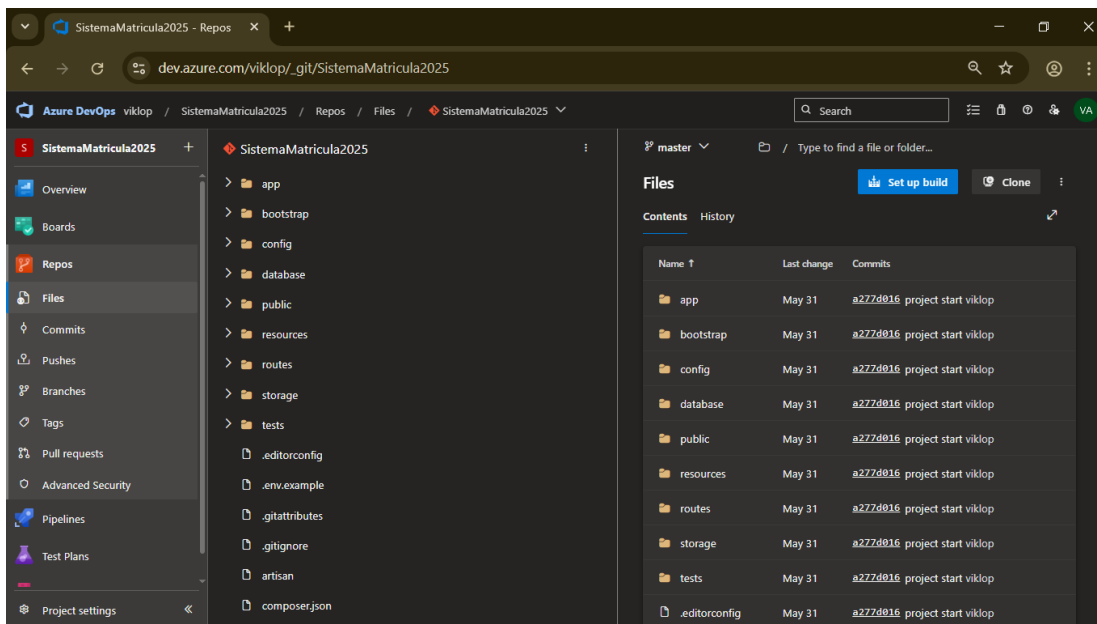
Además, al trabajar con el framework Laravel este sigue un patrón de diseño modelo-vista-controlador (MVC).

#### 4.1.6. Resultados del sprint

Durante el sprint 0 se logró la instalación y configuración del entorno de desarrollo, la creación del repositorio de control de versiones en Azure DevOps, la generación del proyecto base en Laravel y la conexión con la base de datos MySQL.

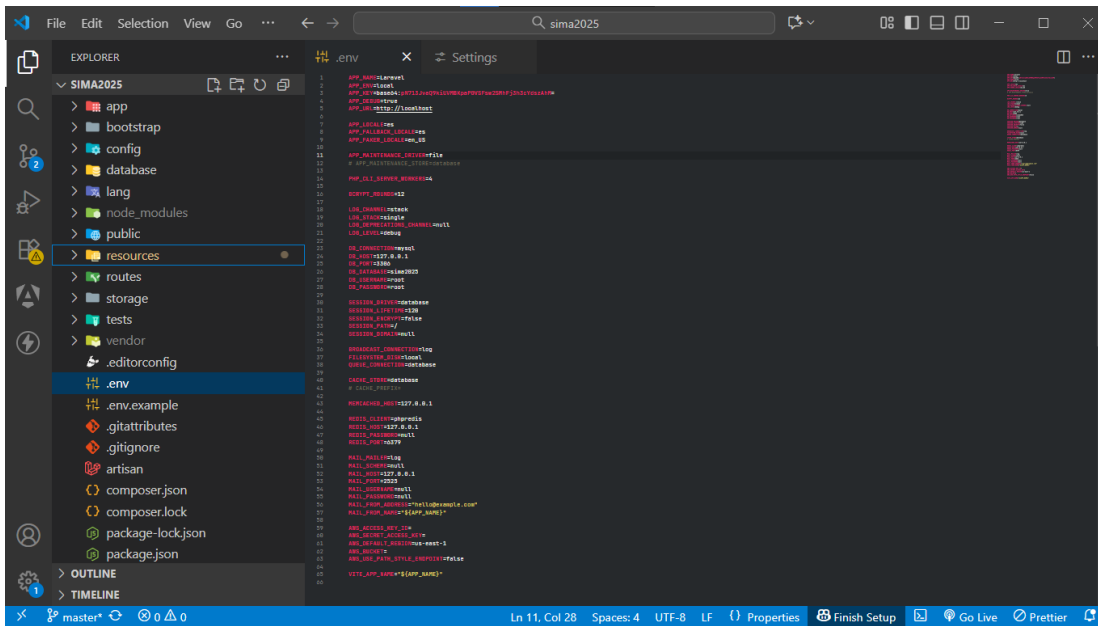
### Figura 8.

*Repositorio de la aplicación configurado en Azure DevOps.*



## Figura 9.

*Estructura inicial del proyecto.*



*Nota.* En la figura se observa la estructura de carpetas y archivos del proyecto base del framework Laravel.

## **1.15. Sprint 1**

### **4.1.7. Sprint planning**

El sprint planning tuvo como objetivo implementar el módulo de seguridad del sistema, dado que constituye la base funcional sobre el cual se desarrollan los demás módulos (configuración, operación y académico). En esta etapa se consideró fundamental garantizar que los accesos al sistema estén correctamente restringidos y gestionados.

Para la planificación, se revisó el product backlog y se seleccionaron las historias de usuario HU-SEG-001, HU-SEG-002, HU-SEG-003 y HU-SEG-004.

La asignación de la prioridad y esfuerzo de las historias de usuario del módulo seguridad se realizó considerando el impacto funcional, los riesgos asociados a la seguridad del sistema y el valor entregado al usuario final. En cuanto a la prioridad, todas las historias del módulo seguridad fueron clasificadas como altas, debido a que constituyen componentes esenciales para garantizar el control de acceso, la autenticación y la administración de permisos dentro de la aplicación. Estas funcionalidades son críticas, ya que sin ellas no sería posible el ingreso seguro, la gestión de usuarios o la restricción adecuada de operaciones según roles definidos.

La estimación de esfuerzo se efectuó mediante la técnica de planning poker, basada en la serie de Fibonacci, esta técnica permitió consensuar la complejidad de cada historia considerando factores como la cantidad de trabajo, el nivel de incertidumbre técnica y las dependencias entre funcionalidades. De esta manera, las historias con mayor complejidad lógica y transversalidad, como la gestión de roles y permisos (HU-SEG-002), recibieron una estimación de 13 puntos, aquellas con un nivel medio de complejidad, como inicio de sesión (HU0SEG-001) y gestión de usuarios (HU-SEG-003), fueron valoradas con 8 puntos y las de menor complejidad, que aplican lógicas ya implementadas, como restricciones de acceso (HU-SEG-004), obtuvieron 5 puntos. Esta evaluación asegura una planificación equilibrada y coherente con la carga real de trabajo durante la iteración correspondiente.

**Tabla 8.***Sprint planning del sprint 1*

Reunión de planificación de sprint 1	
Fecha	06/06/2025
Participantes	<ul style="list-style-type: none"> <li>• Product owner</li> <li>• Scrum team</li> <li>• Scrum master</li> </ul>
Artefactos	<ul style="list-style-type: none"> <li>• Product backlog</li> <li>• Backlog de sprint</li> </ul>
Agenda	<ul style="list-style-type: none"> <li>• Revisión del producto backlog y definición de objetivos del sprint.</li> <li>• Selección de historias de prioridad alta.</li> <li>• Estimación de esfuerzo con planning poker.</li> <li>• Definición de tareas para cada historia seleccionada.</li> </ul>
Objetivos	<ul style="list-style-type: none"> <li>• Desarrollar el módulo de seguridad.</li> <li>• Completar las historias de usuario HU-SEG-001, HU-SEG-002, HU-SEG-003 Y HU-SEG-004.</li> </ul>

*Nota.* La tabla resume la reunión de planificación para el desarrollo del sprint 1.

#### 4.1.8. Sprint backlog

**Tabla 9.**

*Sprint backlog del sprint 1*

ID de la historia	Descripción	Módulo	Sprint	Prioridad	Esfuerzo
HU-SEG-001	Como usuario del sistema, quiero iniciar sesión con mi usuario y contraseña para acceder a mis funciones.	Seguridad	1	Alta	8
HU-SEG-002	Como administrador, quiero crear y administrar roles configurando sus permisos, para controlar qué acciones pueden realizar los usuarios en el sistema.	Seguridad	1	Alta	13
HU-SEG-003	Como administrador, quiero gestionar usuarios mediante un formulario por pasos, que permita registrar sus datos personales, documentos, contactos y asignarles un rol dentro del sistema, para asegurar que accedan únicamente con los permisos adecuados.	Seguridad	1	Alta	8
HU-SEG-004	Como usuario con rol de soporte, quiero visualizar solo los menús y funciones que me permitan mis permisos para evitar acceder a áreas restringidas.	Seguridad	1	Alta	5

*Nota.* La tabla muestra el sprint backlog del sprint 1, agrupa las historias de usuario del módulo seguridad.

#### 4.1.9. Historias de usuario

**Tabla 10.**

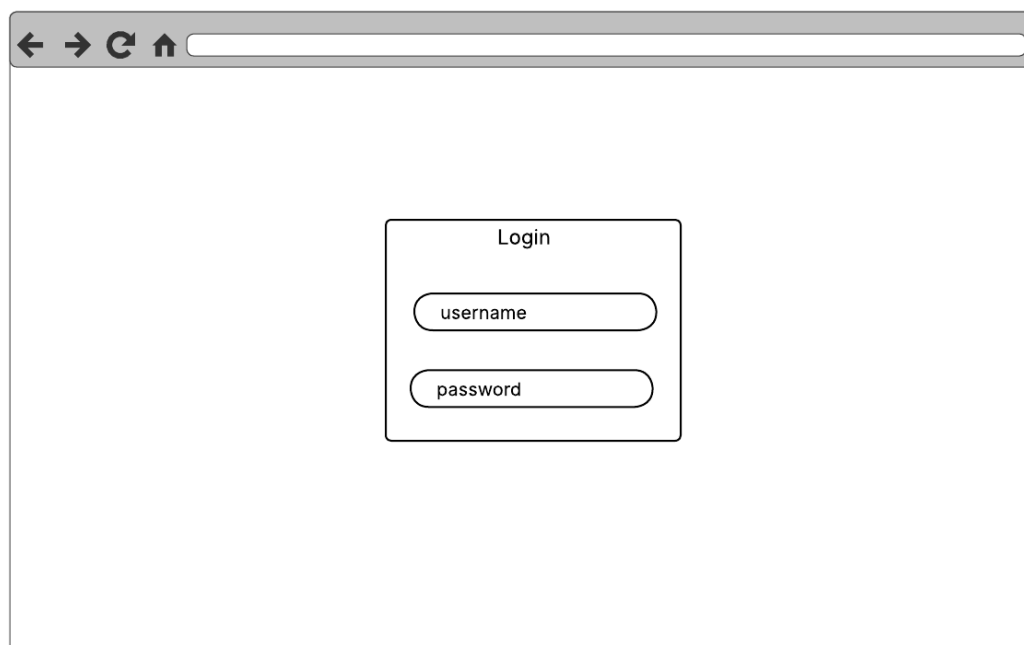
*HU-SEG-001 Inicio de sesión*

Historia de usuario	
Código: HU-SEG-001	Nombre: Inicio de sesión
Módulo: Seguridad	
Responsable: Equipo de desarrollo	Iteración: sprint 1
Prioridad: Alta	Esfuerzo: 8
Descripción	
Como usuario del sistema, quiero iniciar sesión con mi usuario y contraseña para acceder a mis funciones.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se muestra un formulario con usuario y contraseña.</li> <li>• Si las credenciales son válidas, se accede al sistema.</li> <li>• Si las credenciales son incorrectas, se muestra un mensaje de error.</li> <li>• El usuario es redirigido a su vista personalizada según su rol.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-SEG-001.

**Figura 10.**

*Wireframe de la vista de inicio de sesión.*



*Nota.* La figura muestra el bosquejo de la vista de inicio de sesión de la aplicación.

**Tabla 11.**

*HU-SEG-002 Gestión de roles y permisos*

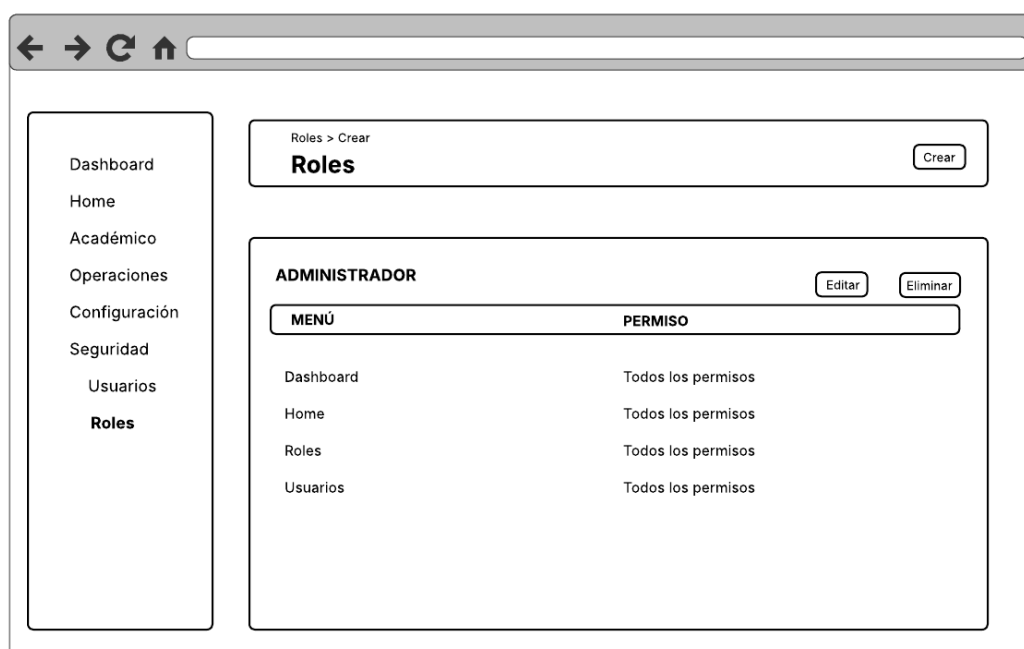
Historia de usuario	
Código: HU-SEG-002	Nombre: Gestión de roles y permisos
Módulo: Seguridad	
Responsable: Equipo de desarrollo	Iteración: sprint 1
Prioridad: Alta	Esfuerzo: 13
Descripción	
Como administrador, quiero crear y administrar roles configurando sus permisos, para controlar qué acciones pueden realizar los usuarios en el sistema.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se puede crear un nuevo rol.</li> <li>• Se valida que no haya roles duplicados.</li> <li>• Se puede editar el nombre y los permisos de un rol existente.</li> </ul>	

- No se puede eliminar un rol si tiene usuarios asignados.
- Se pueden asignar permisos sobre menús: Todos, Solo lectura, Solo crea, Crea/lee/edita, Sin acceso.
- Los permisos definen automáticamente qué botones o acciones aparecen en la interfaz.
- Un rol sin acceso no puede visualizar el menú.

*Nota.* La tabla detalla la historia de usuario HU-SEG-002.

### Figura 11.

*Wireframe de la vista listado de roles.*



*Nota.* La figura muestra el bosquejo de la vista principal del menú roles.

**Figura 12.**

*Wireframe de la vista de creación de roles.*

*Nota.* La figura muestra el bosquejo de la vista utilizada para la creación de roles.

**Tabla 12.**

*HU-SEG-003 Gestión de usuarios*

Historia de usuario	
Código: HU-SEG-003	Nombre: Gestión de usuarios
Módulo: Seguridad	
Responsable: Equipo de desarrollo	Iteración: sprint 1
Prioridad: Alta	Esfuerzo: 8
Descripción	
Como administrador, quiero gestionar usuarios mediante un formulario por pasos, que permita registrar sus datos personales, documentos, contactos y asignarles un rol dentro del sistema, para asegurar que accedan únicamente con los permisos adecuados.	
Criterios de aceptación	

- El sistema permite registrar un nuevo usuario mediante un proceso por pasos:
    1. Datos personales
    2. Documento de identidad
    3. Contacto del usuario
    4. Contacto de emergencia
    5. Asignación de rol
  - Se puede editar y actualizar los datos del usuario en cualquier momento.
  - Se puede cambiar el rol asignado a un usuario.
  - Se listan todos los usuarios registrados mostrando sus datos básicos y rol actual.
  - El sistema permite eliminar un usuario siempre que no tenga operaciones críticas asociadas.
  - La asignación de rol impacta inmediatamente en los menús y permisos visibles del usuario.
- 

*Nota.* La tabla detalla la historia de usuario HU-SEG-003.

**Figura 13.**

Wireframe de la vista de listado de usuarios.

Roles > Lista Crear

**Usuarios**

NOMBRE	APELLIDO PATERNO	APELLIDO MATERNO	ROL	DIRECCIÓN	TIPO DOC.	DOCUMENTO	ACCIONES
Lorem	Lorem	Lorem	Soporte	av. vegas	DNI	85421021	
Lorem	Lorem	Lorem	Soporte	av. vegas	DNI	85421021	
Lorem	Lorem	Lorem	Soporte	av. vegas	DNI	85421021	
Lorem	Lorem	Lorem	Soporte	av. vegas	DNI	85421021	
Lorem	Lorem	Lorem	Soporte	av. vegas	DNI	85421021	
Lorem	Lorem	Lorem	Soporte	av. vegas	DNI	85421021	

*Nota.* La figura muestra el bosquejo de la vista principal del menú de usuarios.

**Figura 14.**

Wireframe de la vista de creación de usuarios - paso 1: Datos personales

Usuarios > crear

**Usuarios**

**Datos personales**

Sexo  
 Masculino  Femenino

Nombre

Apellido paterno

Apellido materno

Dirección

Siguiendo

*Nota.* La figura muestra el bosquejo correspondiente al paso 1 del proceso de creación de usuarios, donde se registra los datos personales.

**Figura 15.**

Wireframe de la vista de creación de usuarios - paso 2: Documento.

The wireframe shows a web browser window with a navigation bar at the top containing back, forward, refresh, and home icons. On the left side, there is a vertical sidebar menu with the following items: Dashboard, Home, Académico, Operación, Configuración, Seguridad, and **Usuarios**. The main content area is titled "Usuarios > crear" and "Usuarios". Below this, there is a section titled "Documento" with two input fields: "Tipo de documento" (containing "--Seleccione--") and "Número". A "Siguiete" button is located at the bottom right of the form area.

*Nota.* La figura muestra el bosquejo correspondiente al paso 2 del proceso de creación de usuarios, donde se registra los datos del documento de identidad.

**Figura 16.**

Wireframe de la vista de creación de usuarios - paso 3: Contacto de emergencia.

The wireframe shows a web browser window with a navigation bar at the top containing back, forward, refresh, and home icons. On the left side, there is a vertical sidebar menu with the following items: Dashboard, Home, Académico, Operación, Configuración, Seguridad, and **Usuarios**. The main content area is titled "Usuarios > crear" and "Usuarios". Below this, there is a section titled "Contacto de emergencia" with four input fields: "Nombre de contacto", "Parentesco" (containing "--Seleccione--"), "Celular", and "Teléfono". There is also an "Email" input field. "Atrás" and "Siguiete" buttons are located at the bottom left and bottom right of the form area, respectively.

*Nota.* La figura muestra el bosquejo correspondiente al paso 3 del proceso de creación de usuarios, donde se registra los datos del contacto de emergencia.

**Figura 17.**

Wireframe de la vista de creación de usuarios - paso 4: Contacto.

The wireframe shows a web browser window with a navigation menu on the left and a main content area. The navigation menu includes: Dashboard, Home, Académico, Operación, Configuración, Seguridad, and **Usuarios**. The main content area has a breadcrumb trail 'Usuarios > crear' and a title 'Usuarios'. Below this is a section titled 'Contacto' containing three input fields: 'Celular', 'Teléfono', and 'Email'. At the bottom of this section are two buttons: 'Atrás' and 'Siguiete'.

*Nota.* La figura muestra el bosquejo corresponde al paso 4 del proceso de creación de usuarios, donde se registra los datos de contacto.

**Figura 18.**

Wireframe de la vista de creación de usuarios - paso 5: Asignación de rol.

The wireframe shows a web browser window with a navigation menu on the left and a main content area. The navigation menu includes: Dashboard, Home, Académico, Operación, Configuración, Seguridad, and **Usuarios**. The main content area has a breadcrumb trail 'Usuarios > crear' and a title 'Usuarios'. Below this is a section titled 'Asignar rol' containing one dropdown menu labeled 'Rol' with the text '--Seleccione--'. At the bottom of this section are two buttons: 'Atrás' and 'Crear'.

*Nota.* La figura muestra el bosquejo correspondiente al paso 5 del proceso de creación de usuarios, donde se asigna un rol al usuario.

**Tabla 13.***HU-SEG-004 Restricciones de acceso en el sistema*

Historia de usuario	
Código: HU-SEG-004	Nombre: Restricciones de acceso en el sistema
Módulo: Seguridad	
Responsable: Equipo de desarrollo	Iteración: sprint 1
Prioridad: Alta	Esfuerzo: 5
Descripción	
Como <i>usuario con rol de soporte</i> , quiero visualizar solo los menús y funciones que me permitan mis permisos para evitar acceder a áreas restringidas.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Si un rol tiene “solo lectura”, los botones de crear/editar se ocultan.</li> <li>• Si un rol tiene “sin acceso”, el menú no se muestra en la interfaz.</li> <li>• Si un rol tiene “todos los permisos”, accede a todas las funciones de ese menú.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-SEG-004.

#### **4.1.10. Taskboard**

Para la planificación del sprint 1 se definió una jornada de trabajo de lunes a viernes con una dedicación de 4 horas diarias. Con base de esta capacidad, las tareas del taskboard se calendarizaron en función de su estimación en horas, distribuyéndose de manera continua en los días hábiles.

**Tabla 14.***Estimación de trabajo en horas del sprint 1*

Equipo Scrum	Horas de trabajo por día	Horas de Trabajo por semana	Semanas de trabajo por mes	Total de horas	Total de días laborales para el proyecto
Tesista	4	20	4	80	20
Total de días disponible para el proyecto					20

*Nota.* La tabla muestra la estimación de horas disponibles y días laborables para el desarrollo del sprint 1.

**Tabla 15.***Taskboard del sprint 1*

Taskboard del sprint 1								
Inicio: lunes 09 de junio de 2025						Duración: 86 horas		
ID HU	ID TA	Tarea	Responsable	Estado	Horas	Inicio	Fin	
HU-SEG-001	T-001	Diseñar y modelar la base de datos del módulo de seguridad.	Equipo de desarrollo	Hecho	4	09/06/2025	09/06/2025	
	T-002	Diseñar y modelar la base de datos del módulo de configuración.	Equipo de desarrollo	Hecho	4	10/06/2025	10/06/2025	
	T-003	Diseñar y modelar la base de datos del módulo de operaciones.	Equipo de desarrollo	Hecho	4	11/06/2025	11/06/2025	
	T-004	Diseñar y modelar la base de datos del módulo de académico.	Equipo de desarrollo	Hecho	4	12/06/2025	12/06/2025	
	T-005	Configurar el sistema de autenticación de usuarios.	Equipo de desarrollo	Hecho	3	13/06/2025	13/06/2025	
	T-005	Implementar control de acceso por roles.	Equipo de desarrollo	Hecho	4	16/06/2025	16/06/2025	
	T-006	Diseñar el maquetado principal de la aplicación.	Equipo de desarrollo	Hecho	4	17/06/2025	17/06/2025	
	T-007	Diseñar el menú dinámico o sidebar de la aplicación.	Equipo de desarrollo	Hecho	4	18/06/2025	18/06/2025	

HU-SEG-002	T-008	Diseñar la interfaz para la gestión de roles y permisos	Equipo de desarrollo	Hecho	3	19/06/2025	19/06/2025
	T-009	Implementar el registro de un nuevo rol con validación de duplicados.	Equipo de desarrollo	Hecho	4	20/06/2025	20/06/2025
	T-010	Configurar la edición de roles existentes, incluyendo la modificación de permisos asociados.	Equipo de desarrollo	Hecho	3	23/06/2025	23/06/2025
	T-010	Implementar la eliminación de roles.	Equipo de desarrollo	Hecho	3	24/06/2025	24/06/2025
	T-011	Configurar permisos por rol en los menús del sistema (Todos, Solo lectura, Solo crea).	Equipo de desarrollo	Hecho	4	25/06/2025	25/06/2025
	T-012	Configurar permisos por rol en los menús del sistema (Crea/lee/edita, Sin acceso).	Equipo de desarrollo	Hecho	4	26/06/2025	26/06/2025
	T-013	Aplicar restricciones dinámicas en la interfaz según los permisos definidos.	Equipo de desarrollo	Hecho	4	27/06/2025	27/07/2025
	T-014	Realizar pruebas de aceptación.	Equipo de desarrollo	Hecho	2	27/06/2025	27//06/2025
HU-SEG-003	T-015	Diseñar la interfaz de registro de usuarios en formato de pasos.	Equipo de desarrollo	Hecho	2	27/07/2025	27/07/2025

	T-016	Implementar el registro de datos personales, documentos, contactos y rol.	Equipo de desarrollo	Hecho	4	30/06/2025	30/06/2025
	T-017	Implementar la edición y actualización de usuarios existentes.	Equipo de desarrollo	Hecho	4	01/07/2025	01/07/2025
	T-018	Implementar la reasignación de roles y actualización automática de permisos.	Equipo de desarrollo	Hecho	2	02/07/2025	02/07/2025
	T-019	Listar usuarios registrados mostrando información básica y rol actual.	Equipo de desarrollo	Hecho	2	02/07/2025	02/07/2025
	T-020	Implementar eliminación de usuarios con validaciones de seguridad.	Equipo de desarrollo	Hecho	2	03/07/2025	03/07/2025
	T-021	Realizar pruebas de aceptación.	Equipo de desarrollo	Hecho	2	03/07/2025	03/07/2025
HU-SEG-004	T-022	Implementar la lógica de visibilidad de menús según permisos del rol.	Equipo de desarrollo	Hecho	3	04/07/2025	04/07/2025
	T-023	Implementar restricciones de acceso en la interfaz de usuario.	Equipo de desarrollo	Hecho	3	04/07/2025	07/07/2025
	T-024	Configurar que los menús sin acceso no aparezcan en la barra de navegación.	Equipo de desarrollo	Hecho	2	07/07/2025	07/07/2025
	T-025	Validar comportamiento de cada opción de permisos	Equipo de desarrollo	Hecho	2	08/07/2025	08/07/2025

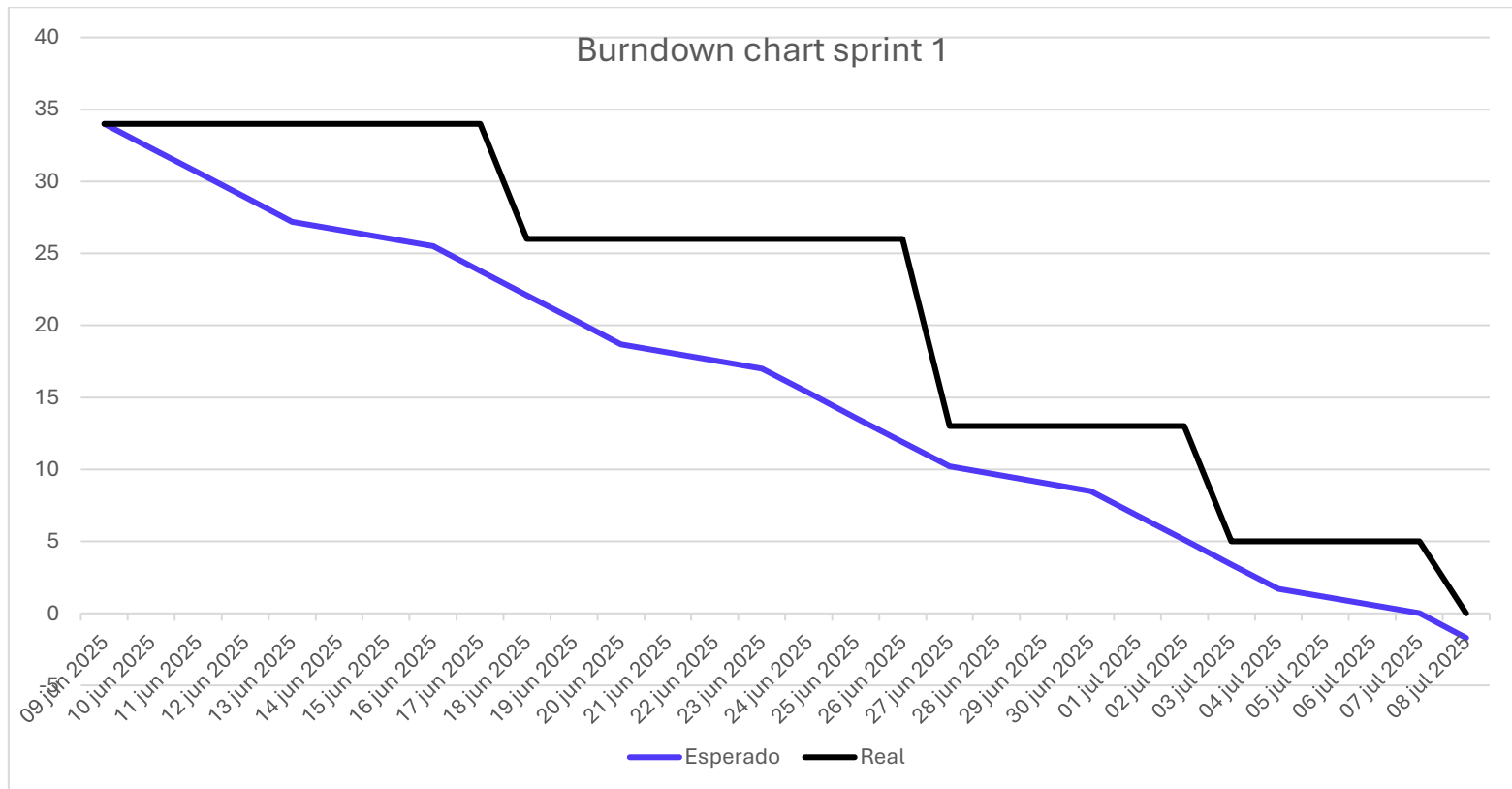
*Nota.* La tabla muestra el taskboard del sprint 1, donde se detalla las tareas ejecutadas en cada historia de usuario.

Si bien la capacidad del sprint 1 fue 80 horas, durante la ejecución el esfuerzo real ascendió a 86 horas. Esta diferencia refleja un ajuste natural en la planificación, ya que algunas tareas requirieron más tiempo del previsto inicialmente. Este aprendizaje permitió refinar las siguientes estimaciones en futuros sprints.

#### 4.1.11. Burndown chart

**Tabla 16.**

*Burndown chart del sprint 1*



*Nota.* El gráfico muestra la comparación entre el progreso esperado y el real durante el desarrollo del sprint 1.

#### 4.1.12. Daily scrum

Para la realización del sprint 1, se implementó la práctica del daily scrum de manera adaptada al contexto del proyecto. Dado que el equipo scrum estuvo conformado únicamente por el tesista, las reuniones diarias se tradujeron en sesiones individuales de auto revisión, documentadas en los hitos más relevantes del avance (finalización de tareas o historias de usuario). Este mecanismo permitió llevar un control del progreso, identificar posibles impedimentos y mantener la trazabilidad de las actividades, cumpliendo con el objetivo principal del daily scrum.

**Tabla 17.**

*Daily scrum del sprint 1*

Fecha	¿Qué hice?	¿Qué haré después?	Impedimentos	Observaciones
09/06/2025	Inicié el diseño y modelado de la base de datos.	Continuar con el modelado.	Ninguno	Avance según lo planificado.
12/06/2025	Finalicé el modelado de la BD.	Configurar de sistema de autenticación.	Ninguno	Terminé dentro del tiempo estimado.
13/06/2025	Configuré el sistema de autenticación.	Preparar control de acceso por roles.	Ajuste de dependencias en librerías	Superado tras pruebas locales.
16/06/2025	Implementé el control de acceso por roles.	Continuar con pruebas de seguridad.	Pequeños ajustes en permisos.	Validado en entorno de pruebas.
17/06/2025	Diseñé el maquetado principal y menú dinámico.	Concluir pruebas de interfaz.	Ninguno	Ninguno
27/06/2025	Completé las tareas de	Verificar la funcionalidad de	Ninguno	Ninguno

	historia de gestión de roles usuario HU- y permisos. SEG-002.			
03/07/2025	Completé las tareas de historia de usuario HU- Validar la funcionalidad de gestión de usuarios. SEG-003.		Ninguno	Ninguno
08/07/2025	Completé las tareas de historia de usuario HU- Realizar pruebas de aceptación de las funcionalidades implementadas. SEG-004.		Ninguno	Ninguno

*Nota.* La tabla muestra los resultados de los daily scrum correspondientes al sprint 1.

#### 4.1.13. Sprint review

Para este primer sprint se verificaron las historias de usuario completadas y se documentaron los entregables alcanzados.

#### Tabla 18.

*Sprint review del sprint 1*

Sprint review	
Fecha:	9/07/2025
Participantes	<ul style="list-style-type: none"> <li>• Scrum Master</li> <li>• Product Owner</li> <li>• Development Team</li> </ul>
Duración	2 horas
Objetivo	Revisar las historias de usuario completadas durante el Sprint 1 y validar su funcionamiento.
Historias de usuario incluidas	<ul style="list-style-type: none"> <li>• HU-SEG-001: Inicio de sesión</li> <li>• HU-SEG-002: Gestión de roles y permisos</li> </ul>

Entregables	<ul style="list-style-type: none"> <li>• HU-SEG-003: Gestión de usuarios</li> <li>• HU-SEG-004: Restricciones de acceso en el sistema</li> <li>• Base de datos modelada e implementada para seguridad.</li> <li>• Funcionalidad de inicio de sesión con control de credenciales.</li> <li>• Sistema de roles y permisos configurado.</li> <li>• Formulario de registro de usuarios con asignación de roles.</li> <li>• Interfaz dinámica (sidebar) que muestra/oculta menús según permisos</li> </ul>
-------------	---

*Nota.* La tabla muestra el resumen del sprint review correspondiente al sprint 1.

#### 4.1.14. Despliegue del sprint

**Tabla 19.**

*Despliegue del sprint 1*

Despliegue del sprint	
Objetivos	Realizar el despliegue del incremento desarrollado del sprint 1 en el entorno de pruebas, garantizando su funcionamiento correcto y la disponibilidad de las funcionalidades del módulo seguridad del sistema.
Actividades realizadas	<ul style="list-style-type: none"> <li>• Se configuró el entorno de despliegue en un servidor de pruebas con PHP 8, MySQL y composer.</li> <li>• Se migraron las bases de datos.</li> <li>• Se inicializaron los datos de prueba en la base de datos.</li> <li>• Se verificó la correcta carga del proyecto y el funcionamiento del módulo de seguridad.</li> </ul>
Resultado	El despliegue del sprint 1 se completó exitosamente, dejando operativo el entorno de pruebas con las funcionalidades del módulo de seguridad.

*Nota.* La tabla detalla el proceso de despliegue del sprint 1 en un servidor de pruebas.

#### 4.1.15. Criterios de aceptación

**Tabla 20.**

*Criterios de aceptación del sprint 1*

Criterios de aceptación		
ID De historia de usuario	Criterio de aceptación	Resultado
HU-SEG-001	<ul style="list-style-type: none"> <li>• Se muestra un formulario con usuario y contraseña.</li> <li>• Si las credenciales son válidas, se accede al sistema.</li> <li>• Si las credenciales son incorrectas, se muestra un mensaje de error.</li> <li>• El usuario es redirigido a su vista personalizada según su rol.</li> </ul>	<ul style="list-style-type: none"> <li>• El sistema muestra un formulario visible en la pantalla de login.</li> <li>• El sistema permite el acceso una vez validado las credenciales (usuario y contraseña).</li> <li>• El sistema muestra el mensaje de error al ingresar credenciales inválidas.</li> <li>• El sistema después del login redirige al usuario con menús correspondiente a su rol.</li> </ul>
HU-SEG-002	<ul style="list-style-type: none"> <li>• Se puede crear un nuevo rol.</li> <li>• Se valida que no haya roles duplicados.</li> <li>• Se puede editar el nombre y los permisos de un rol existente.</li> <li>• No se puede eliminar un rol si tiene usuarios asignados.</li> </ul>	<ul style="list-style-type: none"> <li>• El sistema permite la creación de un rol</li> <li>• El sistema valida que el nombre de los roles sea único.</li> <li>• El sistema permite editar un rol.</li> <li>• El sistema impide eliminar un rol si existe usuarios con este rol asignado.</li> </ul>

- Se pueden asignar permisos sobre menús: Todos, Solo lectura, Solo crea, Crea/lee/edita, Sin acceso.
- Los permisos definen automáticamente qué botones o acciones aparecen en la interfaz.
- Un rol sin acceso no puede visualizar el menú.
- El sistema permite crear roles con sus respectivos permisos.
- El sistema muestra botones de acciones de acuerdo con el rol del usuario asignado.
- El sistema muestra en el sidebar solo los menús autorizados para el rol.

## HU-SEG-003

- El sistema permite registrar un nuevo usuario mediante un proceso por pasos:
  1. Datos personales
  2. Documento de identidad
  3. Contacto del usuario
  4. Contacto de emergencia
  5. Selección de rol
- Se puede editar y actualizar los datos del usuario en cualquier momento.
- Se puede cambiar el rol asignado a un usuario.
- Se listan todos los usuarios registrados mostrando sus datos básicos y rol actual.
- El sistema permite eliminar un usuario siempre que no tenga operaciones críticas asociadas.
- El sistema permite la creación de una cuenta de usuario mediante un formulario por pasos.
- El sistema permite la actualización los datos un usuario.
- El sistema permite cambiar el rol de un usuario.
- El sistema muestra todos los usuarios registrados.
- El sistema permite eliminar un usuario.
- Al cambiar el rol de un usuario el sistema actualiza los menús.

HU-SEG-004	<ul style="list-style-type: none"><li>• La asignación de rol impacta inmediatamente en los menús y permisos visibles del usuario.</li><li>• Si un rol tiene “solo lectura”, los botones de crear/editar se ocultan.</li><li>• Si un rol tiene “sin acceso”, el menú no se muestra en la interfaz.</li><li>• Si un rol tiene “todos los permisos”, accede a todas las funciones de ese menú.</li></ul>	El sistema responde adecuadamente a los permisos asignados a un rol.
------------	---	--

---

*Nota.* La tabla detalla los criterios de aceptación de las historias de usuario correspondiente al sprint 1.

#### 4.1.16. Pruebas unitarias

Para probar el correcto funcionamiento de la aplicación a nivel de código, se utilizó el framework PHPUnit, que permite comprobar la funcionalidad de pequeñas unidades de código. se tomó 8 unidades de código el cual tienen por funcionalidad la autenticación de usuarios, gestión de roles y usuarios. Estas pruebas dieron como resultado exitoso en cual se evidencian en las figuras **Figura 19**, **Figura 20** y **Figura 21**.

**Tabla 21.**

*Prueba unitaria - sprint 1*

No	Historia de usuario	Caso de prueba	Resultado
1	HU-SEG-001	<ul style="list-style-type: none"> <li>• Autenticación</li> </ul>	Exitoso
2	HU-SEG-002	<ul style="list-style-type: none"> <li>• Listar roles</li> <li>• Crear rol</li> <li>• Editar rol</li> <li>• Eliminar rol</li> </ul>	Exitoso
3	HU-SEG-003	<ul style="list-style-type: none"> <li>• Crear usuario</li> <li>• Editar usuario</li> <li>• Eliminar usuario</li> </ul>	Exitoso

*Nota.* La tabla muestra las pruebas unitarias realizadas en cada historia de usuario correspondiente al sprint 1.

**Figura 19.**

*Resultado de las pruebas unitarias de autenticación de usuarios.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test

PASS Tests\Feature\Auth\AuthenticationTest
✓ login screen can be rendered 9.91s
✓ users can authenticate using the login screen 9.67s
✓ users can not authenticate with invalid password 9.53s

Tests: 3 passed (7 assertions)
Duration: 29.36s
```

De la **Figura 19**, se observa el resultado de las pruebas unitarias de autenticación, estas pruebas constan de 3 unidades de código, las cuales validan: el correcto renderizado de la página del login, la correcta autenticación de usuarios y el bloqueo de acceso en caso de una contraseña incorrecta. La prueba da como resultado de aprobado o exitoso en las 3 unidades de código, asegurando la integridad en la aplicación. El código fuente de la prueba unitaria se puede observar en el **Anexo D**,

### Figura 20.

*Resultado de las pruebas unitarias de gestión de roles.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=RoleMenuTest

PASS Tests\Feature\Security\RoleMenuTest
✓ list returns roles 8.06s
✓ show returns details or 404 7.81s
✓ store creates role and rolemenus 7.69s
✓ update replaces roles or 404 7.94s
✓ delete role related as inactive or 404 7.56s

Tests: 5 passed (16 assertions)
Duration: 39.33s
```

De la **Figura 20** se observa el resultado de las pruebas unitarias de gestión de rol, estas pruebas constan de 5 unidades de código, las cuales validan: el listado de roles, la creación de roles, la actualización de un rol y el eliminado de un rol. La prueba da como resultado de aprobado o exitoso en las 5 unidades de código, asegurando la integridad en la aplicación. El código fuente de la prueba unitaria se puede observar en el **Anexo E**,

### Figura 21.

*Resultado de las pruebas unitarias de gestión de usuarios.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=UserTest

PASS Tests\Feature\Security\UserTest
✓ user can be created 8.63s
✓ user can be updated 8.02s
✓ user can be deleted 8.02s

Tests: 3 passed (28 assertions)
Duration: 25.07s
```

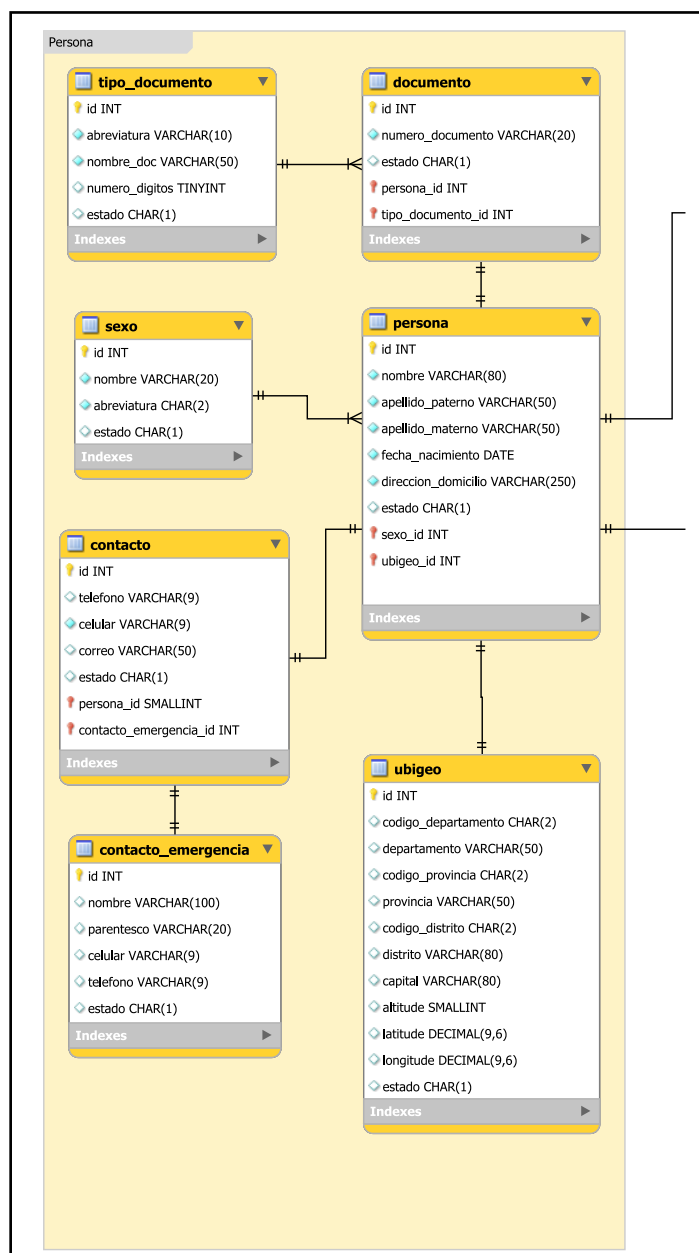
De la **Figura 21** se observa el resultado de las pruebas unitarias de gestión de usuarios, estas pruebas constan de 3 unidades de código, las cuales validan: la creación de usuarios, la actualización de un usuario y el eliminado de un usuario. La prueba da como resultado de

aprobado o exitoso en las 3 unidades de código, asegurando la integridad en la aplicación. El código fuente de la prueba unitaria se puede observar en el **Anexo F**.

#### 4.1.17. Resultados del sprint

**Figura 22.**

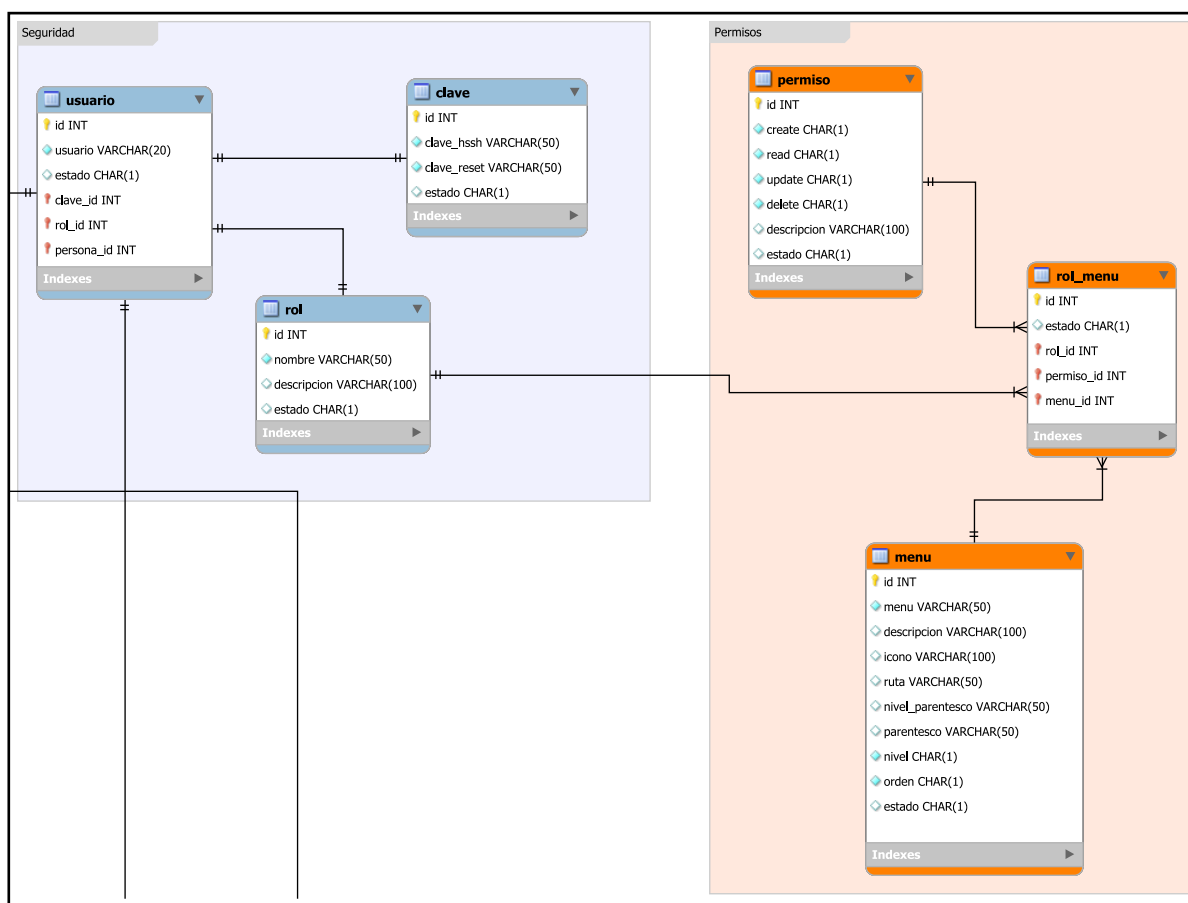
*Modelo físico de base de datos - Gestión de persona*



*Nota.* Las tablas mostradas en la figura corresponden a las implicadas para la gestión de personas.

Figura 23.

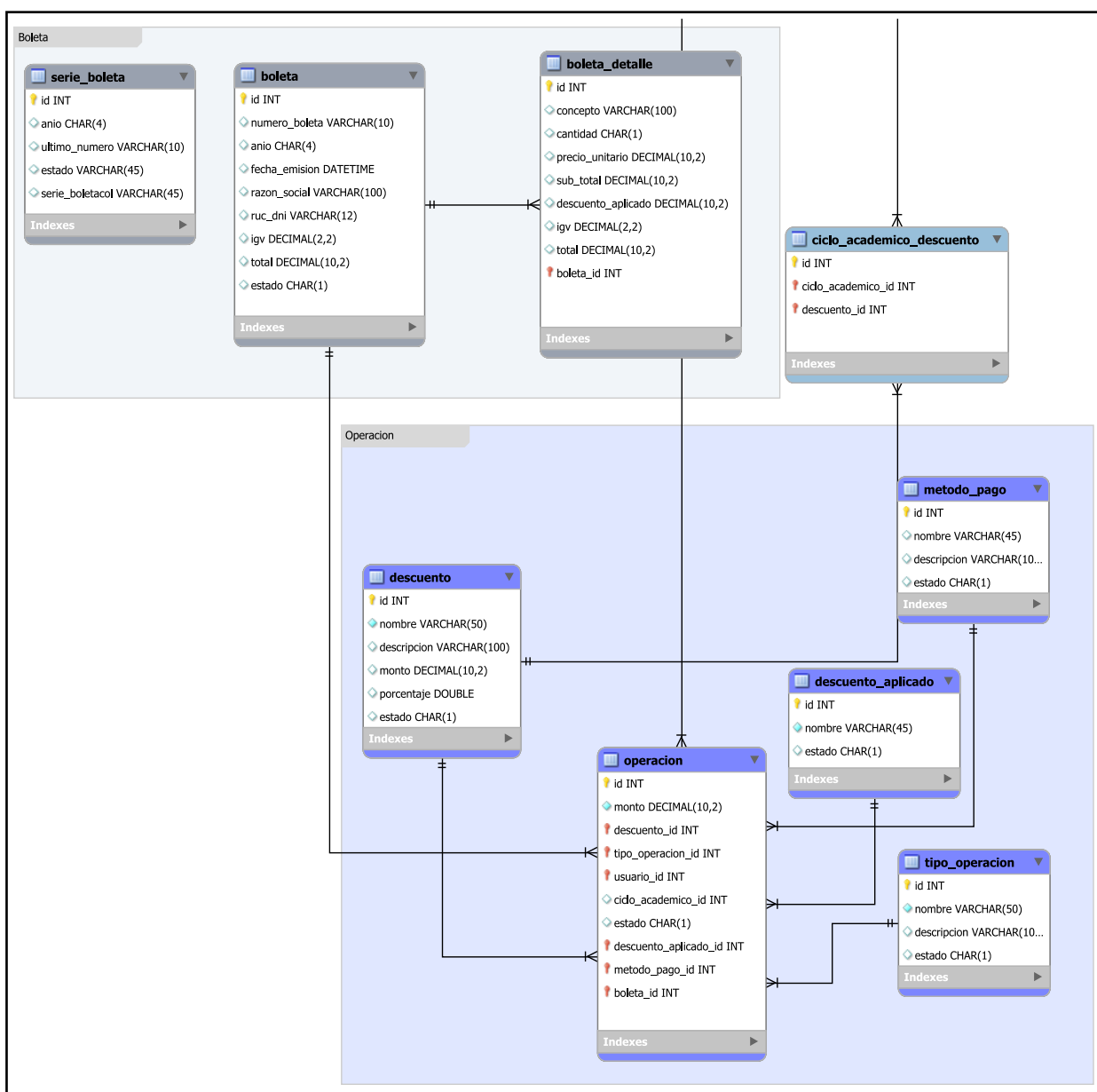
Modelo físico de base de datos - Módulo seguridad



*Nota.* Las tablas mostradas en la figura corresponden a las implicadas para la gestión de usuarios.

Figura 24.

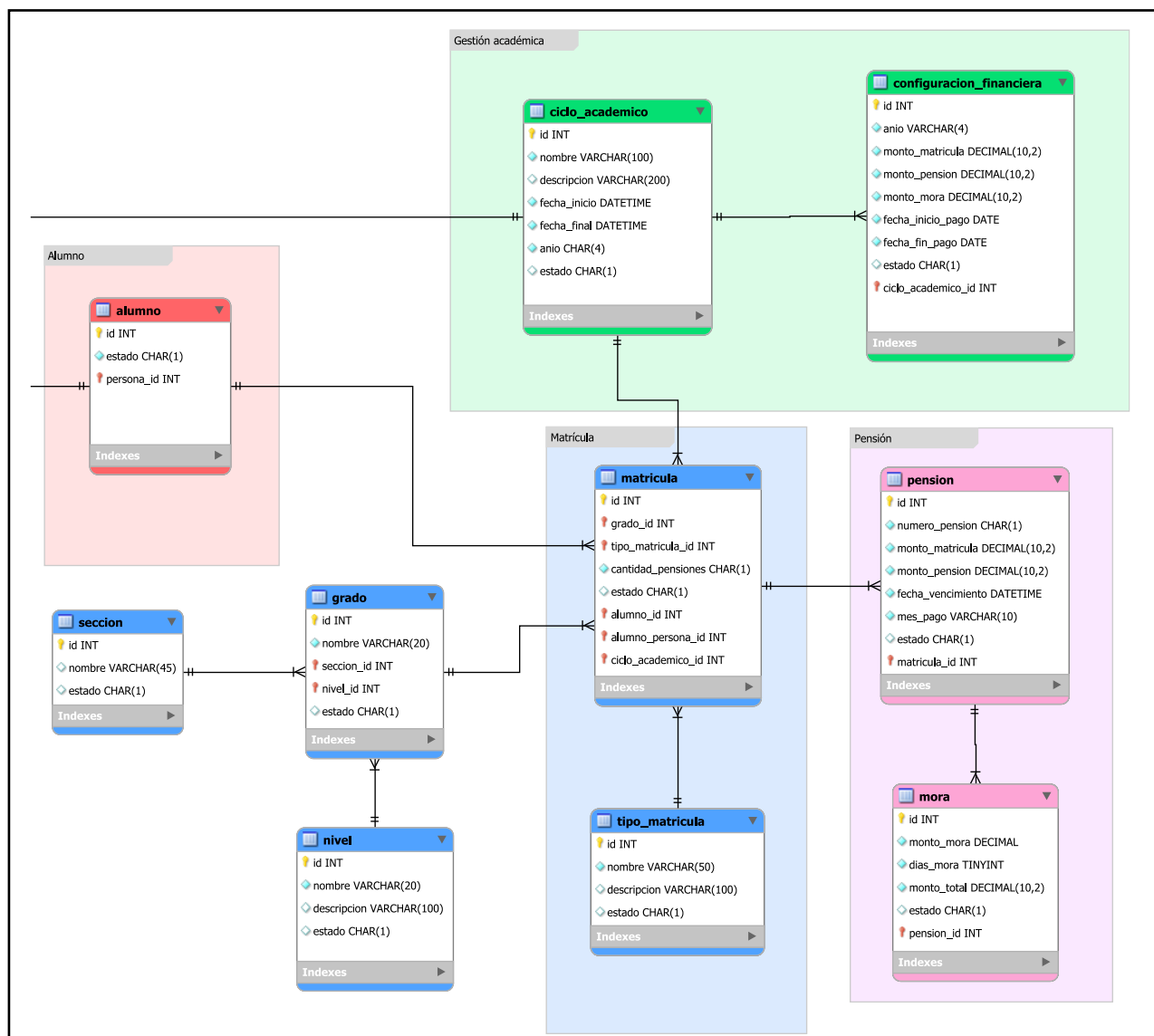
## Modelo físico de base de datos - Módulo operación



*Nota.* Las tablas mostradas en la figura corresponden a las implicadas para la gestión de operaciones (pago de pensiones, pago de matrícula entre otros).

Figura 25.

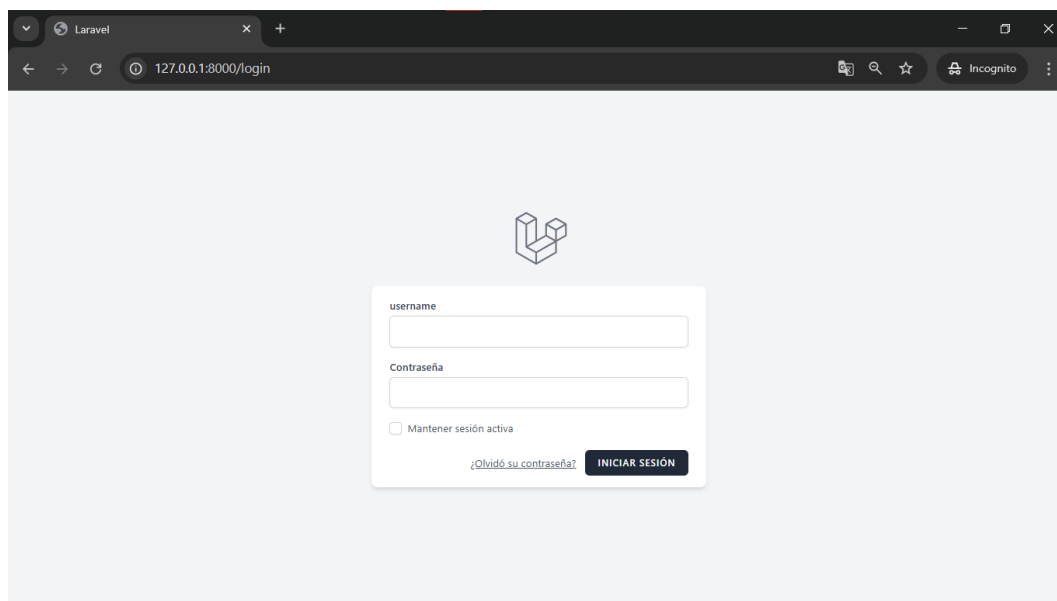
Modelo físico de base de datos - Módulo académico y configuración



*Nota.* Las tablas mostradas en la figura corresponden a las implicadas para la gestión de ciclo académico y el control de pagos.

**Figura 26.**

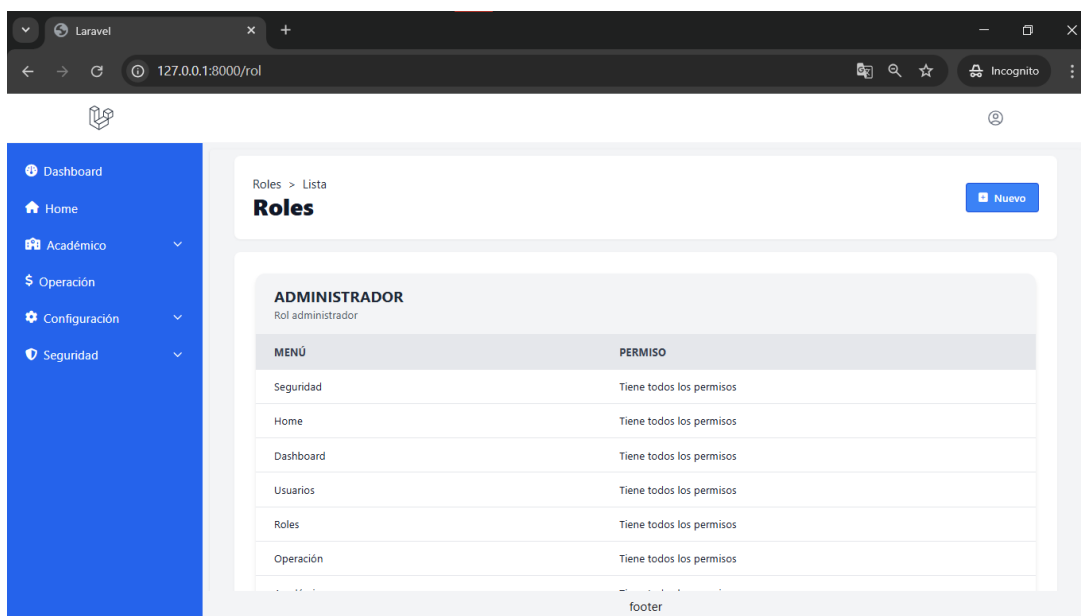
*Interfaz de inicio de sesión de la aplicación.*



*Nota.* La figura muestra la pantalla de inicio de sesión de la aplicación,

**Figura 27.**

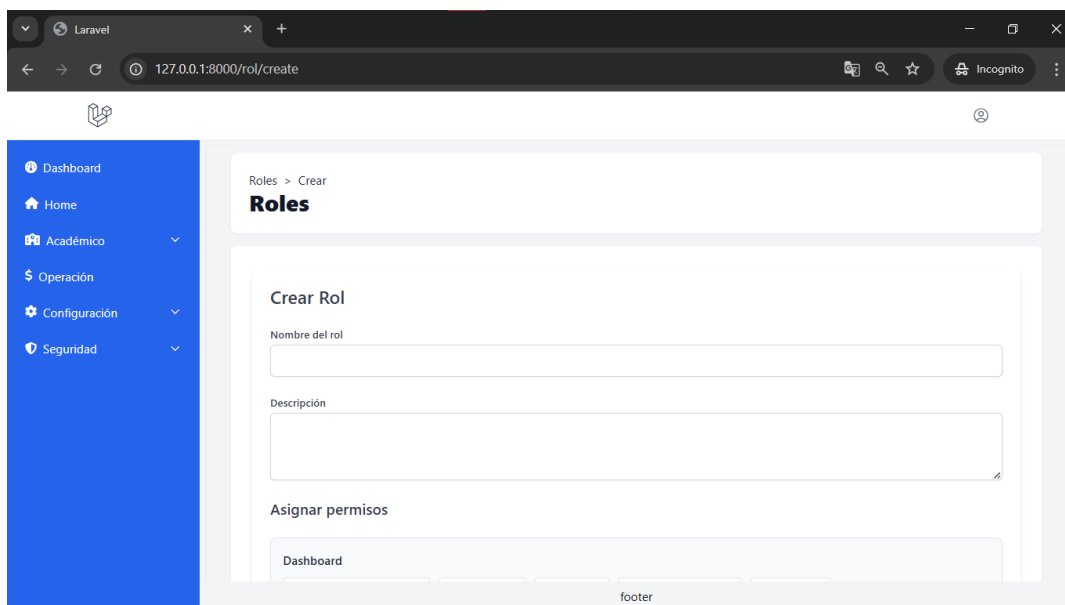
*Interfaz de la página principal de rol.*



*Nota.* La figura muestra la interfaz de la página principal del menú rol.

**Figura 28.**

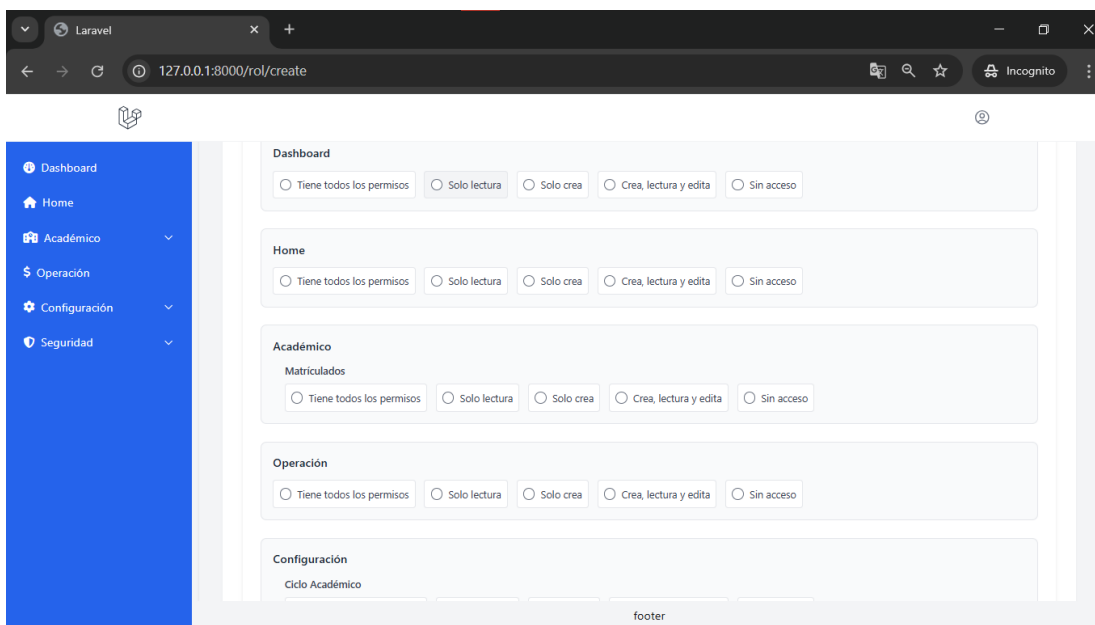
*Interfaz de creación de rol.*



*Nota.* La figura muestra la interfaz de creación de un rol.

**Figura 29.**

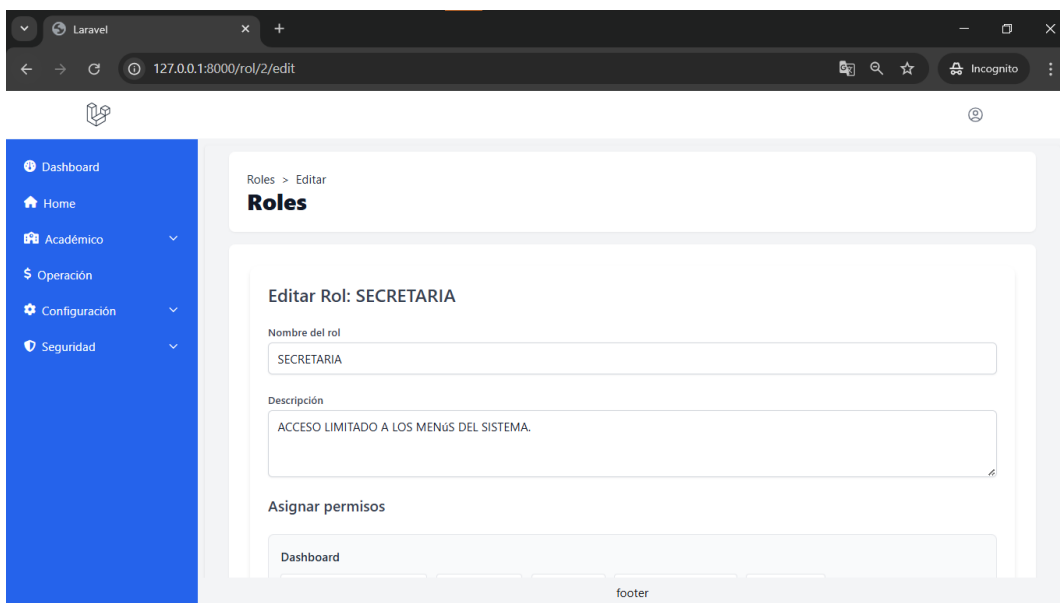
*Interfaz de creación de rol - asignación de permisos.*



*Nota.* Se observa los permisos que puede tener el rol en los menús.

**Figura 30.**

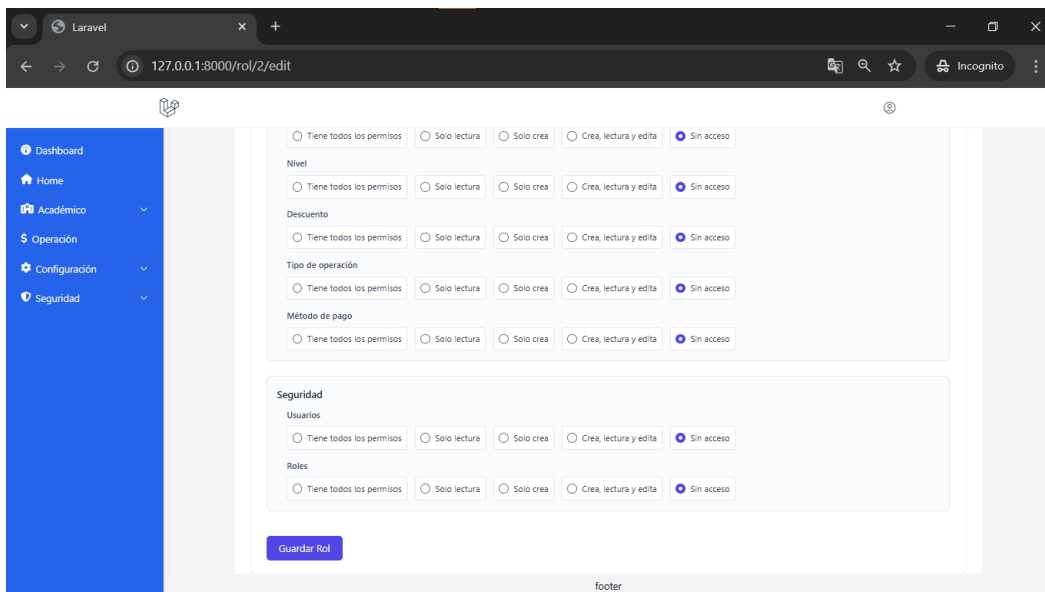
*Interfaz de edición de rol.*



*Nota.* Se observa los permisos que puede tener el rol en los menús.

**Figura 31.**

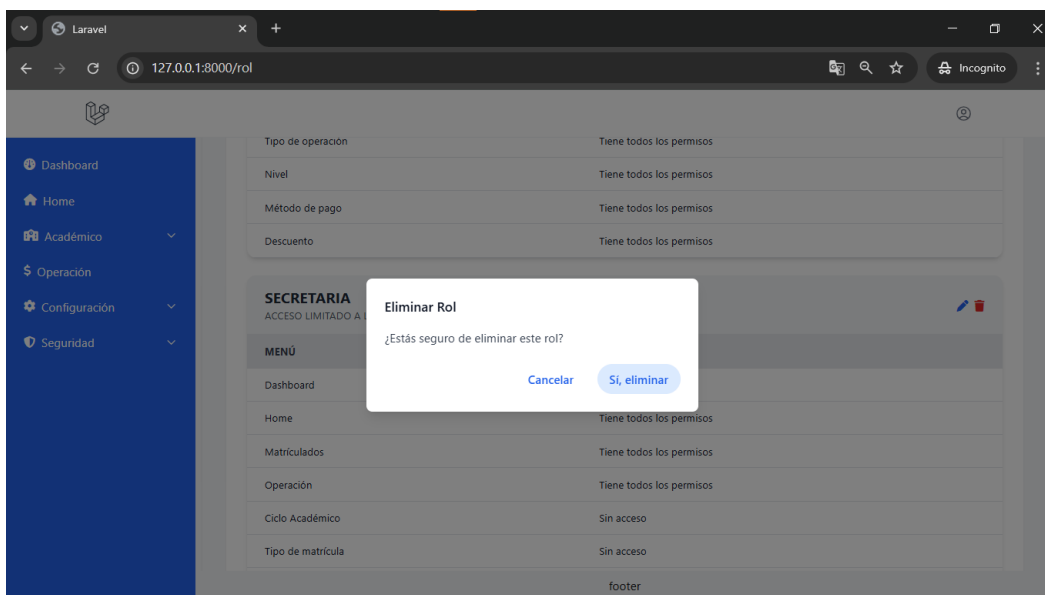
*Interfaz de editar rol - asignación de permisos.*



*Nota.* Se observa la interfaz para actualizar los permisos de un rol.

**Figura 32.**

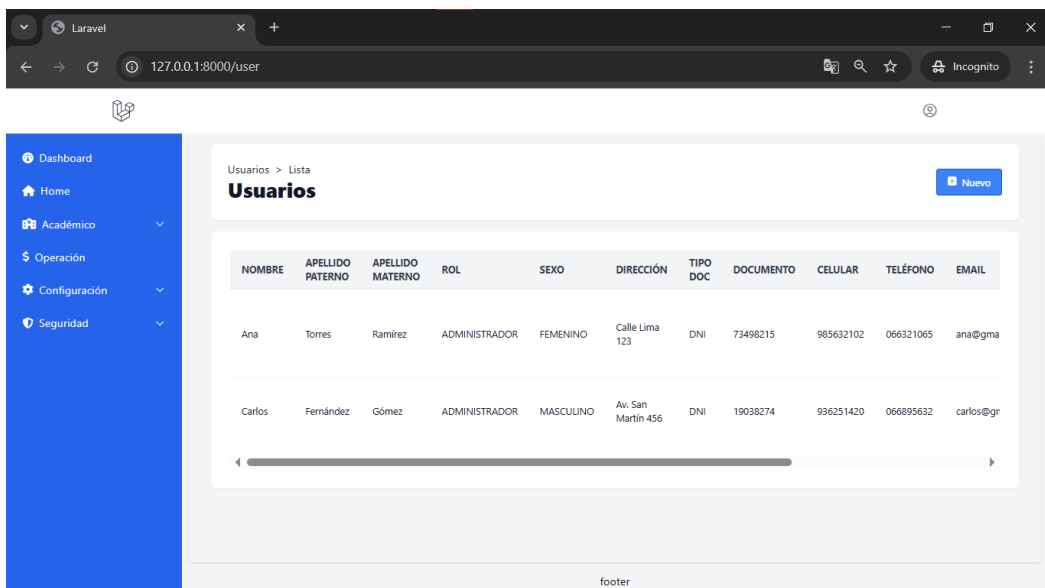
*Interfaz de eliminado de rol.*



*Nota.* La imagen muestra un modal que permite eliminar un rol.

**Figura 33.**

*Interfaz de listado de usuarios.*



*Nota.* La imagen muestra la Interfaz de la página principal de usuarios.

#### 4.1.18. Sprint retrospective

En el Sprint Retrospective, el tesista, en su rol de Scrum Team, evaluó el proceso de trabajo desarrollado durante el sprint, identificando fortalezas, dificultades y oportunidades de mejora para siguientes iteraciones.

**Tabla 22.**

*Sprint retrospective del sprint 1*

Sprint retrospective	
Fecha:	09/07/2025
Participantes	<ul style="list-style-type: none"> <li>• Scrum master</li> <li>• Product owner</li> <li>• Development team</li> </ul>
Objetivo	Reflexionar sobre el desarrollo del Sprint 1 para mejorar la organización y eficiencia en los próximos sprints.
Aspectos positivos	<ul style="list-style-type: none"> <li>• Se cumplieron todas las historias planificadas en el tiempo estimado.</li> <li>• El modelo de base de datos quedó correctamente definido desde el inicio, lo que facilitó las implementaciones.</li> <li>• La división de tareas en el Taskboard permitió visualizar claramente el avance.</li> </ul>
Aspectos para mejorar	<ul style="list-style-type: none"> <li>• La estimación de horas para algunas tareas fue ajustada sobre la marcha (ejemplo: T-001 tomó menos tiempo del esperado, pero T-003 requirió más ajustes).</li> <li>• Se identificó que el diseño de la interfaz requiere más tiempo de prueba con usuarios reales.</li> </ul>
Acciones de mejora para el siguiente sprint	<ul style="list-style-type: none"> <li>• Refinar las estimaciones de tareas considerando mayor tiempo para el diseño de interfaces.</li> <li>• Registrar incidencias y bloqueos en un diario para analizarlos en el siguiente sprint.</li> </ul>

- 
- Reutilizar plantillas de formularios para acelerar futuros desarrollos.
- 

*Nota.* La tabla muestra el resultado de la retrospectiva del sprint 1.

## **1.16. Sprint 2**

### **4.1.19. Sprint planning**

La planificación del sprint 2 tuvo por objetivo implementar el módulo de configuración del sistema ya que constituye la base fundamental, sobre el que se sostendrá el flujo de matrícula y los procesos operativos del sistema en los sprints posteriores.

Durante la planificación se revisó el producto backlog, priorizando historias de usuario con mayor valor para el negocio y dependencia lógica para el desarrollo del sistema. Se seleccionaron ocho historias de usuario: HU-CONF-001, HU-CONF-002, HU-CONF-003, HU-CONF-004, HU-CONF-005, HU-CONF-006, HU-CONF-007 Y HU-CONF-008.

La prioridad y estimación de esfuerzo se definieron considerando el impacto funcional, los riesgos asociados y la complejidad técnica de cada historia. Todas las historias fueron clasificadas con prioridad alta, por ser componentes esenciales que definen la estructura y parámetros base del sistema. La estimación del esfuerzo se realizó mediante la técnica del planning poker. La historia HU-CONF-001 (gestión de ciclo académico) recibió 5 puntos por su complejidad moderada al incluir validaciones, configuración financiera y procesos por pasos: mientras que las historias HU-CONF-002 al HU-CONF-008 fueron valoradas con 3 puntos al corresponder a funcionalidades CRUD básicos, con menor nivel de complejidad y riesgo técnico. Esta estimación permitió planificar un sprint equilibrado, asegurando la entrega de un método funcional y coherente con la arquitectura del sistema.

**Tabla 23.***Sprint planning del sprint 2*

Reunión de planificación de sprint 2	
Fecha	10/07/2025
Participantes	<ul style="list-style-type: none"> <li>• Product owner</li> <li>• Scrum team</li> <li>• Scrum master</li> </ul>
Artefactos involucrados	<ul style="list-style-type: none"> <li>• Product backlog</li> <li>• Backlog de sprint</li> </ul>
Agenda	<ul style="list-style-type: none"> <li>• Revisión del producto backlog y definición de objetivos del sprint.</li> <li>• Selección de historias de prioridad alta.</li> <li>• Estimación de esfuerzo con planning poker.</li> <li>• Definición de tareas para cada historia seleccionada.</li> </ul>
Objetivos	<ul style="list-style-type: none"> <li>• Desarrollar el módulo de configuración.</li> <li>• Completar las historias de usuario HU-CONF-001, HU-CONF-002, HU-CONF-003, HU-CONF-004, HU-CONF-005, HU-CONF-006, HU-CONF-007 y HU-CONF-008.</li> </ul>

*Nota.* La tabla resume la reunión de planificación para el desarrollo del sprint 2.

#### 4.1.20. Sprint backlog

**Tabla 24.**

*Sprint backlog del sprint 2.*

ID de la historia	Descripción	Módulo	Sprint	Prioridad	Esfuerzo
HU-CONF-001	Como administrador, quiero registrar ciclos académicos mediante un proceso por pasos, para organizar la matrícula y su gestión financiera.	Configuración	2	Alta	13
HU-CONF-002	Como administrador, quiero gestionar secciones para organizar a los estudiantes dentro de las aulas.	Configuración	2	Alta	3
HU-CONF-003	Como administrador, quiero gestionar niveles para organizar a los estudiantes por niveles escolares (inicial, primaria y secundaria).	Configuración	2	Alta	3
HU-CONF-004	Como administrador, quiero gestionar grados para organizar a los estudiantes dentro de la institución.	Configuración	2	Alta	5
HU-CONF-005	Como administrador, quiero configurar tipos de matrícula (regular, extraordinaria, traslado) para adaptarme a las necesidades de la institución.	Configuración	2	Media	3
HU-CONF-006	Como administrador, quiero registrar descuentos (hermano, puntualidad, beca) para aplicarlos a los pagos de matrícula y pensiones.	Configuración	2	Media	5

HU-CONF-007	Como administrador, quiero configurar tipos de operación (matrícula, pago de pensión) registrar correctamente las transacciones financieras.	Configuración	2	Media	3
HU-CONF-008	Como administrador, quiero configurar métodos de pago (efectivo, tarjeta, transferencia) para registrar correctamente las transacciones financieras.	Configuración	2	Media	3

---

*Nota.* La tabla muestra el sprint backlog del sprint 1, agrupa las historias de usuario del módulo configuración.

#### 4.1.21. Historias de usuario

**Tabla 25.**

*HU-CONF-001 Gestión de ciclo académico*

Historia de usuario	
Código: HU-CONF-001	Nombre: Gestión de ciclo académico
Módulo: Configuración	
Responsable: Equipo de desarrollo	Iteración: sprint 2
Prioridad: Alta	Esfuerzo: 13
Descripción	
Como administrador, quiero registrar ciclos académicos mediante un proceso por pasos, para organizar la matrícula y su gestión financiera.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• El ciclo académico se crea en pasos secuenciales.</li> <li>• En el paso 1 se ingresan los datos generales: nombre, descripción, fecha de inicio y fin.</li> <li>• No se pueden crear ciclos académicos con fechas pasadas.</li> <li>• La duración mínima de un ciclo académico es de un mes.</li> <li>• En el paso 2 se configura la parte financiera: monto de matrícula, monto de pensión, mora por retraso, fechas de pago.</li> <li>• Si el pago es único, solo se permite registrar una única fecha de pago.</li> <li>• En el paso 3 se seleccionan los descuentos existentes que aplicarán al ciclo académico.</li> <li>• No se puede eliminar un ciclo académico que tenga matrículas asociadas.</li> <li>• Se puede editar un ciclo académico en curso, respetando las validaciones de fechas.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-CONF-001.

**Figura 34.**

*Wireframe de la vista de listado de ciclo académico.*

Roles > Lista

**Ciclo académico** Nuevo

NOMBRE	DESCRIPCIÓN	F. INICIO	F. FIN	AÑO	DURACIÓN	PENSIÓN	ACCIONES
ANUAL	CICLO LECTIVO	31/03/2026	31/12/2026	2026	9 MESES	450	
VERANO	CICLO LECTIVO	1/01/2026	20/02/2026	2026	2 MESES	180	

*Nota.* La figura muestra el bosquejo de la vista principal del menú de ciclo académico.

**Figura 35.**

*Wireframe de la vista de creación de ciclo académico - paso 1.*

Ciclo académico > Crear

**Ciclo académico**

Nombre

Descripción

Fecha de inicio

Fecha de finalización

Siguiente

*Nota.* La figura muestra el bosquejo de la vista utilizada para la creación de un ciclo académico.

**Figura 36.**

*Wireframe de creación de ciclo académico -- paso 2.*

The wireframe shows a web browser window with a navigation menu on the left and a main content area. The navigation menu includes: Dashboard, Home, Académico, Operaciones, Configuración, **Ciclo académico**, and Seguridad. The main content area has a breadcrumb 'Ciclo académico > Crear' and a title 'Ciclo académico'. Below this is a section titled 'Configuración financiera' with the following fields and controls:

- Monto s/.:
- Pensión s/.:
- Monto por retraso s/.:
- Fechas de pago:  Pago único
- Fecha de inicio de pago:
- Último día de pago:

At the bottom of the form are two buttons: 'Atrás' and 'Siguiente'.

*Nota.* La figura muestra el bosquejo de la vista utilizada para la creación de un ciclo académico correspondiente al paso 2 de configuración financiera.

**Figura 37.**

*Wireframe de creación de ciclo académico -- paso 3.*

The wireframe shows a web browser window with a navigation menu on the left and a main content area. The navigation menu includes: Dashboard, Home, Académico, Operaciones, Configuración, **Ciclo académico**, and Seguridad. The main content area has a breadcrumb 'Ciclo académico > Crear' and a title 'Ciclo académico'. Below this is a section titled 'Descuentos' with the following controls:

- HERMANOS
- MEDIA BECA
- ESPECIAL

At the bottom of the form are two buttons: 'Atrás' and 'Guardar'.

*Nota.* La figura muestra el bosquejo de la vista utilizada para la creación de un ciclo académico correspondiente al paso 3 de configuración de descuentos.

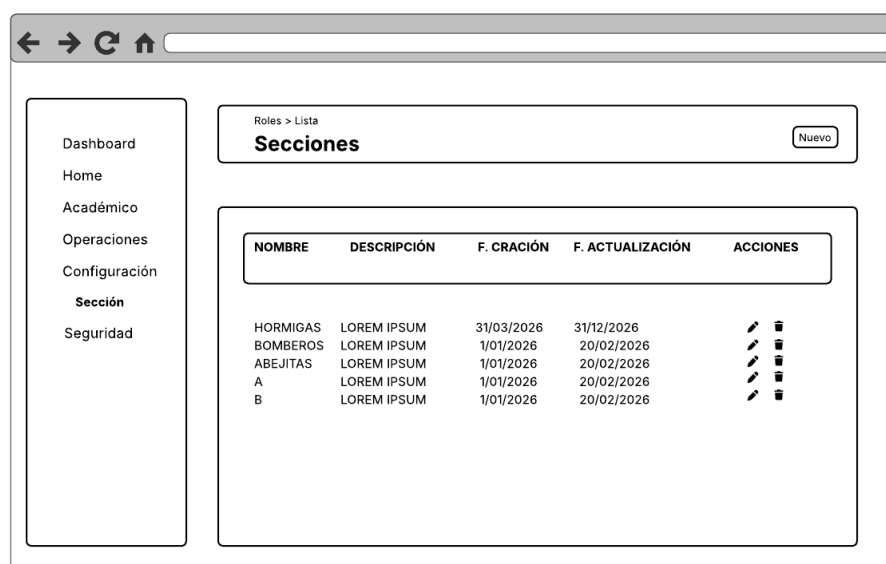
**Tabla 26.***HU-CONF-002 gestión de secciones*

Historia de usuario	
Código: HU-CONF-002	Nombre: Gestión de secciones
Módulo: Configuración	
Responsable: Equipo de desarrollo	Iteración: sprint 2
Prioridad: Alta	Esfuerzo: 3
Descripción	
Como administrador, quiero gestionar secciones para organizar a los estudiantes dentro de las aulas.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se pueden crear secciones de manera independiente.</li> <li>• Se pueden listar todas las secciones activas.</li> <li>• No se permiten secciones duplicadas.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-CONF-002.

**Figura 38.**

*Wireframe de la vista de listado de secciones.*



*Nota.* La figura muestra el bosquejo de la vista principal del menú de sección.

**Figura 39.**

*Wireframe de creación y edición de sección.*

*Nota.* La figura muestra el bosquejo de creación y edición de sección.

**Tabla 27.**

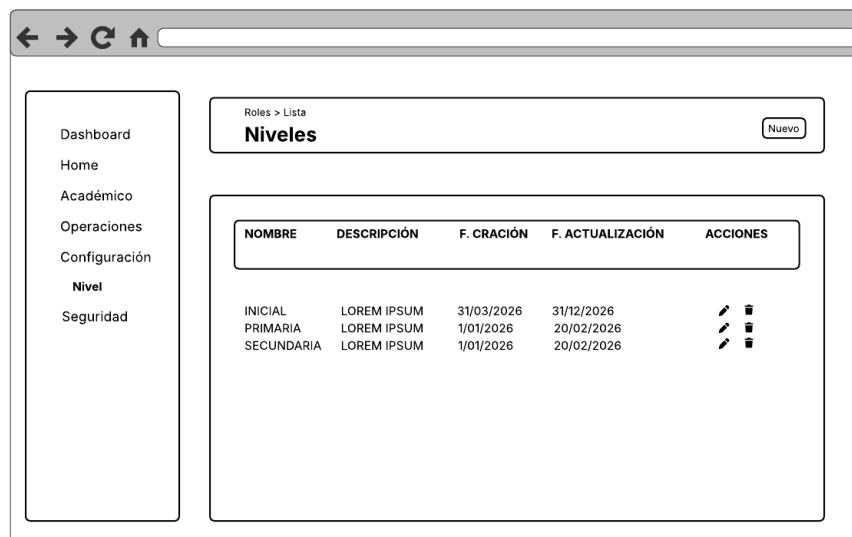
*HU-CONF-003 gestión de niveles*

Historia de usuario	
Código: HU-CONF-003	Nombre: Gestión de niveles
Módulo: Configuración	
Responsable: Equipo de desarrollo	Iteración: sprint 2
Prioridad: Alta	Esfuerzo: 3
Descripción	
Como administrador, quiero gestionar niveles para organizar a los estudiantes por niveles escolares (inicial, primaria y secundaria).	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se pueden registrar niveles educativos.</li> <li>• Se pueden listar todos los niveles activos.</li> <li>• No se permiten niveles duplicados.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-CONF-003.

**Figura 40.**

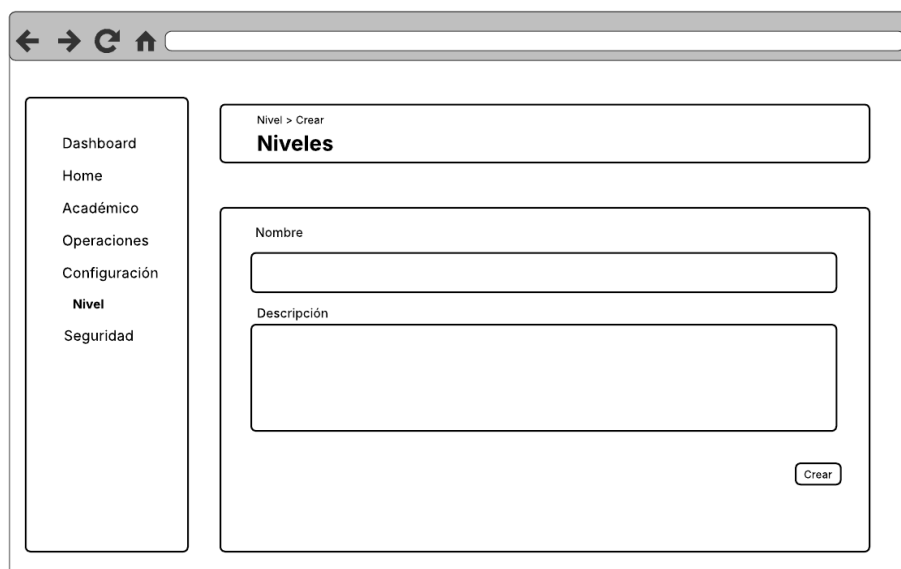
*Wireframe de listado de niveles.*



*Nota.* La figura muestra el bosquejo de la vista principal del menú nivel.

**Figura 41.**

*Wireframe de creación y edición de niveles.*



*Nota.* La figura muestra el bosquejo de creación y edición de niveles.

**Tabla 28.***HU-CONF-004 gestión de grados*

Historia de usuario	
Código: HU-CONF-004	Nombre: Gestión de grados
Módulo: Configuración	
Responsable: Equipo de desarrollo	Iteración: sprint 2
Prioridad: Alta	Esfuerzo: 5
Descripción	
Como administrador, quiero gestionar grados para organizar a los estudiantes dentro de la institución.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se pueden crear grados.</li> <li>• Cada grado se asocia a un nivel y a una sección existente.</li> <li>• Se listan grados con su nivel y sección correspondiente.</li> <li>• No se permiten grados duplicados en el mismo nivel y sección.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-CONF-004.

**Figura 42.***Wireframe de listado de grados.*

El wireframe muestra una interfaz de usuario para la gestión de grados. A la izquierda hay un menú lateral con las opciones: Dashboard, Home, Académico, Operaciones, Configuración, Grado (seleccionado) y Seguridad. El contenido principal está dividido en dos secciones. La superior, titulada 'Grado > Lista Grados', contiene un botón 'Nuevo'. La inferior es una tabla con las siguientes columnas: NOMBRE, SECCIÓN, NIVEL, ESTADO y ACCIONES. La tabla contiene los siguientes datos:

NOMBRE	SECCIÓN	NIVEL	ESTADO	ACCIONES
PRIMERO	A	PRIMARIA	ACTIVO	[icono de editar] [icono de borrar]
SEGUNDO	B	PRIMARIA	ACTIVO	[icono de editar] [icono de borrar]
TERCERO	ÚNICA	SECUNDARIA	ACTIVO	[icono de editar] [icono de borrar]
PRIMERO	ÚNICA	SECUNDARIA	ACTIVO	[icono de editar] [icono de borrar]
SEGUNDO	ÚNICA	SECUNDARIA	ACTIVO	[icono de editar] [icono de borrar]
CUARTO	ÚNICA	SECUNDARIA	ACTIVO	[icono de editar] [icono de borrar]
QUINTO	ÚNICA	SECUNDARIA	ACTIVO	[icono de editar] [icono de borrar]

*Nota.* La figura muestra el bosquejo de la vista principal de grados.

**Figura 43.**

*Wireframe de creación y edición de grados.*

El wireframe muestra una interfaz de usuario para la creación y edición de grados. En la parte superior izquierda hay un navegador con botones de navegación (atrás, adelante, recargar, inicio) y una barra de dirección. A la izquierda del contenido principal hay un menú lateral con los siguientes ítems: Dashboard, Home, Académico, Operaciones, Configuración, **Grado** (destacado) y Seguridad. El contenido principal está dividido en dos secciones. La superior es un encabezado que dice 'Grado > Crear' y 'Grados'. La inferior es un formulario con tres campos de entrada: 'Nombre' (un campo de texto vacío), 'Sección' (un menú desplegable con '--Seleccione--') y 'Nivel' (un menú desplegable con '--Seleccione--'). En la parte inferior derecha del formulario hay un botón 'Crear'.

*Nota.* La figura muestra el bosquejo de creación y edición de grados.

**Tabla 29.**

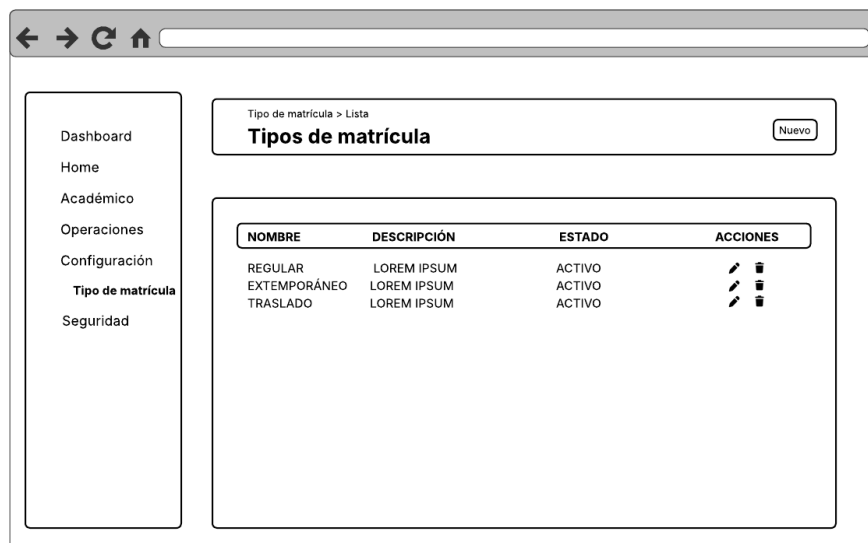
*HU-CONF-005 gestión de tipos de matrícula*

Historia de usuario	
Código: HU-CONF-005	Nombre: Gestión de tipos de matrícula
Módulo: Configuración	
Responsable: Equipo de desarrollo	Iteración: sprint 2
Prioridad: Alta	Esfuerzo: 3
Descripción	
Como administrador, quiero configurar tipos de matrícula (regular, extraordinaria, traslado) para adaptarme a las necesidades de la institución.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se pueden registrar diferentes tipos de matrícula.</li> <li>• Se pueden editar los tipos existentes.</li> <li>• Los tipos de matrícula se pueden asignar a estudiantes en el proceso de inscripción.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-CONF-005.

**Figura 44.**

*Wireframe de listado de tipos de matrícula.*



*Nota.* La figura muestra el bosquejo de la vista principal del menú tipos de matrícula.

**Figura 45.**

*Wireframe de creación y edición de tipo de matrícula.*







*Nota.* La figura muestra el bosquejo de la vista de creación y edición de tipos de matrícula.

**Tabla 30.***HU-CONF-006 gestión de descuentos*

Historia de usuario	
Código: HU-CONF-006	Nombre: Gestión de descuentos
Módulo: Configuración	
Responsable: Equipo de desarrollo	Iteración: sprint 2
Prioridad: Alta	Esfuerzo: 5
Descripción	
Como administrador, quiero registrar descuentos (hermano, puntualidad, beca) para aplicarlos a los pagos de matrícula y pensiones.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se pueden crear distintos tipos de descuento con un porcentaje o monto definido.</li> <li>• Se pueden editar o eliminar descuentos activos.</li> <li>• Los descuentos se aplican automáticamente al registrar una matrícula.</li> <li>• El descuento se puede aplicar a la matrícula o pensiones.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-CONF-006.

**Figura 46.***Wireframe de listado de descuentos.*

Dashboard > Lista		Descuentos <span style="float: right;">Nuevo</span>			
NOMBRE	DESCRIPCIÓN	MONTO DE DESCUENTO \$/.	DESCUENTO %	ACCIONES	
HERMANOS	LOREM	80	-	 	
MEDIDA BECHA	LOREM	-	50%	 	

*Nota.* La figura muestra el bosquejo de la vista principal del menú descuentos.

**Figura 47.**

*Wireframe de creación y edición de descuento.*

El wireframe muestra una interfaz de usuario para la creación y edición de descuentos. En la parte superior, hay una barra de navegación con botones de retroceso, avance, recarga y inicio. A la izquierda, un menú lateral contiene los siguientes ítems: Dashboard, Home, Académico, Operaciones, Configuración, **Descuento** (destacado) y Seguridad. El contenido principal está dividido en dos secciones: una barra superior que indica 'Descuento > Crear' y 'Descuento', y un formulario de entrada. El formulario incluye campos para 'Nombre', 'Descripción', 'Monto s/' y 'Porcentaje', con un botón 'Crear' ubicado al final de la fila de los últimos dos campos.

*Nota.* La figura muestra el bosquejo de la vista de creación y edición de descuentos.

**Tabla 31.**

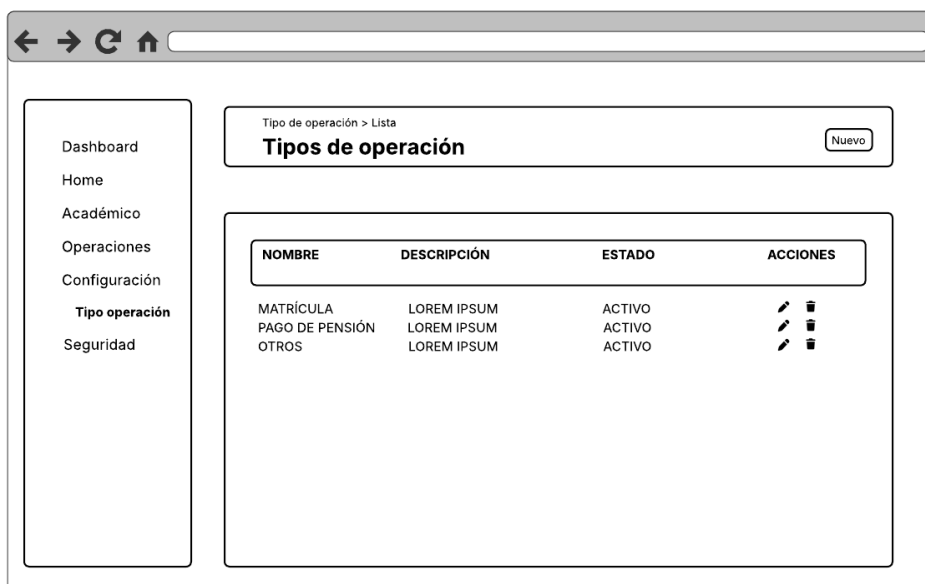
*HU-CONF-007 gestión de tipo de operación*

Historia de usuario	
Código: HU-CONF-007	Nombre: Gestión de tipo de operación
Módulo: Configuración	
Responsable: Equipo de desarrollo	Iteración: sprint 2
Prioridad: Alta	Esfuerzo: 3
Descripción	
Como administrador, quiero configurar tipos de operación (matrícula, pago de pensión) registrar correctamente las transacciones financieras.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se pueden registrar tipos de operación.</li> <li>• Se puede editar o eliminar un tipo de operación.</li> <li>• Se listan todos los tipos de operación existentes.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-CONF-007.

**Figura 48.**

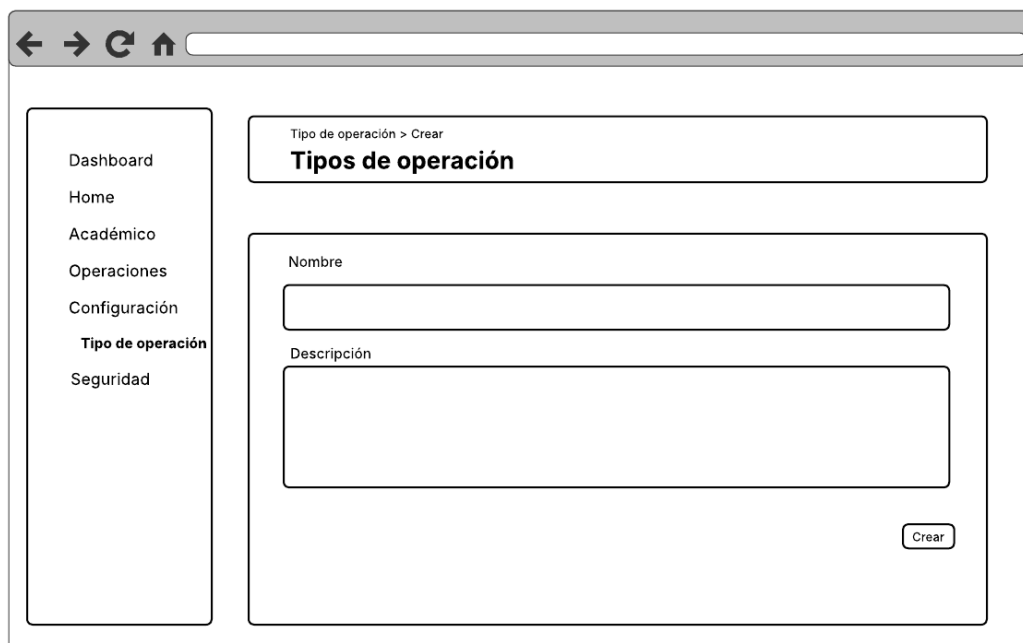
*Wireframe de listado de tipos de operación.*



*Nota.* La figura muestra el bosquejo de la vista principal del menú tipos de operación.

**Figura 49.**

*Wireframe de creación y edición de tipo de operación.*



*Nota.* La figura muestra el bosquejo de la vista de creación y edición de descuentos.

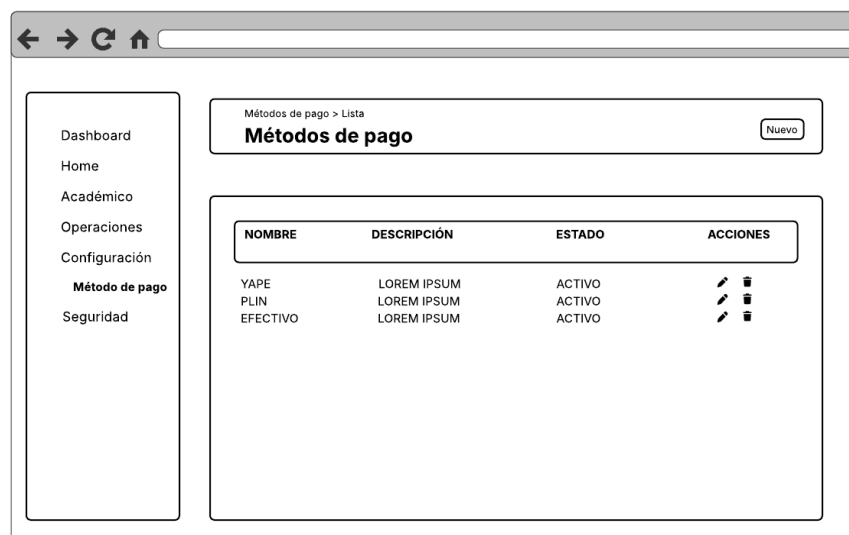
**Tabla 32.***HU-CONF-008 gestión de métodos de pago*

Historia de usuario	
Código: HU-CONF-008	Nombre: Gestión de métodos de pago
Módulo: Configuración	
Responsable: Equipo de desarrollo	Iteración: sprint 2
Prioridad: Alta	Esfuerzo: 3
Descripción	
Como administrador, quiero configurar métodos de pago (efectivo, tarjeta, transferencia) para registrar correctamente las transacciones financieras.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se pueden registrar métodos de pago.</li> <li>• Se puede editar o eliminar un método de pago.</li> <li>• Se listan todos los métodos de pagos existentes.</li> <li>• Los métodos de pago se usan al registrar una operación.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-CONF-008.

**Figura 50.**

*Wireframe de listado de métodos de pago.*



*Nota.* La figura muestra el bosquejo de la vista principal del menú métodos de pago.

**Figura 51.**

*Wireframe de creación y edición métodos de pago.*

*Nota.* La figura muestra el bosquejo de la vista de creación y actualización de métodos de pago.

#### 4.1.22. Taskboard

Para la planificación del sprint 2 se definió una jornada de trabajo de lunes a viernes con una dedicación de 4 horas diarias. Con base de esta capacidad, las tareas del taskboard se calendarizaron en función de su estimación en horas, distribuyéndose de manera continua en los días hábiles.

**Tabla 33.**

*Estimación de trabajo en horas del sprint 2*

Equipo Scrum	Horas de trabajo por día	Horas de Trabajo por semana	Semanas de trabajo por mes	Total de horas	Total de días laborales para el proyecto
Tesista	4	20	4	80	20
Total de días disponible para el proyecto					20

*Nota.* La tabla muestra la estimación de horas disponibles y días laborables para el desarrollo del sprint 2.

**Tabla 34.***Taskboard del sprint 2*

Taskboard del sprint 2								
Inicio: jueves 10 de julio de 2025						Duración: 76 horas		
ID HU	ID TA	Tarea	Responsable	Estado	Horas	Inicio	Fin	
HU-CONF-001	T-023	Diseñar el formulario de datos generales del ciclo académico.	Equipo de desarrollo	Hecho	3	10/07/2025	10/07/2025	
	T-024	Validar fechas (no pasadas, mínimo un mes de duración).	Equipo de desarrollo	Hecho	1	10/07/2025	10/07/2025	
	T-025	Diseñar el formulario de configuración financiera.	Equipo de desarrollo	Hecho	1	11/07/2025	11/07/2025	
	T-026	Implementar las reglas de pago único o por pensiones.	Equipo de desarrollo	Hecho	2	11/07/2025	11/07/2025	
	T-027	Diseñar el formulario de selección de descuentos aplicables.	Equipo de desarrollo	Hecho	2	11/07/2025	13/07/2025	
	T-028	Implementar la navegación entre los pasos del registro del ciclo.	Equipo de desarrollo	Hecho	2	13/07/2025	13/07/2025	
	T-029	Validar que no se pueda eliminar un ciclo con matrículas asociadas.	Equipo de desarrollo	Hecho	1	13/07/2025	13/07/2025	
	T-030	Implementar la edición o actualización de un ciclo académico en curso.	Equipo de desarrollo	Hecho	1	14/07/2025	14/07/2025	

HU-CONF-002	T-031	Probar el flujo completo de registro y edición de un ciclo académico.	Equipo de desarrollo	Hecho	1	14/07/2025	14/07/2025
	T-032	Diseñar la interfaz para la gestión de secciones.	Equipo de desarrollo	Hecho	2	14/07/2025	14/07/2025
	T-033	Implementar la creación de secciones.	Equipo de desarrollo	Hecho	2	15/07/2025	15/07/2025
	T-034	Implementar la edición de una sección.	Equipo de desarrollo	Hecho	2	15/07/2025	15/07/2025
	T-035	Implementar el listado de secciones.	Equipo de desarrollo	Hecho	1	16/07/2025	16/07/2025
	T-036	Implementar la eliminación de una sección.	Equipo de desarrollo	Hecho	1	16/07/2025	16/07/2025
	T-037	Realizar pruebas de aceptación.	Equipo de desarrollo	Hecho	1	16/07/2025	16/07/2025
HU-CONF-003	T-038	Diseñar la interfaz para la gestión de niveles.	Equipo de desarrollo	Hecho	2	17/07/2025	17/07/2025
	T-039	Implementar la creación de niveles.	Equipo de desarrollo	Hecho	2	17/07/2025	17/07/2025
	T-040	Implementar la edición de un nivel.	Equipo de desarrollo	Hecho	2	17/07/2025	18/07/2025
	T-041	Implementar el listado de niveles.	Equipo de desarrollo	Hecho	1	18/07/2025	18/07/2025

	T-042	Implementar la eliminación de un nivel.	Equipo de desarrollo	Hecho	1	18/07/2025	18/07/2025
	T-043	Realizar pruebas de aceptación	Equipo de desarrollo	Hecho	1	18/07/2025	18/07/2025
HU-CONF-004	T-044	Diseñar la interfaz para la gestión de grados.	Equipo de desarrollo	Hecho	2	21/07/2025	21/07/2025
	T-045	Implementar la creación de nuevos grados.	Equipo de desarrollo	Hecho	2	21/07/2025	21/07/2025
	T-046	Implementar la edición de un grado.	Equipo de desarrollo	Hecho	1	22/07/2025	22/07/2025
	T-047	Implementar el listado de niveles.	Equipo de desarrollo	Hecho	1	22/07/2025	22/07/2025
	T-048	Implementar la eliminación de un grado.	Equipo de desarrollo	Hecho	1	22/07/2025	22/07/2025
	T-049	Realizar pruebas de aceptación.	Equipo de desarrollo	Hecho	1	22/07/2025	22/07/2025
HU-CONF-005	T-050	Diseñar la interfaz para la gestión de tipos de matrícula (pantallas de creación, edición y listado).	Equipo de desarrollo	Hecho	2	23/07/2025	23/07/2025
	T-051	Implementar el registro de un tipo de matrícula en el sistema.	Equipo de desarrollo	Hecho	2	23/07/2025	23/07/2025

	T-052	Configurar la edición y actualización de tipos de matrícula existentes.	Equipo de desarrollo	Hecho	2	24/07/2025	24/07/2025
	T-053	Implementar la eliminación de tipos de matrícula.	Equipo de desarrollo	Hecho	1	24/07/2025	24/07/2025
	T-054	Listar tipos de matrícula registrados y mostrar detalles básicos.	Equipo de desarrollo	Hecho	1	24/07/2025	24/07/2025
	T-055	Realizar pruebas de aceptación.	Equipo de desarrollo	Hecho	1	25/07/2025	25/07/2025
HU-CONF-006	T-056	Diseñar la interfaz para la gestión de descuentos (pantallas de creación, edición y listado).	Equipo de desarrollo	Hecho	2	25/07/2025	25/07/2025
	T-057	Implementar el registro de un descuento en el sistema.	Equipo de desarrollo	Hecho	2	25/07/2025	28/07/2025
	T-058	Implementar la edición y actualización de descuentos existentes.	Equipo de desarrollo	Hecho	2	28/07/2025	28/07/2025
	T-059	Implementar la eliminación de descuentos.	Equipo de desarrollo	Hecho	1	28/07/2025	28/07/2025
	T-060	Listar descuentos registrados y mostrar detalles básicos.	Equipo de desarrollo	Hecho	1	29/07/2025	29/07/2025
	T-061	Realizar pruebas de aceptación	Equipo de desarrollo	Hecho	1	29/07/2025	29/07/2025

HU-CONF-007	T-062	Diseñar la interfaz para la gestión de tipo de operación (pantallas de creación, edición y listado).	Equipo de desarrollo	Hecho	2	29/07/2025	29/07/2025
	T-063	Implementar el registro de un tipo de operación en el sistema.	Equipo de desarrollo	Hecho	2	30/07/2025	30/07/2025
	T-064	Implementar la edición y actualización de tipos de operación existentes.	Equipo de desarrollo	Hecho	2	30/07/2025	30/07/2025
	T-065	Implementar la eliminación de tipos de operación.	Equipo de desarrollo	Hecho	1	31/07/2025	31/07/2025
	T-066	Listar tipos de operación registrados y mostrar detalles básicos.	Equipo de desarrollo	Hecho	1	31/07/2025	31/07/2025
	T-067	Realizar pruebas de aceptación	Equipo de desarrollo	Hecho	1	31/07/2025	31/07/2025
	HU-CONF-008	T-068	Diseñar la interfaz para la gestión de métodos de pago pantallas de creación, edición y listado).	Equipo de desarrollo	Hecho	2	31/07/2025
T-069		Implementar el registro de un método de pago en el sistema.	Equipo de desarrollo	Hecho	2	01/08/2025	01/08/2025
T-070		Implementar la edición y actualización de métodos de pago.	Equipo de desarrollo	Hecho	2	01/08/2025	04/08/2025
T-071		Implementar la eliminación de métodos de pago.	Equipo de desarrollo	Hecho	1	04/08/2025	04/08/2025

T-072	Listar métodos de pagos registrados y mostrar detalles básicos.	Equipo de desarrollo	Hecho	1	04/08/2025	04/08/2025
T-073	Realizar pruebas de aceptación.	Equipo de desarrollo	Hecho	1	04/08/2025	04/08/2025

---

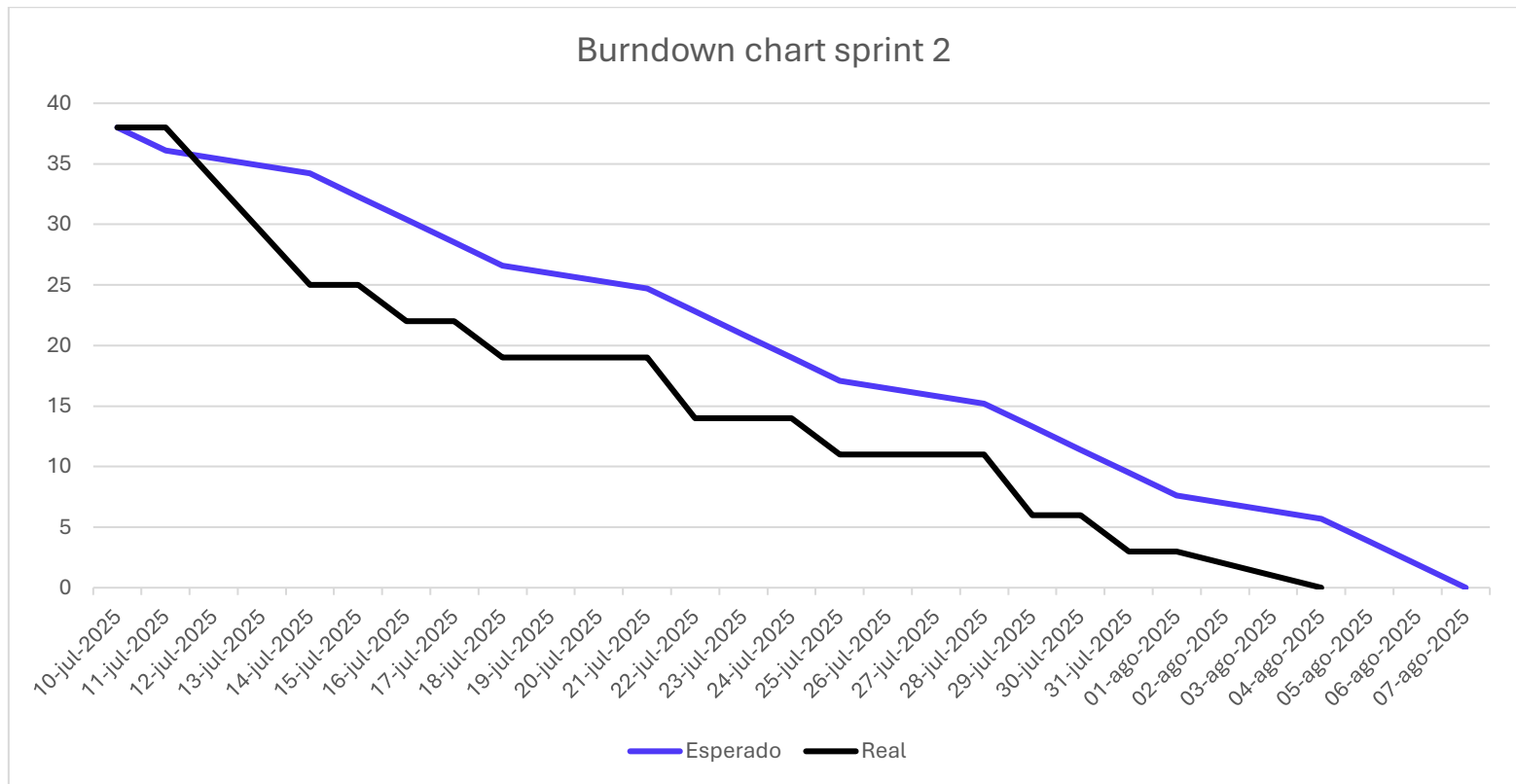
*Nota.* La tabla muestra el taskboard del sprint 2, donde se detalla las tareas ejecutadas en cada historia de usuario.

Para la ejecución del sprint 2 se estimó un tiempo de 80 horas, sin embargo, tal como se observa en la tabla el sprint pudo completarse en 76 horas, el cuál hubo un adelanto de 4 horas.

#### 4.1.23. Burndown chart

**Figura 52.**

*Burndown chart del sprint 2*



*Nota.* El gráfico muestra la comparación entre el progreso esperado y el real durante el desarrollo del sprint 2.

#### 4.1.24. Daily scrum

El daily scrum del sprint 2 se realizó de manera periódica en fechas clave, enfocándose en los avances más significativos de las historias de usuario priorizadas. Dado que el desarrollo fue llevado a cabo por un único integrante, se consolidaron los reportes cada cierto días en lugar de reuniones diarias, evitando redundancias y manteniendo un seguimiento realista del progreso.

**Tabla 35.**

*Daily scrum del sprint 2*

Fecha	¿Qué hice?	¿Qué haré después?	Impedimentos	Observaciones
10/07/2025	Inicié el diseño del formulario del ciclo académico.	Validar los campos, nombre fechas entre otros.	Ninguno	Avance inicial según lo planificado.
14/07/2025	Finalicé la creación de flujo de ciclo a académico en 3 pasos.	Implementar la lógica de reglas de pago único y selección de descuentos.	Ninguno	Ninguno.
22/07/2025	Finalicé la gestión de sección, nivel y grados educativos.	Implementar las historias de usuario HU-CONF-005 Y HU-CONF-006	Ninguno	Ninguno
04/08/2025	Desarrollé las historias HU-CONF-006, HU-CONF-	Realizar pruebas integrales de todo el módulo	Ninguno	Se completaron todas las historias del sprint.

007 y HU- de  
CONF-008. configuración.

*Nota.* La tabla muestra los resultados de los daily scrum correspondientes al sprint 2.

#### 4.1.25. Sprint review

El sprint review del sprint 2 tuvo como propósito evaluar resultados obtenidos en el desarrollo del módulo de configuración, verificando el cumplimiento de los criterios de aceptación definidos en las historias de usuario.

#### Tabla 36.

*Sprint review del sprint 2*

Sprint review	
Fecha:	05/08/2025
Participantes	<ul style="list-style-type: none"> <li>• Scrum Master</li> <li>• Product Owner</li> <li>• Development Team</li> </ul>
Duración	2 horas
Objetivo	Revisar las historias de usuario completadas durante el Sprint 3 y validar su funcionamiento.
Historias de usuario incluidas	<ul style="list-style-type: none"> <li>• HU-CONF-001</li> <li>• HU-CONF -002</li> <li>• HU-CONF -003</li> <li>• HU- CONF -004</li> <li>• HU- CONF -005</li> <li>• HU- CONF -006</li> <li>• HU- CONF -007</li> <li>• HU- CONF -008</li> </ul>
Entregables	<ul style="list-style-type: none"> <li>• Módulo de configuración completado.</li> <li>• Formularios funcionales con validaciones de negocio.</li> </ul>

*Nota.* La tabla muestra el resumen del sprint review correspondiente al sprint 2.

#### 4.1.26. Despliegue del sprint

**Tabla 37.**

*Despliegue del sprint 2*

Despliegue del sprint	
Objetivos	Implementar y desplegar las funcionalidades desarrolladas en el sprint 2 (módulo de configuración), asegurando la correcta integración con la base del sistema y su disponibilidad del entorno de pruebas.
Actividades realizadas	<ul style="list-style-type: none"> <li>• Se integró las nuevas funcionalidades del módulo configuración, que incluye la gestión de ciclo académico, niveles, secciones, grados, tipos de matrícula, descuentos, tipos de operación y métodos de pago.</li> <li>• Se ejecutaron las migraciones y seeders correspondientes para incorporar las nuevas tablas y registros base del módulo.</li> <li>• Se realizaron pruebas funcionales en el entorno de pruebas verificando los flujos CRUD y las validaciones de cada correspondiente.</li> <li>• Se efectuó el despliegue en el entorno de pruebas, asegurando la compatibilidad con la estructura establecida en el sprint 1.</li> </ul>
Resultado	El despliegue del sprint 2 se completó satisfactoriamente, dejando operativo el módulo de configuración dentro del entorno de pruebas.

*Nota.* La tabla detalla el proceso de despliegue del sprint 2 en un servidor de pruebas.

#### 4.1.27. Criterios de aceptación

**Tabla 38.**

*Criterios de aceptación sprint 2*

ID De historia de usuario	Criterio de aceptación	Resultado
HU-CONF-001	<ul style="list-style-type: none"> <li>• El ciclo académico se crea en pasos secuenciales.</li> <li>• En el paso 1 se ingresan los datos generales: nombre, descripción, fecha de inicio y fin.</li> <li>• No se pueden crear ciclos académicos con fechas pasadas.</li> <li>• La duración mínima de un ciclo académico es de un mes.</li> <li>• En el paso 2 se configura la parte financiera: monto de matrícula, monto de pensión, mora por retraso, fechas de pago.</li> <li>• Si el pago es único, solo se permite registrar una única fecha de pago.</li> <li>• En el paso 3 se seleccionan los descuentos existentes que aplicarán al ciclo académico.</li> <li>• No se puede eliminar un ciclo académico que tenga matrículas asociadas.</li> </ul>	<p>Cumple – validado mediante prueba funcional. El formulario por pasos registra correctamente los datos generales, financieros y descuentos. Las restricciones de fechas y eliminaciones se aplicaron conforme a lo esperado.</p>

---

	<ul style="list-style-type: none"> <li>• Se puede editar un ciclo académico en curso, respetando las validaciones de fechas.</li> </ul>	
HU-CONF-002	<ul style="list-style-type: none"> <li>• Se pueden crear secciones de manera independiente.</li> <li>• Se pueden listar todas las secciones activas.</li> <li>• No se permiten secciones duplicadas.</li> </ul>	Cumple – se verificó creación, listado y validación de duplicados.
HU-CONF-003	<ul style="list-style-type: none"> <li>• Se pueden registrar niveles educativos.</li> <li>• Se pueden listar todos los niveles activos.</li> <li>• No se permiten niveles duplicados.</li> </ul>	Cumple – probado con distintos registros; el sistema muestra mensaje de error ante duplicidad.
HU-CONF-004	<ul style="list-style-type: none"> <li>• Se pueden crear grados.</li> <li>• Cada grado se asocia a un nivel y a una sección existente.</li> <li>• Se listan grados con su nivel y sección correspondiente.</li> <li>• No se permiten grados duplicados en el mismo nivel y sección.</li> </ul>	Cumple – relaciones y listados correctamente implementados; validación de duplicidad funcional.
HU-CONF-005	<ul style="list-style-type: none"> <li>• Se pueden registrar diferentes tipos de matrícula.</li> <li>• Se pueden editar los tipos existentes.</li> <li>• Los tipos de matrícula se pueden asignar a estudiantes en el proceso de inscripción</li> </ul>	Cumple – probado en flujo de inscripción; asignación de tipo de matrícula validada.
HU-CONF-006	<ul style="list-style-type: none"> <li>• Se pueden crear distintos tipos de descuento con un porcentaje o monto definido.</li> <li>• Se pueden editar o eliminar descuentos activos.</li> </ul>	Cumple – los descuentos se aplican automáticamente según la configuración.

	<ul style="list-style-type: none"> <li>• Los descuentos se aplican automáticamente al registrar una matrícula.</li> <li>• El descuento se puede aplicar a la matrícula o pensiones.</li> </ul>	
HU-CONF-007	<ul style="list-style-type: none"> <li>• Se pueden registrar tipos de operación.</li> <li>• Se puede editar o eliminar un tipo de operación.</li> <li>• Se listan todos los tipos de operación existentes.</li> </ul>	Cumple – CRUD operativo y validado sin errores.
HU-CONF-008	<ul style="list-style-type: none"> <li>• Se pueden registrar métodos de pago.</li> <li>• Se puede editar o eliminar un método de pago.</li> <li>• Se listan todos los métodos de pagos existentes.</li> <li>• Los métodos de pago se usan al registrar una operación.</li> </ul>	Cumple – CRUD operativo y validado sin errores.

---

*Nota.* La tabla detalla los criterios de aceptación de las historias de usuario correspondiente al sprint 2.

#### 4.1.28. Pruebas unitarias

Para validar el correcto funcionamiento de la aplicación a nivel de código en este segundo sprint, Se tomó 24 unidades de código en total, las cuales tienen por funcionalidad crear, actualizar y eliminar. Estas pruebas dieron como resultado exitoso esto se evidencia en las figuras **Figura 53, Figura 54, Figura 55, Figura 56, Figura 57, Figura 58, Figura 59 y Figura 60.**

**Tabla 39.**

*Prueba unitaria - sprint 2*

No	Historia de usuario	Caso de prueba	Resultado
1	HU-CONF-001	<ul style="list-style-type: none"> <li>• Crear ciclo académico</li> <li>• Editar ciclo académico</li> <li>• Eliminar ciclo académico</li> </ul>	Exitoso
2	HU-CONF-002	<ul style="list-style-type: none"> <li>• Crear sección</li> <li>• Editar sección</li> <li>• Eliminar sección</li> </ul>	Exitoso
3	HU-CONF-003	<ul style="list-style-type: none"> <li>• Crear nivel</li> <li>• Editar nivel</li> <li>• Eliminar nivel</li> </ul>	Exitoso
4	HU-CONF-004	<ul style="list-style-type: none"> <li>• Crear grado</li> <li>• Editar grado</li> <li>• Eliminar grado</li> </ul>	Exitoso
5	HU-CONF-005	<ul style="list-style-type: none"> <li>• Crear tipo de matrícula</li> <li>• Editar tipo de matrícula</li> <li>• Eliminar tipo de matrícula</li> </ul>	Exitoso
6	HU-CONF-006	<ul style="list-style-type: none"> <li>• Crear descuento</li> <li>• Editar descuento</li> <li>• Eliminar descuento</li> </ul>	Exitoso
7	HU-CONF-007	<ul style="list-style-type: none"> <li>• Crear tipo de operación</li> </ul>	Exitoso

	<ul style="list-style-type: none"> <li>• Editar tipo de operación</li> <li>• Eliminar tipo de operación</li> </ul>	
8 HU-CONF-008	<ul style="list-style-type: none"> <li>• Crear método de pago</li> <li>• Editar método de pago</li> <li>• Eliminar método de pago</li> </ul>	Exitoso

*Nota.* La tabla muestra las pruebas unitarias realizadas en cada historia de usuario correspondiente al sprint 1.

### Figura 53.

*Resultado de las pruebas unitarias de gestión de ciclos académicos.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=AcademicCycleTest

PASS Tests\Feature\Configuration\AcademicCycleTest
✓ academic cycle can be created          9.12s
✓ academic cycle can be updated         7.93s
✓ academic cycle can be deleted         9.50s

Tests: 3 passed (11 assertions)
Duration: 26.94s
```

De la **Figura 53** se observa el resultado de las pruebas unitarias de gestión de ciclos académicos, estas pruebas constan de 3 unidades de código, las cuales validan las funcionalidades: creación, actualización y eliminación de ciclos académicos. La prueba da como resultado de aprobado en las 3 unidades de código, asegurando la integridad en el código.

### Figura 54.

*Resultado de las pruebas unitarias de gestión de secciones.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=SectionTest

PASS Tests\Feature\Configuration\SectionTest
✓ section can be created                 7.69s
✓ section can be updated                 7.29s
✓ section can be deleted                 7.18s

Tests: 3 passed (11 assertions)
Duration: 22.49s
```

En la **Figura 54** se observa el resultado de las pruebas unitarias de gestión de secciones, estas pruebas constan de 3 unidades de código, las cuales validan las funcionalidades: creación, actualización y eliminación de secciones. La prueba da como resultado de aprobado en las 3 unidades de código, asegurando la integridad en el código.

**Figura 55.**

*Resultado de las pruebas unitarias de gestión de niveles.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=LevelTest

PASS Tests\Feature\Configuration\LevelTest
✓ level can be created 17.05s
✓ level can be updated 12.76s
✓ level can be deleted 8.90s

Tests: 3 passed (11 assertions)
Duration: 39.00s
```

En la **Figura 55** se observa el resultado de las pruebas unitarias de gestión de niveles, estas pruebas constan de 3 unidades de código, las cuales validan las funcionalidades: creación, actualización y eliminación de niveles. La prueba da como resultado de aprobado en las 3 unidades de código, asegurando la integridad en el código.

**Figura 56.**

*Resultado de las pruebas unitarias de gestión de grados.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=GradeTest

PASS Tests\Feature\Configuration\GradeTest
✓ grade can be created 8.31s
✓ grade can be updated 7.64s
✓ grade can be deleted 7.72s

Tests: 3 passed (9 assertions)
Duration: 23.95s
```

En la **Figura 56** se observa el resultado de las pruebas unitarias de gestión de grados, estas pruebas constan de 3 unidades de código, las cuales validan las funcionalidades: creación, actualización y eliminación de grados. La prueba da como resultado de aprobado en las 3 unidades de código, asegurando la integridad en el código.

**Figura 57.**

*Resultado de las pruebas unitarias de gestión de tipos de matrícula.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=EnrollmentTypeTest

PASS Tests\Feature\Configuration\EnrollmentTypeTest
✓ enrollment type can be created 28.71s
✓ enrollment type can be updated 20.50s
✓ enrollment type can be deleted 7.74s

Tests: 3 passed (8 assertions)
Duration: 57.33s
```

En la **Figura 57** se observa el resultado de las pruebas unitarias de gestión de tipos de matrícula, estas pruebas constan de 3 unidades de código, las cuales validan las funcionalidades:

creación, actualización y eliminación de tipos de matrícula. La prueba da como resultado de aprobado en las 3 unidades de código, asegurando la integridad en el código.

### Figura 58.

*Resultado de las pruebas unitarias de gestión de descuentos.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=DiscountTest
PASS Tests\Feature\Configuration\DiscountTest
✓ discount can be created 8.91s
✓ discount can be updated 13.14s
✓ discount can be deleted 13.23s

Tests: 3 passed (10 assertions)
Duration: 35.66s
```

En la **Figura 58** se observa el resultado de las pruebas unitarias de gestión de tipos de descuentos, estas pruebas constan de 3 unidades de código, las cuales validan las funcionalidades: creación, actualización y eliminación de descuentos. La prueba da como resultado de aprobado en las 3 unidades de código, asegurando la integridad en el código.

### Figura 59.

*Resultado de las pruebas unitarias de gestión de tipos de operación.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=OperationTypeTest
PASS Tests\Feature\Configuration\OperationTypeTest
✓ operation type can be created 13.04s
✓ operation type can be updated 7.89s
✓ operation type can be deleted 8.25s

Tests: 3 passed (12 assertions)
Duration: 29.45s
```

En la **Figura 59** se observa el resultado de las pruebas unitarias de gestión de tipos de operación, estas pruebas constan de 3 unidades de código, las cuales validan las funcionalidades: creación, actualización y eliminación de tipos de operación. La prueba da como resultado de aprobado en las 3 unidades de código, asegurando la integridad en el código.

### Figura 60.

*Resultado de las pruebas unitarias de gestión de métodos de pago.*

```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=PaymentMethodTest
PASS Tests\Feature\Configuration\PaymentMethodTest
✓ payment method can be created 10.20s
✓ payment method can be updated 8.24s
✓ payment method can be deleted 8.41s

Tests: 3 passed (14 assertions)
Duration: 27.32s
```

En la **Figura 60** se observa el resultado de las pruebas unitarias de gestión de tipos de métodos de pago, estas pruebas constan de 3 unidades de código, las cuales validan las funcionalidades: creación, actualización y eliminación de métodos de pago. La prueba da como resultado de aprobado en las 3 unidades de código, asegurando la integridad en el código.

#### 4.1.29. Resultados del sprint

##### Figura 61.

*Interfaz de listado de ciclo académico del módulo configuración.*

INICIO PAGO	FIN PAGO	N° PENSIONES	DESCUENTOS	CREADO POR	ACTUALIZADO POR	F. CREACIÓN	F. ACTUALIZACIÓN	ESTADO	ACCIONES
20/02/2026	20/02/2026	1	HERMANOS MEDIA BECA			10/10/2025	10/10/2025	Activo	
05/04/2026	05/12/2026	9	HERMANOS MEDIA BECA			10/10/2025	10/10/2025	Activo	

*Nota.* La figura muestra la pantalla principal del menú de ciclo académico.

**Figura 62.**

*Interfaz de creación de ciclo académico – paso 1: ciclo académico.*

Ciclo académico > Crear

## Ciclo académico

Ciclo académico

Nombre del ciclo académico

Descripción

Fecha de inicio

Fecha de fin

mm/dd/yyyy

mm/dd/yyyy

> Siguiente

footer

*Nota.* La figura muestra la pantalla de creación de ciclo académico paso 1.

**Figura 63.**

*Interfaz de creación de ciclo académico – paso 2: configuración financiera.*

Ciclo académico > Crear

## Ciclo académico

### Configuración Financiera

Matrícula (\$/)

Pensión (\$/)

Mora por retraso (\$/)

Fechas de pago

Pago único

Fecha de inicio de pago

Fecha de finalización de pago

mm/dd/yyyy

mm/dd/yyyy

Atrás

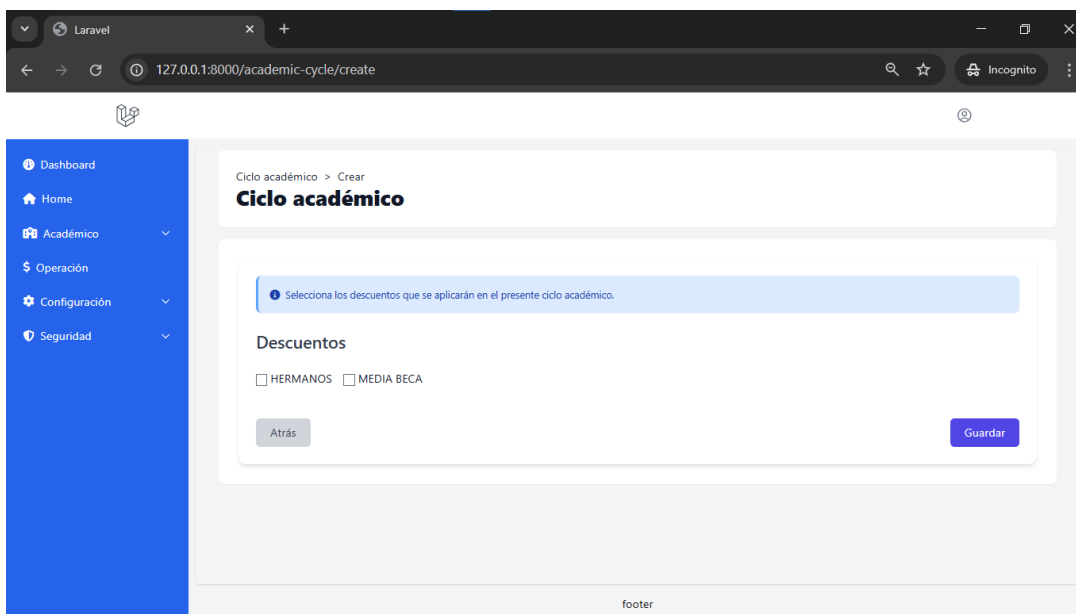
Siguiente

footer

*Nota.* La figura muestra la pantalla de creación de ciclo académico paso 2.

**Figura 64.**

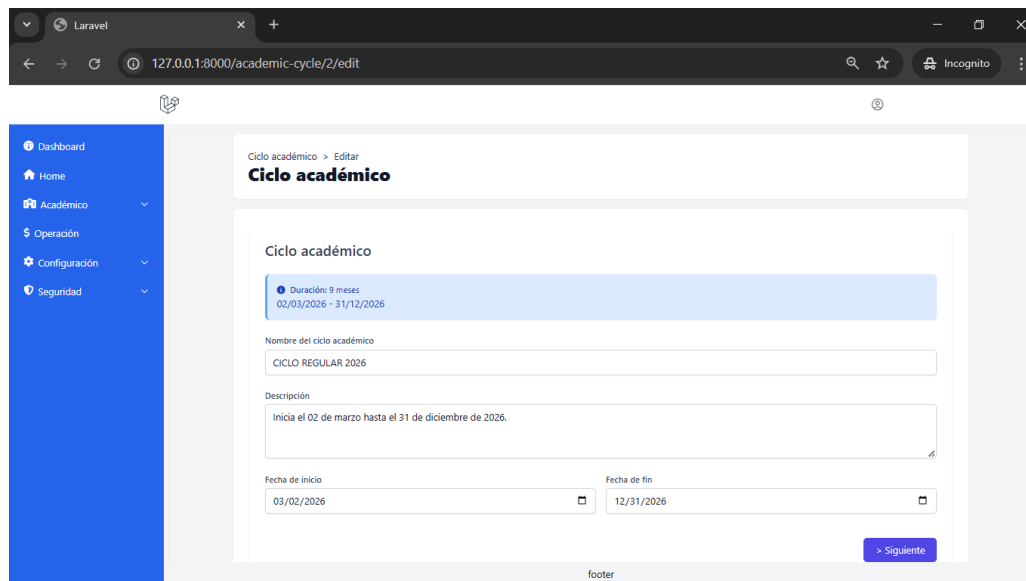
*Interfaz de creación de ciclo académico – paso 3: Descuentos.*



*Nota.* La figura muestra la pantalla de creación de ciclo académico paso 3.

**Figura 65.**

*Interfaz de edición de ciclo académico – paso 1: ciclo académico.*



*Nota.* La figura muestra la pantalla de actualización de ciclo académico paso 1.

**Figura 66.**

*Interfaz de edición de ciclo académico – paso 2: configuración financiera.*

The screenshot shows a web browser window with the URL `127.0.0.1:8000/academic-cycle/2/edit`. The page title is 'Ciclo académico > Editar' and the main heading is 'Ciclo académico'. The form is titled 'Configuración Financiera' and contains the following fields and controls:

- Calendario de pensiones: 05/04/2026 - 05/12/2026 (9 pagos)
- Matrícula (S/): 450.00
- Pensión (S/): 250.00
- Mora por retraso (S/): 0.50
- Fechas de pago:  Pago Único
- Fecha de inicio de pago: 04/05/2026
- Fecha de finalización de pago: 12/05/2026
- Buttons: 'Atrás' and 'Siguiente'

*Nota.* La figura muestra la pantalla de actualización de ciclo académico paso 2.

**Figura 67.**

*Interfaz de edición de ciclo académico – paso 3: descuentos.*

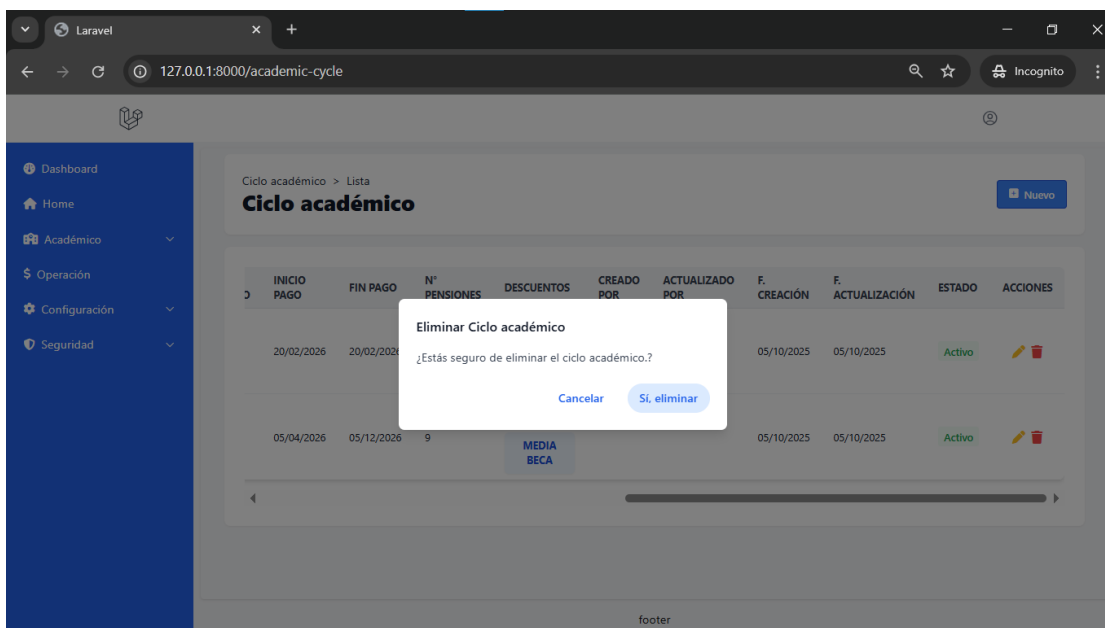
The screenshot shows the same web browser window as Figure 66, but the form is titled 'Descuentos'. It contains the following fields and controls:

- Selección de descuentos: HERMANOS (checked), MEDIA BECA (checked)
- Buttons: 'Atrás' and 'Guardar'

*Nota.* La figura muestra la pantalla de actualización de ciclo académico paso 3.

**Figura 68.**

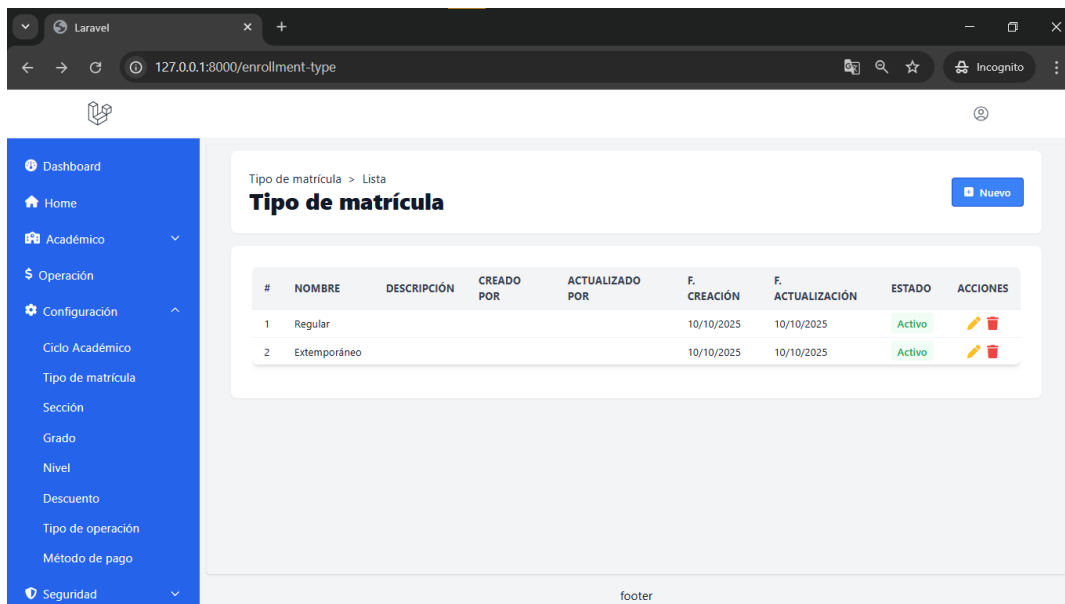
*Interfaz de eliminado de ciclo académico.*



*Nota.* La figura muestra un modal que permite eliminar o anular un ciclo académico.

**Figura 69.**

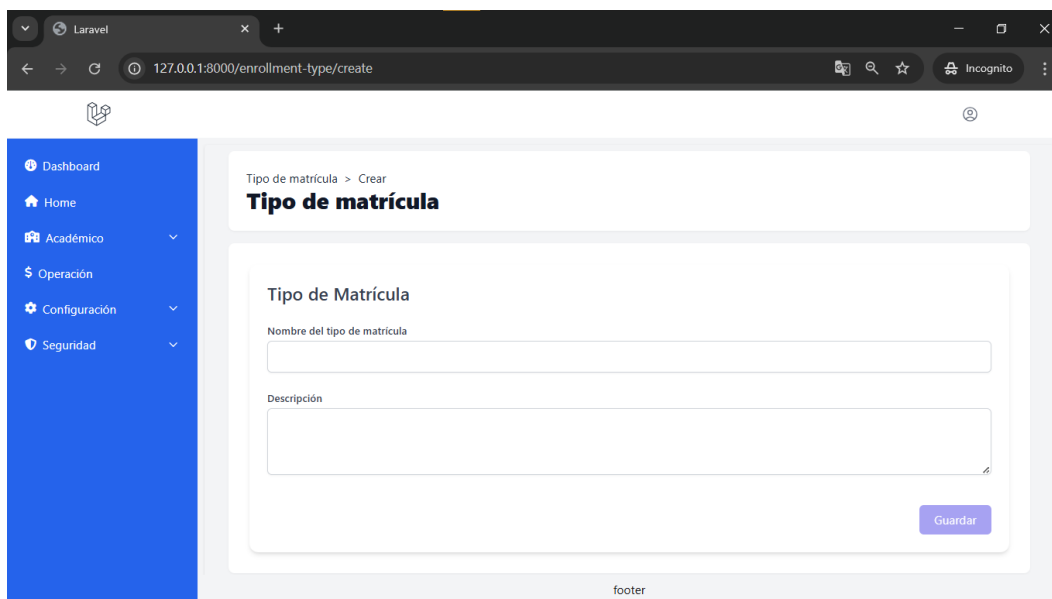
*Interfaz de listado de tipo de matrícula.*



*Nota.* La figura muestra la pantalla principal del menú tipo de matrícula.

**Figura 70.**

*Interfaz de creación de tipo de matrícula.*

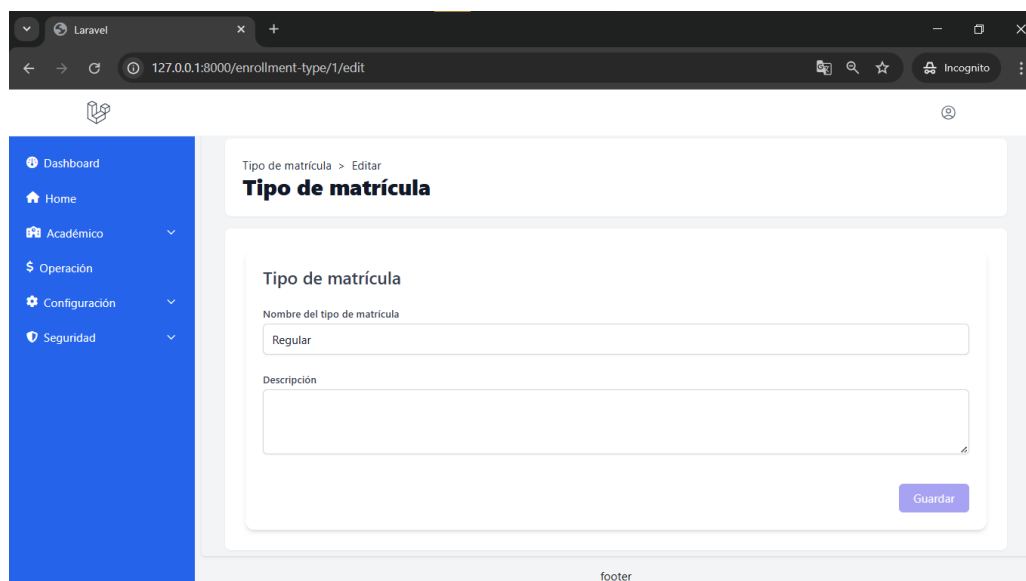


The screenshot shows a web browser window with the URL `127.0.0.1:8000/enrollment-type/create`. The page features a blue sidebar menu on the left with options: Dashboard, Home, Académico, Operación, Configuración, and Seguridad. The main content area is titled 'Tipo de matrícula > Crear' and 'Tipo de matrícula'. It contains a form with two input fields: 'Nombre del tipo de matrícula' and 'Descripción'. A purple 'Guardar' button is located at the bottom right of the form. The footer of the page contains the word 'footer'.

*Nota.* La figura muestra la pantalla de creación de un tipo de matrícula.

**Figura 71.**

*Interfaz de edición de tipo de matrícula.*

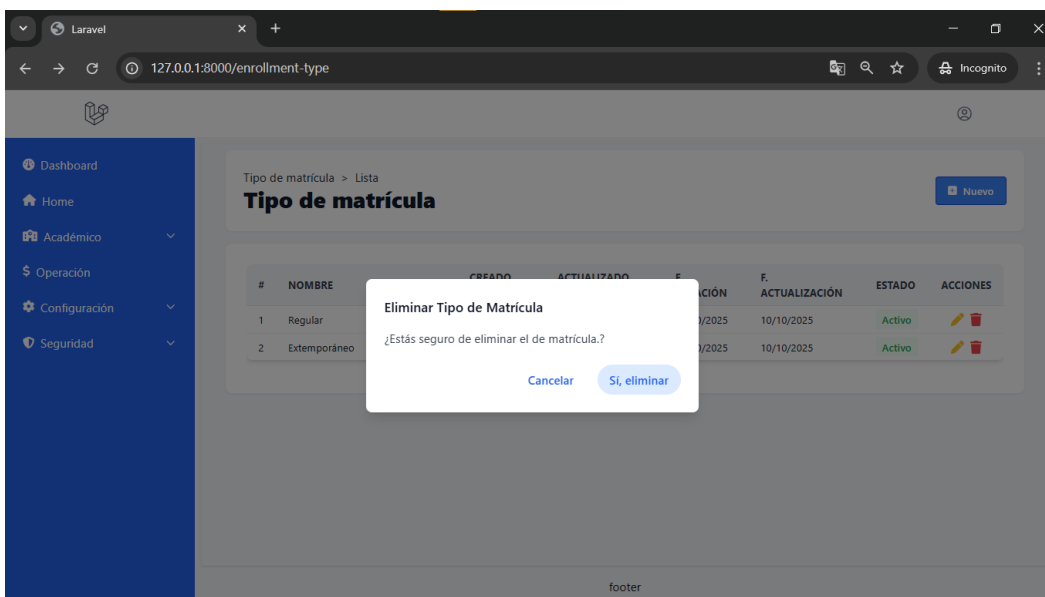


The screenshot shows a web browser window with the URL `127.0.0.1:8000/enrollment-type/1/edit`. The page features a blue sidebar menu on the left with options: Dashboard, Home, Académico, Operación, Configuración, and Seguridad. The main content area is titled 'Tipo de matrícula > Editar' and 'Tipo de matrícula'. It contains a form with two input fields: 'Nombre del tipo de matrícula' (containing the text 'Regular') and 'Descripción'. A purple 'Guardar' button is located at the bottom right of the form. The footer of the page contains the word 'footer'.

*Nota.* La figura muestra la pantalla de actualización de un tipo de matrícula.

**Figura 72.**

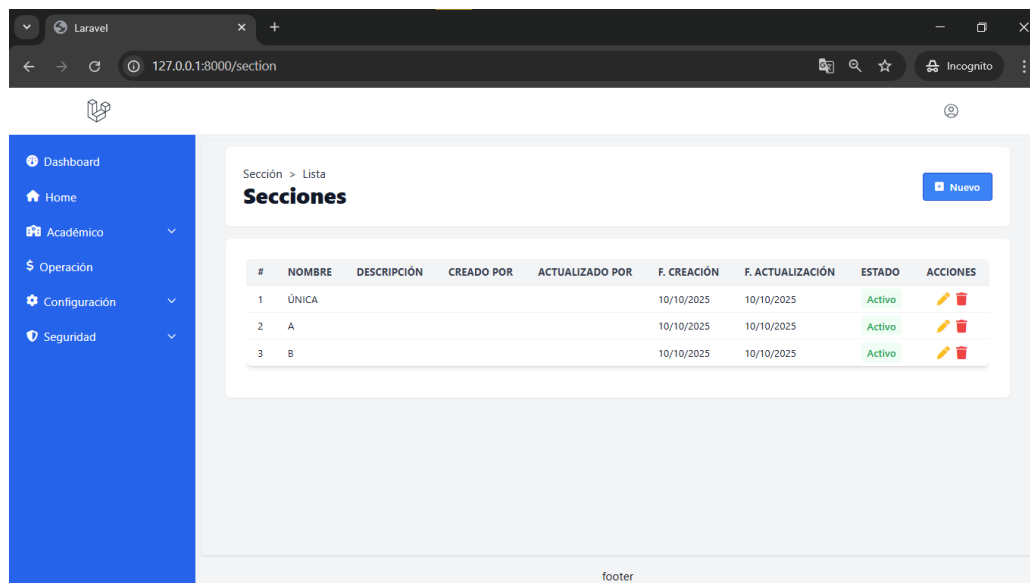
*Interfaz de eliminado de tipo de matrícula.*



*Nota.* La figura muestra un modal que permite eliminar un tipo de matrícula.

**Figura 73.**

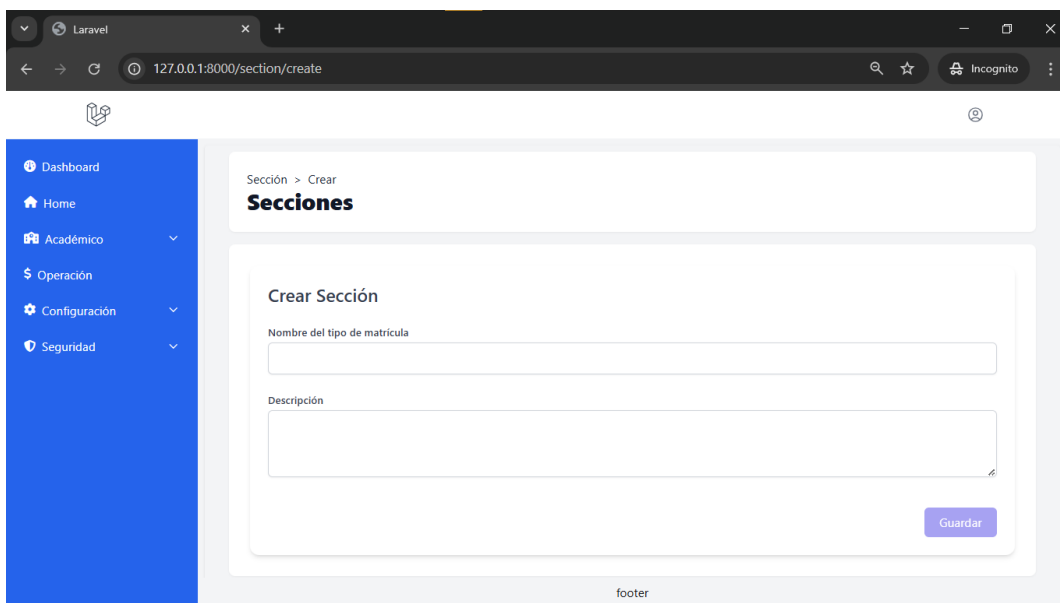
*Interfaz de listado de sección.*



*Nota.* La figura muestra la pantalla principal del menú secciones.

**Figura 74.**

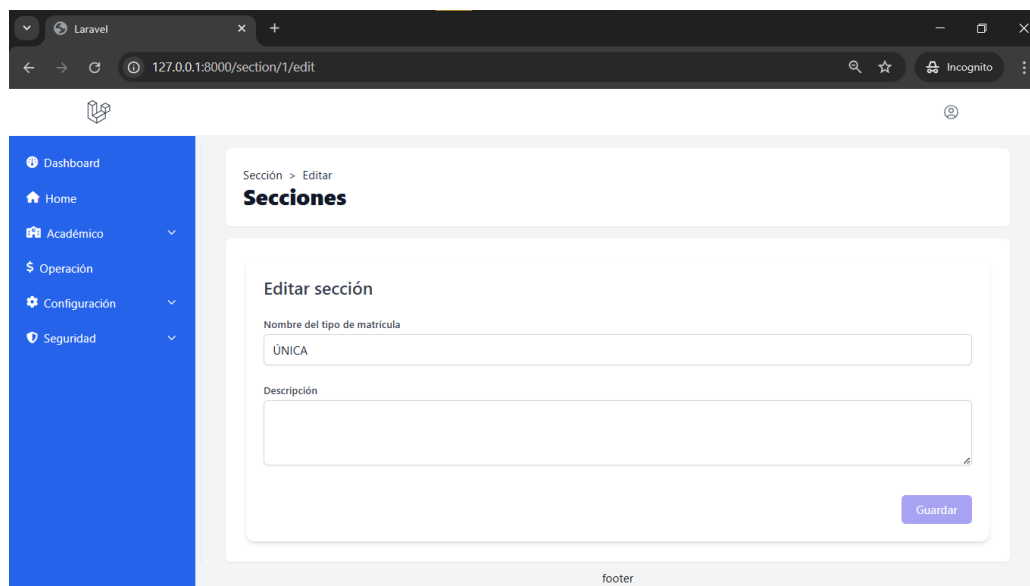
*Interfaz de creación de sección*



*Nota.* La figura muestra la pantalla de creación de una sección.

**Figura 75.**

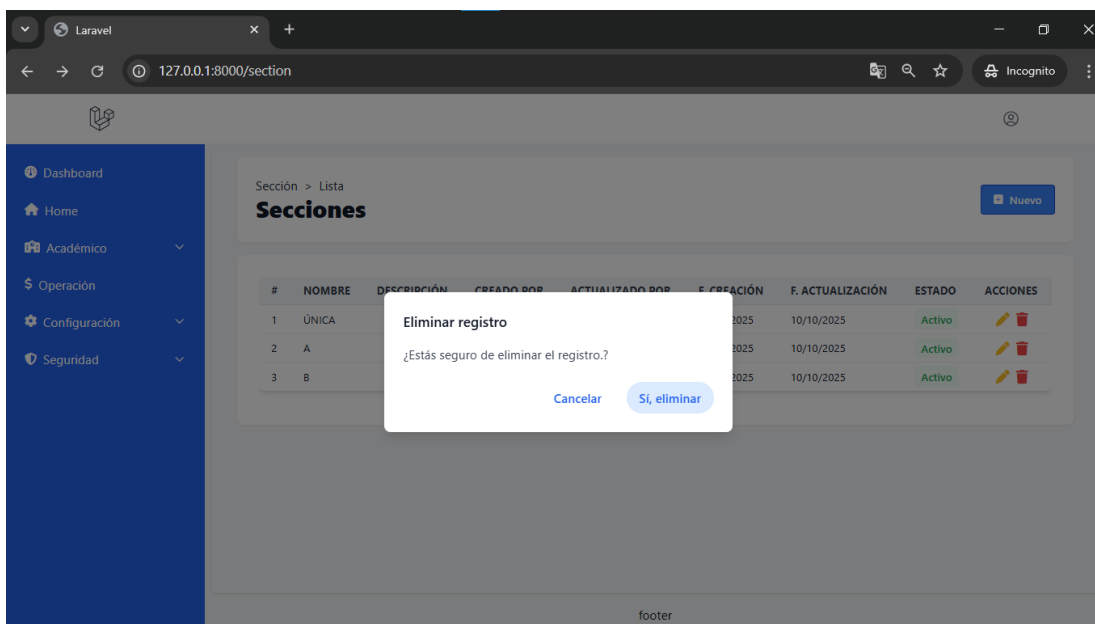
*Interfaz de edición de sección.*



*Nota.* La figura muestra la pantalla de actualización de un tipo de matrícula.

**Figura 76.**

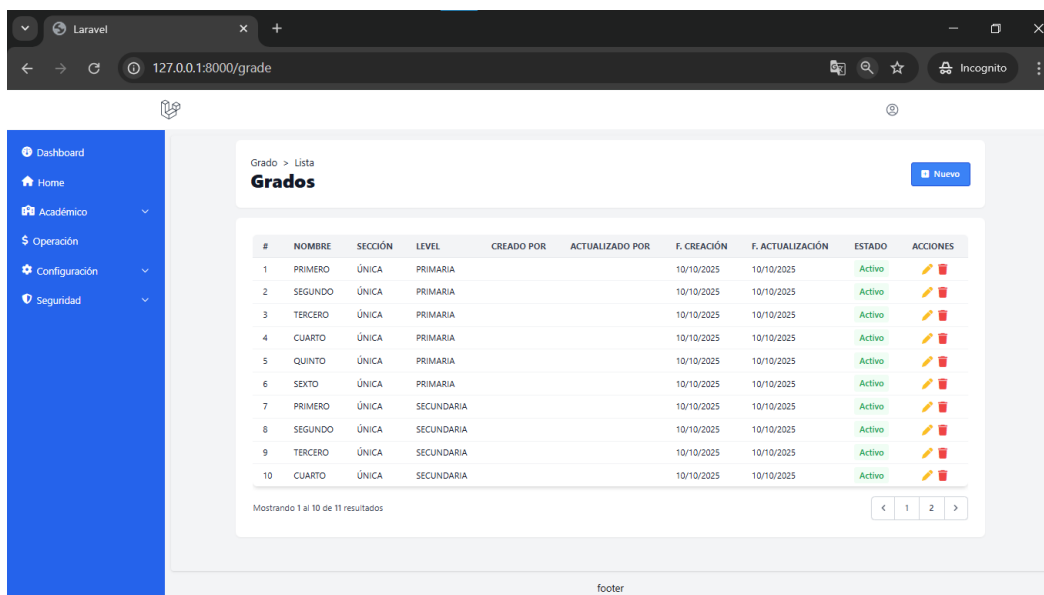
*Interfaz de eliminado de sección.*



*Nota.* La figura muestra un modal que permite eliminar un tipo de matrícula.

**Figura 77.**

*Interfaz de listado de grado.*



*Nota.* La figura muestra la pantalla principal del menú grados.

**Figura 78.**

*Interfaz de creación de grado.*

Grado > Crear

## Grados

Nombre del grado

Sección

Nivel

INICIAL

PRIMARIA

SECUNDARIA

Guardar

footer

*Nota.* La figura muestra la pantalla de creación de grados.

**Figura 79.**

*Interfaz de edición de grado.*

Grado > Editar

## Grados

Nombre del grado

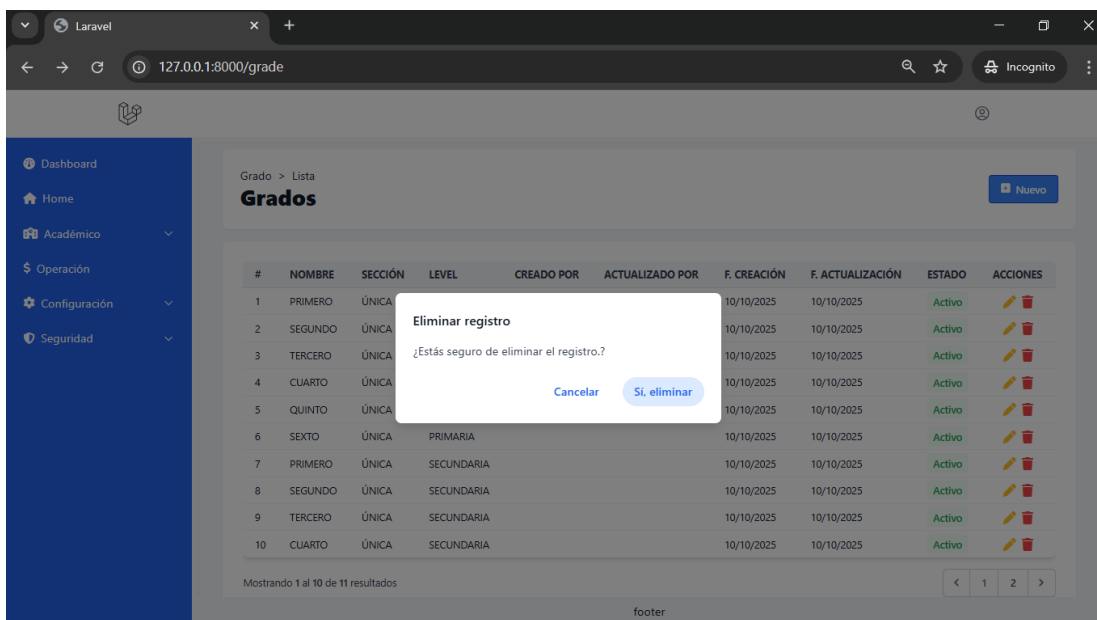
Sección

Nivel

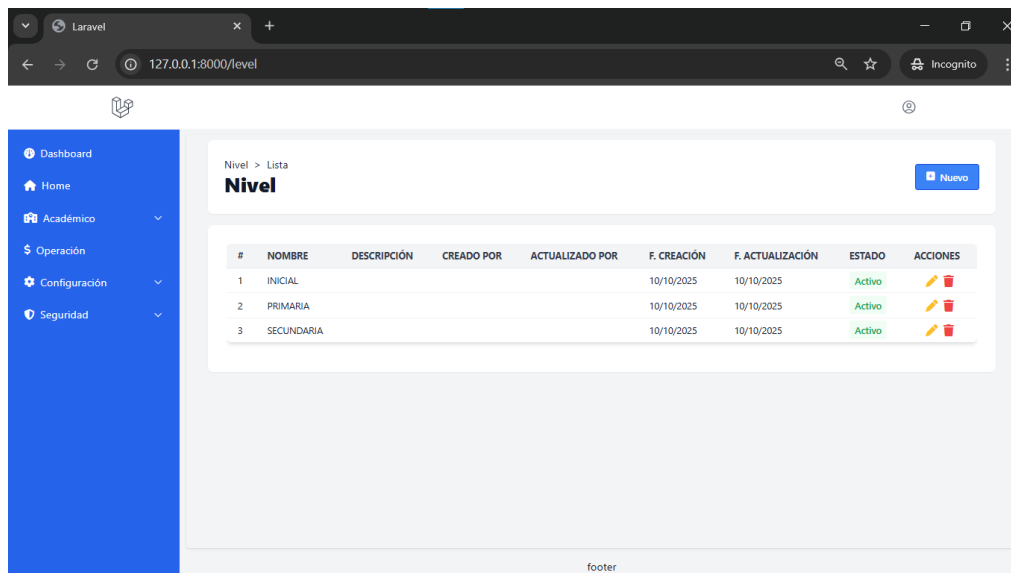
Guardar

footer

*Nota.* La figura muestra la pantalla de actualización de grados.

**Figura 80.***Interfaz de eliminado de grado.*

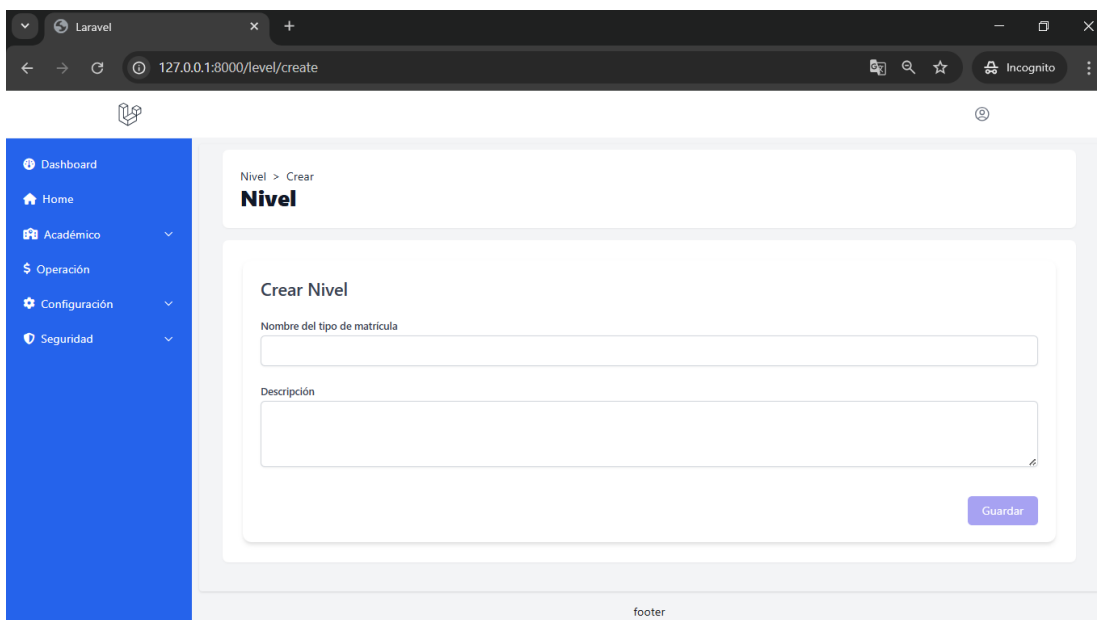
*Nota.* La figura muestra un modal que permite eliminar grados.

**Figura 81.***Interfaz de listado de nivel.*

*Nota.* La figura muestra la pantalla principal del menú niveles.

**Figura 82.**

*Interfaz de creación de nivel.*

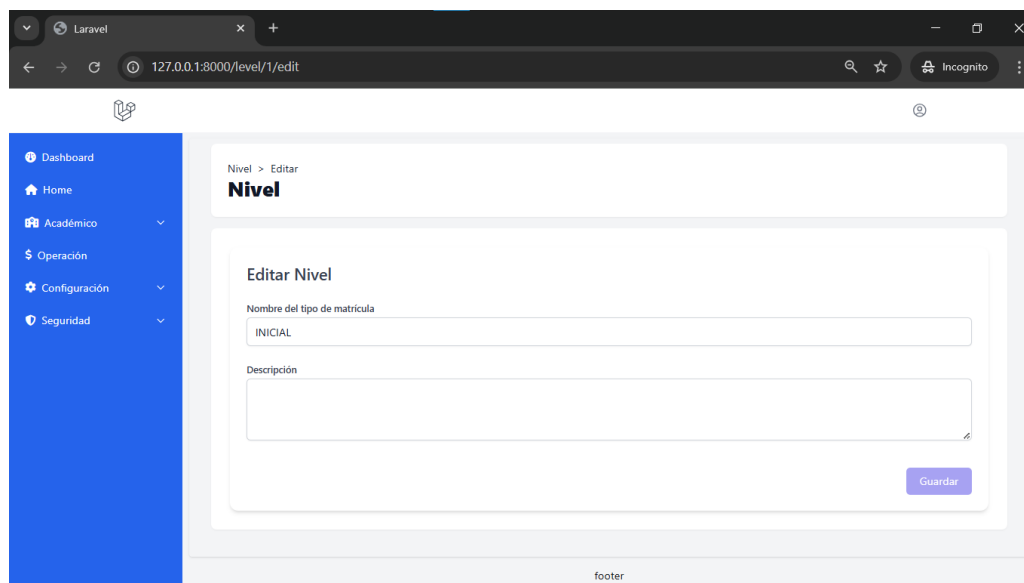


The screenshot shows a web browser window with the URL `127.0.0.1:8000/level/create`. The page features a blue sidebar on the left with navigation links: Dashboard, Home, Académico, Operación, Configuración, and Seguridad. The main content area is titled 'Nivel > Crear' and 'Nivel'. Below this, there is a form titled 'Crear Nivel' with two input fields: 'Nombre del tipo de matrícula' and 'Descripción'. A purple 'Guardar' button is located at the bottom right of the form. The footer of the page contains the word 'footer'.

*Nota.* La figura muestra la pantalla de creación de niveles.

**Figura 83.**

*Interfaz de edición de nivel.*

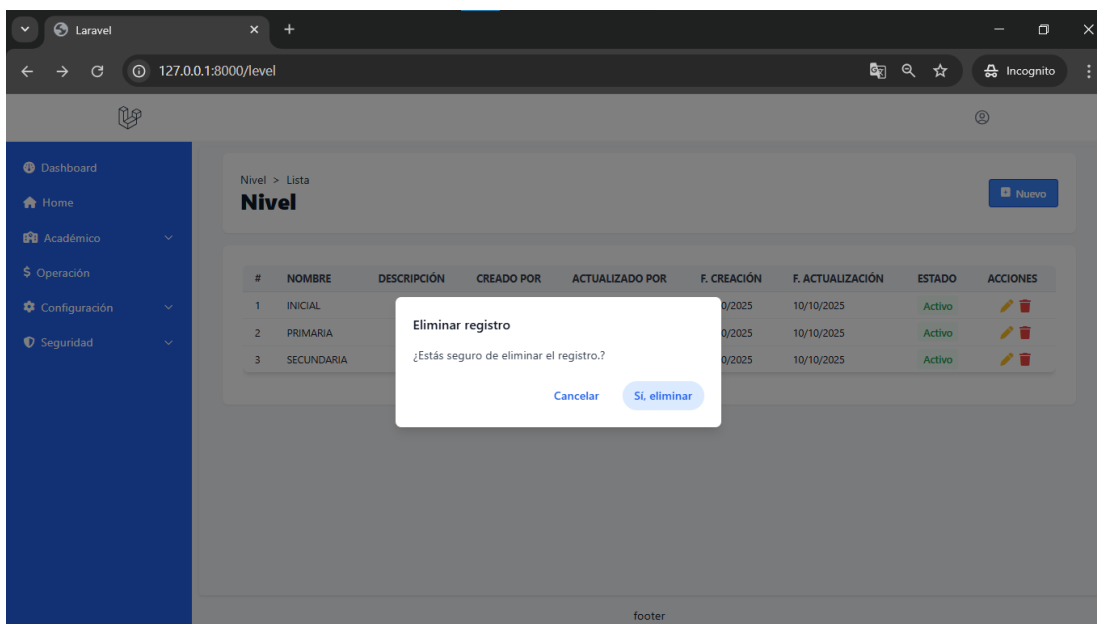


The screenshot shows a web browser window with the URL `127.0.0.1:8000/level/1/edit`. The page features a blue sidebar on the left with navigation links: Dashboard, Home, Académico, Operación, Configuración, and Seguridad. The main content area is titled 'Nivel > Editar' and 'Nivel'. Below this, there is a form titled 'Editar Nivel' with two input fields: 'Nombre del tipo de matrícula' (containing the text 'INICIAL') and 'Descripción'. A purple 'Guardar' button is located at the bottom right of the form. The footer of the page contains the word 'footer'.

*Nota.* La figura muestra la pantalla de actualización de niveles.

**Figura 84.**

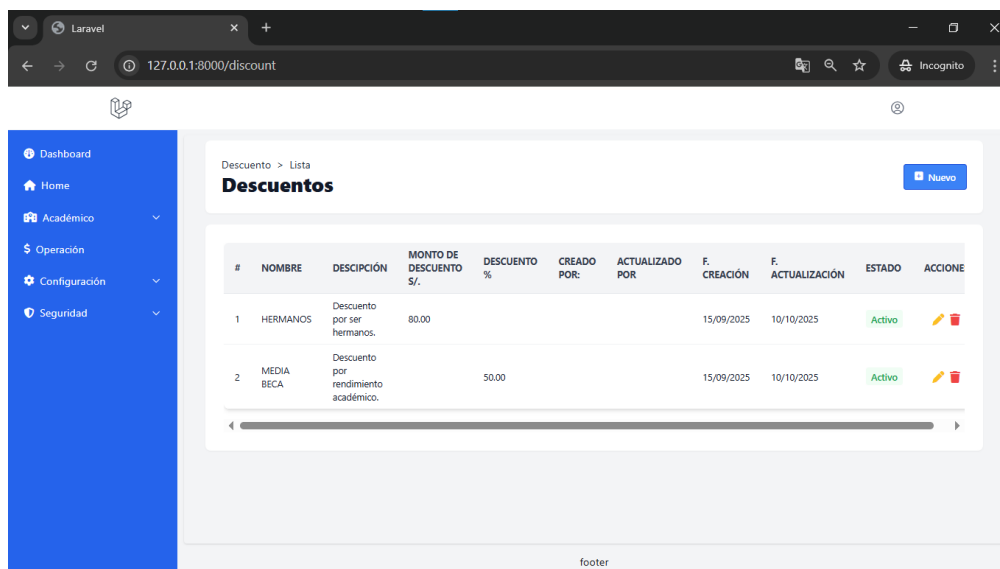
*Interfaz de eliminado de nivel.*



*Nota.* La figura muestra un modal que permite eliminar niveles.

**Figura 85.**

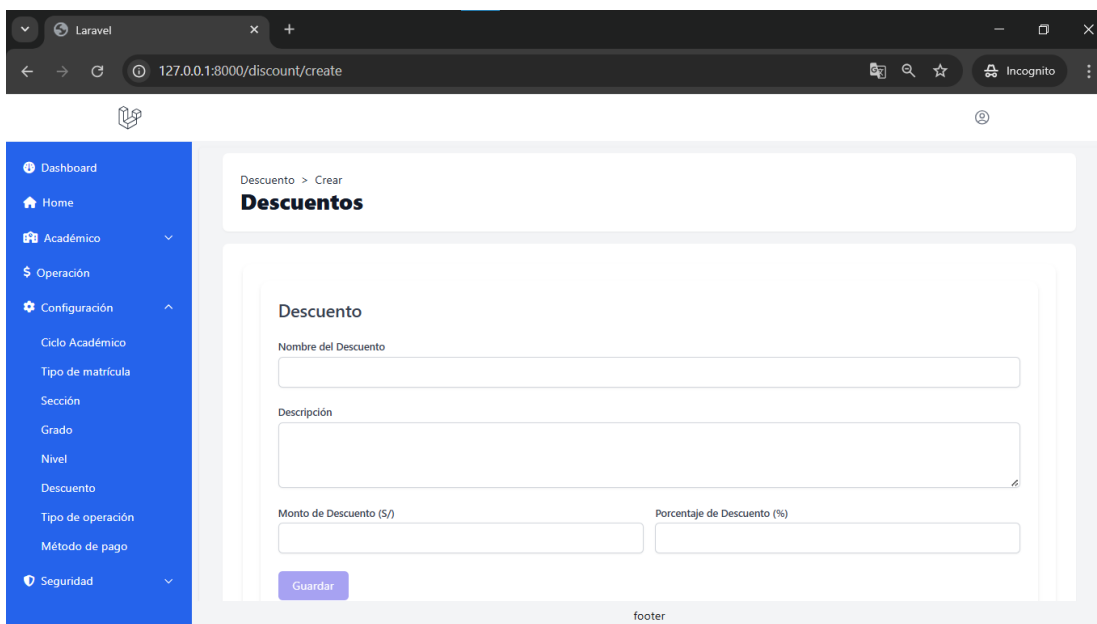
*Interfaz de listado de descuento.*



*Nota.* La figura muestra la pantalla principal del menú descuentos.

**Figura 86.**

*Interfaz de creación de descuento.*

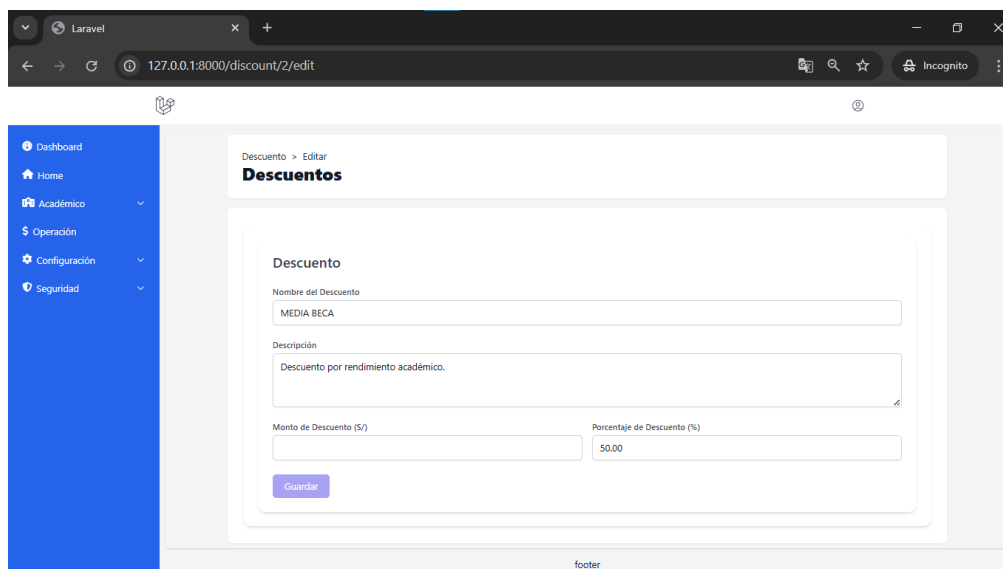


The screenshot shows a web browser window with the URL `127.0.0.1:8000/discount/create`. The page title is "Descuento > Crear" and the main heading is "Descuentos". The interface includes a blue sidebar menu with options like "Dashboard", "Home", "Académico", "Operación", "Configuración", and "Seguridad". The main content area contains a form titled "Descuento" with the following fields: "Nombre del Descuento" (empty), "Descripción" (empty), "Monto de Descuento (\$)" (empty), and "Porcentaje de Descuento (%)" (empty). A purple "Guardar" button is located at the bottom of the form. The footer of the page contains the word "footer".

*Nota.* La figura muestra la pantalla de creación de descuentos.

**Figura 87.**

*Interfaz de edición de descuento.*

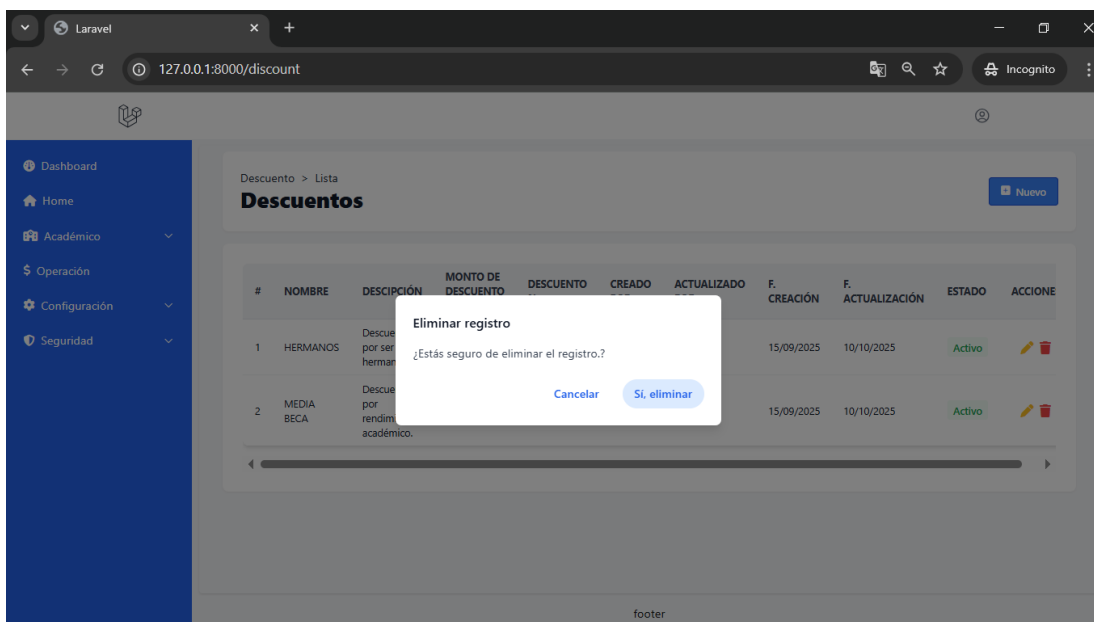


The screenshot shows a web browser window with the URL `127.0.0.1:8000/discount/2/edit`. The page title is "Descuento > Editar" and the main heading is "Descuentos". The interface includes a blue sidebar menu with options like "Dashboard", "Home", "Académico", "Operación", "Configuración", and "Seguridad". The main content area contains a form titled "Descuento" with the following fields: "Nombre del Descuento" (containing "MEDIA BECA"), "Descripción" (containing "Descuento por rendimiento académico."), "Monto de Descuento (\$)" (empty), and "Porcentaje de Descuento (%)" (containing "50.00"). A purple "Guardar" button is located at the bottom of the form. The footer of the page contains the word "footer".

*Nota.* La figura muestra la pantalla de actualización de descuentos.

**Figura 88.**

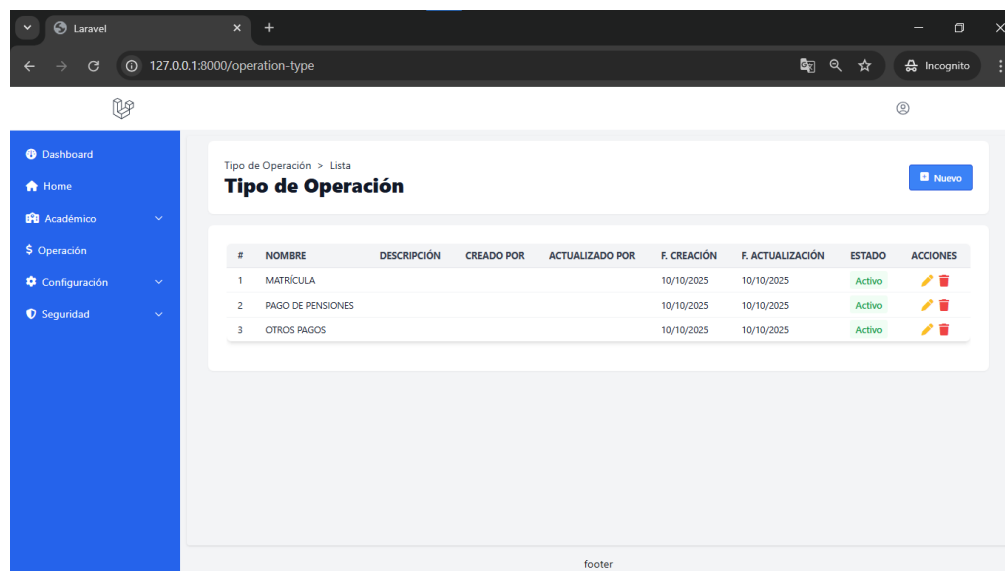
*Interfaz de eliminado de descuento.*



*Nota.* La figura muestra un modal que permite eliminar descuentos.

**Figura 89.**

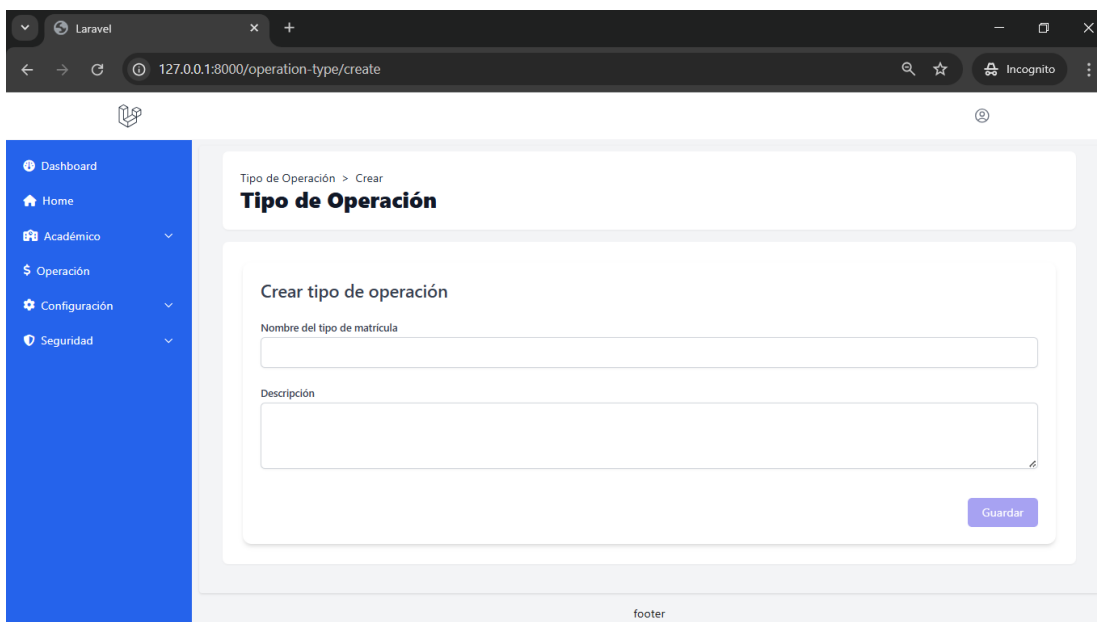
*Interfaz de listado de tipo de operación.*



*Nota.* La figura muestra la pantalla principal del menú tipo de operación.

**Figura 90.**

*Interfaz de creación de tipo de operación*

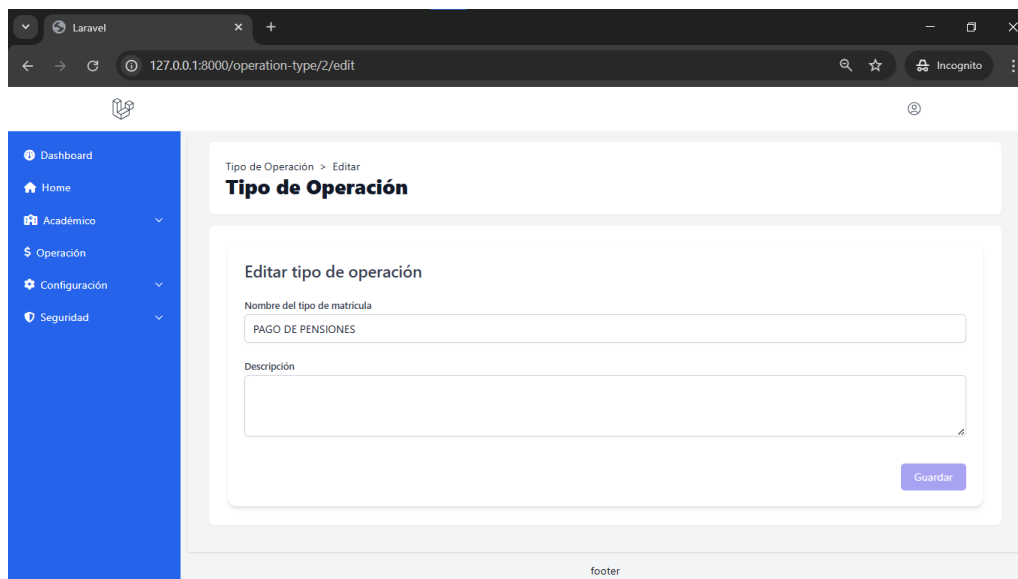


The screenshot shows a web browser window with the URL `127.0.0.1:8000/operation-type/create`. The page features a blue sidebar on the left with navigation links: Dashboard, Home, Académico, Operación, Configuración, and Seguridad. The main content area is titled 'Tipo de Operación > Crear' and 'Tipo de Operación'. Below this, there is a form titled 'Crear tipo de operación' with two input fields: 'Nombre del tipo de matrícula' and 'Descripción'. A 'Guardar' button is located at the bottom right of the form. The browser's address bar shows 'Incognito' mode.

*Nota.* La figura muestra la pantalla de creación de un tipo de operación.

**Figura 91.**

*Interfaz de edición de tipo de operación.*

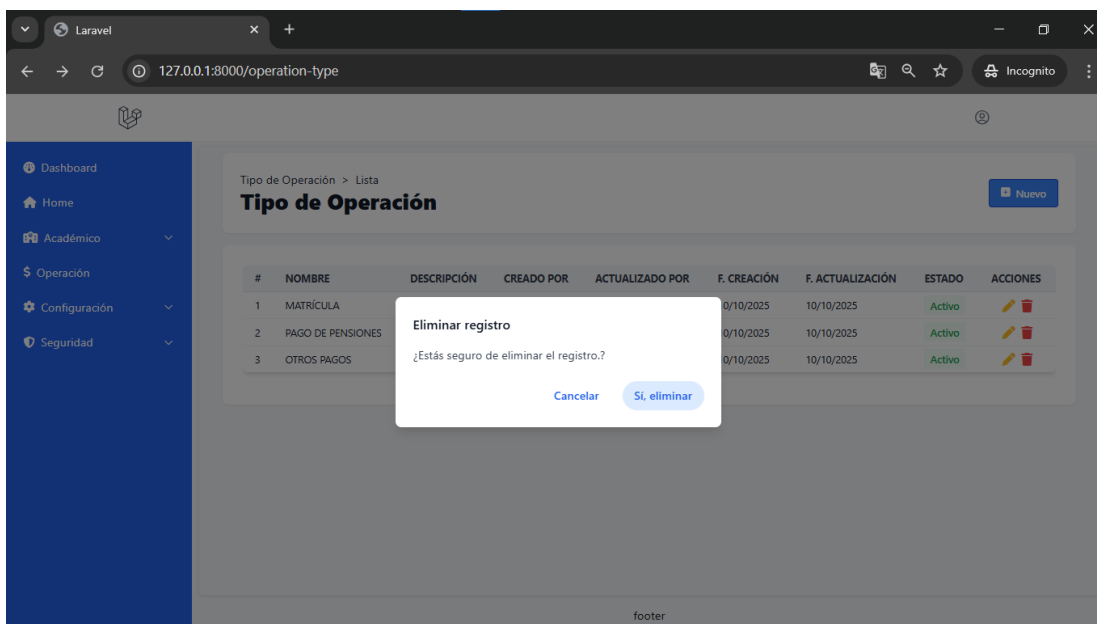


The screenshot shows a web browser window with the URL `127.0.0.1:8000/operation-type/2/edit`. The page features a blue sidebar on the left with navigation links: Dashboard, Home, Académico, Operación, Configuración, and Seguridad. The main content area is titled 'Tipo de Operación > Editar' and 'Tipo de Operación'. Below this, there is a form titled 'Editar tipo de operación' with two input fields: 'Nombre del tipo de matrícula' (containing the text 'PAGO DE PENSIONES') and 'Descripción'. A 'Guardar' button is located at the bottom right of the form. The browser's address bar shows 'Incognito' mode.

*Nota.* La figura muestra la pantalla de actualización de un tipo de operación.

**Figura 92.**

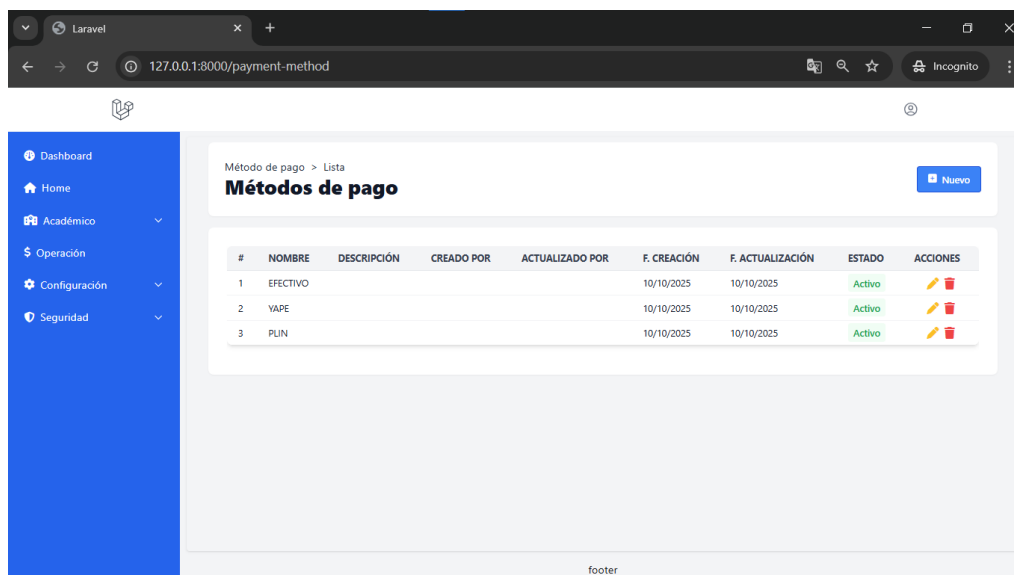
*Interfaz de eliminado de tipo de operación*



*Nota.* La figura muestra un modal que permite eliminar un tipo de operación.

**Figura 93.**

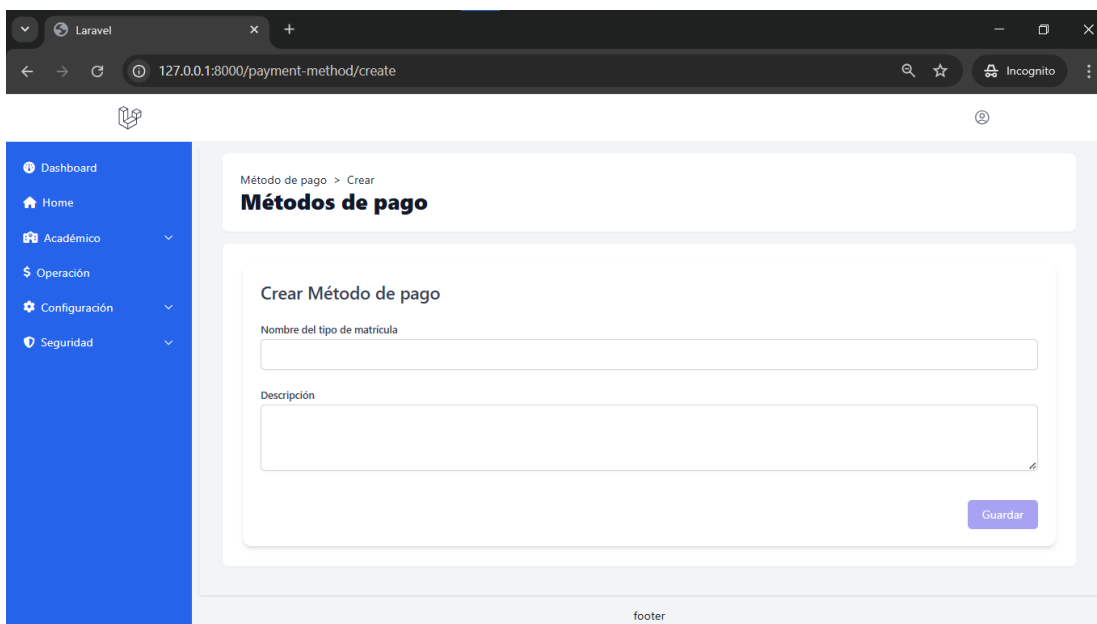
*Interfaz de listado de método de pago.*



*Nota.* La figura muestra la pantalla principal del menú métodos de pago.

**Figura 94.**

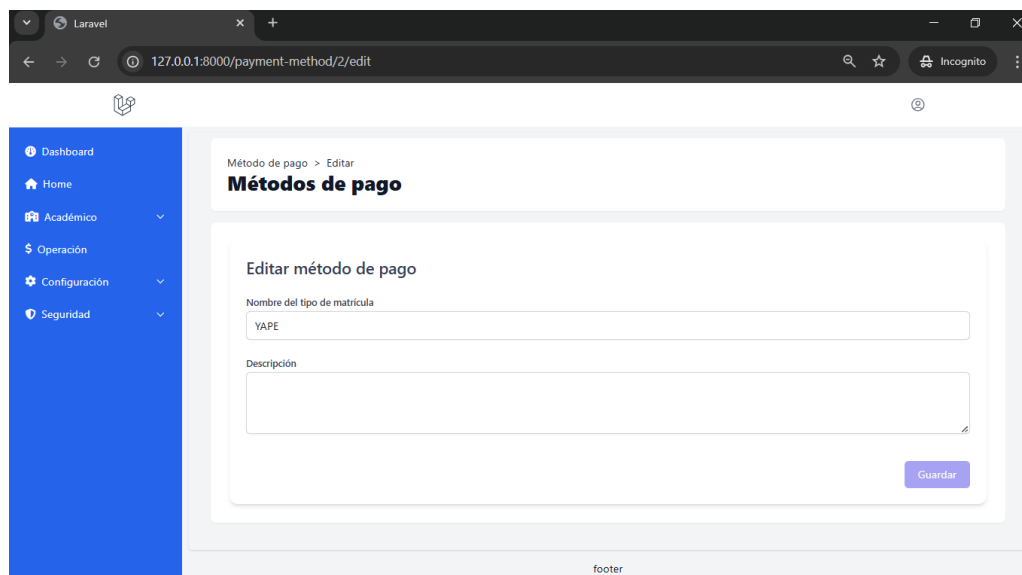
*Interfaz de creación de método de pago.*



*Nota.* La figura muestra la pantalla de creación de un método de pago.

**Figura 95.**

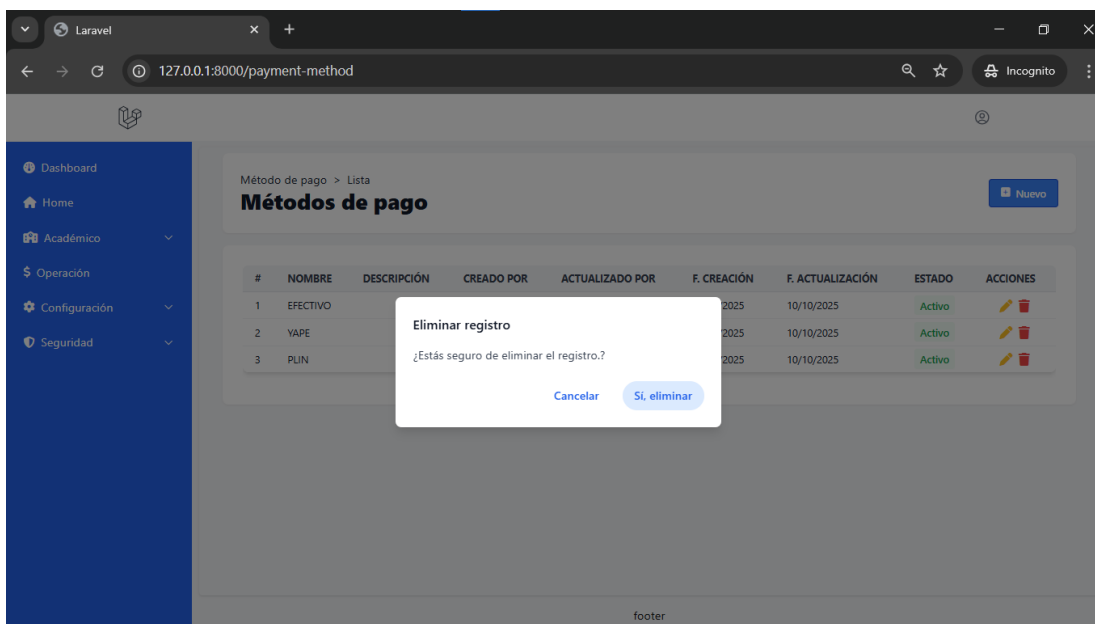
*Interfaz de edición de método de pago.*



*Nota.* La figura muestra la pantalla de actualización de métodos de pago.

**Figura 96.**

*Interfaz de eliminado de método de pago.*



*Nota.* La figura muestra un modal que permite eliminar métodos de pago.

#### 4.1.30. Sprint retrospective

Durante la retrospectiva del sprint 2, se evaluó el rendimiento general del desarrollo. A pesar de haber planificado 40 horas de trabajo, el sprint se completó en 36 horas, lo que evidencia una mejora en la eficiencia y dominio de las herramientas utilizadas. Se identificaron oportunidades de mejora relacionadas con la documentación técnica y el refinamiento de estimaciones de esfuerzo.

**Tabla 40.**

*Sprint retrospective del sprint 2*

Sprint retrospective	
Fecha:	05/08/2025
Participantes	<ul style="list-style-type: none"> <li>• Scrum master</li> <li>• Product owner</li> <li>• Development team</li> </ul>

Objetivo	Analizar el desarrollo del Sprint 2 para identificar aciertos, áreas de mejora y definir acciones que optimicen el desempeño en el Sprint 3.
Aspectos positivos	<ul style="list-style-type: none"> <li>• Se culminaron todas las historias de usuario planificadas dentro del tiempo estimado (36 horas de 40).</li> <li>• Mejor organización en la planificación y priorización de tareas.</li> <li>• Fluidez en la integración de módulos del sistema.</li> <li>• Reducción de retrabajos gracias a pruebas tempranas.</li> </ul>
Aspectos para mejorar	<ul style="list-style-type: none"> <li>• Algunas validaciones funcionales requirieron ajustes de último momento.</li> <li>• Se podría mejorar la estimación de esfuerzo en algunas tareas menores.</li> </ul>
Acciones de mejora para el siguiente sprint	<ul style="list-style-type: none"> <li>• Incorporar revisión técnica intermedia antes del cierre de cada historia.</li> <li>• Documentar cada módulo inmediatamente después de su finalización.</li> <li>• Ajustar los criterios de estimación en el Planning Poker para reflejar mejor la complejidad real.</li> </ul>

---

*Nota.* La tabla muestra el resultado de la retrospectiva del sprint 2.

## 1.17. Sprint 3

### 4.1.31. Sprint planning

En esta tercera y última planificación del sprint, tuvo el objetivo de desarrollar el módulo de operación y académico, considerado una de las partes más críticas del sistema, ya que integra los procesos de matrícula, y pagos. Durante la sesión de planificación, se revisó el producto backlog y se seleccionaron las historias de usuario con prioridad alta, alineadas con los requerimientos funcionales definidos en las fases anteriores.

Se establecieron como entregables las funcionalidades correspondientes a las historias HU-OPER-001 a HU-OPER-005 (módulo operación) y HU-ACA-001, HU-ACA-002, HU-

ACA-003 (módulo académico), cuya finalización permitirá disponer de un sistema plenamente funcional.

La prioridad y esfuerzo se determinaron considerando el impacto funcional, la complejidad técnica y la relevancia operativa de cada historia. Todas las historias del sprint fueron clasificadas con prioridad alta, ya que son indispensables para el flujo principal del sistema: sin ellas, no sería posible registrar estudiantes, procesar pagos o calcular deudas, actividades que constituyen el núcleo transaccional de la aplicación.

La estimación de esfuerzo se realizó mediante la técnica de planning poker: Las historias de HU-OPER-001 (registro de matrícula), HU-OPER-003 (pago de pensiones) y HU-OPER-005 (gestión de deudas y mora) obtuvieron una valoración de 13 puntos por su alta complejidad técnica, al involucrar múltiples pasos, lógica, validaciones y conexión con distintos módulos del sistema. La historia HU-OPER-002 (editar matrícula) fue valorada con 8 puntos, por requerir validaciones y actualizaciones de datos ya registrados. Finalmente, HU-OPER-004 (anular operación) recibió 5 puntos, al implicar lógica de control y verificación de roles, pero con un flujo menos extenso. Esta estimación equilibró la carga de trabajo del sprint, priorizando el desarrollo de funcionalidades críticas y de mayor valor para la operación institucional.

Por su parte, las historias correspondientes al módulo académico (HU-ACA-001 a HU-ACA-003) se valoraron entre 2 y 3 puntos, al tratarse de consultas y reportes de menor complejidad técnica, centradas en la visualización y filtrado de información.

#### **Tabla 41.**

##### *Sprint planning del sprint 3*

Reunión de planificación de sprint 3	
Fecha	06/08/2025
Participantes	<ul style="list-style-type: none"> <li>• Product owner</li> <li>• Scrum team</li> <li>• Scrum master</li> </ul>
Artefactos involucrados	<ul style="list-style-type: none"> <li>• Product backlog</li> <li>• Backlog de sprint</li> </ul>
Agenda	<ul style="list-style-type: none"> <li>• Revisión del producto backlog y definición de objetivos del sprint.</li> </ul>

	<ul style="list-style-type: none"><li>• Selección de historias de prioridad alta.</li><li>• Definición de tareas principales por historia de usuario.</li><li>• Estimación de esfuerzo con planning poker.</li><li>• Determinación de la capacidad del sprint.</li></ul>
Objetivos	<ul style="list-style-type: none"><li>• Desarrollar el módulo de operación.</li><li>• Completar las historias de usuario HU-OPER-001, HU-OPER-002, HU-OPER-003, HU-OPER-004, HU-OPER-005. HU-ACA-001, HU-ACA-002 y HU-ACA-003</li></ul>

---

*Nota.* La tabla resume la reunión de planificación del sprint 3.

#### 4.1.32. Sprint backlog

**Tabla 42.**

*Product backlog del sprint 3*

ID de la historia	Descripción	Módulo	Sprint	Prioridad	Esfuerzo
HU-OPER-001	Como secretaria, quiero registrar la matrícula de un estudiante mediante un proceso por pasos, para almacenar su información personal, académica y de pago.	Operación	Propuesto	Alta	13
HU-OPER-002	Como secretaria, quiero editar los datos de una matrícula ya registrada mediante un proceso similar al registro, para corregir errores o actualizar información del estudiante, contacto o datos financieros.	Operación	Propuesto	Alta	8
HU-OPER-003	Como secretaria, quiero registrar el pago de pensiones de un estudiante mediante un proceso por pasos (búsqueda, selección de pensiones y facturación), para llevar un control financiero actualizado y confiable.	Operación	Propuesto	Alta	13
HU-OPER-004	Como administrador, quiero anular operaciones registradas en el sistema (matrículas o pagos de pensiones) en casos justificados, de modo que se conserve el historial, pero no se consideren como activas en los reportes.	Operación	Propuesto	Alta	5
HU-OPER-005	Como administrador, quiero que el sistema calcule de manera automática la mora diaria en caso de retraso en el pago de	Operación	Propuesto	Alta	13

pensiones, en base a la configuración financiera del ciclo académico, para reflejar correctamente la deuda del estudiante y mantener información precisa para la gestión de cobros.

HU-ACA-001	Como director, quiero visualizar un listado de estudiantes matriculados para obtener información general del ciclo académico.	Académico	4	Media	3
HU-ACA-002	Como secretaria, quiero filtrar estudiantes por ciclo académico para obtener información de un año específico.	Académico	4	Media	2
HU-ACA-003	Como administrador, quiero visualizar estudiantes con pensiones vencidas para realizar el seguimiento de cobros.	Académico	4	Media	3

---

*Nota.* La tabla muestra el sprint backlog del sprint 1, agrupa las historias de usuario del módulo operación y académico.

#### 4.1.33. Historias de usuario

**Tabla 43.**

*HU-OPER-001 registro de matrícula*

Historia de usuario	
Código: HU-OPER-001	Nombre: Registro de matrícula
Módulo: Operación	
Responsable: Equipo de desarrollo	Iteración: sprint 3
Prioridad: Alta	Esfuerzo: 13
Descripción	
Como secretaria, quiero registrar la matrícula de un estudiante mediante un proceso por pasos, para almacenar su información personal, académica y de pago.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• El proceso de matrícula se realiza en pasos secuenciales.</li> <li>• En el paso 1 se ingresan datos personales.</li> <li>• En el paso 2 se ingresan datos de documento de identidad.</li> <li>• En el paso 3 se ingresa contacto de emergencia.</li> <li>• En el paso 4 se ingresa contacto del estudiante.</li> <li>• En el paso 5 se registran datos académicos.</li> <li>• En el paso 6 se registran datos de facturación.</li> <li>• Se muestra un resumen de matrícula con el detalle de pago y descuentos aplicados.</li> <li>• La matrícula queda registrada correctamente al finalizar el flujo.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-OPER-001.

**Tabla 44.***HU-OPER-002 editar matrícula*

Historia de usuario	
Código: HU-OPER-002	Nombre: Editar matrícula
Módulo: Operación	
Responsable: Equipo de desarrollo	Iteración: sprint 3
Prioridad: Alta	Esfuerzo: 8
Descripción	
Como secretaria, quiero editar los datos de una matrícula ya registrada mediante un proceso similar al registro, para corregir errores o actualizar información del estudiante, contacto o datos financieros.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se valida que los datos obligatorios no queden incompletos tras la edición.</li> <li>• Se muestra confirmación al usuario de que la matrícula fue actualizada correctamente.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-OPER-002.

**Figura 97.**

*Wireframe de registro y edición de matrícula - paso 1.*

The wireframe shows a web browser window with a navigation bar at the top containing back, forward, refresh, and home icons, followed by a search bar. On the left side, there is a vertical sidebar menu with the following items: Dashboard, Home, Académico, Operación (highlighted), Configuración, and Seguridad. The main content area is titled 'Operación > Crear Matrícula'. Below this title, there is a section titled 'Datos personales' containing four input fields: 'Sexo' with a dropdown menu showing '--Seleccione--', 'Nombre', 'Apellido paterno', and 'Apellido materno'. Below these fields is a 'Dirección' input field. At the bottom of the form, there are two buttons: 'Atrás' on the left and 'Siguiente' on the right.

*Nota.* Bosquejo de la vista de creación y actualización de una matrícula paso 1.

**Figura 98.**

*Wireframe de registro y edición de matrícula paso 2.*

The wireframe shows a web browser window with a navigation bar at the top containing back, forward, refresh, and home icons, followed by a search bar. On the left side, there is a vertical sidebar menu with the following items: Dashboard, Home, Académico, Operación (highlighted), Configuración, and Seguridad. The main content area is titled 'Operación > Crear Matrícula'. Below this title, there is a section titled 'Documento' containing two input fields: 'Tipo de documento' with a dropdown menu showing '--Seleccione--' and 'Número'. Below these fields, there are two buttons: 'Atrás' on the left and 'Siguiente' on the right.

*Nota.* Bosquejo de la vista de creación y actualización de una matrícula paso 2.

**Figura 99.**

*Wireframe de registro y edición de matrícula paso 3.*

The wireframe shows a browser window with a navigation bar at the top containing back, forward, refresh, and home icons. On the left is a sidebar menu with the following items: Dashboard, Home, Académico, **Operación**, Configuración, and Seguridad. The main content area is titled 'Operación > Crear **Matrícula**'. Below this is a section titled 'Contacto de emergencia' containing the following form fields: 'Nombre de contacto' (text input), 'Parentesco' (dropdown menu with '--Seleccione--'), 'Celular' (text input), 'Teléfono' (text input), and 'Email' (text input). At the bottom of the form are two buttons: 'Atrás' and 'Siguiente'.

*Nota.* Bosquejo de la vista de creación y actualización de una matrícula paso 3.

**Figura 100.**

*Wireframe de registro y edición de matrícula paso 4.*

The wireframe shows a browser window with a navigation bar at the top containing back, forward, refresh, and home icons. On the left is a sidebar menu with the following items: Dashboard, Home, Académico, **Operación**, Configuración, and Seguridad. The main content area is titled 'Operación > Crear **Matrícula**'. Below this is a section titled 'Contacto' containing the following form fields: 'Celular' (text input), 'Teléfono' (text input), and 'Email' (text input). At the bottom of the form are two buttons: 'Atrás' and 'Siguiente'.

*Nota.* Bosquejo de la vista de creación y actualización de una matrícula paso 4.

**Figura 101.**

*Wireframe de registro y edición de matrícula paso 5.*

The wireframe shows a web browser window with a navigation menu on the left and a main content area. The navigation menu includes: Dashboard, Home, Académico, **Operación**, Configuración, and Seguridad. The main content area has a breadcrumb 'Operación > Crear' and a title 'Matrícula'. Below this is a section titled 'Datos académicos' containing three dropdown menus: 'Ciclo académico' (with '--Seleccione--'), 'Grado' (with '--Seleccione--'), and 'Tipo de matrícula' (with '--Seleccione--'). At the bottom of this section are two buttons: 'Atrás' and 'Siguiente'.

*Nota.* Bosquejo de la vista de creación y actualización de una matrícula paso 5.

**Figura 102.**

*Wireframe de registro y edición de matrícula paso 6.*

The wireframe shows a web browser window with a navigation menu on the left and a main content area. The navigation menu includes: Dashboard, Home, Académico, **Operación**, Configuración, and Seguridad. The main content area has a breadcrumb 'Operación > Crear' and a title 'Matrícula'. Below this is a section titled 'Datos de facturación' containing four input fields: 'Razón social', 'RUC / DNI', 'Descuento' (with '--Seleccione--'), and 'Método de pago' (with '--Seleccione--'). At the bottom of this section are two buttons: 'Atrás' and 'Siguiente'.

*Nota.* Bosquejo de la vista de creación y actualización de una matrícula paso 5.

**Tabla 45.***HU-OPER-003 pago de pensiones*

Historia de usuario	
Código: HU-OPER-003	Nombre: Pago de pensiones
Módulo: Operación	
Responsable: Equipo de desarrollo	Iteración: sprint 3
Prioridad: Alta	Esfuerzo: 13
Descripción	
Como secretaria, quiero registrar el pago de pensiones de un estudiante mediante un proceso por pasos (búsqueda, selección de pensiones y facturación), para llevar un control financiero actualizado y confiable.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se busca al estudiante por DNI y ciclo académico.</li> <li>• Se muestran todas las pensiones pendientes y la opción de pagar por adelantado.</li> <li>• Se pueden seleccionar una o más pensiones para el pago.</li> <li>• Se muestra un resumen del pago con subtotales y total a pagar.</li> <li>• Se ingresan los datos de facturación (nombre/razón social, RUC/DNI).</li> <li>• Se selecciona el método de pago.</li> <li>• El pago se confirma y se registra en el sistema.</li> <li>• El estado de la deuda del estudiante se actualiza automáticamente.</li> </ul>	

Nota. La tabla detalla la historia de usuario HU-OPER-003.

**Figura 103.**

*Wireframe de pago de pensión - paso 1.*

*Nota.* Bosquejo de la vista de pago de pensiones paso 1.

**Figura 104.**

*Wireframe de pago de pensión - paso 2.*

Pensión	Mes	Vence	Estado	Acciones
1	marzo	28/03/2026	PAGADO	
2	Abril	28/03/2026	PENDIENTE	<input checked="" type="checkbox"/>
3	mayo	28/03/2026	PENDIENTE	<input checked="" type="checkbox"/>
4	junio	28/03/2026	PENDIENTE	<input type="checkbox"/>
5	julio	28/03/2026	PENDIENTE	<input type="checkbox"/>
6	agosto	28/03/2026	PENDIENTE	<input type="checkbox"/>

*Nota.* Bosquejo de la vista de pago de pensiones paso 2.

**Figura 105.**

*Wireframe de pago de pensión - paso 3.*

*Nota.* Bosquejo de la vista de pago de pensiones paso 3.

**Tabla 46.**

*HU-OPER-004 anular operación*

Historia de usuario	
Código: HU-OPER-004	Nombre: Anular operación
Módulo: Operación	
Responsable: Equipo de desarrollo	Iteración: sprint 3
Prioridad: Alta	Esfuerzo: 5
Descripción	
Como administrador, quiero anular operaciones registradas en el sistema (matrículas o pagos de pensiones) en casos justificados, de modo que se conserve el historial, pero no se consideren como activas en los reportes.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>Solo usuarios con rol autorizado pueden anular operaciones.</li> <li>La anulación solicita confirmación antes de proceder.</li> </ul>	

- Una matrícula anulada no se considera en reportes académicos ni financieros.
- Un pago de pensión anulado no se contabiliza en el estado de deuda ni en el historial de pagos.

---

*Nota.* La tabla detalla la historia de usuario HU-OPER-004.

**Tabla 47.**

*HU-OPER-005 gestión de deudas y mora*

---

Historia de usuario	
Código: HU-OPER-005	Nombre: Gestión de deudas y mora
Módulo: Operación	
Responsable: Equipo de desarrollo	Iteración: sprint 3
Prioridad: Alta	Esfuerzo: 13
Descripción	
Como administrador, quiero que el sistema calcule de manera automática la mora diaria en caso de retraso en el pago de pensiones, en base a la configuración financiera del ciclo académico, para reflejar correctamente la deuda del estudiante y mantener información precisa para la gestión de cobros.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• El sistema identifica automáticamente las pensiones vencidas.</li> <li>• El cálculo de mora se realiza según el monto definido en la configuración financiera del ciclo académico.</li> <li>• La mora se genera día a día hasta que la deuda sea cancelada.</li> <li>• El reporte de deudas muestra: pensiones pendientes, fecha de vencimiento, días de atraso, monto de mora y total endeudado.</li> <li>• Un pago realizado liquida automáticamente la mora acumulada.</li> <li>• No se generan registros de mora si el ciclo académico no tiene configuración financiera activa.</li> <li>• Todas las actualizaciones de mora quedan registradas en el historial financiero del estudiante.</li> </ul>	

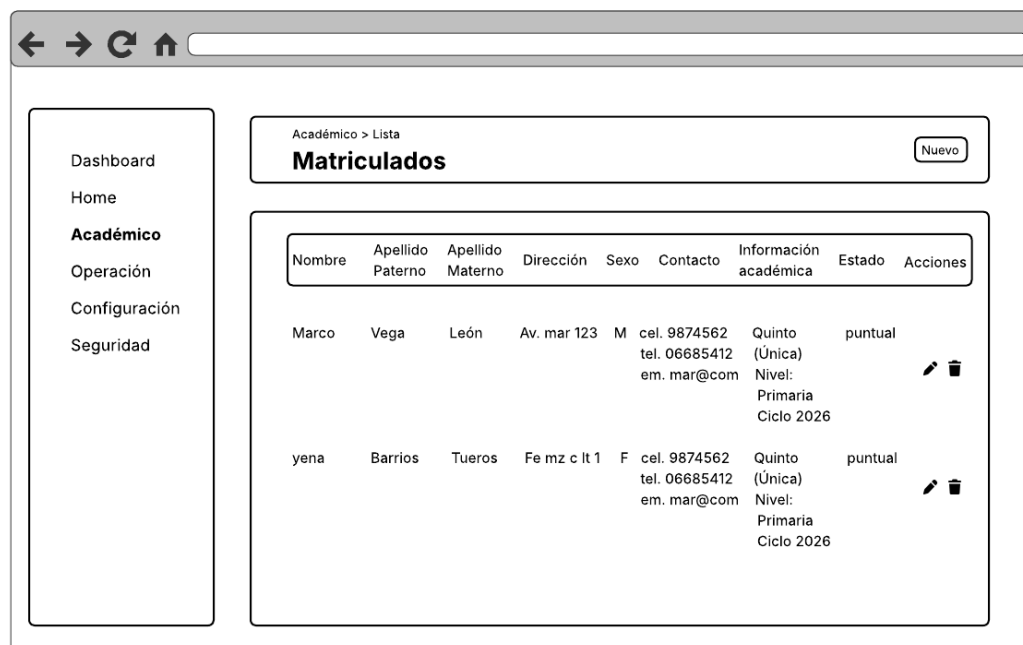
---

*Nota.* La tabla detalla la historia de usuario HU-OPER-005.

**Tabla 48.***HU-ACA-001 reporte de estudiantes*

Historia de usuario	
Código: HU-ACA-001	Nombre: Reporte de estudiantes matriculados
Módulo: Académico	
Responsable: Equipo de desarrollo	Iteración: sprint 4
Prioridad: 1	Esfuerzo: 3
Descripción	
Como director, quiero visualizar un listado de estudiantes matriculados para obtener información general del ciclo académico.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>• Se listan todos los estudiantes matriculados en el ciclo activo.</li> <li>• El reporte incluye datos principales.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-ACA-001.

**Figura 106.***Wireframe de listado de estudiantes matriculados.*

*Nota.* Bosquejo de la vista principal del menú académico.

**Tabla 49.***HU-ACA-002 filtrar estudiantes por ciclo.*

Historia de usuario	
Código: HU-ACA-002	Nombre: Filtrar estudiantes por ciclo académico
Módulo: Académico	
Responsable: Equipo de desarrollo	Iteración: sprint 4
Prioridad:1	Esfuerzo: 2
Descripción	
Como secretaria, quiero filtrar estudiantes por ciclo académico para obtener información de un año específico.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>El filtro muestra solo los estudiantes del ciclo seleccionado.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-ACA-002.**Tabla 50.***HU-ACA-003 reporte de estudiantes con pensiones vencidas.*

Historia de usuario	
Código: HU-ACA-003	Nombre: Reporte de estudiantes con pensiones vencidas
Módulo: Académico	
Responsable: Equipo de desarrollo	Iteración: sprint 4
Prioridad:1	Esfuerzo: 3
Descripción	
Como administrador, quiero visualizar estudiantes con pensiones vencidas para realizar el seguimiento de cobros.	
Criterios de aceptación	
<ul style="list-style-type: none"> <li>Se listan estudiantes con pagos atrasados y fechas de vencimiento.</li> </ul>	

*Nota.* La tabla detalla la historia de usuario HU-ACA-003.

#### 4.1.34. Taskboard

Para esta tercera etapa se definió una jornada de trabajo de lunes a viernes con una dedicación de 4 horas horas diarias. Con base de esta capacidad, las tareas del taskboard se calendarizaron en función de su estimación en horas, distribuyéndose de manera continua en los días hábiles.

**Tabla 51.**

*Estimación de trabajo en horas del sprint 3*

Equipo Scrum	Horas de trabajo por día	Horas de Trabajo por semana	Semanas de trabajo por mes	Total de horas	Total de días laborales para el proyecto
Tesista	4	20	4	80	20
Total de días disponible para el proyecto					20

*Nota.* La tabla muestra la estimación de horas disponibles y días laborables para el desarrollo del sprint 3.

**Tabla 52.***Taskboard del sprint 3*

Taskboard								
Inicio: jueves 07 de agosto de 2025					Duración: 88 horas			
ID HU	ID TA	Tarea	Responsable	Estado	Horas	Inicio	Fin	
HU-OPER-001	T-074	Diseñar el formulario de datos personales del estudiante.	Equipo de desarrollo	Hecho	1	07/08/2025	07/08/2025	
	T-075	Diseñar el formulario de documento de identidad.	Equipo de desarrollo	Hecho	1	07/08/2025	07/08/2025	
	T-076	Diseñar el formulario de contacto de emergencia.	Equipo de desarrollo	Hecho	1	07/08/2025	07/08/2025	
	T-077	Diseñar el formulario de contacto del estudiante.	Equipo de desarrollo	Hecho	1	07/08/2025	07/08/2025	
	T-078	Diseñar el formulario de datos académicos.	Equipo de desarrollo	Hecho	1	08/08/2025	08/08/2025	
	T-079	Diseñar el formulario de facturación.	Equipo de desarrollo	Hecho	1	08/08/2025	08/08/2025	
	T-080	Implementar la navegación entre los pasos del proceso de matrícula.	Equipo de desarrollo	Hecho	4	08/08/2025	11/08/2025	

	T-081	Implementar la visualización del resumen de matrícula.	Equipo desarrollo	de Hecho	4	11/08/2025	12/08/2025
	T-082	Validar los datos ingresados en cada paso del proceso.	Equipo desarrollo	de Hecho	3	12/08/2025	13/08/2025
	T-083	Probar el flujo completo de matrícula desde inicio hasta la confirmación final	Equipo desarrollo	de Hecho	2	13/08/2025	13/08/2025
HU-OPER-002	T-084	Diseñar la interfaz de edición de matrícula.	Equipo desarrollo	de Hecho	3	13/08/2025	14/08/2025
	T-085	Implementar la actualización de datos en los diferentes pasos (datos personales, documento, contacto, datos académicos, facturación).	Equipo desarrollo	de Hecho	4	14/08/2025	15/08/2025
	T-086	Validar que no se pueda editar matrícula en ciclos cerrados.	Equipo desarrollo	de Hecho	1	15/08/2025	15/08/2025
	T-087	Validar que los datos obligatorios no se borren en la actualización.	Equipo desarrollo	de Hecho	1	15/08/2025	15/08/2025
	T-088	Registrar los cambios.	Equipo desarrollo	de Hecho	2	18/08/2025	18/08/2025
	T-089	Realizar pruebas de aceptación.	Equipo desarrollo	de Hecho	1	18/08/2025	18/08/2025

HU-OPER-003	T-090	Diseñar el formulario de búsqueda de estudiante por DNI y ciclo académico	Equipo de desarrollo	de	Hecho	3	18/08/2025	19/08/2025
	T-091	Diseñar la interfaz de listado de pensiones pendientes (con opción de selección múltiple y pago adelantado).	Equipo de desarrollo	de	Hecho	4	19/08/2025	20/08/2025
	T-092	Implementar la visualización del resumen de pensiones seleccionadas	Equipo de desarrollo	de	Hecho	3	20/08/2025	21/08/2025
	T-093	Diseñar el formulario de facturación (nombre/razón social, RUC/DNI, método de pago)	Equipo de desarrollo	de	Hecho	1	21/08/2025	21/08/2025
	T-094	Implementar el registro del pago y confirmación final	Equipo de desarrollo	de	Hecho	4	21/08/2025	22/08/2025
	T-095	Actualizar el estado de deuda del estudiante tras el pago	Equipo de desarrollo		Hecho	1	22/08/2025	22/08/2025
	T-096	Probar el flujo completo de pago (pendientes, adelantados, con distintos métodos de pago)	Equipo de desarrollo		Hecho	2	22/08/2025	25/08/2025
HU-OPER-004	T-097	Diseñar el modal de anulación.	Equipo de desarrollo		Hecho	1	25/08/2025	25/08/2025

	T-098	Validar que solo usuarios con rol autorizado puedan anular operaciones.	Equipo de desarrollo	Hecho	1	25/08/2025	25/08/2025
	T-099	Implementar confirmación previa a la anulación.	Equipo de desarrollo	Hecho	1	25/08/2025	25/08/2025
	T-100	Configurar que matrículas anuladas no se consideren en reportes académicos.	Equipo de desarrollo	Hecho	1	26/08/2025	26/08/2025
	T-101	Configurar que pagos de pensiones anulados no se contabilicen en el estado financiero.	Equipo de desarrollo	Hecho	1	26/08/2025	26/08/2025
	T-102	Registrar operación anulada y conservar.	Equipo de desarrollo	Hecho	1	26/08/2025	26/08/2025
	T-103	Realizar pruebas de aceptación con escenarios de matrícula y de pensiones	Equipo de desarrollo	Hecho	1	26/08/2025	26/08/2025
HU-OPER-005	T-104	Implementar la lógica de identificación de pensiones vencidas.	Equipo de desarrollo	Hecho	4	27/08/2025	27/08/2025
	T-105	Configurar el cálculo automático de mora según parámetros del ciclo académico.	Equipo de desarrollo	Hecho	4	28/08/2025	28/08/2025

	T-106	Implementar la generación diaria de registros de mora hasta el pago.	Equipo de desarrollo	Hecho	3	29/08/2025	29/08/2025
	T-107	Diseñar el reporte de deudas con pensiones vencidas, días de atraso y mora acumulada.	Equipo de desarrollo	Hecho	2	29/08/2025	01/09/2025
	T-108	Validar que los pagos liquiden automáticamente la mora acumulada.	Equipo de desarrollo	Hecho	2	01/09/2025	01/09/2025
	T-109	Probar casos de estudiantes con pagos puntuales, atrasados y múltiples deudas	Equipo de desarrollo	Hecho	2	01/09/2025	02/09/2025
HU-ACA-001	T-110	Diseñar la interfaz del reporte de estudiantes matriculados.	Equipo de desarrollo	Hecho	2	02/09/2025	02/09/2025
	T-111	Implementar listado de estudiantes del ciclo activo.	Equipo de desarrollo	Hecho	2	02/09/2025	03/09/2025
	T-112	Implementar visualización de datos principales	Equipo de desarrollo	Hecho	2	03/09/2025	03/09/2025
HU-ACA-002	T-113	Implementar consulta de estudiantes filtrados por ciclo seleccionado.	Equipo de desarrollo	Hecho	4	03/09/2025	04/09/2025
	T-114	Validar que no se dupliquen registros entre ciclos distintos.	Equipo de desarrollo	Hecho	2	04/09/2025	05/09/2025

HU-ACA-003	T-115	Implementar consulta para filtrar estudiantes con pagos atrasados.	Equipo de desarrollo	Hecho	3	05/09/2025	05/09/2025
------------	-------	--	----------------------	-------	---	------------	------------

---

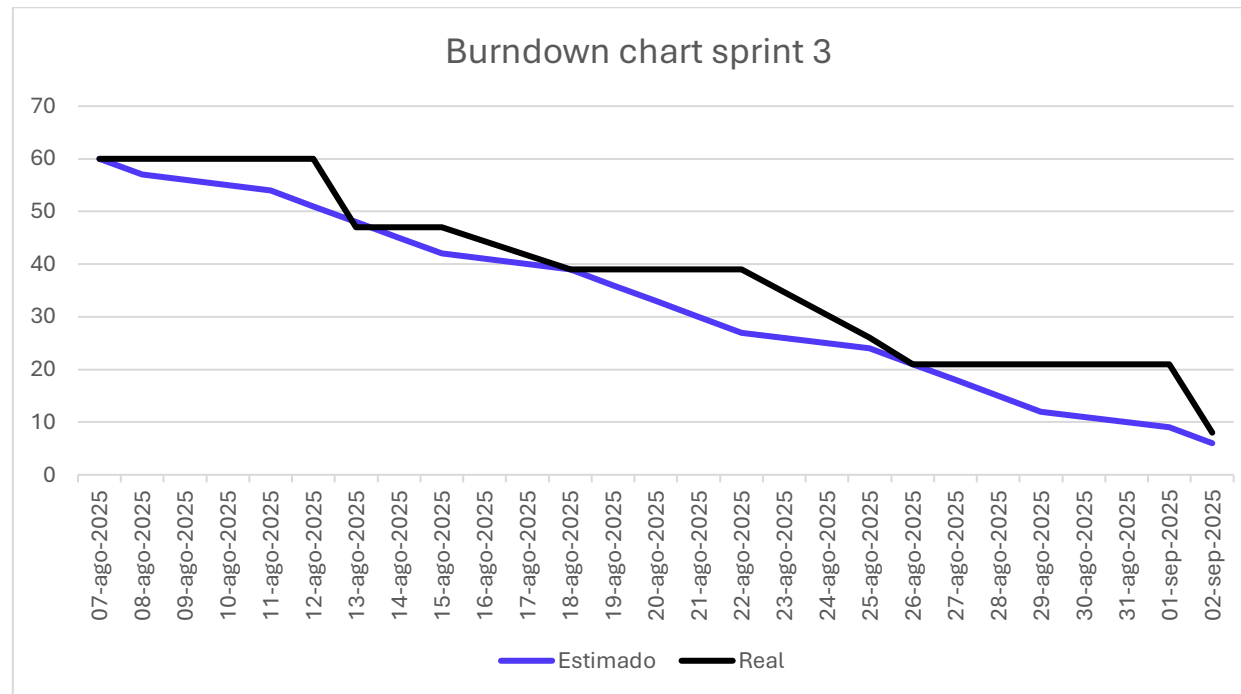
*Nota.* La tabla muestra el taskboard del sprint 3, donde se detalla las tareas ejecutadas en cada historia de usuario.

Para tercer sprint se estimó un tiempo de 80 horas, Sin embargo, durante la ejecución del sprint se registró un tiempo total de 88 horas, evidenciando una ligera desviación respecto a la planificación inicial. Este incremento se debió principalmente a la complejidad técnica de las historias: HU-OPER-001, HU-OPER-003 y HU-OPER-005. A pesar de ello, la diferencia fue mínima y se mantuvo dentro de los márgenes aceptables de la metodología scrum, sin afectar los entregables ni la fecha de cierre del sprint.

#### 4.1.35. Burndown chart

**Figura 107.**

*Burndown chart del sprint 3*



*Nota.* El gráfico muestra la comparación entre el progreso esperado y el real durante el desarrollo del sprint 3.

#### 4.1.36. Daily scrum

Como en los sprints anteriores, las reuniones de daily scrum se adaptaron a la naturaleza del equipo. Se documentaron los avances al cierre de cada historia de usuario, lo que permitió mantener una trazabilidad clara del progreso y asegurar la verificación del cumplimiento de los criterios de aceptación establecidos.

**Tabla 53.**

*Daily scrum - sprint 3*

Fecha	¿Qué hice?	¿Qué haré después?	Impedimentos	Observaciones
13/08/2025	Cierre HU-OPER-001 (Registro de matrícula). Se diseñaron e implementaron los formularios por pasos (T-074..T-079), la navegación entre pasos (T-080), la visualización del resumen de matrícula (T-081) y las validaciones/pruebas finales (T-082, T-083).	Iniciar HU-OPER-002 (Edición de matrícula): diseñar interfaz de edición y comenzar la lógica para actualizar los distintos pasos.	Ninguno	HU-OPER-001 completada y probada; el wizard funciona correctamente y muestra el resumen con descuentos.
18/08/2025	Cierre HU-OPER-002 (Editar matrícula). Se implementó la interfaz de edición (T-084),	Iniciar HU-OPER-003 (Pago de pensiones): diseñar búsqueda por DNI/ciclo y	Validación de reglas: asegurar que no se pueda editar matrícula en ciclos cerrados	Edición segura y confirmada; cambios quedan registrados. Preparado para integrar pagos.

- actualización de lista de (resuelta en T-  
datos en los pasos pensiones 086).  
(T-085..T-088) y pendientes.  
pruebas de  
aceptación (T-089).
- 25/08/2025 Cierre HU-OPER- Iniciar HU- Ajustes en la Flujo de pago  
003 (Pago de OPER-004 lógica para (pendientes y  
pensiones). (Anulación de selección adelantados)  
Se completó la operación): múltiple y validado con  
búsqueda y listado diseñar modal de cálculo de distintos  
de pensiones (T- anulación y subtotales métodos; estado  
090, T-091), la validar permisos. (solucionado). de deuda  
selección múltiple y actualizado  
resumen (T-092), correctamente.  
formulario de  
facturación y  
registro de pago (T-  
093, T-094), y  
actualización del  
estado de deuda (T-  
095, T-096).
- 26/08/2025 Cierre HU-OPER- Iniciar HU- Necesidad de Anulación segura  
004 (Anular OPER-005 conservar implementada:  
operación). (Gestión de historial al anular solo roles  
Se diseñó el modal deudas y mora): (se definió y autorizados y con  
de anulación (T- implementar almacenó motivo registro de  
097), se validó cálculo y metadatos). motivo; reportes  
control de roles (T- automático de actualizados.  
098), se implementó mora y reporte  
confirmación y de deudas.  
reglas para que

	matrículas/pagos anulados no cuenten en reportes (T-099..T-103) y se realizó pruebas.				
02/09/2025	Cierre HU-OPER-005 (Gestión de deudas y mora). Se implementó la identificación de pensiones vencidas (T-104), la lógica de cálculo de mora y configuración (T-105), la generación diaria de registros (T-106), diseño del reporte y validaciones (T-107..T-109).	Preparar Sprint Review y documentación de entregables; pruebas de integración finales con módulos Configuración y Operaciones.	Ajustes finales para evitar duplicados en generación de mora y verificar que pagos el liquiden correctamente la mora (resuelto en T-108/T-109).	HU-OPER-005 completada: la mora se genera día a día según configuración y el reporte muestra deuda, mora y días de atraso.	
05/09/2025	Cierre de las historias de usuario HU-ACA-001, HU-ACA-002 y HU-ACA-003.	Realizar pruebas de aceptación del módulo académico.	Ninguno	Ninguno.	

---

Nota. La tabla muestra los resultados de los daily scrum correspondientes al sprint 3.

#### 4.1.37. Sprint review

El Sprint Review se realizó al finalizar el Sprint 3 con el propósito de validar las funcionalidades desarrolladas en el módulo de operaciones y académico. Durante la sesión, se revisaron las historias de usuario completadas y se confirmó que todas cumplieron con sus

criterios de aceptación, asegurando la integración funcional entre los procesos de matrícula, pagos y gestión de deudas.

**Tabla 54.**

*Sprint review del sprint 3*

Sprint review	
Fecha:	03/09/2025
Participantes	<ul style="list-style-type: none"> <li>• Product Owner</li> <li>• Scrum Master</li> <li>• Development Team</li> </ul>
Duración	2 horas
Objetivo	Revisar las historias de usuario completadas durante el Sprint 3, validar su cumplimiento según los criterios de aceptación y demostrar la funcionalidad implementada en el módulo de Operaciones.
Historias de usuario incluidas	<ul style="list-style-type: none"> <li>• HU-OPER-001</li> <li>• HU-OPER-002</li> <li>• HU-OPER-003</li> <li>• HU-OPER-004</li> <li>• HU-OPER-005</li> <li>• HU-ACA-001</li> <li>• HU-ACA-002</li> <li>• HU-ACA-003</li> </ul>
Entregables	<ul style="list-style-type: none"> <li>• Módulo de operación y académico completado.</li> <li>• Formularios funcionales con validaciones de negocio.</li> </ul>

*Nota.* La tabla muestra el resumen del sprint review correspondiente al sprint 3.

#### 4.1.38. Despliegue del sprint

**Tabla 55.**

*Despliegue del sprint 3.*

Despliegue del sprint	
Objetivos	Realizar el despliegue del módulo operación y académico, garantizando la correcta integración con los módulos previos (seguridad y configuración) y la disponibilidad completa del flujo operativo del sistema.
Actividades realizadas	<ul style="list-style-type: none"> <li>• Se integraron las funcionalidades correspondientes a los procesos principales del sistema registro y edición de matrícula, pago de pensiones, gestión de deudas y moras y anulación de operaciones, junto con los reportes académicos de estudiantes matriculados y con pensiones vencidas.</li> <li>• Se ejecutó las migraciones y los datos iniciales en el entorno de pruebas.</li> <li>• Se validó la correcta comunicación entre los módulos de configuración y operación, garantizando que los datos se propaguen correctamente.</li> <li>• Se validó el funcionamiento general del sistema en el entorno de pruebas, asegurando el flujo completo.</li> </ul>
Resultado	El despliegue del sprint 3 se completó satisfactoriamente, consolidando el núcleo operativo del sistema. Con este despliegue, el sistema alcanzó un nivel funcional completo, integrando todos los módulos principales y permitiendo su validación en entorno de pruebas.

*Nota.* La tabla detalla el proceso de despliegue del sprint 3.

#### 4.1.39. Criterios de aceptación

**Tabla 56.**

*Criterios de aceptación del sprint 3*

Criterios de aceptación		
ID De historia de usuario	Criterio de aceptación	Resultado
HU-OPER-001	<ul style="list-style-type: none"> <li>• El proceso de matrícula se realiza en pasos secuenciales.</li> <li>• En el paso 1 se ingresan datos personales.</li> <li>• En el paso 2 se ingresan datos de documento de identidad.</li> <li>• En el paso 3 se ingresa contacto de emergencia.</li> <li>• En el paso 4 se ingresa contacto del estudiante.</li> <li>• En el paso 5 se registran datos académicos.</li> <li>• En el paso 6 se registran datos de facturación.</li> <li>• Se muestra un resumen de matrícula con el detalle de pago y descuentos aplicados.</li> <li>• La matrícula queda registrada correctamente al finalizar el flujo.</li> </ul>	<ul style="list-style-type: none"> <li>• El proceso de matrícula fue implementado exitosamente mediante un formulario por pasos.</li> <li>• Cada etapa (datos personales, documento, contacto, información académica y facturación) valida campos obligatorios antes de avanzar.</li> <li>• El sistema genera un resumen con los detalles de pago, descuentos aplicados y confirmación final.</li> <li>• La matrícula queda registrada correctamente y se asocia al ciclo académico y estudiante correspondiente.</li> </ul>

HU-OPER-002

- Se valida que los datos obligatorios no queden incompletos tras la edición.
- Se muestra confirmación al usuario de que la matrícula fue actualizada correctamente.

- Se validó que los datos obligatorios no puedan quedar incompletos durante la edición.
- El sistema muestra un mensaje de confirmación tras actualizar la información.
- Las modificaciones se reflejan inmediatamente en el registro del estudiante.

HU-OPER-003

- Se busca al estudiante por DNI y ciclo académico.
- Se muestran todas las pensiones pendientes y la opción de pagar por adelantado.
- Se pueden seleccionar una o más pensiones para el pago.
- Se muestra un resumen del pago con subtotales y total a pagar.
- Se ingresan los datos de facturación (nombre/razón social, RUC/DNI).
- Se selecciona el método de pago.
- El pago se confirma y se registra en el sistema.
- El estado de la deuda del estudiante se actualiza automáticamente.

- Se implementó la búsqueda por DNI y ciclo académico.
- El sistema muestra las pensiones pendientes y permite seleccionar una o varias para el pago, incluyendo pagos adelantados.
- Se generan subtotales, total a pagar y comprobante con datos de facturación.
- La deuda del estudiante se actualiza automáticamente al registrar el pago.

## HU-OPER-004

- Solo usuarios con rol autorizado pueden anular operaciones.
- La anulación solicita confirmación antes de proceder.
- Una matrícula anulada no se considera en reportes académicos ni financieros.
- Un pago de pensión anulado no se contabiliza en el estado de deuda ni en el historial de pagos.

## HU-OPER-005

- El sistema identifica automáticamente las pensiones vencidas.
- El cálculo de mora se realiza según el monto definido en la configuración financiera del ciclo académico.
- La mora se genera día a día hasta que la deuda sea cancelada.
- El reporte de deudas muestra: pensiones pendientes, fecha de vencimiento, días de atraso, monto de mora y total endeudado.
- Un pago realizado liquida automáticamente la mora acumulada.
- No se generan registros de mora si el ciclo académico no tiene configuración financiera activa.

- Solo los usuarios con rol autorizado pueden realizar anulaciones.
- El sistema solicita confirmación antes de proceder con la anulación.
- Las operaciones anuladas quedan registradas en el historial, pero se excluyen de los reportes académicos y financieros.
- Se implementó la detección automática de pensiones vencidas.
- La mora se calcula diariamente en función de la configuración financiera activa del ciclo académico.
- Los reportes de deuda muestran detalle de pensiones pendientes, días de atraso, mora acumulada y total adeudado.
- Los pagos realizados eliminan automáticamente la mora generada.

	<ul style="list-style-type: none"> <li>• Todas las actualizaciones de mora quedan registradas en el historial financiero del estudiante.</li> </ul>	<ul style="list-style-type: none"> <li>• El historial financiero refleja todos los cálculos y actualizaciones de manera automática.</li> </ul>
HU-ACA-001	<ul style="list-style-type: none"> <li>• Se listan todos los estudiantes matriculados en el ciclo activo.</li> <li>• El reporte incluye datos principales.</li> </ul>	<ul style="list-style-type: none"> <li>• El sistema muestra correctamente los estudiantes matriculados del ciclo académico activo.</li> </ul>
HU-ACA-002	<ul style="list-style-type: none"> <li>• El filtro muestra solo los estudiantes del ciclo seleccionado.</li> </ul>	<ul style="list-style-type: none"> <li>• El filtro funciona correctamente.</li> </ul>
HU-ACA-003	<ul style="list-style-type: none"> <li>• Se listan estudiantes con pagos atrasados y fechas de vencimiento.</li> </ul>	<ul style="list-style-type: none"> <li>• El sistema muestra correctamente estudiantes con pagos atrasados.</li> </ul>

---

*Nota.* La tabla detalla los criterios de aceptación de las historias de usuario correspondiente al sprint 3.

#### 4.1.40. Pruebas unitarias

Para validar el correcto funcionamiento de la aplicación a nivel de código en este último sprint, Se tomó 3 unidades de código, estas unidades tienen por funcionalidad validar la creación y actualización de una matrícula, como también la anulación de una operación. Estas pruebas dieron como resultado exitoso esto se evidencia en la **Figura 108**.

**Tabla 57.**

*Prueba unitaria - sprint 3*

No	Historia de usuario	Caso de prueba	Resultado
1	HU-OPER-001	Crear matrícula	Exitoso
2	HU-OPER-002	Editar matrícula	Exitoso
4	HU-OPER-004	Anular operación	Exitoso

*Nota.* La tabla muestra las pruebas unitarias realizadas en cada historia de usuario correspondiente al sprint 3.

**Figura 108.**

*Resultado de las pruebas unitarias de gestión de matrícula.*

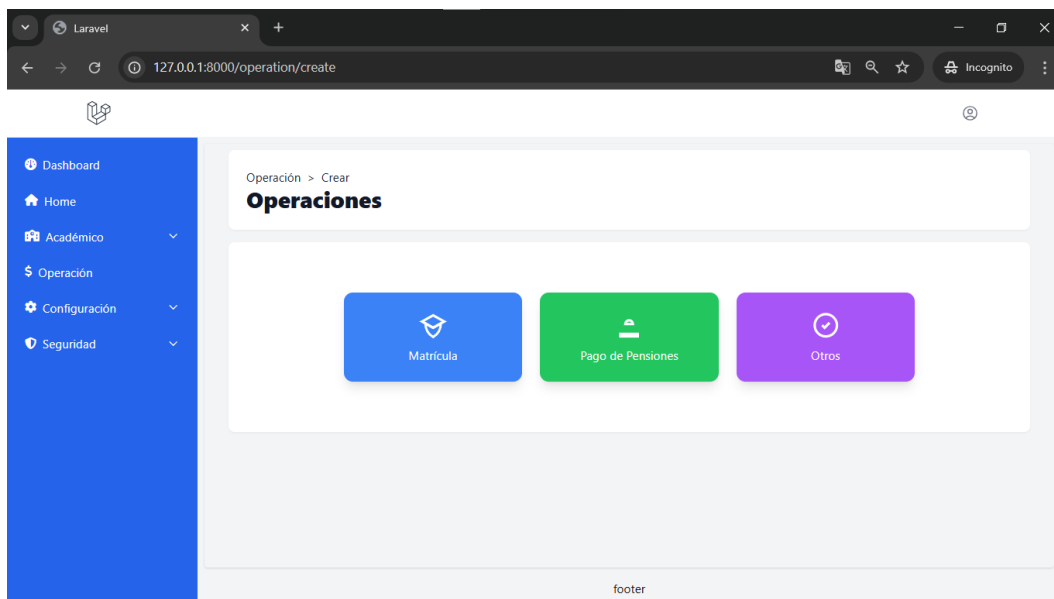
```
PS C:\Users\viklo\OneDrive\Escritorio\projects\sima2025> php artisan test --filter=EnrollmentTest
PASS Tests\Feature\Operation\EnrollmentTest
✓ enrollment can be created 8.88s
✓ enrollment can be updated 8.15s
✓ operation can be deleted 7.71s
Tests: 3 passed (34 assertions)
Duration: 25.03s
```

En la **Figura 108** se observa el resultado de las pruebas unitarias de gestión de matrículas, estas pruebas constan de 3 unidades de código, Las cuales validan la creación y actualización de una matrícula, y el eliminado de una operación. La prueba da como resultado de aprobado en las 3 unidades de código.

#### 4.1.41. Resultados del sprint

**Figura 109.**

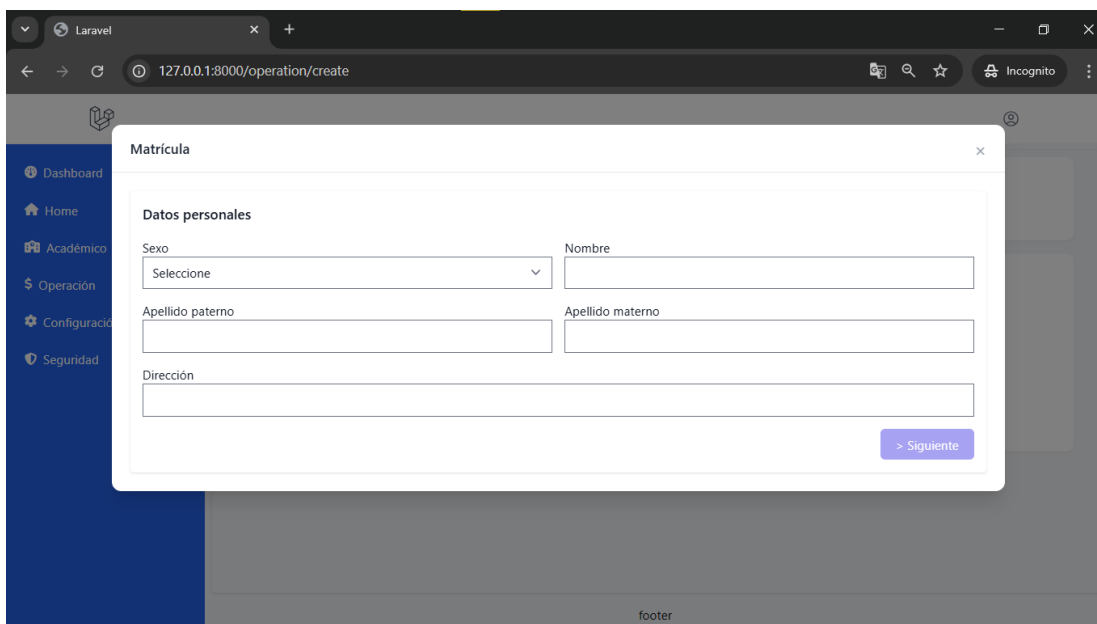
*Interfaz del módulo de operaciones.*



*Nota.* La imagen muestra la pantalla principal de operaciones.

**Figura 110.**

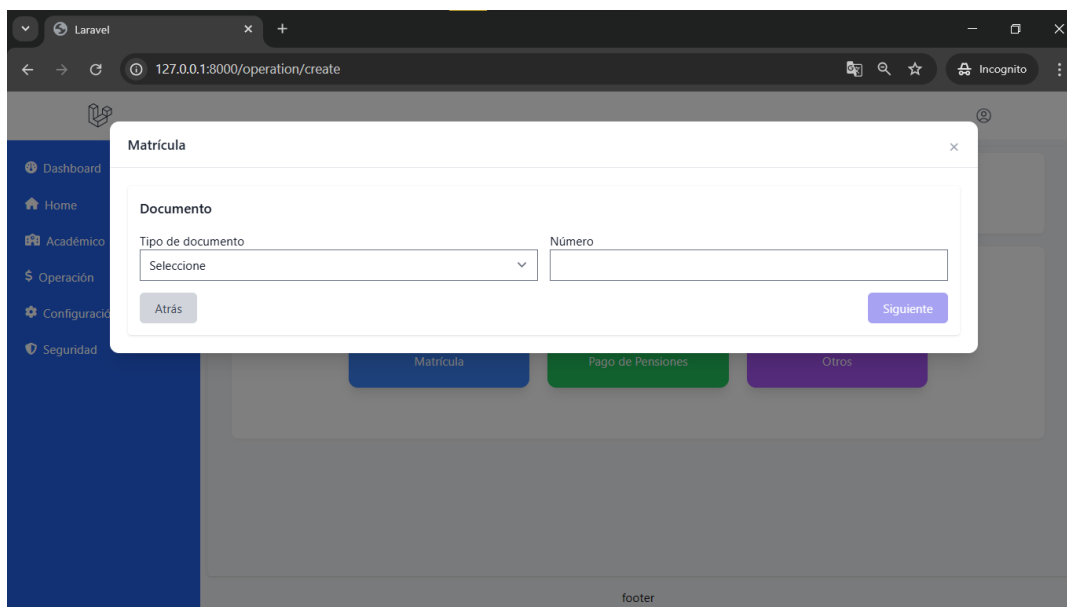
*Interfaz de creación de matrícula - paso 1: datos personales.*



*Nota.* La imagen muestra la pantalla de creación de matrícula paso 1.

**Figura 111.**

*Interfaz de creación de matrícula - paso 2: documento.*

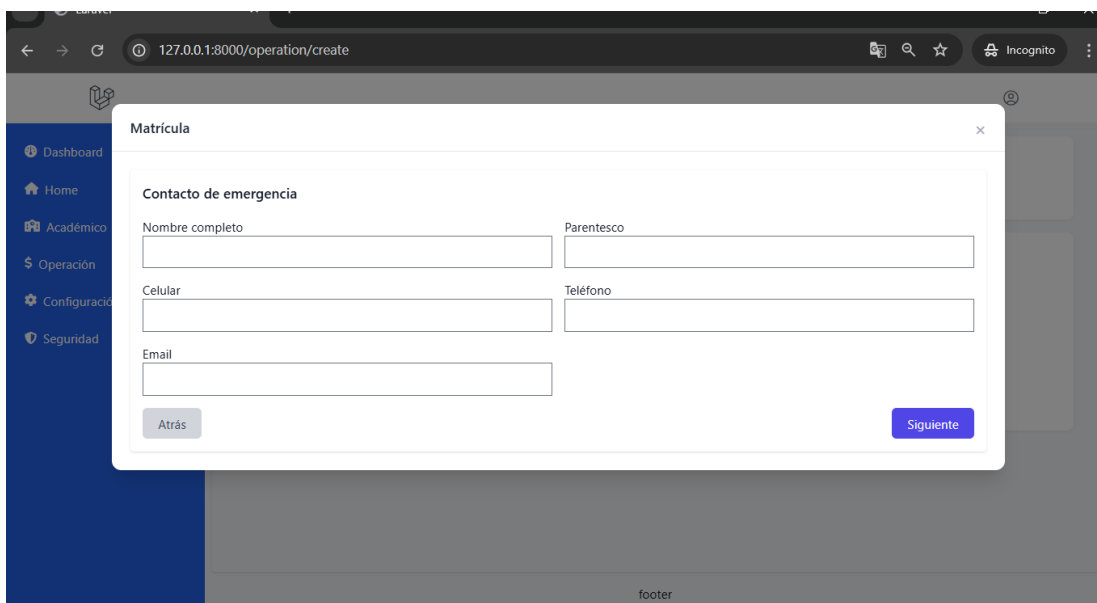


The screenshot shows a web browser window with the URL '127.0.0.1:8000/operation/create'. A modal window titled 'Matrícula' is open, displaying the 'Documento' section. The form includes a dropdown menu for 'Tipo de documento' with the text 'Seleccione' and a text input field for 'Número'. At the bottom of the modal are two buttons: 'Atrás' (grey) and 'Siguiete' (blue). The background shows a sidebar with navigation options: Dashboard, Home, Académico, Operación, Configuración, and Seguridad. Below the modal, three buttons are visible: 'Matrícula' (blue), 'Pago de Pensiones' (green), and 'Otros' (purple). A 'footer' label is at the bottom of the page.

*Nota.* La imagen muestra la pantalla de creación de matrícula paso 2.

**Figura 112.**

*Interfaz de creación de matrícula - paso 3: contacto de emergencia.*

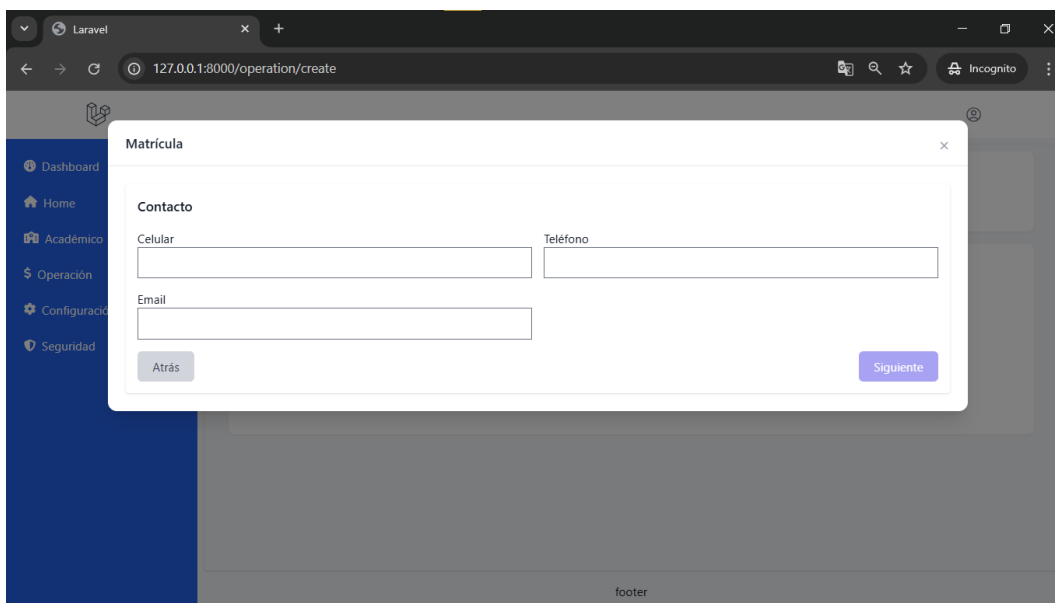


The screenshot shows the same web browser window and modal window. The modal is now titled 'Matrícula' and displays the 'Contacto de emergencia' section. The form includes five input fields: 'Nombre completo', 'Parentesco', 'Celular', 'Teléfono', and 'Email'. At the bottom of the modal are two buttons: 'Atrás' (grey) and 'Siguiete' (blue). The background elements are the same as in the previous screenshot.

*Nota.* La imagen muestra la pantalla de creación de matrícula paso 3.

**Figura 113.**

*Interfaz de creación de matrícula – paso 4: contacto.*

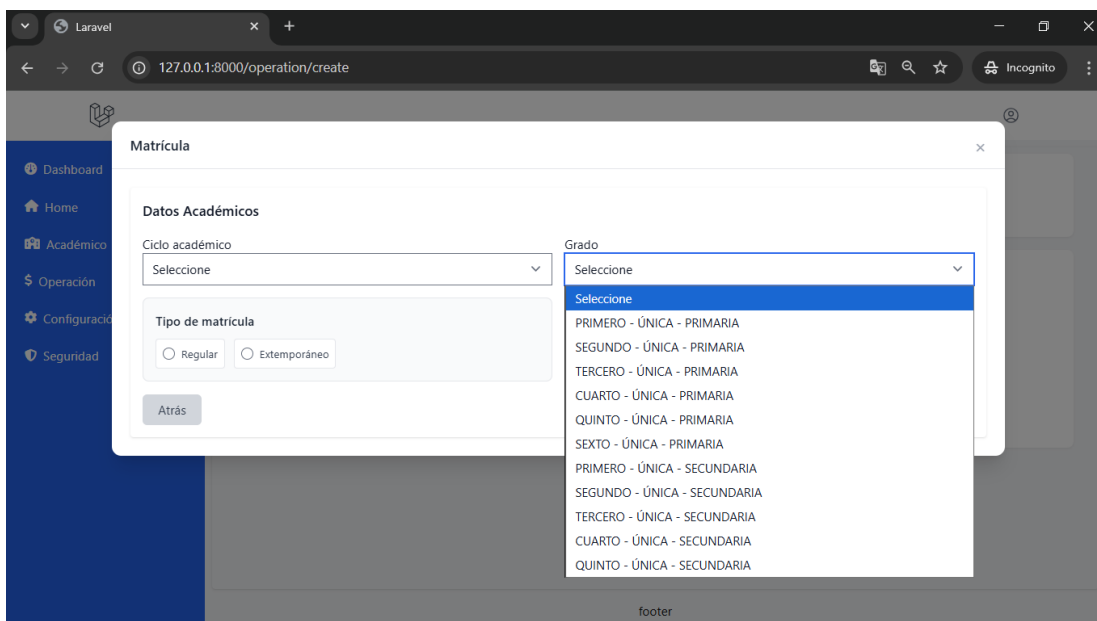


The screenshot shows a web browser window with the URL `127.0.0.1:8000/operation/create`. A modal window titled 'Matrícula' is open, displaying the 'Contacto' section. It contains three input fields: 'Celular', 'Teléfono', and 'Email'. Below the fields are two buttons: 'Atrás' (Back) and 'Siguiente' (Next). The background shows a sidebar with navigation options: Dashboard, Home, Académico, Operación, Configuración, and Seguridad.

*Nota.* La imagen muestra la pantalla de creación de matrícula paso 4.

**Figura 114.**

*Interfaz de creación de matrícula - paso 5: datos académicos.*



The screenshot shows the same web browser window, but the modal window is now on the 'Datos Académicos' step. It features two dropdown menus: 'Ciclo académico' and 'Grado'. The 'Grado' dropdown is open, showing a list of options: 'Selección', 'PRIMERO - ÚNICA - PRIMARIA', 'SEGUNDO - ÚNICA - PRIMARIA', 'TERCERO - ÚNICA - PRIMARIA', 'CUARTO - ÚNICA - PRIMARIA', 'QUINTO - ÚNICA - PRIMARIA', 'SEXTO - ÚNICA - PRIMARIA', 'PRIMERO - ÚNICA - SECUNDARIA', 'SEGUNDO - ÚNICA - SECUNDARIA', 'TERCERO - ÚNICA - SECUNDARIA', 'CUARTO - ÚNICA - SECUNDARIA', and 'QUINTO - ÚNICA - SECUNDARIA'. Below the dropdowns is a 'Tipo de matrícula' section with two radio buttons: 'Regular' and 'Extemporáneo'. The 'Atrás' button is also visible. The background sidebar remains the same.

*Nota.* La imagen muestra la pantalla de creación de matrícula paso 5.

**Figura 115.**

*Interfaz de creación de matrícula - paso 6: datos de facturación.*

The screenshot shows a web browser window with the URL '127.0.0.1:8000/operation/create'. The page displays a modal form titled 'Matrícula' with the following fields and options:

- Datos de facturación**
- Nombre / Razón social:  (Error message: El campo name es obligatorio.)
- RUC / DNI:
- Descuento:
- Aplicar Descuento:  matrícula  pensiones  matrícula y pensiones
- Método de pago:  EFECTIVO  YAPE  PLIN
- Buttons: Atrás, Siguiente

*Nota.* La imagen muestra la pantalla de creación de matrícula paso 6.

**Figura 116.**

*Interfaz de creación de matrícula - paso 7: resumen.*

The screenshot shows the same web browser window, but the modal form is now titled 'Matrícula' and displays a 'Resumen' (Summary) table:

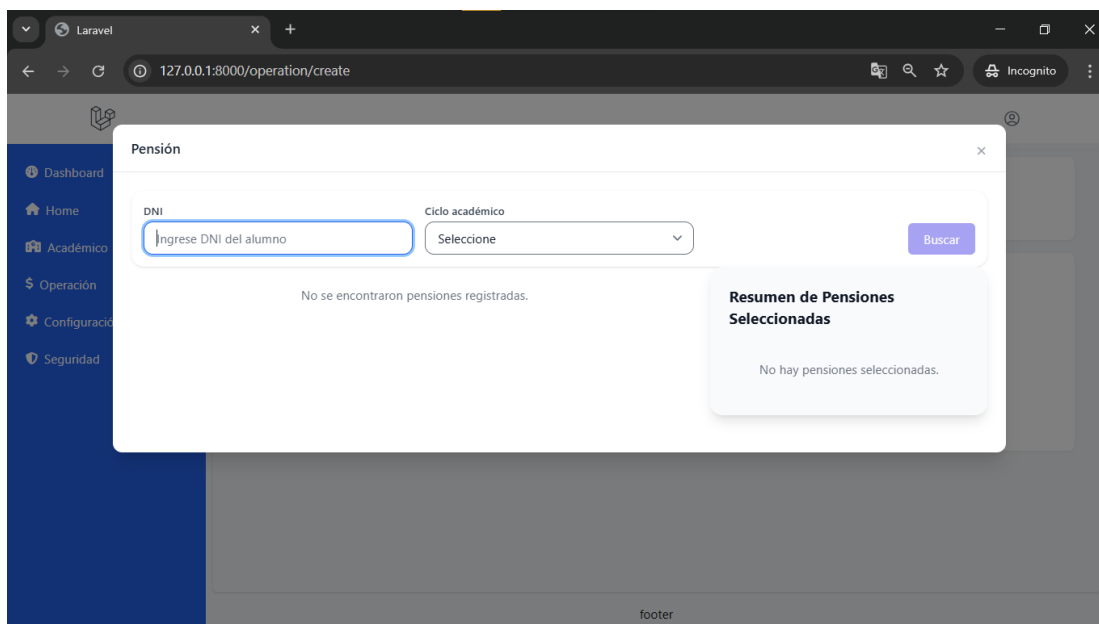
Nombre / Razón social:	carla vayadolid
Concepto:	MATRÍCULA
Ciclo académico:	CICLO VERANO 2026
Subtotal:	S/ 80.00
Descuento (HERMANOS):	-S/ 80.00
Subtotal con descuento:	S/ 0.00
IGV (18%):	S/ 0.00
<b>Total a pagar:</b>	<b>S/ 0.00</b>

Buttons: Atrás, Siguiente

*Nota.* La imagen muestra la pantalla de creación de matrícula paso 7.

**Figura 117.**

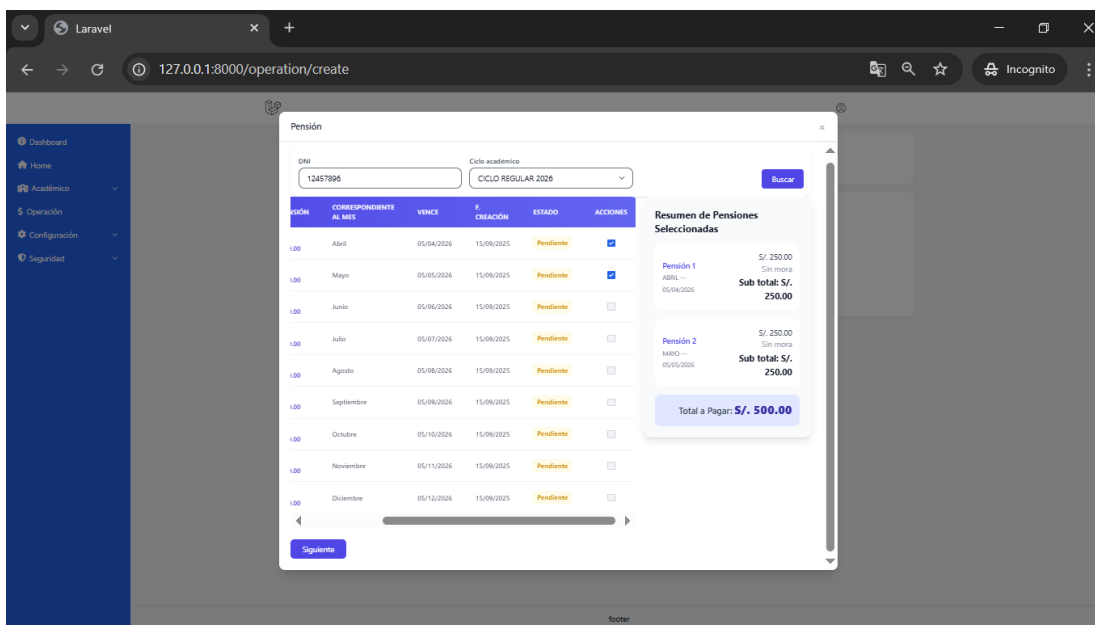
*Interfaz de pago de pensiones – paso 1.*



*Nota.* La imagen muestra la pantalla de pago de pensiones paso 1.

**Figura 118.**

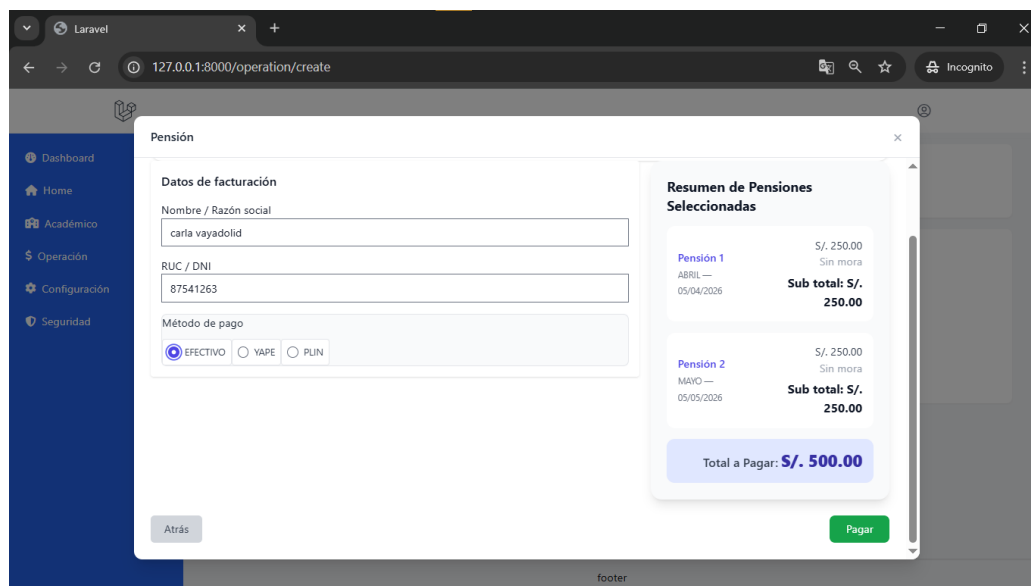
*Interfaz de pago de pensiones. - paso 2.*



*Nota.* La imagen muestra la pantalla de pago de pensiones paso 2.

**Figura 119.**

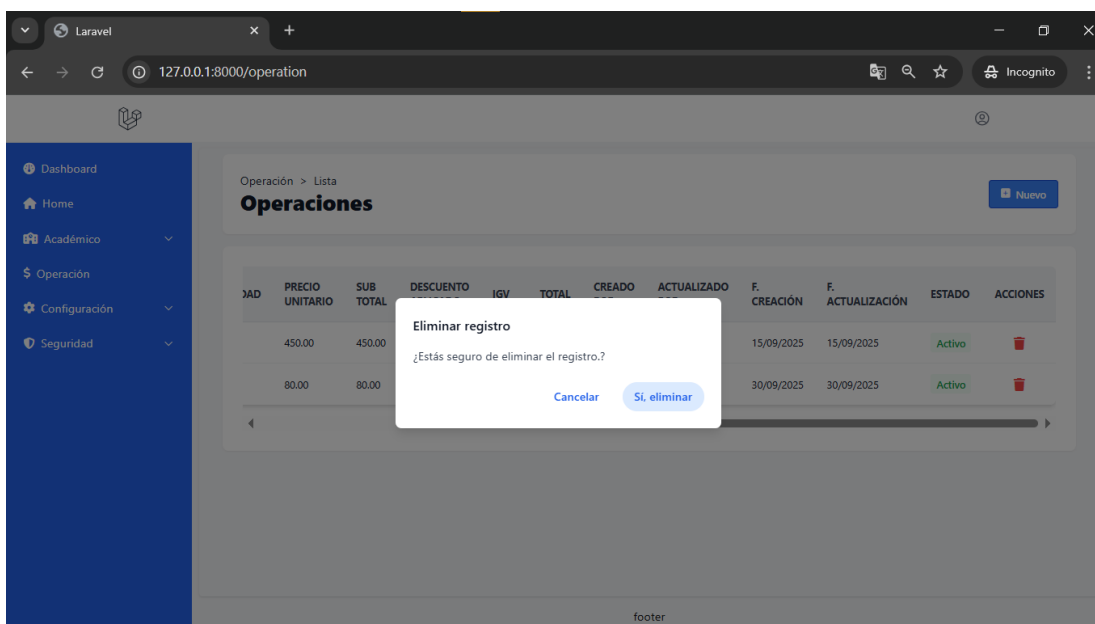
*Interfaz de pago de pensiones - paso 3.*



*Nota.* La imagen muestra la pantalla de pago de pensiones paso 3.

**Figura 120.**

*Interfaz de anular operación.*



*Nota.* La imagen muestra un modal que permite anular o eliminar una operación.

**Figura 121.**

*Reporte de estudiantes matriculados.*

#	NOMBRE	APELLIDO PATERNO	APELLIDO MATERNO	DIRECCIÓN	SEXO	DOCUMENTO	CONTACTO	CONTACTO DE EMERGENCIA	MATRÍCULAS
1	MIRIAN	ALTAMIRANO	VEGA	AVENIDA ROLSALES 1128	FEMENINO	DOCUMENTO NACIONAL DE IDENTIDAD N°: 78541203	963258741 066985632 MIRIAM12@GMAIL.COM	DELIA VEGA MADRE 985123601	PRIMERO (ÚNICA) Nivel: PRIMARIA Ciclo: CICLO REGULAR 2026 <a href="#">Pagos</a>
2	BAVERLY	MARSH	NAZARET	JIRÓN BEGONIAS MZ L LT. 7	FEMENINO	DOCUMENTO NACIONAL DE IDENTIDAD N°: 87451203	963789102		CUARTO (ÚNICA) Nivel: PRIMARIA Ciclo: CICLO REGULAR 2026 <a href="#">Pagos</a>

*Nota.* La imagen muestra la pantalla de estudiantes matriculados.

#### 4.1.42. Sprint retrospective

El Sprint 3 se planificó con una duración estimada de 80 horas y se completó en 83 horas efectivas, lo que refleja una buena gestión del tiempo y un margen mínimo de desviación. Las funcionalidades implementadas fortalecen la parte operativa del sistema, cumpliendo con los objetivos propuestos.

**Tabla 58.**

*Sprint retrospective del sprint 3*

Sprint retrospective	
Fecha:	06/09/2025
Participantes	<ul style="list-style-type: none"> <li>• Scrum master</li> <li>• Product owner</li> <li>• Development team</li> </ul>
Objetivo	Evaluar el desarrollo del Sprint 3, identificar los aciertos, dificultades y establecer mejoras para el siguiente sprint.

Aspectos positivos	<ul style="list-style-type: none"><li>• Se logró completar todas las historias de usuario planificadas dentro del tiempo establecido.</li><li>• El flujo de matrícula y pago de pensiones se implementó exitosamente y pasó las pruebas de aceptación.</li><li>• Se optimizó la generación automática de mora.</li></ul>
Aspectos para mejorar	<ul style="list-style-type: none"><li>• El desarrollo de las interfaces con múltiples pasos (matrícula y pago) demandó más validaciones de lo previsto.</li><li>• Se presentaron leves demoras en la integración del cálculo de mora diaria.</li></ul>
Acciones de mejora para el siguiente sprint	<ul style="list-style-type: none"><li>• Planificar pruebas intermedias para historias de usuarios con grados de dificultad alta, para detectar errores antes de la integración.</li><li>• Mejorar la estimación de esfuerzo para tareas con flujos secuenciales y validaciones múltiples.</li></ul>

---

*Nota.* La tabla muestra el resultado de la retrospectiva del sprint 3.

## 1.18. Resultados

### 4.1.43. Resultados de funcionalidad de la aplicación

La funcionalidad de la aplicación web fue evaluada por el usuario final. Esta evaluación tuvo como propósito verificar el cumplimiento de los requerimientos establecidos en las historias de usuario y validar que cada módulo desarrollado responda adecuadamente a las necesidades funcionales identificadas durante el análisis.

Dicha evaluación se realizó únicamente con fines académicos, en un entorno de prueba, sin llegar a la etapa de implementación. Por tanto, los resultados obtenidos se enfocan en medir la funcionalidad del sistema propuesto.

**Tabla 59.**

*Resultados de evaluación de funcionalidad de la aplicación.*

Pregunta	Cantidad		Porcentaje	
	Si	No	Si	No
¿El sistema permite la autenticación de usuarios?	3	0	100%	0%
¿El sistema permite crear roles?	3	0	100%	0%
¿El sistema permite editar un rol?	3	0	100%	0%
¿El sistema permite eliminar un rol?	3	0	100%	0%
¿El sistema permite crear un usuario?	3	0	100%	0%
¿El sistema permite editar un usuario?	3	0	100%	0%
¿El sistema permite eliminar un usuario?	3	0	100%	0%
¿El sistema permite crear un ciclo académico?	3	0	100%	0%
¿El sistema permite editar un ciclo académico?	3	0	100%	0%
¿El sistema permite eliminar un ciclo académico?	3	0	100%	0%
¿El sistema permite crear una sección?	3	0	100%	0%
¿El sistema permite editar una sección?	3	0	100%	0%
¿El sistema permite eliminar una sección?	3	0	100%	0%
¿El sistema permite crear un nivel?	3	0	100%	0%
¿El sistema permite editar un nivel?	3	0	100%	0%
¿El sistema permite eliminar un nivel?	3	0	100%	0%
¿El sistema permite crear un grado?	3	0	100%	0%

¿El sistema permite editar un grado?	3	0	100%	0%
¿El sistema permite eliminar un grado?	3	0	100%	0%
¿El sistema permite crear un tipo de matrícula?	3	0	100%	0%
¿El sistema permite editar un tipo de matrícula?	3	0	100%	0%
¿El sistema permite eliminar un tipo de matrícula?	3	0	100%	0%
¿El sistema permite crear un descuento?	3	0	100%	0%
¿El sistema permite editar un descuento?	3	0	100%	0%
¿El sistema permite eliminar un descuento?	3	0	100%	0%
¿El sistema permite crear un tipo de operación?	3	0	100%	0%
¿El sistema permite editar un tipo de operación?	3	0	100%	0%
¿El sistema permite eliminar un tipo de operación?	3	0	100%	0%
¿El sistema permite crear un método de pago?	3	0	100%	0%
¿El sistema permite editar un método de pago?	3	0	100%	0%
¿El sistema permite eliminar un método de pago?	3	0	100%	0%
¿El sistema permite registrar una matrícula?	3	0	100%	0%
¿El sistema permite editar una matrícula?	3	0	100%	0%
¿El sistema permite pagar una o más pensiones?	3	0	100%	0%
¿El sistema permite anular una operación (matrícula o pago de pensión)?	3	0	100%	0%
¿El sistema permite el cálculo automático de moras a pensiones vencidas?	3	0	100%	0%

---

*Nota.* La tabla muestra la validación de las funcionalidades de la aplicación por tres usuarios, equipo de desarrollo y dos usuarios finales.

En la **Tabla 59**, se observa el resultado de la evaluación de funcionalidad de la aplicación, mediante un cuestionario se evaluó la usabilidad de la aplicación por parte del usuario final, dando como resultado que la aplicación cumple y funciona al 100%.

**Tabla 60.**

*Tiempos registrados en la ejecución de los procesos evaluados.*

Proceso evaluado	Tiempo
Gestión académica	5 min
Gestión de información del estudiante	5 min
Control de pagos.	4 min

*Nota.* Los valores corresponden a la medición directa de los tiempos de ejecución del prototipo, registrados mediante una ficha de registro.

En la **Tabla 60** es el resultado de uso de la aplicación, donde se midió el tiempo que tomó a cada proceso. Al tratarse de un trabajo académico y solo un prototipo esta medición se realizó una sola vez por un usuario final. Dando un promedio de 5 minutos que tomó a cada proceso.

#### **4.1.44. Resultados derivados de la funcionalidad de la aplicación**

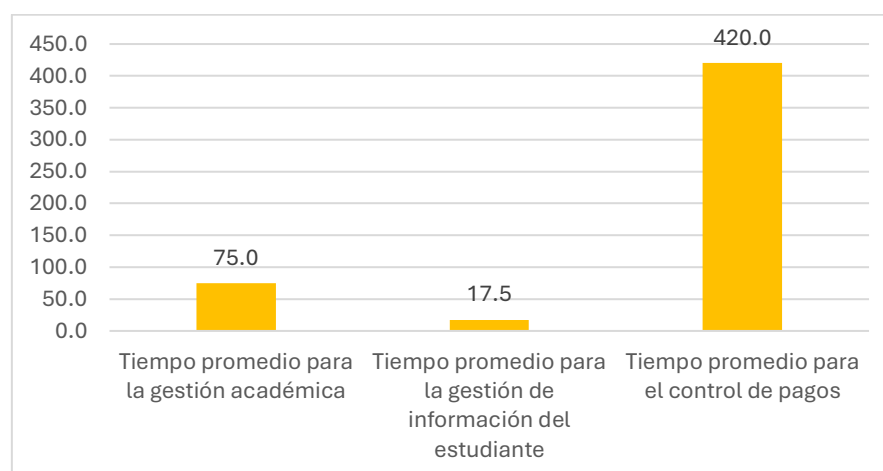
Los indicadores cuantitativos fueron estimados en base a los datos identificados en el planteamiento del problema, contrastados con las funcionalidades desarrolladas. Si bien no se realizó una validación en campo, los resultados funcionales de la aplicación demuestran técnicamente que las tareas automatizadas reducirán significativamente el tiempo y el riesgo de error frente al proceso manual.

**Tabla 61.**

Resultados comparativos en la gestión de matrícula.

Indicador	Situación actual	Situación con la aplicación	Reducción absoluta	Reducción de tiempo
I1. Tiempo promedio para la gestión académica.	75 minutos	5 minutos	70 minutos	93.33%
I2. Tiempo promedio para la gestión de información del estudiante.	17.5 minutos	5 minutos	12.5 minutos	71.43%
I3. Tiempo promedio para el control de pagos.	420 minutos	4 minutos	416 minutos	99.05%

*Nota.* Los valores presentados son aproximaciones obtenidas por el análisis de la situación actual y simulación del prototipo.

**Figura 122.***Situación actual en la gestión de matrícula.*

*Nota.* La figura ilustra el tiempo promedio (minutos) que toma a cada indicador en la situación actual.

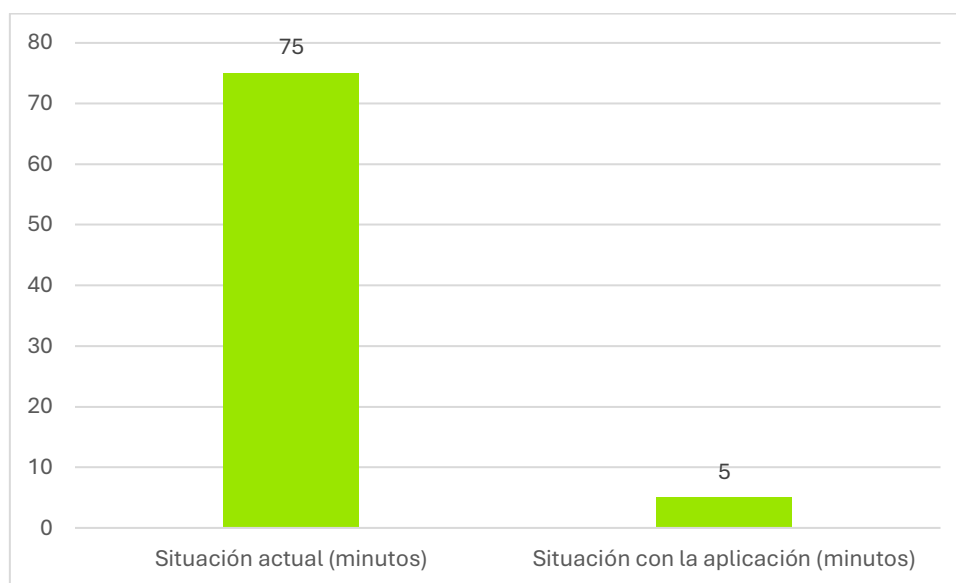
En la **Figura 122**, se observa los valores en la situación actual de la institución para la gestión de matrícula en los tres indicadores definidos. El indicador 1 registra 75 minutos, el

indicador 2 registra 17.5 minutos y el indicador 3 registra 420 minutos, convirtiéndose en el indicador de mayor duración, evidenciando dificultades en la situación actual.

Los indicadores graficados permiten observar que en la situación actual presenta tiempos muy elevados a causa de la dependencia del trabajo manual.

**Figura 123.**

*Indicador 1: Tiempo promedio para la gestión académica.*

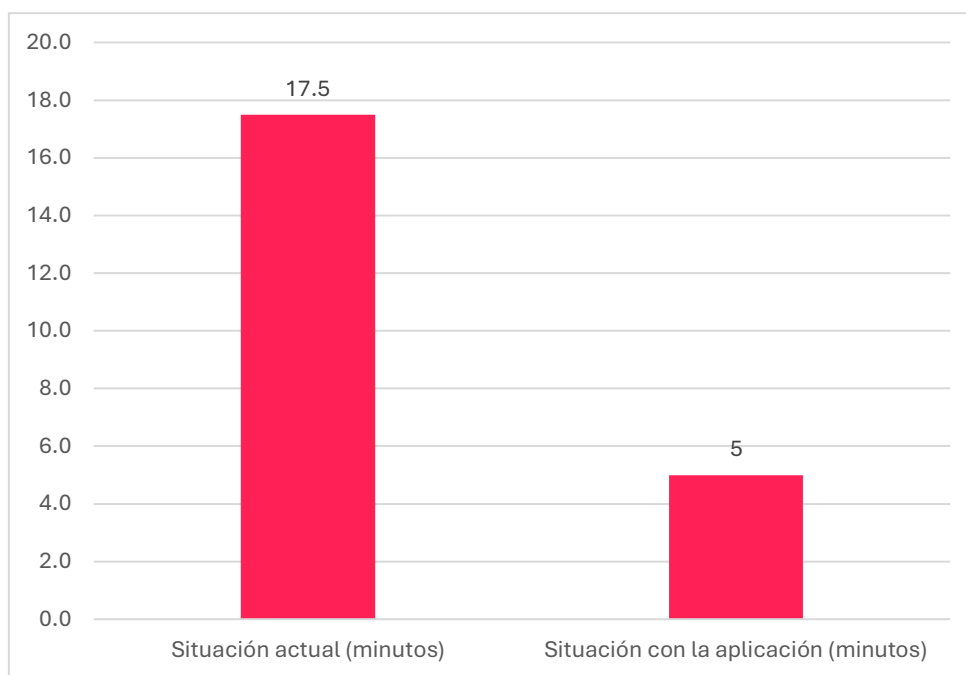


Nota. La figura ilustra la comparación del indicador 1 con la situación actual y con la aplicación.

En la **Figura 123**, se muestra una comparación del indicador 1 entre la situación actual y con el uso de la aplicación, dando una brecha demasiado alta. Esto evidencia que con el uso de la aplicación para la gestión académica reducirá el tiempo en 70 minutos.

**Figura 124.**

*Indicador 2: Tiempo promedio para la gestión de información del estudiante.*

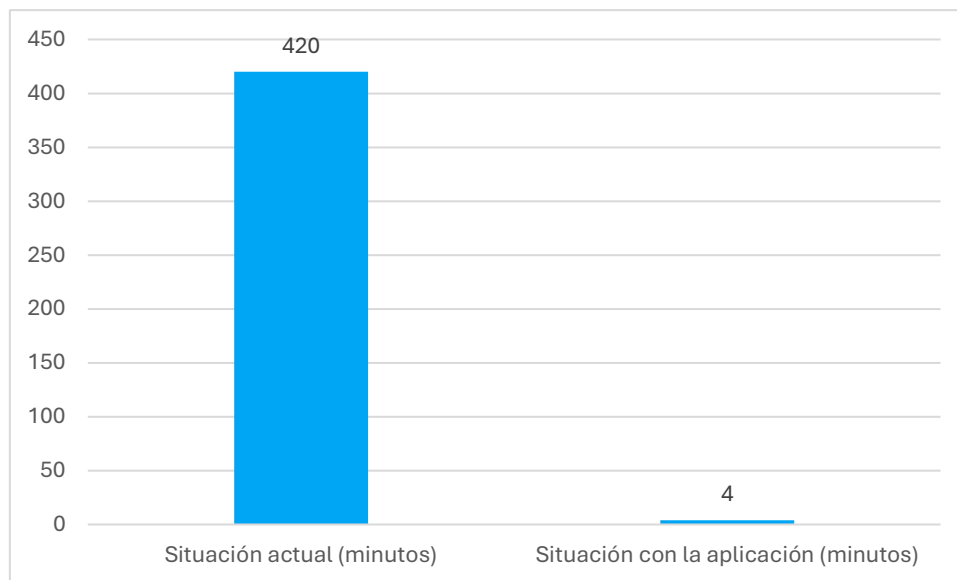


*Nota.* La figura ilustra la comparación del indicador 2 con la situación actual y con la aplicación.

En la **Figura 124**, se observa la comparación del indicador 2, entre la situación actual y con el uso de la aplicación, del mismo modo la brecha es alta. Lo cual evidencia que con el uso de la aplicación para la gestión de información del estudiante el tiempo reducirá en 12.5 minutos.

**Figura 125.**

*Indicador 3: Tiempo promedio para el control de pagos.*

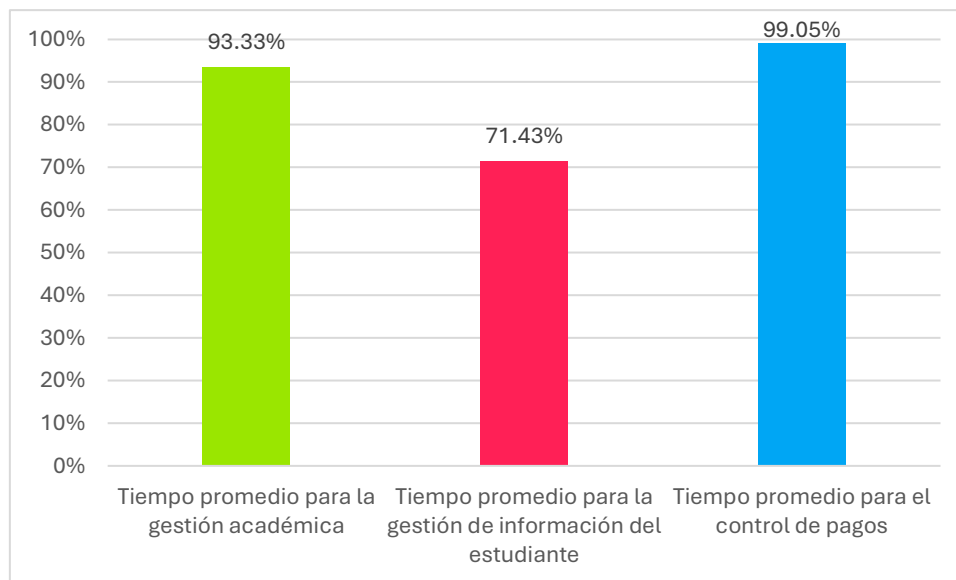


*Nota.* La figura ilustra la comparación del indicador 3 con la situación actual y con la aplicación.

En la **Figura 125**, se observa la comparación del indicador 3 en la situación actual y con el uso de la aplicación. La diferencia es bastante alta, ya que el tiempo actual es de 420 minutos y con el uso de la aplicación se reduce a 4 minutos. Reflejando una mejora drástica en el control de pagos. La figura demuestra que este indicador es el más beneficiado con el uso de la aplicación web.

**Figura 126.**

*Porcentaje de reducción de tiempo con el uso de la aplicación.*



*Nota.* La figura ilustra la reducción del tiempo (%) con el uso de la aplicación para la gestión de matrícula, en las tres dimensiones.

En la **Figura 126**, se sintetiza el porcentaje de reducción obtenido en los tres indicadores analizados. Se observa disminuciones del 93.33% para el indicador 1, 71.43% para el indicador 2 y 99.05% para el indicador 3. Esta grafica permite evidenciar de manera directa la magnitud del impacto en cada indicador, mostrando que cada indicador mejora significativamente en la reducción de tiempo con el uso de la aplicación web.

## CAPÍTULO V

### CONCLUSIONES Y RECOMENDACIONES

#### 1.19. Conclusiones

El presente trabajo de investigación se completó con el desarrollo de una aplicación web mediante el marco de trabajo Scrum. La aplicación tiene los módulos de seguridad, configuración, operación y académico. Para determinar los requerimientos se utilizó los instrumentos de ficha de registro y la guía de entrevista. Los artefactos generados tras cumplir el ciclo de desarrollo de software se pueden observar en las figuras **Figura 109** al **Figura 121**. Culminado el desarrollo se validó el funcionamiento de la aplicación con un usuario final, mediante el instrumento de cuestionario, dando como resultado que la aplicación es funcional en un 100% tal como se puede observar en la **Tabla 59**. Seguidamente se pudo hacer la medición de los tiempos con el que se ejecuta los procesos, mediante el instrumento de ficha de registro como se muestra en la **Tabla 60**. Esto permitió hacer las diversas comparaciones con la situación actual y la situación usando la aplicación para la gestión de matrícula en la Institución Educativa “Nuestra Señora de Guadalupe” con lo cual se concluye:

- Se ha determinado que la aplicación web mejorará de manera significativa la gestión académica, reduciendo el tiempo del proceso en un 93.33%, como se puede observar en la **Figura 123**.
- Se ha determinado que la aplicación web mejorará de manera significativa la gestión de información del estudiante, reduciendo el tiempo del proceso en un 71.43%, como se puede observar en la **Figura 124**.
- Se ha determinado que la aplicación web mejorará de manera significativa el control de pago de pensiones, reduciendo el tiempo del proceso en un 99.05%, siendo el proceso de mayor beneficio ya que diversas tareas fueron automatizadas, se puede observar en la **Figura 125**.

### **1.20. Recomendaciones**

- Se recomienda a la Institución Educativa considerar la implementación de la aplicación web, con el fin de validar su funcionamiento y la eficiencia en condiciones reales.
- Se sugiere incorporar en futuras versiones módulos complementarios, como la gestión de notas, asistencia entre otros, con el propósito de ampliar la funcionalidad y fortalecer la integración del sistema.
- Se recomienda realizar pruebas adicionales de rendimiento y seguridad una vez que la aplicación sea implementada, a fin de asegurar la integridad de los datos y la estabilidad del sistema ante un uso real.

## BIBLIOGRAFÍA

- Alburqueque Arevalo, N. S. (2022). *Implementación de un sistema de matrícula web en la I.E. Ignacio Merino - Piura; 2020 [Tesis pregrado, Universidad Católica Los Ángeles de Chimbote]*. Repositorio institucional. <https://repositorio.uladech.edu.pe/handle/20.500.13032/26251?show=full>
- Apache HTTP Server Project. (2024). *About the Apache HTTP Server Project*. Retrieved 12 de diciembre de 2024, from The Apache Software Foundation: <https://httpd.apache.org/>
- AWS. (2024). *¿Cuál es la diferencia entre la arquitectura monolítica y la de microservicios?* Retrieved 12 de diciembre de 2024, from AWS: <https://aws.amazon.com/es/compare/the-difference-between-monolithic-and-microservices-architecture/#:~:text=Una%20arquitectura%20monol%C3%ADtica%20es%20un,de%20datos%20dentro%20del%20sistema.>
- AWS. (2025). *¿Qué son las pruebas unitarias?* AWS: <https://aws.amazon.com/es/what-is/unit-testing/>
- Bernal, C. (2006). *Metodología de la investigación: Para administración, economía, humanidades y ciencias sociales*. Pearson Educación.
- Blanco, I., y Quesada, V. M. (2014). *La gestión académica, criterio clave de la calidad de la gestión en las instituciones de educación superior (Artículo)*. Repositorio. [https://www.researchgate.net/publication/228908649\\_LA\\_GESTION\\_ACADEMICA\\_CRITERIO\\_CLAVE\\_DE\\_LA\\_CALIDAD\\_DE\\_LA\\_GESTION\\_DE\\_LAS\\_INSTITUCIONES\\_DE\\_EDUCACION\\_SUPERIOR](https://www.researchgate.net/publication/228908649_LA_GESTION_ACADEMICA_CRITERIO_CLAVE_DE_LA_CALIDAD_DE_LA_GESTION_DE_LAS_INSTITUCIONES_DE_EDUCACION_SUPERIOR)
- Calamani Salas, J. M. (2020). *SISTEMA DE INFORMACIÓN PARA EL REGISTRO Y SEGUIMIENTO ACADÉMICO [Tesis de pregrado, UNIVERSIDAD PÚBLICA DE EL ALTO]*. Repositorio institucional. <http://repositorio.upea.bo/jspui/handle/123456789/223>
- Centre Metal Lúrgic. (s.f.). *GESTIÓN DE LA TESORERÍA (Cash Management)*. Centrum: <https://www.centrem.cat/ecomu/upfiles/publicacions/gestion.pdf>

- Chiuca Paredes, A., y Laguna Marcos, F. L. (2023). *Sistema web para la gestión de matrícula de la Institución Educativa Mártir José Olaya, Musho, 2023 [Tesis pregrado, Universidad César Vallejo]*. Repositorio institucional. <https://hdl.handle.net/20.500.12692/134459>
- Cohn, M. (2025). *Mountain Goat Software*. Planning Poker: An Agile Estimating and Planning Technique: <https://www.mountaingoatsoftware.com/agile/planning-poker>
- Dioses Madrid, W. F., Núñez Astudillo, H. X., Ventura Sánchez, C. G., y Zapata Bacón, I. A. (2022). *Implementación de un sistema web, utilizando metodología Scrum para mejorar los procesos de matrícula y control de pagos en la Institución Educativa Particular El Triunfo [Tesis pregrado, Universidad Nacional de Piura]*. Repositorio institucional. <https://repositorio.unp.edu.pe/handle/20.500.12676/3688>
- DoneTonic. (2025). *DoneTonic*. Burndown Chart • Métricas agile para medir tu progreso.: <https://donetonic.com/es/burndown-chart-metricas-agile/>
- Flórez Fernández, H. A. (2012). *Programación Orientada a Objetos Usando JAVA*. Bogotá: ECOE EDICIONES.
- Franco Magallanes, A. I., y Cerezo Escobar, C. A. (2024). *Desarrollo de un sistema de gestión para el control de estudiantes en los cursos del centro de emprendimiento y optimización municipal de Samborondón (Ceyom)*. <https://repositorio.ecotec.edu.ec/handle/123456789/1124>
- Freire Mendoza, B. A., y Sánchez Estrella, A. J. (2023). *Desarrollo de una aplicación web para la sistematización de la gestión de los horarios de docentes, estudiantes y aulas para la Universidad Técnica de Cotopaxi – Extensión La Maná [Tesis pregrado, UNIVERSIDAD TÉCNICA DE COTAPAXI]*. Repositorio institucional. <http://repositorio.utc.edu.ec/handle/27000/11316>
- Git. (2024). *Git*. Retrieved 09 de diciembre de 2024, from Git: <https://git-scm.com/>
- Haro Sarango, A. F., Chisag Pallmay, E. R., Ruiz Sarzosa, J. P., y Caicedo Pozo, J. E. (2024). *Tipos y clasificación de las investigaciones*. LATAM Revista Latinoamericana de Ciencias Sociales y Humanidades.

- Hernández Sampieri, R., Fernández Collado, C., y Baptista Lucio, P. (2006). *Metodología de la investigación*. McGraw-Hili.
- Hernández Sampieri, R., Fernández Collado, C., y Baptista Lucio, P. (2006). *Metodología de la investigación*. McGraw-Hili.
- JetBrains. (2024). *DataGrip Documentation*. Retrieved 12 de diciembre de 2024, from JetBrains: <https://www.jetbrains.com/datagrip/documentation>
- Joyanes Aguilar, L. (2012). *Computación en la Nube Estrategias de Cloud Computing en las Empresas*. México: Alfaomega.
- Joyanes Aguilar, L., y Zahonero Martinez, I. (2011). *Programación en Java 6 Algoritmos y programación orientada a objetos*. México: McGraw-Hill.
- Kendall, K. E., y Kendall, J. E. (2011). *Análisis y Diseño De Sistemas* (Octava ed.). PEARSON EDUCACIÓN.
- Laravel. (2024). *Introducción a Laravel*. Retrieved 09 de diciembre de 2024, from Laravel: <https://laravel.com/>
- Luján Mora, S. (2002). *Programación de aplicaciones web historia, principios básicos y clientes web*. Editorial Club Universitario (ECU). <https://dialnet.unirioja.es/servlet/libro?codigo=204176>
- Mannino, M. V. (2007). *Administración de bases de datos Diseño y desarrollo de aplicaciones* (Tercera ed.). McGraw-Hill.
- Martínez Ruiz, H. (2012). *Metodología de la investigación*. Cengage Learning Editores, S.A.
- Microsoft. (2024). *Introducing Azure DevOps*. Retrieved 12 de diciembre de 2024, from Microsoft: <https://azure.microsoft.com/en-us/>
- Mozilla Developer Network. (2024). *What is JavaScript?* Retrieved 13 de diciembre de 2024, from Mozilla Developer Network: [https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First\\_steps/What\\_is\\_JavaScript](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript)
- MySQL. (2024). *What is MySQL?* Retrieved 09 de diciembre de 2024, from MySQL: <https://dev.mysql.com>

- Palacio, M. (2024). *SCRUM MASTER* (Versión 4.0 ed.). Scrum Manager.
- Perez Alvarez, L. P. (2023). *Implementación de un sistema web para la gestión del proceso de matrícula en un colegio privado de Trujillo, 2023 [Tesis pregrado, Universidad privada del Norte]*. Repositorio Institucional. <https://hdl.handle.net/11537/37731>
- PHP Group. (2024). *What is PHP?* Retrieved 9 de Diciembre de 2024, from PHP Group: <https://www.php.net/manual/es/intro-what-is.php>
- phpunit. (2025). *¿What is PHPUnit?* phpunit: <https://phpunit.de/index.html>
- Pressman, R. S. (2010). *INGENIERÍA DEL SOFTWARE. UN ENFOQUE PRÁCTICO* (Séptima ed.). The McGraw-Hill.
- Pulido Fuentes, K. N., y Martínez, M. J. (2022). *Desarrollo de una aplicación web para la inscripción de cursos libres del Departamento de computación de la UNAN-LEÓN, mediante la utilización del framework Laravel Livewire de PHP [Tesis pregrado, Universidad Nacional Autónoma de Nicaragua UNAN LEÓN]*. Repositorio institucional. <http://riul.unanleon.edu.ni:8080/jspui/handle/123456789/9717>
- Recalde Araujo, H. M., y Zurita Lara, B. N. (2020). *Sistema web para la gestión académica y administrativa de empresa de capacitación profesional DIENAV [Tesis de Pregrado, UNIVERSIDAD TECNOLÓGICA ISRAEL]*. Repositorio Institucional. <https://repositorio.uisrael.edu.ec/handle/47000/2489>
- Red Educa. (2024). *Matrícula*. RED EDUCA: <https://www.rededuca.net/contexto-educativo/m/matricula>
- Rugel Ruiz, J. K. (2022). *Implementación de un sistema web de gestión de matrícula en la I.E. Inmaculada Concepción - Tumbes; 2020 [Tesis pregrado, Universidad Católica los Ángeles de Chimbote]*. Repositorio institucional. <https://hdl.handle.net/20.500.13032/28935>
- Satpathy, T. (2017). *Una guía para el cuerpo de conocimiento de SCRUM (Guía SBoK™)* (Tercera ed.). SCRUMstudy™.

Schwaber, K., y Sutherland, J. (2020). *La Guía Scrum La Guía Definitiva de Scrum: Las Reglas del Juego*. <https://scrumguides.org/>

Sommerville, I. (2011). *Ingeniería De Software*. México: PEARSON EDUCACIÓN.

Tailwind Labs. (12 de diciembre de 2024). *Tailwind CSS Documentation*. Tailwind CSS: <https://tailwindcss.com/>

Turban, E., Pollard, C., y Wood, G. (2018). *Information Technology For Management* (11TH ed.). Wiley.

## ANEXOS

**Anexo A:** Matriz de consistencia

**Título:** Aplicación Web para la gestión de matrícula de estudiantes en la Institución Educativa Nuestra Señora De Guadalupe, 2025

<b>PROBLEMAS</b>	<b>OBJETIVOS</b>	<b>VARIABLES</b>	<b>METODOLOGÍA</b>
<p><b>PROBLEMA GENERAL</b></p> <p>¿En qué medida el uso de la aplicación web desarrollado mediante el marco de trabajo Scrum, mejorará la gestión de matrícula de estudiantes en la Institución Educativa “Nuestra Señora De Guadalupe”?</p>	<p><b>OBJETIVO GENERAL</b></p> <p>Determinar en qué medida el uso de la aplicación web desarrollado mediante el marco de trabajo Scrum, mejorará la gestión de matrícula de estudiantes en la Institución Educativa “Nuestra Señora De Guadalupe”.</p>	<p><b>VARIABLE DE INTERÉS</b></p> <p>X: Gestión de matrícula</p>	<p><b>ENFOQUE DE INVESTIGACIÓN</b></p> <p>Cuantitativa</p>
<p><b>PROBLEMAS ESPECÍFICOS</b></p> <p>a. ¿En qué medida el uso de la aplicación web reducirá el tiempo en la gestión académica?</p> <p>b. ¿En qué medida el uso de la aplicación web reducirá el tiempo en la gestión de información del estudiante?</p>	<p><b>OBJETIVOS ESPECÍFICOS</b></p> <p>a. Determinar en qué medida el uso de la aplicación web reducirá el tiempo en la gestión académica.</p> <p>b. Determinar en qué medida el uso de la aplicación web reducirá el tiempo en la gestión de información del estudiante.</p>	<p><b>VARIABLES DESCRIPTIVAS</b></p> <p>X1: Gestión académica. X2: Gestión de información del estudiante. X3: Control de pagos.</p>	<p><b>TIPO DE INVESTIGACIÓN</b></p> <p>Observacional, retrospectiva, transversal y descriptivo.</p> <p><b>NIVEL DE INVESTIGACIÓN</b></p> <p>Descriptivo</p> <p><b>DISEÑO</b></p> <p>No experimental</p> <p><b>POBLACIÓN</b></p> <p>La población está compuesta por los procesos de gestión académica, gestión de información del estudiante y el control de pagos en la Institución Educativa "Nuestra</p>

<p>c. ¿En qué medida el uso de la aplicación web reducirá el tiempo en el control de pagos?</p>	<p>c. Determinar en qué medida el uso de la aplicación web reducirá el tiempo en el control de pagos.</p>		<p>Señora De Guadalupe" durante el año 2025.</p> <p><b>MUESTRA</b> Dado que la población en esta investigación está conformada por un total de 3 procesos (gestión académica, gestión de información del estudiante y control de pago de pensiones), y considerando que se tiene acceso directo a todos los procesos involucrados, se optará por un muestreo censal.</p> <p><b>TÉCNICA</b></p> <ul style="list-style-type: none"> <li>• Análisis documental</li> <li>• Entrevista</li> <li>• Encuesta</li> </ul> <p><b>INSTRUMENTO</b></p> <ul style="list-style-type: none"> <li>• Ficha de registro</li> <li>• Guía de entrevista</li> <li>• Cuestionario</li> </ul>
---	---	--	--



**Anexo C: Diccionario de datos**

<b>alumno</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
persona_id	Llave foránea	INT			Si	No	

<b>boleta</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
numero_boleta	Número correlativo asignado a la boleta.	VARCHAR	10			No	Si
anio	Año actual.	CHAR	4			No	
fecha_emision	Fecha de emisión de la boleta.	DATETIME				No	
razón_social	Nombre legal de la persona jurídica o personal.	VARCHAR	100			No	
ruc_dni	Número de Ruc o DNI.	VARCHAR	12			No	
igv	Monto (porcentaje) del impuesta general a las ventas.	DECIMAL				No	
total	Importe total.	DECIMAL				No	

<b>boleta_detalle</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
concepto	Descripción de la venta.	VARCHAR	100			Si	
cantidad	Número de unidades del servicio.	CHAR	1			No	
precio_unitario	Precio unitario de la venta.	DECIMAL				No	
sub_total	Monto sin recargos.	DECIMAL				No	
descuento_aplicado	Descuento aplicado a matrícula, pensiones o matrícula y pensiones.	DECIMAL				Si	
igv	Monto (porcentaje) del impuesto general a las ventas.	DECIMAL				No	
total	Importe total de la venta.	DECIMAL				No	
boleta_id	Llave foránea	INT				No	

<b>ciclo_academico</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
nombre	Nombre del ciclo académico.	VARCHAR	100			No	

descripción	Descripción del ciclo académico.	VARCHAR	200			Si	
fecha_inicio	Fecha de inicio del ciclo académico.	DATETIME				No	
fecha_final	Fecha de finalización del ciclo académico.	DATETIME				No	
año	Año en el que se realizará el ciclo académico.	VARCHAR	4			No	

<b>ciclo_academico_descuento</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
Id	Identificador (autoincremental)	INT		Si		No	Si
descuento_id	Llave foránea	INT			Si	No	
ciclo_academico_id	Llave foránea	INT			Si	No	

<b>clave</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
clave_hssh	Contraseña hasheada.	VARCHAR	50			No	
clave_reset	Contraseña hasheada de respaldo.	VARCHAR	50			No	

<b>configuracion_financiera</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
anio	Año actual	VARCHAR	4			No	
monto_matricula	Precio de la matrícula.	DECIMAL				No	
monto_pension	Precio de la pensión.	DECIMAL				No	
monto_mora	Monto de la mora por retraso en el pago de la pensión.	DECIMAL				No	
fecha_inicio_pago	Fecha en el que inicia el pago de las pensiones.	DECIMAL				No	
fecha_fin_pago	Fecha en el que finaliza el pago de las pensiones.	DECIMAL				No	
ciclo_academico_id	Llave foránea.	DECIMAL			Si	No	

<b>contacto</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
telefono	Número de teléfono.	VARCHAR	9			Si	
celular	Número de celular.	VARCHAR	9			No	
correo	Correo electrónico.	VARCHAR	50			Si	
Persona_id	Llave foránea.	INT			Si	No	
Contacto_emergencia_id	Llave foránea.	INT			Si	No	

<b>contacto_emergencia</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
nombre	Nombre del apoderado o familiar del estudiante.	VARCHAR	100			SI	
parentesco	Parentesco.	VARCHAR	20			SI	
celular	Número de celular del apoderado.	VARCHAR	9			SI	

telefono	Número de teléfono del apoderado.	VARCHAR	9			SI	
----------	-----------------------------------	---------	---	--	--	----	--

descuento							
Nombre De columna	Descripción	Tipo de dato	Tamaño	PK	FK	Nulo	Único
id	Identificador (auto incremental)	INT		Si		No	Si
nombre	Nombre del descuento.	VARCHAR	50			No	
descripcion	Descripción del descuento.	VARCHAR	100			Si	
monto	Monto del descuento (s/).	DECIMAL				Si	
porcentaje	Porcentaje del descuento (%).	DOUBLE				Si	

descuento_aplicado							
Nombre De columna	Descripción	Tipo de dato	Tamaño	PK	FK	Nulo	Único
id	Identificador (auto incremental)	INT		Si		No	Si
nombre	Nombre del descuento donde se aplicará.	VARCHAR	50			No	

<b>documento</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
numero_documento	Número del documento (DNI, carné de extranjería.).	VARCHAR	20			No	
persona_id	Llave foránea.	INT			Si	No	
tipo_documento_id	Llave foránea.	INT			Si	No	

<b>grado</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
nombre	Nombre del grado.	VARCHAR	20			No	
seccion_id	Llave foránea.	INT			Si	No	
nivel_id	Llave foránea.	INT			Si	No	

<b>matricula</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
grado_id	Llave foránea.	INT			Si	No	
tipo_matricula_id	Llave foránea.	INT			Si	No	
cantidad_pensiones	Indica el número de pensiones que tiene la matrícula.	CHAR	1			No	
ciclo_academico_id	Llave foránea.	INT			Si	No	
alumno_id	Llave foránea.	INT			Si	No	

<b>menu</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
menu	Nombre del menú de la aplicación.	VARCHAR	50			No	
descripcion	Descripción del menú.	VARCHAR	100			Si	
icono	Ícono del menú.	VARCHAR	100			Si	
ruta	Ruta del menú.	VARCHAR	50			Si	

nivel_parentesco	Clave única para indicar a que menú pertenece el submenú.	VARCHAR	50			Si	Si
parentesco	Clave única para indicar que el menú no tiene submenús.	VARCHAR	50			Si	Si
nivel	Indica el nivel del menú (1 menú, 2 submenú).	CHAR	1			No	
orden	Indica el orden en que se mostrará el menú.	CHAR	1			No	

<b>metodo_pago</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
nombre	Nombre del método de pago.	VARCHAR	50			No	
Descripción	Descripción del método de pago.	VARCHAR	100			Si	

<b>mora</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si

monto_mora	Monto de la mora que se registra cada día.	DECIMAL				No	
dias_mora	Número de días que se va acumulando la mora.	INT				No	
monto_total	Suma total de la mora acumulada.	DECIMAL				No	
pension_id	Llave foránea.	INT			Si	No	

<b>nivel</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
nombre	Nombre del nivel (inicial, primaria y secundaria).	VARCHAR	50			No	
Descripción	Descripción del nivel.	VARCHAR	100			Si	

<b>operacion</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
monto	Monto total de la operación.	DECIMAL				No	
descuento_id	Llave foránea.	INT			Si	No	
tipo_operacion_id	Llave foránea.	INT			Si	No	
usuario_id	Llave foránea.	INT			Si	No	
ciclo_academico_id	Llave foránea.	INT			Si	No	
descuento_aplicado_id	Llave foránea.	INT			Si	No	
metodo_pago_id	Llave foránea.	INT			Si	No	
boleta_id	Llave foránea.	INT			Si	No	

<b>pension</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
numero_pension	Indica el número de pensión.	CHAR	1			No	
monto_matricula	Indica el precio de la matrícula.	DECIMAL				No	

monto_pension	Indica el precio de la pensión.	DECIMAL				No	
fecha_vencimiento	Indica el plazo máximo a pagar la pensión.	DATETIME					
mes_pago	Indica el mes correspondiente de la pensión.	VARCHAR	10			No	
matricula_id	Llave foránea.	INT			Si	No	

<b>permiso</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
create	Indica si se tiene permiso para realizar la operación de CREAR.	CHAR	1			No	
read	Indica si se tiene permiso para realizar la operación de LECTURA.	CHAR	1			No	
update	Indica si se tiene permiso para realizar la operación EDITAR	CHAR	1			No	
delete	Indica si se tiene permiso para realizar la operación de ELIMINAR.	CHAR	1			No	
descripcion	Descripción del permiso.	VARCHAR	100			Si	

<b>persona</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
nombre	Nombre de la persona.	VARCHAR	80			No	
apellido_paterno	Apellido paterno de la persona.	VARCHAR	50			No	
apellido_materno	Apellido materno de la persona.	VARCHAR	50			No	
fecha_nacimiento	Fecha de nacimiento de la persona.	DATETIME				No	
direccion_domicilio	Dirección o domicilio completo de la persona.	VARCHAR	250			No	
sexo_id	Llave foránea.	INT			Si	No	
ubigeo_id	Llave foránea	INT			Si	Si	

<b>rol</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
nombre	Nombre del rol.	VARCHAR	50			No	
Descripción	Descripción del rol	VARCHAR	100			Si	

<b>rol_menu</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
permiso_id	Llave foránea.	INT			Si	No	
menu_id	Llave foránea.	INT			Si	No	
rol_id	Llave foránea.	INT			Si	No	

<b>seccion</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
nombre	Nombre de la sección.	VARCHAR	50			No	

<b>serie_boleta</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si

anio	Año actual.	VARCHAR	4			No	
ultimo_numero	Indica el ultimo número de boleta emitido.	VARCHAR	10			No	

sexo							
Nombre De columna	Descripción	Tipo de dato	Tamaño	PK	FK	Nulo	Único
id	Identificador (autoincremental)	INT		Si		No	Si
nombre	“Masculino”, “Femenino”	VARCHAR	20			No	
abreviatura	“M”, ”F”.	CHAR	2			No	

tipo_documento							
Nombre De columna	Descripción	Tipo de dato	Tamaño	PK	FK	Nulo	Único
id	Identificador (autoincremental)	INT		Si		No	Si
abreviatura	Indica la abreviatura del tipo de documento.	VARCHAR	10			No	
nombre_doc	Indica el nombre del documento.	VARCHAR	50			No	

numero_digitos	Indica el número de dígitos que tiene el documento.	VARCHAR	2			NO	
----------------	---	---------	---	--	--	----	--

<b>tipo_matricula</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
nombre	Nombre del tipo de matrícula.	VARCHAR	50			No	
Descripción	Descripción del tipo de matrícula.	VARCHAR	100			Si	

<b>tipo_operacion</b>							
<b>Nombre De columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (autoincremental)	INT		Si		No	Si
nombre	Nombre del tipo de operación.	VARCHAR	50			No	

Descripción	Descripción del tipo de operación.	VARCHAR	100			Si	
-------------	------------------------------------	---------	-----	--	--	----	--

<b>ubigeo</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
codigo_departamento	Indica el código del departamento.	VARCHAR	2			No	
departamento	Indica el nombre del departamento.	VARCHAR	50			No	
codigo_provincia	Indica el código de la provincia.	VARCHAR	2			No	
provincia	Indica el nombre de la provincia.	VARCHAR	100			No	
codigo_distrito	Indica el código del distrito.	VARCHAR	2			No	
distrito	Indica el nombre del distrito.	VARCHAR	100			No	
capital	Indica el nombre de la capital del departamento.	VARCHAR	100			No	
altitud	Indica la altitud del departamento.	DECIMAL				No	

latitud	Indica la latitud del departamento.	DECIMAL				No	
longitud	Indica la longitud del departamento.	DECIMAL				No	

<b>usuario</b>							
<b>Nombre de columna</b>	<b>Descripción</b>	<b>Tipo de dato</b>	<b>Tamaño</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Único</b>
id	Identificador (auto incremental)	INT		Si		No	Si
usuario	Nombre del usuario para iniciar sesión en la aplicación (Número de documento).	VARCHAR	20			No	
clave_id	Llave foránea.	INT			Si	No	
rol_id	Llave foránea.	INT			Si	NO	
persona_id	Llave foránea.	INT			Si	No	

## Anexo D: Código fuente de la prueba unitaria de autenticación de usuarios.

```

1  class AuthenticationTest extends TestCase
2  {
3      use RefreshDatabase;
4
5      public function setUp(): void
6      {
7          parent::setUp();
8
9          Artisan::call('migrate:fresh');
10         Artisan::call('db:seed');
11     }
12
13     public function test_login_screen_can_be_rendered(): void
14     {
15         $response = $this->get('/Login');
16
17         $response
18             ->assertOk()
19             ->assertSeeVolt('pages.auth.login');
20     }
21
22     public function test_users_can_authenticate_using_the_login_screen(): void
23     {
24         $person = Person::create([
25             'sexo_id' => 1,
26             'name' => 'Juan',
27             'paternal_surname' => 'Pérez',
28             'maternal_surname' => 'López',
29             'home_address' => 'Calle Falsa 123',
30         ]);
31
32         $role = Role::create([
33             'name' => 'Admin',
34             'description' => 'Rol de administrador',
35         ]);
36
37         $user = User::create([
38             'username' => '12345678',
39             'password' => bcrypt('admin'),
40             'person_id' => $person->id,
41             'role_id' => $role->id,
42             'state' => 1,
43             'password_reset' => bcrypt('admin')
44         ]);
45
46         Volt::test('pages.auth.login')
47             ->set('form.username', '12345678')
48             ->set('form.password', 'admin')
49             ->call('login')
50             ->assertHasNoErrors()
51             ->assertRedirect('/dashboard');
52
53         $this->assertAuthenticated();
54     }
55 }
56
57 public function test_users_can_not_authenticate_with_invalid_password(): void
58 {
59     $person = Person::create([
60         'sexo_id' => 1,
61         'name' => 'Juan',
62         'paternal_surname' => 'Pérez',
63         'maternal_surname' => 'López',
64         'home_address' => 'Calle Falsa 123',
65     ]);
66
67     $role = Role::create([
68         'name' => 'Admin',
69         'description' => 'Rol de administrador',
70     ]);
71
72     $user = User::create([
73         'username' => 'admin',
74         'password' => bcrypt('admin'),
75         'person_id' => $person->id,
76         'role_id' => $role->id,
77         'state' => 1,
78         'password_reset' => bcrypt('admin')
79     ]);
80
81     $response = $this->post('/login', [
82         'username' => 'admin',
83         'password' => 'incorrecta',
84     ]);
85
86     $this->assertGuest();
87 }
88 }
89

```

**Anexo E:** Código fuente de la prueba unitaria de gestión de roles.

```
1
2 class RoleMenuTest extends TestCase
3 {
4     use RefreshDatabase;
5
6     public function setUp(): void
7     {
8         parent::setUp();
9
10        Artisan::call('migrate:fresh');
11        Artisan::call('db:seed');
12    }
13
14    public function test_list_returns_grouped_roles()
15    {
16        $service = app(RoleMenuService::class);
17        $result = $service->list();
18        $this->assertCount(1, $result);
19    }
20
21    public function test_store_creates_role_and_rolemenus()
22    {
23        $user = User::find(1);
24        $this->actingAs($user);
25        $service = app(RoleMenuService::class);
26        $menu = Menu::first();
27        $perm = Permission::first();
28
29        $payload = [
30            'name' => 'Nuevo rol',
31            'description' => 'descripcion',
32            'permissions' => [
33                [
34                    'menu_id' => $menu->id,
35                    'permission_id' => $perm->id,
36                ],
37            ],
38        ];
39
40        $service->store($payload);
41        $expectedName = strtoupper(trim($payload['name']));
42        $expectedDesc = strtoupper(trim($payload['description']));
43
44        $this->assertDatabaseHas('roles', [
45            'name' => $expectedName,
46            'description' => $expectedDesc,
47            'created_by' => $user->id,
48        ]);
49        $role = Role::where('name', $expectedName)->first();
50        $this->assertNotNull($role);
51        $this->assertDatabaseHas('role_menus', [
52            'role_id' => $role->id,
53            'menu_id' => $menu->id,
54            'permission_id' => $perm->id,
55            'created_by' => $user->id,
56        ]);
57    }
58
59
```

```

59
60 public function test_update_replaces_roles_or_404()
61 {
62     $service = app(RoleMenuService::class);
63     $user = User::first();
64     $this->actingAs($user);
65     $role = Role::find(1);
66     $menuOld = Menu::first();
67     $permOld = Permission::first();
68
69     RoleMenu::create([
70         'role_id' => $role->id,
71         'menu_id' => $menuOld->id,
72         'permission_id' => $permOld->id,
73         'state' => '1',
74         'created_by' => $user->id
75     ]);
76
77     $menuNew = Menu::skip(1)->first();
78     $permNew = Permission::skip(1)->first();
79
80     $payload = [
81         'name' => 'nuevo nombre',
82         'description' => 'nueva desc',
83         'permissions' => [
84             [
85                 'menu_id' => $menuNew->id,
86                 'permission_id' => $permNew->id,
87             ]
88         ],
89     ];
90     $service->update($role->id, $payload);
91     $role->refresh();
92     $this->expectException(NotFoundHttpException::class);
93     $service->update(999999, $payload);
94 }
95
96 public function test_delete_role_related_as_inactive_or_404()
97 {
98
99     $user = User::find(1);
100    $this->actingAs($user);
101    $service = app(RoleMenuService::class);
102    $role = Role::find(1);
103
104    $userWithRole = User::create([
105        'username' => '12345678',
106        'password' => bcrypt('admin'),
107        'person_id' => 1,
108        'role_id' => $role->id,
109        'state' => 1,
110        'password_reset' => bcrypt('admin')
111    ]);
112
113    $menu = Menu::first();
114    $perm = Permission::first();
115
116    RoleMenu::create([
117        'role_id' => $role->id,
118        'menu_id' => $menu->id,
119        'permission_id' => $perm->id,
120        'state' => '1',
121        'created_by' => $user->id
122    ]);
123
124    $service->delete($role->id);
125    $role->refresh();
126    $userWithRole->refresh();
127    $this->assertEquals('0', $role->state);
128    $this->assertDatabaseHas('role_menus', [
129        'role_id' => $role->id,
130        'state' => '0',
131    ]);
132
133    $this->assertDatabaseHas('users', [
134        'id' => $userWithRole->id,
135        'role_id' => $role->id,
136        'state' => '0',
137    ]);
138    $this->expectException(NotFoundHttpException::class);
139    $service->delete(999999);
140 }
141 }
142

```

## Anexo F: Código fuente de la prueba unitaria de gestión de usuarios.

```

1
2 class UserTest extends TestCase
3 {
4     use RefreshDatabase;
5
6     public function setUp(): void
7     {
8         parent::setUp();
9
10        Artisan::call('migrate:fresh');
11        Artisan::call('db:seed');
12    }
13
14    public function test_user_can_be_created()
15    {
16        $authUser = User::first();
17        Auth::shouldReceive('user')->andReturn($authUser);
18
19        $personService = Mockery::mock(PersonService::class);
20        $documentService = Mockery::mock(DocumentService::class);
21        $ecService = Mockery::mock(EmergencyContactService::class);
22        $contactService = Mockery::mock(ContactService::class);
23        $service = new UserService(
24            $personService,
25            $documentService,
26            $ecService,
27            $contactService
28        );
29
30        $data = [
31            'person' => [
32                'sexo_id' => 1,
33                'name' => 'Juan',
34                'paternal_surname' => 'Perez',
35                'maternal_surname' => 'Lopez',
36                'home_address' => 'Av Lima',
37            ],
38            'document' => [
39                'document_type_id' => 1,
40                'number' => '12345678',
41            ],
42            'emergency_contact' => [
43                'full_name' => 'Maria Perez',
44                'relationship' => 'Madre',
45                'cell_phone' => '999888777',
46                'telephone' => '0123456',
47                'email' => 'maria@mail.com',
48            ],
49            'contact' => [
50                'cell_phone' => '999555444',
51                'telephone' => '01444555',
52                'email' => 'juan@mail.com',
53            ],
54            'role_id' => 1
55        ];
56
57        $realPerson = Person::create([
58            'sexo_id' => 1,
59            'name' => 'JUAN',
60            'paternal_surname' => 'PEREZ',
61            'maternal_surname' => 'LOPEZ',
62            'home_address' => 'AV LIMA',
63            'created_by' => $authUser->id
64        ]);
65
66        $realDocument = Document::create([
67            'person_id' => $realPerson->id,
68            'document_type_id' => 1,
69            'number' => '12345678',
70            'created_by' => $authUser->id
71        ]);
72
73        $realEmergencyContact = EmergencyContact::create([
74            'full_name' => 'MARIA PEREZ',
75            'relationship' => 'MADRE',
76            'cell_phone' => '999888777',
77            'telephone' => '0123456',
78            'email' => 'MARIA@MAIL.COM',
79            'created_by' => $authUser->id
80        ]);
81
82        $realContact = Contact::create([
83            'person_id' => $realPerson->id,
84            'emergency_contact_id' => $realEmergencyContact->id,
85            'cell_phone' => '999555444',
86            'telephone' => '01444555',
87            'email' => 'JUANG@MAIL.COM',
88            'usuario_crea' => $authUser->id
89        ]);
90
91        $personService
92            ->shouldReceive('create')
93            ->once()
94            ->with($data['person'])
95            ->andReturn($realPerson);
96
97        $documentService
98            ->shouldReceive('create')
99            ->once()
100           ->with($realPerson, $data['document'])
101           ->andReturn($realDocument);
102
103        $ecService
104            ->shouldReceive('create')
105            ->once()
106            ->with($data['emergency_contact'])
107            ->andReturn($realEmergencyContact);
108
109        $contactService
110            ->shouldReceive('create')
111            ->once()
112            ->with($realPerson, $realEmergencyContact, $data['contact'])
113            ->andReturn($realContact);
114
115        $created = $service->create($data);
116
117        $this->assertInstanceOf(User::class, $created);
118        $this->assertDatabaseHas('users', [
119            'id' => $created->id,
120            'username' => '12345678',
121            'person_id' => $realPerson->id,
122            'role_id' => 1,
123        ]);
124
125        $this->assertTrue(Hash::check('12345678', $created->password));
126        $this->assertEquals($authUser->id, $created->created_by);
127    }
128

```

```

1
2 public function test_user_can_be_updated()
3 {
4     $authUser = User::first();
5     $this->assertNotNull($authUser, 'No hay usuarios en la DB de test - verifica los seeders.');
6     Auth::shouldReceive('user')->andReturn($authUser);
7
8     $personService = Mockery::mock(PersonService::class);
9     $documentService = Mockery::mock(DocumentService::class);
10    $secService = Mockery::mock(EmergencyContactService::class);
11    $contactService = Mockery::mock(ContactService::class);
12
13    $service = new UserService(
14        $personService,
15        $documentService,
16        $secService,
17        $contactService
18    );
19
20    $existingPerson = Person::create([
21        'sexo_id' => 1,
22        'name' => 'PEDRO',
23        'paternal_surname' => 'SUAREZ',
24        'maternal_surname' => 'RAMOS',
25        'home_address' => 'CALLIE 1',
26        'created_by' => $authUser->id
27    ]);
28
29    $existingDocument = Document::create([
30        'person_id' => $existingPerson->id,
31        'document_type_id' => 1,
32        'number' => 'SSSSSS55',
33        'created_by' => $authUser->id
34    ]);
35
36    $existingEmergencyContact = EmergencyContact::create([
37        'full_name' => 'ANA SUAREZ',
38        'relationship' => 'MADRE',
39        'cell_phone' => '911111111',
40        'telephone' => '44444444',
41        'email' => 'ANA@MAIL.COM',
42        'created_by' => $authUser->id
43    ]);
44
45    $existingContact = Contact::create([
46        'person_id' => $existingPerson->id,
47        'emergency_contact_id' => $existingEmergencyContact->id,
48        'cell_phone' => '999111111',
49        'telephone' => '6333333',
50        'email' => 'PEDRO@MAIL.COM',
51        'usuario_crea' => $authUser->id
52    ]);
53
54    $user = User::create([
55        'username' => 'SSSSSS555',
56        'password' => Hash::make('oldpass'),
57        'password_reset' => Hash::make('oldpass'),
58        'person_id' => $existingPerson->id,
59        'role_id' => 1,
60        'created_by' => $authUser->id
61    ]);
62
63    $data = [
64        'person' => [
65            'sexo_id' => 2,
66            'name' => 'Juan',
67            'paternal_surname' => 'Perez',
68            'maternal_surname' => 'Lopez',
69            'home_address' => 'Av Lima',
70        ],
71        'document' => [
72            'document_type_id' => 2,
73            'number' => '87654321',
74        ],
75        'emergency_contact' => [
76            'full_name' => 'Maria Lopez',
77            'relationship' => 'Hermana',
78            'cell_phone' => '98888888',
79            'telephone' => '0120876',
80            'email' => 'marial@mail.com',
81        ],
82        'contact' => [
83            'cell_phone' => '944444444',
84            'telephone' => '01777777',
85            'email' => 'Juan@mail.com',
86        ],
87        'role_id' => 1
88    ];
89
90    $updatedDocument = Document::create([
91        'person_id' => $existingPerson->id,
92        'document_type_id' => $data['document']['document_type_id'],
93        'number' => $data['document']['number'],
94        'created_by' => $authUser->id
95    ]);
96
97    $personService
98        ->shouldReceive('update')
99        ->once()
100        ->with($existingPerson->id, $data['person'])
101        ->andReturn($existingPerson);
102    $documentService
103        ->shouldReceive('update')
104        ->once()
105        ->with($existingDocument->id, $data['document'])
106        ->andReturn($updatedDocument);
107
108    $secService
109        ->shouldReceive('update')
110        ->once()
111        ->with($existingEmergencyContact->id, $data['emergency_contact'])
112        ->andReturnNull();
113
114    $contactService
115        ->shouldReceive('update')
116        ->once()
117        ->with($existingContact->id, $data['contact'])
118        ->andReturnNull();
119
120    $returnedUser = $service->update($user->id, $data);
121    $returnedUser = $returnedUser->fresh();
122    $this->assertInstanceOf(User::class, $returnedUser);
123    $this->assertEquals($existingPerson->id, $returnedUser->person_id);
124    $this->assertEquals($data['role_id'], $returnedUser->role_id);
125    $this->assertEquals($updatedDocument->number, $returnedUser->username);
126    $this->assertTrue(Hash::check($updatedDocument->number, $returnedUser->password_reset));
127    $this->assertEquals($authUser->id, $returnedUser->updated_by);
128    $this->assertDatabaseHas('users', [
129        'id' => $returnedUser->id,
130        'username' => $updatedDocument->number,
131        'role_id' => $data['role_id'],
132        'person_id' => $existingPerson->id,
133        'updated_by' => $authUser->id,
134    ]);
135 }

```

```

1 public function test_user_can_be_deleted()
2 {
3     $authUser = User::first();
4     Auth::shouldReceive('user')->andReturn($authUser);
5     $documentService = new DocumentService();
6     $ecService = new EmergencyContactService();
7     $contactService = new ContactService();
8     $personService = new PersonService($documentService, $ecService, $contactService);
9
10    $service = new UserService(
11        $personService,
12        $documentService,
13        $ecService,
14        $contactService
15    );
16
17    $person = Person::create([
18        'sexo_id' => 1,
19        'name' => 'CARLOS',
20        'paternal_surname' => 'RAMIREZ',
21        'maternal_surname' => 'ROJAS',
22        'home_address' => 'AV 1',
23        'created_by' => $authUser->id
24    ]);
25
26    $document = Document::create([
27        'person_id' => $person->id,
28        'document_type_id' => 1,
29        'number' => '12345678',
30        'created_by' => $authUser->id
31    ]);
32
33    $emergencyContact = EmergencyContact::create([
34        'full_name' => 'MARIA RAMIREZ',
35        'relationship' => 'MADRE',
36        'cell_phone' => '90011222',
37        'telephone' => '456789',
38        'email' => 'MARIA@MAIL.COM',
39        'created_by' => $authUser->id
40    ]);
41
42    $contact = Contact::create([
43        'person_id' => $person->id,
44        'emergency_contact_id' => $emergencyContact->id,
45        'cell_phone' => '999777666',
46        'telephone' => '456123',
47        'email' => 'CARLOS@MAIL.COM',
48        'usuario_crea' => $authUser->id
49    ]);
50
51    $user = User::create([
52        'username' => '12345678',
53        'password' => Hash::make('1234'),
54        'password_reset' => Hash::make('1234'),
55        'person_id' => $person->id,
56        'role_id' => 1,
57        'created_by' => $authUser->id
58    ]);
59
60    $response = $service->delete($user->id);
61
62    $this->assertNotNull($response);
63    $this->assertEquals(302, $response->getStatusCode());
64
65    $this->assertDatabaseHas('users', [
66        'id' => $user->id,
67        'state' => '0',
68        'updated_by' => $authUser->id,
69    ]);
70
71    $this->assertDatabaseHas('people', [
72        'id' => $person->id,
73        'state' => '0',
74    ]);
75
76    $this->assertDatabaseHas('documents', [
77        'id' => $document->id,
78        'state' => '0',
79    ]);
80
81    $this->assertDatabaseHas('contacts', [
82        'id' => $contact->id,
83        'state' => '0',
84    ]);
85
86    $this->assertDatabaseHas('emergency_contacts', [
87        'id' => $emergencyContact->id,
88        'state' => '0',
89    ]);
90    $this->assertEquals(
91        'Usuario eliminado exitosamente.',
92        session('message')
93    );
94 }

```

**Anexo G:** Código fuente de la función principal de la aplicación, crear operación (matricular y pago de pensiones).

```

1
2 public function store(array $data)
3 {
4     // Matricula
5     if ($data['operation_id'] === 1) {
6
7         // crea boleta
8         $enrollmentTypeId = $data['enrollment']['enrollment_type_id'];
9
10        $invoice = $this->invoiceService->store($data);
11
12        $discount_id = null;
13        $apply_discount_id = null;
14        $discount_applied = null;
15
16        if ($data['operation']['discount_id']) {
17            $discount_id = $data['operation']['discount_id'];
18            $apply_discount_id = $data['operation']['apply_discount_id'];
19            $discount_applied = $data['summary']['discount'];
20        }
21
22        // registrar operación
23        Operation::create([
24            'discount_id' => $discount_id ?? null,
25            'operation_type_id' => $data['operation_id'],
26            'payment_method_id' => $data['operation']['payment_method_id'],
27            'apply_discount_id' => $apply_discount_id ?? null,
28            'invoice_id' => $invoice->id,
29            'academic_cycle_id' => $data['enrollment']['academic_cycle_id'],
30            'user_id' => Auth::id(),
31            'quantity' => 1,
32            'unit_price' => $data['summary']['subtotal'],
33            'sub_total' => $data['summary']['subtotal'],
34            'discount_applied' => $discount_applied ?? 0,
35            'igv' => $data['summary']['igv'],
36            'total' => $data['summary']['total'],
37            'created_at' => now(),
38            'created_by' => Auth::id()
39        ]);
40
41        // crear persona
42        $person = $this->personService->create($data);
43
44        // crear alumno
45        $student = $this->studentService->store($person);
46
47        // registrar matricula
48        $enrollment = $this->enrollmentService->store($data, $student->id);
49
50        // crear las pensiones
51        $tuitionFee = $this->tuitionFeeService->store($enrollment->id, $data, $enrollmentTypeId);
52    }
53
54    // si es un pago de pensión
55
56 }
57

```

## Anexo H: Código fuente de la funcionalidad de pago de pensiones.

```

1
2 public function storePayTuitionFeeOperation(array $tuitionFeeIds, array $billingForm, int $invoiceId, int $userId)
3 {
4     $totals = $this->calculateTotals($tuitionFeeIds, $billingForm);
5
6     return Operation::create([
7         'operation_type_id' => 2,
8         'payment_method_id' => $billingForm['payment_method_id'],
9         'invoice_id' => $invoiceId,
10        'user_id' => $userId,
11        'quantity' => $totals['quantity'],
12        'unit_price' => $totals['quantity'] > 0 ? $totals['sub_total'] / $totals['quantity'] : 0,
13        'sub_total' => $totals['sub_total'],
14        'discount_applied' => $totals['discount_applied'],
15        'igv' => $totals['igv'],
16        'total' => $totals['total'],
17        'academic_cycle_id' => $billingForm['academic_cycle_id'] ?? null,
18        'created_by' => $userId,
19    ]);
20 }
21

```

```

1
2 public function calculateTotals(array $tuitionFeeIds, array $billingForm): array
3 {
4     $fees = app(TuitionFeeService::class)
5         ->findByIds($tuitionFeeIds);
6
7     $quantity = count($fees);
8     $subTotal = collect($fees)->sum('tuition_fee_amount');
9     $moraTotal = collect($fees)->sum(fn($f) => $f['late_fees'] ? collect($f['late_fees']->last()['total_amount'] : 0);
10
11    $discountApplied = $billingForm['discount'] ?? 0;
12    $igv = $billingForm['igv'] ?? 0;
13    $total = $subTotal + $moraTotal - $discountApplied + $igv;
14
15    return [
16        'quantity' => $quantity,
17        'sub_total' => $subTotal + $moraTotal,
18        'mora_total' => $moraTotal,
19        'discount_applied' => $discountApplied,
20        'igv' => $igv,
21        'total' => $total,
22    ];
23 }
24

```

```

1
2 public function storePayTuitionFeeOperation(array $tuitionFeeIds, array $billingForm, int $invoiceId, int $userId)
3 {
4     $totals = $this->calculateTotals($tuitionFeeIds, $billingForm);
5
6     return Operation::create([
7         'operation_type_id' => 2,
8         'payment_method_id' => $billingForm['payment_method_id'],
9         'invoice_id' => $invoiceId,
10        'user_id' => $userId,
11        'quantity' => $totals['quantity'],
12        'unit_price' => $totals['quantity'] > 0 ? $totals['sub_total'] / $totals['quantity'] : 0,
13        'sub_total' => $totals['sub_total'],
14        'discount_applied' => $totals['discount_applied'],
15        'igv' => $totals['igv'],
16        'total' => $totals['total'],
17        'academic_cycle_id' => $billingForm['academic_cycle_id'] ?? null,
18        'created_by' => $userId,
19    ]);
20 }
21

```

## Anexo I: Código fuente de la funcionalidad para generar moras diarias para pensiones vencidas.

```

1 public function handle()
2 {
3   $today = Carbon::today();
4   $count = 0;
5
6   $fees = TuitionFee::with(['enrollment.academicCycle.financialConfigs', 'lateFees'])
7     →where('due_date', '<', $today)
8     →get();
9
10  foreach ($fees as $fee) {
11    $financialConfig = $fee→enrollment→academicCycle
12      →financialConfigs()
13      →where('state', '1')
14      →first();
15
16    if (!$financialConfig) {
17      $this→warn("No hay configuración financiera para ciclo {$fee→enrollment→academic_cycle_id}");
18      continue;
19    }
20
21    $lateFeePerDay = $financialConfig→late_fee;
22
23    $dueDate = Carbon::parse($fee→due_date);
24    $daysLate = $dueDate→diffInDays($today);
25
26    $existingDays = $fee→lateFees()→count();
27
28    if ($daysLate > $existingDays) {
29      $missingDays = $daysLate - $existingDays;
30
31      for ($i = 1; $i ≤ $missingDays; $i++) {
32        $dayNumber = $existingDays + $i;
33        $lateDate = $dueDate→copy()→addDays($dayNumber);
34
35        LateFee::create([
36          'tuition_fee_id' ⇒ $fee→id,
37          'late_fee_amount' ⇒ $lateFeePerDay,
38          'number_days' ⇒ $dayNumber,
39          'total_amount' ⇒ $lateFeePerDay * $dayNumber,
40          'state' ⇒ '1',
41          'created_by' ⇒ Auth::id() ?? 1,
42          'created_at' ⇒ $lateDate→endOfDay(),
43        ]);
44
45        $count++;
46      }
47    }
48  }
49
50  $this→info("Se generaron {$count} registros de mora.");
51 }
52

```

## Anexo J: Guía de entrevista.

**Proyecto de investigación:** Aplicación web para la gestión de matrícula de estudiantes en la Institución Educativa Nuestra Señora de Guadalupe, 2025.

**Instrumento:** Guía de entrevista

**Nombre:**

**Cargo:**

1. ¿Cómo se realiza actualmente el proceso de matrícula de estudiantes en la institución?
2. ¿Qué áreas o personas intervienen en este proceso?
3. ¿En qué periodos del año se realiza la matrícula?
4. ¿Qué tipo de información se solicita a los estudiantes o padres de familia durante la matrícula?
5. ¿Cómo se registra actualmente la información?
6. ¿Cómo se almacena y resguarda la información de los estudiantes?
7. ¿Cómo se consulta o recupera la información cuando es necesaria?
8. ¿Qué dificultades se presentan durante el proceso de matrícula?
9. ¿Se ha presentado pérdida o inconsistencia de información?
10. ¿Quiénes tienen acceso a la información de matrícula?
11. ¿Existen niveles de acceso o roles definidos?
12. ¿Cómo se garantiza la seguridad de la información?
13. ¿Se realizan respaldos de la información? ¿Con qué frecuencia?

**Anexo K: Cuestionario de validación de funcionalidad de la aplicación.**

Nombre:

Cargo:

1. ¿El sistema permite la autenticación de usuarios?

Si  No 

2. ¿El sistema permite crear roles?

Si  No 

3. ¿El sistema permite editar un rol?

Si  No 

4. ¿El sistema permite eliminar un rol?

Si  No 

5. ¿El sistema permite crear un usuario?

Si  No 

6. ¿El sistema permite editar un usuario?

Si  No 

7. ¿El sistema permite eliminar un usuario?

Si  No 

8. ¿El sistema permite crear un ciclo académico?

Si  No 

9. ¿El sistema permite editar un ciclo académico?

Si  No 

10. ¿El sistema permite eliminar un ciclo académico?

Si  No 

11. ¿El sistema permite crear una sección?

Si  No 

12. ¿El sistema permite editar una sección?

Si  No 

13. ¿El sistema permite eliminar una sección?

Si  No 

14. ¿El sistema permite crear un nivel?

Si  No 

15. ¿El sistema permite editar un nivel?

Si  No

16. ¿El sistema permite eliminar un nivel?

Si  No

17. ¿El sistema permite crear un grado?

Si  No

18. ¿El sistema permite editar un grado?

Si  No

19. ¿El sistema permite eliminar un grado?

Si  No

20. ¿El sistema permite crear un tipo de matrícula?

Si  No

21. ¿El sistema permite editar un tipo de matrícula?

Si  No

22. ¿El sistema permite eliminar un tipo de matrícula?

Si  No

23. ¿El sistema permite crear un descuento?

Si  No

24. ¿El sistema permite editar un descuento?

Si  No

25. ¿El sistema permite eliminar un descuento?

Si  No

26. ¿El sistema permite crear un tipo de operación?

Si  No

27. ¿El sistema permite editar un tipo de operación?

Si  No

28. ¿El sistema permite eliminar un tipo de operación?

Si  No

29. ¿El sistema permite crear un método de pago?

Si  No

30. ¿El sistema permite editar un método de pago?

Si  No

31. ¿El sistema permite eliminar un método de pago?

Si  No

32. ¿El sistema permite registrar una matrícula?

Si  No

33. ¿El sistema permite editar una matrícula?

Si  No

34. ¿El sistema permite pagar una o más pensiones?

Si  No

35. ¿El sistema permite anular una operación (matrícula o pago de pensión)?

Si  No

36. ¿El sistema permite el cálculo automático de moras a pensiones vencidas?

Si  No

**Anexo L: Solicitud de autorización para la realización de tesis.**



Fecha: .....  
 Hora : .....  
 Firma: .....

SOLICITO: Autorización para realizar tesis

SEÑOR DIRECTOR DE LA INSTITUCIÓN EDUCATIVA NUESTRA SEÑORA DE GUADALUPE


Victor Alberto López Amiquero  
 identificado con DNI: 70251967,  
 Bachiller en Ingeniería de Sistemas de  
 la Universidad Nacional San Cristóbal  
 de Huamanga, me dirijo a usted con el  
 debido respeto y expongo:

Que, encontrándome en la etapa de investigación y sumada la aprobación del proyecto de tesis “**APLICACIÓN WEB PARA LA GESTIÓN DE MATRÍCULA DE ESTUDIANTES EN LA INSTITUCIÓN EDUCATIVA NUESTRA SEÑORA DE GUADALUPE, 2025**”, solicito se me autorice el desarrollo de mi tesis en su institución.

El desarrollo de esta investigación se realizará respetando la confidencialidad de la información institucional. Asimismo, los resultados obtenidos serán compartidos con la institución para su posible implementación y mejora de sus procesos administrativos.

Agradezco de antemano la atención brindada y quedo atento a su respuesta adjunto la resolución decanal de aprobación de mi proyecto de tesis.

Ayacucho, 23 de mayo de 2025

  
 VÍCTOR ALBERTO LÓPEZ AMIQUERO

DNI: 70251967

Contacto:  
 Celular 934848514

**Anexo M: Carta de aceptación.**

"Año de la recuperación y consolidación de la economía peruana"

Ayacucho, 27 mayo de 2025

**CARTA MÚLTIPLE N° 1****Señor:****VÍCTOR ALBERTO LÓPEZ AMIQUERO****Asunto:** Se otorga permiso para realizar tesis**Referencia:** Solicitud N° 01

Por medio del presente me dirijo a usted previo saludo a fin de informarle en respuesta a su solicitud de autorización para realizar su tesis titulado **"APLICACIÓN WEB PARA LA GESTIÓN DE MATRÍCULA DE ESTUDIANTES EN LA INSTITUCIÓN EDUCATIVA NUESTRA SEÑORA DE GUADALUPE, 2025"**.

En este sentido, se concede la autorización para realizar lo necesario y facilidades correspondientes para su tesis.

Atentamente;



*[Firma manuscrita]*  
DNI 28310665

**Anexo N: Acta de aceptación de las funcionalidades de la aplicación web.****ACTA DE ACEPTACIÓN DE LA APLICACIÓN WEB**

En la ciudad de Ayacucho, a los 08 días del mes de setiembre del año 2025, se deja constancia que el tesista VÍCTOR ALBERTO LÓPEZ AMIQUERO, Bachiller en Ingeniería de Sistemas, ha desarrollado la Aplicación Web para la Gestión de Matrícula de Estudiantes en la Institución Educativa Nuestra Señora de Guadalupe, como parte del trabajo de tesis.

El presente software fue desarrollado siguiendo la metodología ágil Scrum, considerando los módulos de: Seguridad, configuración, operación y Académico

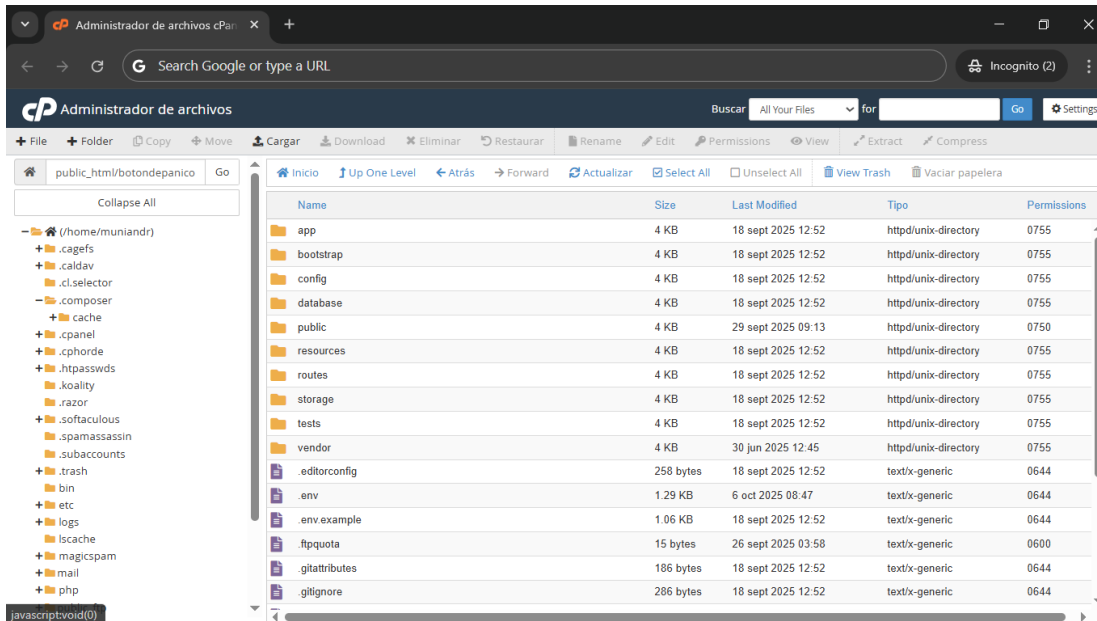
Tras la revisión del funcionamiento general y la verificación de los requerimientos definidos, se acepta la aplicación web como producto final del desarrollo, evidenciando el logro de los objetivos planteados en la investigación.

Firman en señal de conformidad:

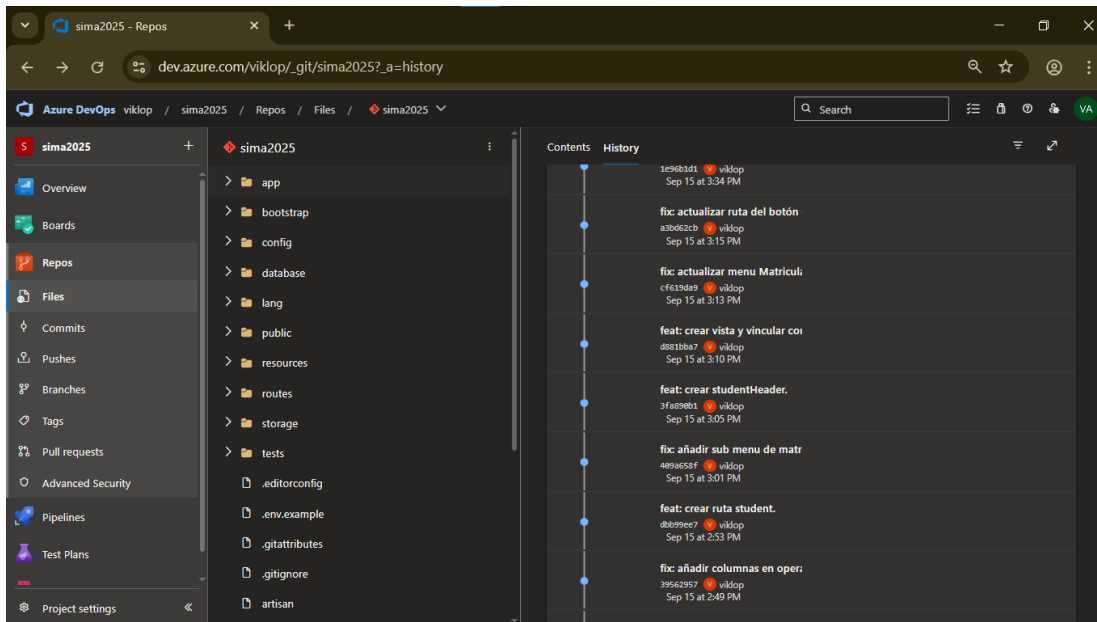


ANALI GARAY DELGADILLO  
DNI N° 71246802  
SECRETARIA

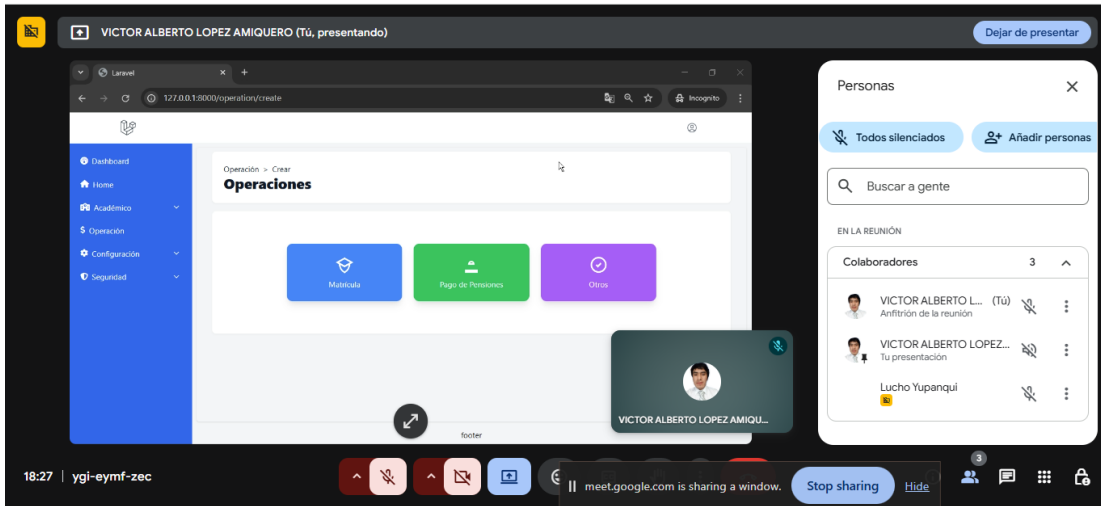
## Anexo O: Despliegue de la aplicación en un servidor de pruebas.



## Anexo P: Historial de commits durante el desarrollo de la aplicación.



### Anexo Q: Sprint review del sprint 3.





**UNSCH**

FACULTAD DE  
**INGENIERÍA**  
DE MINAS, GEOLOGÍA Y CIVIL

**ACTA DE SUSTENTACIÓN DE TESIS N° 04-2026-FIMGC**

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS**

En la Universidad Nacional de San Cristóbal de Huamanga, en la ciudad de Ayacucho, en cumplimiento a la **Resolución Decanal No 024-2026-FIMGC-D**, a los **veintitrés días del mes de marzo de 2026**, siendo las **10:00 a.m.**, reunidos en el **Auditorio de la Escuela Profesional de Ingeniería de Minas**, bajo la presidencia del **MSc. Ing. José Ernesto ESTRADA CÁRDENAS**, y los miembros: **Ing. José Elías YAURI VIDALON**, **Ing. Richard ZAPATA CASAVARDE** y **Ing. Lissette Elvira FERNANDEZ JERI**, actuando como secretario docente el **Ing. Saul Walter RETAMOZO FERNANDEZ**, para proceder a la sustentación de tesis para optar el **Título Profesional de Ingeniero de Sistemas**, del bachiller en Ingeniería de Sistemas:

**Victor Alberto LOPEZ AMIQUERO**

Quien presentó la tesis denominada:

**Aplicación web para la gestión de matrícula de estudiantes en la Institución Educativa Nuestra Señora de Guadalupe, 2025**

Los señores miembros del jurado luego de expuesta la tesis y absueltas las preguntas, deliberaron y declararon:

**Aprobado con 16 (dieciséis)**

Siendo las **11:45 a.m.** del día **23 de marzo del 2026**, culmina el acto de sustentación de tesis, y en conformidad de lo actuado los miembros del jurado firmamos al pie del presente.

MSc. Ing. José Ernesto ESTRADA CÁRDENAS  
Presidente

Ing. José Elías YAURI VIDALON  
Miembro

Ing. Lissette Elvira FERNANDEZ JERI  
Miembro

Ing. Richard ZAPATA CASAVARDE  
Miembro - Asesor

Ing. Saul Walter RETAMOZO FERNANDEZ  
Secretario docente de la FIMGC



## CONSTANCIA DE ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN

### CONSTANCIA N° 035-2026-KPS-FIMGC/UNSCH

El que suscribe; responsable verificador de originalidad de trabajos de tesis de pregrado con el software Turnitin, en segunda instancia para las **Escuelas Profesionales** de la **Facultad de Ingeniería de Minas, Geología y Civil**; en cumplimiento a la **Resolución de Consejo Universitario N° 039-2021-UNSCH-CU**, Reglamento de Originalidad de Trabajos de Investigación de la Universidad Nacional San Cristóbal de Huamanga y **Resolución Decanal N° 697-2024-FIMGC-D**, deja constancia de originalidad de trabajo de investigación, que el/la Sr./Srta.

**Nombres y Apellidos** : Victor Alberto Lopez Amiquero  
**Escuela Profesional** : INGENIERÍA DE SISTEMAS  
**Título de la Tesis** : Aplicación web para la gestión de matrícula de estudiantes en la Institución Educativa Nuestra Señora de Guadalupe, 2025  
**Evaluación de la Originalidad** : 9% Índice de Similitud  
**Identificador de la entrega** : 2945667063

Por tanto, según los Artículos 12, 13 y 17 del Reglamento de Originalidad de Trabajos de Investigación, es **PROCEDENTE** otorgar la **Constancia de Originalidad** para los fines que crea conveniente.

En señal de conformidad y verificación se firma la presente constancia

Ayacucho, 30 de abril de 2026



Firmado digitalmente por:  
PERALTA SOTOMAYOR Karel  
FAU 20143680754 soft  
Motivo: Soy el autor del documento  
Fecha: 30/04/2026 03:00:07-0500

# Aplicación web para la gestión de matrícula de estudiantes en la Institución Educativa Nuestra Señora de Guadalupe, 2025

*por* Victor Alberto LOPEZ AMIQUERO

---

**Fecha de entrega:** 27-abr-2026 12:21p. m. (UTC-0500)

**Identificador de la entrega:** 2945667063

**Nombre del archivo:** MEMORANDO\_Nº\_178-2026-FIMGC-UNSCH-CERTIFICADO\_DE\_ORIGINALIDAD-VICTOR\_ALBERTO\_LOPEZ\_AMIQUERO.pdf (12.12M)

**Total de palabras:** 35787

**Total de caracteres:** 214302

# Aplicación web para la gestión de matrícula de estudiantes en la Institución Educativa Nuestra Señora de Guadalupe, 2025

## INFORME DE ORIGINALIDAD

9%

INDICE DE SIMILITUD

9%

FUENTES DE INTERNET

4%

PUBLICACIONES

5%

TRABAJOS DEL ESTUDIANTE

## FUENTES PRIMARIAS

1	<a href="https://hdl.handle.net">hdl.handle.net</a> Fuente de Internet	1%
2	Submitted to Universidad Nacional de San Cristóbal de Huamanga Trabajo del estudiante	1%
3	<a href="https://repositorio.ucv.edu.pe">repositorio.ucv.edu.pe</a> Fuente de Internet	1%
4	<a href="https://repositorio.unsch.edu.pe">repositorio.unsch.edu.pe</a> Fuente de Internet	1%
5	<a href="https://repositorio.uladech.edu.pe">repositorio.uladech.edu.pe</a> Fuente de Internet	<1%
6	<a href="https://repositorio.espe.edu.ec">repositorio.espe.edu.ec</a> Fuente de Internet	<1%
7	Submitted to Universidad Tecnológica de los Andes Trabajo del estudiante	<1%
8	<a href="https://bibdigital.epn.edu.ec">bibdigital.epn.edu.ec</a> Fuente de Internet	<1%

9	<a href="http://repositorio.unp.edu.pe">repositorio.unp.edu.pe</a> Fuente de Internet	<1 %
10	<a href="http://repositorio.unamba.edu.pe">repositorio.unamba.edu.pe</a> Fuente de Internet	<1 %
11	<a href="http://ribuni.uni.edu.ni">ribuni.uni.edu.ni</a> Fuente de Internet	<1 %
12	<a href="http://apirepositorio.unu.edu.pe">apirepositorio.unu.edu.pe</a> Fuente de Internet	<1 %
13	<a href="http://ebufete.es">ebufete.es</a> Fuente de Internet	<1 %
14	<a href="http://repositorio.umariana.edu.co">repositorio.umariana.edu.co</a> Fuente de Internet	<1 %
15	Submitted to Universidad Nacional Federico Villarreal Trabajo del estudiante	<1 %
16	Submitted to Universidad Catolica De Cuenca Trabajo del estudiante	<1 %
17	<a href="http://elblogpython.com">elblogpython.com</a> Fuente de Internet	<1 %
18	<a href="http://d.documentop.com">d.documentop.com</a> Fuente de Internet	<1 %
19	<a href="http://repositorio.unicesar.edu.co">repositorio.unicesar.edu.co</a> Fuente de Internet	<1 %

Submitted to Universidad Pública de Navarra

20

Trabajo del estudiante

&lt;1 %

21

Submitted to Universidad Carlos III de Madrid

Trabajo del estudiante

&lt;1 %

22

repositorio.unan.edu.ni

Fuente de Internet

&lt;1 %

23

repositorio.utc.edu.ec

Fuente de Internet

&lt;1 %

24

Submitted to Universidad Cesar Vallejo

Trabajo del estudiante

&lt;1 %

25

repositorio.uancv.edu.pe

Fuente de Internet

&lt;1 %

26

scrummanager.net

Fuente de Internet

&lt;1 %

27

Submitted to Universidad Internacional Isabel I de Castilla

Trabajo del estudiante

&lt;1 %

28

emmanuelagmf.com

Fuente de Internet

&lt;1 %

29

repositorio.ulead.edu.ec

Fuente de Internet

&lt;1 %

30

repositorio.unasam.edu.pe

Fuente de Internet

&lt;1 %

31

Submitted to Universidad Privada del Norte

Trabajo del estudiante

<1 %

32

Submitted to Corporación Universitaria  
Iberoamericana

Trabajo del estudiante

<1 %

33

Submitted to Universidad Nacional del Centro  
del Peru

Trabajo del estudiante

<1 %

34

Submitted to Universidad Peruana de  
Ciencias Aplicadas

Trabajo del estudiante

<1 %

35

Submitted to Universidad del Istmo de  
Panamá

Trabajo del estudiante

<1 %

36

[repositorioacademico.upc.edu.pe](http://repositorioacademico.upc.edu.pe)

Fuente de Internet

<1 %

Excluir citas

Activo

Excluir coincidencias < 30 words

Excluir bibliografía

Activo