UNIVERSIDAD NACIONAL DE CAJAMARCA
FACULTAD DE INGENIERÍA

ESCUELA PROFESIONAL
DE INGENIERÍA DE SISTEMAS

“IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024”

TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS

AUTOR

Bach. JHAN CARLOS VILLANUEVA VARGAS

ASESOR

MANUEL ENRIQUE MALPICA RODRÍGUEZ

Ingeniero de Sistemas

CAJAMARCA – PERÚ

SEPTIEMBRE 2025

CONSTANCIA DE INFORME DE ORIGINALIDAD

- FACULTAD DE INGENIERÍA –

1. **Investigador:** Jhan Carlos Villanueva Vargas
   DNI: 73414616
   Escuela Profesional: Ingeniería de Sistemas

2. **Asesor:** Ing. Manuel Enrique Malpica Rodríguez
   Facultad: Ingeniería

3. **Grado académico o título profesional**

   ☐Bachiller ■Título profesional ☐Segunda especialidad

   ☐Maestro ☐Doctor

4. **Tipo de Investigación:**

   ■ Tesis ☐ Trabajo de investigación ☐ Trabajo de suficiencia profesional

   ☐ Trabajo académico

5. **Título de Trabajo de Investigación:**

   Impacto de un sistema de información web en la gestión académica de la Escuela Superior de Formación Artística pública Mario Urteaga Alvarado, Cajamarca 2024

6. **Fecha de evaluación:** 24 de setiembre de 2025
7. **Software antiplagio:** ■ **TURNITIN** ☐ **URKUND (OURIGINAL)** ***(*)***
8. **Porcentaje de Informe de Similitud:** 17%
9. **Código Documento: 503092228**
10. **Resultado de la Evaluación de Similitud:**

    ■ **APROBADO** ☐ **PARA LEVANTAMIENTO DE OBSERVACIONES O DESAPROBADO**

Fecha Emisión: 25 de setiembre de 2025

FIRMA DIGITAL

Firmado digitalmente por: BAZAN DIAZ Laura Sofia FAU 20148258601 soft
Motivo: En señal de conformidad
Fecha: 25/09/2025 22:51:26-0500

FIRMA DEL ASESOR
Ing. Manuel Enrique Malpica Rodríguez
DNI: 26707158

UNIDAD DE INVESTIGACIÓN FI

ACTA DE SUSTENTACIÓN PÚBLICA DE TESIS.

TITULO : *IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024*

ASESOR : *Dr. Ing. Manuel Enrique Malpica Rodríguez.*

En la ciudad de Cajamarca, dando cumplimiento a lo dispuesto por el Oficio Múltiple Nº 0659-2025-PUB-SA-FI-UNC, de fecha 06 de octubre de 2025, de la Secretaría Académica de la Facultad de Ingeniería, a los ***nueve días del mes de octubre de 2025***, siendo las doce horas (12:00 p.m.) en la Sala de Audiovisuales (Edificio 4K) de la Escuela Profesional de Ingeniería de Sistemas, de la Facultad de Ingeniería se reunieron los Señores Miembros del Jurado Evaluador:

Presidente : Dr. Ing. Oscar Gilberto Zocón Alva.
Vocal : Dr. Ing. Edwin Alberto Valencia Castillo.
Secretaria : Dra. Ing. Karim Ivette Cruzado Villar.

Para proceder a escuchar y evaluar la sustentación pública de la tesis titulada: *IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024*, presentado por el Bachiller en Ingeniería de Sistemas ***JHAN CARLOS VILLANUEVA VARGAS***, asesorado por el Dr. Ing. Manuel Enrique Malpica Rodríguez, para la obtención del Título Profesional

Los Señores Miembros del Jurado replicaron al sustentante debatieron entre sí en forma libre y reservada y lo evaluaron de la siguiente manera:

EVALUACIÓN PRIVADA : ..7.. PTS.
EVALUACIÓN PÚBLICA : ..10.. PTS.
EVALUACIÓN FINAL : ..17.. PTS ..Diecisiete.. (En letras)

En consecuencia, se lo declara ..Aprobado.. con el calificativo de ..17.. acto seguido, el presidente del jurado hizo saber el resultado de la sustentación, levantándose la presente a las ..13:00.. horas del mismo día, con lo cual se dio por terminado el acto, para constancia se firmó por quintuplicado.

Dr. Ing. Oscar Gilberto Zocón Alva.
Presidente

Dr. Ing. Edwin Alberto Valencia Castillo.
Vocal

Dra. Ing. Karim Ivette Cruzado Villar.
Secretario

Dr. Ing. Manuel Enrique Malpica Rodríguez.
Asesor

COPYRIGHT © 2025
JHAN CARLOS VILLANUEVA VARGAS
Todos los Derechos Reservados ®

AGRADECIMIENTO

A Dios, por darme la fortaleza y la oportunidad de culminar esta importante etapa de mi vida, y por permitirme conocer a personas excepcionales en el camino.

Al Ing. Manuel Malpica Rodríguez, mi asesor, por su valiosa orientación y constante apoyo, así como a todos los docentes que, con sus enseñanzas, contribuyeron a mi formación profesional.

A mis amigos y compañeros, con quienes compartí gratos momentos durante esta etapa. En especial, a Erick y a Jamil, por su apoyo brindado durante el desarrollo de esta investigación.

DEDICATORIA

A mi familia, en especial a mis padres, César y Miriam, por acompañarme en cada etapa de este proceso con su guía, sus sabios consejos y su apoyo incondicional.

A mis hermanos, Alex y Sofía, cuya presencia y ejemplo han sido una constante fuente de inspiración y motivación para superarme día a día.

A mis amigos, por su respaldo y amistad genuina, que han sido fundamentales a lo largo de este camino.

CONTENIDO

CAPÍTULO I. INTRODUCCIÓN ---------- 1

CAPÍTULO II. MARCO TEÓRICO ---------- 4

2.1 Antecedentes teóricos ---------- 4

2.1.1 Antecedentes internacionales ---------- 4

2.1.2 Antecedentes nacionales ---------- 5

2.1.3 Antecedentes locales ---------- 5

2.2 Bases teóricas ---------- 6

2.2.1 Gestión académica ---------- 6

2.2.2 Sistema de información web ---------- 8

2.3 Definición de términos básicos ---------- 14

CAPÍTULO III. MATERIALES Y MÉTODOS ---------- 16

3.1 Procedimiento ---------- 16

3.1.1 Inicio del proyecto ---------- 19

3.1.2 Configuración del tablero Kanban ---------- 22

3.1.3 Aplicación del tablero Kanban ---------- 27

3.1.4 Seguimiento y mejora continua ---------- 98

3.2 Tratamiento, análisis de datos y presentación de resultados ---------- 102

3.2.1 Tratamiento y análisis de datos ---------- 102

3.2.2 Presentación de resultados ---------- 107

CAPÍTULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS ---------- 119

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES ---------- 121

5.1. Conclusiones ---------- 121

5.2. Recomendaciones ---------- 122

REFERENCIAS BIBLIOGRÁFICAS ---------- 123

ANEXOS ---------- 130

Anexo 1: Matriz de operacionalización de variables ---------- 130

Anexo 2: Matriz de cosistencia ---------- 131

Anexo 3: Cuestionario – variable independiente ---------- 132

Anexo 4: Ficha de observación de la dimensión tiempo ---------- 134

Anexo 5: Ficha de análisis documental de la dimensión fiabilidad ---------- 134

Anexo 6: Ficha de cotejo de la dimensión transparencia ---------- 135

Anexo 7: Validación de instrumentos ---------- 136

Anexo 8: Cálculo de la confiabilidad del cuestionario para medir la calidad del software 140

ÍNDICE DE TABLAS

Tabla I Equipo de trabajo 23

Tabla II. Product backlog de requerimientos funcionales 28

Tabla III Product backlog de requerimientos no funcionales 30

Tabla IV Comparativa técnica entre bases de datos 34

Tabla V HU02 “Gestionar estudiantes” 35

Tabla VI Prueba de caja negra “Gestionar estudiantes” 40

Tabla VII HU3 “Gestionar docentes” 41

Tabla VIII Prueba de caja negra “Gestionar docentes” 43

Tabla IX HU04 “Gestionar periodos” 44

Tabla X Prueba de caja negra “Gestionar periodos” 45

Tabla XI HU07 “Gestionar jefes de departamento” 46

Tabla XII Prueba de caja negra “Gestionar jefes de departamento” 47

Tabla XIII HU08 “Gestionar cursos” 48

Tabla XIV Prueba de caja negra “Gestionar cursos” 50

Tabla XV HU10 “Gestionar cursos a calificar” 51

Tabla XVI Prueba de caja negra “Gestionar cursos a calificar” 53

Tabla XVII HU16 “Gestionar pagos” 54

Tabla XVIII Prueba de caja negra “Gestionar pagos" 56

Tabla XIX HU18 “Generar matrícula” 57

Tabla XX Prueba de caja negra “Generar matrícula” 58

Tabla XXI HU05 “Gestionar matrículas” 60

Tabla XXII Prueba de caja negra “Gestionar matrícula” 61

Tabla XXIII HU01. Iniciar sesión 62

Tabla XXIV Prueba de caja negra “Iniciar sesión” 65

Tabla XXV HU09 “Gestionar usuarios del sistema” 66

Tabla XXVI Prueba de caja negra “Gestionar usuarios del sistema” 67

Tabla XXVII HU11 “Gestionar sesiones de clases” 68

Tabla XXVIII Prueba de caja negra “Gestionar sesiones de clases” 70

Tabla XXIX HU12 “Gestionar desempeño de estudiantes” 71

Tabla XXX Prueba de caja negra “Gestionar desempeño de estudiantes” 73

Tabla XXXI HU13 “Gestionar actividades por cada sesión de clase” 74

Tabla XXXII Prueba de caja negra “Gestionar actividades por cada sesión de clase” 75

Tabla XXXIII HU18 “Visualizar sesiones de clase, porcentaje de asistencia y calificaciones por cada curso” 76

Tabla XXXIV Prueba de caja negra “Detalles por curso del estudiante” 78

Tabla XXXV HU19 “Visualizar y desarrollar actividades” 79

Tabla XXXVI Prueba de caja negra “Visualizar y desarrollar actividades” 80

Tabla XXXVII HU22 “Visualizar pagos” 81

Tabla XXXVIII Prueba de caja negra “Visualizar pagos” 82

Tabla XXXIX HU06 “Filtrar y visualizar notas de cada estudiante” 83

Tabla XL Prueba de caja negra “Filtrar y visualizar notas de cada estudiante” 84

Tabla XLI HU14 “Generar acta de notas por cada curso” 84

Tabla XLII Prueba de caja negra “Generar acta de notas por cada curso” 86

Tabla XLIII HU15 “Visualizar horario de clases (docente)” 87

Tabla XLIV Prueba de caja negra “Visualizar horario de clases (docente)” 88

Tabla XLV HU20 “Visualizar horario de clases (estudiante)” 89

Tabla XLVI Prueba de caja negra “Visualizar horario de clases (estudiante)” 90

Tabla XLVII HU21 “Visualizar historial de notas” 90

Tabla XLVIII Prueba de caja negra “Visualizar historial de notas” 91

Tabla XLIX Throughput semanal 99

Tabla L Resumen de resultados del cuestionario referente a la calidad de software 102

Tabla LI Número de ciclos de observación según el método General Electric 103

Tabla LII Tiempos promedio pre y post test en los procesos pertenecientes a la dimensión tiempo 104

Tabla LIII Número de casos inconsistentes referentes a la dimensión fiabilidad 105

Tabla LIV Número de días de retraso en ejecución de actividades referentes a la dimensión transparencia 106

Tabla LV Porcentaje de mejora en los procesos relacionados a la dimensión tiempo 111

ÍNDICE DE FIGURAS

Fig. 1 Representación de la metodología Kanban 13

Fig. 2 Fases y pasos definidos para la implementación de la metodología Kanban 19

Fig. 3 Diagrama del proceso “Gestión de matrícula” 20

Fig. 4 Diagrama del proceso “Gestión de calificaciones” 20

Fig. 5 WIP en el tablero Kanban 24

Fig. 6 Tablero de trabajo 26

Fig. 7 Atributos de una historia de usuario 27

Fig. 8 Incorporación de tarjetas en el tablero Kanban 31

Fig. 9 Leyenda de etiquetas en el tablero Kanban 31

Fig. 10 Arquitectura backend 33

Fig. 11 Prototipo de la interfaz “Gestionar estudiantes” 36

Fig. 12 Diagrama de clases “Gestionar estudiantes” 36

Fig. 13 Clase del modelo “estudiante” 37

Fig. 14 Clase del modelo “persona” 37

Fig. 15 Clase del modelo “carreraProfesional” 38

Fig. 16 Clase del controlador “estudiante” 38

Fig. 17 Clase del controlador “persona” 39

Fig. 18 Código de la vista “Gestión de estudiantes” 39

Fig. 19 Vista “Gestionar estudiantes” 40

Fig. 20 Formulario de datos “Gestionar estudiantes” 40

Fig. 21 Prototipo de la interfaz “Gestionar docentes” 42

Fig. 22 Diagrama de clases “Gestionar docentes” 42

Fig. 23 Vista “Gestionar docentes” 42

Fig. 24 Formulario de datos “Gestionar docentes” 43

Fig. 25 Prototipo de la interfaz “Gestionar periodos” 44

Fig. 26 Diagrama de clases “Gestionar periodos” 44

Fig. 27 Vista “Gestionar periodos” 45

Fig. 28 Prototipo de la interfaz “Gestionar jefes de departamento” 46

Fig. 29 Diagrama de clases “Gestionar jefes de departamento” 46

Fig. 30 Vista “Gestionar jefes de departamento” 47

Fig. 31 Formulario de datos “Gestionar jefes de departamento” 47

Fig. 32 Prototipo de la interfaz “Gestionar cursos” 49

Fig. 33 Diagrama de clases “Gestionar cursos” 49

Fig. 34 Vista “Gestionar cursos” 50

Fig. 35 Formulario de datos “Gestionar cursos” 50

Fig. 36 Prototipo de la interfaz “Gestionar cursos a calificar” 52

Fig. 37 Diagrama de clases “Gestionar cursos a calificar” 52

Fig. 38 Vista “Gestionar cursos a calificar” 53

Fig. 39 Prototipo de la interfaz “Gestionar pagos" 55

Fig. 40 Diagrama de clases “Gestionar pagos" 55

Fig. 41 Vista “Gestionar pagos" 56

Fig. 42 Prototipo de la interfaz “Generar matrícula” 57

Fig. 43 Diagrama de clases “Generar matrícula” 58

Fig. 44 Vista “Generar matrícula” 58

Fig. 45 Diagrama de clases “Gestionar matrícula” 60

Fig. 46 Vista “Gestionar matrícula” 61

Fig. 47 Diagrama de clases “Iniciar sesión” 63

Fig. 48 Arquitectura de seguridad de acceso 63

Fig. 49 Vista “Iniciar sesión” 65

Fig. 50 Diagrama de clases “Gestionar usuarios del sistema” 67

Fig. 51 Vista “Gestionar usuarios del sistema” 67

Fig. 52 Diagrama de clases “Gestionar sesiones de clases” 69

Fig. 53 Vista cursos activos por docente 69

Fig. 54 Vista “Gestionar sesiones de clases” 70

Fig. 55 Diagrama de clases “Gestionar desempeño de estudiantes” 72

Fig. 56 Vista “Gestionar desempeño de estudiantes” 72

Fig. 57 Vista “Gestionar desempeño de estudiantes” toma de asistencia 72

Fig. 58 Diagrama de clases “Gestionar actividades por cada sesión de clase” 75

Fig. 59 Vista “Gestionar actividades por cada sesión de clase” 75

Fig. 60 Vista “Lista de cursos del estudiante” 77

Fig. 61 Vista “Sesiones de clase del estudiante por curso” 77

Fig. 62 Vista “Calificaciones del estudiante por curso” 78

Fig. 63 Vista “Visualizar y desarrollar actividades” 80

Fig. 64 Vista “Visualizar pagos” 82

Fig. 65 Vista “Filtrar y visualizar notas de cada estudiante” 83

Fig. 66 Diagrama de clases “Generar acta de notas por cada curso” 85

Fig. 67 Vista “Generar acta de notas por cada curso” 86

Fig. 68 Vista “Visualizar horario de clases (docente)” 88

Fig. 69 Vista “Visualizar horario de clases (estudiante)” 89

Fig. 70 Vista “Visualizar historial de notas” 91

Fig. 71 Diagrama de despliegue (arquitectura) 92

Fig. 72 Diagrama de clases general 95

Fig. 73 Diagrama de base de datos 96

Fig. 74 Diagrama de componentes 97

Fig. 75 Diagrama final del proceso “Gestión de matrícula” 97

Fig. 76 Diagrama final del proceso “Gestión de calificaciones” 98

Fig. 77 Gráfico circular-calidad de software-adecuación funcional 107

Fig. 78 Gráfico circular-calidad de software-eficiencia de desempeño 108

Fig. 79 Gráfico circular-calidad de software-usabilidad 108

Fig. 80 Gráfico circular-calidad de software-fiabilidad 109

Fig. 81 Gráfico circular-calidad de software-seguridad 109

Fig. 82 Gráfico circular-calidad de software-mantenibilidad 110

Fig. 83 Gráfico circular-calidad de software-portabilidad 110

Fig. 84 Registro de tiempos en procesos de la gestión académica 111

Fig. 85 Prueba de normalidad de datos relacionados a la dimensión tiempo 112

Fig. 86 Prueba de rangos con signo de Wilcoxon para muestras relacionadas P1 113

Fig. 87 Prueba de rangos con signo de Wilcoxon para muestras relacionadas P2 113

Fig. 88 Prueba T de Student para muestras relacionadas P3 y P4 114

Fig. 89 Número de casos que representan falta de fiabilidad dentro de la gestión académica 114

Fig. 90 Prueba de normalidad de datos relacionados a la dimensión fiabilidad 115

Fig. 91 Prueba de rangos con signo de Wilcoxon para muestras relacionadas - d_ fiabilidad 116

Fig. 92 Número de días de retraso que representan falta de trasparencia dentro de la gestión académica. 116

Fig. 93 Prueba de normalidad de datos relacionados a la dimensión transparencia 117

Fig. 94 Prueba de rangos con signo de Wilcoxon para muestras relacionadas I1 117

Fig. 95 Prueba de rangos con signo de Wilcoxon para muestras relacionadas I2 118

Fig. 96 Cuestionario de la dimensión calidad – hoja 1 132

Fig. 97 Cuestionario de la dimensión calidad – hoja 2 133

Fig. 98 Ficha de observación de la dimensión tiempo 134

Fig. 99 Ficha de análisis documental de la dimensión fiabilidad 135

Fig. 100 Ficha de cotejo de la dimensión transparencia 135

Fig. 101 Ficha para validación del cuestionario 136

Fig. 102 Ficha para validación de la ficha de observación 137

Fig. 103 Ficha para validación de la ficha de análisis documental 138

Fig. 104 Ficha para validación de la ficha de cotejo 139

Fig. 105 Coeficiente de fiabilidad para el cuestionario 140

RESUMEN

La Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado, ubicada en la ciudad de Cajamarca, presentaba deficiencias en su gestión académica debido a la ejecución manual de procesos como matrículas, control de asistencias, registro de calificativos y pagos. Estas tareas, realizadas mediante documentos físicos y hojas de cálculo, generaban duplicidad de datos, retrasos, errores, desorganización, pérdida de información y dificultades para la toma de decisiones. Frente a esta problemática, la presente investigación tuvo como objetivo general determinar el impacto de la implementación de un sistema de información web en la gestión académica de la ESFAP MUA. Para el desarrollo del sistema se utilizó la metodología ágil Kanban, permitiendo una gestión flexible de tareas y entregas incrementales. Se emplearon tecnologías como Next.js para el frontend, Node.js para la lógica de negocio, y MySQL como sistema gestor de base de datos, garantizando un software de calidad. Los resultados obtenidos evidencian un impacto positivo, reflejado en una reducción promedio del 19% en los tiempos de ejecución de los procesos académicos más relevantes, como el registro de asistencia, la creación y evaluación de actividades, y la publicación de notas. Asimismo, se eliminaron los casos que comprometían la veracidad del proceso evaluativo, así como también los retrasos en la publicación de notas finales y presentación de actas. En cuanto a la calidad del software, se obtuvo una aceptación del 90% en términos generales, destacando un 100% en adecuación funcional, 93% en usabilidad y seguridad, y 93% en portabilidad. Estos resultados respaldan la hipótesis de que un sistema de información web impacta de manera positiva en la gestión académica de la institución.

Palabras clave:

Gestión académica, sistema de información web, impacto, automatización de procesos.

ABSTRACT

The Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado, located in the city of Cajamarca, exhibited significant deficiencies in its academic management due to the manual execution of processes such as enrollment, attendance tracking, grade recording, and payment management. These tasks, carried out through physical documents and spreadsheets, resulted in data duplication, delays, errors, disorganization, information loss, and challenges in decision-making. In response to this issue, the present research aimed to determine the impact of implementing a web-based information system on the academic management of ESFAP MUA. The system was developed using the agile Kanban methodology, which allowed for flexible task management and incremental deliveries. Technologies such as Next.js for the frontend, Node.js for business logic, and MySQL as the database management system were employed to ensure high-quality software. The results obtained demonstrate a positive impact, reflected in an average reduction of 19% in the execution times of the most relevant academic processes, such as attendance registration, the creation and evaluation of activities, and the publication of grades. Additionally, instances compromising the integrity of the evaluation process, as well as delays in final grade publication and report submission, were eliminated. Regarding software quality, the system achieved a 90% overall acceptance rate, with 100% in functional suitability, and 93% in both usability and security, and portability. These results support the hypothesis that a web-based information system positively impacts the academic management of the institution.

Keywords:

Academic management, web information system, impact, process automation.

CAPÍTULO I. INTRODUCCIÓN

En el ámbito global de la educación, la gestión académica enfrenta una problemática significativa vinculada con la disponibilidad oportuna y confiable de información para la toma de decisiones. La persistencia de procesos manuales y registros físicos limita la eficiencia en la administración de datos académicos, lo que retrasa la implementación, el seguimiento y la evaluación de planes de estudio, matrículas, calificaciones y demás actividades formativas. Estas limitaciones generan deficiencias en la organización y control de los recursos educativos, afectando directamente la calidad de los procesos de enseñanza-aprendizaje y, en consecuencia, el logro de los objetivos pedagógicos establecidos [1].

La gestión académica en América Latina enfrenta múltiples desafíos que impactan directamente en los procesos de enseñanza-aprendizaje. En primer lugar, la equidad en el acceso a la educación sigue siendo limitada, lo que afecta la matrícula de estudiantes, especialmente en contextos rurales frente a los urbanos. Asimismo, los mecanismos de evaluación académica requieren una mejora sustancial en su calidad y pertinencia, incorporando criterios que reconozcan las diferencias individuales y promuevan la equidad en los resultados. Otro reto importante es la adecuada gestión de los recursos educativos, particularmente en lo que respecta a infraestructura y tecnología, elementos esenciales para garantizar experiencias de aprendizaje de calidad. Finalmente, el seguimiento académico debe orientarse hacia un enfoque preventivo, utilizando datos oportunos y confiables que permitan detectar a tiempo riesgos de fracaso escolar y fortalecer los logros pedagógicos [2],[3],[4],[5].

En Perú, la gestión académica enfrenta diversos obstáculos que ralentizan el desarrollo de la educación y comprometen la calidad de los aprendizajes. Uno de los principales problemas es la ausencia de métodos de evaluación rigurosos y basados en evidencias, lo que dificulta la identificación de áreas de mejora y la toma de decisiones pedagógicas fundamentadas [6]. Asimismo, el seguimiento académico presenta deficiencias notorias debido al uso de registros tradicionales, que generan errores, datos imprecisos y pérdida de tiempo, limitando la capacidad docente para monitorear de manera efectiva el progreso estudiantil y aplicar intervenciones oportunas. A ello se suma la falta de capacitación continua de los docentes y la limitada incorporación de infraestructura tecnológica, factores que restringen la innovación en las prácticas educativas. Finalmente, la ausencia de un marco estratégico orientado a la mejora continua impide consolidar procesos académicos eficientes y sostenibles [7],[8].

En la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado, en la ciudad de Cajamarca, la gestión académica se encuentra restringida al manejo de calificativos, asistencias y matrículas a través de fichas físicas y registros en Excel, lo que genera una serie de dificultades en el desarrollo de los procesos educativos. Entre los principales problemas se identifican retrasos en el registro y publicación de notas y asistencias, errores de digitación, desigualdad en los criterios de evaluación entre asignaturas y el incumplimiento del calendario académico. Asimismo, el uso de documentos físicos ocasiona duplicidad de datos, inseguridad de la información y sobrecarga en los tiempos de trámite. Estas limitaciones no solo afectan la organización de las actividades académicas, sino que también reducen la capacidad institucional para generar información oportuna y confiable que respalde la toma de decisiones pedagógicas. Ante este panorama, resulta evidente la necesidad de implementar sistemas de información que fortalezcan la gestión de los procesos académicos, optimicen el acceso a datos y contribuyan a mejorar la calidad educativa.

Por lo expuesto anteriormente, con el objetivo de abordar los retos académicos de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado (ESFAP MUA), considerando el uso de herramientas tecnológicas como lo es un sistema de información, se planteó la siguiente **pregunta**: ¿Cuál es el impacto de la implementación de un sistema de información web en la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado? El **alcance** de esta investigación se basa en mejorar la gestión académica de la ESFAP MUA, a través de la reducción de tiempos, mejora de la fiabilidad y transparencia de sus procesos. La ejecución de la investigación en un marco temporal determinado puede tener **restricciones** relacionadas con la duración del proyecto y la posibilidad de medir las consecuencias a largo plazo de la implementación de un sistema de información, sumado a ello los usuarios que intervienen en el proceso de gestión académica no tienen el mismo nivel de conocimientos en el uso de sistemas informáticos.

La ejecución de esta tesis tiene como **objetivo general** evaluar el impacto de la implementación de un sistema de información web en la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado, y **específicamente** determinar el impacto en el tiempo para realizar tareas, en la fiabilidad y en la trasparencia de dicha gestión. Un sistema de información es un sistema que permite almacenar y procesar información, facilita las operaciones y la toma de decisiones, además, si se ejecuta en la web brinda un plus en cuanto a accesibilidad, en base a ello, se planteó la siguiente **hipótesis**: un sistema de información web

impacta de manera positiva en la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

Esta tesis se **justifica** desde el punto de vista **académico**, porque aporta el diseño, desarrollo e implementación de un sistema de información web en la gestión académica de una institución, los resultados de este estudio podrían servir como base para investigaciones futuras, estimulando un cuerpo de conocimiento en constante evolución. Desde el punto de vista **práctico**, pretende lograr la automatización de ciertos procesos dentro de la gestión académica de una institución, busca ayudar a los actores intervinientes a manejar información de una manera más rápida, segura y organizada, mejorando la experiencia del usuario. A nivel **social** supone la implementación de un sistema de información web académico, el cual permite a los estudiantes y a sus familiares obtener información sin tener que desplazarse largas distancias, a los docentes y administrativos les propicia consultar y gestionar información, dichas actividades pueden realizarse desde cualquier lugar, espacio y tiempo, sólo a través de un dispositivo con capacidades de procesamiento y conexión a internet, minimizando la brecha de desigualdad en el acceso e intercambio de información.

La presente tesis está estructurada y desarrollada en 5 capítulos, el Capítulo I: conformado por la Introducción donde se menciona el problema, hipótesis, justificación, alcances y objetivos, a grandes rasgos. El Capítulo II: contiene el marco teórico (antecedentes, bases teóricas y definición de términos básicos) relacionados a la ingeniería de software de la aplicación y el desarrollo de esta. El Capítulo III: corresponde a los materiales y métodos que se utilizaron en el desarrollo del proyecto de investigación, además de la explicación del tratamiento de los datos, para la obtención de los resultados. El Capítulo IV: muestra la descripción, explicación y discusión de los resultados obtenidos. Finalmente, el Capítulo V: presenta las conclusiones de acuerdo con los objetivos planteados además de las recomendaciones que se consideran necesarias para tratar aspectos que no se incluyeron, y de esta manera seguir ampliando los conocimientos sobre el tema de la investigación.

CAPÍTULO II. MARCO TEÓRICO

2.1 ANTECEDENTES TEÓRICOS

2.1.1 ANTECEDENTES INTERNACIONALES

En Colombia, Diaz [9] en su investigación, tiene como objetivo principal desarrollar e implementar un sistema de información para llevar a cabo el proceso de matrícula, control de asistencias y gestión de notas académicas en una institución educativa, esto a partir de un estudio que evidencia la necesidad de automatizar dichos procesos que se encuentran relacionados a largos tiempos de espera y a un gasto de papel innecesario. En base a ello obtiene como resultado el desarrollo e implementación de un sistema de información óptimo en cuanto a rendimiento y aceptación por parte de los usuarios. Así concluye que el sistema propuesto beneficia en la educación de los estudiantes de manera positiva, sirviendo de apoyo en la implementación de planes de acción y mejora continua. Este antecedente se relaciona con la presente investigación en el desarrollo de un de sistemas información relacionado al proceso de matrículas, control de asistencias y gestión de notas dentro de una institución educativa.

Rodas y Cárdenas [10] en su estudio, tienen como objetivo determinar las causas que originan disconformidad en los usuarios de los planteles educativos en la gestión académica. Los resultados obtenidos manifiestan que la mayoría de las instituciones educativas presentan necesidades en el área de administración, existiendo una tasa del 73% de insatisfacción sobre el desarrollo de actividades relacionadas a dicha área; se expone que la razón de esa inconformidad es que los procesos asociados con el registro de datos se realizan forma manual en documentos físicos, lo cual conlleva a que se genere incongruencia de información difícil de manejar para el personal administrativo, de allí que un 98% de los encuestados exponen estar de acuerdo con implementar un sistema informático que permita administrar información de una forma ordenada y por tanto mejorar la gestión académica. Como conclusión se expresa que existe un caos organizativo en los centros de educación al momento de realizar alguna consulta, lo que no solo resulta en un proceso más lento, sino que también aumenta el riesgo de perder información al estar almacenada en lugares inapropiados y propensos a daños naturales; y que con la implementación y puesta en marcha de un sistema de gestión digital se puede obtener cambios representativos en la satisfacción de los usuarios. Este antecedente ayuda a la presente investigación a confirmar que el uso de un sistema de información es considerado como una de las soluciones a problemas relacionados con la gestión académica.

2.1.2 ANTECEDENTES NACIONALES

En Chiclayo, Ortiz y Pingo [11] en su tesis, tienen como objetivo principal implementar un acceso web y sistema de información que permita integrar los procesos académicos de una escuela de formación artística, de una forma rápida y eficiente. En base a lo descrito obtienen como resultados una mejoría en el nivel de redundancia de datos de la institución, luego de la implementación del sistema pasó de un 20% a un 5%, asimismo la pérdida de tiempo de la consulta de información pasó de 30 minutos a 5 minutos. Los tesistas concluyen que la implementación del portal y sistema web académico influye de manera positiva en la automatización de los procesos académicos, teniendo el software un óptimo desempeño gracias a un correcto estudio previo de la problemática y elección de métodos y herramientas. Este antecedente se relaciona con la presente investigación en que en ambos se busca implementar un sistema de información con el propósito de mejorar la gestión académica dentro de una institución de formación artística.

Enríquez [12] en su tesis, tiene como objetivo principal mejorar la gestión académica de la institución en mención a través de la implementación de un nuevo sistema de información. En base a ello obtiene como resultados un 81.40% de aceptación con respecto al uso del nuevo sistema, una reducción del 66.67% en el tiempo que toma realizar los procesos académicos, una reducción el 93.30% en el tiempo de elaboración de reportes y una mejora del 84.20% en la percepción de apoderados referente al uso del nuevo sistema de información. Como conclusión se expresa que el sistema de información desarrollado demuestra mayor impacto positivo frente al que ya se estaba utilizando teniendo en consideración los resultados para inferir dicha conclusión. Esta tesis ayuda a la presente investigación a comprender que los sistemas de información como cualquier solución tecnológica necesitan actualizarse según las necesidades.

2.1.3 ANTECEDENTES LOCALES

Aliaga y Álvarez [13] en su tesis, tienen como objetivo principal mejorar el proceso de matrículas en la I.E. Túpac Amaru II, mediante la implementación de un sistema académico web, esto en base a un deficiente desarrollo del proceso mencionado que implica riesgos relacionados al manejo de información. En base a ello obtienen como resultados que en promedio el 65% de encuestados asegura que el uso del sistema académico web es de gran apoyo en el proceso de matrículas de la institución en estudio. La conclusión detalla que la base en el buen desarrollo de un proyecto implica un buen análisis, y que el sistema académico implementado está validado en funcionalidad, portabilidad, usabilidad y mantenibilidad; cumpliendo con el objetivo principal. Este antecedente ayuda a la presente investigación a

considerar la buena aplicación de las encuestas para conocer la influencia de los sistemas de información luego de su implementación.

Pérez [14] en su tesis, tiene el objetivo general de desarrollar un sistema de registro de calificaciones como solución a la problemática relacionada al proceso de control y seguimiento de la evaluación académica. Basándose en ello obtiene como resultado que el proceso de la publicación oficial de un calificativo desde su registro a través de un método tradicional toma un aproximado de 355.88 horas, mientras que haciendo uso del sistema de registro de calificaciones tan solo 32.55 horas, evidenciando un decremento del 87.28%. De esta forma concluye con el desarrollo de una solución de software adaptable, que cumple con los objetivos planteados tanto en efectividad, eficiencia y satisfacción, demostrando su incidencia en el proceso de control y seguimiento de la evaluación académica. Esta tesis ayuda a la presente investigación a comprender la gran efectividad del uso de sistemas de información dentro de los procesos académicos cuando es implementado correctamente.

2.2 BASES TEÓRICAS

2.2.1 GESTIÓN ACADÉMICA

Según [9], la gestión académica es el proceso que se realiza en una institución educativa para mejorar la calidad y pertinencia de la oferta y el desarrollo de los programas y proyectos educativos, así como de los procesos pedagógicos, curriculares y de evaluación; implica la coordinación, planificación, ejecución y seguimiento de las actividades académicas, así como la optimización de los recursos humanos, materiales y financieros; de la misma forma, busca también responder a las necesidades y demandas de la sociedad, el mercado laboral y el desarrollo científico y tecnológico.

La gestión académica se refiere a la administración y organización de instituciones, programas y recursos educativos para garantizar una enseñanza y un aprendizaje eficaces. Implica planificar, implementar y evaluar diversos aspectos de los procesos académicos para alcanzar las metas y objetivos educativos, he aquí algunos conceptos de importancia dentro de la gestión académica [15]:

- Análisis de datos y toma de decisiones: La gestión académica implica recopilar y analizar datos sobre el rendimiento de los estudiantes, los resultados de los programas y otras métricas relevantes para fundamentar la toma de decisiones y mejorar los procesos educativos.

- Mejora continua: La gestión académica implica el seguimiento y la evaluación continuos de los procesos y resultados académicos para identificar áreas de mejora y aplicar estrategias que mejoren la calidad de la educación.
- Cumplimiento y asuntos normativos: Los gestores académicos deben garantizar el cumplimiento de las leyes, reglamentos y políticas pertinentes que rigen la educación, incluidas las relacionadas con las normas curriculares, la seguridad de los estudiantes y la igualdad de oportunidades.

Asimismo, está relacionada a una serie de indicadores que miden su correcto desarrollo, entre ellos tenemos:

- El tiempo como indicador es fundamental en la gestión académica, ya que permite medir la eficacia y el progreso de los procesos educativos a lo largo del tiempo, algunos de los aspectos clave donde el tiempo se utiliza como indicador incluyen la planificación y ejecución de evaluaciones, ya que es importante establecer objetivos claros y planificar la evaluación, asignando tiempo adecuado para cada tarea; así como también en el monitoreo de aprendizaje, esencial para entender la evolución de los procesos educativos y para tomar decisiones informadas sobre cómo mejorar la calidad y el rendimiento educativo [16],[17].
- La fiabilidad es un indicador importante en la gestión académica, se refiere a la consistencia y precisión de los datos utilizados para evaluar el rendimiento y el aprendizaje de los estudiantes, y en general el desempeño de una institución educativa; para garantizar la fiabilidad, es esencial que los datos sean coherentes, comparables y auditables [18].
- La transparencia es un indicador clave en la gestión académica y se relaciona con la apertura y claridad en la comunicación de información relevante, en el contexto institucional, la transparencia puede evaluarse desde varios aspectos, entre ellos la trasparencia en la calidad educativa, esto incluye la divulgación de indicadores de desempeño, resultados de evaluaciones y programas de mejora continua según los parámetros establecidos; este indicador es esencial para garantizar que el proceso sea justo, equitativo y que los resultados sean confiables y aceptados por toda la comunidad educativa [19],[20].

En general, la gestión académica desempeña un papel crucial a la hora de garantizar el funcionamiento eficiente y eficaz de las instituciones educativas y la consecución de los objetivos educativos, requiere una combinación de liderazgo, planificación estratégica y habilidades operativas para crear un entorno propicio para el aprendizaje y la excelencia académica, se pudo constatar que los estudios sobre la gestión en el ámbito de la educación son

variados y multifactoriales, todos concuerdan en la necesidad de una gestión académica eficiente para el logro de procesos de calidad en los centros educativos [21].

2.2.2 SISTEMA DE INFORMACIÓN WEB

Un sistema de información es un conjunto de procesos que, a partir de un conjunto de datos estructurados, de acuerdo con las necesidades de una empresa, recopilan, preparan y difunden selectivamente información relevante para la gestión de dicha empresa y para las funciones de gestión y control correspondientes, y estos apoyan, al menos parcialmente, los procesos de toma de decisiones necesarios para implementar su estrategia [22].

Todo sistema de información utiliza datos como materia prima, los cuales almacena, procesa y transforma para obtener información como resultante final, dicha información es distribuida a los diversos usuarios del sistema, existiendo además un proceso de feedback, en la cual se ha de verificar que la información obtenida responda a lo esperado, junto a los datos los otros dos elementos básicos que integran un sistema de información son los usuarios (directivos, empleados cualquier persona de la organización que utilice la información en su área de trabajo) y los equipos informáticos (software, hardware y tecnologías de la información y de las comunicaciones) [23].

Un sistema de información web (SIW) es un sistema informático diseñado para gestionar y suministrar información a través de Internet o una intranet, abarca varios componentes y tecnologías para recopilar, procesar, almacenar y presentar información a los usuarios en un entorno basado en la web [24]. He aquí algunos conceptos clave asociados a los sistemas de información web:

- La infraestructura frontend: Se refiere al conjunto de tecnologías y herramientas que se utilizan para desarrollar la parte delantera de un sitio web o aplicación, es decir, todo lo que los usuarios ven e interactúan directamente, incluye aspectos como la estructura de la página, el diseño, los estilos, los colores, las animaciones y la interactividad; las tecnologías más comunes utilizadas en el desarrollo frontend son HTML (para la estructura del contenido), CSS (para el estilo y diseño visual) y JavaScript (para la interactividad y funcionalidades dinámicas) [25]. Next.js es un marco web de desarrollo front-end de React de código abierto, incorpora una experiencia de desarrollo con características listas para producción, como renderizado del lado del servidor, compatibilidad con TypeScript, agrupamiento inteligente, prefetching de rutas, etc., le da estructura a las funcionalidades de React.js mientras introduce algunas de sus propias características, tiene una visión sobre

cómo debería organizarse la aplicación, enrutando las páginas a través de un sistema simple de carpetas de archivos [26].

- Infraestructura backend: Es el conjunto de recursos de hardware, infraestructura de red y tecnología de software que permite que una aplicación funcione correctamente, comprende servidores, bases de datos y aplicaciones de software responsables de procesar las peticiones de los usuarios, gestionar los datos y generar respuestas. Incluye componentes como servidores web, servidores de aplicaciones, sistemas de gestión de bases de datos (DBMS) y lenguajes de scripting del lado del servidor [25]. Node.js, una plataforma de ejecución la cual permite ejecutar JavaScript de parte del servidor que al combinarse con Express.js que es un framework que funciona sobre Node.js permitiendo generar enrutamiento para trabajar sobre el protocolo HTTP, como las principales características de Node.js tenemos que permite desarrollar aplicaciones de red escalables, impulsada por eventos asíncronos, si no hay trabajo por hacer Node.js permanece dormido hasta que haya una conexión para que se active una llamada al servicio, todo esto es posible ya que se usa JavaScript del lado de servidor [27].
- Gestión de bases de datos: Los SIW suelen implicar la gestión de grandes volúmenes de datos estructurados y no estructurados, los sistemas de gestión de bases de datos (SGBD) se utilizan para organizar, almacenar, recuperar y manipular datos de forma eficiente, las opciones más populares son MySQL, PostgreSQL, MongoDB y Oracle, cada uno con sus propias características y usos específicos [28]. MySQL es un sistema de gestión de bases de datos (SGBD) muy popular y utilizado debido a su simplicidad y excelente rendimiento, es una opción atractiva tanto para aplicaciones comerciales como de entretenimiento, gracias a su facilidad de uso y rápido tiempo de implementación; además, su distribución gratuita en Internet bajo la licencia GPL lo hace aún más accesible; MySQL Workbench es la interfaz gráfica de usuario de Oracle para consultar y gestionar MySQL Server, junto con MySQL Shell, se puede considerar como una de las dos herramientas todo en uno para trabajar con MySQL [29].
- Integración de datos: Los SIW suelen integrar datos de diversas fuentes y formatos para ofrecer una visión unificada a los usuarios, esta integración de datos implica procesos como la extracción, transformación y carga de datos, así como el uso de API y servicios web para acceder a fuentes de datos externas [30].

Tradicionalmente, el software ha sido utilizado principalmente por personas a través de interfaces de usuario gráficas. Sin embargo, en la actualidad, es cada vez más común que

las aplicaciones informáticas interactúen entre sí sin intervención humana directa. Esta evolución ha llevado al desarrollo y uso de las Interfaces de Programación de Aplicaciones (API, por sus siglas en inglés), las cuales constituyen un mecanismo esencial para habilitar la conexión, integración y expansión de sistemas de software. Más concretamente, las API se utilizan para construir sistemas distribuidos cuyos componentes están débilmente acoplados, lo que favorece la escalabilidad y mantenibilidad de las aplicaciones. En el presente estudio, se trabajará con API web, las cuales se implementan como servicios web y permiten el acceso a recursos de datos mediante una pila de tecnologías web, facilitando así la interoperabilidad entre distintos sistemas a través de la red [31].

- La seguridad: La seguridad es un aspecto crítico de los sistemas de información web para proteger los datos de accesos no autorizados, manipulaciones y violaciones, incluye técnicas de encriptación, autenticación de usuarios, mecanismos de control de acceso, prácticas de codificación seguras y auditorías periódicas de seguridad [31]. JSON Web Token (JWT) es un estándar abierto (RFC 7519) que define una forma compacta y autónoma de transmitir información de forma segura entre partes como un objeto JSON, esta información se puede verificar y es confiable porque está firmada digitalmente; uno de los principales usos de JWT es la autorización. En el backend se puede generar un token JWT que contenga la información del usuario y una firma de seguridad. Este token se envía al frontend, y posteriormente el frontend lo adjunta en las solicitudes para acceder a las rutas de la API que están protegidas [32].
- Escalabilidad y rendimiento: A medida que crecen los sistemas de información web, necesitan gestionar volúmenes crecientes de tráfico y datos sin comprometer el rendimiento; la escalabilidad se refiere a la capacidad de un sistema para manejar un aumento en la carga de trabajo, como más usuarios o transacciones, sin comprometer la calidad del servicio; por otro lado, el rendimiento se relaciona con la rapidez y eficiencia con la que un sistema de información web puede procesar las solicitudes y entregar los resultados a los usuarios [33].
- Compatibilidad móvil: Se refiere a la capacidad de un sitio web para funcionar correctamente y ofrecer una experiencia de usuario óptima en dispositivos móviles como teléfonos inteligentes y tabletas; con el aumento del uso de dispositivos móviles para acceder a internet, es crucial que los sitios web estén diseñados para adaptarse a diferentes tamaños de pantalla y condiciones de uso, algunos aspectos importantes de la compatibilidad móvil incluyen el diseño responsivo, tiempos de carga rápidos, navegación intuitiva, texto legible y evitar ventanas emergentes innecesarias [34].

Para el desarrollo del presente estudio también es propicio describir el concepto de sistema de información académico, el cual puede definirse como una plataforma informática para la interacción de usuarios y equipos informáticos, la cual facilita la captura, almacenamiento, procesamiento, acceso y salida de información confiable y actualizada relacionada con temas, proyectos y actividades académicas [35], representando un gran apoyo hacia el proceso de evaluación académica, en el que se recoge y analiza información sobre el aprendizaje y el desempeño, con el fin de mejorar la calidad y la eficacia de la educación, orientar el proceso de enseñanza y aprendizaje, proporcionando retroalimentación y orientación, de la misma manera, demostrar la pertinencia y el impacto de la educación [36]. Estos sistemas son fundamentales para el manejo eficiente de los procesos educativos y administrativos, y suelen incluir funcionalidades como [37]:

- Registro y admisión de estudiantes: Automatización de los procesos de inscripción y matrícula.
- Gestión de cursos y horarios: Organización de la oferta académica, asignación de aulas y horarios.
- Seguimiento académico: Monitoreo del progreso y rendimiento de los estudiantes.
- Evaluaciones y calificaciones: Registro y análisis de las evaluaciones y resultados académicos.
- Comunicación entre docentes y estudiantes: Facilitación de la interacción y el intercambio de información.
- Administración de recursos educativos: Gestión de materiales didácticos y bibliotecas virtuales.

Un sistema de información es una herramienta de gran importancia para cualquier proceso, es por ello que dicho sistema debe contar con una serie de características que demuestren la calidad del mismo. La ISO/IEC 25010 es una norma que está direccionada hacia la usabilidad, en el cual se establecen las características de calidad que se deben tener en cuenta en el momento de evaluar las propiedades de un producto software terminado, entre las características establecidas se comprende [38]:

- Adecuación funcional. Representa la capacidad del producto de software para brindar funciones que satisfacen las necesidades de los usuarios, ya sean declaradas o implícitas, cuando el producto se utiliza en las condiciones especificadas.

- Eficiencia de desempeño. Representa el desempeño de un producto en el desarrollo de sus funciones dentro de unos parámetros de tiempo y rendimiento especificados y con un uso eficiente de recursos (CPU, memoria, almacenamiento, entre otros) utilizados bajos condiciones determinadas.
- Usabilidad. Capacidad del producto de software para que el usuario interactúe a través de su interfaz intercambiando información para culminar determinadas tareas, esta contiene a su vez subcaracterísticas como la capacidad del producto que permite al usuario entender si el software es adecuado para sus necesidades, la capacidad que permite aprender su funcionamiento, capacidad que permite operarlo con facilidad, capacidad para prevenir errores en su operación y de presentar sus funciones e información de forma atractiva, fomentando la interacción continua.
- Fiabilidad. Hace referencia a la capacidad del producto de llevar a cabo sus funciones sin fallos bajo características normales de operación como también bajo presencia de fallos en hardware o software, capacidad de estar operativo y disponible para uso cuando sea requerido asimismo la capacidad para recuperar datos afectados en caso de interrupción o fallo.
- Seguridad. Capacidad de protección de información y datos de manera que los usuarios tengan el grado de acceso a los datos según a sus tipos y niveles de autorización, y para defenderse de los patrones de ataque de agentes malintencionados.
- Flexibilidad. Capacidad del producto para adaptarse a cambios en sus requisitos, contextos de uso o entornos de hardware o software.

Desarrollar un software de calidad depende de un gran número de actividades y etapas, donde el impacto de elegir la metodología para un equipo en un determinado proyecto es de vital importancia para el éxito del producto, la metodología es la que brinda el conjunto de prácticas, técnicas y herramientas a utilizar por los equipos de desarrollo para planificar, diseñar, construir, probar y entregar software de manera eficiente y efectiva, además de que establece una estructura para el ciclo de vida del software, que comprende la definición de requisitos, el diseño, la codificación, la prueba, la implementación y el mantenimiento; también establece roles y responsabilidades para los integrantes del equipo, procesos para la gestión de proyectos, la comunicación y el seguimiento del progreso; los tipos de metodologías se dividen en dos grupos, metodologías tradicionales y metodologías ágiles [39].

Un modelo de desarrollo ágil, generalmente es un proceso incremental, cooperativo, sencillo y adaptativo; proporcionan una serie de pautas y principios que hacen que la entrega de proyecto

sea menos complicada y más satisfactoria tanto para los clientes como para los equipos de trabajo, siendo más módica en tiempo y recursos, obteniendo los mismos, o en ciertos casos, mejores resultados ante las metodologías tradicionales, dentro de las metodologías ágiles están Scrum, Programación Extrema, Kanban, entre otras [40].

En particular, la metodología Kanban es un sistema visual de gestión del flujo de trabajo utilizado para mejorar la eficiencia y la productividad en el desarrollo de software. Su objetivo principal es controlar el flujo de trabajo y evitar la sobrecarga de tareas. Algunos conceptos clave de la metodología Kanban son [41]:

- Tablero Kanban: Es la herramienta visual principal. Un tablero (físico o digital) se divide en columnas que representan las diferentes etapas del proceso (por ejemplo, "Por hacer", "En progreso", "Hecho"). Las tareas o ítems se representan con tarjetas que se mueven de columna en columna a medida que avanzan en el proceso (ver Fig. 1) [42].
- Límites de Trabajo en Progreso (WIP - Work in Progress): En Kanban, se establecen límites para la cantidad de tareas que pueden estar en progreso en cada etapa del proceso, evitando así la sobrecarga y ayudando a mantener el flujo constante de trabajo.
- Flujo continuo: La idea es que las tareas fluyan de manera constante y sin interrupciones entre las diferentes fases del proceso. El objetivo es maximizar la eficiencia y evitar cuellos de botella.
- Mejora continua: Kanban también fomenta la mejora continua, ya que permite a los equipos analizar el flujo de trabajo y hacer ajustes cuando sea necesario para optimizar el proceso.

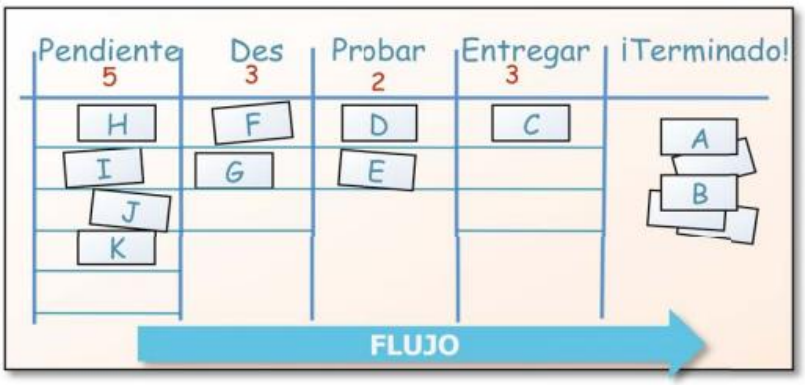

Fig. 1 Representación de la metodología Kanban

La metodología Kanban se enfoca en la gestión visual y en mantener un flujo de trabajo constante, ajustándose a las necesidades y requisitos de cada proyecto sin tener que seguir estrictamente una secuencia de pasos o fases preestablecidas; demás, se basa en la mejora

continua de los procesos mediante métodos empíricos, implementando los cambios necesarios para adaptarlos y optimizarlos a través de la observación y la experiencia [43].

En Kanban, varios equipos o personas pueden compartir el mismo tablero, no hay roles prescritos, la priorización de tareas es opcional y siempre que haya capacidad disponible, se pueden añadir tareas. Lo único que establece Kanban es que el flujo de trabajo debe ser visible y que se debe limitar la cantidad de trabajo en proceso. El propósito es lograr un flujo continuo a través del sistema y reducir el tiempo de entrega [42].

2.3 DEFINICIÓN DE TÉRMINOS BÁSICOS

- **Impacto.** Cambio en el resultado de un proceso o producto, dependiendo de cómo se realiza el proceso, de la prácticas que se utilizan, y de la persona o personas que la ejecutan [44].
- **Proceso.** Conjunto de actividades que se encuentran interrelacionadas, las cuales se caracterizan por requerir de ciertas entradas (inputs) para obtener resultados (outputs) [45].
- **Gestión académica.** Conjunto de procesos que permiten planificar, organizar, coordinar y controlar las actividades educativas de una institución, incluyendo matrícula, evaluación, control de asistencia y registro de calificaciones [46].
- **Transparencia académica.** Acceso claro, veraz y verificable a la información institucional para todos los actores del proceso educativo, promoviendo la rendición de cuentas y la confianza [47].
- **Fiabilidad de datos académicos.** Capacidad del sistema para generar y mantener información académica precisa, consistente y libre de errores, garantizando la integridad de los registros [48].
- **Tiempo de gestión académica.** Lapso requerido para ejecutar tareas propias de la gestión institucional (registro de asistencia, calificaciones, matrícula, etc.), cuya reducción es indicativa de un sistema más eficiente [49].
- **Metodología.** Una metodología es un conjunto integrado de técnicas y métodos que permite abordar de forma homogénea y abierta cada una de las actividades del ciclo de vida de un proyecto de desarrollo [50].
- **Software.** Soporte lógico de un sistema informático, hace a un programa o conjunto de programas de cómputo que incluyen datos, reglas e instrucciones para poder comunicarse con el ordenador y hacen posible su funcionamiento [51].

- **Tecnologías web.** Conjunto de lenguajes, herramientas y marcos tecnológicos utilizados para desarrollar aplicaciones web modernas. En este contexto, tecnologías como SQL, Node.js y Next.js ofrecen una infraestructura completa que permite construir aplicaciones altamente escalables, eficientes y coherentes en cuanto a lógica de servidor y cliente [52].
- **Stakeholder.** Traducido al español como "grupos de interés" o "partes interesadas", son las personas, grupos u organizaciones que se ven afectadas por las acciones y decisiones de una empresa [53].
- **Product backlog.** Es una lista de trabajo ordenado por prioridades para el equipo de desarrollo que se obtiene de su hoja de ruta y sus requisitos [54].

CAPÍTULO III. MATERIALES Y MÉTODOS

La Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado, es una institución que forma Artistas y Docentes en arte utilizando los avances de la ciencia y la tecnología, comprometidos en la investigación y proyección del arte con un enfoque intercultural. Su funcionamiento se viene dando desde el 19 de diciembre de 1986, bajo el nombre inicial de Instituto Superior de Arte "José Sabogal", actualmente cuenta con 4 carreras profesionales las cuales son: Docencia en Artes Visuales, Docencia en Música, Artista Profesional en Pintura y Artista profesional en Escultura.

El desarrollo de la investigación se llevó a cabo en un periodo de 8 meses, considerados a partir del 20 de mayo de 2024 hasta el 20 de enero de 2025. Se realizó un seguimiento de las coordinaciones necesarias para iniciar el ciclo académico 2024-II y los procesos administrativos relacionados, que sirvió como base para el desarrollo del Sistema Informático Académico, a nivel de requerimientos, asimismo se consideró su análisis, diseño, pruebas, implementación y resultados.

3.1 PROCEDIMIENTO

La presente investigación fue de tipo aplicada, debido a que buscó generar conocimiento con una aplicación directa en la gestión académica de la Escuela Superior de Formación Artística Pública Urteaga Alvarado, además, este tipo de investigación se caracteriza por transformar el conocimiento teórico en prototipos y productos que respondan a las necesidades reales de la sociedad, en este caso el producto es un sistema de información web [55].

El nivel de investigación fue **explicativo**, pues se enfoca en comprender el impacto que provoca la utilización de un sistema de información web en la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado [56].

Para evaluar el impacto de un sistema de información sobre la gestión académica de la institución, se aplicó un diseño experimental **pre-experimental**, el cual involucra la manipulación de la variable independiente (pre-test post-test) para evaluar el efecto que tiene sobre la variable dependiente; en una situación previa y posterior referente a un solo grupo [57].

El método utilizado en la presente investigación es el **inductivo**, ya que, a partir del estudio de casos específicos, como el control de tiempos, la fiabilidad de cálculos, la transparencia de

procesos, se buscó obtener conclusiones generales sobre el impacto de un sistema de información web en la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado [58].

La población de estudio estuvo compuesta por 29 docentes de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado, dado que la población es pequeña, la muestra fue conformada por todos los docentes de la mencionada institución; y la unidad de análisis estuvo representada por cada uno de dichos docentes. En consecuencia, en el estudio se incluyó una matriz de operacionalización de variables (Anexo 1) para establecer las variables y sus indicadores, así como una matriz de consistencia (Anexo 2) para garantizar la coherencia entre el problema, los objetivos y la hipótesis.

En el proceso de recopilación de datos se emplearon diversas técnicas e instrumentos enfocados en analizar las variables de estudio. Para examinar la variable independiente, *sistema de información web*, se aplicó la encuesta como técnica y un cuestionario como instrumento (Anexo 3). Para la variable dependiente, *gestión académica*, destinada al estudio de la dimensión tiempo, se utilizó la técnica de observación con su ficha de observación (Anexo 4). Para la dimensión fiabilidad, se empleó la técnica de revisión documentaria con su ficha de análisis documental (Anexo 5), y para la dimensión transparencia, también se utilizó la observación como técnica, con su ficha de cotejo como instrumento (Anexo 6).

Los instrumentos de recolección de datos (cuestionario, ficha de observación, análisis documental y ficha de cotejo) fueron validados por un juicio de expertos (Anexo 7); quienes realizaron evaluaciones de manera objetiva e imparcial, basadas en razonamiento y juicio propio, que pueden o no reflejar la utilidad de los instrumentos.

Para el proceso de recolección de datos de la variable independiente se creó el cuestionario con el objetivo de evaluar la calidad del sistema desarrollado. Las interrogantes del cuestionario fueron elaboradas considerando la ISO 25010 [38] evaluando su adecuación funcional, eficiencia de desempeño, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad; para cada uno de los parámetros se plantearon preguntas cerradas que emplean una escala de Likert, lo que facilitó la recopilación de información sobre la percepción de un experto. Una vez desarrollado el cuestionario, se evaluó la confiabilidad de este a través del coeficiente Alfa de Cronbach, haciendo uso de un software de licencia libre especializado en el procesamiento de datos y análisis estadísticos, obteniendo un resultado de 0.881 lo que indica una buena confiabilidad (Anexo 8).

En el proceso de recolección de datos, se aplicó una ficha de observación (Anexo 4) antes y después de la implementación del sistema de información web; esta se utilizó en una clase por cada docente y, con la ayuda de un cronómetro, se midió el tiempo que tomaban en realizar tareas relacionadas con procesos como la toma de asistencia por clase, la creación de actividades, la evaluación de actividades, y el registro y publicación de notas, con el fin de realizar una comparativa entre los resultados de ambos momentos. En esta investigación también se diseñó y aplicó una ficha de análisis documental de pre test y post test (Anexo 5), con el objetivo de validar la fiabilidad en el cálculo de las notas y los porcentajes de asistencia de los estudiantes; en el pre test, se revisaron algunos de los registros manuales de los docentes, con el fin de identificar posibles errores de digitación o cálculo y en el post test, se analizaron los documentos exportados desde el sistema implementado, con el mismo propósito. Asimismo, se diseñó y utilizó una ficha de cotejo (Anexo 6) para registrar la cantidad de días que un docente excedía las fechas establecidas en los procesos de publicación de notas y presentación de actas; con esta información, se buscó determinar el nivel de transparencia en función de la puntualidad en su ejecución, tanto antes como después de la implementación del sistema web, esta acción tuvo como propósito prevenir posibles manipulaciones indebidas y respaldar el cumplimiento de los plazos establecidos institucionalmente.

Finalmente, se especifica el tratamiento que se llevó a cabo con los datos recolectados, el cual fue realizado mediante el uso de un software estadístico especializado; este procesamiento tiene como propósito analizar e interpretar los resultados obtenidos de manera rigurosa.

El desarrollo del sistema de información web para la gestión académica de la ESFAP MUA se llevó a cabo utilizando la metodología Kanban, en virtud de las ventajas que ofrece frente a otras metodologías ágiles. Entre sus principales beneficios destacan la gestión visual de las tareas y la optimización del flujo continuo de trabajo, lo que permitió una mayor flexibilidad en la organización de actividades y una rápida capacidad de adaptación frente a los cambios y requerimientos surgidos durante el proceso de desarrollo.

Aunque Kanban no tiene “fases” en sentido estricto, se pueden identificar etapas clave para su implementación, en esta investigación se optó por considerar las siguientes fases y pasos (Ver Fig. 2):

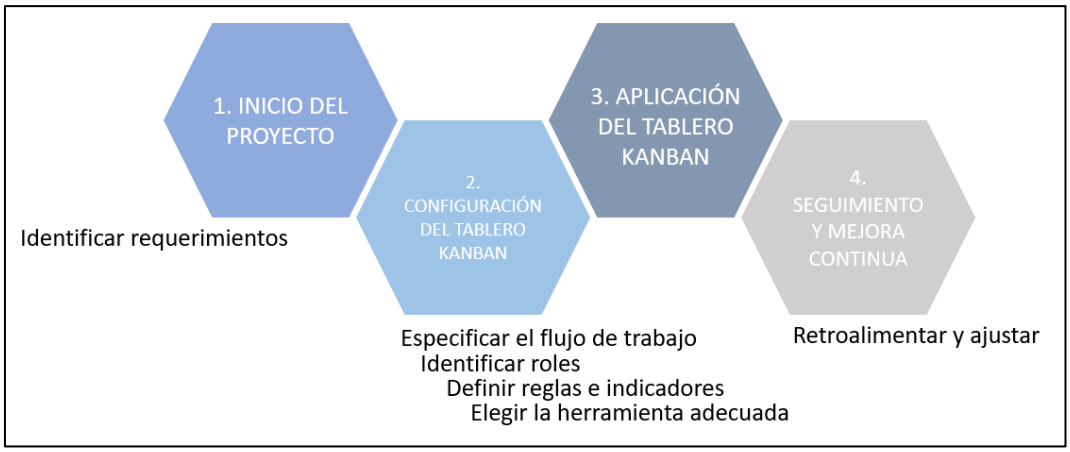

Fig. 2 Fases y pasos definidos para la implementación de la metodología Kanban

3.1.1 INICIO DEL PROYECTO

3.1.1.1 IDENTIFICAR REQUERIMIENTOS

Para comprender el flujo del negocio actual de la organización, se elaboró una representación gráfica de dos de los procesos más importantes, estas representaciones serán propicias en la identificación de puntos de mejora en el desarrollo de los procesos académicos de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado. Estos procesos se representaron mediante diagramas de flujo, utilizando el software Bizagi Modeler.

- **Gestión de matrícula**

  El proceso inicia con el pago por derecho de matrícula, el cual se realiza cada inicio de ciclo de manera presencial por cada estudiante, luego de realizado el pago el estudiante debe acercarse al centro de estudios para continuar con su proceso de matrícula, a continuación, en la Fig. 3, se muestra el diagrama del flujo del proceso en mención, incluyendo tiempos para las actividades que son comunes entre todos los estudiantes y así obtener un el tiempo promedio que representa el proceso de matrícula por cada estudiante.

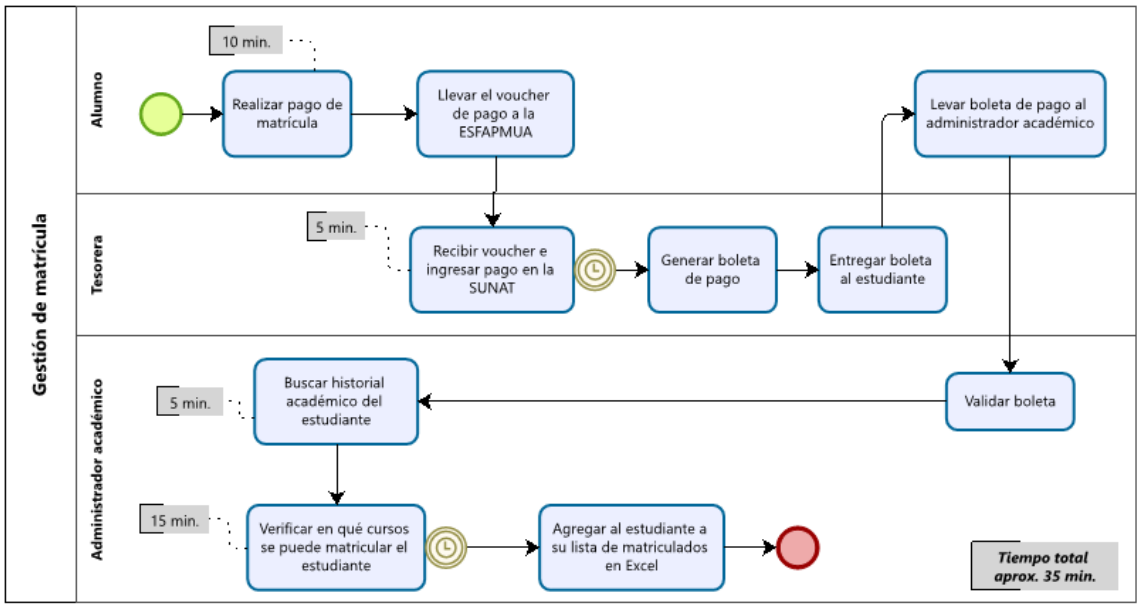

Fig. 3 Diagrama del proceso "Gestión de matrícula"

- **Gestión de calificaciones por curso**

  El proceso inicia con el registro de notas de cada docente durante el ciclo, pero debido a que esta tarea varía en cada uno de ellos tanto en fechas como en la cantidad de notas que consideran, se decidió graficar desde el hecho en el que el docente ya tiene su consolidado de notas finales en adelante, tal y como se muestra en la Fig. 4.

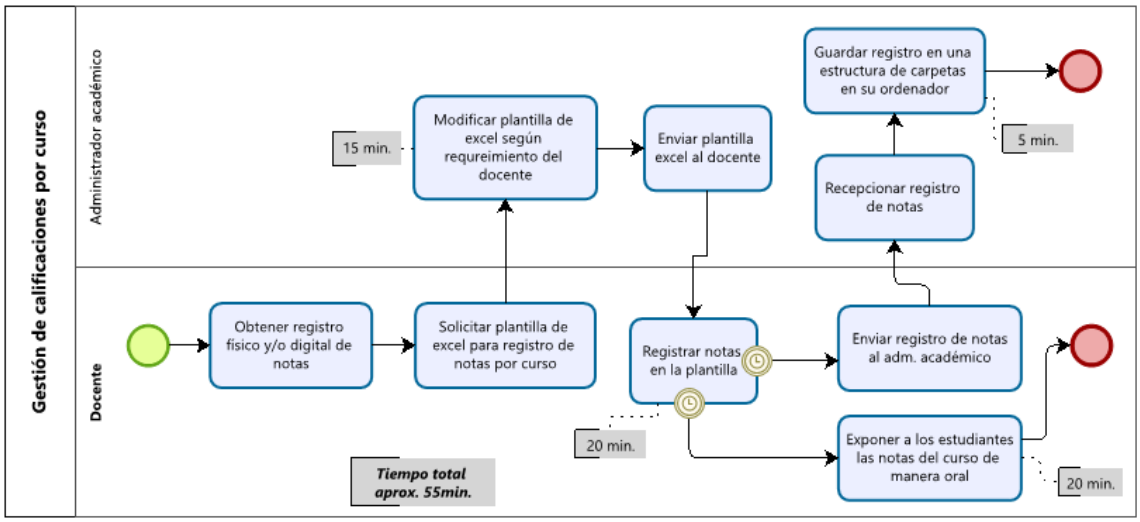

Fig. 4 Diagrama del proceso "Gestión de calificaciones"

Tras la descripción y presentación de los diagramas de flujo iniciales de algunos de los procesos que intervienen en la gestión académica de la institución en estudio, además del estudio del estado actual de los procesos en otras áreas, se identificó la necesidad de contar con un sistema informático para la gestión de sus procesos, debido a que las actividades requieren orden

seguimiento y control, asimismo se requiere que los datos y la información esté concentrada y disponible, en base a ello se procedió a identificar sus requerimientos, siendo los siguientes:

Requerimientos funcionales

- RF01: Inicio de sesión mediante el uso de un correo y contraseña

- *Perfil administrador*
  - RF02: Crear, listar, editar y eliminar estudiantes
  - RF04: Crear, listar, editar y eliminar docentes
  - RF05: Crear, listar, editar y eliminar periodos
  - RF06: Listar, editar, eliminar matrículas
  - RF07: Buscar y listar historial de notas por cada estudiante
- *Perfil soporte*
  - RF08: Crear, listar, editar y eliminar jefes de departamento
  - RF09: Crear, listar, editar y eliminar cursos
  - RF10: Crear, listar, editar y eliminar usuarios del sistema
- *Perfil jefe de departamento*
  - RF10: Crear, listar, editar, eliminar, asignar horario y asignar docente a cursos por dictar
- Perfil docente
  - RF11: Listar cursos por dictar, listar, editar, eliminar sesiones de clase
  - RF12: Listar, calificar, tomar asistencia a alumnos por cada curso
  - RF13: Crear, editar, eliminar, calificar actividades por cada sesión de clase
  - RF14: Generar acta de notas por cada curso al finalizar el ciclo académico
  - RF15: Visualizar horario de clases
- Perfil tesorería
  - RF16: Crear, listar, editar y eliminar pagos
- Perfil estudiante
  - RF17: Generar matrícula
  - RF18: Visualizar sesiones de clase, notas y porcentaje de asistencia por cada curso matriculado
  - RF19: Completar y visualizar su respectiva nota de cada una de las actividades generadas por los docentes
  - RF20: Visualizar horarios de clase
  - RF21: Visualizar historial de notas
  - RF22: Visualizar pagos

Requerimientos no funcionales

- RNF01: El sistema debe desarrollarse con tecnologías que permitan su mantenimiento futuro
- RNF02: Se debe utilizar una base de datos SQL
- RNF03: El sistema debe ser accesible desde un computador y celular sin que se distorsione su apariencia
- RNF04: Los datos sensibles como contraseñas deben mantenerse cifrados
- RNF05: El sistema debe desplegarse en un servidor físico adquirido por la institución
- RNF06: Se debe utilizar el dominio de internet actual con el que cuenta la institución
- RNF07: Las API deben responder en menos de 200 ms bajo condiciones normales de carga
- RNF08: La interfaz del sistema debe hacer uso de colores que identifiquen a la institución

3.1.2 CONFIGURACIÓN DEL TABLERO KANBAN

3.1.2.1 ESPECIFICAR EL FLUJO DE TRABAJO

El flujo de trabajo utilizado para el desarrollo del proyecto durante la elaboración del sistema web fue el siguiente:

- Backlog. Lista general de tareas, tanto las que pertenecen a los requerimientos funcionales, como a los no funcionales.
- Ready. Tareas ya analizadas, con criterios de aceptación claros y listas para empezar.
- Design. Para cuando las historias necesitan maquetación o diseño previo.
- Doing. Tareas sobre las que se está trabajando.
- Code Review. La tarea ya tiene código y está pendiente de revisión por otro desarrollador.
- Testing. Tareas que están siendo sometidas a pruebas para verificar su funcionamiento correcto.
- Done. Quinta etapa, aquí se ubican las tareas completadas satisfactoriamente y que están listas para el pase a producción.

3.1.2.2 IDENTIFICAR ROLES

En Kanban no existen roles rígidos, pero se fomenta la colaboración dentro de equipos multidisciplinarios con el objetivo de acelerar la entrega de valor. Según Anderson [59], uno de

los principios fundamentales de Kanban es “fomentar el liderazgo en todos los niveles” y “habilitar a los equipos para autoorganizarse alrededor del flujo de trabajo”. En este contexto, se conformó un equipo de trabajo cuyos roles fueron definidos según el nivel de intervención en la investigación y los aportes realizados al desarrollo de la solución, la cual se detalla en la Tabla I.

Tabla I Equipo de trabajo

Rol en Kanban	**Quién lo asume**	**Responsabilidades clave**
Service Delivery Manager - Service Request Manager	Jhan Villanueva Vargas	- Monitorea las métricas de Kanban - Se encarga de supervisar que el flujo de trabajo sea estable y eficiente. - Definir la visión, estrategia y objetivos del producto. - Definir el alcance y prioridades de las historias de usuario - Validar entregables y criterios de aceptación - Mantener el Product Backlog actualizado - Asegurar que el producto tenga un impacto real.
Equipo de Desarrollo	- Jhan Villanueva Vargas - 02 Programadores	- Implementar funcionalidades según las historias asignadas - Realizar pruebas unitarias - Reportar avances y bloqueos en el Daily Kanban
Stakeholders / Asesores	- Daniel Cotrina Rowe (Jefe Unidad Académica ESFAP MUA) - Ing. Manuel Malpica Rodríguez	- Revisar entregas periódicas - Aportar feedback académico y metodológico

El desarrollo del sistema web propuesto fue liderado por el autor de esta tesis, quien asumió la responsabilidad integral del proyecto: desde el análisis de requerimientos, diseño de la arquitectura, desarrollo e implementación del sistema, hasta la validación final con los usuarios. No obstante, durante la etapa de desarrollo técnico se contó con el apoyo puntual de dos colaboradores, cuya intervención se restringió a tareas específicas de codificación, realizadas siempre bajo supervisión directa del autor y representando aproximadamente el 30 % del trabajo total en esta fase.

3.1.2.3 DEFINIR REGLAS E INDICADORES

En cuanto a las reglas y límites del trabajo en curso (WIP), se estableció un máximo de tres tareas por cada una de las etapas del flujo de trabajo que involucran procesos en ejecución (ni pendientes ni finalizados), considerando el número de desarrolladores asignados al proyecto (3). Este límite implica que cada desarrollador solo puede mantener activa una tarjeta por etapa de forma simultánea. En otras palabras, un integrante del equipo no podrá añadir una nueva tarea en determinada columna mientras ya tenga una en esa misma etapa, hasta que la actual haya avanzado a la siguiente fase del flujo. Esta restricción permitió mantener el enfoque individual, prevenir la sobrecarga de tareas y facilitar la detección temprana de cuellos de botella durante el desarrollo del sistema web. Es importante que el WIP esté indicado las columnas correspondientes, así como se visualiza en la Fig. 5:

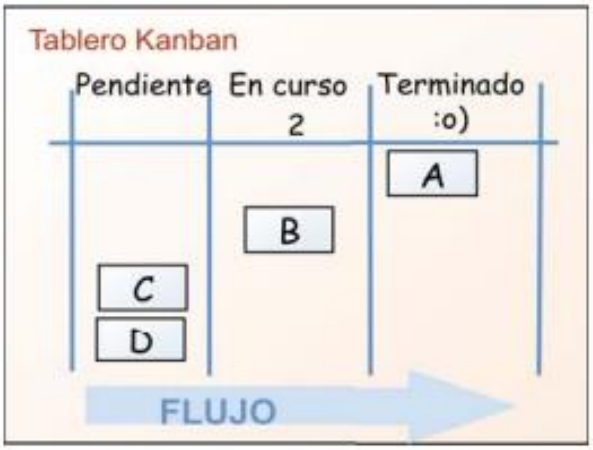

Fig. 5 WIP en el tablero Kanban

Indicadores Kanban

En Kanban se suelen usar indicadores que ayudan a tener una visión más completa del flujo de trabajo y la eficiencia del equipo, por cada tarea del backlog se usaron los siguientes:

- Lead time.

  Qué es: El tiempo total que transcurre desde que una tarea se solicita (entra al backlog) hasta que se entrega (llega a la columna "Done").

  Perspectiva: Mide la velocidad de entrega de valor al cliente.

- Cycle time.

  Qué es: El tiempo que tarda una tarea desde que empieza a trabajarse (pasa de pendiente a en progreso) hasta que se termina.

  Perspectiva: Se centra en la eficiencia interna del equipo, no en la espera del cliente.

- Flow Efficiency.

  Qué es: Porcentaje de tiempo en el que una tarea realmente se está trabajando frente al tiempo total que pasa en el sistema.

  Fórmula: $FE = \frac{Tiempo\ activo}{Lead\ Time}\ x\ 100$

  Qué aporta: Si la eficiencia es baja, hay mucho tiempo muerto o de espera entre etapas.

3.1.2.4 ELEGIR LA HERRAMIENTA ADECUADA

Durante el desarrollo web del sistema académico, se utilizó Trello como herramienta digital para la implementación del tablero Kanban, con el objetivo de organizar, visualizar y controlar el flujo de trabajo del equipo de desarrollo. Trello es una plataforma de gestión de proyectos basada en la web que se adapta fácilmente a entornos que aplican metodologías ágiles, como Kanban, por su enfoque visual, su facilidad de uso y su capacidad de colaboración en tiempo real. El tablero de Trello se estructura mediante tableros, listas y tarjetas. Cada tarjeta representa una tarea o requerimiento relacionado con el desarrollo web (por ejemplo, diseño de interfaces, implementación de funcionalidades, corrección de errores o pruebas), y puede moverse entre listas que reflejan los distintos estados del proceso, tales como "Pendiente", "En desarrollo", "En revisión" y "Completado". Esta estructura permite monitorear el avance del proyecto de manera clara y ordenada.

Entre los principales beneficios del uso de Trello en el contexto del desarrollo web se destacan los siguientes:

- Interfaz intuitiva y accesible: Trello ofrece una experiencia de usuario simple y clara, lo que facilita su adopción por parte de los miembros del equipo, sin requerir conocimientos técnicos avanzados. Al estar basado en la nube, permite acceder al proyecto desde cualquier dispositivo con conexión a Internet.
- Colaboración efectiva: La herramienta permite a los integrantes del equipo trabajar de forma simultánea, comentar tareas, adjuntar archivos, asignar responsables y establecer fechas de vencimiento, promoviendo una comunicación fluida y centralizada.
- Adaptabilidad al proyecto: Trello se adapta a las necesidades específicas del desarrollo web, ya que permite incorporar etiquetas de colores para clasificar tareas (por ejemplo, por prioridad o tipo) o listas de verificación para etapas de desarrollo (como validación de requisitos, pruebas de funcionalidad y revisión de código).

- Transparencia y visibilidad del flujo de trabajo: Gracias a su estructura visual, Trello permite identificar cuellos de botella, priorizar tareas críticas y mantener una visión global del estado del proyecto, lo cual es clave para garantizar una entrega oportuna y de calidad.
- Escalabilidad e integraciones: Si bien Trello es ideal para equipos pequeños y medianos, también puede integrarse con herramientas complementarias como, GitHub, Google Drive o Figma, lo que amplía sus capacidades y lo hace útil en proyectos de mayor complejidad.

En resumen, Trello se consolidó como una herramienta clave durante el desarrollo web del sistema académico, al permitir una gestión ágil, colaborativa y visual del proyecto. Su implementación contribuyó significativamente a mejorar la organización de tareas, la coordinación del equipo y la eficiencia general del proceso de desarrollo.

Para la elaboración del tablero en Trello, se creó un proyecto dentro de la plataforma con el nombre “SIA ESFAP MUA”. En este espacio se diseñó la estructura en base al flujo de trabajo definido. Este diseño permitió una visualización clara de las tareas, la limitación del trabajo en curso (WIP). La Figura 6 muestra la organización del tablero implementado, donde se evidencia la distribución de las columnas y la manera en que las tareas fueron asignadas y gestionadas.

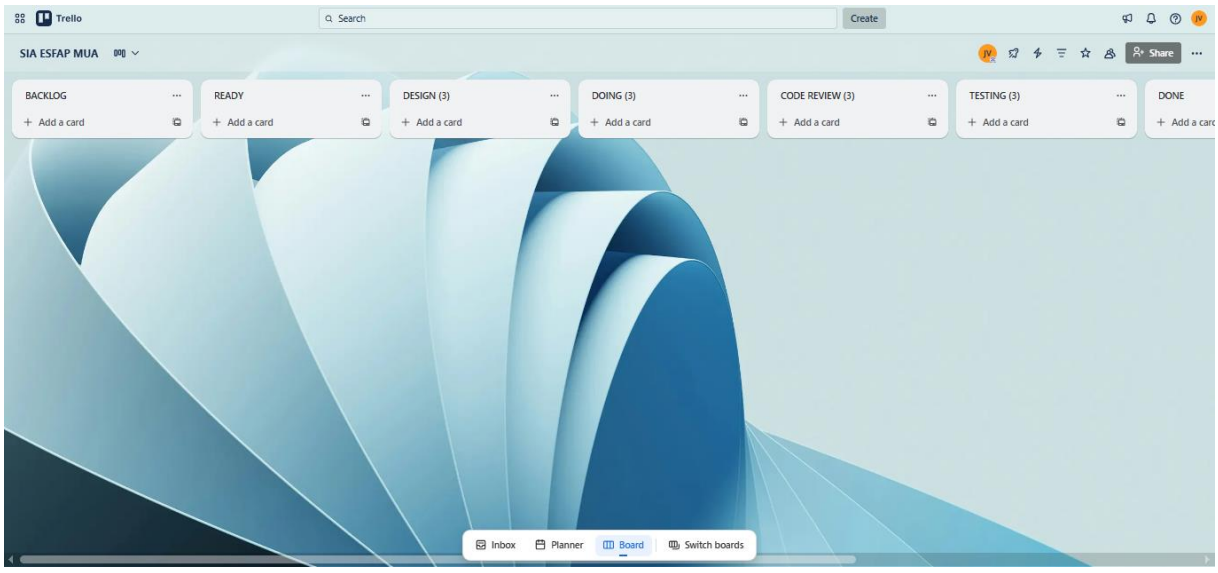

Fig. 6 Tablero de trabajo

3.1.3 APLICACIÓN DEL TABLERO KANBAN

3.1.3.1 PRODUCT BACKLOG

A continuación, en la Tabla II se presenta el product backlog de los requerimientos funcionales, el cual incluye la descripción de las historias de usuario (HU) junto con sus respectivas prioridades. Si bien en la metodología Kanban la priorización es opcional, en este caso se consideró necesaria con el fin de establecer un orden en la ejecución de las tareas. Las HU fueron elaboradas a partir de los requerimientos funcionales identificados en la fase de análisis, transformando cada necesidad del sistema en una narrativa centrada en el usuario. Para ello se empleó la estructura: "Como [tipo de usuario], quiero [objetivo o acción] para [beneficio esperado]", lo que permitió vincular directamente los requerimientos técnicos con el valor que aportan a los actores involucrados.

La utilización de historias de usuario resulta fundamental porque favorece la comunicación entre el equipo de desarrollo y los usuarios finales, al expresar los requerimientos en un lenguaje simple y comprensible. Asimismo, permiten mantener la trazabilidad entre los objetivos del sistema y su implementación, garantizan que las funcionalidades desarrolladas respondan a necesidades reales y contribuyen a una mejor priorización y planificación del trabajo dentro del tablero Kanban. En la Fig. 7 se muestran los atributos que debe tener una historia de usuario.

Fig. 7 Atributos de una historia de usuario

Tabla II. Product backlog de requerimientos funcionales

ID	Historia de Usuario	Perfil	Prioridad	Estimación aprox.	Responsable
HU01	Iniciar sesión	Todos	Media	5 días	Jhan Villanueva
HU02	Gestionar estudiantes	Administrado	Alta	5 días	Jhan Villanueva
HU03	Gestionar docentes	Administrado	Alta	5 días	Jhan Villanueva
HU04	Gestionar periodos	Administrado	Alta	4 días	Jhan Villanueva
HU05	Gestionar matrículas	Administrado	Media	7 días	Equipo de desarrollo
HU06	Filtrar y visualizar notas de cada estudiante	Administrado	Baja	4 días	Jhan Villanueva
HU07	Gestionar jefes de departamento	Soporte	Alta	4 días	Jhan Villanueva
HU08	Gestionar cursos	Soporte	Alta	7 días	Jhan Villanueva
HU09	Gestionar usuarios del sistema	Soporte	Media	5 días	Jhan Villanueva
HU10	Gestionar cursos a calificar	Jefe Dpto.	Alta	7 días	Equipo de desarrollo
HU11	Gestionar sesiones de clases	Docente	Media	5 días	Equipo de desarrollo
HU12	Gestionar desempeño de estudiantes	Docente	Media	5 días	Equipo de desarrollo
HU13	Gestionar actividades por cada sesión de clase	Docente	Media	5 días	Equipo de desarrollo
HU14	Generar acta de notas por cada curso	Docente	Baja	4 días	Jhan Villanueva

ID	Historia de Usuario	Perfil	Prioridad	Estimación aprox.	Responsable
HU15	Visualizar horario de clases	Docente	Baja	4 días	Jhan Villanueva
HU16	Gestionar pagos	Tesorería	Alta	5 días	Equipo de desarrollo
HU17	Generar matrícula	Estudiante	Alta	7 días	Equipo de desarrollo
HU18	Visualizar sesiones de clase, porcentaje de asistencia y calificaciones por cada curso	Estudiante	Media	5 días	Equipo de desarrollo
HU19	Visualizar y desarrollar actividades	Estudiante	Media	5 días	Equipo de desarrollo
HU20	Visualizar horario de clase	Estudiante	Baja	3 días	Jhan Villanueva
HU21	Visualizar historial de notas	Estudiante	Baja	4 días	Jhan Villanueva
HU22	Visualizar pagos	Estudiante	Media	3 días	Equipo de desarrollo

Cabe mencionar que las historias de usuario clasificadas con prioridad baja manifestaron que su desarrollo e implementación dependió del desarrollo previo de otra historia con prioridad media o alta, estas tareas fueron representadas dentro del tablero como tarjetas (ver Fig. 8), las cuales se fueron moviendo entre las columnas según fue el caso.

Seguidamente, en la Tabla III, se presenta el product backlog de los requerimientos no funcionales, en el cual también se asignaron prioridades. En este contexto, un requerimiento no funcional con prioridad "alta" indica que fue abordado antes del inicio de la codificación del sistema; una prioridad "media" significa que su implementación se dio paralelamente al desarrollo de las historias de usuario (HU); y una prioridad "baja" corresponde a funcionalidades ejecutadas después de completadas todas las HU, siendo particularmente relevantes durante el despliegue del sistema. Asimismo, según su modo de ejecución o validación, los requerimientos no funcionales fueron agrupados e identificados mediante colores.

Estos colores se utilizaron como etiquetas (labels) en las tarjetas del tablero Kanban. Cuando una historia de usuario alcanzaba la etapa de "Testing", además de evaluarse con base en sus criterios de aceptación, se verificaba también el cumplimiento de los requerimientos no funcionales correspondientes, según la leyenda de etiquetas (Ver Fig. 9).

Tabla III Product backlog de requerimientos no funcionales

ID	Tarea	Prioridad	Descripción	Modo de ejecución/validación
RNF01	Identificar y seleccionar las tecnologías adecuadas para el desarrollo y mantenimiento a largo plazo (por ejemplo, frameworks, lenguajes, herramientas de desarrollo).	Alta	Asegurarse de que el equipo de desarrollo seleccione herramientas y tecnologías escalables y con buen soporte a largo plazo.	Pre - desarrollo
RNF02	Seleccionar una base de datos SQL (MySQL, PostgreSQL, SQL Server, etc.) y configurar el entorno de base de datos.	Alta	Definir la base de datos adecuada para el sistema, crear esquemas y estructuras necesarias.	Pre - desarrollo
RNF03	Implementar diseño responsivo para garantizar la accesibilidad en dispositivos móviles y de escritorio.	Media	Crear una interfaz adaptable que se ajuste correctamente a diferentes tamaños de pantalla.	Por cada Historia de Usuario
RNF04	Implementar técnicas de cifrado (como bcrypt o hashing) para almacenar contraseñas de forma segura.	Media	Asegurarse de que todos los datos sensibles estén cifrados y protegidos adecuadamente.	Según lo requiera la Historia de Usuario
RNF05	Configurar el servidor físico para el despliegue del sistema, incluyendo las instalaciones de software necesarias.	Baja	Preparar el servidor físico para alojar el sistema, asegurando su correcto funcionamiento y seguridad.	Post - desarrollo
RNF06	Configurar el sistema para que se ejecute en el dominio institucional, incluyendo la configuración DNS.	Baja	Asegurarse de que el sistema esté disponible bajo el dominio institucional sin necesidad de modificar la URL.	Post - desarrollo
RNF07	Optimizar las API para garantizar tiempos de respuesta inferiores a 200 ms en condiciones normales de uso.	Alta	Implementar y probar medidas de optimización, como caché, balanceo de carga, y eficiencia en las consultas de base de datos.	Por cada Historia de Usuario
RNF08	Utilizar una paleta de colores que sea coherente con la identidad visual de la institución y aplicarla en la interfaz del sistema.	Alta	Asegurarse de que la interfaz tenga una apariencia consistente con la marca institucional.	Por cada Historia de Usuario

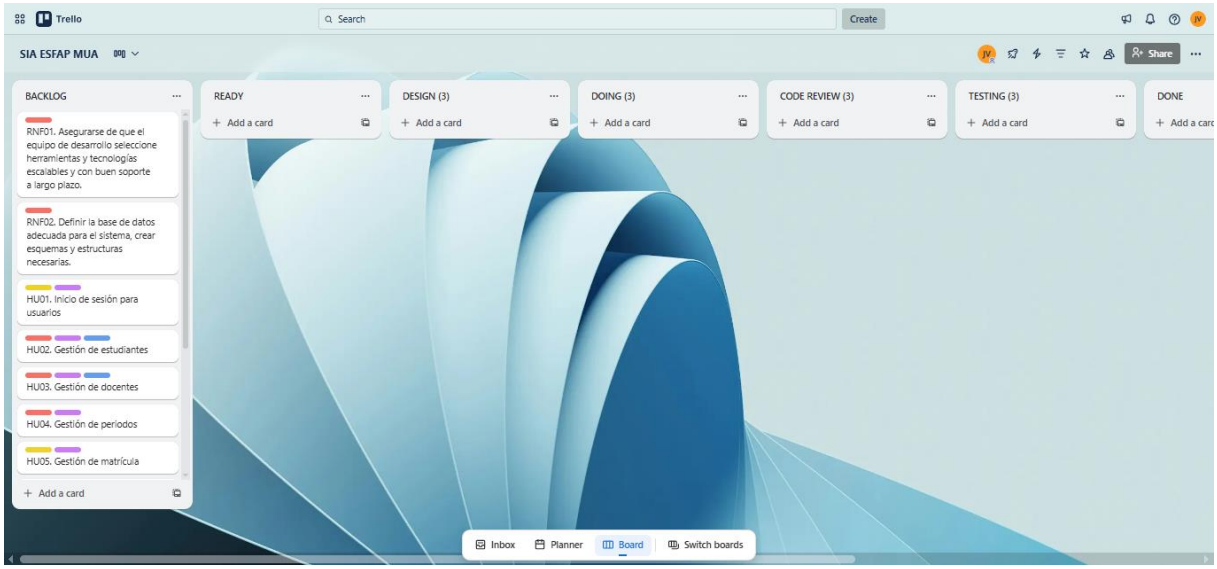

Fig. 8 Incorporación de tarjetas en el tablero Kanban

LEYENDA		
PRIORIDADES		Alta
		Media
		Baja
VALIDAR RNF		RNF03 - RNF07 - RNF08
		RNF04

Fig. 9 Leyenda de etiquetas en el tablero Kanban

3.1.3.2 DESARROLLO DEL PRODUCTO

A continuación, se detallan las actividades realizadas durante el desarrollo de las Historias de Usuario incluidas en el product backlog, derivadas tanto de los requerimientos funcionales (ver Tabla II) como de los no funcionales (ver Tabla III).

Para que una tarea del backlog pudiera trasladarse a la columna “Ready”, era requisito que la Historia de Usuario estuviera previamente detallada. En el caso de las Historias de Usuario vinculadas a los requerimientos funcionales, estas avanzaban a la columna “Design”, donde se elaboraban prototipos de interfaz empleando la herramienta Figma. Posteriormente, en la columna “Doing”, se ejecutaban las tareas correspondientes; para las Historias de Usuario de requerimientos funcionales, esta etapa incluía el diseño del diagrama de clases, así como la codificación en los módulos de backend y frontend. Cuando era necesaria la validación del código por otro desarrollador, la tarea pasaba a la columna “Code Review”. A continuación, en la columna “Testing”, se llevaban a cabo pruebas de caja negra basadas en los criterios de aceptación establecidos para cada Historia de Usuario, además de la validación de los

requerimientos no funcionales asociados. Finalmente, el flujo concluía en la columna “Done”, que indicaba la finalización y aceptación de la tarea.

En cuanto a las tareas relacionadas con los requerimientos no funcionales, se documentaron las actividades ejecutadas y, cuando fue necesario, se realizaron las pruebas correspondientes.

Estas acciones permitieron garantizar que tanto los aspectos funcionales como los no funcionales del sistema fueran desarrollados, verificados y validados de forma integral, asegurando así la calidad y el cumplimiento de los objetivos del proyecto.

RNF01. El sistema debe desarrollarse con tecnologías que permitan su mantenimiento futuro.

El sistema web académico que se ha desarrollado ha sido construido utilizando tecnologías modernas y escalables que permiten un rendimiento óptimo y una experiencia de usuario fluida. En el backend, se ha optado por Node.js, un entorno de ejecución altamente eficiente, acompañado de Express como framework para la creación de APIs RESTful, lo que facilita la implementación de las funcionalidades requeridas.

Para la gestión de la base de datos, Sequelize se ha integrado como un ORM que simplifica la interacción con las bases de datos relacionales, utilizando JavaScript como lenguaje principal de desarrollo.

En el frontend, se ha utilizado Next.js, un framework de React que permite la renderización del lado del servidor, lo que mejora el SEO y la carga inicial de la aplicación. Tailwind CSS se ha implementado para el diseño de la interfaz de usuario, permitiendo un desarrollo ágil y la creación de una experiencia visual atractiva y responsive. Además, TypeScript se ha adoptado para agregar robustez y mejorar la mantenibilidad del código, proporcionando tipado estático y mejor autocompletado. La integración de AJAX permite realizar solicitudes asíncronas de datos, optimizando la interacción del usuario sin necesidad de recargar la página, lo que mejora la fluidez en el uso de la aplicación.

Indicadores kanban:

- Lead Time: 1d

- Cycle Time: 4h

- Flow Efficiency: FE = 3.8/4 * 100 = 95

Estas tecnologías combinadas han resultado en una solución sólida, eficiente y de alto rendimiento para el sistema web académico. En la Fig. 10 se muestra la arquitectura del backend utilizada, así como su relación con el cliente y la base de datos.

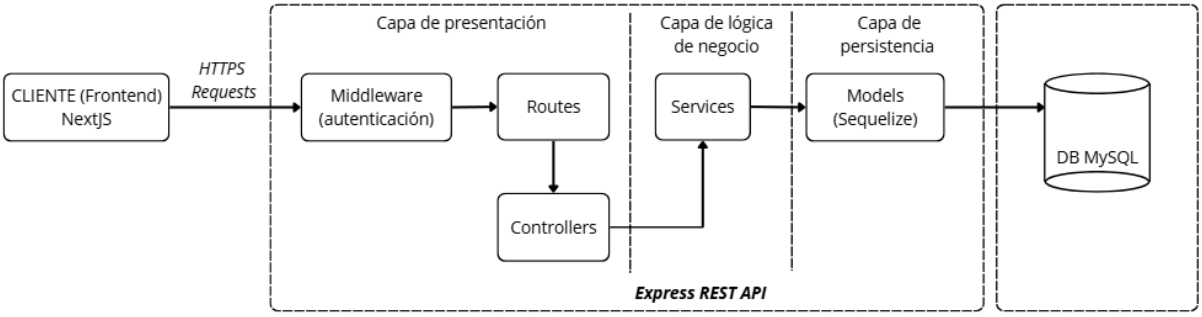

Fig. 10 Arquitectura backend

- **Cliente:** El cliente (aplicación web o frontend) interactúa con el backend mediante solicitudes HTTP/HTTPS, enviando peticiones a los endpoints API y recibiendo respuestas estructuradas en JSON.
- **Middleware:** Capa intermedia que procesa las solicitudes antes de llegar a los controladores. Incluye autenticación JWT, validación de datos, seguridad y logging. Actúa como filtro para garantizar la integridad y seguridad de las peticiones.
- **Routes**: Sistema de enrutamiento que define los endpoints de la API y asocia cada ruta con su controlador correspondiente. Organiza las URL de la aplicación y dirige las solicitudes a la lógica de negocio apropiada.
- **Controller**: Actúa como intermediario entre las solicitudes HTTP y la lógica de negocio. Su responsabilidad principal es recibir las peticiones del cliente, validar los parámetros de entrada, invocar a los servicios correspondientes y formular la respuesta HTTP adecuada (éxito, error, etc.).
- **Services:** Capa que contiene el núcleo de la lógica de negocio de la aplicación. Los servicios orquestan las operaciones complejas, coordinando múltiples modelos, aplicando reglas específicas del dominio y gestionando transacciones.
- **Model (Sequelize ORM):** Representa la capa de acceso a datos y de mapeo objeto-relacional (ORM). Implementados con Sequelize, los modelos definen la estructura de las entidades (tablas) en la base de datos, incluyendo sus atributos, validaciones y asociaciones. Su responsabilidad se limita a proporcionar una interfaz en JavaScript para interactuar con la base de datos sin necesidad de escribir SQL manualmente.

- **Base de Datos MySQL:** Sistema de gestión de bases de datos relacional que almacena persistentemente la información de la aplicación. Almacena tablas, relaciones, y ejecuta las operaciones solicitadas a través de Sequelize ORM.

El flujo de procesamiento sigue una secuencia bien definida a través de las capas arquitectónicas. En la capa de presentación, los controladores reciben y validan las solicitudes HTTP. Posteriormente, derivan la ejecución a la capa de servicios, donde reside la lógica de negocio esencial. Esta capa media aplica las reglas del dominio y coordina los recursos necesarios. Para las operaciones de datos, los servicios consumen la capa de persistencia, implementada con Sequelize ORM, que abstrae la complejidad del motor de base de datos MySQL. Esta organización promueve un diseño cohesivo donde cada capa se especializa en una función particular, optimizando la claridad y robustez del sistema.

RNF02. Se debe utilizar una base de datos SQL

Para la elección de la tecnología de base de datos del sistema académico, se inició realizando una comparativa exhaustiva entre las principales alternativas del mercado, centrándonos en MySQL, PostgreSQL y Microsoft SQL Server, la cual se muestra a continuación:

Tabla IV Comparativa técnica entre bases de datos

Característica	MySQL	PostgreSQL	Microsoft SQL Server
Rendimiento en lecturas	Excelente y fiable para cargas web típicas.	Muy bueno, aunque su optimización puede resultar más compleja.	Aceptable, pero mejor: "depende considerablemente de la edición y su configuración.
Simplicidad y usabilidad	Fácil de instalar, configurar y gestionar. Herramientas consolidadas (Workbench, phpMyAdmin).	Potente, pero con una curva de aprendizaje más pronunciada.	Excelentes herramientas (SSMS), aunque la plataforma en general presenta una mayor complejidad.
Ecosistema y comunidad	Comunidad masiva y documentación extensa. Soporte amplio en todos los lenguajes de programación y frameworks de desarrollo.	Comunidad sólida y muy técnica, pero más pequeña.	Comunidad grande centrada en el ecosistema Microsoft.
Seguridad	Buen nivel de seguridad: cifrado de conexiones, roles, privilegios.	Robusta: autenticación flexible (LDAP, GSSAPI, etc.), cifrado de datos nativo, seguridad a nivel de fila (row-level security).	Excelente seguridad integrada con Active Directory, transparent data encryption (en ediciones pagas), auditing.

Característica	MySQL	PostgreSQL	Microsoft SQL Server
Adopción en la web	Una de las DBMS más usadas en el mundo, lo que avala su estabilidad.	Adopción en crecimiento, especialmente en aplicaciones modernas.	Fuera del stack Microsoft, su adopción en entornos web es menor.
Costo total	Sin costo (versión Community). Ideal para presupuestos académicos.	Gratuito (Open Source puro).	Costo muy alto; la versión gratuita es limitada.

Tras sopesar las ventajas y desventajas de cada opción en el contexto específico de los requisitos académicos, que priorizan la fiabilidad y un ecosistema robusto de herramientas, se seleccionó MySQL como la solución óptima. MySQL es una base de datos extremadamente probada, con un rendimiento excelente para cargas de trabajo web típicas como las que manejará el sistema, garantizando una alta velocidad de respuesta en las operaciones de consulta y gestión de datos de estudiantes, cursos y calificaciones, etc. Su amplia adopción global asegura una gran cantidad de documentación, herramientas de gestión intuitivas como phpMyAdmin y MySQL Workbench, y una vasta comunidad de soporte, lo que facilita enormemente el desarrollo, la resolución de problemas y el encontrar talento con experiencia necesaria para su soporte fututo. Además, al ser open source en su versión Community, se ajusta perfectamente a los presupuestos típicos del ámbito académico, eliminando costos de licencias sin sacrificar confiabilidad o escalabilidad. Es una opción robusta, práctica y confiable para un proyecto crítico como este.

Indicadores kanban:

- Lead Time: 1d
- Cycle Time: 2h
- Flow Efficiency: FE = 1.45/2 * 100 = 75

HU02. Gestionar estudiantes

Tabla V HU02 "Gestionar estudiantes"

Historia de usuario	
Código: HU02	**Usuario**: Administrador académico
Nombre de la historia: Gestionar estudiantes	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: Ninguna	
Responsable: Jhan Villanueva	

Descripción: Como administrador, quiero crear, listar, editar y eliminar estudiantes, para gestionar la base estudiantil.
Criterios de aceptación: - El sistema permite agregar nuevos estudiantes a través de un formulario con campos validados - El sistema permite editar la información de los estudiantes registrados - El sistema permite eliminar (deshabilitar) estudiantes

Seguidamente se realizó el prototipo de la interfaz, esto permitió validar rápidamente la comprensión de la historia con stakeholders, incluso antes de definir aspectos técnicos. Para diseñar el prototipo de interfaz se decidió utilizar Figma (ver Fig. 11).

Fig. 11 Prototipo de la interfaz "Gestionar estudiantes"

Luego de validar el prototipo se procedió a realizar el diagrama de clases correspondiente, esto para analizar cómo implementar técnicamente la funcionalidad representada en el prototipo, definiendo estructuras de datos, relaciones entre entidades, etc. (ver Fig. 12).

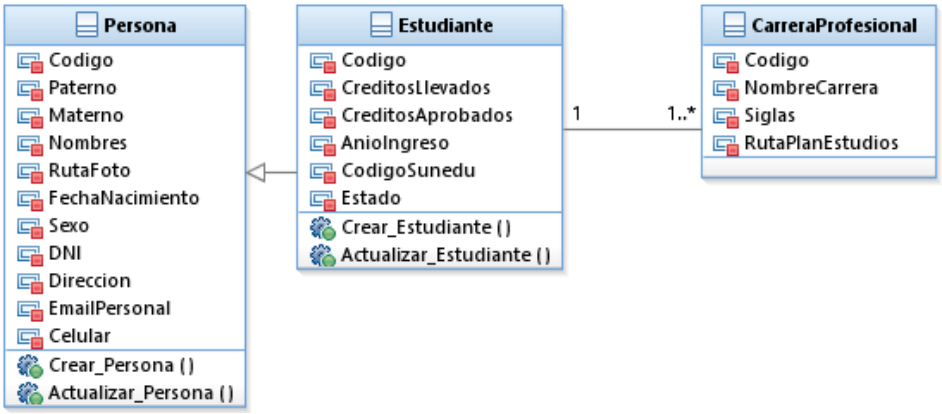

Fig. 12 Diagrama de clases "Gestionar estudiantes"

Para la codificación del sistema se utilizó el patrón de diseño arquitectónico MVC (Modelo – Vista - Controlador), que organiza una aplicación en tres capas interconectadas, los **modelos** son responsable de acceder a la base de datos, procesar datos y aplicar las reglas de negocio, serán representaciones de las entidades de la base de datos (ver Fig.13, 14 y 15).

estudiante.model.js

server > models > estudiante.model.js > ...

```
const { DataTypes } = require('sequelize');
const { sequelize } = require('../config/database');

const Estudiante = sequelize.define('Estudiante', {
  Codigo:
  {
    type: DataTypes.INTEGER,
    primaryKey: true,
    autoIncrement: true
  },
  CreditosLlevados: DataTypes.STRING(20),
  CreditosAprobados: DataTypes.STRING(20),
  AnioIngreso: DataTypes.STRING(4),
  CodigoSunedu: DataTypes.STRING(10),
  Estado: DataTypes.BOOLEAN,
  CodigoPersona: DataTypes.INTEGER,
  CodigoCarreraProfesional: DataTypes.INTEGER,
});

module.exports = Estudiante;
```

Fig. 13 Clase del modelo "estudiante"

persona.model.js

server > models > persona.model.js > ...

```
const { DataTypes } = require('sequelize');
const { sequelize } = require('../config/database');

const Persona = sequelize.define('Persona', {
  Codigo:
  {
    type: DataTypes.INTEGER,
    primaryKey: true,
    autoIncrement: true
  },
  Paterno: DataTypes.STRING(20),
  Materno: DataTypes.STRING(20),
  Nombres: DataTypes.STRING(40),
  RutaFoto: DataTypes.STRING(100),
  FechaNacimiento: DataTypes.DATE,
  Sexo: DataTypes.CHAR(1),
  DNI: {
    type: DataTypes.CHAR(8),
    unique: true
  },
  Direccion: DataTypes.STRING(100),
  EmailPersonal: DataTypes.STRING(100),
  Celular: DataTypes.CHAR(9)
});

module.exports = Persona;
```

Fig. 14 Clase del modelo "persona"

carreraProfesional.model.js

server > models > carreraProfesional.model.js > ...

```
1  const { DataTypes } = require('sequelize');
2  const { sequelize } = require('../config/database');
3  const JefeDepartamento = require('./jefeDepartamento.model');
4
5  const CarreraProfesional = sequelize.define('CarreraProfesional', {
6      Codigo:
7      {
8        type: DataTypes.INTEGER,
9        primaryKey: true,
10       autoIncrement: true
11     },
12     NombreCarrera: DataTypes.STRING(60),
13     Siglas: DataTypes.CHAR(2),
14     RutaPlanEstudios: DataTypes.STRING(45),
15     CodigoJefeDepartamento: DataTypes.INTEGER
16   });
17
18   CarreraProfesional.belongsTo(JefeDepartamento, {foreignKey: 'CodigoJefeDepartamento'})
19
20 module.exports = CarreraProfesional;
```

Fig. 15 Clase del modelo "carreraProfesional"

Los **controladores** son las clases intermedias entre la vista y el modelo, recibe las acciones del usuario, procesa las solicitudes, interactúa con el modelo y decide qué vista mostrarse (ver Fig.16 y 17).

estudiante.controller.js

server > controllers > estudiante.controller.js > crearEstudiante

```
177  const crearEstudiante = async (req, res) => {
178    try {
179      let _persona = await Persona.findOne({ where: { DNI: req.body.DNI } });
180      if (_persona) {
181        return res.status(403).json({ error: "El DNI ya existe" });
182      }
183
184      let _usuario = await Usuario.findOne({ where: { Email: req.body.Email } });
185      if (_usuario) {
186        return res.status(403).json({ error: "El email ya existe" });
187      }
188
189      let persona = null;
190      let estudiante = null;
191      let usuario = null;
192
193      await sequelize.transaction(async (t) => {
194        persona = await Persona.create(
195          {
196            Codigo: null,
197            Paterno: req.body.Paterno,
198            Materno: req.body.Materno,
199            Nombres: req.body.Nombres,
200            RutaFoto: req.body.RutaFoto,
201            FechaNacimiento: req.body.FechaNacimiento,
202            Sexo: req.body.Sexo,
203            DNI: req.body.DNI,
204            Direccion: req.body.Direccion,
205            EmailPersonal: req.body.EmailPersonal,
206            Celular: req.body.Celular,
207          },
208          { transaction: t }
209        );
```

Fig. 16 Clase del controlador "estudiante"

persona.controller.js

server > controllers > persona.controller.js > ...

```
const { Persona} = require("../config/relations")

const crearPersona = async (req, res = response) => {
    try {

        const paterno = req.body.paterno;
        const materno = req.body.materno;
        const nombres = req.body.nombres;
        const rutaFoto = req.body.rutaFoto;
        const fechaNacimiento = req.body.fechaNacimiento;
        const sexo = req.body.sexo;
        const DNI = req.body.DNI;
        const correo = req.body.correo;

        const persona = new Persona(req.body);

        await persona.save();

        res.json({
            ok: true,
            persona
        });

    } catch (error) {
        console.log(error);
        res.status(500).json({
            ok: false,
            msg: ''
        });
    }
}
```

Fig. 17 Clase del controlador "persona"

Las **vistas** son las encargadas de mostrar los datos al usuario, define cómo se presentará visualmente la información (ver Fig. 18, 19 y 20).

page.tsx

client > app > administrador > estudiantes > page.tsx > Page

```
export default function Page() {
    return (
        <div className="grid crud-demo">
            <div className="col-12">
                <div className="card">
                    <Toast ref={toast} />
                    <Toolbar className="mb-4" left={leftToolbarTemplate} right={rightToolbarTemplate}></Toolbar>
                    <DataTable
                        ref={dt}
                        value={estudiantes}
                        dataKey="Codigo"
                        paginator
                        rows={10}
                        rowsPerPageOptions={[5, 10, 25]}
                        className="datatable-responsive"
                        paginatorTemplate="FirstPageLink PrevPageLink PageLinks NextPageLink LastPageLink CurrentPageReport RowsPerPageDropdown"
                        currentPageReportTemplate="Mostrando de {first} a {last} de {totalRecords} estudiantes"
                        globalFilter={globalFilter}
                        emptyMessage="Sin estudiantes registrados"
                        header={header}
                    >
                        <Column field="CodigoSunedu" header="COD" sortable />
                        <Column field="CarreraProfesional.NombreCarrera" header="Carrera" sortable />
                        <Column field="Persona.DNI" header="DNI" sortable />
                        <Column field="Persona.Paterno" header="Paterno" sortable />
                        <Column field="Persona.Materno" header="Materno" sortable />
                        <Column field="Persona.Nombres" header="Nombres" sortable />
                        <Column field="AnioIngreso" header="Ingreso" sortable />
                        <Column field="Estado" body={statusBodyTemplate} header="Estado" sortable align='center' />
                        <Column body={actionBodyTemplate} headerStyle={{ minWidth: '10rem' }}></Column>
                    </DataTable>
```

Fig. 18 Código de la vista "Gestión de estudiantes"

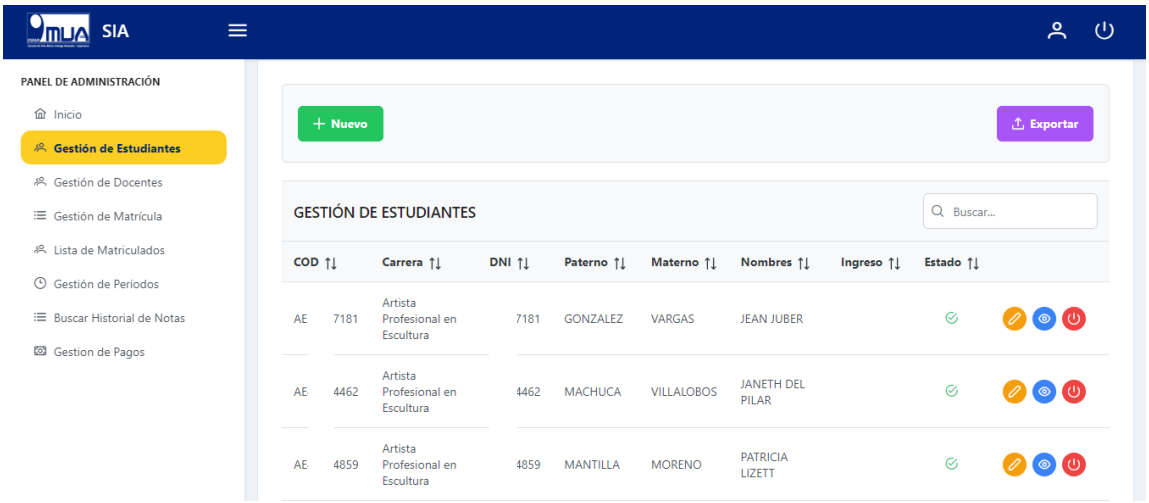

Fig. 19 Vista "Gestionar estudiantes"

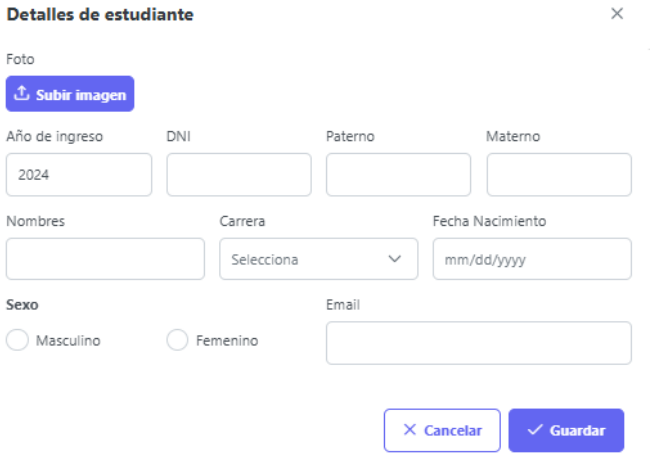

Fig. 20 Formulario de datos "Gestionar estudiantes"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla VI Prueba de caja negra "Gestionar estudiantes"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU02	El sistema permite agregar nuevos estudiantes a través de un formulario con campos validados	Datos almacenados correctamente respetando las validaciones	Exitoso
	El sistema permite editar la información de los estudiantes registrados	Datos de los cursos editados correctamente	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	El sistema permite eliminar (deshabilitar) estudiantes	El estado de los estudiantes puede ser modificado entre "habilitado" y "deshabilitado" y el cambio se muestra de manera visual	Exitoso

Indicadores kanban:

- Lead Time: 6d
- Cycle Time: 5.5d
- Flow Efficiency: FE = 5.5/6 * 100 = 91

En esta Historia de Usuario se mostró a mayor detalle las actividades presentes durante el paso de las actividades por las columnas "Ready", "Design", "Doing" y "Testing" del tablero Kanban, las siguientes Historias de Usuario serán documentadas a menor detalle para no cubrir el presente documento con información redundante que puede interpretarse a partir del ejemplo anteriormente detallado.

HU03. Gestionar docentes

Tabla VII HU3 "Gestionar docentes"

Historia de usuario	
Código: HU03	**Usuario**: Administrador académico
Nombre de la historia: Gestionar docentes	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: Ninguna	
Responsable: Jhan Villanueva	
Descripción: Como administrador, quiero gestionar docentes (crear, listar, editar, eliminar), para mantener su información actualizada.	
Criterios de aceptación: - El sistema debe permitir agregar nuevos docentes a través de un formulario con campos validados - El sistema debe permitir editar la información de los docentes registrados - El sistema debe permitir eliminar (deshabilitar) docentes	

Luego de detallar la HU, se realizó el prototipo de la interfaz, para lo cual se decidió utilizar Figma (ver Fig. 21).

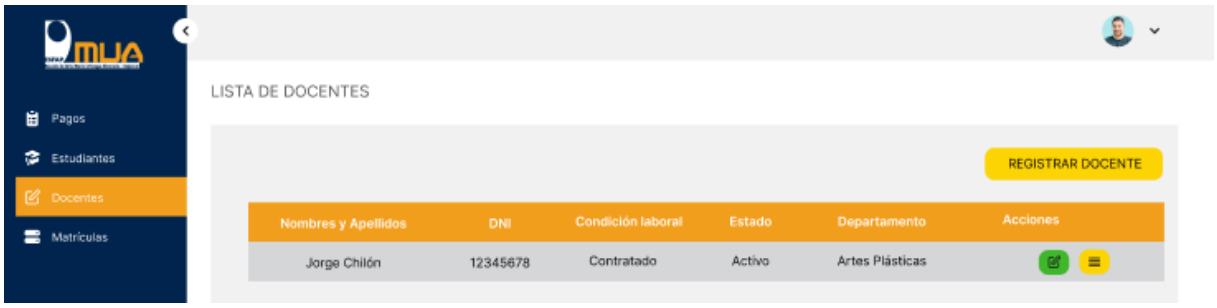

Fig. 21 Prototipo de la interfaz "Gestionar docentes"

Posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 22).

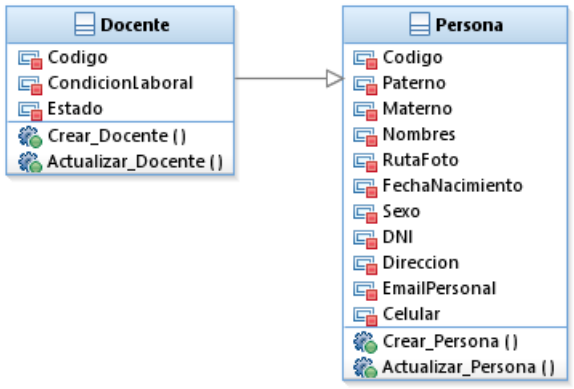

Fig. 22 Diagrama de clases "Gestionar docentes"

Más adelante se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondientes, teniendo como resultado las interfaces mostradas en las Fig. 23 y 24.

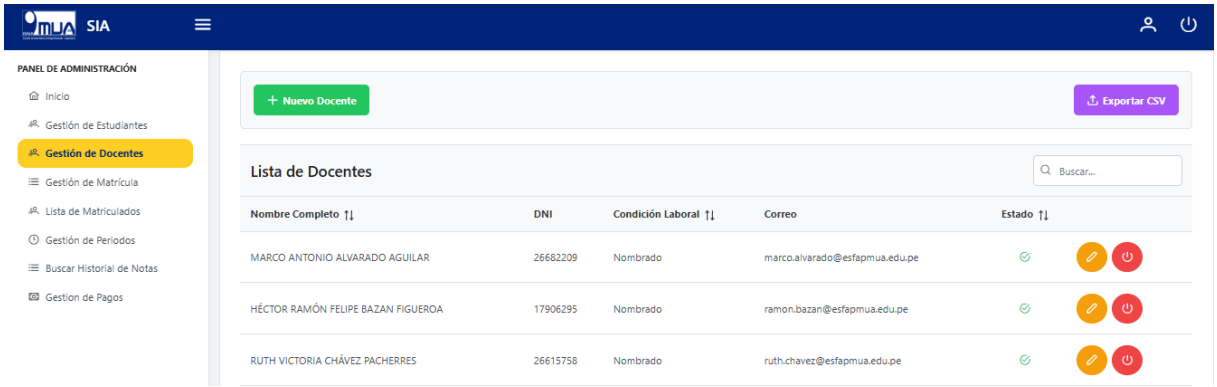

Fig. 23 Vista "Gestionar docentes"

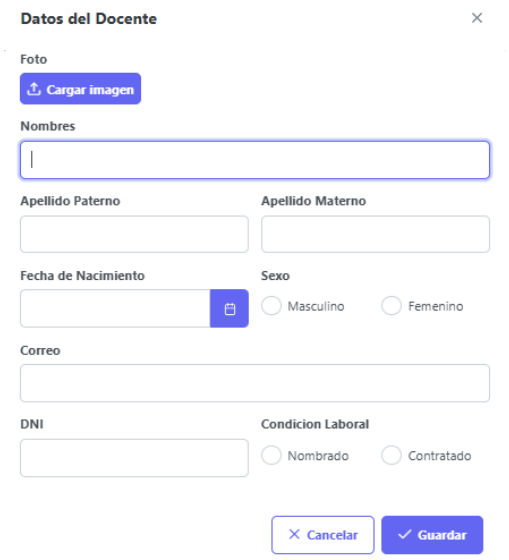

Fig. 24 Formulario de datos "Gestionar docentes"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla VIII Prueba de caja negra "Gestionar docentes"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU02	El sistema debe permitir agregar nuevos docentes a través de un formulario con campos validados	Datos almacenados correctamente respetando las validaciones	Exitoso
	El sistema debe permitir editar la información de los docentes registrados	Datos de los cursos editados correctamente	Exitoso
	El sistema permite eliminar (deshabilitar) docentes	El estado de los docentes puede ser modificado entre "habilitado" y "deshabilitado" y el cambio se muestra de manera visual	Exitoso

Indicadores kanban:

- Lead Time: 5d
- Cycle Time: 4.5d
- Flow Efficiency: FE = 4.5/5 * 100 = 90

HU04. Gestionar periodos

Tabla IX HU04 "Gestionar periodos"

Historia de usuario	
Código: HU04	**Usuario**: Administrador académico
Nombre de la historia: Gestionar periodos	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: Ninguna	
Responsable: Jhan Villanueva	
Descripción: Como administrador, quiero gestionar los periodos académicos, para organizar el calendario académico.	
Criterios de aceptación: - El sistema debe permitir agregar nuevos periodos a través de un formulario con campos validados - El sistema debe permitir editar la información de los periodos registrados - El sistema debe permitir eliminar periodos	

Luego de detallar la HU, se realizó el prototipo de la interfaz, para lo cual se decidió utilizar Figma (ver Fig. 25).

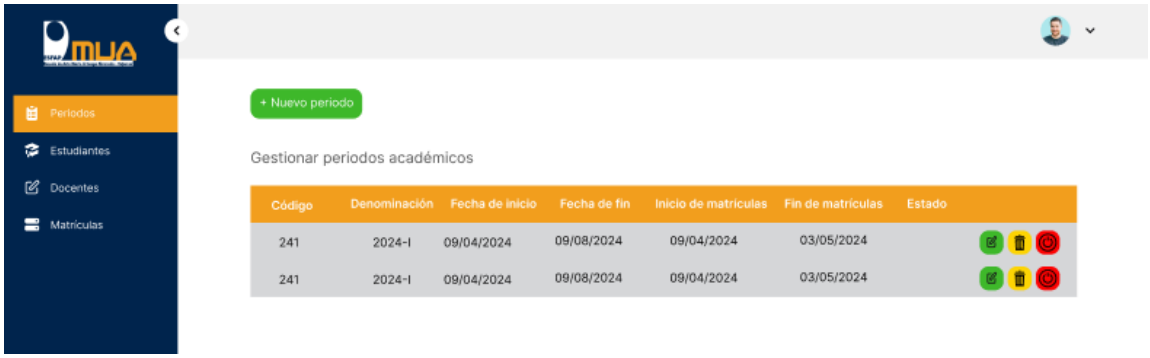

Fig. 25 Prototipo de la interfaz "Gestionar periodos"

Posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 26).

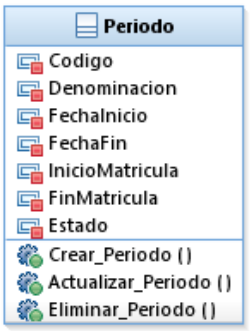

Fig. 26 Diagrama de clases "Gestionar periodos"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 27.

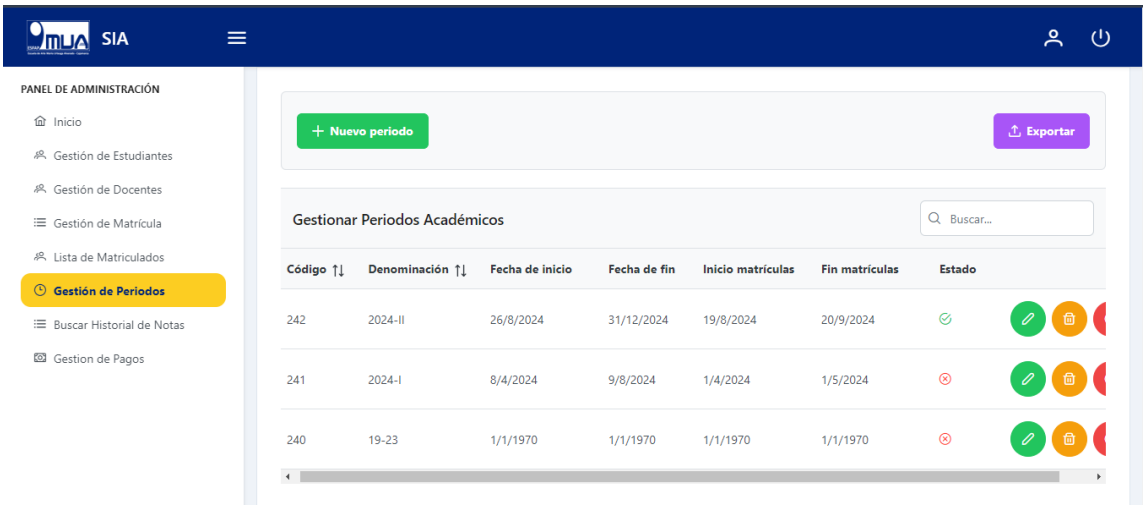

Fig. 27 Vista "Gestionar periodos"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios.

Tabla X Prueba de caja negra "Gestionar periodos"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU04	El sistema debe permitir agregar nuevos periodos a través de un formulario con campos validados	Datos almacenados correctamente respetando las validaciones	Exitoso
	El sistema debe permitir editar la información de los periodos registrados	Datos de un periodo editados correctamente	Exitoso
	El sistema debe permitir eliminar periodos siempre y cuando no tengan datos relacionados con otro registro en la base de datos	Periodo eliminado correctamente	Exitoso

Indicadores kanban:

- Lead Time: 4d
- Cycle Time: 3.5d
- Flow Efficiency: FE = 3.5/4 * 100 = 88

HU07. Gestionar jefes de departamento

Tabla XI HU07 "Gestionar jefes de departamento"

Historia de usuario	
Código: HU07	**Usuario**: Soporte
Nombre de la historia: Gestionar jefes de departamento	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: Ninguna	
Responsable: Jhan Villanueva	
Descripción: Como soporte, quiero gestionar jefes de departamento (crear, listar, editar, eliminar), para mantener su información actualizada.	
Criterios de aceptación: - El sistema debe permitir agregar nuevos jefes de departamento a través de un formulario con campos validados - El sistema debe permitir editar la información de los jefes de departamento registrados - El sistema debe permitir eliminar (deshabilitar) jefes de departamento	

Luego de detallar la HU, se realizó el prototipo de la interfaz, para lo cual se decidió utilizar Figma (ver Fig. 28).

Fig. 28 Prototipo de la interfaz "Gestionar jefes de departamento"

Posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 29).

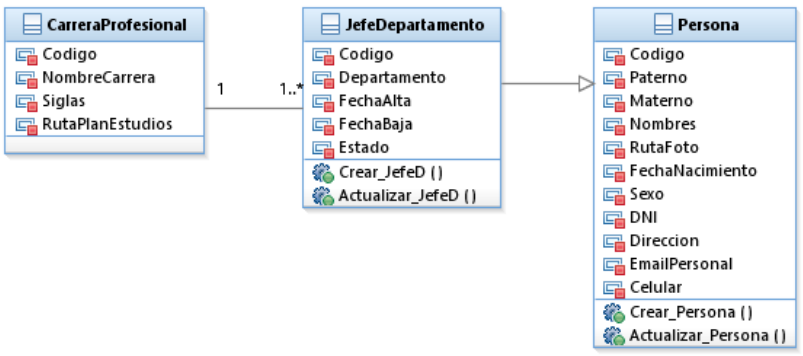

Fig. 29 Diagrama de clases "Gestionar jefes de departamento"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado las interfaces mostradas en las Fig. 30 y 31.

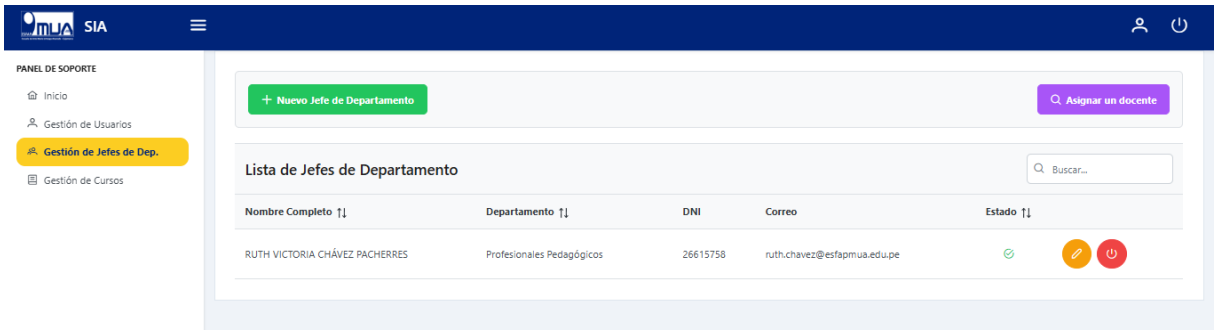

Fig. 30 Vista "Gestionar jefes de departamento"

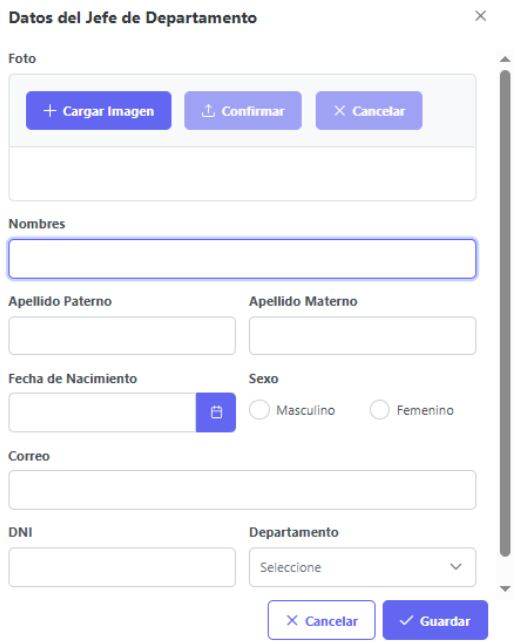

Fig. 31 Formulario de datos "Gestionar jefes de departamento"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XII Prueba de caja negra "Gestionar jefes de departamento"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU02	El sistema debe permitir agregar nuevos jefes de departamento a través de	Datos almacenados correctamente respetando las validaciones	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	un formulario con campos validados		
	El sistema debe permitir editar la información de los jefes de departamento registrados	Datos de los cursos editados correctamente	Exitoso
	El sistema debe permitir eliminar (deshabilitar) jefes de departamento	El estado de los estudiantes puede ser modificado entre "habilitado" y "deshabilitado" y el cambio se muestra de manera visual	Exitoso

Indicadores kanban:

- Lead Time: 4d
- Cycle Time: 3.5d
- Flow Efficiency: FE = 3.5/4 * 100 = 88

HU08. Gestionar cursos

Tabla XIII HU08 "Gestionar cursos"

Historia de usuario	
Código: HU08	**Usuario**: Soporte
Nombre de la historia: Gestionar cursos	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: Ninguna	
Responsable: Jhan Villanueva	
Descripción: Como soporte, quiero gestionar los cursos (crear, listar, editar, eliminar), para configurar el plan académico.	
Criterios de aceptación: - El sistema debe permitir agregar nuevos cursos a través de un formulario con campos validados - El sistema debe permitir editar la información de los cursos registrados - El sistema debe permitir eliminar (deshabilitar) cursos	

Luego de detallar la HU, se realizó el prototipo de la interfaz, para lo cual se decidió utilizar Figma (ver Fig. 32).

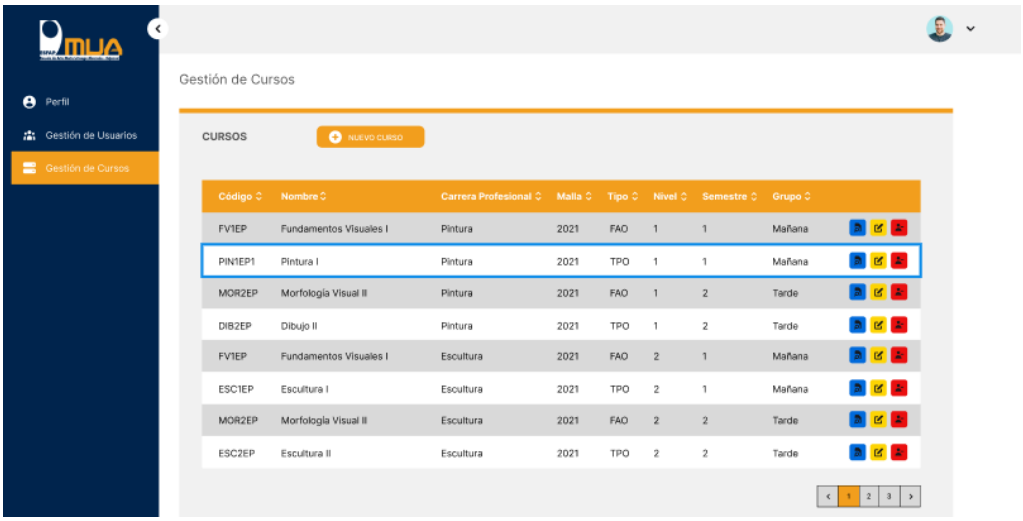

Fig. 32 Prototipo de la interfaz "Gestionar cursos"

Posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 33).

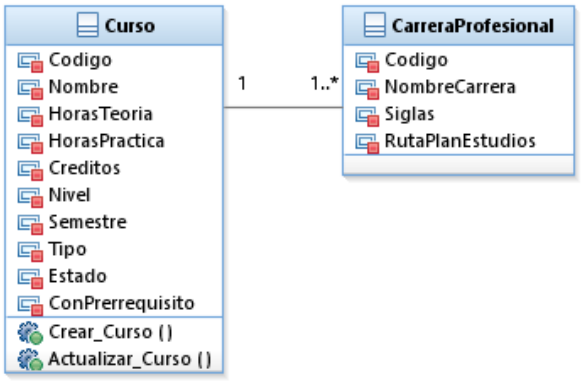

Fig. 33 Diagrama de clases "Gestionar cursos"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado las interfaces mostradas en las Fig. 34 y 35.

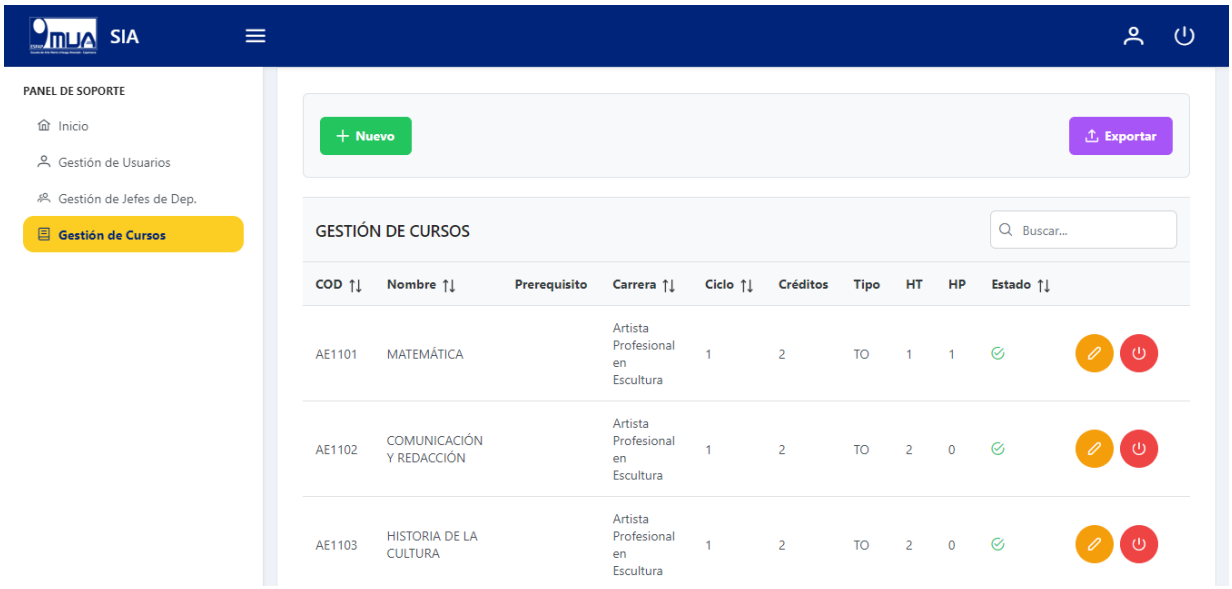

Fig. 34 Vista "Gestionar cursos"

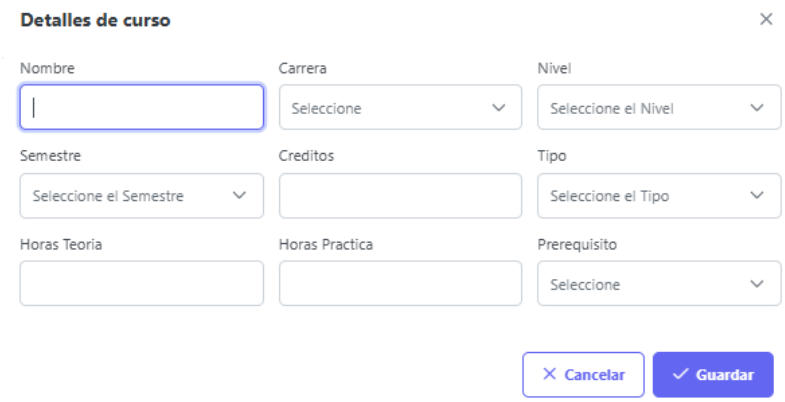

Fig. 35 Formulario de datos "Gestionar cursos"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XIV Prueba de caja negra "Gestionar cursos"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU08	El sistema debe permitir agregar nuevos cursos a través de un formulario con campos validados	Datos almacenados correctamente respetando las validaciones	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	El sistema debe permitir editar la información de los cursos registrados	Datos de los cursos editados correctamente	Exitoso
	El sistema debe permitir eliminar (deshabilitar) cursos	El estado de los cursos puede ser modificado entre "habilitado" y "deshabilitado" y el cambio se muestra de manera visual	Exitoso

HU10. Gestionar cursos a calificar

Tabla XV HU10 "Gestionar cursos a calificar"

Historia de usuario	
Código: HU10	**Usuario**: Jefe de Dpto.
Nombre de la historia: Gestionar cursos a calificar	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU07, HU08	
Responsable: Equipo de desarrollo	
Descripción: Como jefe de departamento, quiero gestionar cursos por dictar, asignar docentes y horarios, para planificar la enseñanza.	
Criterios de aceptación: - El sistema debe permitir a los jefes de departamento habilitar cursos a calificar según las carreras que les corresponde - El sistema debe permitir asignar un docente a cada curso a calificar - El sistema debe permitir reasignar un docente a cada curso a calificar - El sistema debe permitir asignar un horario a cada curso a calificar - El sistema debe permitir editar el horario asignado a un curso a calificar - El sistema debe permitir habilitar cursos a calificar en bloque, ya sea los cursos que pertenecen a periodo par o a periodo impar - El sistema debe permitir eliminar cursos a calificar - El sistema debe permitir habilitar y deshabilitar el ingreso de notas de los cursos a calificar - El sistema debe permitir habilitar y deshabilitar el ingreso de asistencias de los cursos a calificar	

Luego de detallar la HU, se realizó el prototipo de la interfaz, para lo cual se decidió utilizar Figma (ver Fig. 36).

Fig. 36 Prototipo de la interfaz "Gestionar cursos a calificar"

Posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 37).

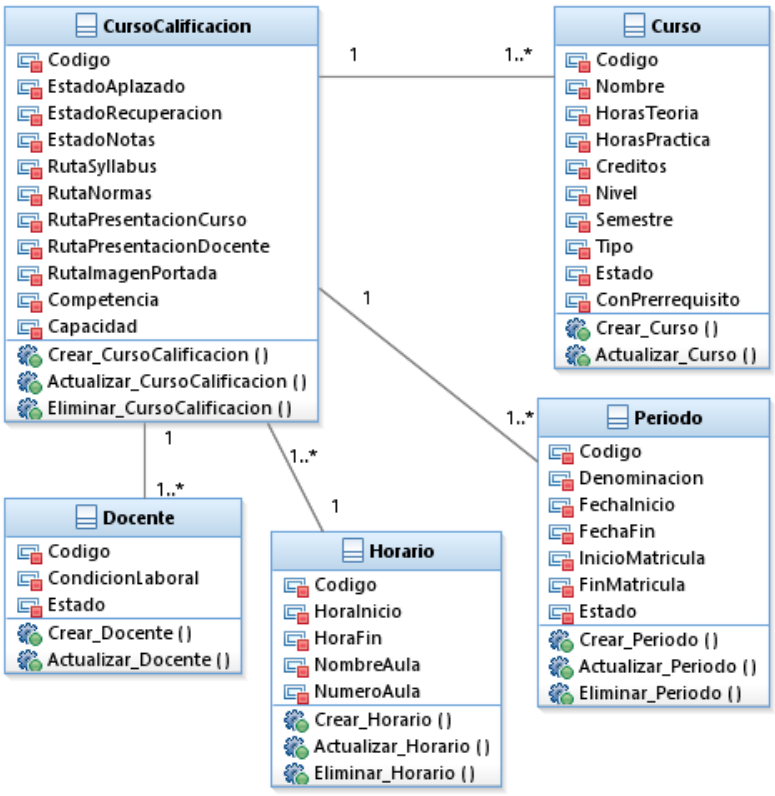

Fig. 37 Diagrama de clases "Gestionar cursos a calificar"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 38.

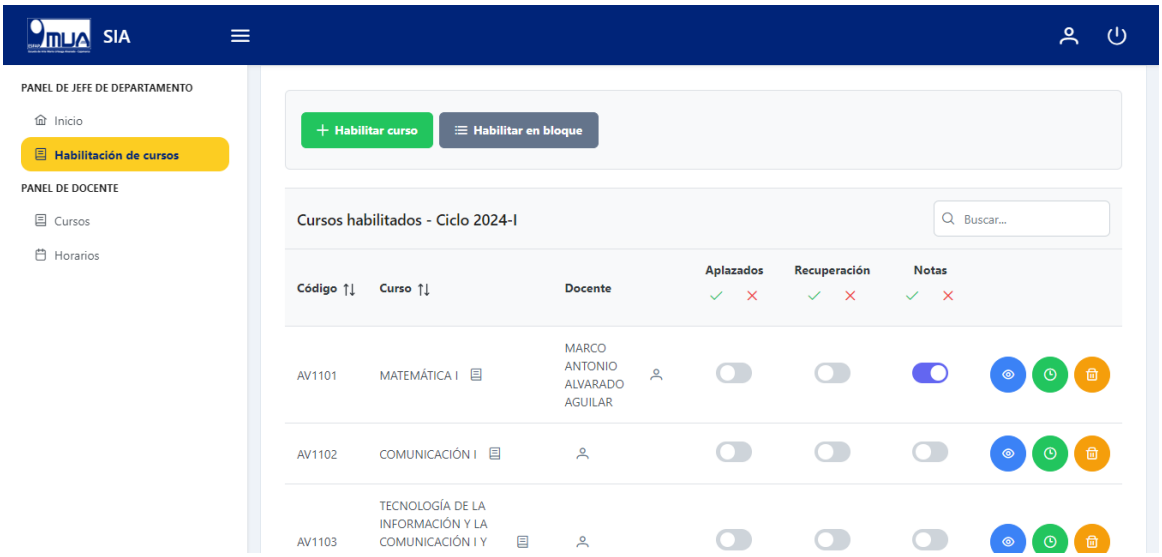

Fig. 38 Vista "Gestionar cursos a calificar"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XVI Prueba de caja negra "Gestionar cursos a calificar"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU08	El sistema debe permitir a los jefes de departamento habilitar cursos a calificar según las carreras que les corresponde	Los jefes de departamento solo pueden ver y habilitar los cursos a calificar que les corresponde	Exitoso
	El sistema debe permitir asignar un docente a cada curso a calificar	El docente es asignado correctamente a los cursos a calificar	Exitoso
	El sistema debe permitir reasignar un docente a cada curso a calificar	El docente es reasignado correctamente a los cursos a calificar	Exitoso
	El sistema debe permitir asignar un horario a cada curso a calificar	El horario es asignado correctamente a cada curso a calificar	Exitoso
	El sistema debe permitir editar el horario asignado a un curso a calificar	El horario es editado correctamente de cada curso a calificar	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	El sistema debe permitir habilitar cursos a calificar en bloque, ya sea los cursos que pertenecen a periodo par o a periodo impar	Los cursos pertenecientes son habilitados en bloque correctamente	Exitoso
	El sistema debe permitir eliminar cursos a calificar	Los cursos a calificar son eliminados correctamente	Exitoso
	El sistema debe permitir habilitar y deshabilitar el ingreso de notas de los cursos a calificar	El ingreso de notas se habilita y deshabilita correctamente	Exitoso
	El sistema debe permitir habilitar y deshabilitar el ingreso de asistencias de los cursos a calificar	El ingreso de asistencias se habilita y deshabilita correctamente	Exitoso

Indicadores kanban:

- Lead Time: 6d
- Cycle Time: 5.5d
- Flow Efficiency: FE = 5.5/6 * 100 = 92

HU16. Gestionar pagos

Tabla XVII HU16 "Gestionar pagos"

Historia de usuario	
Código: HU16	**Usuario**: Tesorería
Nombre de la historia: Gestionar pagos	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU02	
Responsable: Equipo de desarrollo	
Descripción: Como tesorería, quiero gestionar pagos (crear, listar, editar, eliminar), para llevar el control financiero.	
Criterios de aceptación: - El sistema debe permitir agregar nuevos pagos a través de un formulario con campos validados - El sistema debe permitir modificar el estado de los pagos registrados - El sistema debe permitir generar en formato PDF un comprobante del pago registrado, para luego poder ser descargado o imprimido	

Luego de detallar la HU, se realizó el prototipo de la interfaz, para lo cual se decidió utilizar Figma (ver Fig. 39).

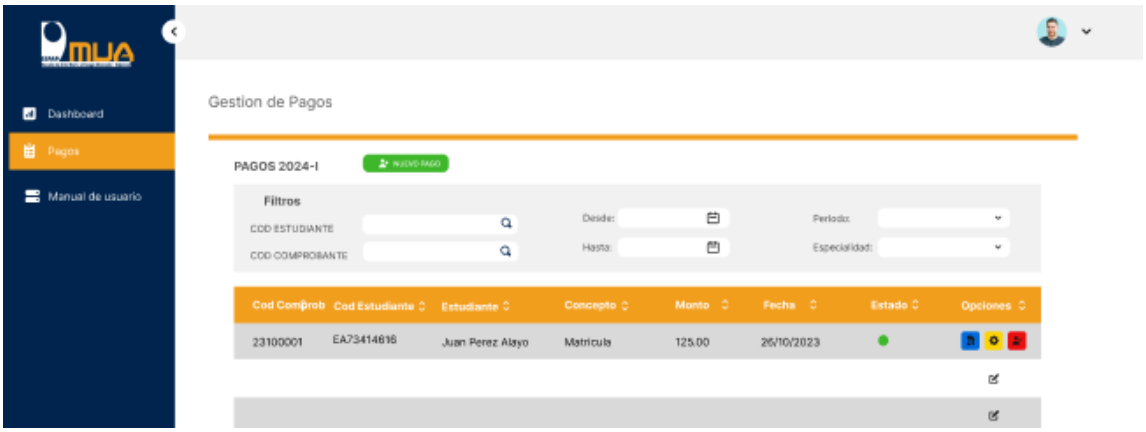

Fig. 39 Prototipo de la interfaz "Gestionar pagos"

Posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 40).

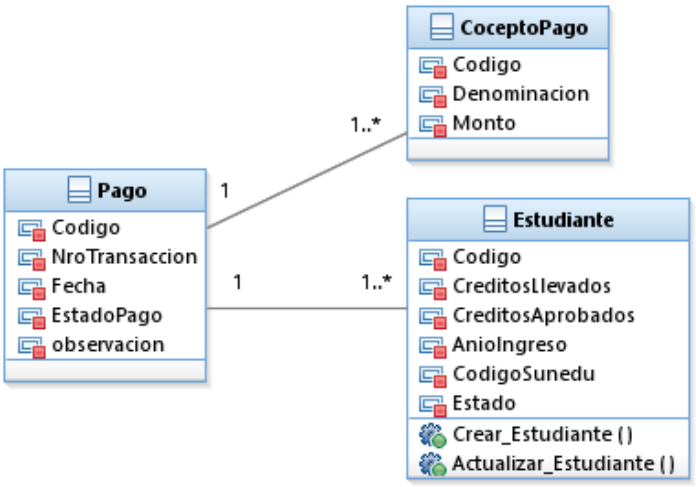

Fig. 40 Diagrama de clases "Gestionar pagos"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 41.

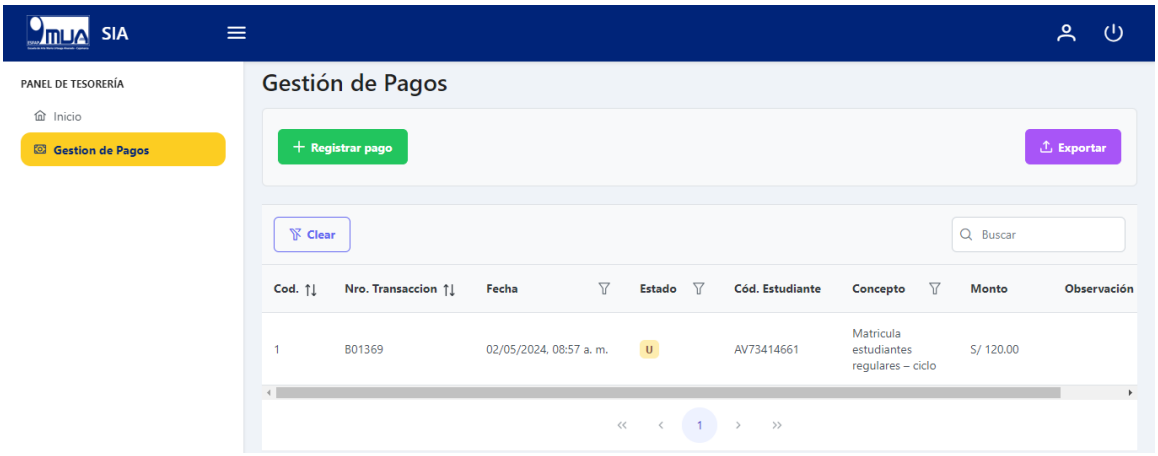

Fig. 41 Vista "Gestionar pagos"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XVIII Prueba de caja negra "Gestionar pagos"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU16	El sistema debe permitir agregar nuevos pagos a través de un formulario con campos validados	Datos almacenados correctamente respetando las validaciones	Exitoso
	El sistema debe permitir modificar el estado de los pagos registrados	Estado de los pagos modificados correctamente	Exitoso
	El sistema debe permitir generar en formato PDF un comprobante del pago registrado, para luego poder ser descargado o imprimido	Generación del comprobante de pago en formato PDF el cual puede ser descargado o imprimido	Exitoso

Indicadores kanban:

- Lead Time: 5d
- Cycle Time: 4.5d
- Flow Efficiency: FE = 4.5/5 * 100 = 90

HU17. Generar matrícula

Tabla XIX HU18 "Generar matrícula"

Historia de usuario	
Código: HU17	**Usuario**: Estudiante
Nombre de la historia: Generar matrícula	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Alta
Dependencia para su desarrollo: HU02, HU04, HU10	
Responsable: Equipo de desarrollo	
Descripción: Como estudiante, quiero generar mi matrícula, para inscribirme en el ciclo académico.	
Criterios de aceptación: - El sistema permite habilitar la matrícula de un estudiante entre las fechas correspondientes siempre y cuando se haya registrado su pago bajo el concepto de matrícula - El sistema debe mostrar la lista de cursos a los cuales un estudiante puede matricularse según su estado académico - El sistema debe mostrar una segunda lista con los cursos seleccionados por el estudiante en los cuales desea matricularse - El sistema permite deseleccionar un curso si así lo decide el estudiante - El sistema debe mostrar un botón de finalizar matrícula, el cual genera la matrícula del estudiante con los cursos que ha seleccionado - El sistema debe permitir generar la constancia de matrícula de un estudiante en formato PDF y poder ser descargado o imprimido	

Luego de detallar la HU, se realizó el prototipo de la interfaz, para lo cual se decidió utilizar Figma (ver Fig. 42).

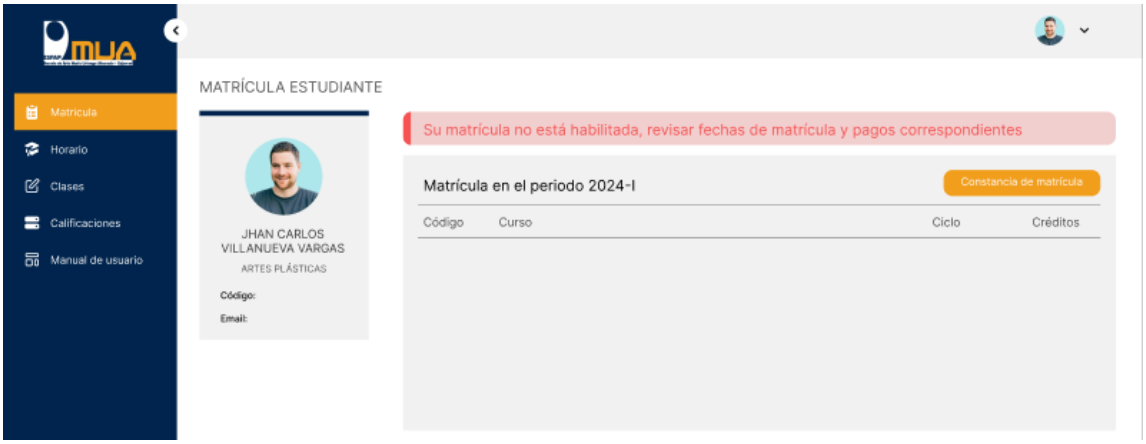

Fig. 42 Prototipo de la interfaz "Generar matrícula"

Posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 43).

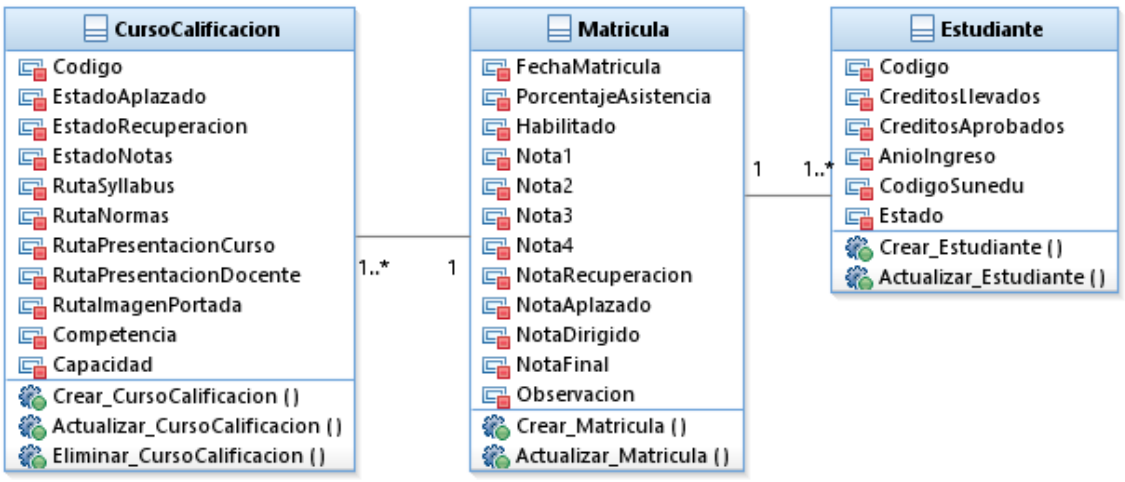

Fig. 43 Diagrama de clases "Generar matrícula"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 44.

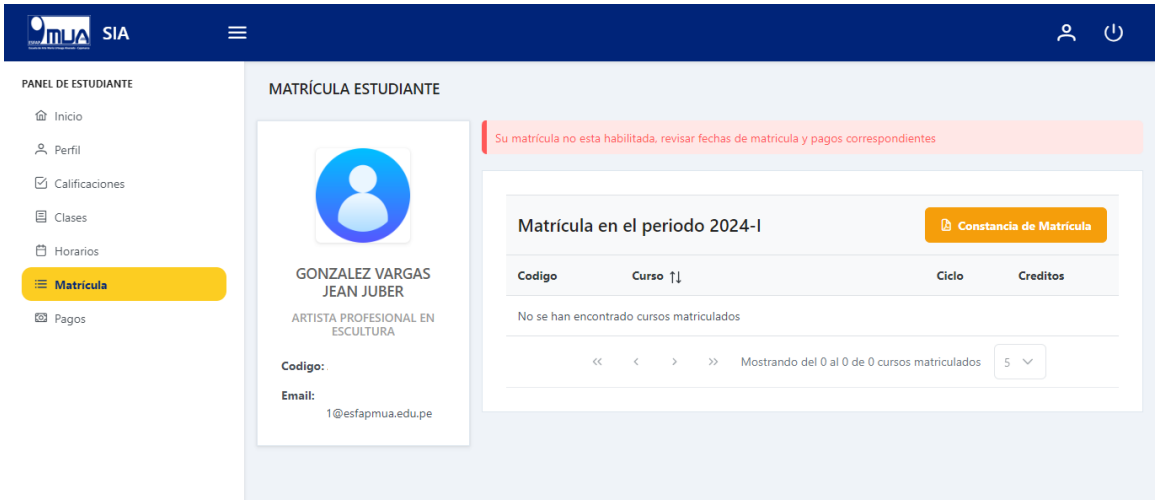

Fig. 44 Vista "Generar matrícula"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XX Prueba de caja negra "Generar matrícula"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU17	El sistema permite habilitar la matrícula de un estudiante entre las fechas	La matrícula se habilita según las restricciones de fecha y pago	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	correspondientes siempre y cuando se haya registrado su pago bajo el concepto de matrícula	correspondientes, en caso contrario muestra un mensaje según sea el caso	
	El sistema debe mostrar la lista de cursos a los cuales un estudiante puede matricularse según su estado académico	La lista de cursos se muestra correctamente	Exitoso
	El sistema debe mostrar una segunda lista con los cursos seleccionados por el estudiante en los cuales desea matricularse	La segunda lista muestra correctamente los cursos que han sido seleccionados por el estudiante	Exitoso
	El sistema debe permitir deseleccionar un curso si así lo decide el estudiante	Los cursos de deselecciona correctamente, desapareciendo de la segunda lista y volviendo a aparecer en la lista inicial	Exitoso
	El sistema debe mostrar un botón de finalizar matrícula, el cual genera la matrícula del estudiante con los cursos que ha seleccionado	Se muestra el botón de finalizar matrícula que cumple con la función de generar la matrícula del estudiante y así mismo deshabilita su edición	Exitoso
	El sistema debe permitir generar la constancia de matrícula de un estudiante en formato PDF y poder ser descargado o imprimido	Luego de finalizar una matrícula se visualiza un botón que permite generar la constancia de esta en formato PDF, la cual puede ser descargada o imprimida	Exitoso

Indicadores kanban:

- Lead Time: 7d
- Cycle Time: 6.5d
- Flow Efficiency: FE = 6.5/7 * 100 = 93

HU05. Gestionar matrículas

Tabla XXI HU05 "Gestionar matrículas"

Historia de usuario	
Código: HU05	**Usuario**: Administrador académico
Nombre de la historia: Gestionar matrículas	
Prioridad en la organización: Alta	**Riesgo en desarrollo**: Media
Dependencia para su desarrollo: HU03, HU17	
Responsable: Equipo de desarrollo	
Descripción: Como administrador, quiero listar, editar y eliminar matrículas, para mantener la información de matrícula al día.	
Criterios de aceptación: - El sistema debe contar con la opción de buscar la matrícula de un estudiante ingresando el número de su DNI - El sistema debe mostrar la lista de cursos a los cuales un estudiante puede matricularse según su estado académico - El sistema debe mostrar una segunda lista con los cursos en los cuales se ha matriculado el estudiante - El sistema debe permitir agregar cursos a la lista de matrícula de un estudiante - El sistema debe permitir eliminar cursos de la lista de matrícula de un estudiante	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 45).

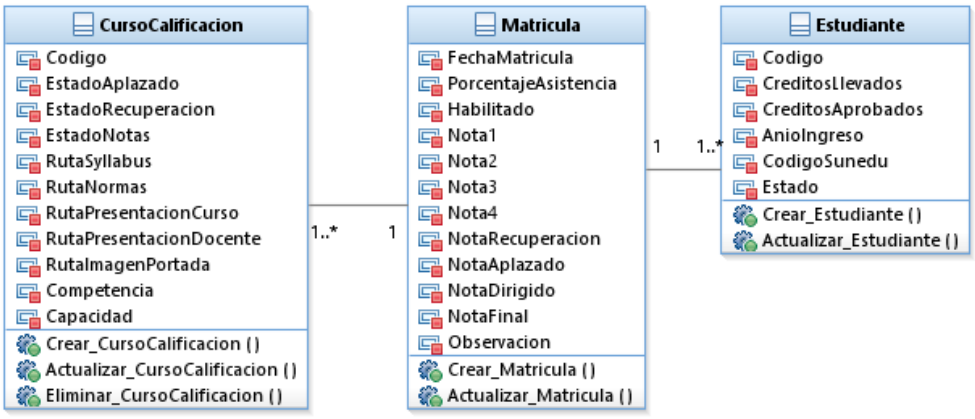

Fig. 45 Diagrama de clases "Gestionar matrícula"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 46.

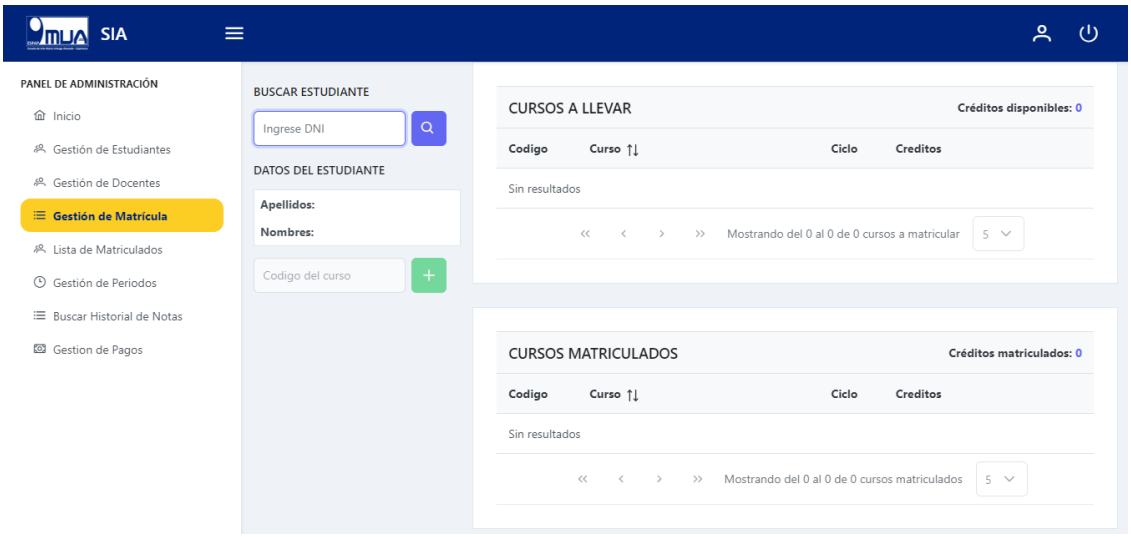

Fig. 46 Vista "Gestionar matrícula"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXII Prueba de caja negra "Gestionar matrícula"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU05	El sistema debe contar con la opción de buscar la matrícula de un estudiante ingresando el número de su DNI	Se muestra un input para buscar la matrícula de un estudiante haciendo uso de su número de DNI	Exitoso
	El sistema debe mostrar la lista de cursos a los cuales un estudiante puede matricularse según su estado académico	Se muestra correctamente la lista de los cursos a los cuales puede matricularse un estudiante	Exitoso
	El sistema debe mostrar una segunda lista con los cursos en los cuales se ha matriculado el estudiante	Se muestra correctamente la lista de cursos en los cuales se ha matriculado el estudiante	Exitoso
	El sistema debe permitir agregar cursos a la lista de matrícula de un estudiante	Cursos agregados correctamente	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	El sistema debe permitir eliminar cursos de la lista de matrícula de un estudiante	Cursos eliminados correctamente	Exitoso

Indicadores kanban:

- Lead Time: 6d
- Cycle Time: 5.5d
- Flow Efficiency: FE = 5.5/6 * 100 = 92

HU01. Iniciar sesión

Tabla XXIII HU01. Iniciar sesión

Historia de usuario	
Código: HU01	**Usuario**: Iniciar sesión
Nombre de la historia: Gestión de matrículas	
Prioridad en la organización: Media	**Riesgo en desarrollo**: Media
Dependencia para su desarrollo: Ninguna	
Responsable: Jhan Villanueva	
Descripción: Como usuario del sistema, quiero iniciar sesión con mi correo y contraseña, para acceder al sistema.	
Criterios de aceptación: - El sistema debe permitir al usuario ingresar su correo y contraseña. - El campo de contraseña debe ocultar los caracteres mientras se escribe. - Si el usuario ingresa credenciales válidas, es redirigido al panel principal del sistema y debe visualizar solo las interfaces según su perfil de usuario. - Si las credenciales son incorrectas, el sistema debe mostrar un mensaje de error claro sin especificar cuál campo es incorrecto. - El sistema debe permitir al usuario permanecer autenticado durante un periodo definido (15 minutos de inactividad).	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 47).

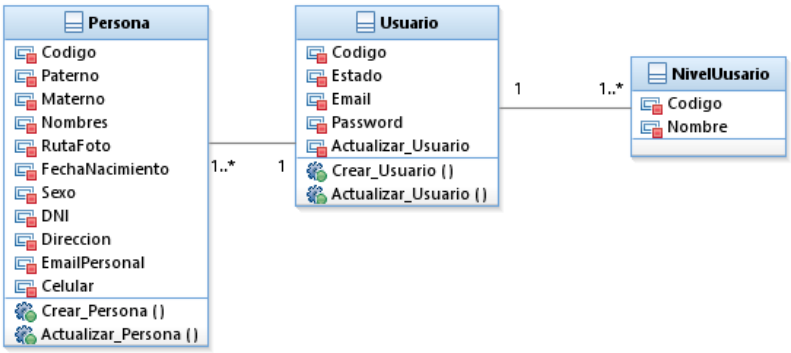

Fig. 47 Diagrama de clases "Iniciar sesión"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente.

En la Fig. 48 se muestra la arquitectura sobre la cual se desarrolló la lógica del componente de acceso al sistema.

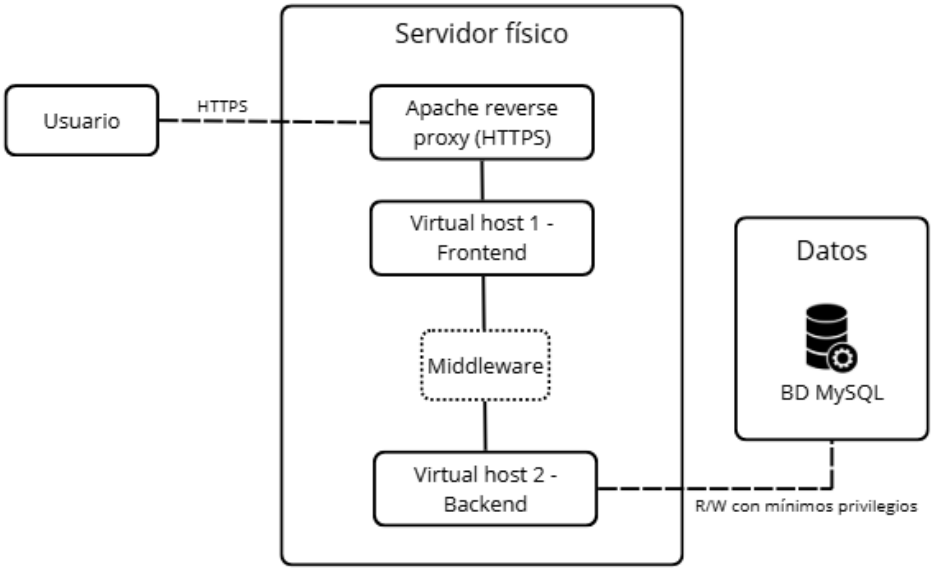

Fig. 48 Arquitectura de seguridad de acceso

A. Componentes principales del diagrama

a. Usuario (navegador web)

El acceso siempre se realiza bajo el protocolo HTPPS, asegurando el cifrado de extremo a extremo de las credenciales y de la sesión de trabajo.

b. Servidor físico

Reverse proxy: filtra tráfico malicioso, protege contra ataques comunes y aplica reglas de firewall a nivel de aplicación.

Virtual host 1-frontend: recibe peticiones de los usuarios y muestra la interfaz. No accede directamente a la base de datos.

Middleware: actúa como un filtro que intercepta las solicitudes entrantes antes de que lleguen a las rutas protegidas, utiliza una biblioteca JWT para verificar la validez del token.

Virtual host 2-backend/API: contiene la lógica de negocios y es el único con permisos controlados de lectura/escritura en la base de datos.

c. Capa de datos

Base de datos MySQL: accesible únicamente por el backend, con usuarios de servicio de privilegios mínimos. Contiene datos de todo el sistema académico (estudiantes, docentes, matrículas, cursos, etc.).

B. Seguridad en el acceso (autenticación)

- Credenciales almacenadas con algoritmos robustos de hash (bycript).
- Estándar JWT para transmitir información de forma segura entre backend y frontend mediante un token firmado.
- Sesiones seguras mediante cookies con atributos de seguridad.
- Contraseña de 8 caracteres como mínimo.

C. Seguridad en los perfiles

Se aplica RBAC (Role Based Access Control):

- Administrador: Acceso a la gestión de estudiantes, docentes, matrícula, periodos académicos.
- Soporte: Acceso a la gestión de usuarios, jefes de departamento, cursos.
- Jefe de departamento: Acceso a la habilitación y seguimiento de cursos a dictar.
- Docente: Acceso a la gestión de sus cursos (lista de estudiantes, clases, asistencias, calificaciones, actividades) y visualización de sus horarios.
- Estudiante: Acceso a su perfil, historial de calificaciones, clases, horarios, matrícula, resumen de pagos.
- Tesorería: Gestión de pagos.

La interfaz de acceso para los usuarios se muestra en la Fig. 49.

Fig. 49 Vista "Iniciar sesión"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXIV Prueba de caja negra "Iniciar sesión"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU01	El sistema debe permitir al usuario ingresar su correo y contraseña	Una interfaz de inicio de sesión que incluya un formulario de ingreso de datos	Exitoso
	El campo de contraseña debe ocultar los caracteres mientras se escribe	Al ingresar un carácter en el input de contraseña este sea reemplazado visualmente con asteriscos (*)	Exitoso
	Si el usuario ingresa credenciales válidas, debe ser redirigido al panel principal del sistema y debe poder visualizar solo las interfaces según su perfil de usuario.	Ser redireccionado al panel principal y visualizar las interfaces según su perfil de usuario	Exitoso
	Si las credenciales son incorrectas, el sistema debe mostrar un mensaje de error	Visualizar un mensaje que indique que hay error en las credenciales ingresadas	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	claro sin especificar cuál campo es incorrecto		
	El sistema debe permitir al usuario permanecer autenticado durante un período definido (15 minutos de inactividad)	Luego de 15 minutos de inactividad en el sistema ser redirigido a la interfaz de inicio de sesión	Exitoso

Indicadores kanban:

- Lead Time: 6d
- Cycle Time: 4.5d
- Flow Efficiency: FE = 4.5/6 * 100 = 75

HU09. Gestionar usuarios del sistema

Tabla XXV HU09 "Gestionar usuarios del sistema"

Historia de usuario	
Código: HU09	**Usuario**: Soporte
Nombre de la historia: Gestionar usuarios del sistema	
Prioridad en la organización: Media	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU01	
Responsable: Jhan Villanueva	
Descripción: Como soporte, quiero gestionar usuarios del sistema, para controlar los accesos y roles.	
Criterios de aceptación: - El sistema debe permitir agregar nuevos usuarios de rol administrador o tesorería a través de un formulario con campos validados - El sistema debe permitir editar la información de los usuarios registrados - El sistema de permitir eliminar (deshabilitar) usuarios - El sistema debe permitir reestablecer o cambiar la contraseña de todos los usuarios según sea requerido	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 50).

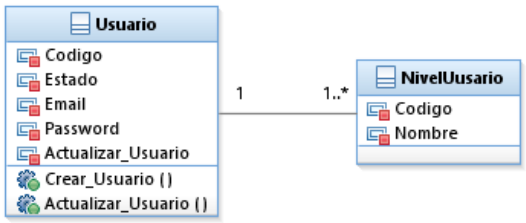

Fig. 50 Diagrama de clases "Gestionar usuarios del sistema"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 51.

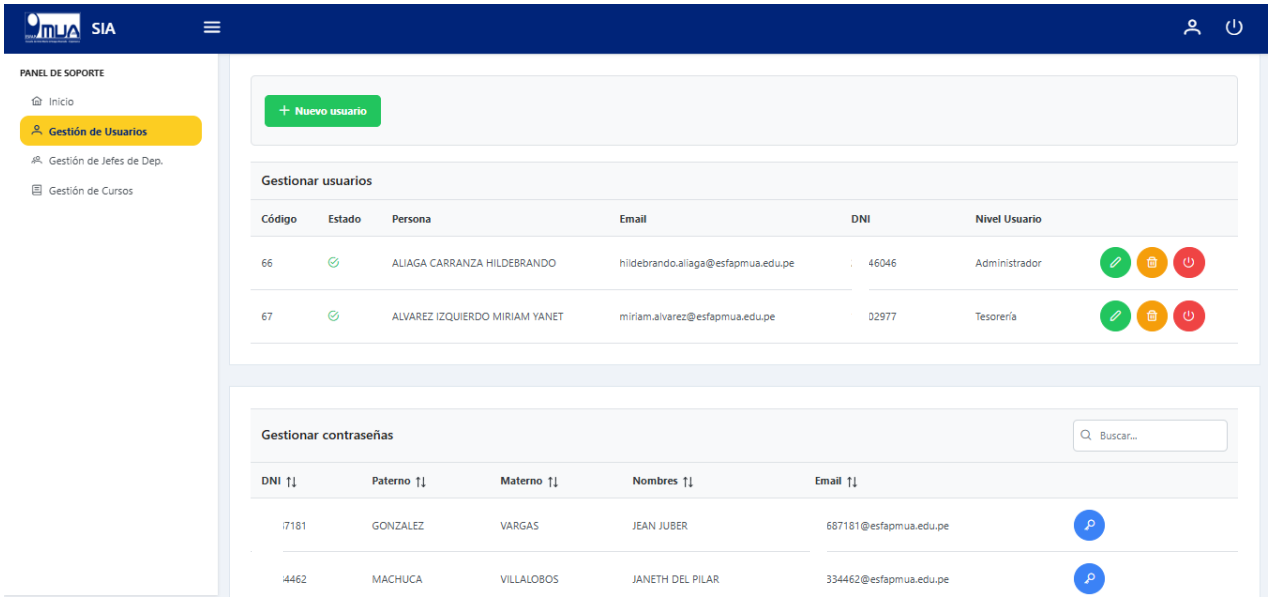

Fig. 51 Vista "Gestionar usuarios del sistema"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXVI Prueba de caja negra "Gestionar usuarios del sistema"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU09	El sistema debe permitir agregar nuevos usuarios de rol administrador o tesorería a través de un formulario con campos validados	Datos almacenados correctamente respetando las validaciones	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	El sistema debe permitir editar la información de los usuarios registrados	Datos de los usuarios editados correctamente	Exitoso
	El sistema de permitir eliminar (deshabilitar) usuarios	El estado de los usuarios se modifica y el cambio se muestra de manera visual	Exitoso
	El sistema debe permitir reestablecer o cambiar la contraseña de todos los usuarios según sea requerido	Ingresar al sistema con la nueva contraseña	Exitoso

Indicadores kanban:

- Lead Time: 4d
- Cycle Time: 3.5d
- Flow Efficiency: FE = 3.5/4 * 100 = 88

HU11. Gestionar sesiones de clases

Tabla XXVII HU11 "Gestionar sesiones de clases"

Historia de usuario	
Código: HU11	**Usuario**: Docente
Nombre de la historia: Gestionar sesiones de clases	
Prioridad en la organización: Media	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU17	
Responsable: Jhan Villanueva	
Descripción: Como docente, quiero listar mis cursos asignados, gestionar sesiones de clase (crear, listar, editar, eliminar), para estructurar mi enseñanza.	
Criterios de aceptación: - El sistema debe mostrar al docente solo los cursos que tiene asignados - El docente debe poder seleccionar un curso para ver o gestionar sus sesiones de clase - El docente debe poder crear una nueva sesión de clase especificando: fecha, hora y tema - El docente debe poder editar los datos de una sesión previamente creada - El docente debe poder eliminar una sesión de clase y confirmar la acción antes de que se aplique	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 52).

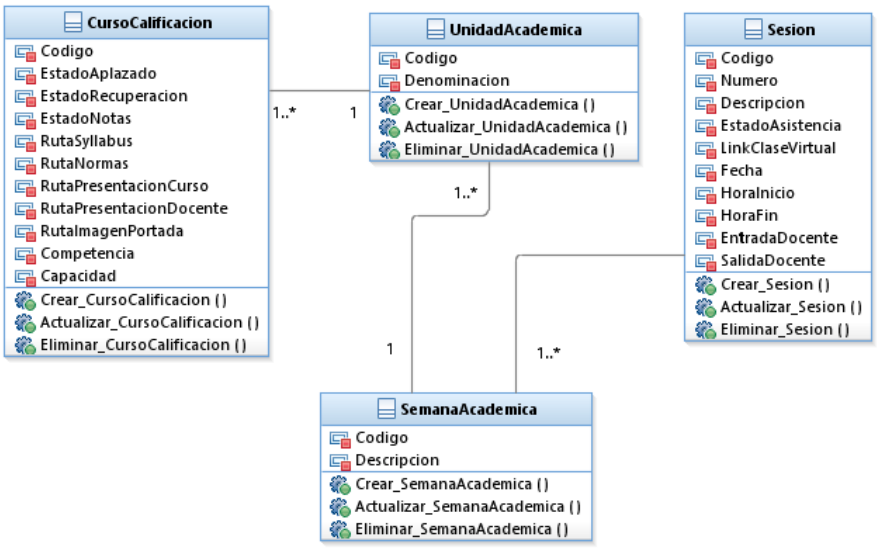

Fig. 52 Diagrama de clases "Gestionar sesiones de clases"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado las interfaces mostradas en las Fig. 53 y 54.

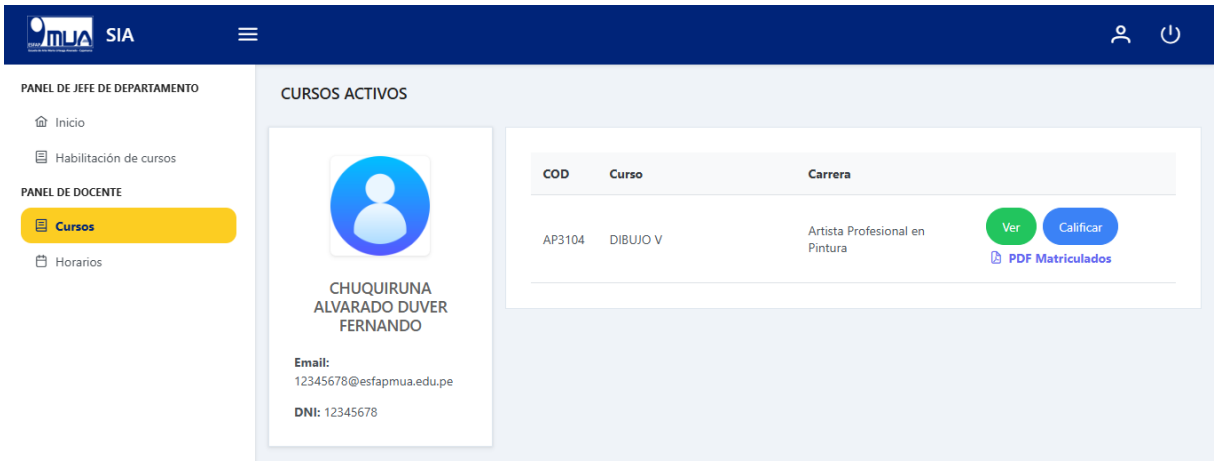

Fig. 53 Vista cursos activos por docente

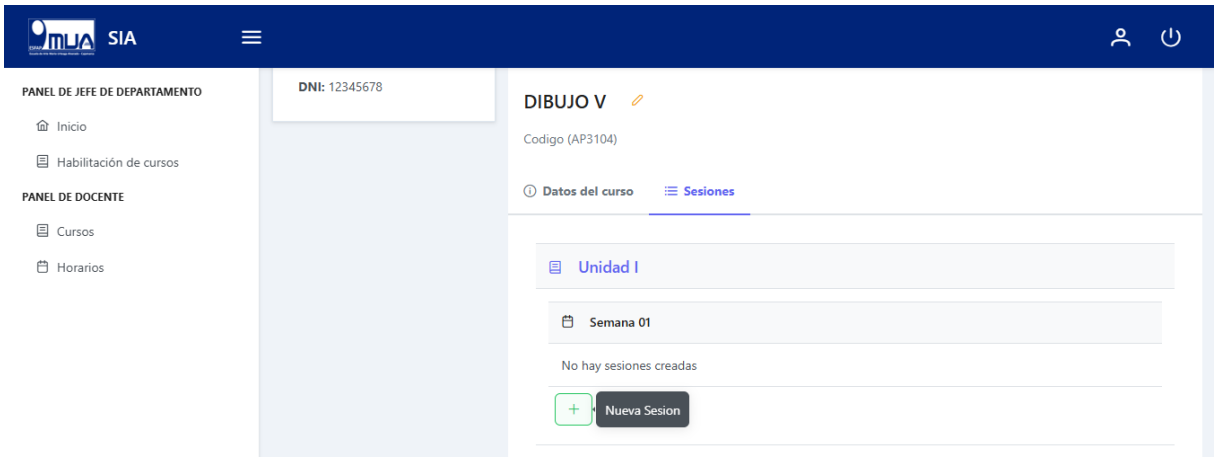

Fig. 54 Vista "Gestionar sesiones de clases"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXVIII Prueba de caja negra "Gestionar sesiones de clases"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU11	El sistema debe mostrar al docente solo los cursos que tiene asignados	Solo se muestran los cursos pertenecientes al docente	Exitoso
	El docente debe poder seleccionar un curso para ver o gestionar sus sesiones de clase	Cada curso muestra un botón (ver) que lleva al detalle del curso y sus sesiones	Exitoso
	El docente debe poder crear una nueva sesión de clase especificando: fecha, hora y tema	Se muestra un botón que permite abrir un modal y registrar una nueva sesión	Exitoso
	El docente debe poder editar los datos de una sesión previamente creada	Se muestra un botón que permite la edición de datos de una sesión	Exitoso
	El docente debe poder eliminar una sesión de clase y confirmar la acción antes de que se aplique	Se muestra un botón que permite eliminar una sesión que no tenga datos relacionados	Exitoso

Indicadores kanban:

- Lead Time: 7d
- Cycle Time: 6.5d
- Flow Efficiency: FE = 6.5/7 * 100 = 93

HU12. Gestionar desempeño de estudiantes

Tabla XXIX HU12 "Gestionar desempeño de estudiantes"

Historia de usuario	
Código: HU12	**Usuario**: Docente
Nombre de la historia: Gestionar desempeño de estudiantes	
Prioridad en la organización: Media	**Riesgo en desarrollo**: Media
Dependencia para su desarrollo: HU11	
Responsable: Equipo de desarrollo	
Descripción: Como docente, quiero listar alumnos por curso, tomar asistencia y calificar para registrar su desempeño.	
Criterios de aceptación: - El docente debe poder seleccionar un curso asignado y visualizar el listado de alumnos inscritos - El listado debe incluir al menos: código SUNEDU, nombre completo y porcentaje de asistencia del alumno - El docente debe poder registrar la asistencia de cada alumno por sesión de clase - El docente debe poder editar la asistencia registrada hasta cierta fecha límite (configurable) - El docente debe poder ingresar calificaciones cuatro notas principales del curso, así como la nota por sustitución o aplazado para cada alumno - El sistema debe validar el rango de calificaciones permitido - El sistema debe mostrar mensajes de éxito o error tras guardar o modificar una calificación	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 55).

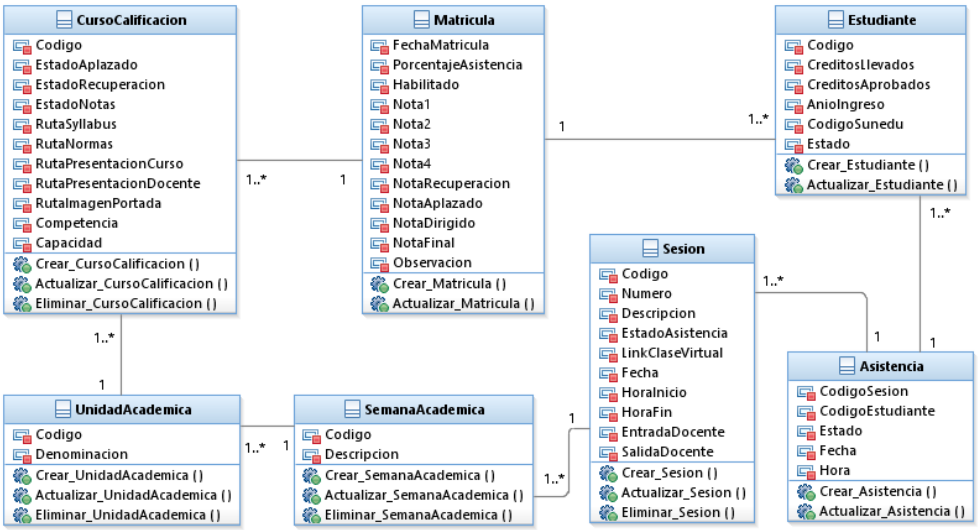

Fig. 55 Diagrama de clases "Gestionar desempeño de estudiantes"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado las interfaces mostradas en las Fig. 56 y 57.

Fig. 56 Vista "Gestionar desempeño de estudiantes"

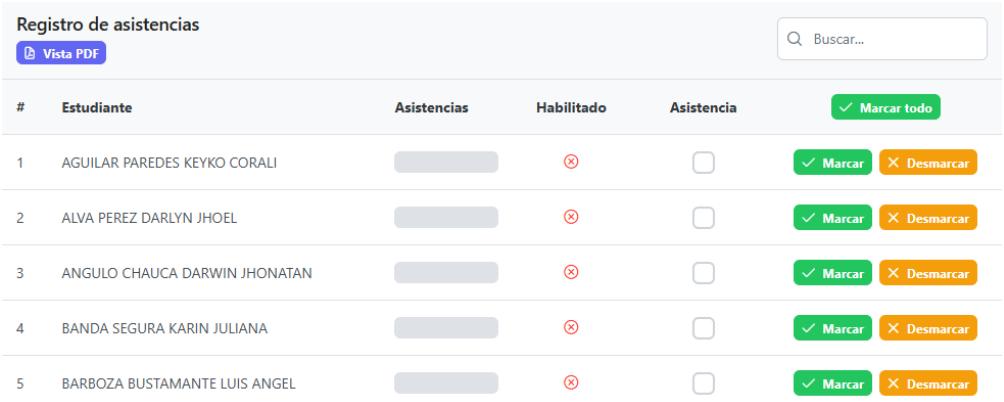

Fig. 57 Vista "Gestionar desempeño de estudiantes" toma de asistencia

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXX Prueba de caja negra "Gestionar desempeño de estudiantes"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU11	El docente debe poder seleccionar un curso asignado y visualizar el listado de alumnos inscritos	Visualizar un botón "calificar" en la interfaz de cursos del docente que dirige a una interfaz con la lista de estudiantes.	Exitoso
	El listado debe incluir al menos: código SUNEDU, nombre completo y porcentaje de asistencia del alumno	Visualizar la lista de estudiantes con los ítems mencionados	Exitoso
	El docente debe poder registrar la asistencia de cada alumno por sesión de clase	Visualizar en la interfaz de sesiones de clases, un botón que dirige a una vista para tomar asistencia de alumnos	Exitoso
	El docente debe poder editar la asistencia registrada hasta cierta fecha límite (configurable)	Se puede editar una lista de asistencia según una fecha límite determinada por el comité admirativo	Exitoso
	El docente debe poder ingresar calificaciones cuatro notas principales del curso, así como la nota por sustitución o aplazado para cada alumno	La interfaz muestra campos para ingresar las notas mencionadas para cada alumno, las cuales se habilitan y deshabilitan desde el panel del jefe de departamento	Exitoso
	El sistema debe validar el rango de calificaciones permitido	Está validado el rango de notas por cada tipo	Exitoso
	El sistema debe mostrar mensajes de éxito o error tras guardar o modificar una calificación	El sistema muestra mensajes de confirmación o de error que son comprensibles	Exitoso

Indicadores kanban:

- Lead Time: 8d
- Cycle Time: 7.5d
- Flow Efficiency: FE = 7.5/8 * 100 = 94

HU13. Gestionar actividades por cada sesión de clase

Tabla XXXI HU13 "Gestionar actividades por cada sesión de clase"

Historia de usuario	
Código: HU13	**Usuario**: Docente
Nombre de la historia: Gestionar actividades por cada sesión de clase	
Prioridad en la organización: Media	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU11	
Responsable: Equipo de desarrollo	
Descripción: Como docente, quiero gestionar actividades por sesión (crear, editar, calificar, eliminar), para evaluar a los estudiantes.	
Criterios de aceptación: - El docente debe poder crear una actividad asociada a una sesión de clase específica - La actividad debe incluir al menos: título, descripción, fecha de apertura y fecha de cierre - El docente debe poder editar los datos de una actividad creada, siempre que no haya sido calificada aún - El docente debe poder ingresar una calificación individual por alumno para cada actividad - El sistema debe validar el rango de calificación definido (0 a 20) - El docente debe poder eliminar una actividad, siempre que aún no haya sido calificada	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 58).

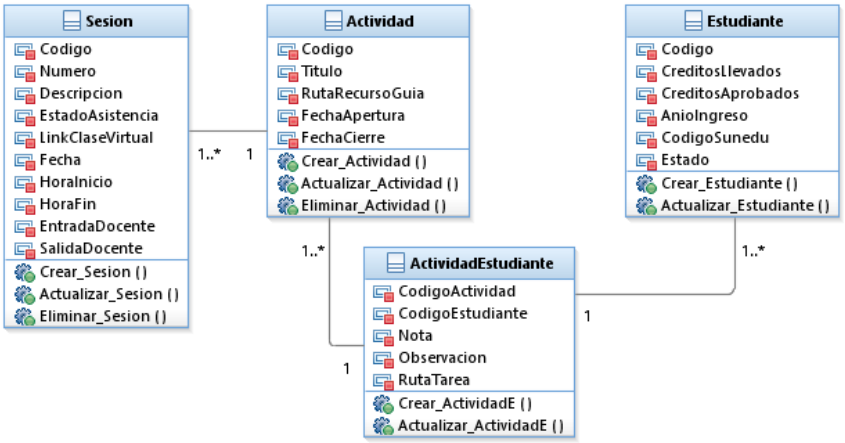

Fig. 58 Diagrama de clases "Gestionar actividades por cada sesión de clase"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 59

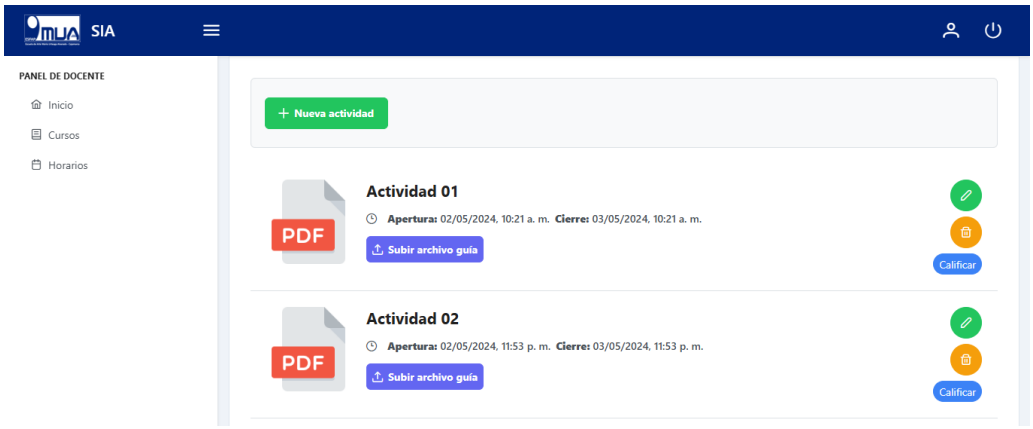

Fig. 59 Vista "Gestionar actividades por cada sesión de clase"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXXII Prueba de caja negra "Gestionar actividades por cada sesión de clase"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU13	El docente debe poder crear una actividad asociada a una sesión de clase específica	En la interfaz donde se listan las sesiones de clase, por cada una de ellas, se visualiza un botón para poder agregar actividades	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	La actividad debe incluir al menos: título, descripción, fecha de apertura y fecha de cierre	Se muestra un formulario para ingresar los datos de una nueva actividad con los campos especificados	Exitoso
	El docente debe poder editar los datos de una actividad creada, siempre que no haya sido calificada aún	Se puede editar los datos de una actividad que aún no se ha calificado	Exitoso
	El docente debe poder ingresar una calificación individual por alumno para cada actividad	Luego de finalizada la fecha de cierre de la actividad, el docente puede asignar calificaciones	Exitoso
	El sistema debe validar el rango de calificación definido (0 a 20)	Solo se puede ingresar calificaciones que sean numéricas dentro del rango especificado	Exitoso
	El docente debe poder eliminar una actividad, siempre que aún no haya sido calificada	Se puede eliminar una actividad que aún no se ha calificado	Exitoso

Indicadores kanban:

- Lead Time: 5d
- Cycle Time: 4.5d
- Flow Efficiency: FE = 4.5/5 * 100 = 90

HU18. Visualizar sesiones de clase, porcentaje de asistencia y calificaciones por cada curso

Tabla XXXIII HU18 "Visualizar sesiones de clase, porcentaje de asistencia y calificaciones por cada curso"

Historia de usuario	
Código: HU18	**Usuario**: Estudiante
Nombre de la historia: Visualizar sesiones de clase, porcentaje de asistencia y calificaciones por cada curso	
Prioridad en la organización: Media	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU12	
Responsable: Equipo de desarrollo	
Descripción: Como estudiante, quiero ver mis sesiones, notas y asistencia por curso, para hacer seguimiento a mi rendimiento.	
Criterios de aceptación:	

- El estudiante debe poder ver una lista de sesiones correspondientes a cada curso en el que está inscrito
- El estudiante debe poder ver las calificaciones obtenidas por actividad evaluativa, organizadas por curso
- El estudiante debe poder ver su registro de asistencia por sesión en cada curso
- El sistema debe mostrar el porcentaje general de asistencia por curso

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 54).

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado las interfaces mostradas en las Fig. 60, 61 y 62.

Fig. 60 Vista "Lista de cursos del estudiante"

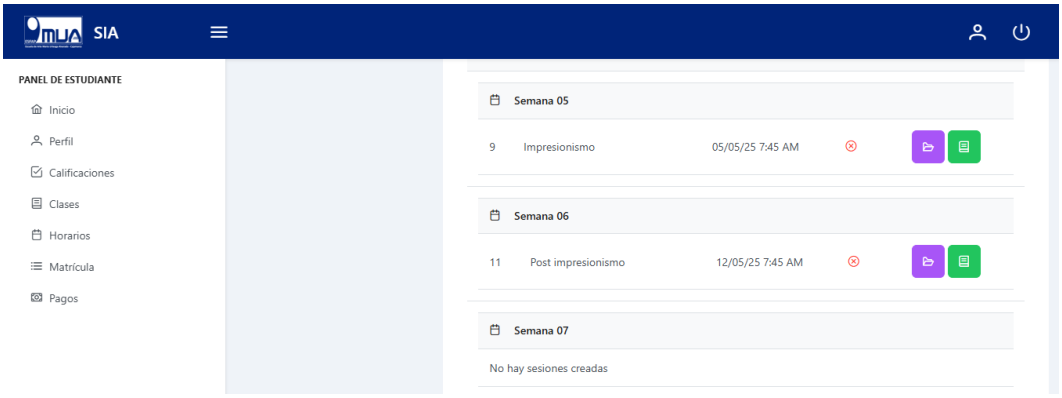

Fig. 61 Vista "Sesiones de clase del estudiante por curso"

Fig. 62 Vista "Calificaciones del estudiante por curso"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXXIV Prueba de caja negra "Detalles por curso del estudiante"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU18	El estudiante debe poder ver una lista de sesiones correspondientes a cada curso en el que está inscrito	Se visualiza la lista de cursos a los que está matriculado, y cada uno tiene un botón que dirige a los detalles del curso y sus sesiones correspondientes	Exitoso
	El estudiante debe poder ver las calificaciones obtenidas por actividad evaluativa, organizadas por curso	Se visualizan el resumen de las notas representativas por cada curso	Exitoso
	El estudiante debe poder ver su registro de asistencia por sesión en cada curso	En la interfaz de las sesiones se puede ver el estado de asistencia	Exitoso
	El sistema debe mostrar el porcentaje general de asistencia por curso	Se visualiza el porcentaje de asistencia por cada curso en base al número de sus sesiones	Exitoso

Indicadores kanban:

- Lead Time: 5d
- Cycle Time: 4.5d
- Flow Efficiency: FE = 4.5/5 * 100 = 90

HU19. Visualizar y desarrollar actividades

Tabla XXXV HU19 "Visualizar y desarrollar actividades"

Historia de usuario	
Código: HU19	**Usuario**: Estudiante
Nombre de la historia: Visualizar y desarrollar actividades	
Prioridad en la organización: Media	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU13	
Responsable: Equipo de desarrollo	
Descripción: Como estudiante, quiero completar y ver mis notas por actividad, para saber mi progreso.	
Criterios de aceptación: - El estudiante debe poder visualizar las actividades disponibles para completar, organizadas por curso y sesión - Cada actividad debe mostrar claramente su título, descripción y fecha límite de entrega - El estudiante debe poder enviar su respuesta o archivo según el tipo de actividad (texto, archivo, etc.) - No se debe permitir el envío de actividades fuera de la fecha límite - El estudiante debe poder ver su calificación por cada actividad una vez que haya sido evaluada	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 57).

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 63.

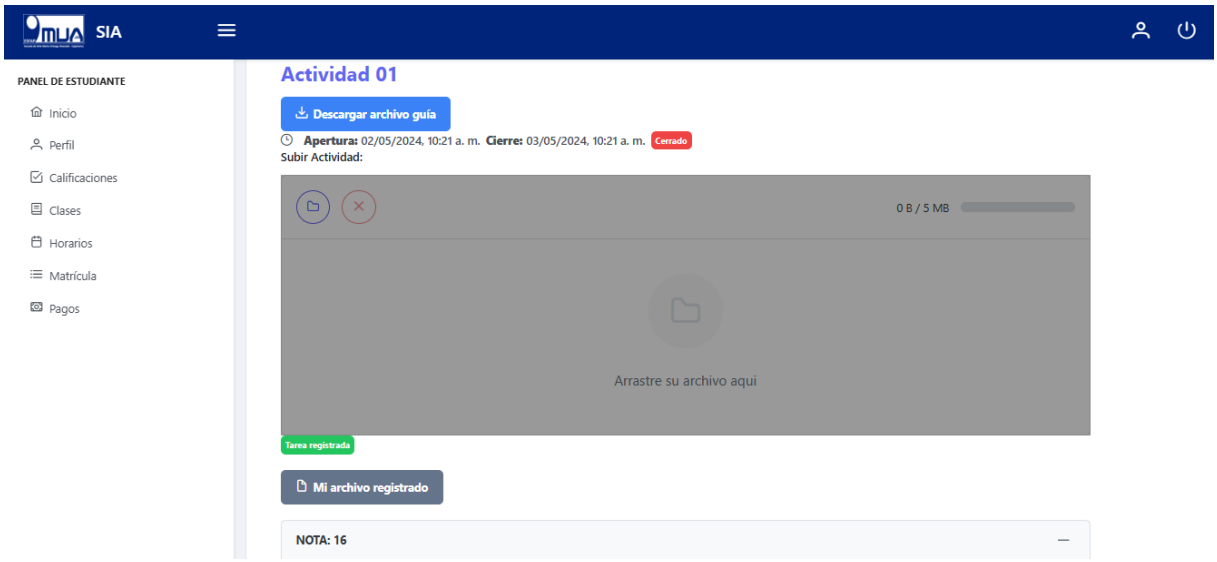

Fig. 63 Vista "Visualizar y desarrollar actividades"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXXVI Prueba de caja negra "Visualizar y desarrollar actividades"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU19	El estudiante debe poder visualizar las actividades disponibles para completar, organizadas por curso y sesión	En la interfaz de sesiones de clases, por cada una de ellas, se muestra un botón que dirige a la vista de actividades	Exitoso
	Cada actividad debe mostrar claramente su título, descripción y fecha límite de entrega	Se visualiza la actividad y su detalle claramente	Exitoso
	El estudiante debe poder enviar su respuesta o archivo según el tipo de actividad (texto, archivo, etc.)	La respuesta se registra correctamente indistintamente del tipo de actividad	Exitoso
	No se debe permitir el envío de actividades fuera de la fecha límite	Se deshabilita la opción de enviar alguna respuesta cuando se está fuera de la fecha límite	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	El estudiante debe poder ver su calificación por cada actividad una vez que haya sido evaluada	En caso el docente haya registrado una calificación esta se muestra correctamente	Exitoso

Indicadores kanban:

- Lead Time: 6d
- Cycle Time: 5.5d
- Flow Efficiency: FE = 5.5/6 * 100 = 92

HU22. Visualizar pagos

Tabla XXXVII HU22 "Visualizar pagos"

Historia de usuario	
Código: HU22	**Usuario**: Estudiante
Nombre de la historia: Visualizar pagos	
Prioridad en la organización: Media	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo:	
Responsable: Equipo de desarrollo	
Descripción: Como estudiante, quiero visualizar mis pagos, para verificar que estén al día.	
Criterios de aceptación: - El estudiante debe poder acceder a una sección donde se muestren todos los pagos realizados - El sistema debe listar cada pago con al menos: concepto (ej. matrícula, certificados), monto, fecha de pago, estado (pagado/pendiente), y número de transacción - Los pagos deben estar organizados según su concepto, los que corresponden a matrículas y los demás agrupados en "otros pagos"	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 39).

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 64.

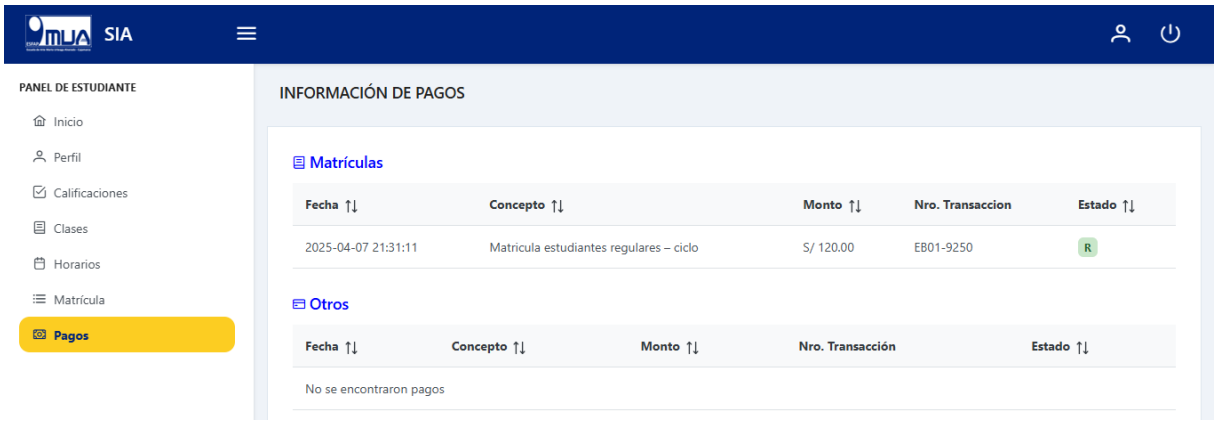

Fig. 64 Vista "Visualizar pagos"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XXXVIII Prueba de caja negra "Visualizar pagos"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU19	El estudiante debe poder acceder a una sección donde se muestren todos los pagos realizados	En el panel del estudiante se muestra una opción que dirige a la visualización de pagos	Exitoso
	El sistema debe listar cada pago con al menos: concepto (ej. matrícula, certificados), monto, fecha de pago, estado (pagado/pendiente), y número de transacción	Se visualizan los pagos con los detalles esperados	Exitoso
	Los pagos deben estar organizados según su concepto, los que corresponden a matrículas y los demás agrupados en "otros pagos"	Los pagos se muestran agrupados correctamente	Exitoso

Indicadores kanban:

- Lead Time: 3d
- Cycle Time: 2.5d
- Flow Efficiency: FE = 2.5/3 * 100 = 83

HU06. Filtrar y visualizar notas de cada estudiante

Tabla XXXIX HU06 "Filtrar y visualizar notas de cada estudiante"

Historia de usuario	
Código: HU06	**Usuario**: Administrador Académico
Nombre de la historia: Filtrar y visualizar notas de cada estudiante	
Prioridad en la organización: Baja	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU12	
Responsable: Jhan Villanueva	
Descripción: Como administrador, quiero ver el historial de notas por estudiante, para hacer seguimiento académico.	
Criterios de aceptación: - El sistema debe permitir al administrador autenticado acceder al historial de notas de cualquier estudiante registrado - El sistema debe permitir buscar estudiantes por DNI - El sistema debe mostrar, para cada curso inscrito, el estudiante, la nota obtenida, el nombre del curso y el ciclo académico - El sistema debe ofrecer la opción de exportar el historial de notas en formato PDF	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 54).

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 65.

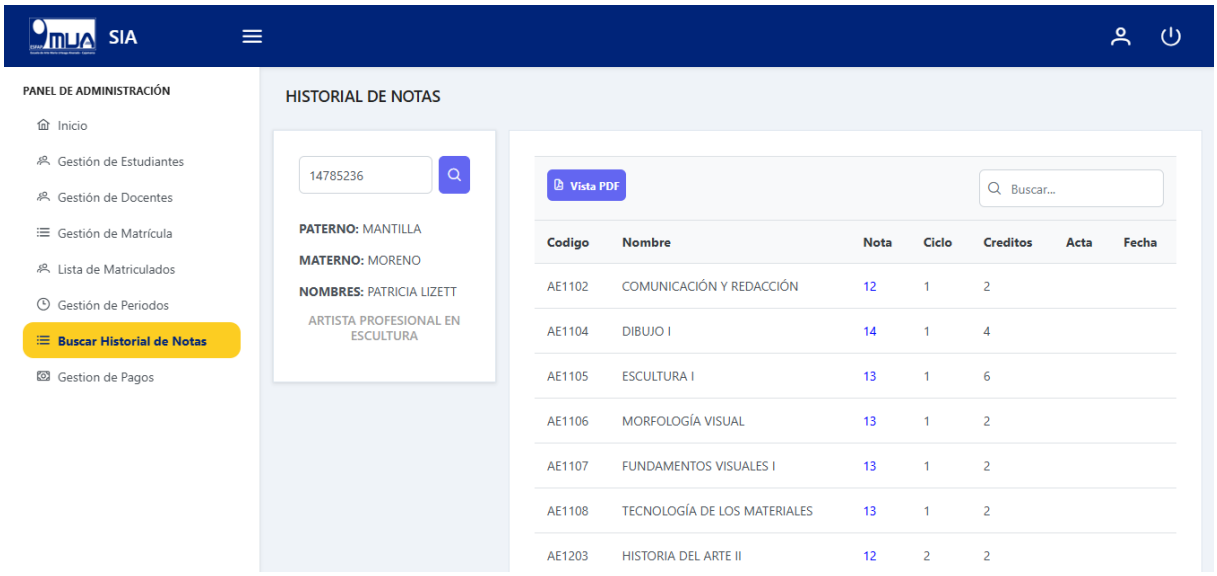

Fig. 65 Vista "Filtrar y visualizar notas de cada estudiante"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera

iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XL Prueba de caja negra "Filtrar y visualizar notas de cada estudiante"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU06	El sistema debe permitir al administrador autenticado acceder al historial de notas de cualquier estudiante registrado	El sistema muestra una sección para visualizar historiales de notas de los estudiantes	Exitoso
	El sistema debe permitir buscar estudiantes por DNI	Se visualiza un input donde ingresar el DNI a buscar y el botón correspondiente	Exitoso
	El sistema debe mostrar, para cada curso inscrito, el estudiante, la nota obtenida, el nombre del curso y el ciclo académico	El sistema muestra los datos esperados	Exitoso
	El sistema debe ofrecer la opción de exportar el historial de notas en formato PDF	Se visualiza una opción "vista PDF" que permite exportar los datos del historial	Exitoso

Indicadores kanban:

- Lead Time: 3d
- Cycle Time: 2.5d
- Flow Efficiency: FE = 2.5/3 * 100 = 83

HU14. Generar acta de notas por cada curso

Tabla XLI HU14 "Generar acta de notas por cada curso"

Historia de usuario	
Código: HU14	**Usuario**: Docente
Nombre de la historia: Generar acta de notas por cada curso	
Prioridad en la organización: Baja	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU12	
Responsable: Jhan Villanueva	
Descripción: Como docente, quiero generar el acta de notas al finalizar el ciclo, para entregar reportes oficiales.	

Criterios de aceptación:

- El sistema debe permitir generar un acta de notas por cada curso asignado, al finalizar el ciclo
- El acta de notas debe contener: nombre del curso, código del curso, nombre del docente, ciclo académico, listado de estudiantes matriculados, y sus respectivas notas finales
- El acta de notas debe generarse en un formato oficial predefinido por la institución (PDF con membrete institucional y espacio para firma)
- Las notas registradas en el acta deben coincidir exactamente con las notas almacenadas en el sistema académico al momento de su generación
- El sistema debe permitir la descarga del acta en formato PDF
- Una vez generada el acta oficial, el sistema no debe permitir modificaciones a las notas reflejadas en dicha acta, salvo que el proceso institucional lo permita mediante autorización especial

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 66).

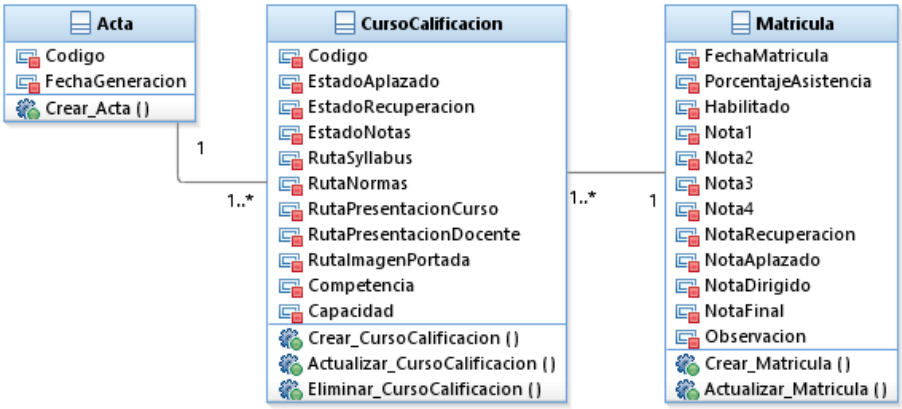

Fig. 66 Diagrama de clases "Generar acta de notas por cada curso"

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 67.

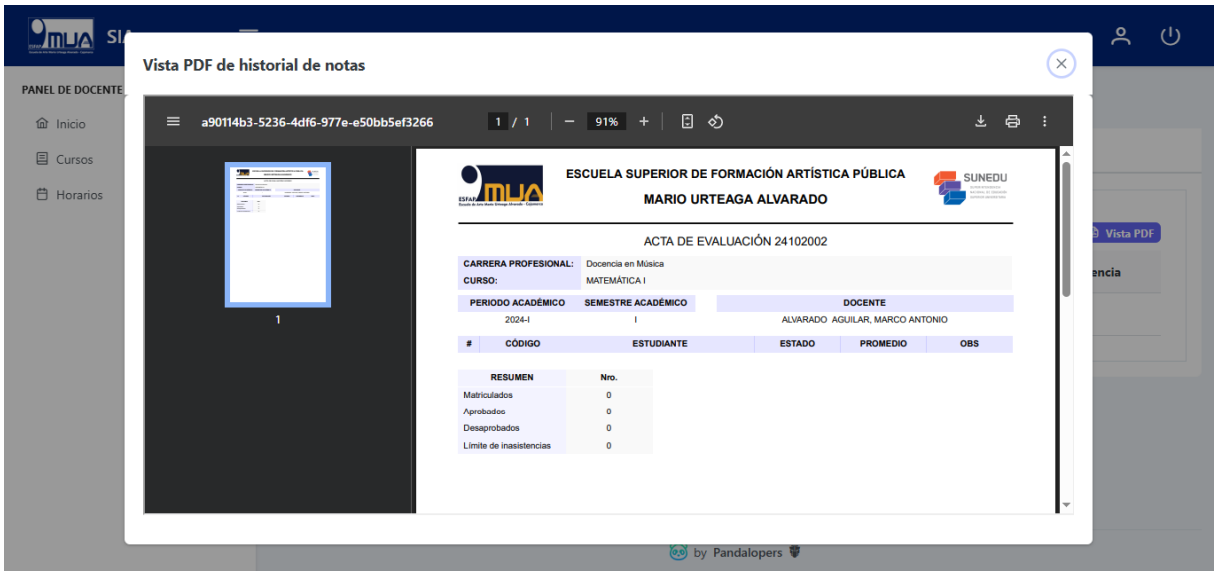

Fig. 67 Vista "Generar acta de notas por cada curso"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XLII Prueba de caja negra "Generar acta de notas por cada curso"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU14	El sistema debe permitir generar un acta de notas por cada curso asignado, al finalizar el ciclo	Pasada la fecha de término del ciclo actual, el sistema muestra una opción para generar el acta de notas previa confirmación	Exitoso
	El acta de notas debe contener: nombre del curso, código del curso, nombre del docente, ciclo académico, listado de estudiantes matriculados, y sus respectivas notas finales	El acta contiene todos los datos requeridos	Exitoso
	El acta de notas debe generarse en un formato oficial predefinido por la institución (PDF con membrete institucional y espacio para firma)	El acta se genera según especificaciones oficiales de la institución	Exitoso

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	Las notas registradas en el acta deben coincidir exactamente con las notas almacenadas en el sistema académico al momento de su generación	No existe ninguna divergencia entre las notas guardadas en el sistema y las que aparecen el acta	Exitoso
	El sistema debe permitir la descarga del acta en formato PDF	El acta se puede descargar correctamente en formato PDF	Exitoso
	Una vez generada el acta oficial, el sistema no debe permitir modificaciones a las notas reflejadas en dicha acta, salvo que el proceso institucional lo permita mediante autorización especial	La opción inicial de "Generar acta" ya no se visualiza, por el contrario, aparece un mensaje resaltante que menciona "Acta generada"	Exitoso

Indicadores kanban:

- Lead Time: 5d
- Cycle Time: 4.5d
- Flow Efficiency: FE = 4.5/5 * 100 = 90

HU15. Visualizar horario de clases (docente)

Tabla XLIII HU15 "Visualizar horario de clases (docente)"

Historia de usuario	
Código: HU15	**Usuario**: Docente
Nombre de la historia: Visualizar horario de clases	
Prioridad en la organización: Baja	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU10	
Responsable: Jhan Villanueva	
Descripción: Como docente, quiero visualizar mi horario de clases, para planificar mi tiempo.	
Criterios de aceptación: - El sistema debe permitir al docente autenticado acceder únicamente a su propio horario de clases - El sistema debe mostrar el horario de clases en un formato visual organizado - Cada clase del horario debe incluir: nombre del curso, aula asignada, hora de inicio y fin, y grupo correspondiente	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 35).

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 68.

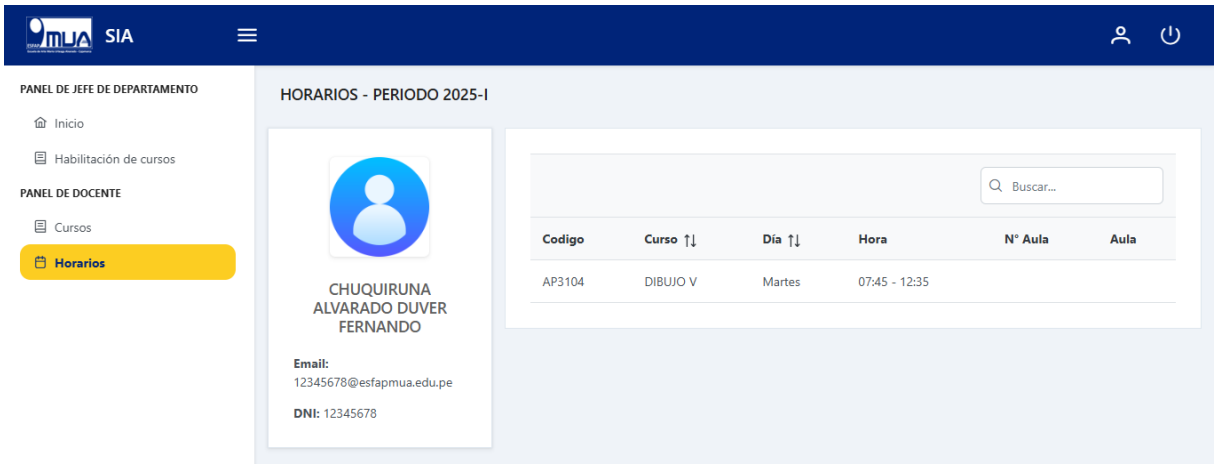

Fig. 68 Vista "Visualizar horario de clases (docente)"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XLIV Prueba de caja negra "Visualizar horario de clases (docente)"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU15	El sistema debe permitir al docente autenticado acceder únicamente a su propio horario de clases	El sistema muestra únicamente el detalle del horario del docente autenticado	Exitoso
	El sistema debe mostrar el horario de clases en un formato visual organizado	Los datos del horario se muestran organizados y entendibles	Exitoso
	Cada clase del horario debe incluir: nombre del curso, aula asignada, hora de inicio y fin, y grupo correspondiente	Se muestra la información solicitada correctamente	Exitoso

Indicadores kanban:

- Lead Time: 4d
- Cycle Time: 3.5d
- Flow Efficiency: FE = 3.5/4 * 100 = 88

HU20. Visualizar horario de clases (estudiante)

Tabla XLV HU20 "Visualizar horario de clases (estudiante)"

Historia de usuario	
Código: HU20	**Usuario**: Estudiante
Nombre de la historia: Visualización de horario de clases del estudiante	
Prioridad en la organización: Baja	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU10	
Responsable: Jhan Villanueva	
Descripción: Como estudiante, quiero visualizar mi horario de clases, para asistir puntualmente.	
Criterios de aceptación: - El sistema debe permitir al estudiante autenticado acceder a su propio horario de clases, así como también visualizar los horarios generales filtrados por nivel y ciclo - El sistema debe mostrar el horario de clases en un formato visual organizado - Cada clase del horario debe incluir: nombre del curso, aula asignada, hora de inicio y fin, y grupo correspondiente	

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 35).

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 69.

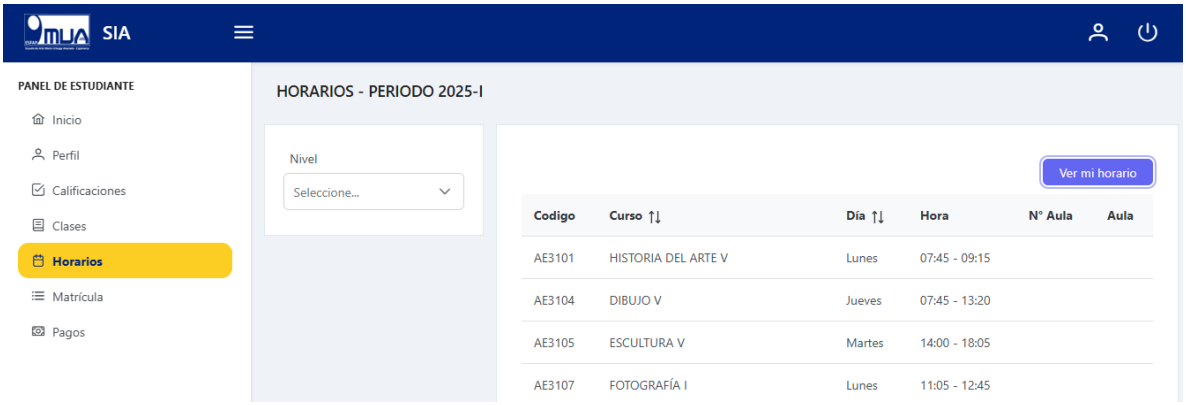

Fig. 69 Vista "Visualizar horario de clases (estudiante)"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XLVI Prueba de caja negra "Visualizar horario de clases (estudiante)"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU15	El sistema debe permitir al estudiante autenticado acceder a su propio horario de clases, así como también visualizar los horarios generales filtrados por nivel y ciclo	El sistema muestra el detalle del horario del estudiante autenticado, así como también la opción de ver los horarios generales	Exitoso
	El sistema debe mostrar el horario de clases en un formato visual organizado	Los datos del horario se muestran organizados y entendibles	Exitoso
	Cada clase del horario debe incluir: nombre del curso, aula asignada, hora de inicio y fin, y grupo correspondiente	Se muestra la información solicitada correctamente	Exitoso

Indicadores kanban:

- Lead Time: 3d
- Cycle Time: 2.5d
- Flow Efficiency: FE = 2.5/3 * 100 = 83

HU21. Visualizar historial de notas

Tabla XLVII HU21 "Visualizar historial de notas"

Historia de usuario	
Código: HU21	**Usuario**: Estudiante
Nombre de la historia: Visualizar historial de notas	
Prioridad en la organización: Baja	**Riesgo en desarrollo**: Baja
Dependencia para su desarrollo: HU14	
Responsable: Jhan Villanueva	
Descripción: Como estudiante, quiero ver mi historial de notas, para revisar mi desempeño anterior.	
Criterios de aceptación:	

- El sistema debe mostrar únicamente el historial de notas correspondiente al estudiante que ha iniciado sesión
- El historial de notas debe incluir, para cada curso: el nombre del curso, el código, la nota final, el ciclo académico, número de créditos, código de acta y fecha
- Las notas deben estar organizadas cronológicamente por ciclo académico, de más reciente a más antiguo
- El sistema debe permitir descargar el historial de notas en formato PDF

Luego de detallar la HU, se realizó el prototipo de la interfaz y posteriormente se procedió a realizar el diagrama de clases correspondiente (ver Fig. 35).

Seguidamente se realizó la codificación de sus clases MVC (modelo, vista, controlador) correspondiente, teniendo como resultado la interfaz mostrada en la Fig. 70.

Fig. 70 Vista "Visualizar historial de notas"

Una vez finalizada la codificación, esta Historia de Usuario (HU) se trasladó a la columna "Testing" del tablero Kanban. En esta fase, se llevaron a cabo pruebas de caja negra de manera iterativa, repitiendo el proceso hasta obtener resultados satisfactorios para cada uno de los criterios de aceptación especificados.

Tabla XLVIII Prueba de caja negra "Visualizar historial de notas"

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
HU15	El sistema debe mostrar únicamente el historial de notas correspondiente al estudiante que ha iniciado sesión	El historial de notas corresponde al estudiante autenticado	Exitoso
	El historial de notas debe incluir, para cada curso: el nombre del curso, el código, la nota final, el	Se visualizan correctamente los datos especificados	

Historia de usuario	Criterio de aceptación	Resultado esperado	Resultado obtenido
	ciclo académico, número de créditos, código de acta y fecha		
	Las notas deben estar organizadas cronológicamente por ciclo académico, de más reciente a más antiguo	Las notas se visualizan en orden cronológico	Exitoso
	El sistema debe permitir descargar el historial de notas en formato PDF	Se visualiza la opción correspondiente y se descargan los datos en formato PDF	Exitoso

Indicadores kanban:

- Lead Time: 3d
- Cycle Time: 2.5d
- Flow Efficiency: FE = 2.5/3 * 100 = 88

RNF05. Configurar el servidor físico para el despliegue del sistema, incluyendo las instalaciones de software necesarias

Este requerimiento implica la preparación del servidor físico que alojará el sistema académico. Incluye la instalación y configuración del sistema operativo, servidores web y de bases de datos, así como cualquier otro software necesario para el correcto funcionamiento y despliegue de la aplicación. El objetivo es garantizar un entorno estable, seguro y optimizado para la operación del sistema en producción, en la Fig. 71 podemos visualizar el diagrama de despliegue considerado.

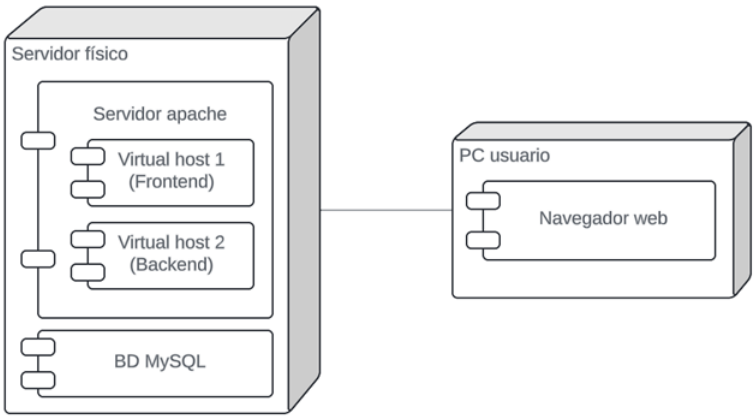

Fig. 71 Diagrama de despliegue (arquitectura)

Actividades que se realizaron en el servidor:

- Se instaló el sistema operativo Ubuntu Server en su última versión.
- Se instaló el servidor web Apache.
- Se instaló el servidor de base de datos MySQL.
- Se instaló Node.js, npm, el administrador de procesos PM2 y Git.
- Se crearon dos virtual host, uno para el backend y otro para el frontend.
- Se clonó el proyecto desde Git.
- Se activó y configuró el firewall usando UFW.
- Se configuró con una dirección IP fija y acceso remoto seguro vía SSH.
- Se realizaron pruebas para verificar que los servicios instalados están activos y accesibles desde otras máquinas en la red, y que el entorno está listo para recibir el despliegue del sistema.

Indicadores kanban:

- Lead Time: 4d
- Cycle Time: 3d
- Flow Efficiency: FE = 3/4 * 100 = 75

RNF06. Configurar el sistema para que se ejecute en el dominio institucional, incluyendo la configuración DNS

Este requerimiento contempla la configuración del sistema para que sea accesible desde el dominio institucional oficial. Incluye la asignación del nombre de dominio, la configuración de registros DNS, y el enlace entre el servidor donde está desplegado el sistema y el dominio, garantizando su disponibilidad a través de una URL institucional segura y fácil de recordar.

Actividades que se realizaron:

- Solicitar un IP pública estática al proveedor de internet.
- Configurar el reenvío de puertos en el router.
- Configurar correctamente los registros DNS necesarios apuntando al servidor donde está alojado el sistema.
- Validar la accesibilidad mediante un subdominio perteneciente al dominio institucional: https://sia.esfapmua.edu.pe/

- Validar que la accesibilidad al sistema se ejecute desde cualquier navegador web utilizando el subdominio asignado, sin necesidad de ingresar una dirección IP
- Agregar al sistema un certificado SSL válido, de manera que el dominio sea accesible vía HTTPS, garantizando una conexión segura.
- Se verificó que la propagación DNS se haya completado exitosamente y que el dominio funcione correctamente desde distintas redes.

Indicadores kanban:

- Lead Time: 6d
- Cycle Time: 4d
- Flow Efficiency: FE = 4/6 * 100 = 67

La implementación del sistema de información web presentó desafíos tanto técnicos como organizativos. En el plano tecnológico, uno de los principales inconvenientes se relacionó con el uso de un servidor físico, ya que la dirección IP proporcionada por el proveedor de internet, que debía ser estática, experimentaba cambios inesperados. Esta situación generaba interrupciones en la disponibilidad del sistema y dificultaba el acceso remoto de los usuarios, afectando la continuidad del servicio. Una vez identificado y resuelto este problema, se logró estabilizar el sistema, lo que permitió disponer de un periodo de dos semanas destinado a la adaptación de los usuarios.

En este lapso se evidenció resistencia al cambio, pues el sistema implicaba modificar la rutina de trabajo tradicional y adoptar nuevas dinámicas apoyadas en la tecnología. Para contrarrestar esta situación, se ejecutaron capacitaciones dirigidas que facilitaron la comprensión de las funcionalidades, se elaboró documentación funcional de fácil acceso para servir de guía en las actividades diarias y se fortaleció el soporte técnico con el fin de responder oportunamente a las dudas y dificultades que surgieran. Estas acciones contribuyeron a reducir la brecha de aceptación, permitiendo una transición más fluida hacia el uso cotidiano del sistema.

3.1.3.3 DIAGRAMA DE CLASES GENERAL

El diagrama de clases general se realizó haciendo uso de IBM Rational Software Architect Designer (ver Fig. 72).

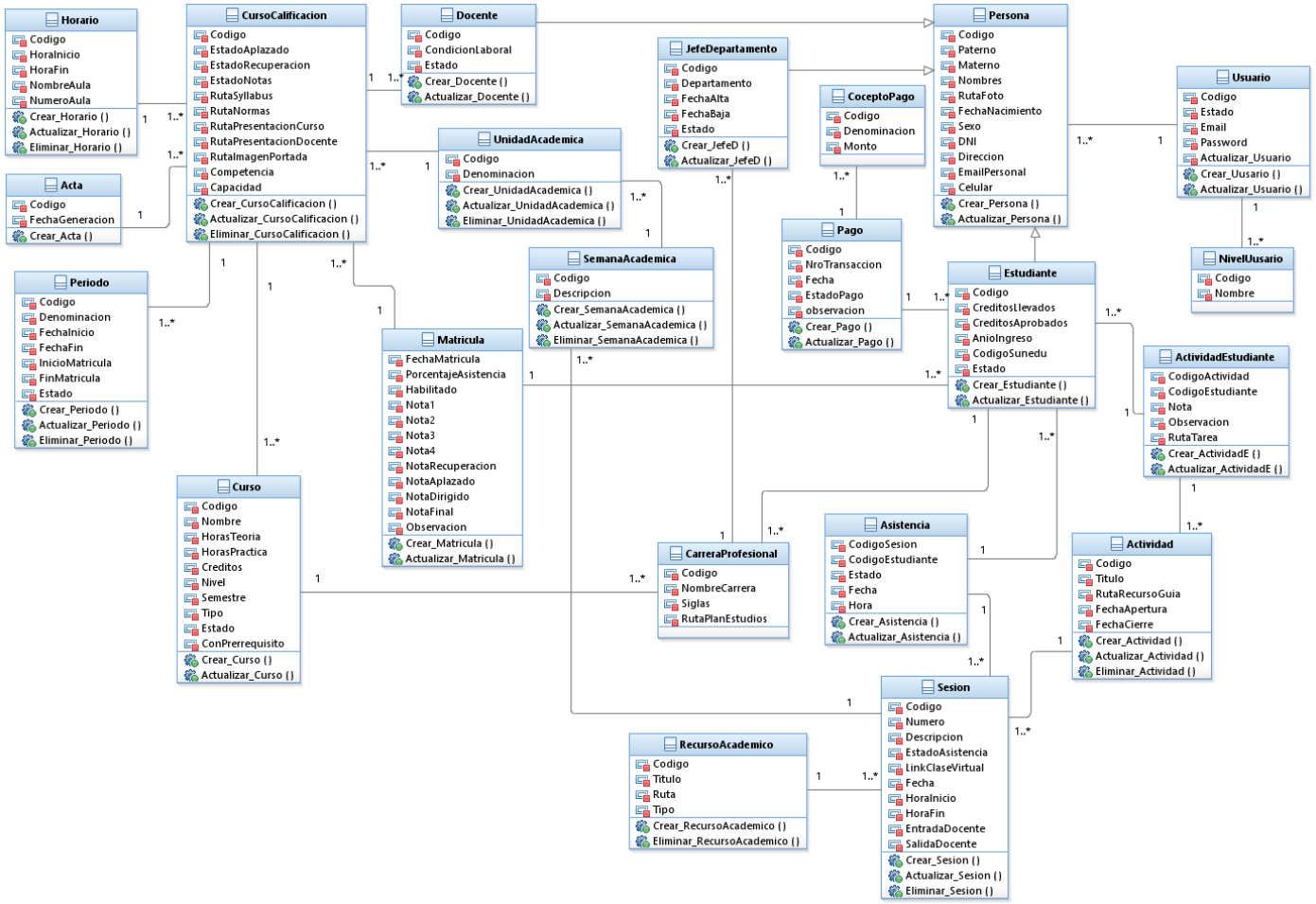

Fig. 72 Diagrama de clases general

3.1.3.4 DIAGRAMA DE BASE DE DATOS

El diagrama de BD se realizó haciendo uso de MySQL Workbench (ver Fig. 73).

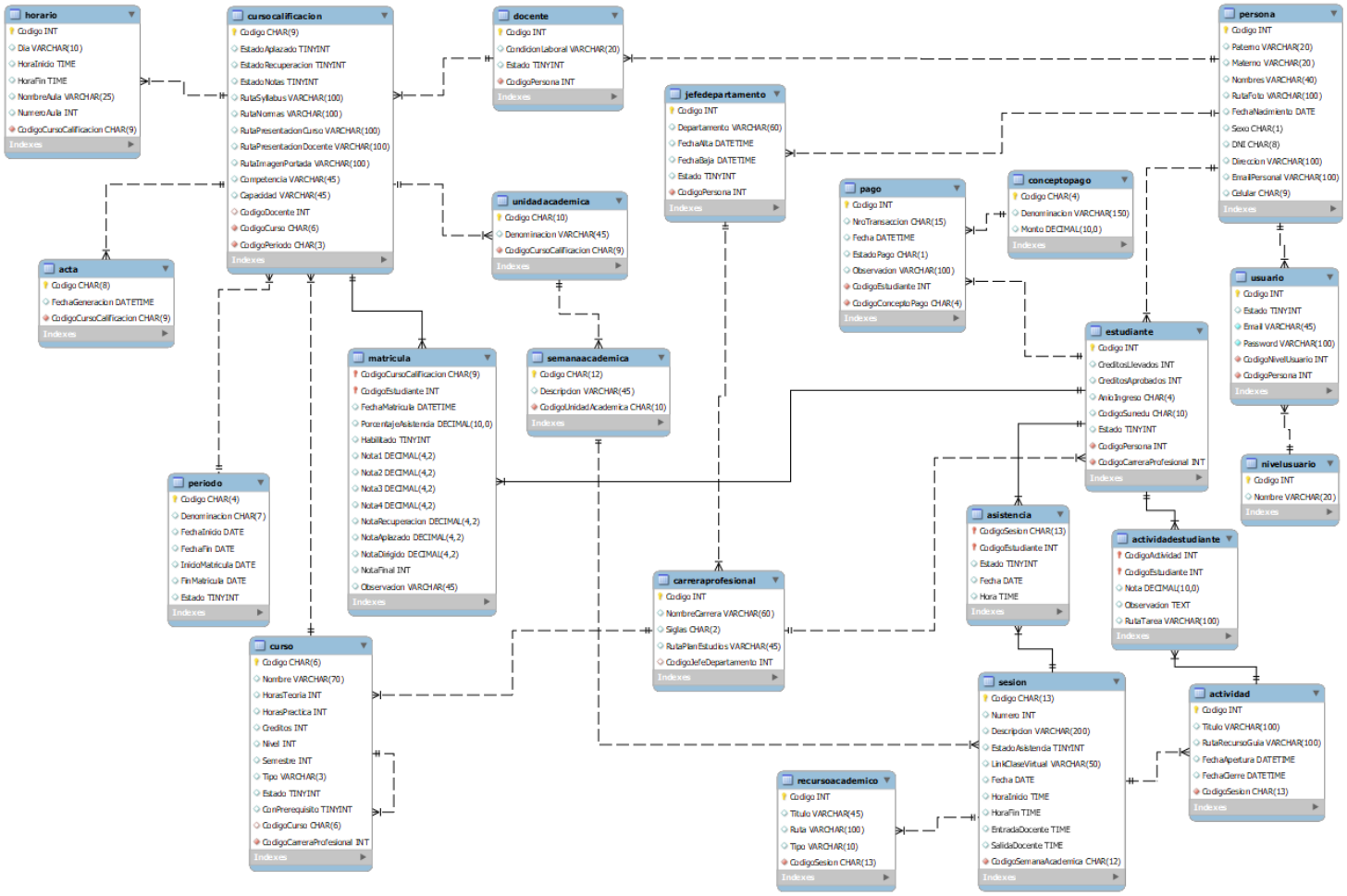

Fig. 73 Diagrama de base de datos

3.1.3.5 DIAGRAMA DE COMPONENTES

El diagrama de componentes es importante ya que permite representar de manera clara la estructura modular del sistema, sus relaciones y dependencias. Esta representación contribuye a una mejor comprensión del diseño, facilita el mantenimiento, la reutilización de componentes y mejora la planificación del desarrollo (Ver Fig. 74).

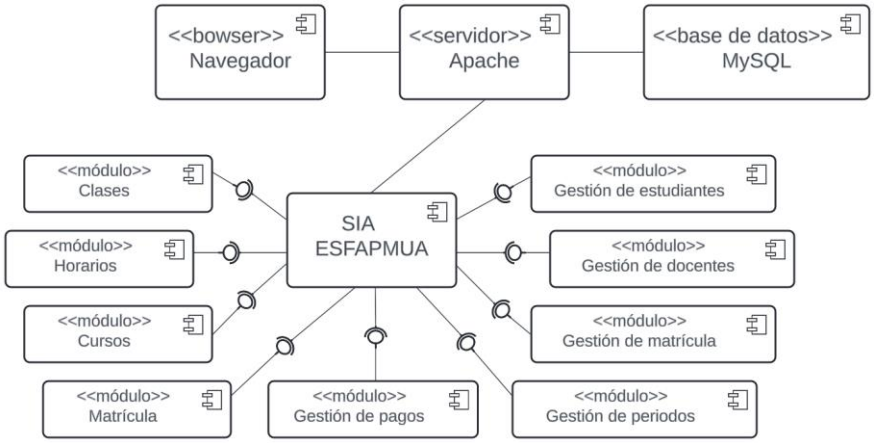

Fig. 74 Diagrama de componentes

3.1.3.6 DIAGRAMAS DE FLUJO FINALES

A continuación, se muestran los diagramas de flujo finales de los procesos inicialmente considerados, los cuales contemplan la intervención del sistema web desarrollado, mostrando una mejora en los tiempos al minimizar tareas manuales (Ver Fig. 75 y 76).

- **Gestión de matrícula**

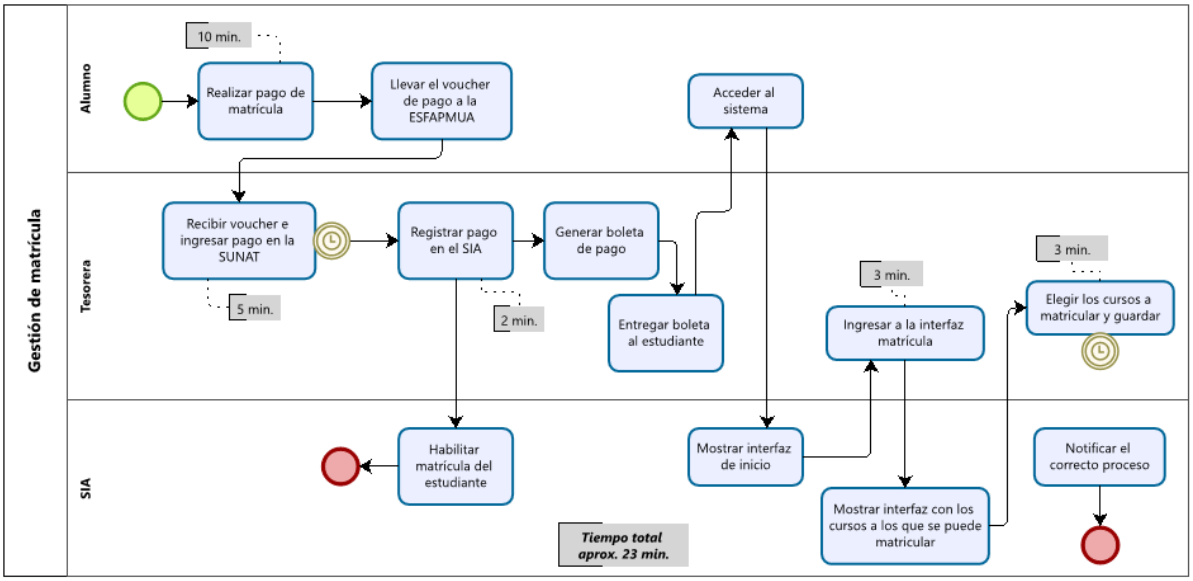

Fig. 75 Diagrama final del proceso "Gestión de matrícula"

- **Gestión de calificaciones por curso**

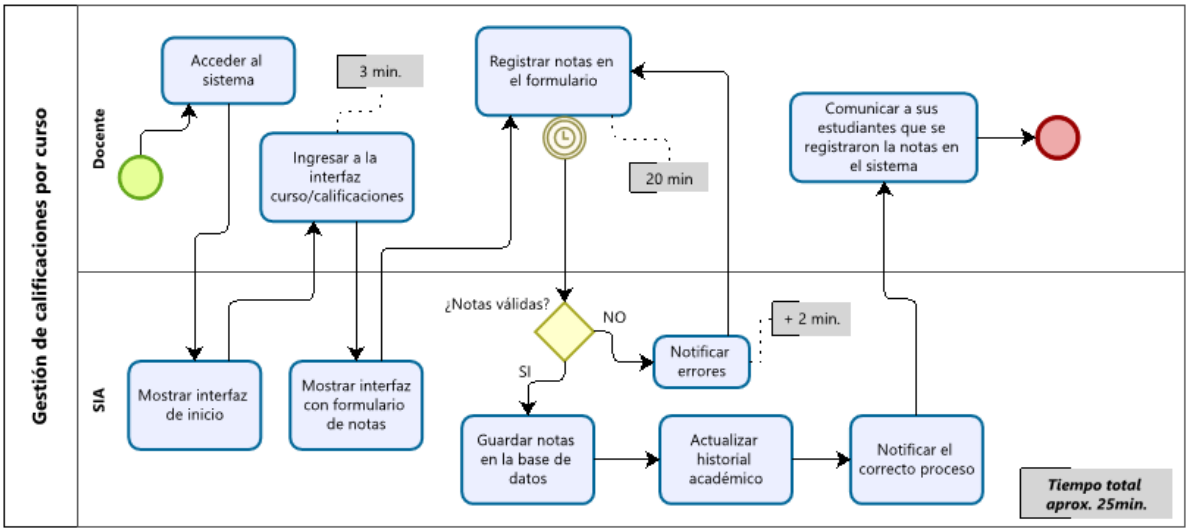

Fig. 76 Diagrama final del proceso "Gestión de calificaciones"

3.1.4 SEGUIMIENTO Y MEJORA CONTINUA

La implementación de un sistema de información web no concluye con su desarrollo inicial; por el contrario, requiere un proceso constante de seguimiento y mejora para asegurar su efectividad, adaptabilidad y alineación con las necesidades del usuario. En este contexto, la metodología Kanban se presenta como una herramienta ágil y eficiente para gestionar y optimizar continuamente dicho sistema, permite visualizar el flujo de trabajo mediante tableros divididos en columnas que representan las distintas etapas del proceso; esta representación visual facilita el seguimiento de cada tarea relacionada con el sistema, desde la corrección de errores hasta la implementación de nuevas funcionalidades, promoviendo la transparencia y la colaboración dentro del equipo.

Uno de los pilares fundamentales de Kanban es la mejora continua, a través de la revisión constante del flujo de trabajo, el equipo puede identificar cuellos de botella, tiempos de espera innecesarios y tareas repetitivas, lo que permite aplicar ajustes que incrementen la eficiencia. Además, el límite de trabajo en curso ayuda a mantener un ritmo de trabajo sostenible, evitando la sobrecarga de tareas y fomentando una entrega de valor constante. El uso de esta metodología en el seguimiento, en este caso de un sistema de información web, facilita una gestión visual, flexible y orientada a la mejora continua, esto se traduce en un sistema más robusto, ágil y alineado con las expectativas de los usuarios y de la organización.

Otros indicadores Kanban

- Throughput.

  Qué es: Número de tareas finalizadas en un período de tiempo determinado (día, semana, sprint, mes).

  Objetivo: Medir la capacidad del equipo.

  En la siguiente tabla se muestra el número de tareas completadas semanalmente durante catorce semanas de desarrollo y despliegue del sistema web académico, junto con observaciones relevantes que influyeron en el flujo de trabajo.

Tabla XLIX Throughput semanal

Semana	Tareas completadas	Observaciones
Semana 1	3	Primeras configuraciones del entorno de trabajo y desarrollo de la primera HU
Semana 2	1	Bloqueo por espera de información de parte de la institución por cambio de malla curricular
Semana 3	2	
Semana 4	1	Complejidad de la HU05
Semana 5	2	
Semana 6	2	
Semana 7	1	Complejidad de la HU12
Semana 8	2	
Semana 9	2	
Semana 10	2	
Semana 11	3	Optimización de rendimiento: mejorar consultas SQL
Semana 12	2	
Semana 13	2	Entrega parcial a la institución
Semana 14	1	Bloqueo por espera de IP estática

  Resultado: throughput = 2 tareas / semana aprox.

- Work Item Age (Edad de los Ítems de Trabajo)

  Qué es: Tiempo que lleva una tarea en el sistema desde que entró al flujo (o desde que empezó a trabajarse, según la definición).

Qué aporta: Detecta tareas que están "atascadas" o demoradas más de lo esperado.

Uso en el desarrollo del proyecto: El Work Item Age permitió medir el estado actual del flujo de trabajo, identificando en tiempo real aquellas tareas que llevaban demasiado tiempo en proceso sin finalizar. Gracias a esta métrica fue posible detectar retrasos y posibles bloqueos en etapas específicas, priorizar la atención de ítems que estaban "envejeciendo" más de lo esperado y mantener el control sobre los tiempos de entrega comprometidos. De esta manera, se facilitó la visibilidad del avance, la prevención de cuellos de botella y la toma de decisiones oportunas para asegurar un flujo de trabajo más estable y predecible.

La HU05, relacionada con la interfaz de registro de matrículas presentó un **Work Item Age** de 2 días en la columna *Design*, mientras que el promedio era de 1 día. El retraso se debió a que los prototipos en Figma requirieron varias iteraciones para ajustarse a las observaciones de los docentes. La métrica permitió detectar el envejecimiento de la tarea y priorizar su conclusión, evitando que afectara el inicio del desarrollo backend.

- Blockers (Elementos Bloqueados)

  Qué es: Registro de tareas que no pueden avanzar por falta de información, dependencias o problemas externos. El tiempo que la tarea permanezca bloqueada impactará el Work Item Age y puede convertirse en un cuello de botella si no se resuelve pronto.

  Qué aporta: Ayuda a identificar los factores que más afectan la continuidad del flujo.

  Tarea bloqueada: *Desplegar el sistema académico en el servidor institucional*

  Situación:

  El equipo de desarrollo tenía planificado realizar el despliegue del sistema en un servidor local, configurando el acceso mediante una **IP estática** para que los usuarios pudieran ingresar de forma confiable desde la red institucional.

  Problema:

  El proveedor de internet, a pesar de haber ofrecido una IP estática, presentó demoras en la asignación y la dirección configurada cambiaba constantemente. Esto ocasionó que el sistema no fuese accesible de manera estable.

  Consecuencia:

  - La tarea de despliegue quedó **bloqueada**, ya que dependía de un recurso externo (el proveedor).

- Ningún desarrollador podía avanzar con pruebas de acceso remoto ni con la capacitación a docentes hasta resolver el problema.
- La tarjeta en el tablero Kanban se marcó como **Blocker**, reflejando que estaba en espera de factores externos.

- Escaped Defects (Defectos Escapados)

  Qué es: Número de errores o retrabajos detectados después de que una tarea fue marcada como terminada.

  Qué aporta: Mide la calidad del trabajo y el impacto de la falta de validaciones previas.

  Tarea entregada: Módulo de generación de constancias de estudios.

  Situación:

  El equipo finalizó el desarrollo y pruebas del módulo, validando que las constancias se generaban correctamente en PDF con los datos del estudiante (nombre, ciclo, promedio, etc.). La tarea pasó a la columna Done y fue considerada terminada.

  Problema detectado después:

  Durante el uso real por parte de los docentes, se reportó que en algunos casos los apellidos de los estudiantes con caracteres especiales (por ejemplo: "Ñ" o tildes) no se mostraban correctamente en el PDF. Este error no fue detectado en las pruebas iniciales porque no se probaron suficientes casos con caracteres especiales.

  Consecuencia:

  - El defecto se considera un escaped defect, ya que apareció en producción tras la entrega.
  - Se tuvo que crear una nueva tarjeta en el backlog para corregir la codificación de caracteres y asegurar la correcta generación de documentos.
  - El indicador de escaped defects permite medir cuántos errores no fueron detectados en Testing y lograron llegar al usuario final.

3.2 TRATAMIENTO, ANÁLISIS DE DATOS Y PRESENTACIÓN DE RESULTADOS

3.2.1 TRATAMIENTO Y ANÁLISIS DE DATOS

Para el procesamiento de los datos, se emplearon herramientas de gestión y análisis numérico basadas en hojas de cálculo, integradas dentro de una suite ofimática ampliamente utilizada. Estas herramientas permitieron la creación de tablas y representaciones gráficas que facilitaron la sistematización y el análisis eficiente de la información recolectada. Los datos fueron organizados en matrices, clasificados y codificados según categorías previamente definidas. Asimismo, se hizo uso de un programa especializado en análisis estadístico, diseñado para la recopilación, validación y procesamiento de datos cuantitativos, lo cual permitió una exploración más profunda de los resultados mediante pruebas estadísticas y procedimientos descriptivos.

El análisis de datos cuantitativos se emplea con el objetivo de identificar relaciones y diferencias relevantes respecto a la hipótesis planteada, centrándose en la información recopilada durante el estudio. Según la naturaleza del proyecto, se determina la escala adecuada para las variables involucradas. Posteriormente, se evalúa la distribución de los datos para comprobar su ajuste a una distribución específica, en el presente estudio se consideró utilizar Shapiro-Wilk. En función de estos resultados, se seleccionan las pruebas estadísticas correspondientes, ya sean paramétricas o no paramétricas, para asegurar la validez y precisión en la interpretación de los datos.

- **Variable independiente**

Dimensión: calidad de software

Para esta variable, se aplicó un cuestionario de forma anónima a expertos en desarrollo de software. El resumen de los resultados se presenta en la Tabla L, donde se muestra el porcentaje de participantes que consideran que el software evaluado cumple con cada una de las características que reflejan su calidad.

Tabla L Resumen de resultados del cuestionario referente a la calidad de software

N°	Características	Valor cualitativo				
		Totalmente de acuerdo	De acuerdo	Indiferente	En desacuerdo	Totalmente en desacuerdo
1	Adecuación funcional	100%	-	-	-	-

N°	Características	Valor cualitativo				
		Totalmente de acuerdo	De acuerdo	Indiferente	En desacuerdo	Totalmente en desacuerdo
2	Eficiencia de desempeño	33%	67%	-	-	-
3	Usabilidad	67%	33%	-	-	-
4	Fiabilidad	-	100%	-	-	-
5	Seguridad	67%	33%	-	-	-
6	Mantenibilidad	-	100%	-	-	-
7	Portabilidad	67%	33%	-	-	-

- **Variable independiente**

Dimensión: tiempo

Mediante el uso de una ficha de observación, se registraron los tiempos requeridos para llevar a cabo las actividades correspondientes a ciertos procesos dentro de la gestión académica realizada por los docentes. En la etapa inicial, se calculó el número requerido de ciclos de observación por indicador siguiendo los lineamientos del método de observación sistemática de General Electric (método del cociente de la media) [60].

Tabla LI Número de ciclos de observación según el método General Electric

Indicador	Tiempo (min.)	Número de observaciones
P1: Registro de asistencias por clase	6	10
P2: Creación de actividades	15	8
P3: Evaluación de actividades	76	3
P4: Registro y publicación de notas	20	5

Para la recolección de datos en esta investigación se aplicó la ficha de observación durante un único ciclo a todos los participantes de la muestra, conformada por 29 personas. Esta decisión metodológica se fundamenta en que la muestra es suficientemente representativa del grupo objetivo, lo que permite obtener una visión integral y confiable del comportamiento y desarrollo del proceso evaluado en la institución educativa.

Asimismo, se consideró que un solo ciclo de observación es adecuado debido a la naturaleza estable del proceso, que no presenta variaciones significativas a lo largo del tiempo. De esta forma, se garantiza la obtención de datos pertinentes y relevantes para el análisis, evitando redundancias y optimizando el tiempo y recursos disponibles para la investigación.

Se reconoce que aplicar la ficha en múltiples ciclos podría ofrecer una perspectiva más amplia sobre la evolución del proceso; sin embargo, dadas las características y objetivos específicos de este estudio, el uso de un único ciclo resulta suficiente para cumplir con los fines propuestos.

En la Tabla LII se presentan los resultados del estudio comparativo sobre el tiempo requerido para la ejecución de distintas actividades propias de la labor docente. Dichas actividades fueron cronometradas en dos modalidades: primero, cuando se realizan de manera manual, siguiendo los procedimientos tradicionales; y luego, cuando se efectúan mediante el uso del sistema de información web desarrollado. El registro del tiempo se llevó a cabo con cada uno de los 29 docentes participantes, a fin de obtener un promedio representativo en minutos por proceso. El registro de los tiempos se hizo en minutos, calculando posteriormente un promedio por cada proceso. Esta comparación permite evidenciar de manera cuantitativa la diferencia entre ambos métodos.

Tabla LII Tiempos promedio pre y post test en los procesos pertenecientes a la dimensión tiempo

Docentes	**Proceso 1**		**Proceso 2**		**Proceso 3**		**Proceso 4**	
	Pre test	**Post test**	**Pre test**	**Post test**	**Pre test**	**Post test**	**Pre test**	**Post test**
D1	7	5	16	15	77	69	21	11
D2	6	6	14	14	75	67	19	13
D3	5	5	16	16	74	65	22	10
D4	5	5	15	15	78	68	20	12
D5	6	5	14	14	76	71	18	11
D6	5	5	16	15	75	66	21	13
D7	7	7	14	14	77	69	19	10
D8	6	6	17	15	74	67	22	12
D9	5	5	15	14	78	70	20	11
D10	7	6	14	14	76	68	18	13
D11	5	5	16	15	75	71	21	10
D12	7	7	14	14	77	68	19	12
D13	5	5	16	15	74	66	22	11
D14	6	5	15	14	78	67	20	13
D15	7	7	14	14	76	70	18	10
D16	5	6	16	15	75	68	21	12
D17	6	5	14	14	77	71	19	11
D18	5	5	16	15	74	66	22	13
D19	6	6	15	15	78	69	20	10
D20	7	6	14	14	76	67	18	12
D21	6	5	16	15	75	70	21	11
D22	7	5	14	14	77	68	19	13
D23	6	5	17	16	74	71	22	10
D24	7	6	15	15	78	67	20	12
D25	8	6	15	15	76	69	18	11
D26	5	5	16	15	75	67	21	13
D27	7	6	14	14	77	70	19	10
D28	6	5	17	15	74	68	22	12
D29	6	5	15	14	78	71	20	11
Promedio	6.07	5.52	15.17	14.62	76.00	68.41	20.07	11.48

Dimensión: Fiabilidad

A través de una ficha de análisis documental, se registraron aportes y observaciones relacionadas con posibles errores en el cálculo de promedios, inconsistencias en la asignación de notas y faltantes en los registros, con el fin de contribuir a la precisión del proceso evaluativo. Se revisaron los registros de calificaciones de 29 docentes, identificando situaciones que evidenciaron la falta de fiabilidad en los datos. El resumen de estas inconsistencias y casos relevantes en el pre y post test se presenta en la Tabla LIII.

Tabla LIII Número de casos inconsistentes referentes a la dimensión fiabilidad

Docentes	Pre test		Post test	
	N° Casos	Observaciones	N° Casos	Observaciones
D1	0		0	
D2	0		0	
D3	0		0	
D4	0		0	
D5	1	Mal cálculo en promedio de notas de un estudiante	0	
D6	0		0	
D7	0		0	
D8	0		0	
D9	0		0	
D10	0		0	
D11	0		0	
D12	0		0	
D13	0		0	
D14	2	Nota no correspondiente de un estudiante	0	
D15	0		0	
D16	0		0	
D17	0		0	
D18	0		0	
D19	0		0	
D20	1	Mal cálculo en promedio de notas de un estudiante	0	
D21	0		0	
D22	0		0	
D23	1	Nota faltante de un estudiante	0	
D24	0		0	
D25	0		0	
D26	0		0	
D27	0		0	
D28	1	Nota desactualizada de un estudiante	0	
D29	0		0	
Total	**6**		**0**	

Dimensión: calidad de software

Mediante una ficha de cotejo, se registró la cantidad de días de retraso transcurridos entre la fecha de finalización del periodo evaluativo y la realización de actividades que deben ser ejecutadas por los docentes. Esto permitió asegurar la transparencia de los procesos y respaldar el cumplimiento de los plazos establecidos por la institución. Los datos correspondientes a los 29 docentes evaluados se presentan en la Tabla LIV.

Tabla LIV Número de días de retraso en ejecución de actividades referentes a la dimensión transparencia

Docentes	**Indicador 1**		**Indicador 2**	
	Número de días de retraso con respecto a la fecha de publicación de notas finales.		Número de días de retraso con respecto a la fecha de presentación de actas de notas.	
	Pre test	**Post test**	**Pre test**	**Post test**
D1	0	0	0	0
D2	1	0	0	0
D3	0	0	0	0
D4	0	0	0	0
D5	0	0	1	0
D6	0	0	0	0
D7	1	0	2	0
D8	0	0	0	0
D9	0	0	0	0
D10	0	0	0	0
D11	0	0	0	0
D12	0	0	0	0
D13	0	0	0	0
D14	2	0	3	0
D15	0	0	0	0
D16	0	0	0	0
D17	0	0	0	0
D18	0	0	0	0
D19	0	0	0	0
D20	0	0	0	0
D21	0	0	0	0
D22	0	0	0	0
D23	1	0	1	0
D24	0	0	0	0
D25	0	0	0	0
D26	0	0	0	0
D27	1	0	1	0
D28	2	0	1	0
D29	0	0	0	0

Docentes	Indicador 1		Indicador 2	
	Número de días de retraso con respecto a la fecha de publicación de notas finales.		Número de días de retraso con respecto a la fecha de presentación de actas de notas.	
	Pre test	**Post test**	**Pre test**	**Post test**
Número de docentes que no cumplieron la fecha límite	**6**	**0**	**6**	**0**
Número de días de retraso	**2**	**0**	**3**	**0**

3.2.2 PRESENTACIÓN DE RESULTADOS

Después de aplicar los instrumentos de recolección de datos para medir el impacto de un sistema de información web en la gestión académica de la Escuela Superior de Formación Artística Mario Urteaga Alvarado de la ciudad de Cajamarca; se ha obtenidos resultados de una mejora significativa como se muestran en las siguientes figuras para la variable independiente y dependiente.

- **Variable independiente**

Dimensión calidad de software

Con el objetivo de evaluar la calidad del sistema web desarrollado, se aplicó un cuestionario a un grupo de expertos en desarrollo de software, quienes valoraron el sistema en función de las características establecidas por el estándar ISO/IEC 25010. A continuación, se presentan los resultados obtenidos, los cuales ofrecen una visión crítica y técnica sobre el cumplimiento de los atributos de calidad del software.

Adecuación funcional. Como se muestra en la Fig. 77, el 100% de los expertos están "totalmente de acuerdo" en que el sistema cumple con los requerimientos funcionales y no funcionales descritos inicialmente.

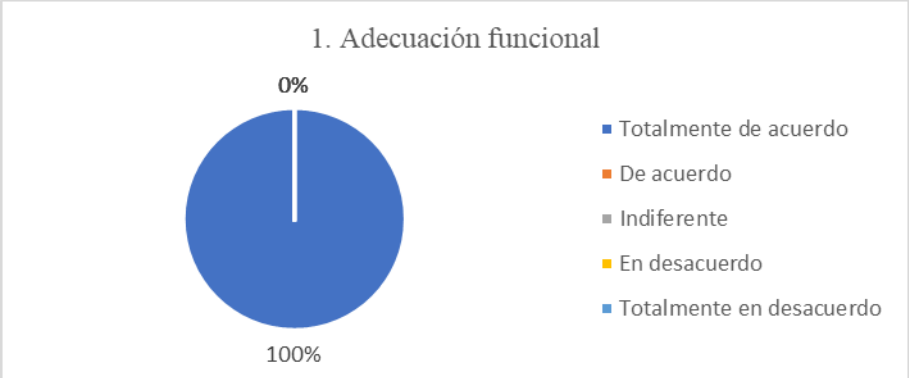

Fig. 77 Gráfico circular-calidad de software-adecuación funcional

En base a la puntuación según la escala de Likert y al número de participantes se calculó el porcentaje cuantitativo correspondiente a cada característica. En particular, la característica de "adecuación funcional" obtuvo un 100%.

Eficiencia de desempeño. Como se muestra en la Fig. 78, el 33% de los expertos consultados indicaron estar "totalmente de acuerdo" con que el sistema cumple con esta característica, destacando su buen tiempo de respuesta y uso adecuado de los recursos. Por otro lado, el 67% manifestó estar "de acuerdo", aunque señalaron que existen oportunidades de mejora, especialmente en la optimización de consultas SQL y en el manejo de archivos del lado del cliente, lo cual contribuiría a una mayor eficiencia general del sistema.

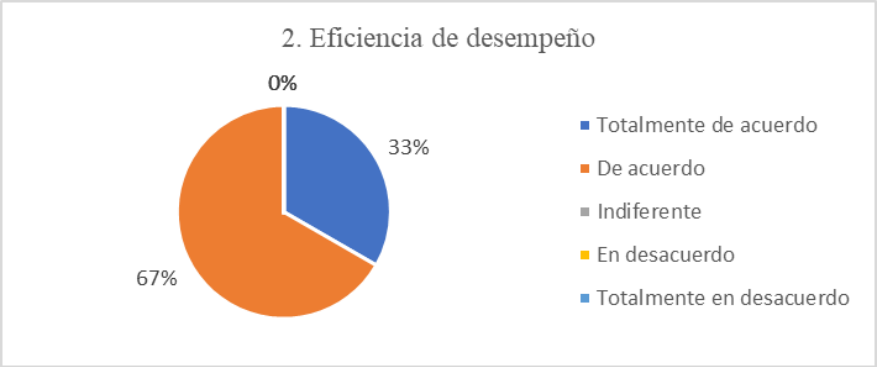

Fig. 78 Gráfico circular-calidad de software-eficiencia de desempeño

El porcentaje cuantitativo correspondiente a la característica de "eficiencia de desempeño" obtuvo un 87%.

Usabilidad. Según los resultados mostrados en la Fig. 79, el 67% de los expertos consultados están "totalmente de acuerdo" en que el sistema cumple con esta característica, destacando que presenta una curva de aprendizaje baja, lo cual facilita su adopción por parte de los usuarios. El 33% restante indicó estar "de acuerdo", aunque sugirió que algunos componentes de la interfaz podrían reorganizarse para mejorar aún más la experiencia de usuario.

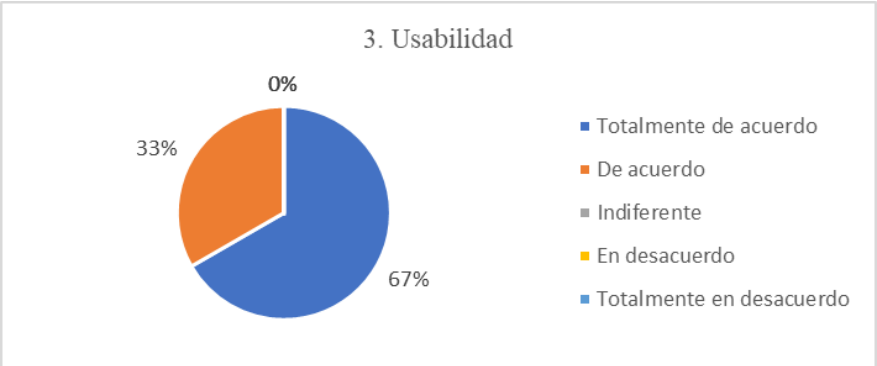

Fig. 79 Gráfico circular-calidad de software-usabilidad

El porcentaje cuantitativo correspondiente a la característica de “usabilidad” obtuvo un 93%.

Fiabilidad. Tal como se muestra en la Fig. 80, el 100% de los expertos consultados están “de acuerdo” en que el sistema cumple con esta característica. No obstante, señalaron que el aspecto de la disponibilidad puede mejorarse, dado que actualmente el sistema se encuentra alojado en un servidor físico susceptible a fallos por cortes en el suministro eléctrico, lo cual representa una posible limitación en entornos de uso continuo.

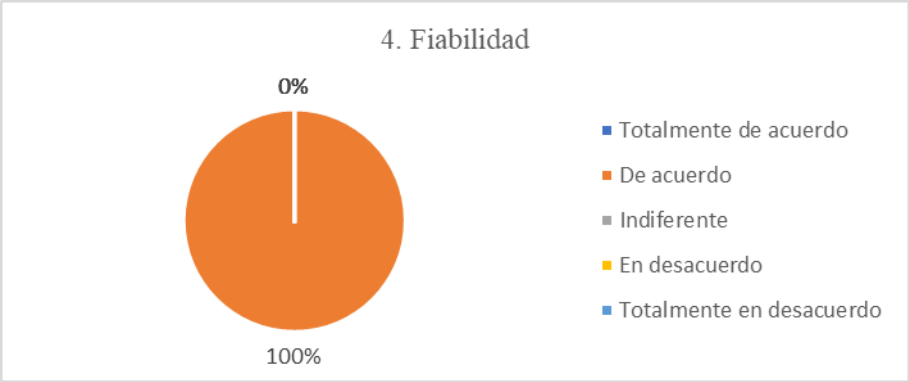

Fig. 80 Gráfico circular-calidad de software-fiabilidad

El porcentaje cuantitativo correspondiente a la característica de “fiabilidad” obtuvo un 80%.

Seguridad. Como se observa en la Figura 81, el 67% de los expertos consultados están “totalmente de acuerdo” en que el sistema cumple con esta característica, destacando el buen manejo de la accesibilidad y la protección de datos sensibles. El 33% restante manifestó estar “de acuerdo”, aunque sugirió mejorar la seguridad en el manejo de variables en la URL, recomendando que estas también sean cifradas para reforzar la protección frente a posibles vulnerabilidades.

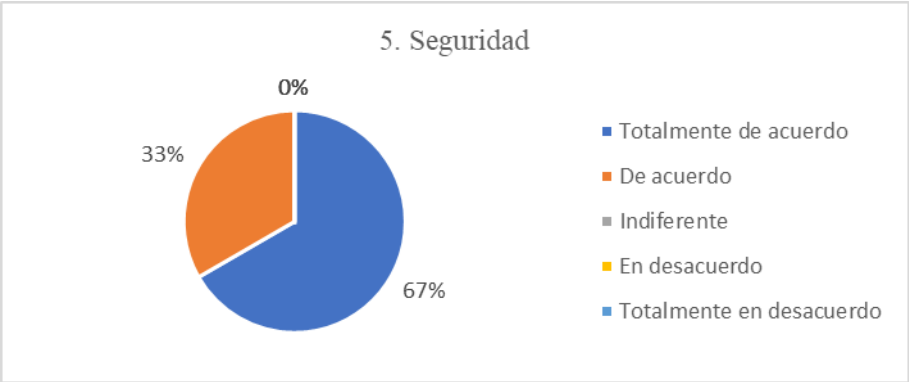

Fig. 81 Gráfico circular-calidad de software-seguridad

El porcentaje cuantitativo correspondiente a la característica de “seguridad” obtuvo un 93%.

Mantenibilidad. De acuerdo con los resultados presentados en la Fig. 82, el 100% de los expertos están "de acuerdo" en que el sistema cumple con esta característica. Sin embargo, señalaron que la documentación técnica puede ser más completa y que es recomendable fortalecer las prácticas de codificación, con el fin de facilitar futuras actualizaciones, correcciones y ampliaciones del sistema.

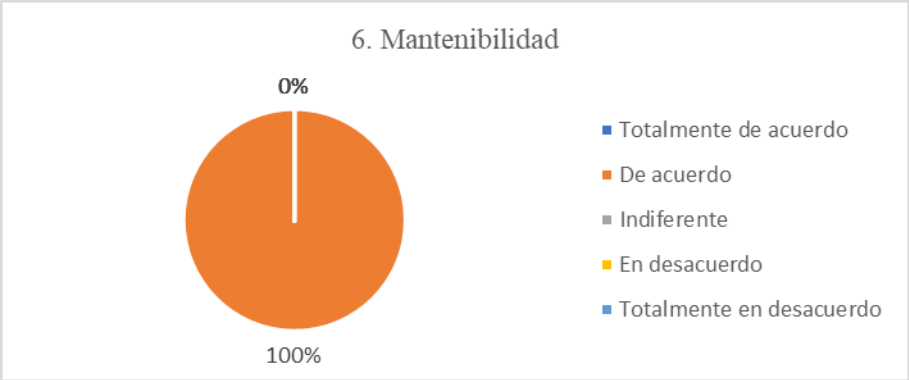

Fig. 82 Gráfico circular-calidad de software-mantenibilidad

El porcentaje cuantitativo correspondiente a la característica de "mantenibilidad" obtuvo un 80%.

Portabilidad. Como se muestra en la Fig. 83, el 67% de los expertos están "totalmente de acuerdo" en que el sistema cumple con esta característica, ya que consideran que ha sido desarrollado con tecnologías que favorecen su portabilidad entre distintos entornos. Por su parte, el 33% restante expresó estar "de acuerdo", aunque señalaron que algunas interfaces podrían mejorar su adaptabilidad en dispositivos móviles, lo que permitiría una experiencia de uso más uniforme en diversas plataformas.

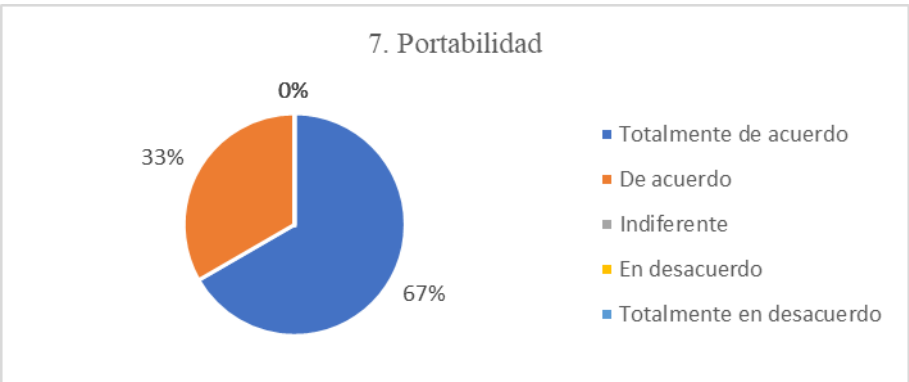

Fig. 83 Gráfico circular-calidad de software-portabilidad

El porcentaje cuantitativo correspondiente a la característica de "portabilidad" obtuvo un 93%.

- **Variable dependiente**

Dimensión tiempo

En la Tabla LV se presenta el porcentaje de mejora de cada proceso considerado dentro de la gestión académica realizada por los docentes, en relación con la dimensión tiempo. Dicho porcentaje fue calculado a partir de los datos obtenidos en la aplicación de un pre test y un post test, realizados antes y después de la implementación del sistema de información web.

Tabla LV Porcentaje de mejora en los procesos relacionados a la dimensión tiempo

ID	Proceso	Pre test (min.)	Post test (min.)	Reducción (min.)	% Mejora
P1	Registro de asistencias por clase	5.82	5.32	0.50	9%
P2	Creación de actividades	14.87	14.32	0.55	4%
P3	Evaluación de actividades	75.87	67.49	8.38	11%
P4	Registro y publicación de notas	20.15	9.87	10.28	51%

En la Fig. 84 se puede observar una comparativa gráfica de la mejora de tiempos pre y post test de los procesos considerados en esta dimensión.

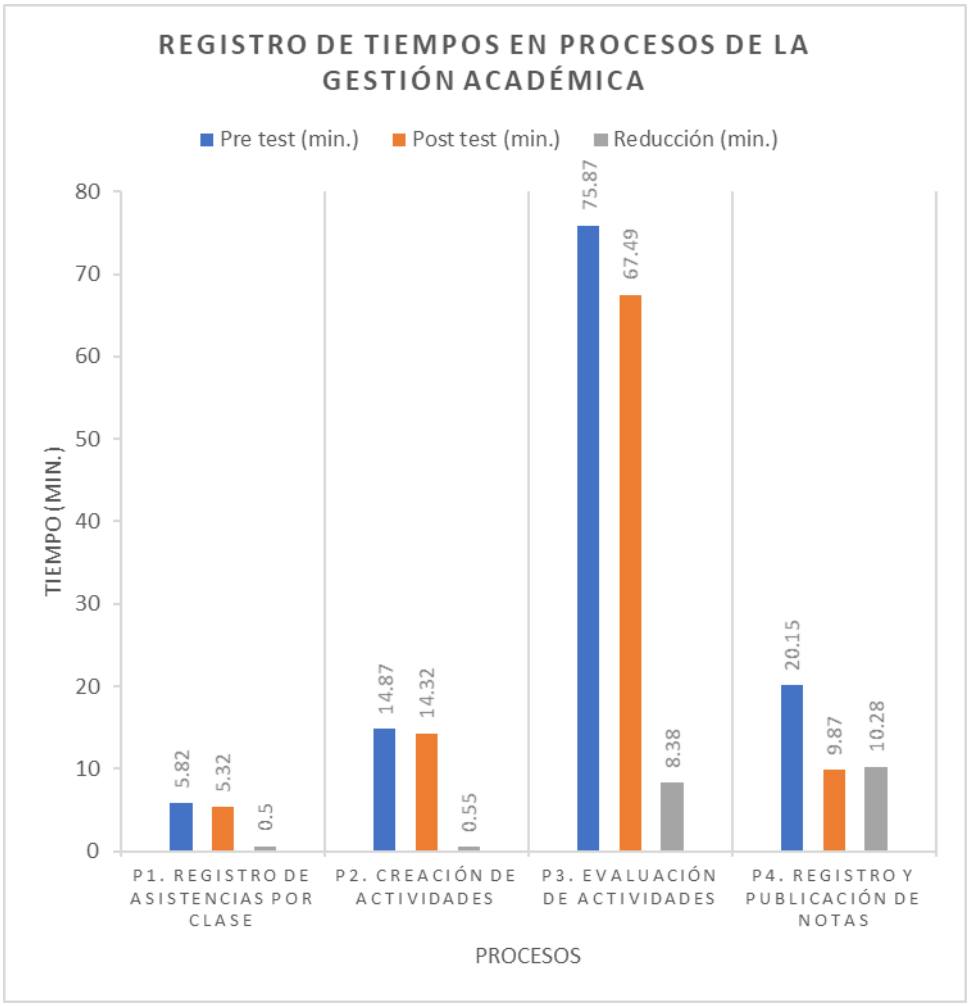

Fig. 84 Registro de tiempos en procesos de la gestión académica

Contrastación de hipótesis

- Prueba de normalidad de datos

Para la determinación de la prueba de hipótesis, se realizó un análisis preliminar mediante la aplicación de la prueba de normalidad de Shapiro-Wilk a los datos recolectados. Este procedimiento se efectuó considerando un nivel de confianza del 95% y un margen de error del 5%, con el fin de establecer si los datos cumplen con el supuesto de normalidad requerido para las pruebas paramétricas.

Las hipótesis formuladas para la prueba de normalidad fueron las siguientes:

- ✓ H0: Los datos obtenidos presentan una distribución normal
- ✓ H1: Los datos obtenidos no presentan una distribución normal

Pruebas de normalidad

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
P1	.290	29	<.001	.829	29	<.001
P2	.326	29	<.001	.739	29	<.001
P3	.173	29	.026	.955	29	.240
P4	.110	29	.200*	.967	29	.491

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Fig. 85 Prueba de normalidad de datos relacionados a la dimensión tiempo

Conforme a los resultados obtenidos en la prueba de normalidad (Fig. 85), los conjuntos de datos correspondientes a los procesos P1 y P2 presentan valores de significancia (p) inferiores a 0.05, lo cual indica que no se cumple el supuesto de normalidad, rechazándose la hipótesis nula y aceptándose la hipótesis alternativa. En cambio, para los procesos P2 y P3, los valores de p son superiores a 0.05, lo que sugiere que los datos no difieren significativamente de una distribución normal, por lo que no se rechaza la hipótesis nula.

En función de estos resultados, se optó por aplicar la prueba T de Student para aquellos conjuntos de datos que cumplen con el supuesto de normalidad, y la prueba no paramétrica de Wilcoxon para aquellos que no lo cumplen.

- Formulación de hipótesis

  H_1: El uso de un sistema de información web disminuye el tiempo que toma realizar tareas pertenecientes a la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

  H_0: El uso de un sistema de información web no disminuye el tiempo que toma realizar tareas pertenecientes a la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

  Nivel de significancia: $\alpha = 0.05$

- Valor estadístico

  En las Fig. 86 y 87 se presentan los resultados del procesamiento de datos correspondientes a muestras relacionadas no paramétricas asociadas a los procesos P1 y P2, respectivamente.

P1pre, P1pos

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas

N total	29
Estadístico de prueba	6.500
Error estándar	16.545
Estadístico de prueba estandarizado	-3.234
Sig. asintótica (prueba bilateral)	.001

Fig. 86 Prueba de rangos con signo de Wilcoxon para muestras relacionadas P1

P2pre, P2pos

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas

N total	29
Estadístico de prueba	.000
Error estándar	14.761
Estadístico de prueba estandarizado	-3.557
Sig. asintótica (prueba bilateral)	<.001

Fig. 87 Prueba de rangos con signo de Wilcoxon para muestras relacionadas P2

En la Fig. 88 se presentan los resultados del procesamiento de datos correspondientes a muestras relacionadas paramétricas asociadas a los procesos P3 y P4.

Prueba de muestras emparejadas

		Diferencias emparejadas							Significación	
		Media	Desv. estándar	Media de error estándar	95% de intervalo de confianza de la diferencia Inferior	Superior	t	gl	P de un factor	P de dos factores
Par 1	P3pre - P3pos	7.586	1.881	.349	6.871	8.302	21.722	28	<.001	<.001
Par 2	P4pre - P4pos	8.586	1.803	.335	7.900	9.272	25.643	28	<.001	<.001

Fig. 88 Prueba T de Student para muestras relacionadas P3 y P4

- Toma de decisión

  En las Fig. 85, 86 y 87 se observa que el p-valor obtenido es menor a 0.001; lo cual es inferior al nivel de significancia establecido (0.05). En consecuencia, se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alternativa (H_1). Esto permite concluir que, el uso de un sistema de información web disminuye el tiempo que toma realizar tareas pertenecientes a la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

Dimensión fiabilidad

En la tabla LIII y en la Fig. 89, se puede observar una reducción total en los casos relacionados a errores en registro de notas, pasando de 6 casos en la medición inicial (pre test) a 0 en la medición final (post test). Esta disminución representa una mejora en esta categoría, lo que sugiere un efecto positivo atribuible a la implementación del sistema de información web.

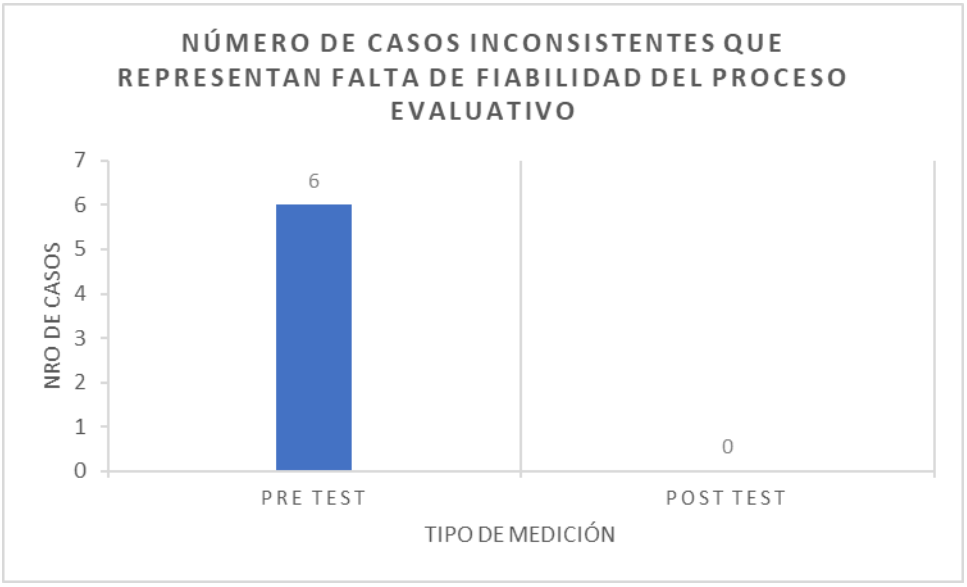

Fig. 89 Número de casos que representan falta de fiabilidad dentro de la gestión académica

Contrastación de hipótesis

- Prueba de normalidad de datos

  Las hipótesis formuladas para la prueba de normalidad fueron las siguientes:

- ✓ H0: Los datos obtenidos presentan una distribución normal
- ✓ H1: Los datos obtenidos no presentan una distribución normal

Pruebas de normalidad

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
I1	.485	29	<.001	.500	29	<.001

a. Corrección de significación de Lilliefors

Fig. 90 Prueba de normalidad de datos relacionados a la dimensión fiabilidad

Conforme a los resultados obtenidos en la prueba de normalidad (Fig. 90), los conjuntos de datos presentan valores de significancia (p) inferiores a 0.05, lo cual indica que no se cumple el supuesto de normalidad, rechazándose la hipótesis nula y aceptándose la hipótesis alternativa.

En función de estos resultados, se optó por la prueba no paramétrica de Wilcoxon para ya que los datos no cumplen el supuesto de normalidad.

- Formulación de hipótesis

H_1: El uso de un sistema de información web mejora la fiabilidad de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

H_0: El uso de un sistema de información web no mejora la fiabilidad de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

Nivel de significancia: $\alpha = 0.05$

- Valor estadístico

En la Fig. 91 se presentan los resultados del procesamiento de datos correspondientes a muestras relacionadas no paramétricas asociadas.

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas

N total	29
Estadístico de prueba	.000
Error estándar	4.287
Estadístico de prueba estandarizado	-2.449
Sig. asintótica (prueba bilateral)	.014

Fig. 91 Prueba de rangos con signo de Wilcoxon para muestras relacionadas - d_fiabilidad

- Toma de decisión

  En la Fig. 90 se observa que el p-valor obtenido es 0.014, lo cual es inferior al nivel de significancia establecido (0.05). En consecuencia, se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alternativa (H_1). Esto permite concluir que, el uso de un sistema de información web mejora la fiabilidad de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

Dimensión transparencia

En la tabla LIV y en la Fig. 92, se puede observar una reducción total en el número de días de retraso con respecto a la fecha de publicación de notas finales (I1), así como en el número de días de retraso con respecto a la fecha de presentación de actas de notas (I2), esto entre la medición inicial (pre test) y la medición final (post test). Este retraso podría ser un indicador de problemas de gestión o falta de responsabilidad en cuanto a informar a los interesados, sin embargo, esta disminución representa una mejora en esta categoría, lo que sugiere un efecto positivo atribuible a la implementación del sistema de información web.

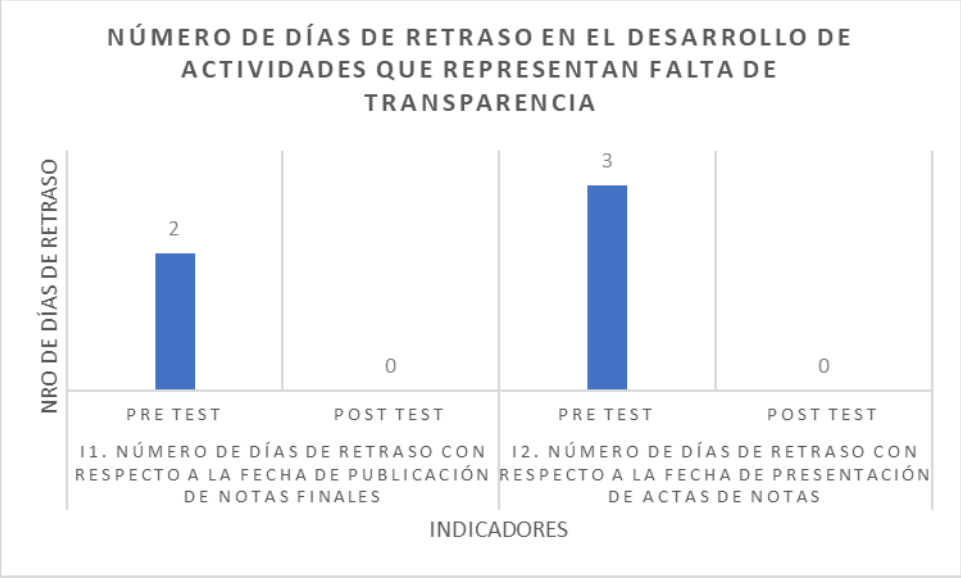

Fig. 92 Número de días de retraso que representan falta de trasparencia dentro de la gestión académica.

Contrastación de hipótesis

- Prueba de normalidad de datos

  Las hipótesis formuladas para la prueba de normalidad fueron las siguientes:

  - ✓ H0: Los datos obtenidos presentan una distribución normal
  - ✓ H1: Los datos obtenidos no presentan una distribución normal

Pruebas de normalidad

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
I1	.473	29	<.001	.524	29	<.001
I2	.462	29	<.001	.508	29	<.001

a. Corrección de significación de Lilliefors

Fig. 93 Prueba de normalidad de datos relacionados a la dimensión transparencia

Conforme a los resultados obtenidos en la prueba de normalidad (Fig. 93), los conjuntos de datos presentan valores de significancia (p) inferiores a 0.05, lo cual indica que no se cumple el supuesto de normalidad, rechazándose la hipótesis nula y aceptándose la hipótesis alternativa. En función de estos resultados, se optó por la prueba no paramétrica de Wilcoxon para ya que los datos no cumplen el supuesto de normalidad.

- Formulación de hipótesis

H_1: El uso de un sistema de información web refuerza la transparencia de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

H_0: El uso de un sistema de información web no refuerza la transparencia de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

Nivel de significancia: $\alpha = 0.05$

- Valor estadístico

En las Fig. 94 y 95 se presentan los resultados del procesamiento de datos correspondientes a muestras relacionadas no paramétricas asociadas a los indicadores I1 e I2, respectivamente.

I1pre, I1pos

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas

N total	29
Estadístico de prueba	.000
Error estándar	4.623
Estadístico de prueba estandarizado	-2.271
Sig. asintótica (prueba bilateral)	.023

Fig. 94 Prueba de rangos con signo de Wilcoxon para muestras relacionadas I1

I2pre, I2pos

Resumen de prueba de rangos con signo de Wilcoxon para muestras relacionadas

N total	29
Estadístico de prueba	.000
Error estándar	4.637
Estadístico de prueba estandarizado	-2.264
Sig. asintótica (prueba bilateral)	.024

Fig. 95 Prueba de rangos con signo de Wilcoxon para muestras relacionadas I2

- Toma de decisión

  En la Fig. 93 y Fig. 94 se observa que los p-valor obtenidos son 0.023 y 0.024 respectivamente, los cuales son inferiores al nivel de significancia establecido (0.05). En consecuencia, se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alternativa (H_1). Esto permite concluir que, el uso de un sistema de información web refuerza la transparencia de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

El rendimiento académico se encuentra estrechamente vinculado con la eficiencia de los procesos de gestión académica, pues la disponibilidad de información precisa y oportuna es un factor clave para el monitoreo del progreso estudiantil y la adopción de medidas pedagógicas oportunas. En este sentido, el sistema de información web desarrollado contribuye significativamente a este indicador al ofrecer mejoras cuantificables. Por ejemplo, la digitalización de las matrículas permite reducir hasta en un 34.28% el tiempo de registro por estudiante, lo que agiliza el inicio de clases, evita retrasos en el calendario académico y disminuye errores en la inscripción de asignaturas, factores que impactan directamente en el aprovechamiento del tiempo de aprendizaje y en la continuidad curricular. Asimismo, la automatización del control de asistencias y calificaciones disminuye en gran porcentaje los errores de digitación en comparación con el uso de formatos físicos y hojas de cálculo dispersas. La centralización de datos en una plataforma digital facilita el acceso inmediato a información relevante tanto para docentes como para autoridades académicas, lo que fortalece la toma de decisiones basadas en evidencias. De esta manera, el sistema no solo optimiza la transparencia y equidad en los procesos evaluativos, sino que también contribuye a la implementación de mecanismos de alerta temprana que permiten identificar estudiantes en riesgo académico y brindar un acompañamiento oportuno, generando un impacto directo en la mejora del rendimiento académico.

CAPÍTULO IV. ANÁLISIS Y DISCUSIÓN DE RESULTADOS

Una vez culminada la presente investigación y tras la implementación del sistema de información web en la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado, se observaron mejoras significativas en la eficiencia de diversos procesos académicos, especialmente en las labores realizadas por los docentes. El tiempo requerido para el registro de asistencia por clase se redujo en un 9%, mientras que la creación de actividades experimentó una disminución del 4%. Asimismo, el tiempo destinado a la evaluación de actividades se redujo en un 11%, y el registro y publicación de notas mostró una mejora notable con una reducción del 51%.

Además, se logró eliminar en un 100% los casos que comprometían la fiabilidad del proceso evaluativo de los estudiantes. Igualmente, se erradicaron al 100% los retrasos en la publicación de notas finales y en la presentación de actas de notas, los cuales afectaban la transparencia de los procesos.

En cuanto a la calidad del software desarrollado, la evaluación realizada por expertos arrojó un promedio general del 90%, con los siguientes niveles por característica: 100% en adecuación funcional, 87% en eficiencia de desempeño, 93% en usabilidad, 80% en fiabilidad, 93% en seguridad, 80% en mantenibilidad, y 93% en portabilidad. Estos valores reflejan un alto grado de conformidad con los estándares de calidad del software y validan la efectividad del sistema en su contexto de aplicación.

El contexto en el que se llevó a cabo este estudio está relacionado a la investigación desarrollada por Ortiz y Pingo [11], ambas comparten un contexto institucional similar, una escuela superior de formación artística, y persiguen el mismo objetivo: mejorar la gestión académica mediante la implementación de un sistema de información web. Los resultados coinciden en demostrar que la automatización de procesos reduce tiempos, mejora la disponibilidad de la información y contribuye a una gestión más ordenada y eficiente.

De la misma manera, la investigación de Diaz [9] se relaciona directamente con la presente investigación, ya que ambos estudios abordan el desarrollo de sistemas web académicos orientados a la automatización de procesos clave como la matrícula, el control de asistencia y la gestión de calificaciones. Asimismo, coinciden en el impacto positivo del sistema propuesto sobre la eficiencia institucional y la mejora de la experiencia docente y estudiantil.

En el estudio de Enríquez [12] se evidencia que un sistema actualizado no solo reduce significativamente los tiempos de ejecución y generación de reportes, sino que también mejora la aceptación por parte de los usuarios, lo cual es coherente con los resultados obtenidos en esta investigación, donde también se registraron mejoras en la eficiencia y percepción del software.

De forma complementaria, el trabajo de Aliaga y Álvarez [13] destaca la importancia de realizar un análisis adecuado del problema y de aplicar instrumentos de medición confiables, como encuestas, para evaluar la funcionalidad y el impacto del sistema implementado, aspecto que también ha sido considerado en la presente tesis.

Por su parte, el estudio de Rodas y Cárdenas [10] refuerza la necesidad de migrar de procesos manuales a digitales para reducir la insatisfacción de los usuarios y evitar errores en la gestión académica, problemática que también motivó el desarrollo del sistema propuesto.

Finalmente, la investigación de Pérez [14] demuestra que la digitalización de procesos académicos como el registro de calificaciones puede reducir el tiempo de gestión en más del 87%, validando que un sistema correctamente implementado no solo optimiza la eficiencia, sino que también cumple con criterios de efectividad y satisfacción. En conjunto, estos antecedentes fortalecen el fundamento teórico de la presente investigación, respaldando la idea de que los sistemas de información académicos bien diseñados tienen un impacto directo y positivo en la mejora de la gestión académica.

En síntesis, los estudios revisados evidencian una problemática común en las instituciones educativas: la deficiente gestión académica causada por el uso de procesos manuales, el almacenamiento físico de información y la falta de integración entre actividades administrativas y académicas. Las investigaciones analizadas demuestran que la implementación de sistemas de información académicos contribuye significativamente a optimizar tiempos, reducir errores, mejorar la satisfacción de los usuarios y fortalecer el control de procesos clave como la matrícula, la asistencia, la evaluación y la publicación de notas. Asimismo, resaltan la importancia de un análisis adecuado de la problemática, el uso de herramientas tecnológicas pertinentes y la evaluación sistemática del impacto del software desarrollado. Estos estudios respaldan la presente investigación, al confirmar que el desarrollo de un sistema de información académico web no solo es una solución viable, sino también una necesidad estratégica para mejorar la eficiencia, la fiabilidad y la calidad de los servicios educativos, especialmente en instituciones como la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

La implementación del sistema de información web permitió optimizar los tiempos operativos en los procesos de gestión académica, evidenciando una disminución promedio del 19% en actividades como el registro de asistencia, creación y evaluación de actividades, y publicación de calificativos. Esta mejora en eficiencia operativa demuestra una reducción significativa en la carga administrativa, validando la influencia positiva del sistema en la dimensión temporal del proceso académico.

El uso del sistema de información web incidió directamente en la mejora de la fiabilidad del proceso académico, al reducir la ocurrencia de errores de transcripción, inconsistencias en los datos y duplicidad de información. La centralización de los registros en una única plataforma, con mecanismos de validación, integridad y seguridad, incrementó la precisión y confiabilidad de la información académica gestionada.

La implementación del sistema contribuyó a mejorar los niveles de transparencia institucional al proporcionar acceso oportuno y uniforme a la información académica por parte de todos los actores involucrados. La trazabilidad de las operaciones, la estandarización de criterios de evaluación y la disponibilidad de reportes actualizados fortalecieron los mecanismos de control, rendición de cuentas y equidad en los procesos educativos.

El sistema de información web fue desarrollado utilizando el enfoque ágil Kanban, lo que permitió una gestión iterativa, flexible y adaptada a los requerimientos funcionales y no funcionales identificados durante la investigación. Esta metodología facilitó el seguimiento continuo del progreso, la priorización de tareas críticas y la retroalimentación constante con los usuarios, lo que se tradujo en un producto funcional alineado a necesidades reales.

Durante la implementación del sistema de información web, se identificaron obstáculos como la resistencia al cambio por parte de los usuarios, dificultades iniciales en el manejo del sistema y limitaciones técnicas relacionadas con el uso de un servidor local. Para mitigar estos problemas, se ejecutaron capacitaciones dirigidas, se elaboró documentación funcional accesible y se fortaleció el soporte técnico. Estas acciones facilitaron una adopción progresiva, promoviendo la apropiación del sistema por parte de la comunidad educativa y asegurando su operatividad en el contexto institucional.

5.2. RECOMENDACIONES

Considerando que actualmente el sistema de información académico opera en un servidor físico, el cual está expuesto a interrupciones por cortes de energía eléctrica, se recomienda migrar a una infraestructura en la nube (como Amazon Web Services o Mirosoft Azure) o contratar servicios de hosting con alta disponibilidad, respaldo energético y soporte técnico especializado, además es recomendable una actualización continua de librerías y dependencias, con el fin de prevenir vulnerabilidades comunes en sistemas web.

Se recomienda que la institución designe o capacite a una persona con conocimientos técnicos y funcionales sobre el sistema de información implementado, que actúe como soporte interno para los usuarios. Esta figura será clave para resolver dudas, atender incidencias básicas y acompañar a los docentes, estudiantes y administrativos en el uso adecuado del sistema. Contar con un responsable de soporte garantiza un mayor aprovechamiento de las funcionalidades del sistema, previene su desuso por falta de orientación y contribuye a la sostenibilidad del proyecto a largo plazo.

Se recomienda establecer un proceso continuo de análisis y levantamiento de nuevos requerimientos funcionales y no funcionales, con el fin de asegurar que el sistema académico se mantenga alineado con las necesidades cambiantes de la institución.

REFERENCIAS BIBLIOGRÁFICAS

[1] UNESCO, «Gestión, monitoreo y evaluación de la educación | UNESCO». Accedido: 3 de marzo de 2024. [En línea]. Disponible en: https://www.unesco.org/es/education-management

[2] M. Arana, K. Unmdp, Bianculli, E. Unmdp, Uriondo, y Unmdo, «Gestión académica y estrategias para la retención de la matrícula: Programas de Tutorías en la universidad pública», *Revista Gestão Universitária na América Latina - GUAL*, vol. 2, jun. 2012, doi: 10.5007/1983-4535.2009v2n1p13.

[3] UNESCO, «Gestión de la educación en América Latina y el Caribe: ¿Vamos por un buen camino? - UNESCO Biblioteca Digital». Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://unesdoc.unesco.org/ark:/48223/pf0000149269

[4] G. Palmeros, G. Medina, y M. López, «La Gestión Educativa en el Escenario Latinoamericano: Avances y Retos», Red AGE. Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://www.redage.org/publicaciones/la-gestion-educativa-en-el-escenario-latinoamericano-avances-y-retos

[5] P. Llaque, «Los retos para la gestión y la docencia en la educación superior de América Latina». Accedido: 3 de marzo de 2024. [En línea]. Disponible en: https://www.academia.edu/104906627/Los_retos_para_la_gesti%C3%B3n_y_la_docencia_en_la_educaci%C3%B3n_superior_de_Am%C3%A9rica_Latina?uc-g-sw=32908202

[6] J. E. Espino Wuffarden *et al.*, «El desarrollo de la calidad educativa en educación superior universitaria: Revisión sistemática 2019-2023», *Comuni@cción*, vol. 14, n.º 4, pp. 348-359, oct. 2023, doi: 10.33595/2226-1478.14.4.876.

[7] D. Surco, «Gestión Academica y Desempeño Docente, según los estudiantes de una universidad privada en Lima, Perú | Industrial Data». Accedido: 3 de marzo de 2024. [En línea]. Disponible en: https://revistasinvestigacion.unmsm.edu.pe/index.php/idata/article/view/14914

[8] Y. A. Huapaya Capcha, «Gestión por procesos hacia la calidad educativa en el Perú», *Revista Arbitrada Interdisciplinaria Koinonía*, vol. 4, n.º 8, pp. 243-261, 2019.

[9] Diaz Fernandez, Jainer Manuel, «Sistema de información para el proceso de matriculas, control de asistencias y gestión de notas académicas en la institución educativa departamental integrada Antonio Nariño del municipio de Apulo.», Thesis, Universidad Piloto de Colombia, 2021. Accedido: 8 de febrero de 2024. [En línea]. Disponible en: http://repository.unipiloto.edu.co/handle/20.500.12277/10465

[10] J. Rodas Silva y J. Cárdenas Cobo, «Sistemas de Gestión Digital para mejorar los procesos académicos en instituciones educativas», *Universidad, Ciencia y Tecnología*, vol. 18, n.º 73, pp. 143-155, dic. 2014.

[11] S. E. Ortiz Ibañez y J. N. Pingo Ayala, «Implementación de un portal web y sistema de información para la gestión académica de la Escuela Superior de Formación Artística Pública "Ernesto López Mindreau" - Chiclayo – 2016.», 2019, Accedido: 26 de diciembre de 2023. [En línea]. Disponible en: http://repositorio.unprg.edu.pe/handle/20.500.12893/4278

[12] E. Díaz y E. Iván, «"sistema de información web y su mejora en la gestión académica del colegio privado hans kelsen del distrito de florencia de mora-trujillo"», sep. 2016. Accedido: 30 de abril de 2024. [En línea]. Disponible en: https://www.semanticscholar.org/paper/%E2%80%9Csistema-de-informaci%C3%B3n-web-y-su-mejora-en-la-del-D%C3%ADaz-Iv%C3%A1n/0d07fd2fd909abfb94ebc332195326d6146dedd5

[13] I. D. Aliaga Cabrera y E. M. Álvarez Sarmiento, «Implementación de un Sistema Académico Web y su influencia para mejorar el proceso de matrículas en la I.E. Túpac Amaru II N° 16498 - Huarango - San Ignacio - Cajamarca», *Universidad Privada Antonio Guillermo Urrelo*, sep. 2023, Accedido: 26 de diciembre de 2023. [En línea]. Disponible en: http://repositorio.upagu.edu.pe/handle/UPAGU/2912

[14] W. R. Pérez Estrada, «Desarrollo de un sistema de registro de calificaciones para mejorar el proceso de control y seguimiento de la evaluación académica en la facultad de ingeniería de la Universidad Nacional de Cajamarca», *Universidad Nacional de Cajamarca*, 2017, Accedido: 26 de diciembre de 2023. [En línea]. Disponible en: http://repositorio.unc.edu.pe/handle/20.500.14074/1146

[15] A. Hockley, J. van der Horst Jansen, M. S. Laughner, y R. White, Eds., «Academic management», en *From Teacher to Manager: Managing Language Teaching*

Organizations, en Cambridge Educational Management. , Cambridge: Cambridge University Press, 2008, pp. 201-232. doi: 10.1017/9781009406505.009.

[16] OECD, *Synergies for Better Learning: An International Perspective on Evaluation and Assessment*. en OECD Reviews of Evaluation and Assessment in Education. OECD, 2013. doi: 10.1787/9789264190658-en.

[17] F. Martínez Riso, «Los indicadores como herramientas para la evaluación de la calidad de los sistemas educativos». Accedido: 9 de junio de 2024. [En línea]. Disponible en: https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-109X2010000200004

[18] M. E. M. Galván, «Indicadores De Gestion Universitaria en Las Instituciones De Educacion Superior», ene. 2014, Accedido: 9 de junio de 2024. [En línea]. Disponible en: https://www.academia.edu/92894464/Indicadores_De_Gestion_Universitaria_en_Las_Instituciones_De_Educacion_Superior

[19] J. A. Tuesta Panduro *et al.*, «Percepción sobre la transparencia en la web de universidades peruanas», *Conrado*, vol. 17, n.º 81, pp. 28-34, ago. 2021.

[20] A. Buendía Espinosa, I. A. Salas Durazo, A. Buendía Espinosa, y I. A. Salas Durazo, «Mirar la transparencia desde el discurso. Un acercamiento a las universidades públicas mexicanas», *Gestión y política pública*, vol. 29, n.º 1, pp. 3-35, jun. 2020, doi: 10.29265/gypp.v29i1.655.

[21] S. M. V. Andrade, «La gestión académica del modelo pedagógico sociocrítico en la Institución Educativa: Rol del docente», 2018.

[22] R. Peiró, «Sistema de información», Economipedia. Accedido: 12 de febrero de 2024. [En línea]. Disponible en: https://economipedia.com/definiciones/sistema-de-informacion.html

[23] A. H. Trasobares, «Los sistemas de información: evolución y desarrollo», *Proyecto social: Revista de relaciones laborales*, n.º 10, pp. 149-165, 2003.

[24] N. E. Mosquera Arboleda, «Sistema de información Web», 2009, Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://hdl.handle.net/11059/1325

[25] S. G. Pérez Ibarra, J. R. Quispe, F. F. Mullicundo, y D. A. Lamas, «Herramientas y tecnologías para el desarrollo web desde el FrontEnd al BackEnd», presentado en XXIII Workshop de Investigadores en Ciencias de la Computación (WICC 2021, Chilecito, La Rioja), 2021. Accedido: 7 de mayo de 2024. [En línea]. Disponible en: http://sedici.unlp.edu.ar/handle/10915/120476

[26] S. Shraddha, S. Harsh, L. Nisha, S. Agam, y J. Devansh, «A Comprehensive Review of Next.js Technology: Advancements, Features, and Applications by Shraddha Srivastava, Harsh Shukla, Nisha Landge, Agam Srivastava, Devansh Jindal :: SSRN». Accedido: 15 de septiembre de 2025. [En línea]. Disponible en: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4831070

[27] H. Edward, T. Guarda, A. Zambrano, y G. Ninahualpa, «Desarrollo backend para aplicaciones web, Servicios Web Restful: Node.js vs Spring Boot - ProQuest». Accedido: 30 de enero de 2025. [En línea]. Disponible en: https://www.proquest.com/openview/a78cfaa62708fd24f38ac8d1025050eb/1?pq-origsite=gscholar&cbl=10063

[28] DTS, «Qué es un sistema de gestión de bases de datos». Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://thedataschools.com/que-es/sistema-de-gestion-de-bases-de-datos/

[29] J. W. Krogh, «MySQL Workbench», en *MySQL 8 Query Performance Tuning: A Systematic Method for Improving Execution Speeds*, J. W. Krogh, Ed., Berkeley, CA: Apress, 2020, pp. 199-226. doi: 10.1007/978-1-4842-5584-1_11.

[30] Data Ladder, «La integración de datos explicada: Definición, tipos, proceso y herramientas - Data Ladder». Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://dataladder.com/es/la-integracion-de-datos-explicada-definicion-tipos-proceso-y-herramientas/

[31] A. Rudra, «¿Qué es la seguridad web? Buenas prácticas de seguridad de sitios web». Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://powerdmarc.com/es/web-security-website-security-explained/

[32] M. Moa, «Using JWT (JSON Web Tokens) to authorize users and protect API routes», Medium. Accedido: 30 de enero de 2025. [En línea]. Disponible en:

https://medium.com/@maison.moa/using-jwt-json-web-tokens-to-authorize-users-and-protect-api-routes-3e04a1453c3e

[33] AppMaster, «Escalabilidad y rendimiento: los pilares de una arquitectura de software exitosa | AppMaster». Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://appmaster.io/es/blog/arquitectura-de-software-de-escalabilidad-y-rendimiento

[34] AppMaster, «Crear un sitio web compatible con dispositivos móviles | AppMaster». Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://appmaster.io/es/blog/creacion-de-sitios-web-adaptados-a-dispositivos-moviles

[35] Y. Pinnola, «Sistemas de informacion, definición y conceptos. Origenes y elementos. - Studocu». Accedido: 8 de enero de 2024. [En línea]. Disponible en: https://www.studocu.com/es-ar/document/instituto-superior-de-formacion-docente-no-49-brandsen/sistemas-de-la-informacion/sistemas-de-informacion-definicion-y-conceptos-origenes-y-elementos/14136649

[36] M. A. Rigo Lemini, «Evaluación académica», *Perfiles educativos*, vol. 22, n.º 87, pp. 98-100, 2000.

[37] UNA, «Sistema de Información Académica». Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://www.cienciassociales.una.ac.cr/index.php/sistema-de-informacion-academica

[38] ISO25000, «ISO 25010». Accedido: 30 de abril de 2024. [En línea]. Disponible en: https://iso25000.com/index.php/normas-iso-25000/iso-25010

[39] E. G. Maida y J. Pacienzia, «Metodologías de desarrollo de software», 2015, Accedido: 30 de abril de 2024. [En línea]. Disponible en: https://repositorio.uca.edu.ar/handle/123456789/522

[40] J. Gaete *et al.*, «Enfoque de aplicación ágil con Serum, Lean y Kanban», *Ingeniare. Revista chilena de ingeniería*, vol. 29, n.º 1, pp. 141-157, mar. 2021, doi: 10.4067/S0718-33052021000100141.

[41] W. Zayat y O. Senvar, «Framework Study for Agile Software Development Via Scrum and Kanban». Accedido: 30 de enero de 2025. [En línea]. Disponible en: https://www.worldscientific.com/doi/epdf/10.1142/S0219877020300025

[42] H. Kniberg y M. Skarin, «Kanban and Scrum - Making the Most of Both», InfoQ. Accedido: 30 de enero de 2025. [En línea]. Disponible en: https://www.infoq.com/minibooks/kanban-scrum-minibook/

[43] N. Figuerola, «Kanban: su uso en el desarrollo de software».

[44] B. E. Libera Bonilla, «Impacto, impacto social y evaluación del impacto», *Revista Cubana de Información en Ciencias de la Salud*, vol. 15, n.º 3, p. 7, 2007.

[45] M. Á. Mallar, «La Gestión Por Procesos: Un Enfoque De Gestión Eficiente», *Revista Científica «Visión de Futuro»*, vol. 13, n.º 1, 2010, Accedido: 7 de julio de 2024. [En línea]. Disponible en: https://www.redalyc.org/articulo.oa?id=357935475004

[46] A. Flórez, Y. Ramírez, L. Baleta, y E. Noro, «Gestión académica en instituciones de educación superior: reflexiones y experiencias exitosas», ResearchGate. Accedido: 4 de agosto de 2025. [En línea]. Disponible en: https://www.researchgate.net/publication/334175407_Gestion_academica_en_instituciones_de_educacion_superior_reflexiones_y_experiencias_exitosas

[47] A.-R. Pacios, M. Vianello-Osti, y B. Rodrí-guez-Bravo, «Transparencia y acceso a la información sobre los proyectos de investigación en las universidades públicas españolas», *Profesional de la información*, vol. 25, n.º 5, Art. n.º 5, oct. 2016.

[48] M. P. Ascue, «Calidad de datos en sistemas de gestión académica universitaria basado en ISO/IEC 25012», *Interfases*, n.º 014, Art. n.º 014, dic. 2021, doi: 10.26439/interfases2021.n014.5393.

[49] K. H. Samperio-Gómez y M. L. Cáceres-Mesa, «La gestión escolar y su impacto en la función directiva. Reflexiones teórico-metodológicas», *Revista Metropolitana de Ciencias Aplicadas*, vol. 7, n.º Suplemento 1, Art. n.º Suplemento 1, feb. 2024, doi: 10.62452/rs29bw18.

[50] M. A. Ciudad y R. M. Rivalles, «Uso de modelos de calidad en las mypes productoras de software de Lima», *Ingeniería Industrial*, n.º 37, pp. 81-99, 2019, doi: https://doi.org/10.26439/ing.ind2019.n037.4543.

[51] CISET, «Software - Concepto y tipos». Accedido: 7 de julio de 2024. [En línea]. Disponible en: https://www.ciset.es/glosario/480-software-concepto-y-tipos

[52] Códigos Informáticos, «¿Qué es la tecnologia web? Las más actuales y relevantes en 2025 | codigos informaticos», INFORMÁTICA Y TECNOLOGÍA. Accedido: 4 de agosto de 2025. [En línea]. Disponible en: https://www.codigosinformaticos.com/que-es-la-tecnologia-web/

[53] Conexión Esan, «Cómo identificar a los stakeholders de tu organización | Conexión ESAN». Accedido: 7 de julio de 2024. [En línea]. Disponible en: https://www.esan.edu.pe/conexion-esan/como-identificar-a-los-stakeholders-de-tu-organizacion

[54] Atlassian, «Backlog del producto: qué es y cómo crearlo», Atlassian. Accedido: 7 de julio de 2024. [En línea]. Disponible en: https://www.atlassian.com/es/agile/scrum/backlogs

[55] D. Rodríguez, «Investigación aplicada: características, definición, ejemplos», Lifeder. Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://www.lifeder.com/investigacion-aplicada/

[56] D. Rodríguez, «Investigación explicativa: características, técnicas, ejemplos», Lifeder. Accedido: 7 de mayo de 2024. [En línea]. Disponible en: https://www.lifeder.com/investigacion-explicativa/

[57] C. Ramos-Galarza, «Diseños de investigación experimental», *CienciAmérica*, vol. 10, n.º 1, pp. 1-7, feb. 2021, doi: 10.33210/ca.v10i1.356.

[58] J. L. Abreu, «El Método de la Investigación. | Revista Daena: International Journal of Good Conscience | EBSCOhost». Accedido: 8 de mayo de 2024. [En línea]. Disponible en: https://openurl.ebsco.com/contentitem/gcd:116281281?sid=ebsco:plink:crawler&id=ebsco:gcd:116281281

[59] D. J. Anderson, *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press, 2010.

[60] B. Niebel, «Análisis de tiempos de trabajo - Mejoramiento del proceso de ensamble en la línea chasis para a». Accedido: 20 de septiembre de 2025. [En línea]. Disponible en: https://1library.co/article/an%C3%A1lisis-tiempos-trabajo-mejoramiento-proceso-ensamble-l%C3%ADnea-chasis.y8grg4wz

ANEXOS

ANEXO 1: MATRIZ DE OPERACIONALIZACIÓN DE VARIABLES

VARIABLE	DEFINICIÓN CONCEPTUAL	DIMENSIÓN	INDICADOR	TÉCNICA/INSTRUMENTO
SISTEMA DE INFORMACIÓN WEB	Sistema informático diseñado para gestionar y suministrar información a través de internet. Para las métricas de calidad de este sistema se tendrá en cuenta la ISO 25000, se empleará la encuesta como técnica y el cuestionario como instrumento.	Calidad de software	▪ Adecuación funcional ▪ Eficiencia de desempeño ▪ Usabilidad ▪ Fiabilidad ▪ Seguridad ▪ Mantenibilidad ▪ Portabilidad	Encuesta/Cuestionario
GESTIÓN ACADÉMICA	Proceso que se realiza en una institución educativa para mejorar los programas y proyectos educativos, así como de los procesos pedagógicos, curriculares y de evaluación. Para las métricas de esta gestión se empleará técnicas como la observación y la revisión documentaria con los instrumentos pertinentes.	Tiempo	▪ Tiempo en el registro de asistencias por clase. ▪ Tiempo en la creación de actividades. ▪ Tiempo en la evaluación de actividades. ▪ Tiempo en el registro y publicación de notas.	Observación/Ficha de observación
		Fiabilidad	▪ Nivel de fiabilidad en el cálculo de notas. ▪ Nivel de fiabilidad en el cálculo de porcentajes de asistencia.	Revisión documentaria/Ficha de análisis documental
		Trasparencia	▪ Publicación de notas en las fechas establecidas. ▪ Presentación de actas en el tiempo establecido.	Observación/Ficha de cotejo

ANEXO 2: MATRIZ DE COSISTENCIA

PROBLEMA GENERAL	OBJETIVO GENERAL	HIPÓTESIS GENERAL	VARIABLES
¿Cuál es el impacto de la implementación de un sistema de información web en la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado?	Evaluar el impacto de la implementación de un sistema de información web en la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.	Un sistema de Información web impacta de manera positiva en la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.	**Variable independiente:** Sistema de información web. **Variable dependiente:** Gestión académica.
PROBLEMAS ESPECÍFICOS	**OBJETIVOS ESPECÍFICOS**	**HIPÓTESIS ESPECÍFICAS**	
▪ ¿Cuál es el impacto de la implementación de un sistema de información web en el tiempo que toma realizar tareas pertenecientes a la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado? ▪ ¿Cuál es el impacto de la implementación de un sistema de información web en la fiabilidad de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado? ▪ ¿Cuál es el impacto de la implementación de un sistema de información web en la transparencia de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado?	▪ Determinar el impacto del uso de un sistema de información web en el tiempo para realizar tareas de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado. ▪ Determinar el impacto del uso de un sistema de información web en la fiabilidad de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado. ▪ Determinar el impacto del uso de un sistema de información web en la transparencia de la gestión académica en la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.	▪ El uso de un sistema de información web disminuye el tiempo que toma realizar tareas pertenecientes a la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado. ▪ El uso de un sistema de información web mejora la fiabilidad de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado. ▪ El uso de un sistema de información web refuerza la transparencia de la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado.	

ANEXO 3: CUESTIONARIO – VARIABLE INDEPENDIENTE

"IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024"

CUESTIONARIO APLICADO A EXPERTOS EN EL DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN

Objetivo

El siguiente cuestionario está diseñado para medir la dimensión de "calidad" de un sistema de información web utilizado en la gestión académica de la Escuela de Formación Artística Pública Mario Urteaga Alvarado, ubicada en la ciudad de Cajamarca. Esta evaluación se basa en los siguientes indicadores: adecuación funcional, eficiencia del desempeño, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad.

Consentimiento informado

La participación en esta investigación es voluntaria; asimismo, la información que recolecta será confidencial, y anónima y las respuestas proporcionadas no se usarán para ningún otro propósito fuera de la investigación.

Instrucciones, marque con una "x" en el recuadro que más se acerque a su respuesta, considerando la escala de 1 al 5:

Descripción	Valor cualitativo	Valor cuantitativo
Valores de los ítems del cuestionario con un rango cerrado de preguntas cerradas.	Totalmente en desacuerdo	1
	En desacuerdo	2
	Indiferente	3
	De acuerdo	4
	Totalmente de acuerdo	5

Fig. 96 Cuestionario de la dimensión calidad – hoja 1

Preguntas		Valor cuantitativo				
		1	2	3	4	5
Adecuación funcional						
1	El sistema cumple con los requisitos funcionales especificados, permitiendo a los usuarios realizar todas las tareas necesarias de manera eficiente.					x
2	La interfaz de usuario es intuitiva y accesible, facilitando a los usuarios navegar y utilizar todas las características del sistema sin necesidad de capacitación extensa.					x
Eficiencia de desempeño						
3	El sistema puede manejar un número elevado de usuarios concurrentes sin degradar su rendimiento, manteniendo la estabilidad y la rapidez en las operaciones.					x
Usabilidad						
4	Los usuarios pueden completar tareas específicas en el sistema sin confusiones ni errores, existiendo comodidad al utilizar la interfaz sin necesidad de asistencia constante.				x	
5	La interfaz proporciona una navegación clara y coherente, con etiquetas y elementos visuales que ayudan a los usuarios a comprender rápidamente cómo interactuar con el sistema.					x
Fiabilidad						
6	Se han implementado mecanismos de respaldo y recuperación de datos para garantizar la integridad y la disponibilidad de la información en caso de fallos o interrupciones.				x	
Seguridad						
7	Se han implementado controles de acceso adecuados, garantizando que solo los usuarios autorizados puedan acceder a funciones y datos sensibles del sistema					x
8	Se utilizan protocolos de cifrado para proteger la transmisión de datos sensibles entre el cliente y el servidor, así como para almacenar información crítica en la base de datos.					x
Mantenibilidad						
9	El código fuente está documentado adecuadamente.				x	
10	Se utilizan patrones de diseño y mejores prácticas de codificación.				x	
Portabilidad						
11	El sistema se puede desplegar en diferentes entornos sin modificaciones significativas.					x
12	Se utilizan tecnologías y estándares abiertos para el desarrollo.					x

Fig. 97 Cuestionario de la dimensión calidad – hoja 2

ANEXO 4: FICHA DE OBSERVACIÓN DE LA DIMENSIÓN TIEMPO

A continuación, se muestra el instrumento ficha de observación usado para recolectar datos de la dimensión tiempo, el proceso se ha llevado a cabo en un solo ciclo de observación aplicado a todos docentes, los cuales fueron considerados tanto en el pre como en el post test (ver Fig. 98).

Ficha de observación			
Título de la investigación: Impacto de un sistema de información web en la gestión académica de la Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado, Cajamarca 2024"		**Lugar:** Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado	
Objetivo: Registrar y analizar el tiempo que los docentes dedican a diferentes actividades dentro de su gestión académica, con el fin de identificar la distribución y uso del tiempo en su labor diaria.			
Observado por: Jhan Carlos Villanueva Vargas	**Fecha**: _____	**Herramienta**: cronómetro	
Docente/curso:			
Procesos	**Actividades**	**Tiempo prom. (min.)**	
P1. Registro de asistencias por clase.	Obtención del listado		
	Llamado o revisión visual		
	Marcación en papel o sistema digital		
	Notificación sobre el estado de asistencia a los estudiantes		
P2. Creación de actividades.	Definir propósito pedagógico y contenido de la tarea		
	Redacción de las instrucciones detalladas		
	Socialización de materiales de apoyo		
	Explicación de la tarea		
P3. Evaluación de actividades	Recepción de tareas		
	Lectura y análisis del contenido		
	Asignación y registro de puntajes		
	Socialización de notas a los estudiantes		
P4. Registro y publicación de notas	Consolidación de calificaciones por criterios o rubros establecidos		
	Registro de notas		
	Verificación de exactitud y descarte de errores		
	Socialización de resultados		

Fig. 98 Ficha de observación de la dimensión tiempo

ANEXO 5: FICHA DE ANÁLISIS DOCUMENTAL DE LA DIMENSIÓN FIABILIDAD

Seguidamente se muestra la ficha de análisis documental empleada para obtener datos que permitan catalogar la fiabilidad en procesos relacionados al cálculo de notas de los estudiantes, así como el porcentaje de asistencia que determina que un estudiante pueda permanecer activo en un curso (Ver Fig. 99).

Ficha de análisis documental		
Objetivo: Revisar y analizar los registros de calificaciones de los estudiantes con el fin de identificar posibles errores en el cálculo de promedios, inconsistencias en la asignación de notas o faltantes en los registros, para contribuir a la veracidad del proceso evaluativo.		
Dimensión: Fiabilidad	**Tipo de medición:** ☐ Pre test ☐ Post test	
Lugar: Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado		
Observado por: Jhan Carlos Villanueva Vargas	**Fecha:**	
Nombre del documento:		
Tipo de documento:		
Ubicación:		
Aporte de la información contenida a la investigación:		
Observaciones:		

Fig. 99 Ficha de análisis documental de la dimensión fiabilidad

ANEXO 6: FICHA DE COTEJO DE LA DIMENSIÓN TRANSPARENCIA

Continuamente se muestra la ficha de cotejo utilizada para obtener datos relacionados a la divergencia durante la entrega de notas de los estudiantes y la entrega de actas de estas (Ver Fig. 100).

Ficha de cotejo												
Objetivo: Registrar y verificar el tiempo transcurrido entre la finalización del periodo evaluativo y la ejecución de actividades específicas por parte del docente, con el fin de asegurar la transparencia del proceso, prevenir posibles manipulaciones indebidas y respaldar el cumplimiento de los plazos establecidos institucionalmente.												
Dimensión: Transparencia											**Tipo de medición/actividad:** ☐ Pre test/manual ☐ Post test/sistematizada	
Lugar: Escuela Superior de Formación Artística Pública Mario Urteaga Alvarado												
Observado por: Jhan Carlos Villanueva Vargas							**Fecha:**				**Periodo:**	
N°	**Indicadores**	**Cursos**										**Promedio**
		C1	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	
1	Número de días de retraso con respecto a la fecha de publicación de notas finales.											
2	Número de días de retraso con respecto a la fecha de presentación de actas de notas.											

Fig. 100 Ficha de cotejo de la dimensión transparencia

ANEXO 7: VALIDACIÓN DE INSTRUMENTOS

"IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024"

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

TESIS: "IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024"

1. **INFORMACIÓN SOBRE EL EXPERTO**
   - 1.1. **Experto:** Julio César Guailupo Alvarez
   - 1.2. **Especialidad:** Lic. en Estadística
   - 1.3. **Cargo actual:** Director de Sistema Administrativo
   - 1.4. **Grado académico:** Magister en Salud Pública
   - 1.5. **Institución:** Dirección Regional de Salud

2. **INFORMACIÓN SOBRE EL INSTRUMENTO A EVALUAR**
   - 2.1. **Tipo de instrumento:** Cuestionario – dimensión calidad (software)
   - 2.2. **Lugar y fecha:** Cajamarca 03/10/24

3. **TABLA DE VALORACIÓN POR EVIDENCIAS**

N°	Evidencias	Valoración					
		5	4	3	2	1	0
1	Pertinencia de indicadores.	x					
2	Formulación con lenguaje adecuado.	x					
3	Adecuado para los sujetos en estudio.	x					
4	Facilita la prueba de hipótesis.	x					
5	Suficiencia para medir variable.	x					
6	Facilita la interpretación del instrumento.	x					
7	Acorde al avance de la ciencia y la tecnología.	x					
8	Expresado en hechos perceptibles.	x					
9	Tiene secuencia lógica.	x					
10	Basado en aspectos teóricos.	x					
Total		50					

Coeficiente de valoración porcentual: c= 100%

4. **OBSERVACIONES Y/O RECOMENDACIONES**

Mg. Julio César Guailupo Alvarez
COESPE N° 254

Fig. 101 Ficha para validación del cuestionario

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

TESIS: "IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024"

1. INFORMACIÓN SOBRE EL EXPERTO

- **1.1. Experto:** Julio César Guailupo Alvarez
- **1.2. Especialidad:** Lic. en Estadística
- **1.3. Cargo actual:** Director de Sistema Administrativo
- **1.4. Grado académico:** Magister en Salud Pública
- **1.5. Institución:** Dirección Regional de Salud

2. INFORMACIÓN SOBRE EL INSTRUMENTO A EVALUAR

- **2.1. Tipo de instrumento:** Ficha de observación – dimensión tiempo
- **2.2. Lugar y fecha:** Cajamarca 03/10/24

3. TABLA DE VALORACIÓN POR EVIDENCIAS

N°	Evidencias	Valoración					
		5	4	3	2	1	0
1	Pertinencia de indicadores.	x					
2	Formulación con lenguaje adecuado.	x					
3	Adecuado para los sujetos en estudio.	x					
4	Facilita la prueba de hipótesis.	x					
5	Suficiencia para medir variable.	x					
6	Facilita la interpretación del instrumento.	x					
7	Acorde al avance de la ciencia y la tecnología.	x					
8	Expresado en hechos perceptibles.	x					
9	Tiene secuencia lógica.	x					
10	Basado en aspectos teóricos.	x					
Total		50					

Coeficiente de valoración porcentual: c= 100%

4. OBSERVACIONES Y/O RECOMENDACIONES

Fig. 102 Ficha para validación de la ficha de observación

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

TESIS: "IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024"

1. INFORMACIÓN SOBRE EL EXPERTO

- **1.1. Experto:** Julio César Guailupo Alvarez
- **1.2. Especialidad:** Lic. en Estadística
- **1.3. Cargo actual:** Director de Sistema Administrativo
- **1.4. Grado académico:** Magister en Salud Pública
- **4.2. Institución:** Dirección Regional de Salud

2. INFORMACIÓN SOBRE EL INSTRUMENTO A EVALUAR

- **2.1. Tipo de instrumento:** Ficha de análisis documental – dimensión fiabilidad
- **2.2. Lugar y fecha:** Cajamarca 03/10/24

3. TABLA DE VALORACIÓN POR EVIDENCIAS

N°	Evidencias	Valoración					
		5	4	3	2	1	0
1	Pertinencia de indicadores.	x					
2	Formulación con lenguaje adecuado.	x					
3	Adecuado para los sujetos en estudio.	x					
4	Facilita la prueba de hipótesis.	x					
5	Suficiencia para medir variable.	x					
6	Facilita la interpretación del instrumento.	x					
7	Acorde al avance de la ciencia y la tecnología.	x					
8	Expresado en hechos perceptibles.	x					
9	Tiene secuencia lógica.	x					
10	Basado en aspectos teóricos.	x					
Total		50					

Coeficiente de valoración porcentual: c= 100%

4. OBSERVACIONES Y/O RECOMENDACIONES

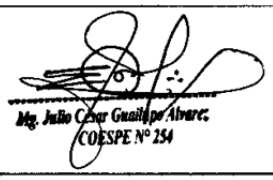

Fig. 103 Ficha para validación de la ficha de análisis documental

FICHA PARA VALIDACIÓN DEL INSTRUMENTO

TESIS: "IMPACTO DE UN SISTEMA DE INFORMACIÓN WEB EN LA GESTIÓN ACADÉMICA DE LA ESCUELA SUPERIOR DE FORMACIÓN ARTÍSTICA PÚBLICA MARIO URTEAGA ALVARADO, CAJAMARCA 2024"

1. INFORMACIÓN SOBRE EL EXPERTO

- **1.1. Experto:** Julio César Guailupo Alvarez
- **1.2. Especialidad:** Lic. en Estadística
- **1.3. Cargo actual:** Director de Sistema Administrativo
- **1.4. Grado académico:** Magister en Salud Pública
- **1.5. Institución:** Dirección Regional de Salud

2. INFORMACIÓN SOBRE EL INSTRUMENTO A EVALUAR

- **2.1. Tipo de instrumento:** Ficha de cotejo – dimensión transparencia
- **2.2. Lugar y fecha:** Cajamarca 03/10/24

3. TABLA DE VALORACIÓN POR EVIDENCIAS

N°	Evidencias	Valoración					
		5	4	3	2	1	0
1	Pertinencia de indicadores.	x					
2	Formulación con lenguaje adecuado.	x					
3	Adecuado para los sujetos en estudio.	x					
4	Facilita la prueba de hipótesis.	x					
5	Suficiencia para medir variable.	x					
6	Facilita la interpretación del instrumento.	x					
7	Acorde al avance de la ciencia y la tecnología.	x					
8	Expresado en hechos perceptibles.	x					
9	Tiene secuencia lógica.	x					
10	Basado en aspectos teóricos.	x					
Total		50					

Coeficiente de valoración porcentual: c= 100%

4. OBSERVACIONES Y/O RECOMENDACIONES

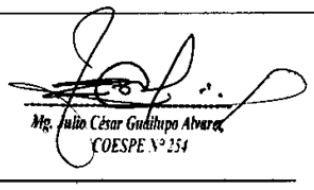

Fig. 104 Ficha para validación de la ficha de cotejo

ANEXO 8: CÁLCULO DE LA CONFIABILIDAD DEL CUESTIONARIO PARA MEDIR LA CALIDAD DEL SOFTWARE

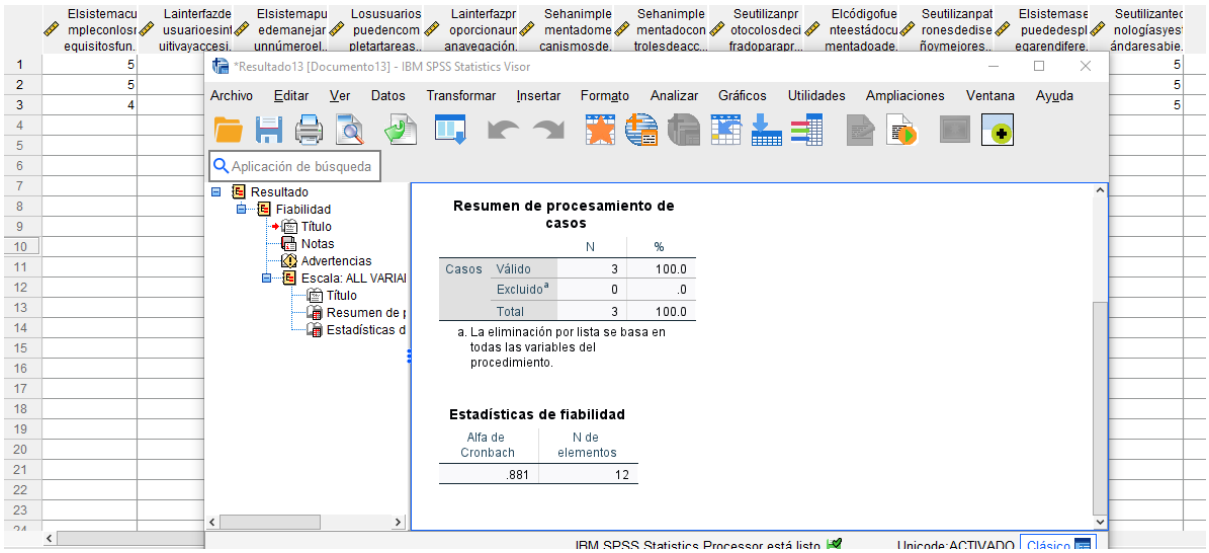

Fig. 105 Coeficiente de fiabilidad para el cuestionario